CAR ACCIDENT DETECTION , WILDLIFE RECOGNITION AND AVOIDANCE : ACCIDENT SHIELD

A SYNOPSIS
Submitted by

Himanshu Raj (1BM20CS057)
Keerthan S Gowda (1BM20CS070)
Kushal C (1BM20CS077)
Mehul Tej (1BM20CS085)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

Under the Guidance of

Sheetal V A
Assistant Professor, BMSCE
Dept of CSE, BMSCE

B. M. S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019

MAY 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project entitled "**CAR ACCIDENT DETECTION, WILDLIFE RECOGNITION AND AVOIDANCE : ACCIDENT SHIELD**" is a bonafide work carried out by HIMANSHU RAJ (1BM20CS057), KUSHAL C (1BM20CS077), MEHUL TEJ (1BM20CS085), KEERTHAN S GOWDA (1BM20CS070) in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the academic year 2023 - 24. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Guide	**Head of Department**
Sheetal V A , Assistent Professor, Dept of CSE, B.M.S. College of Engineering	**Dr. Jyothi S Nayak** , Professor and HOD, Dept of CSE, B.M.S. College of Engineering

Principal

Dr. S. Muralidhara
B.M.S. College of Engineering

External Viva

Name of the Examiners	Signature with Date
1.	
2.	

Table of Contents

Abstract

With the rapid increase in vehicular traffic, road safety has become a critical concern worldwide. Car accidents not only result in substantial economic , wildlife losses but also pose significant threats to human lives. This project presents a comprehensive Car Accident Detection System (CADS) designed to enhance road safety by leveraging advanced technologies.

The proposed CADS integrates computer vision, machine learning, data to accurately identify and assess potential car accidents. A network of cameras strategically placed on roadways captures video footage, which is processed using state-of-the-art deep learning algorithms. These algorithms analyze the video frames to detect patterns indicative of potential accidents, such as sudden deceleration, abrupt lane changes, and collisions.

Upon detecting a potential accident, the CADS system automatically triggers real-time alerts to relevant emergency services, law enforcement, and nearby vehicles equipped with compatible communication systems. This immediate response can significantly reduce emergency response times, enhancing the chances of minimizing injuries and wildlife damage.

The collected data can be utilized to identify accident-prone zones, study traffic patterns, and develop targeted safety measures to further improve overall road safety.

This project aims to contribute to the ongoing efforts towards creating intelligent transportation systems that prioritize the safety of both drivers and wildlife. The integration of cutting-edge technologies in the CADS demonstrates its potential as a proactive and effective tool for accident detection, wildlife recognition, and avoidance in the context of modern urban mobile

DECLARATION

We, hereby declare that the dissertation work entitled **"CAR ACCIDENT DETECTION, WILDLIFE RECOGNITION AND AVOIDANCE : ACCIDENT SHIELD"** is a bonafide work and has beencarried out by us under the guidance of Sheetal V A**,** Assistant Professor, Department of Computer Science and Engineering, B.M.S. College of Engineering, Bengaluru**,** in partial fulfillment of the requirements of the degree of Bachelor of Engineering in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi.

I further declare that, to the best of my knowledge and belief, this project has not been submitted either in part or in full to any other university for the award of any degree.

Candidate details:

SL. NO.	Student Name	USN	Student's Signature
1	Himanshu Raj	1BM20CS057	
2	Kushal C	1BM20CS077	
3	Keerthan S Gowda	1BM20CS070	
4	Mehul Tej	1BM20CS085	

Place: Bengaluru

Date:

Certified that these candidates are students of Computer Science and Engineering Department of B.M.S. College of Engineering. They have carried out the project work of titled "CAR ACCIDENT DETECTION, WILDLIFE RECOGNITION AND AVOIDANCE : ACCIDENT SHIELD" as final year (7^{th} and 8^{th} Semester) dissertation project. It is in partial fulfillment for completing the requirementfor the award of B.E. degree by VTU. The works is original and duly certify the same.

Guide Name Signature

Sheetal V A

Date:

Acknowledgment

The authors would like to thank Sheetal V A Assistant Professor Department of CSE, B.M.S College of Engineering who provided precious insights and guided us along the entire process. She has been a beacon of light and a source of inspiration throughout the development of the project.

List of tables

List of figures

Chapter 1
Introduction

1.1 Overview

Car accidents are a major safety concern worldwide, often caused by human errors.Another, less-known problem is accidents involving animals on roads, leading to harm and economic losses.

Artificial intelligence (AI) and machine learning (ML) are being used to tackle these issues.AI and ML offer real-time monitoring and smart decision-making for road safety.

This project explores how these technologies help in accident detection and preventing animal-vehicle collisions.We will discuss key technologies and strategies for making roads safer.AI and ML promise to reduce accidents, save lives, and protect the environment.

