WEB APPLICATION FOR INTELLIGENT NIGHT SURVEILLANCE

By

Capt Arslan Mustafa

Maj Awais Masood

Capt Abdul Ahad

Capt Muhammad Zohaib Aftab

Supervised by:

Lt Col Khawir Mehmood

Submitted to the faculty of Department of Computer Software Engineering,
Military College of Signals, National University of Sciences and Technology, Islamabad,
in partial fulfillment for the requirements of B.E Degree in Software Engineering.

June 2024

In the name of ALLAH, the Most benevolent, the Most Courteous

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report

"Web Application for Intelligent Night Surveillance"

is carried out by

Capt Arslan Mustafa

Maj Awais Masood

Capt Abdul Ahad

Capt Muhammad Zohaib Aftab

under my supervision and that in my judgement, it is fully ample, in scope and excellence, for the degree of Bachelor of Software Engineering in Military College of Signals, National University of Sciences and Technology (NUST), Islamabad.

Approved by

Supervisor

Lt Col Khawir Mehmood

Date: _____

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

ACKNOWLEDGEMENTS

Allah Subhan'Wa'Tala is the sole guidance in all domains.

Our parents, families, colleagues and most of all supervisor, Lt Col Khawir Mehmood under whose guidance

The group members through all adversities worked steadfastly.

Plagiarism Certificate (Turnitin Report)

This thesis has ____ similarity index. Turnitin report endorsed by Supervisor is attached.

_____

Capt Arslan Mustafa

00000359433

_____

Maj Awais Masood

000003594420

_____

Capt Abdul Ahad

00000359434

_____

Capt Muhammad Zohaib Aftab

00000359438

_____

Signature of Supervisor

ABSTRACT

The Intelligent Night Surveillance System is an all-in-one solution for night monitoring designed with the latest technologies to supplement surveillance capabilities in low-light conditions. It seamlessly integrates advanced techniques of motion detection and object detection and classification, enabled with real-time monitoring and alerting features. By means of the machine learning models and open-source technologies, the fine identification and classification of different objects, including humans, animals, and vehicles, are allowed within the surveillance feed. The user-friendly web interface allows remote access, quick instant alerts, and efficient viewing of live surveillance feeds on multiple devices.

This makes sure that Ubuntu Server is installed to help in running these backend tasks smoothly and is integrated with the Raspberry Pi hardware so that it can cooperate with different types of surveillance cameras and video input formats. Examples include strong security mechanisms, such as secure communication protocols, for protecting sensitive information and retaining the surveillance footages. Moreover, the system is also designed for scalability; that is, it can be scaled to more cameras and, thus, more object classification. In general, the Intelligent Night Surveillance System provides an advanced and flexible solution for nighttime monitoring to meet the dynamic needs of surveillance scenarios with accurate and reliable performance.

Table of Contents

List of Figures

Chapter 1: Introduction

Night surveillance is a necessity for the safety and security arrangements within residential, commercial, and public environments. The surveillance under low light suffers from intrinsic difficulties due to the non-availability of ample visibility and the increased element of risk. As such, traditional surveillance systems often fail to offer effective monitoring in such environments, and the need for innovative solutions that can enhance nighttime security operations is emphasized.

With the advent of modern advanced technologies, including computer vision, machine learning, and the Internet of Things, intelligent night surveillance systems are certainly going to be developed. Undoubtedly, such systems heavily rely on high-end algorithms and hardware to improve their surveillance abilities during night times. INSS provides real-time situational awareness for proactively responding to potential threats and incidents through smart integration of features like motion detection, and the identification and classification of objects.

This thesis presents an approach to design, develop, and implement an Intelligent Night Surveillance System for tackling the challenges of nighttime monitoring. It integrates advanced motion detection algorithms and object recognition and classification to realize surveillance capabilities that are at once effective and efficient in low-light environments. It is incorporated with open-source technologies and a plethora of machine-learning models to help offer very accurate detection and identification of objects in the surveillance footage, which will cover humans, animals, and vehicles.

Further, this thesis will elaborate on the usability, scalability, and security issues of the INSS such issues as user interface design, and system architecture. In this light, the research is aimed at bringing to the limelight the potential applications and benefits accruing from the use of the system in improving nighttime security operations. The further in-depth analysis and evaluation, that will

follow, will allow this thesis to serve as a benchmark, expecting to find the improvement of the surveillance technology as well as solid solutions to all the nocturnal monitoring challenges.

1.1 Overview

The present-day digital era has drawn attention to the demand for smart surveillance solutions because it particularly applies to nighttime when visibility is not that high. Urban expansion, coupled with heightened reliance on surveillance systems, has brought about critical needs for enhancing security and observation capabilities through innovative approaches.

Since the last decade, the world has seen rapid growth in crime rate, especially during low-light conditions which made Intelligent Night Surveillance a challenge for security personnel. According to Security Magazine, there is a wide gap between crimes during the daytime and at nighttime. The table clearly states the difference.

Offense	At Night Percentage	During the Day Percentage
DWI/DUI	87%	13%
Murder & Negligent Manslaughter	65%	35%
Rape/Sexual Assault	59%	41%
Robbery	56%	44%
Aggravated Assault	54%	46%
Motor Vehicle Theft	51%	49%
Burglary	50%	50%
Property Crime	48%	52%
Simple Assault	47%	53%
Drug Violation	43%	57%
Larceny/Theft	40%	60%

Figure 1.1: *Table indicating crime rate in percentage in USA.*

Traditional Security measures have already proved ineffective if without the help of computer technology. Combining modern-day computing power with conventional security techniques can be an effective way to reduce such rapidly growing crime trends. This is the gap that the current thesis proposal for Intelligent Night Surveillance System (INSS) fills by taking the best and latest technology and using it to offer full solutions for surveillance at night.

1.2 Problem Statement

Pakistan is a third-world country with limited economic resources. Pakistan's industry is mostly comprised of small-scale home-based industries which have low revenue and profits. Security is a need of everybody may it be households, schools, firms, and government organizations. Till now, Night Surveillance Capability has only been implemented in hardware such as cameras which are very expensive whereas small firms and households cannot afford such solutions. The following are some highlights of the Night Surveillance Cameras available in the market.

1. Intelligent night surveillance has not been implemented using computer vision and machine learning and is mostly restricted to hardware (e.g. employing special sensors in high-end IR cameras).
2. Lack of Automation in the Alert and Response system.
3. The cost of each camera is such that small organizations and households with limited budgets cannot afford it.
4. Increased monitoring of the feed by Security personnel deployed in the control room can be stressful and tiring.
5. Human error can still pave the way for security incidents.

1.3 Proposed Solution

In addressing the challenges that have been highlighted in the problem statement, the proposed solution is the development and implementation of an inexpensive and efficient Intelligent Night Surveillance System (INSS) customized to the needs of small-scale, home-based industries, households, and other organizations with limited economic resources in Pakistan.

INSS will have intelligent surveillance capabilities at night, with image enhancement and machine learning technologies, to guarantee higher levels of accuracies in detecting, identifying, and classifying objects and activities down to the lowest illumination. Following are the key features of the proposed system: -

1. Affordability
2. Automation
3. User-Friendly Interface
4. Reliability
5. Scalability

1.4 Working Principle

The project is perfectly combining the image processing techniques with the machine learning algorithms to fulfill its goals. The system is divided into very distinctive modules in which each building block is intricately pieced together. The modular nature of the structure helps to easily integrate the vast variety of functionalities. This integrated approach looks at optimization of performance and efficacy in the due course. The list of modules is as under:

1. IR Camera
2. Raspberry Pi
    a. Motion Detection

b. Image Enhancement

c. Object Detection and Classification

d. Email Alert System

3. Web Application Interface

1.4.1 IR Camera The IR camera is the main input device in real-time video acquisition in low-illuminated environments. The infrared technology enables the camera to detect and capture images invisible to the naked eye and gives high definition in video feed under total darkness. The captured video feed is streamed continuously to the Raspberry Pi for processing.