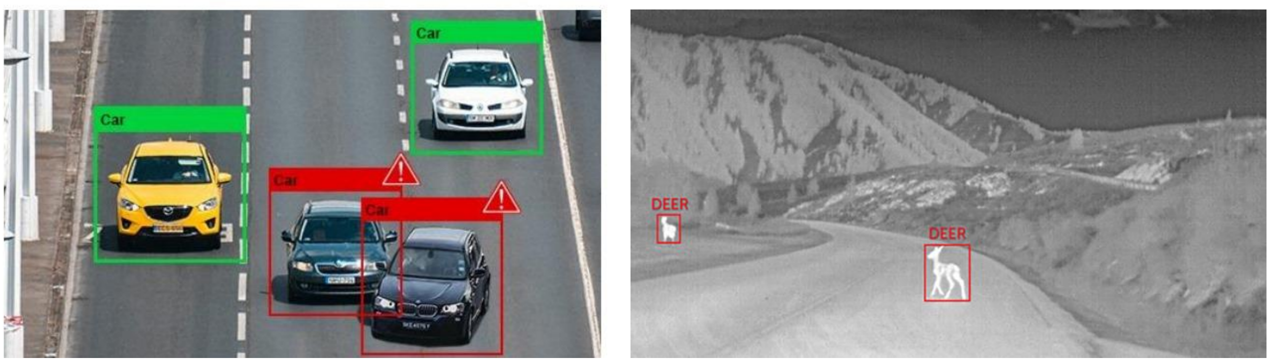


Figure 1. Overview of the car accident, wildlife recognition system

1.2 Motivation

The motivation behind this project stems from a deep concern for road safety and wildlife conservation. Car accidents and wildlife collisions are significant contributors to road fatalities and environmental harm, posing risks to both human lives and animal populations. By developing an integrated system for car accident detection, wildlife recognition, and avoidance, we aim to address these challenges head-on. Our motivation is to leverage advanced technologies to proactively identify and mitigate risks on roads, ultimately saving lives and preserving wildlife habitats. Through this project, we seek to contribute to safer roads and a healthier ecosystem, fostering a safer and more sustainable environment for all. Furthermore, the increasing frequency of car accidents and wildlife collisions underscores the urgency of implementing effective solutions. Traditional methods have proven insufficient in adequately addressing these challenges, highlighting the need for innovative approaches. By harnessing the power of deep learning models like YOLOv8 and CNN, coupled with robust data preprocessing tools such as Roboflow, our project aims to fill this gap. Our goal is not only to detect and mitigate incidents but also to empower drivers with real-time information and proactive avoidance strategies. Through collaboration and innovation, we aspire to create a safer and more harmonious coexistence between humans and wildlife on our roadways.

1.3 Objective

1. Integrating these models into a real-time system capable of providing timely alerts to drivers and authorities in the event of an impending collision.
2. Utilizing advanced data preprocessing techniques, such as those offered by Roboflow, to enhance the quality and efficiency of our model training process.
3. Enhancing road safety by empowering drivers with proactive avoidance strategies based on real-time hazard detection and recognition.
4. Contributing to wildlife conservation efforts by reducing the incidence of wildlife collisions and minimizing harm to vulnerable animal populations.
5. Evaluating the effectiveness of our system through rigorous testing and validation processes, with the ultimate aim of deployment in real-world environments to save lives and mitigate environmental impact.

1.4 Scope

Our project aims to develop an integrated system leveraging YOLOv8 and CNN models for car accident detection and wildlife recognition, respectively. By integrating these advanced deep learning technologies, we seek to create a unified solution capable of real-time hazard detection on roadways. This includes the identification of vehicles, pedestrians, and wildlife, enabling timely alerts to drivers and authorities to prevent accidents and collisions.

In addition to the technical integration of detection and recognition models, our scope extends to data preprocessing and model optimization using tools like Roboflow. We will preprocess and augment datasets to ensure the robustness and accuracy of our models across diverse road and environmental conditions. Furthermore, our system will operate in real-time, providing instantaneous alerts to drivers via a user-friendly interface, facilitating prompt response and avoidance strategies.

Beyond technical implementation, our project scope encompasses considerations for scalability, stakeholder collaboration, and ethical implications. We aim to design a system that can be scaled across different geographic regions and road types, collaborating with stakeholders such as government agencies and wildlife conservation organizations to ensure alignment with regulatory requirements and conservation goals. Moreover, we will address ethical considerations related to privacy, accessibility, and environmental impact, ensuring that our solution upholds ethical standards while contributing to road safety and wildlife conservation efforts.

1.5 Existing System

The current system for car accident detection and wildlife recognition relies primarily on traditional methods such as manual reporting and limited technological solutions like traffic cameras and signage. However, these methods often suffer from delays, limited coverage, and lack of real-time capabilities, particularly in identifying wildlife hazards. There is a clear need for a more comprehensive and integrated approach that leverages advanced technologies such as computer vision and machine learning to provide real-time detection, recognition, and avoidance strategies, thereby enhancing road safety and minimizing wildlife collisions.

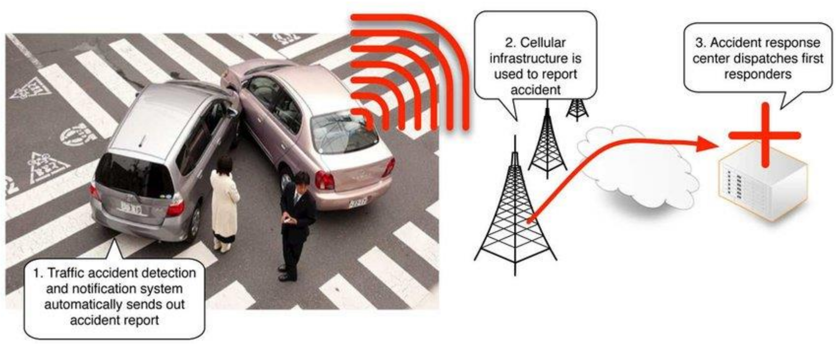


Figure 2. current car accident system

1.6 Proposed System

The proposed system aims to revolutionize road safety by integrating cutting-edge technologies for car accident detection and wildlife recognition, coupled with proactive avoidance strategies. Leveraging advanced deep learning models like YOLOv8 and CNN, alongside efficient data preprocessing tools like Roboflow, our system will provide real-time detection and recognition of potential hazards on roadways. By seamlessly integrating these components into a unified platform, we will enable swift and accurate identification of vehicles, pedestrians, and wildlife, empowering drivers with timely alerts and avoidance guidance. Through collaboration with stakeholders and adherence to ethical standards, our proposed system seeks to redefine road safety, mitigating risks and preserving lives while fostering harmony between humans and wildlife.

Chapter 2
Literature Survey

This section describes the main proposals found in the literature review.

[1] The increasing reliance on technology and infrastructure has spurred a demand for advanced accident detection systems. This particular system leverages cutting-edge technologies such as GPS, GSM, and VANET (Vehicular Ad-Hoc Network). It utilizes an optimized AODV (Ad-hoc On-Demand Distance Vector) route to accurately monitor and transmit accident data. Operating within the VANET framework, the system promotes road safety by enabling rapid communication between vehicles. Despite facing challenges related to maintaining routing stability, the system achieves an impressive accuracy rate of 76% in accident detection. This highlights its potential to contribute significantly to improving road safety through efficient and timely accident detection.

[2] The described accident detection systems utilize sensors, such as vibration or accelerometers, along with low-cost microcontrollers for processing. GPS technology plays a crucial role in ensuring precise accident location coordinates. While these systems benefit from advantages such as fast communication through Wi-Fi and MQTT (Message Queuing Telemetry Transport), they may encounter challenges leading to false alarms, particularly influenced by road conditions. Despite this limitation, the systems achieve a commendable accuracy rate of 65% in accident detection, showcasing their potential effectiveness in providing timely responses to road incidents.

[3] Vehicular networks play a crucial role in facilitating vehicle-to-vehicle (V2V) and vehicle-to-roadside (V2R) communication, enabling applications such as collision and distance warnings. The use of DSRC/WAVE (Dedicated Short-Range Communications/Wireless Access in Vehicular Environments) standards ensures efficient communication, and cellular networks extend the exchange of messages among vehicles. The implementation of these technologies results in a remarkable 98% accuracy rate, underscoring the effectiveness of vehicular networks in enhancing communication and safety features on the road.

[4] In the context of real-time object detection, Fast R-CNN is employed for identifying various elements such as vehicles, pedestrians, and road signs from video feeds. For accident prediction, logistic regression is utilized to model the relationship between binary outcomes and predictor variables, which may include factors like weather and road conditions. Additionally, the Random Forest algorithm, akin to decision trees, is employed to predict accident outcomes based on historical data. The application of these techniques results in an impressive accuracy rate of 90.3%, showcasing the efficacy of the chosen methodologies in accident prediction.

[5] The system described focuses on the detection of damaged cars through surveillancecamera footage, specifically addressing object detection within the realm of machine vision.It employs a supervised learning approach with three stages, incorporating support vector machines trained on features extracted from Histogram of Gradients (HOG) and Gray Level Co-occurrence Matrix (GLCM). The system compiles two datasets of damaged cars, achieving accuracies of 81.83% and 64.37% for different image qualities. This underscores the effectiveness of the proposed methodology in accurately identifying damaged cars in varying conditions, showcasing its potential utility in applications related to vehicle safetyand surveillance.

[6] The system utilizes YOLOv3, a deep learning algorithm, for the detection of accidents andobject classification in live video streams. To enhance accuracy, it incorporates object tracking and the Violent Flow Descriptor (VIF), utilizing support vector machines. The system achieves an impressive 89% accuracy, aiming to identify accidents from live CCTV video on highways. Overall, it attains a high accuracy rate of 92.3%, underscoring theefficacy of the integrated approach in accurately recognizing and classifying accidents in real-time video footage.

[7] The system integrates various sensors, including accelerometers and GPS, with microcontrollers to swiftly detect vehicle issues. In the event of an accident, the system triggers GSM alerts to emergency services. This rapid transmission of accident details is crucial for facilitating quick responses, especially in remote areas. The automated detection process enhances efficiency, providing faster aid and contributing to improved safety on the roads. This integrated approach leverages technology to streamline the detection and reporting of vehicle-related incidents, emphasizing the importance of swift responses inemergency situations.

[8] The system employs a smartphone's front camera to monitor the driver's condition, specifically targeting signs of drowsiness. It captures a photo of the driver's face and analyzesthe frequency of eye opening and closing per minute. In the presence of abnormal patterns indicative of drowsiness, the system triggers a voiced warning message, such as "are you sleepy?", to alert the driver. Utilizing various smartphone sensors, including the front camera, ambient light sensor, gyro detector, accelerometer, and GPS functionality, the system processes the data to make accurate assessments. The reported accuracy rate of 60% indicatesits effectiveness in detecting and warning against potential driver drowsiness, contributing to enhanced road safety.

[9] The advanced accident detection system described utilizes technologies such as LIDAR, RADAR, and V2V (Vehicle-to-Vehicle) communication to minimize road casualties by providing early alerts to drivers, thereby reducing fatalities. This unified module integrates both pre-accident detection and post-accident alerts. By forecasting potential accidents and promptly notifying emergency services, the system contributes to enhancing overall road safety measures. The proactive approach of early detection and swift response aims to mitigate the severity of accidents and improve overall road safety outcomes.

[10] The study emphasizes the critical need to address traffic accidents in Saudi Arabia to reduce fatalities and financial losses. To tackle this challenge, the research proposes an Internet of Things (IoT)-based security framework that prioritizes driver privacy through encryption and efficient data transmission. The IoT system presented in the research is designed with lightweight cryptography, showcasing a secure and efficient solution for swift accident detection. By prioritizing privacy, the proposed framework aims to contribute significantly to improving road safety and enhancing resource efficiency in addressing traffic accidents in the context of Saudi Arabia.

[11] The Smart Traffic Accident Monitoring System employs a combination of GPS, GSM, ultrasonic, and vibration sensors to detect collisions and swiftly notify emergency units, addressing the growing concerns related to road accidents. Emphasizing the importance of the"golden hour" following accidents, particularly in rural areas, the system is designed for an immediate response. Notable features include alcohol detection, distance alerts, and notifications to relevant units. Additionally, the system proposes future enhancements, indicating a commitment to continuous improvement and the adoption of advanced featuresto enhance its effectiveness in monitoring and responding to traffic accidents.

[12] The global prevalence of road accidents, fueled by issues such as overcrowding and rule violations, results in substantial fatalities and financial losses. Critical factors like drunk driving, overspeeding, and distractions contribute to both immediate and prolonged fatalities. While existing systems incorporate technologies such as GSM/GPS modules and Convolutional Neural Networks (CNN) with Long Short-Term Memory (LSTM) models,they reveal limitations. This underscores the necessity for more effective accident detection methods to enhance road safety. The acknowledgment of current system limitations signals a call for advancements in technology and strategies to address the complex challenges associated with road accidents and improve overall safety measures.

[13] The proposed accident detection systems, while integrating various technologies, encounter limitations that need consideration. One such limitation is the lack of victimmedical history, leading to delays in providing aid. Additionally, there is a risk of overlookingminor accidents, which could result in resource wastage. The reliance on smartphones introduces challenges, including data noise that may affect accuracy, and there is a potential for technical issues if the phone is unavailable or damaged during accidents. These considerations highlight the importance of addressing not only the technological aspects but also the broader implications and potential drawbacks associated with the implementation of accident detection systems.

[14] The increasing prevalence of car usage has led to a rise in accidents, often attributed to speeding. Recognizing this challenge, there is a growing need for an automatic warning system. This particular system employs an integrated approach, utilizing Arduino, GSM (Global System for Mobile Communications), GPS (Global Positioning System), and MEMS (Micro-Electro-Mechanical Systems) sensors. The system is designed to swiftly identify and signal accidents through a two-level detection method. The incorporation of traditional safety tools, such as MEMS sensors, with modern accident detection techniques enhances theoverall response. This integration showcases the effectiveness of combining diverse sensors and adopting Internet of Things (IoT) approaches to improve accident detection and response mechanisms.