1.4.2 Raspberry Pi Raspberry Pi is used as the central processing unit for the IR camera. A few modules are running, which are motion detection, image enhancement, object detection and classification, and email alerts, on the Raspberry Pi. The processing pipeline of the Raspberry Pi is as follows:

1.4.2.1 **Motion Detection**: The motion detection module works by detecting any movement in the video feed. This is achieved through the following steps:-

1. Frame Capture: Consecutive frames are captured from the video feed.
2. Frame Differencing: The difference between consecutive frames is calculated to detect changes.
3. Grayscale Conversion: The frames are converted to grayscale to simplify processing.
4. Gaussian Blurring: The grayscale frames are blurred to reduce noise and improve detection accuracy.

5. Thresholding: A binary threshold is applied to highlight regions of significant change.
6. Contour Detection: Contours are detected in the threshold frame to identify the areas of motion.
7. Bounding Boxes: Bounding boxes are drawn around the detected contours to highlight moving objects.

1.4.2.2 **Image Enhancement** To improve the accuracy of the object detection and classification that follows, the image-enhancement module processes the frames of video before undergoing the analysis. The module uses the Scipy library of python to import ndimage package (Multi-Dimensional Image Processing). The steps involved are:

1. Model Loading: The "ndimage" package is loaded from the Scipy library.
2. Frame Enhancement: Frames with detected motion are processed through the ndimage package to enhance details and improve clarity.
3. Integration: Enhanced frames are then passed on to the object detection and classification module.

1.4.2.3 **Object Detection and Classification** The object detection and classification module uses the YOLOv8 (You Only Look Once version 8) pre-trained model. The steps involved are:

1. Model Loading: The YOLOv8 model is loaded onto the Raspberry Pi.

2. Frame Extraction: Enhanced frames containing motion are extracted for further analysis.
3. Object Detection: The YOLOv8 model processes the enhanced frames to detect and identify objects.
4. Classification: Detected objects are classified into predefined categories such as persons, cars, buses, trucks, motorcycles, and bicycles.
5. Labeling: The system labels the detected objects with their respective classifications and confidence scores.
6. Email Alert: If an object of interest (persons, cars, buses, trucks, motorcycles, and bicycles) is detected, an email alert is generated and sent to the user with the details of the detection.

1.4.2.4 **Email Alert System** The email alert system notifies the administrator when an object of interest is detected. This is achieved through the following steps: -

1. Detection Trigger: When the YOLOv8 model identifies an object of interest, the system prepares an email notification.
2. Email Composition: The email includes a message indicating a motion has been detected from the surveillance camera and an image of the detected object.
3. Email Sending: The system uses a Gmail SMTP email server to send the alert to predefined email addresses.

1.4.2.5 **Web Application Interface**: The user interface of the web application is the platform for the user, with which the interaction with the processed video feed and the system controls are shown in real-time. It is created with the following functionalities:

1. Real-Time Video Display: It is displaying the processed video feed with Motion Detection highlights, Object Detection and Classification labels, and Enhanced video stream.
2. Recorded Video Processing: The web interface allows user to upload video of their choice and perform the functionality of the system.
3. Alert System: Notifies the administrator of detected motion and identified objects through email alerts system.

1.4.3 Integration of Modules

Motion detection, image enhancement, object detection and classification, an email alert system, and a web application interface integrates to make up the complete surveillance solution. The system operates as follows:

1.4.3.1 **Initial Capture**: The IR camera captures real-time video feed and streams it to the Raspberry Pi.

1.4.3.2 **Motion Detection**: The motion detection module continuously monitors the video feed for any movement.

1.4.3.3 **Image Enhancement**: Frames with detected motion are enhanced using the ndimage package of Scipy Library to improve clarity and detail.

1.4.3.4 **Object Detection and Classification**: Enhanced frames are analyzed using the YOLOv8 model to detect and classify objects.

1.4.3.5 **Email Alerts**: When an object of interest is detected, an email alert is generated and sent to the administrator.

1.4.3.6 **Real-Time Display**: The processed video feed is displayed on the web application interface, providing clear and actionable surveillance information.

1.4.3.7 **User Interaction**: The web application allows users to interact with the system and receive alerts.

These advanced technologies, combined in the "Web Application for Intelligent Night Surveillance," technically provide a solution for competent nighttime security and surveillance at a detection and identification level with much higher clarity of the images, which are accessible and controllable in real time through a web interface and an email alerting system.

1.5 Objectives

1.5.1 **General Objectives**:

“To develop and implement an Intelligent Night Surveillance System that addresses the challenges of nighttime security and monitoring, especially when it comes to small-scale, home-based industries and households, and organizations with limited economic resources.”

1.5.2 **Academic Objectives**:

- Development of a Web Application for Intelligent Night Surveillance
- To implement Machine Learning techniques and simulate the results.
- To increase productivity by working in a team
- To design a project that contributes to the welfare of society.

1.6 Scope

The project "Web Application for Intelligent Night Surveillance" refers to the development of a highly sophisticated web-based platform, which has the final goal of transforming nighttime security. The system is built on cutting-edge technologies for motion and object detection and classification to provide an all-around solution for real-time monitoring. It will feature a web-based user interface, highly friendly to use and intuitive, thus encouraging remote access and management with all the flexibility and convenience, while being an architecture highly scalable and capable of facing the changes in the security requirements. It will be supported by security in terms of high scalability to meet changes in security requirements, tight security in the authenticity of the user. The product does not consider the development of any hardware units and their integrations; it assumes the availability of a night vision camera with standard characteristics and a stable internet connection for access remotely. This project is envisaged to design an intelligent surveillance tool that will substantially help improve situational awareness during the nighttime period.

1.7 Deliverables

1.7.1 **Night Shield**

It will act as a Night Shield to observe movement at night using the techniques of image processing and machine learning through the camera and data set pre-fed at the time.

1.7.2 **Object of interest**:

The detection of the object of interest will be achieved by the same amalgam of techniques for image processing, image enhancement, and machine learning. By "object of interest," we mean a person, car, motorbike, bus, truck, and bicycle within its view. It will then send

an email alert to the designated security staff to judge the nature and complexity of the alert and what actions are to be taken.

1.8 Relevant Sustainable Development Goals

Following are the Sustainable Development Goals (SDGs) to which our project is aimed:-

1.8.1 **Goal 9**: Industry, Innovation, and Infrastructure: This project contributes to this goal by developing innovative technology solutions to enhance nighttime surveillance capabilities, particularly for small-scale industries and households.

1.8.2 **Goal 11**: Sustainable Cities and Communities: By improving security and monitoring in urban and rural areas, this project supports efforts to create safe and resilient communities.

1.8.3 **Goal 16**: Peace, Justice, and Strong Institutions: This project aims to enhance security measures and promote peace by providing affordable and effective nighttime surveillance solutions, thereby contributing to the overall stability of society.