[15] This section reviews a range of accident detection techniques, incorporating deep learning, object detection, and hardware components. The implemented model, leveraging OpenCV, Keras, and TensorFlow, demonstrated notable performance with a commendable test accuracy of 92%, surpassing the 80% validation accuracy. The successful implementationhighlights the effectiveness of these technologies in achieving accurate accident detection, showcasing the potential for real-world applications in enhancing road safety.

[16] In this implementation, Artificial Intelligence (AI), specifically Convolutional Neural Networks (CNN), is leveraged to advance road accident detection, aligning with the goal of Vision Zero. The implemented model achieved a remarkable 92% test accuracy, surpassing the 80% validation accuracy. This study underscores the potential of transfer learning and deep learning techniques for binary image classification in the effective detection of road accidents. The results obtained are deemed satisfactory, and there are opportunities for furtherenhancement through the integration of real-world data and broader applicability, particularlyin remote monitoring and collaboration with emergency services.

[17] The paper introduces a method aimed at improving accident detection accuracy within video image detector systems. This enhancement involves analyzing vehicle traces and implementing level spacing distribution techniques, resulting in a substantial 90% improvement compared to other existing approaches. Additionally, the proposed methoddemonstrates the capability to identify lane deviations and aggregate traffic information. This innovation holds promise for advancing the precision and scope of accident detection systems, offering potential benefits in terms of both safety and traffic management.

[18] The study introduces an intelligent real-time traffic accident detection system that leverages a combination of vehicular ad-hoc networks and machine learning algorithms. Notably, the Random Forest algorithm is employed and achieves a high accuracy rate of 91.56% in effectively distinguishing between accidents and normal traffic conditions. This suggests the potential efficacy of the proposed system in promptly identifying and respondingto traffic accidents, contributing to enhanced road safety and more efficient traffic management.

[19] The paper unveils a machine learning and deep learning-based accident detection model with an impressive accuracy of 94.14%. This advanced system incorporates key frame extraction, vehicle distance analysis, K-means clustering, and Support Vector Machine (SVM) classification techniques, applied to traffic surveillance camera footage. The utilization of these sophisticated methods underscores the model's capability to effectively analyze and interpret complex visual data from traffic surveillance cameras, contributing to a high level of accuracy in detecting and classifying traffic accidents.

[20] The paper introduces a real-time road accident detection system that leverages deep learning models, achieving an accuracy of over 90%. The system employs YOLOv3 for crashdetection, Support Vector Machines (SVMs) for damage assessment, and machine learning techniques for forward collision detection. The emphasis on adaptability to Indian traffic conditions underscores the system's potential to address the unique challenges and characteristics of traffic scenarios in India, contributing to effective and accurate accident detection in real-time.

[21] The study suggests an autonomous accident detection system employing computational intelligence techniques on 2015 Istanbul highway data, demonstrating high accuracy of over 99% and room for further improvements in real-time deployment.

[22] The research focuses on the application of Convolutional Neural Networks (CNNs) for animal detection in natural environments, emphasizing the importance of wildlife monitoring in the face of human-induced ecological changes. Utilizing deep neural networks, specificallyCNNs, the study aims to automate the analysis of animal behavior. The research employs a dedicated dataset divided into training and test subsets to train the CNN model. The CNN operations, including input analysis, convolution, ReLU (Rectified Linear Unit), and pooling mechanisms, are outlined in the paper. The system showcases high precision in identifying animals and proves to be efficient in extracting valuable data from camera-trap images. This underscores the system's potential for conservation efforts and wildlife management.

[23] The paper focuses on mitigating human-animal collisions in Indian forests and proposes an alert system to improve road safety. The technological components include camera-trap networks, multilevel graph cuts, DCNN (Deep Convolutional Neural Network) features,motion sensors, IP cameras, and LED signboards. The system utilizes machine learning for image analysis, implementing a CNN framework that combines Prototype detection and CNNfor object detection. With an achieved detection accuracy of 82.5%, the system validates its performance using metrics such as average precision and speed, showcasing its effectiveness in addressing the challenges of human-animal collisions in Indian forest areas**.**

[24] The study focuses on using deep learning, particularly convoluted neural networks (CNNs), to detect animals on roads with the goal of reducing related accidents. The technology employed addresses previous models' shortcomings, specifically aiming to reducepower consumption. The model achieves a notable 91% accuracy. This high success rate positions the model as a promising and cost-effective solution for enhancing global road safety. The study is titled "Evaluating YOLO-based Object Detectors for Detecting Road-Killed Endangered Brazilian Animals.”

[25] The study is focused on automated animal detection on Brazilian roads, specifically usingYOLO-based detectors to reduce roadkill incidents. The research evaluates YOLO architectures, with a notable emphasis on the Scaled-YOLOv4 model, particularly its application on edge devices and validation with the BRA-Dataset. Scaled-YOLOv4 proves effective in detecting endangered animals, addressing challenges such as occlusion and camouflage. The model demonstrates strong performance, particularly in minimizing false negatives, showcasing its potential for effective road safety measures.

[26] The study focuses on using RCNNs (Region-based Convolutional Neural Networks) for wildlife monitoring, highlighting their significance in preventing human-wildlife conflictsand animal accidents. RCNNs process camera trap images and motion sensor data, employing layers like CONV, RELU, and POOL. The research attains an impressive 93.8% accuracy in automated animal identification, showcasing the effectiveness of deep convolutional neural networks. The RCNN system exhibits high precision, advocating its application for real-time wildlife protection measures.

[27] This research employs a modified YOLO v4 algorithm for detecting abnormal activities on roadways and railways, encompassing harassment and accidents, both during the day and night. The study utilizes the YOLO v4 algorithm in conjunction with Darknet, OpenCV, and deep learning techniques for swift and precise object detection. The YOLO v4 algorithm efficiently detects various objects, incorporating techniques such as dropout, batchnorm, and route layers to enhance performance. With detection times ranging between 0.02 to 0.09 seconds, the modified YOLO v4 surpasses prior versions and provides reliable results in identifying abnormal activities on roadways and railways.

[28] This document introduces a pedestrian detection system designed for driver assistance on highways with the objective of mitigating accidents involving pedestrians and animals. The system utilizes image processing, infrared cameras, Arduino controllers, and machine learning algorithms such as Cascade Random Forest and Kalman Filtering. Additionally, USB-connected LCD units are employed for display purposes. The system is capable of identifying sudden obstacles on highways, issuing warnings to drivers, adjusting vehicle speed, and displaying details about detected objects. It also focuses on enhancing image quality and addressing traffic congestion. Evaluated based on true and false positive rates, the system outperforms existing solutions, demonstrating superior accuracy in detecting andpreventing highway accidents.

[29] The document introduces an Animal Classifier System designed for video surveillance and forest monitoring using Raspberry Pi, addressing challenges in wildlife observation and species record-keeping. The system employs Python, TensorFlow models, OpenCV, and Raspberry Pi hardware with a camera module. It integrates Raspbian OS, NumPy, and pandasfor data management. The features include live animal classification from images, datastorage in a database for wildlife monitoring, identification of new species with segregationof their data for research purposes, and wireless control via the Rasp Controller mobile app. Powered by the MobileNet v3 model and extensive training data, the system accurately counts animals and objects within frames, displaying data on monitors or smartphones.

[30] The document details an animal detection system designed for Brazilian roads, leveragingmachine learning techniques such as KNN (K-Nearest Neighbors) and RF (Random Forest). The system utilizes cameras and microwave sensors to detect animals and promptly alerts drivers through a roadside signal. Developed in Python 2.7, the system interfaces with a warning signal through a Java 8 Client/Server application, operating on a 2.50 GHz Intel Corei7 processor with 6 GB RAM. Various color spaces, including RGB and HSV, are employed for feature extraction to enhance animal detection capabilities. Performance evaluation, measured using the F-measure metric, reached its peak at 0.6243 with the KNN algorithm, showcasing the system's potential for wildlife preservation and road safety.

[31] The study focused on binary image classification of road accidents using Finnish road surveillance images, employing transfer learning models such as MobileNetV2 and EfficientNetB1. TensorFlow was the primary tool for model implementation, and the researchaimed at establishing a proof-of-concept for potential online deployment. The models, trainedon a dataset comprising real and synthetic images, demonstrated the ability to distinguish between accident scenes and normal scenarios. In terms of accuracy, EfficientNetB1 outperformed MobileNetV2, achieving a mean Average Precision (mAP) of 0.89 and a Matthews Correlation Coefficient (MCC) of 0.78, while MobileNetV2 achieved an mAP of 0.87 and an MCC of 0.68. This highlighted the slightly superior performance of EfficientNetB1 in acci dent detection.

[32] The study focused on utilizing transfer learning models, MobileNetV2 and EfficientNetB1, in deep learning for binary image classification aimed at detecting potential road accidents. The models were trained on a dataset containing both synthetic and real images captured by a surveillance camera on Regional Road 102 in Finland. TensorFlow was the primary tool for model training, with Tensorflow-cpu 2.8.0 used for online deployment. Trained on a dataset of 134 real images, both models demonstrated satisfactory outcomes in distinguishing between accident and non-accident scenarios, emphasizing their potential for early detection. In terms of accuracy, EfficientNetB1 outperformed MobileNetV2, achievinga mean Average Precision (mAP) of 0.89 and a Matthews Correlation Coefficient (MCC) of 0.78, while MobileNetV2 recorded an mAP of 0.87 and MCC of 0.68. The EfficientNetB1 model displayed superior balance with fewer false positives, and both models averaged an F1-score of 0.88. The research suggests the potential integration of these models into aremote monitoring system for real-time accident detection.

[33] The document delves into deep learning techniques, specifically CNN, Deep Residual Network behavior, and VGG16, for the detection of abnormal driving behavior. Leveraging a Kaggle dataset comprising 22,424 driver frames, the study aims to enhance driver andpassenger safety through accurate behavior prediction. CNN is employed for image-level feature extraction, ResNet for its deep residual learning capabilities, and VGG16 for high- accuracy image classification with versatile model weights. In terms of accuracy and performance, the results indicate varying success rates: CNN achieved 48.44%, ResNetreached 87.44%, and VGG16 attained 79.51%. Notably, the proposed method using ResNet achieved the highest accuracy at 87.44%, surpassing the existing model's 82%. The study underscores the potential of deep learning models in predicting abnormal driving behavior, contributing to advancements in driver safety systems.

[34] The paper introduces an animal detection and classification system that utilizes machine learning and deep learning techniques to address human-animal conflicts in diverse environments. The system incorporates Convolutional Neural Networks (CNN), eXtreme Gradient Boosting (XGBoost), and Particle Swarm Optimization (PSO) for accurate animal detection and classification. Additionally, image normalization and dataset processing techniques are employed to enhance the overall accuracy of the models. The system integrates camera traps for image collection, and the data preprocessing steps involve normalization and wrangling of the dataset. The combination of CNN, XGBoost, and PSO algorithms allows for effective animal classification and species prediction, showcasing the potential of advanced technologies in mitigating conflicts between humans and animals in various settings.

[35] The paper introduces a machine learning model aimed at preventing road accidents by detecting driver drowsiness through the analysis of historical data and real-time eye states. Developed using Python, Jupyter notebook, and libraries like cv2 and tensorflow, the model incorporates drowsiness detection capabilities. It employs a classification algorithm trainedon eye images to monitor driver alertness, incrementing a sleep counter in response to eye closure and triggering an alarm when critical levels are reached. The system demonstrates a high accuracy rate of 90%, showcasing its effectiveness in identifying drowsiness and its potential to contribute to reducing road accidents and enhancing overall safety.

[36] The document presents a Car Crash Detection System using Machine Learning and Deep Learning, with a focus on employing Convolutional Neural Network (CNN) for impactevaluation and comparing the efficiency of Logistic Regression and Random Forest algorithms. Utilizing Python and Google Colab, the study integrates these algorithms to analyze live dash cam data and assess car crash severity and damage. The system achieves an 89.93% accuracy in car damage assessment using CNN. Logistic Regression and Random Forest algorithms yield accuracies of 68.23% and 75.63%, respectively, for crash severity. The study validates the efficacy of each algorithm through performance metrics, including precision, recall, F1 Score, sensitivity, and specificity.