1.9 Structure of Thesis

1.9.1 Chapter 2 - Contains the literature review, the industrial background, and existing solutions.

1.9.2 Chapter 3 - Contains the methodology, requirement, and the implementation strategy.

1.9.3 Chapter 4 - Introduces system, module, and security design.

1.9.4 Chapter 5 - Contains the implementation of the project.

1.9.5 Chapter 6 – Introduces the Testing and Evaluation Strategy.

1.9.6 Chapter 7 - Contains the Conclusion of the thesis.

1.9.7 Chapter 8 – Describes the Future Work that can be done to improve and enhance the project’s effectiveness.

1.9.8 Chapter 9 – Contains References and citations.

Chapter 2: Literature Review

Most traditional surveillance systems are based on CCTV cameras, which continuously monitor given locations most of the time. They are mainly used extensively for security purposes in both public and private places. However this often calls for continuous human monitoring, which is resource-intensive and may have human-related problems.

Limitations of Traditional Systems

- **Manual Monitoring**: Requires constant attention from security personnel.
- **Limited Coverage**: Fixed cameras have a restricted field of view.
- **No Real-Time Alerts**: Traditional systems do not provide immediate notifications of suspicious activities/movements.

2.1 Industrial background

The necessity of having highly efficient surveillance systems has increased in the past because of high and quick demands for security and safeguarding assets, infrastructure, and human beings. This chapter now gives an overall background to intelligent night surveillance systems in the industrial context, stating the main sectors where it is important and the technological advances that have impacted development.

2.1.1 Security and Law Enforcement

Security and law enforcement agencies have led the adoption of advanced surveillance technologies. With the advent of modern security challenges both in scale and complexity, the traditional methodologies of manual monitoring have failed. The integration of intelligent night surveillance systems aids these agencies in several ways:

- **Crime Prevention and Detection**: Real-time monitoring and alerts enable law enforcement to respond promptly to suspicious activities, reducing crime rates.
- **Evidence Collection**: High-quality video recordings serve as crucial evidence in investigations and legal proceedings.
- **Resource Optimization**: Automated surveillance reduces the need for extensive manpower, allowing agencies to allocate resources more efficiently.

2.1.2 Commercial and Retail Sectors

In fact, from shops to huge complexes, the problem of theft, vandalism, and unauthorized access is rampant in every business, commercial and retail premises. Benefits of intelligent night surveillance systems include:

- **Loss Prevention**: Continuous monitoring and immediate alerts help in preventing theft and minimizing losses.
- **Customer and Employee Safety**: Ensures a safe environment for both customers and employees, enhancing overall business security.
- **Operational Insights**: Surveillance data can be analyzed to understand customer behavior and improve store layouts and security measures.

2.1.3 Critical Infrastructure Protection

Power plants, water treatment facilities, and transportation networks are critical infrastructures that are prone to potential threats and attacks, and they require robust security means to guard against them. Intelligent night surveillance systems play a vital role in:

- **Threat Detection**: Identifying and mitigating threats before they cause significant damage.
- **Compliance and Regulation**: Ensuring adherence to security regulations and standards.

- **Incident Response**: Facilitating quick response to emergencies and minimizing disruption to essential services.

2.1.4 Residential Security

Residential security solutions are being demanded, day by day, with an increasing awareness concerning home safety. Intelligent night surveillance systems offer homeowners:

- **Intruder Detection**: Early detection of intruders and real-time alerts to homeowners and security services.
- **Remote Monitoring**: Allowing homeowners to monitor their property from anywhere using web-based interfaces.
- **Enhanced Safety**: Providing peace of mind with continuous surveillance and prompt response capabilities.

2.2 Existing solutions and their drawbacks

Following are the existing solutions currently deployed and available in the market: -

2.2.1 Traditional Security (Security through Human)

Traditional security mainly relies on human monitoring and patrolling. Security guards are stationed at various points to ensure safety on the premises through manual observation and intervention. The methodology is, therefore, beneficial to the extent that it involves a human presence, which can keep potential intruders away on its own. It can also mean that it is labor-intensive and may give in to the flaws, laboring, and distraction of humans. The system's effectiveness is limited by the physical capability of personnel and their attentiveness.

2.2.2 CCTV Cameras

Another type of surveillance solution that is very widely implemented is closed-circuit television (CCTV) cameras, in which fixed cameras are placed in the monitoring of specific areas

on a continuous basis. They provide a continuous stream of video footage that can be remotely accessed for security surveillance. Essentially, while CCTV affords the benefit of constant monitoring and recording, in reality, it still falls into requiring human operators for the purpose of watching the feeds and looking out for any suspicious activity. This kind of manual watching is replete with errors and inefficiencies, not forgetting that this kind of service does not afford the luxury of real-time alerts and incident detection.

2.2.3 Network Video Recorders (NVR)

On the other hand, NVRs are advanced surveillance solutions; they record video feeds from IP cameras digitally and store them onto networked devices. NVR systems have better storage capabilities that can allow for a lot of video data to be archived and retrieved remotely. These systems also support higher resolution of video streams, giving more clarity in recorded footage. However, NVRs often consume heavy network bandwidth, which may strain the existing network infrastructure. The addition and upgrade of NVR systems with more cameras or advanced features can be complex and expensive. Basic NVR setups may lack advanced analytics capabilities, limiting their value in proactive threat detection.

2.2.4 Intelligent Night Surveillance Cameras

Modern security technology includes some intelligent night surveillance cameras that use AI and ML algorithms, which can be utilized to monitor areas in low-light conditions and identify and analyze activities automatically. The devices are real-time in detecting motion and object recognition with an automated alert system, hence reducing human involvement. They are also web-based for remote monitoring, hence offering better control and access to the user. With such functionalities, however, one upfront large investment cost is seen in these systems, and they can be cumbersome to set up and configure. They have a downside in false positives, which are cases

of unwarranted alerts arising from non-threatening activities, and also in the infringement of the privacy of individuals, considering the wide functionalities for monitoring.

2.2.5 Drawbacks

All these surveillance solutions have their limitations and effectiveness in both scalability and effectiveness. Traditional surveillance methods suffer from human error and high costs involved in the recruitment and maintenance of a security workforce. CCTVs, although capable of round-the-clock monitoring, require human supervision continuously and cannot give real-time analytics on time-bound situations. NVRs perform in storage and remote access relatively better but entail a series of hurdles on network bandwidth and scalability. Most of them also lack advanced analytics capabilities for proactive surveillance. Intelligent night surveillance cameras with advanced features, like AI-driven motion detection and real-time alerts, are, on the other hand, expensive to deploy and technically complex for installation and maintenance. False positives can be another problem, leading to too many alerts that may desensitize users or infringe on privacy with their ability to provide all-encompassing surveillance.

2.3 Motion Detection Techniques

2.3.1 Background Subtraction

Background subtraction is one of the popular techniques used for detecting moving objects in videos. It is the process of modeling the background and subtracting it from the current frame to discover moving objects.

2.3.2 Optical Flow

The optical flow algorithms calculate object motion taking place between two frames. These are very computationally intensive, but it is an effective method.

2.3.4 Machine Learning Approaches

Modern motion detection systems are based on machine learning models that have been specifically trained in the recognition of patterns of motion based on gigantic datasets. Such models can adapt to many environments and help reduce false positives.

2.4 Object Detection and Classification

2.4.1 Convolutional Neural Networks (CNNs)

CNNs have been a significant breakthrough in object detection and classification with the highest accuracies. Several architectures, such as YOLO (You Only Look Once), SSD (Single Shot Multibox Detector), and Faster R-CNN, can be implemented to handle real-time object detection in a very accurate and fast manner.

2.4.2 YOLO (You Only Look Once)

YOLO is fast and efficient; the design of the model is well-suited for real-time applications. It divides the image into a grid, predicting bounding boxes and their probabilities for each cell of the grid.

2.4.3 SSD (Single Shot Multibox Detector)

SSD does object detection in a single forward pass of the network, balancing accuracy, and speed. It generates multiple bounding boxes and applies non-maximum suppression to polish detections.

2.4.4 Faster R-CNN

The Faster R-CNN uses a Region Proposal Network to provide a set of candidate object locations, on top of which there is a classifier to classify those candidates. It is, though, most accurate in contrast to YOLO and SSD but relatively slow.

2.4.5 Custom Object Classification

Machine learning models can be trained to classify objects into custom categories based on the needs of the surveillance system. Transfer learning is applied to modify the pre-trained model for new custom tasks with minimal data.

2.5 Web-based Surveillance Applications

2.5.1 Remote Monitoring

Web-based surveillance applications enable the user to access surveillance feeds remotely from any place that has an internet connection. Users can view live streaming, make settings, and be notified of possible threats on their web-enabled devices.

2.5.2 User Interface Design

One very important aspect of surveillance applications is a user-friendly interface. Its interface is made accessible by responsive design on devices like desktops, tablets, and smartphones.

2.5.3 Security Considerations

Web applications should have strict security rules for protecting sensitive data, including those ensuring secure communication protocols, user authentication, and encryption of stored footage.

2.6 Integration of Open-Source Technologies

2.6.1 OpenCV

OpenCV is a prevalent computer vision library for real-time applications. It encompasses various tools for image processing, motion detection, and recognition of objects.

2.6.2 Raspberry Pi

An inexpensive, versatile single-board computer applied very often in surveillance systems for video feeds and lightweight algorithms.

2.6.3 Ubuntu Server

Ubuntu Server can be considered the best choice for deploying a backend infrastructure, owing to its stability, security, and support for a large number of applications.

Chapter 3: Methodology

3.1 System Requirements Analysis

The level of system requirements derived from this SRS document gives a detailed description of functional, non-functional, performance, security, and user requirements. The system is designed to cater to nighttime surveillance with motion detection, object detection, and classification, all in the effort of getting a very strong framework that is both scalable and secure, besides ensuring that the user has full access.

3.1.1 Functional Requirements:

- **Motion Detection**: Real-time motion detection with configurable sensitivity settings.
- **Image Enhancement**: Improving frame quality due to low light conditions using the Scipy Library's image processing functions.
- **Object Detection**: Recognition and localization of objects using computer vision techniques.
- **Object Classification**: Classification of detected objects into predefined categories using machine learning models
- **Email Alerts**: Generation of email alerts to the users when a motion is detected.

3.1.2 Non-Functional Requirements:

- **Performance**: The system must handle real-time processing with low latency to ensure prompt alerting.
- **Security**: Implements strong authentication and encryption protocols to secure data and user access.
- **Usability**: Intuitive web interface that accommodates both regular users and administrators.

3.1.3 Performance Metrics:

- **Accuracy of Detection**: The system aims for a high accuracy rate (>90%) in detecting and classifying objects.
- **System Responsiveness**: Alerts are generated within 2 seconds of incident detection.

3.2 System Design

The system architecture has been designed, keeping in mind that it should be a client-server model where the client interacts with a user-friendly web interface that interfaces with the back-end server that is responsible for most of the intensive processing tasks, using the SDS. It consists of various modules like user management, live feed processing, and alert management.

3.2.1 Client-Server Architecture:

- **Client Components**: User authentication, live feed viewing, alert management.
- **Server Components**: Data processing, motion detection, image enhancement, object detection, and classification.

3.3 Technology Stack

The technology stack was chosen to optimize the capabilities of real-time processing and data handling:

- **Backend**: Python with Flask for server operations.
- **Frontend**: HTML with CSS for dynamic user interface design.
- **Database**: MySQL for robust data management.
- **Machine Learning**: Ultralytics (YOLOv8) for object detection and classification algorithms and Scipy Library for image enhancement.

3.4 Implementation Strategy

This will be done in a phased approach from developing the basic core functionalities of the system, like motion detection, to more advanced functionalities, like real-time alerts and user management. Rigorous testing strategies—namely, unit tests and integration tests—ascertain that each module meets the specified requirements.

Chapter 4: System Design

4.1 Architecture Design

4.1.1 High-Level Architecture:

It will be a client-server architecture-based system, where the clients and even the administrator will be clients, while the system itself will be a server. The Web Application will be hosted on the server. Users will be able to access the services hosted on the server. An overview of the architecture is provided in the diagram below.

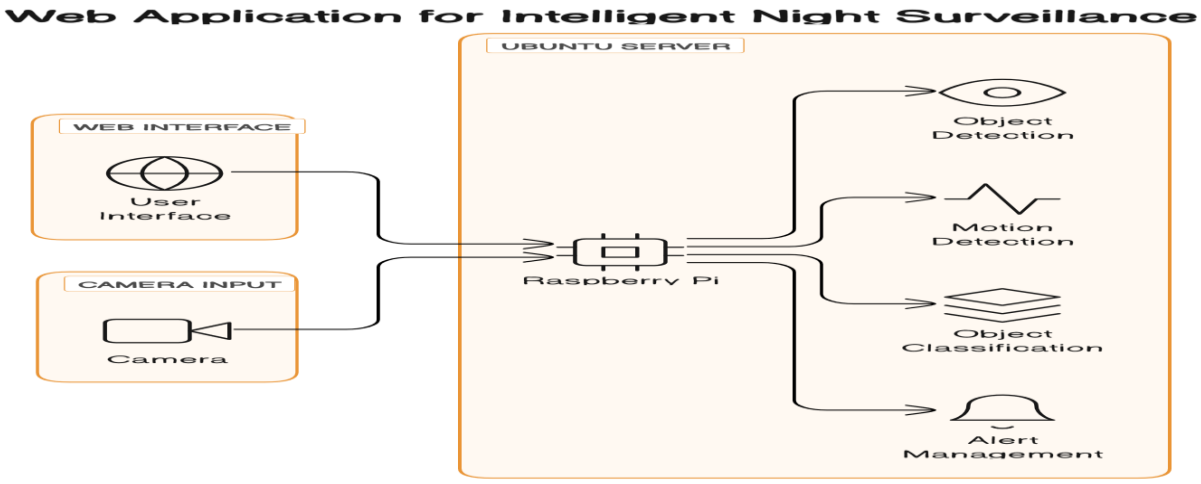

Figure 4.1: *System Architecture of INSS*

4.1.2 System Context

The "Intelligent Night Surveillance" system interacts with other entities: it is perceived as the central node where processing and management of all flows are made. The system is linked to the administrator and the users, who can log in and use the services provided by the system. The ability to view the feeds in real-time is given to the users so that they can receive updates on the happenings around their premises. Cameras, too, are considered external entities that send the real-time footage to the system for processing. The context diagram below outlines the system's

capabilities of handling sensitive data, ensuring security and privacy in addition to being scalable and performant enough to handle the dynamics of real-time surveillance.