[37] The document outlines the tools and technology used in a study focused on road accident detection and severity classification. The study employs Python, Keras 2.1.0 with a Tensorflow 1.14.0 backend, and scikit-learn 0.21.3, utilizing the ADADELTA optimizer for training deep neural networks. The features and accuracy/performance of the methods employed are discussed. Traditional handcrafted features, CNN, LSTM, and AE are considered. Notably, CNN features combined with SVM achieve the highest accuracy at 85.72% and an average F1 score of 79.10%. The research also highlights instances where Random Forest (RF) outperforms Support Vector Machine (SVM) in classificationperformance.

[38] The document introduces an Animal Classifier System designed for video surveillance and forest monitoring using Raspberry Pi, addressing challenges in wildlife observation and species record-keeping. The system employs Python, TensorFlow models, OpenCV, and Raspberry Pi hardware with a camera module. It integrates Raspbian OS, NumPy, and pandasfor data management. The features include live animal classification from images, identification of new species with segregationof their data for research purposes, and wireless control via the Rasp Controller mobile app. Powered by the MobileNet v3 model and extensive training data, the system accurately counts animals and objects within frames, displaying data on monitors or smartphones.

[39] The study focuses on using RCNNs (Region-based Convolutional Neural Networks) for wildlife monitoring, highlighting their significance in preventing human-wildlife conflictsand animal accidents. RCNNs process camera trap images and motion sensor data, employinglayers like CONV, RELU, and POOL. The research attains an impressive 93.8% accuracy in automated animal identification, showcasing the effectiveness of deep convolutional neural networks. The RCNN system exhibits high precision, advocating its application for real-time wildlife protection measures.

2.1 Deep Learning Models

A. Yolo V8 Small

"YOLOv8 Small" refers to a variant of the YOLO (You Only Look Once) object detection algorithm specifically optimized for detecting small objects within images or video frames. This version of YOLO utilizes a smaller and more efficient architecture compared to its counterparts, making it suitable for scenarios where computational resources are limited or real-time performance is crucial. YOLOv8 Small retains the core principles of the YOLO algorithm, including its ability to detect multiple objects in a single pass through the network, while focusing on improving accuracy and speed specifically for small-sized objects. This makes it valuable for applications such as surveillance, robotics, and autonomous vehicles where the precise detection of small objects is essential.

B. Yolo V8 Large

"YOLOv8 Large" represents a variant of the YOLO (You Only Look Once) object detection algorithm optimized for high accuracy and robustness, particularly suited for scenarios where detecting larger objects with high precision is crucial. This version of YOLO incorporates a larger and more complex architecture compared to its counterparts, allowing for deeper and more detailed feature extraction, which can lead to improved detection performance, especially for larger objects. YOLOv8 Large maintains the core principles of YOLO, including real-time processing and detection of multiple objects in a single pass through the network, while focusing on maximizing accuracy and handling complex scenes with larger objects. This makes it valuable for applications such as surveillance, autonomous vehicles, and medical imaging, where precise and reliable object detection is essential for decision-making.

Chapter 3
Software and Hardware Requirement Specification

3.1 Functional Requirements

The functional requirements for the project encompass :

- Alert Generation: The system should generate timely alerts to drivers and authorities upon detection of potential hazards, providing information on the type and location of the hazard.

- User Interface: The system should have a user-friendly interface for drivers to receive alerts and access avoidance strategies, ensuring ease of use and accessibility.

- Integration with Existing Infrastructure: It should seamlessly integrate with existing road infrastructure, such as traffic cameras, sensors, and communication networks, to facilitate data exchange and coordination.

- Scalability: The system should be scalable to accommodate varying levels of traffic and environmental conditions, ensuring consistent performance across different regions and road types.

- Data Management: It should efficiently manage and process large volumes of data, including image data from cameras and sensors, to support real-time detection and recognition tasks.

- Training and Updating: The system should support training and updating of deep learning models, enabling continuous improvement and adaptation to changing conditions and emerging threats.

- Performance Monitoring: It should monitor the performance of the system, including detection accuracy, response time, and false alarm rate, to ensure reliability and effectiveness in real-world scenarios.

- Compliance and Standards: The system should comply with relevant regulations and standards for road safety and data privacy, ensuring ethical and legal compliance in its operation.

By fulfilling these functional requirements, the project aims to develop a robust and effective system for car accident detection, wildlife recognition, and avoidance, contributing to enhanced road safety.

3.2 Non-functional Requirements

- Non-functional requirements specify the criteria that describe the system's operation rather than its specific behaviors. Here are the non-functional requirements for the project:

- Performance : The system should exhibit high performance, with minimal latency in detecting hazards and providing alerts to users. It should be capable of handling high volumes of data and concurrent user requests efficiently.

- Accuracy: The system's detection and recognition algorithms should achieve high levels of accuracy in identifying vehicles, pedestrians, and wildlife, minimizing false positives and negatives.

- Reliability: The system should be reliable and available for use at all times, with minimal downtime or service interruptions. It should have failover mechanisms in place to ensure continuous operation in the event of hardware or software failures.

- Scalability: The system should be scalable to accommodate increases in user traffic and data volume over time. It should be able to scale both vertically (increasing resources on a single server) and horizontally (distributing workload across multiple servers) as needed.

- Security: The system should adhere to stringent security measures to protect sensitive data and ensure the integrity and confidentiality of user information. This includes encryption of data in transit and at rest, access control mechanisms, and regular security audits.

- Usability: The system should be user-friendly and intuitive, with clear and concise interfaces for both drivers and system administrators. It should support multiple languages and accessibility features to accommodate diverse user needs.

- Maintainability: The system should be easy to maintain and update, with modular architecture and well-documented code. It should support version control and automated testing to facilitate continuous integration and deployment.