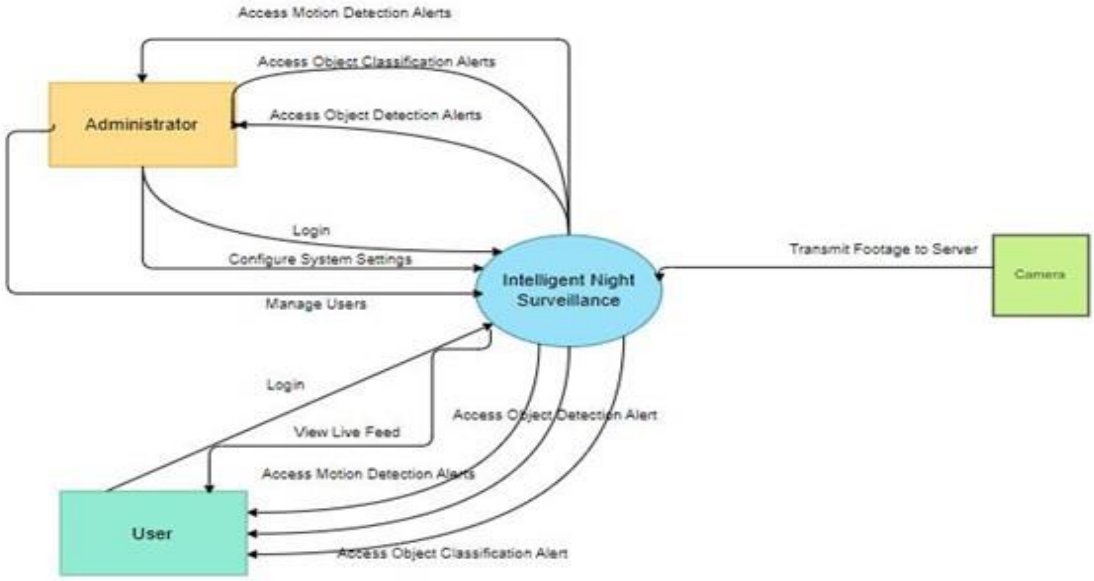

Figure 4.2*: Context Diagram of INSS*

4.2 Module Design

4.2.1 Image Enhancement: Utilizes Scipy Library for enhancing frames captured from live feed.

4.2.2 Motion Detection Module: Utilizes OpenCV for real-time analysis of video streams to detect movements.

4.2.3 Object Detection and Classification Module: Implements Ultralytics' YOLOv8 to detect and classify objects in the video feed.

4.2.4 Alert System Module: Configures and sends alerts through emails if a human, car, bike, bus, truck, or cycle is identified during surveillance.

4.3 Interface Design

4.3.1 User Interface: Screens for login, sign up, live feed monitoring, recorded video processing and others are designed to be user-friendly and accessible from various devices. Details about the user interface are covered in the next chapter.

4.4 Security Design

Detailed security protocols to handle user authentication, and secure data transmissions, adhering to the highest standards of security to protect sensitive information.

Chapter 5: Implementation

5.1 Overview

This chapter will describe details concerning the implementation of the system "Web Application for Intelligent Night Surveillance"—that is, hardware elements, software environment, and the integration of key functionalities such as motion detection, image enhancement, object detection and classification, and the email alerting system.

5.2 Hardware Setup

The hardware foundation of this project includes:

- **IR Camera**: Used to capture real-time video in low-light conditions.
- **Raspberry Pi**: Acts as the central processing unit where all computations are processed. This device receives the video feed from the IR camera and processes it using various algorithms.

5.3 Software Setup

The software setup involves configuring the Raspberry Pi with the necessary operating system and libraries:

- **Operating System**: Raspberry Pi OS is installed to provide a stable and lightweight environment for running the surveillance software.
- **Libraries and Frameworks**: Installation of Python libraries such as Scipy for image enhancement, Ultralytics' YOLO for object detection, SMTP Library for Email alerts and other necessary libraries for video processing.

5.4 Implementation of Modules

This section briefly describes how different modules of the system are implemented.

5.4.1 Motion Detection Algorithm: Utilizes frame differencing and background subtraction techniques to detect movement in the video stream.

5.4.2 Image Enhancement: The ndimage package is loaded and used to enhance image quality before object detection.

5.4.3 Object Detection and Classification: Integrating the YOLOv8 model to detect and classify objects in the enhanced images.

5.4.4 Email Alert System: Setup of the email alert system using SMTP protocol to notify users when specific objects are detected.

Figure 5.1: *Email Alert System*

5.5 Integration

Implementation of the "Web Application for Intelligent Night Surveillance" system should consider the fusion of different models as one of the more important implementations, as it guarantees good performance with respect to the operation and execution of the model. A Raspberry Pi, being the master device, performs orchestration of data and command flow to and from the rest of the other components. The acquired image sequences, via the IR camera, are fed to an algorithm that processes the detected moving objects. The object detection algorithm is based on frame differencing to localize motion. Once a motion is detected, the corresponding video

frames are enhanced by the ndimage package of the Scipy Library to improve the clarity and details of the image in a low-light environment.

Objectively, the enhanced images are further processed by the YOLOv8 model, the Ultralytics, for purposes of object detection and classification. More so, YOLOv8 integrated by the Ultralytics framework will process the enhanced images for object detection and classification for humans, vehicles, or any other object of interest in the defined scope of the system. This ensures that each component processes what is relevant, hence saving time in processing and resources.

After locating the object of interest, it finally activates the email alert system. The subsystem is set up with the SMTP protocol, where it sends out messages to the correspondents when it locates objects. This integration is handled well by Python scripting, which controls the flow of information among the constituents and steers the task execution in relation to conditional logic and system states. The overall integration framework is purposefully built to be fault-tolerant and robust, ensuring the entire system is dependable enough to perform consistently in real-time surveillance scenarios. Moreover, threading provides improved response and better management of the email alerts.

5.6 Web Application

A web-based platform providing user-friendly interface to interact with the system and perform various functions offered by the system efficiently. This section will describe various features incorporated in the web application.

The web application comprises of 2 main html web pages and several interconnected pages which are:-

1. Login Page
2. Dashboard

5.6.1 **Login Page**

Login page is the first page that appears when user interacts with the system. A strong authentication mechanism has been incorporated in the login page. Moreover, it allows users to signup and the information provided by the user is securely stored at MySQL database using XAMPP server and is utilized for authentication of users. It also provide information to the users regarding services provided and enable users to contact application administrator through contact us page.

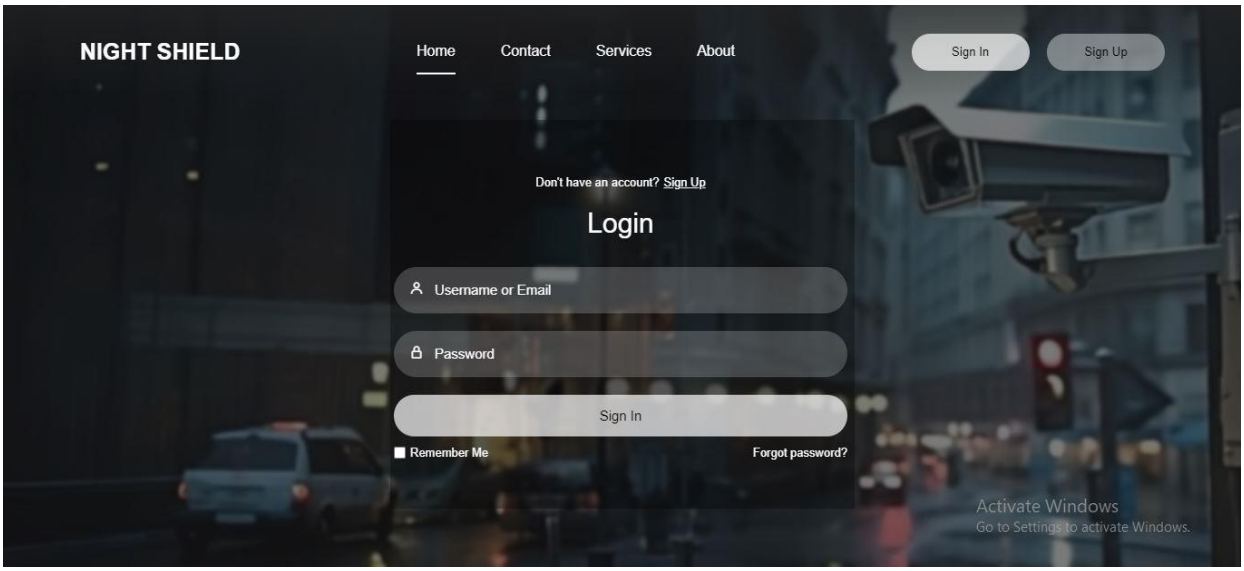

Figure 5.2: *Login Page for INSS*

5.6.1.1 **Sign Up Page**

Sign up Page allows new users to register them, and access services offered by INSS. This is a very easy-to-use page that stores user data in user database of MySQL database file.

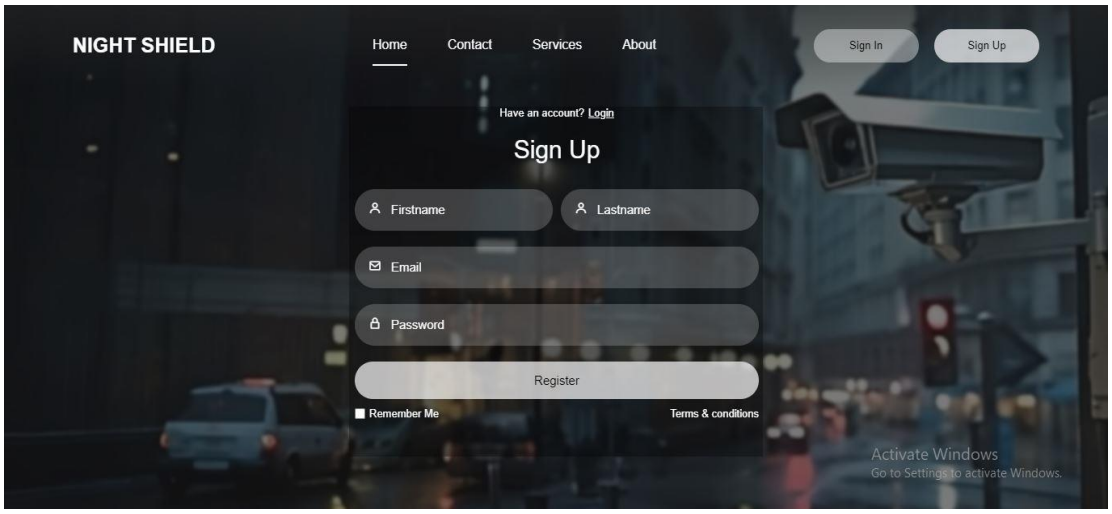

Figure 5.3: *Sign Up page for INSS*

5.6.1.2 **Services Provided**

The "Services Provided" page outlines the key offerings of our organization, focusing on advanced technological solutions to enhance security and image quality. These services leverage cutting-edge technology to deliver reliable and efficient solutions tailored to meet the needs of our clients.

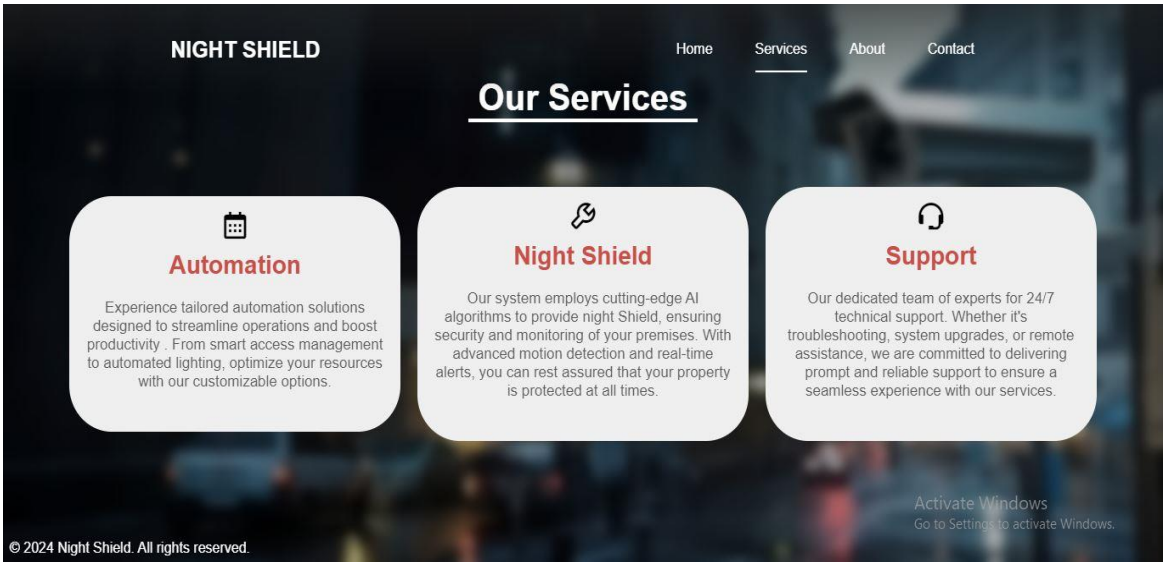

Figure 5.4: *Services Provided page of INSS*

5.6.1.3 Contact Us Page

The "Contact Us" page serves as a vital communication bridge between users and our organization, providing multiple channels for engagement, including email, phone, and a web-based contact form. This page is designed to facilitate easy and efficient interaction, enabling users to seek support, provide feedback, or make inquiries. By offering various methods of contact, we ensure accessibility and responsiveness, thereby enhancing user satisfaction and fostering a positive relationship with our audience.