3.3 Hardware Requirements

- Processor: Quad Core, 2.0 GHz or greater
- Memory: 8 GB of RAM or more.
- Hard Disk: 20 GB of available space or more
- Android Phone with camera

3.4 Software Requirements

- Windows 7+
- Google Colab
- Python prerequisite

3.5 Cost Estimation

Our cost breakdown structure as shown in the following table:

Table 1. Cost Estimation of the proposed system

Unit No.	Item Name	Quantity	Total Estimated cost in Rs (for 6 months)
1	RoboFlow Premium	1	1000/- (on demand)
3	Deep learning framework ie. TensorFlow	-	500/-
4	Accident Scenario	-	500/-

Total Cost Estimated = 1000 + 500 + 500 = Rs. 2000 /-

Chapter 4
Design

4.1 High Level Design

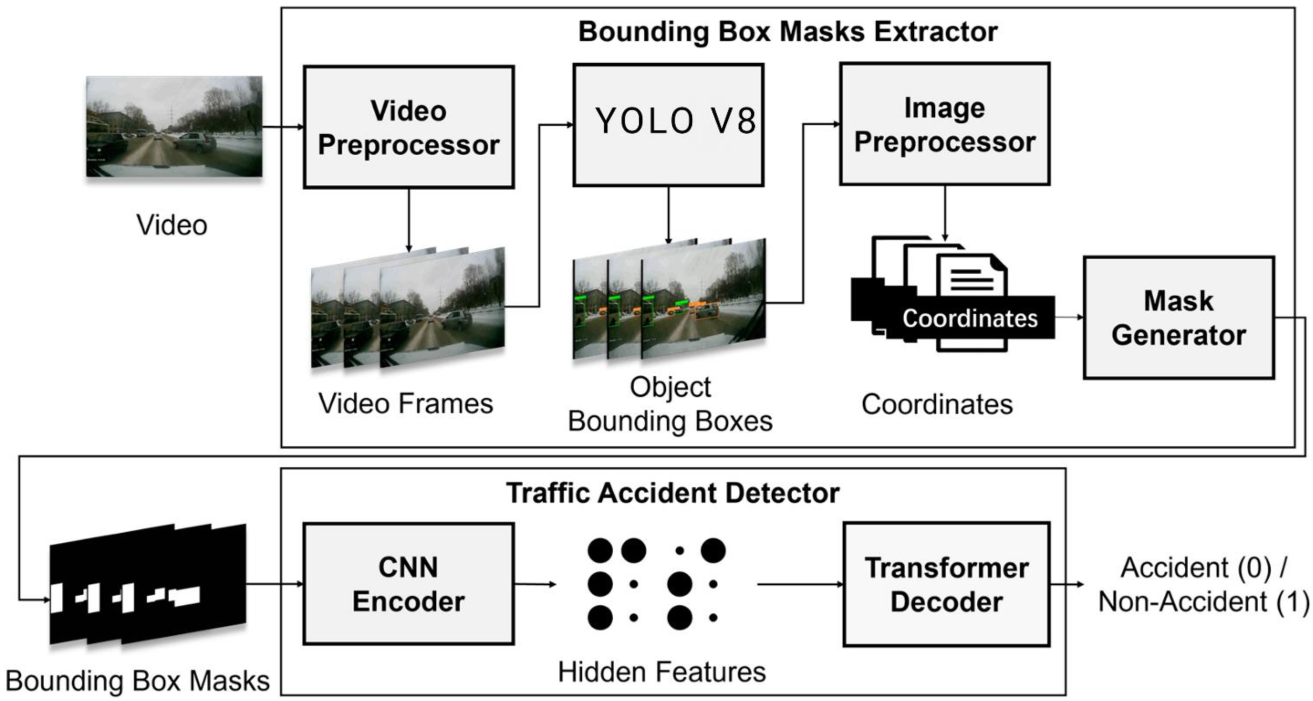


Figure 3. High Level Design of car accident detection, wildlife recognition and avoidance system

The above high-level design of the car accident detection and wildlife recognition system outlines the overall architecture and key components of the system. Here's a description of the high-level design:

1. System Architecture:

- The system follows a modular architecture, with distinct components responsible for different functionalities.
- It employs a client-server architecture, where the client-side interface interacts with the server-side processing modules.

2. Core Components:

- Input Data Processing Module: Responsible for processing input data from various sources such as live CCTV feeds, photos, or videos. It preprocesses data to extract relevant features for detection and recognition.
- Detection Module: Handles the detection of car accidents and recognition of wildlife using machine learning or computer vision algorithms. It analyzes preprocessed data to identify incidents.
- Notification Module: Manages the generation and dispatch of email notifications to emergency services and family members upon detection of incidents.
- User Interface Module: Provides a user-friendly interface for users to interact with the system. It includes options for selecting detection preferences and input types.
- Authentication Module: Handles user authentication and authorization. It verifies user credentials and controls access to system functionalities based on user roles.

3. Data Flow:

- Input data from various sources is processed by the Input Data Processing Module to extract relevant features.
- Processed data is then fed into the Detection Module, which analyzes it using machine learning or computer vision techniques to detect incidents such as car accidents or wildlife on the road.
- Upon detection of incidents, the Notification Module generates and dispatches email notifications to emergency services and family members, providing details of the incidents.
- Users interact with the system through the User Interface Module, which provides a graphical interface for selecting detection options, input types, and viewing detection results.

4. Integration Points:

- The system integrates with external services such as email servers for sending notifications to emergency services and family members.
- It may also integrate with external databases or APIs for accessing additional data or services related to incident response.

5. Scalability and Flexibility:

- The system is designed to be scalable, allowing for the addition of new functionalities or modules in the future.
- It is also flexible, with configurable parameters and settings that can be adjusted to suit different use cases or environments.

Overall, the high-level design provides a blueprint for the architecture and functionality of the car accident detection and wildlife recognition system, outlining how different components interact to achieve the system's objectives..