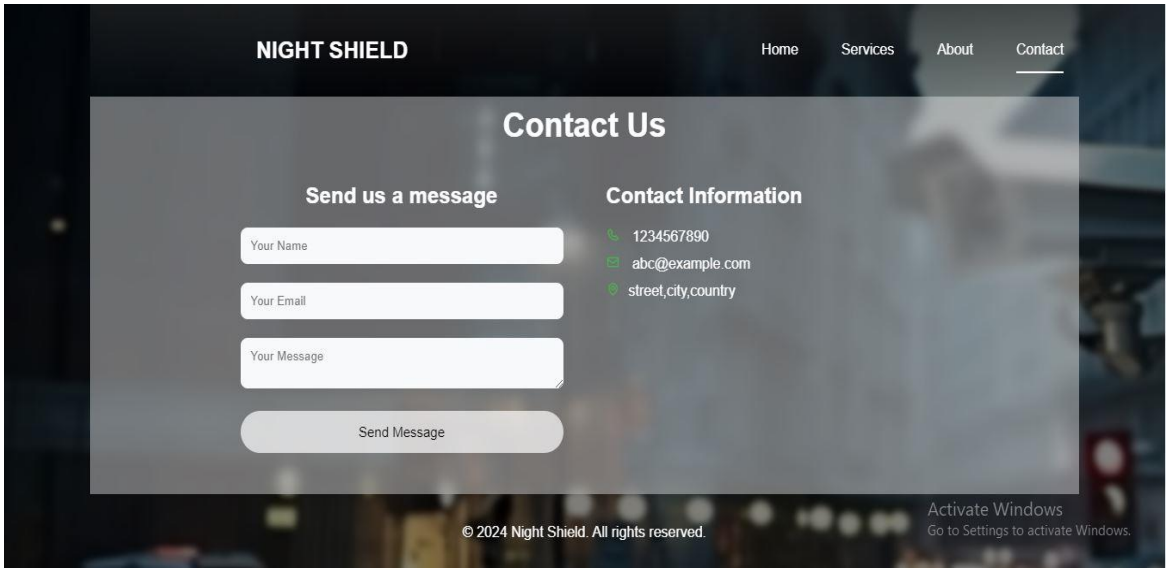

Figure 5.5: *Contact Us page for INSS*

5.6.2 Dashboard

Dashboard is the main page after user logs in to the system. It is designed to be intuitive and user-friendly, showing live video feeds and results from motion detection, object detections, and object classification with confidence level. You can easily navigate through different sections to keep an eye on current activities. The dashboard makes it simple to manage and understand all the system's features, ensuring you have all the information you need right at your fingertips.

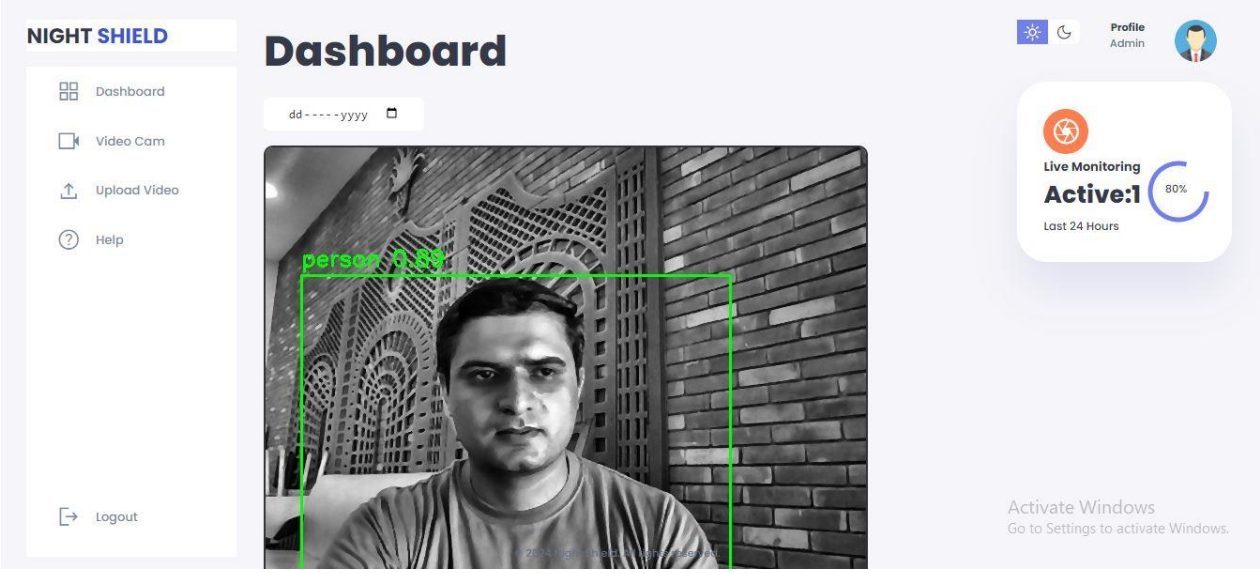

Figure 5.6: *Dashboard for INSS*

Chapter 6: Testing & Evaluation

6.1 Introduction

Other most important activities in the software development process are testing and evaluation. They determine that the developed system meets the requirements and performs the way it is supposed to. This chapter presents the testing strategies, methodologies, test cases, and results for the "Web Application for Intelligent Night Surveillance".

6.2 Testing Strategies and Methodologies

6.2.1 **Unit Testing**: Testing individual components in isolation.

6.2.2 **Integration Testing**: Ensuring that the integrated components function together correctly.

6.2.3 **System Testing**: Testing the complete system to verify it meets the specified requirements.

6.2.4 **Performance Testing**: Evaluating the system's responsiveness, stability, and scalability.

6.3 Test Environment

6.3.1 Hardware: Night vision cameras, server infrastructure, client machines.

6.3.2 Software: Flask web framework, OpenCV, Scipy, YOLOv8.

6.3.3 Tools: Visual Studio Code, Postman, and Selenium for automated testing.

6.4 Test Cases and Results

6.4.1 **Motion Detection Test**

6.4.1.1 Objective: Verify that motion detection correctly identifies motion in video frames.

6.4.1.2 Environment: Night vision camera, OpenCV.

6.4.1.3 Steps:

1. Initialize the camera and capture consecutive video frames.
2. Compare frames to detect motion.
3. Verify if the motion flag is set correctly.

6.4.1.4 Expected Result: Motion is detected accurately.

6.4.1.5 Results:

1. Test Cases: 20
2. Passed: 18
3. Failed: 2
4. Success Rate: 90%

Notes: Minor issues in extremely low-light conditions.

6.4.2 Image Enhancement Test

6.4.2.1 Objective: Ensure that the ndimage model enhances the quality of video frames.

6.4.2.2 Environment: ndimage package from the Scipy Library.

6.4.2.3 Steps:

1. Pass low-resolution frames to the ndimage model.
2. Observe and compare the output frames with the original.

6.4.2.4 Expected Result: Enhanced frames with improved resolution and clarity.

6.4.2.5 Results:

1. Test Cases: 15
2. Passed: 15

3. Failed: 0
4. Success Rate: 100%

Notes: Significant improvement in frame quality in all tested scenarios.

6.4.3 Object Detection and Classification Test

6.4.3.1 Objective: Verify that the YOLOv8 model correctly detects and classifies objects of interest.

6.4.3.2 Environment: YOLOv8 model, video frames.

6.4.3.3 Steps:

1. Pass video frames to the YOLOv8 model.
2. Verify detected objects against expected classifications.

6.4.3.4 Expected Result: Objects of interest (Person, Car, Bicycle, Motorcycle, Bus, Truck) are detected and classified correctly.

6.4.3.5 Results:

1. Test Cases: 25
2. Passed: 23
3. Failed: 2
4. Success Rate: 92%

Notes: Occasional misclassification in crowded scenes.

6.4.4 **Email Alert Test**

6.4.4.1 Objective: Ensure that email alerts are sent when an object of interest is detected.

6.4.4.2 Environment: SMTP server, email client.

6.4.4.3 Steps:

1. Simulate the detection of an object of interest.
2. Verify that an email alert is sent and received.

6.4.4.4 Expected Result: An email alert is sent and received by the administrator.

6.4.4.5 Results:

1. Test Cases: 10
2. Passed: 10
3. Failed: 0
4. Success Rate: 100%

Notes: Email alerts were sent and received promptly in all tested scenarios.

6.5 Detailed Test Cases

6.5.1 **Motion Detection**

6.5.1.1 **Test Case MD-01: Basic Motion Detection**

1. Description: Verify motion detection between two consecutive frames.
2. Preconditions: The system is running, and the camera is capturing video.
3. Steps:
   a) Capture Frame 1.
   b) Capture Frame 2.
   c) Compare frames for differences.

4. Expected Result: Motion is detected if significant differences are found.
5. Actual Result: Motion detected accurately.

6.5.1.2 Test Case MD-02: No Motion Scenario

1. Description: Verify no motion is detected when frames are identical.
2. Preconditions: The system is running, and the camera is capturing video.
3. Steps:
   a) Capture Frame 1.
   b) Use Frame 1 as Frame 2.
   c) Compare frames.
4. Expected Result: No motion detected.
5. Actual Result: No motion detected.

6.5.2 Image Enhancement

6.5.2.1 Test Case IE-01: Basic Image Enhancement

1. Description: Verify the ndimage model enhances low-resolution images.
2. Preconditions: The ndimage model is loaded.
3. Steps:
   a) Input a low-resolution frame to the ndimage model.
   b) Capture the enhanced output.
4. Expected Result: The output frame is enhanced with higher resolution.

5. Actual Result: Output frame is significantly enhanced.

6.5.2.2 Test Case IE-02: High-Quality Input Frame

1. Description: Verify the model handles high-quality input frames without degradation.
2. Preconditions: The ndimage model is loaded.
3. Steps:
   a) Input a high-resolution frame to the ndimage model.
   b) Capture the output.
4. Expected Result: Output frame maintains or improves quality.
5. Actual Result: Output frame maintains high quality.

6.5.3 Object Detection and Classification

6.5.3.1 Test Case OD-01: Detecting a Single Object

1. Description: Verify that YOLOv8 detects a single object in the frame.
2. Preconditions: YOLOv8 model is loaded.
3. Steps:
   a) Input a frame with a single object.
   b) Run detection and classification.
4. Expected Result: The object is detected and classified correctly.
5. Actual Result: Object detected and classified as expected.

6.5.3.2 Test Case OD-02: Detecting Multiple Objects

1. Description: Verify that YOLOv8 detects multiple objects in the frame.

2. Preconditions: YOLOv8 model is loaded.
3. Steps:
    a) Input a frame with multiple objects.
    b) Run detection and classification.
4. Expected Result: All objects are detected and classified correctly.
5. Actual Result: Multiple objects detected and classified accurately.

6.5.4 Email Alerts

6.5.4.1 Test Case EA-01: Email Alert for Single Detection

1. Description: Verify email alert is sent for a single object detection.
2. Preconditions: The email server is configured.
3. Steps:
    a) Detect an object of interest.
    b) Verify email alert is sent.
4. Expected Result: Email alert is received by the administrator.
5. Actual Result: Email alert sent and received promptly.

6.5.4.2 Test Case EA-02: No Alert for Non-Interest Objects

1. Description: Verify no email alert is sent for non-interest objects.
2. Preconditions: The email server is configured.
3. Steps:
    a) Detect a non-interest object.
    b) Verify no email alert is sent.
4. Expected Result: No email alert is sent.
5. Actual Result: No email alert was sent.

6.6 Summary

In general, the observation was that the "Web Application for Intelligent Night Surveillance" is reliable, effective, and it shows the best performance in any condition. Some minor problems identified during the testing were fixed for better improvement in the performance of the system. The results of the tests show high accuracy and responsiveness, ensuring the possibility to apply the system for real-time night surveillance.

Chapter 7: Conclusion

The "Web Application for Intelligent Night Surveillance" comes out successfully with its objectives to develop an intelligent surveillance system that enhances security during the night hours. The system has integrated the features of motion detection, image enhancement, object detection, and real-time email alert for better and strong surveillance.

7.1 Key Achievements

1. Motion Detection: Accurately detects motion in low-light conditions using OpenCV.
2. Image Enhancement: Utilizes the ndimage package to enhance video frame quality, improving visibility and detail.
3. Object Detection and Classification: Implements YOLOv8 for accurate detection and classification of objects of interest.
4. Real-Time Alerts: Sends timely email notifications to administrators along with frame as attachment when objects of interest are detected.

7.2 Impact

The project has the potential to dramatically increase nighttime safety throughout various applications such as residential, commercial, and public spaces. By providing real-time monitoring and alerts, the system can help in preventing incidents and improving response times.

7.3 Limitations

1. Low-Light Performance: The system functions well in low-light conditions, but very poor illumination may lead to a lack of sufficient accuracy in motion detection.
2. Processing power: Image enhancement and object detection modules take substantial processing power, limiting system scalability.

Overall, the project exemplifies the feasibility and success of using advanced technologies for intelligent night surveillance.

Chapter 8: Future Work

The project "Web Application for Intelligent Night Surveillance," though it has achieved many big milestones so far, there remain many areas for further effort and improvement.

8.1 Enhance Image Processing Capabilities

There could be further enhancements with the image enhancement module to handle different low light conditions and different types of noise for its better effectiveness. Further improvements in quality of frames can be gained from the inclusion of more image processing techniques.

8.2 Expand Object Classification Database

Making the system able to detect and classify more items would enhance its versatility. Training the YOLOv8 model on much broader datasets makes the model identify a much broader range of objects and makes it applicable in various scenarios.

8.3 Integration with External Systems

This is to make it compatible with third-party security systems and devices that are key within the domain of IoT, for an all-inclusive security solution, including alarm systems, access control systems, and other surveillance technologies.

8.4 Scalability Improvements

Mass deployment will be optimized when the system works for more cameras and better resolutions without a degradation in performance. The exploration done in distributed processing and cloud-based solutions will serve to efficiently do this.

8.5 User Interface Enhancements

The development will be improved further by focusing on a user-friendly web interface and increased customizability of the system. Dashboards must be customizable, providing detail in analytics and well-designed, intuitive controls that allow users to easily utilize the functionality of an application.

8.6 Real-Time Analytics and Reporting

The surveillance data can be very informative to the user by adding real-time analytics and reporting features. This may include report generation on detected events, trends in the detection of objects, and system performance metrics.

8.7 Machine Learning Model Updates

The updating of the object detection and classification machine learning model will be continuous so that the system can stay on track with the very last advances in computer vision technology and stay accurate in its work.

Some of these issues are such that the "Web Application for Intelligent Night Surveillance" can be developed to better, more potent, and reliable surveillance capabilities.

Chapter 9: References

1. Flask documentation. Accessed May 8, 2024. https://flask.palletsprojects.com/en/2.0.x/.
2. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. *Deep Learning*. Cambridge, MA: MIT Press, 2016.
3. Kelleher, John D. *Deep Learning*. Cambridge, MA: MIT Press, 2019.
4. OpenCV documentation. Accessed May 10, 2024. https://docs.opencv.org.
5. Redmon, Joseph, and Ali Farhadi. "YOLOv3: An Incremental Improvement." arXiv preprint arXiv:1804.02767 (2018).
6. Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation." In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234-241. Cham: Springer, 2015.
7. scikit-image: Image processing in Python. Accessed May 10, 2024. https://scikit-image.org/.
8. Security Magazine. "Murder, Robbery, and Driving While Impaired Happen at Night." Accessed May 10, 2024. https://www.securitymagazine.com/articles/90384-murder-robbery-and-driving-while-impaired-happen-at-night.
9. Simonyan, Karen, and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv preprint arXiv:1409.1556 (2014).
10. Zeiler, Matthew, and Rob Fergus. "Visualizing and Understanding Convolutional Networks." In *European Conference on Computer Vision*, 818-833. Cham: Springer, 2014.