4.1.1 System Architecture

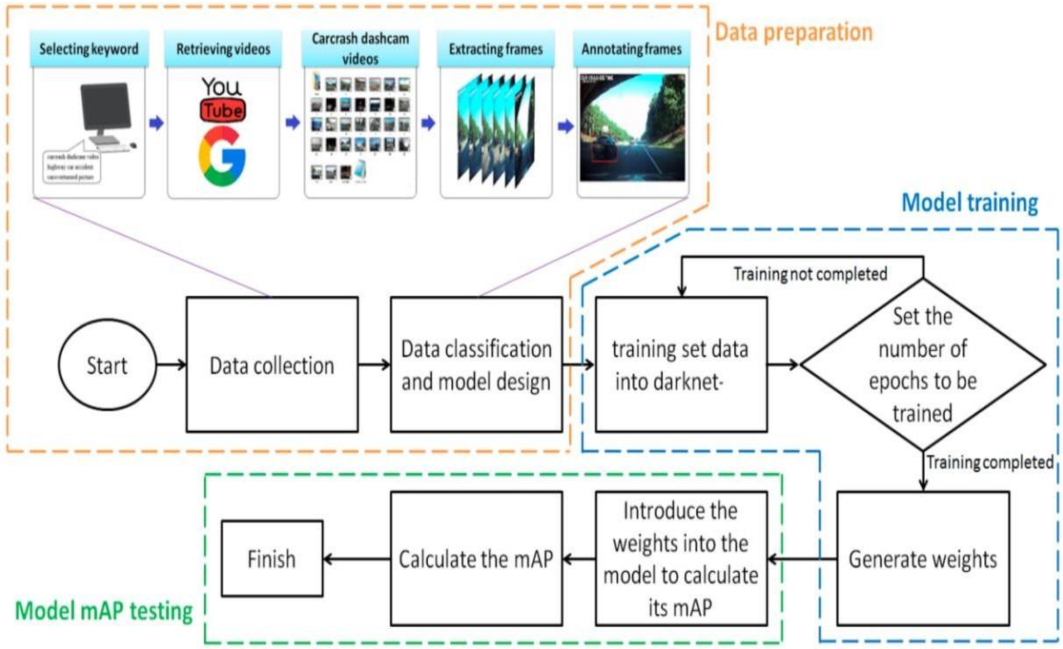


Figure 4. System Architecture of the proposed system

The car accident detection and wildlife recognition system architecture leverages YOLO v8 for real-time object detection. Here's a concise overview:

1. Input Processing: Captures and preprocesses data from diverse sources like live CCTV feeds, photos, or videos.

2. Object Detection: Utilizes YOLO v8 deep learning algorithm to detect cars, pedestrians, and wildlife like cows, sheep, dogs, and bulls.

3. Incident Severity Classification: Categorizes detected car accidents into severity levels: minor, moderate, and critical.

4. Notification System: Sends email alerts to emergency services and relevant authorities for critical incidents, facilitating swift response.

5. User Interface: Provides a user-friendly interface for interaction, enabling users to select detection preferences and input types.

6. Authentication: Manages user authentication and authorization to ensure secure access to system functionalities.

This architecture ensures efficient real-time incident detection and notification, enhancing road safety measures.

4.1.2 Use -case Diagram

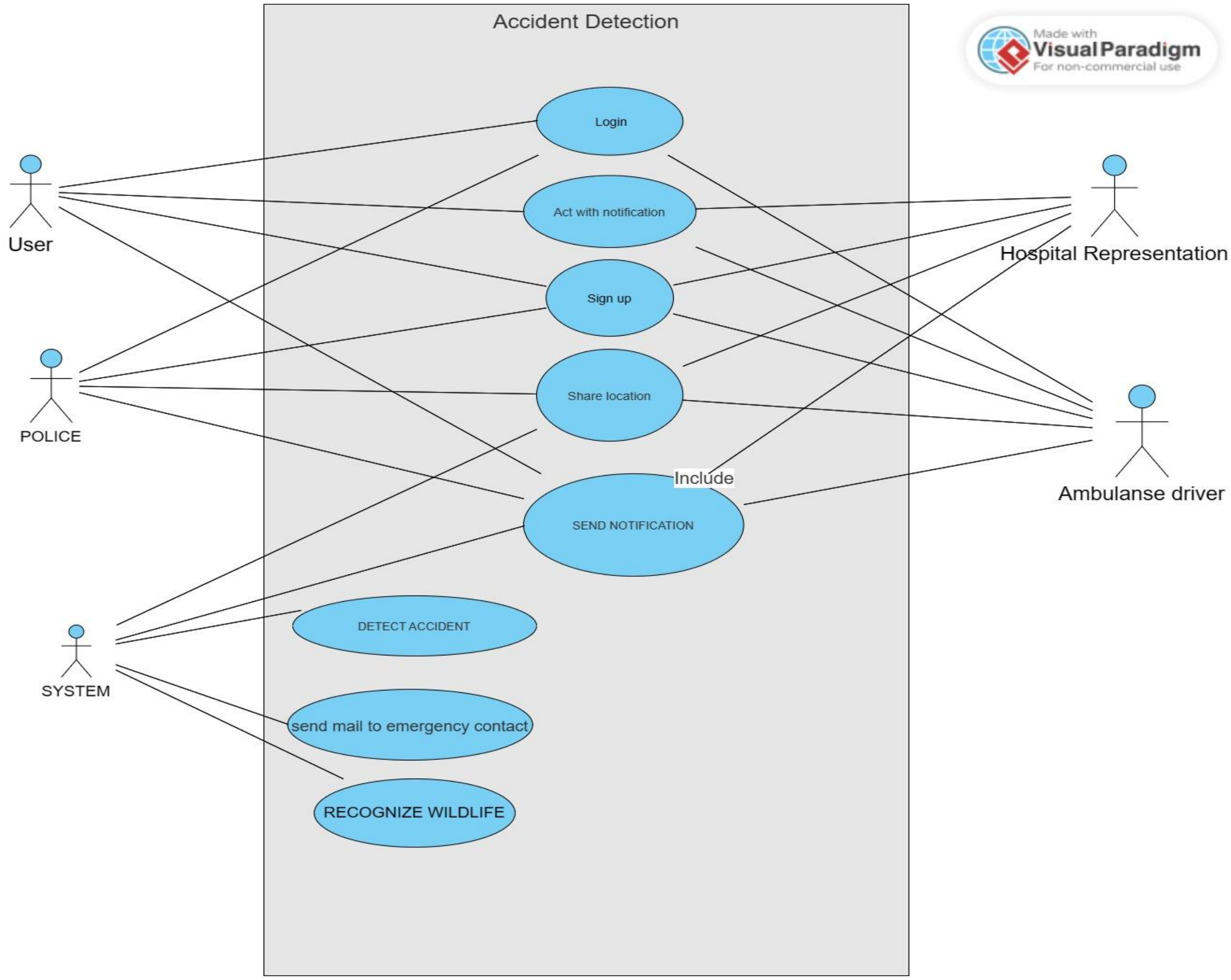


Figure 5. Use Case Diagram

4.2 Class Diagram

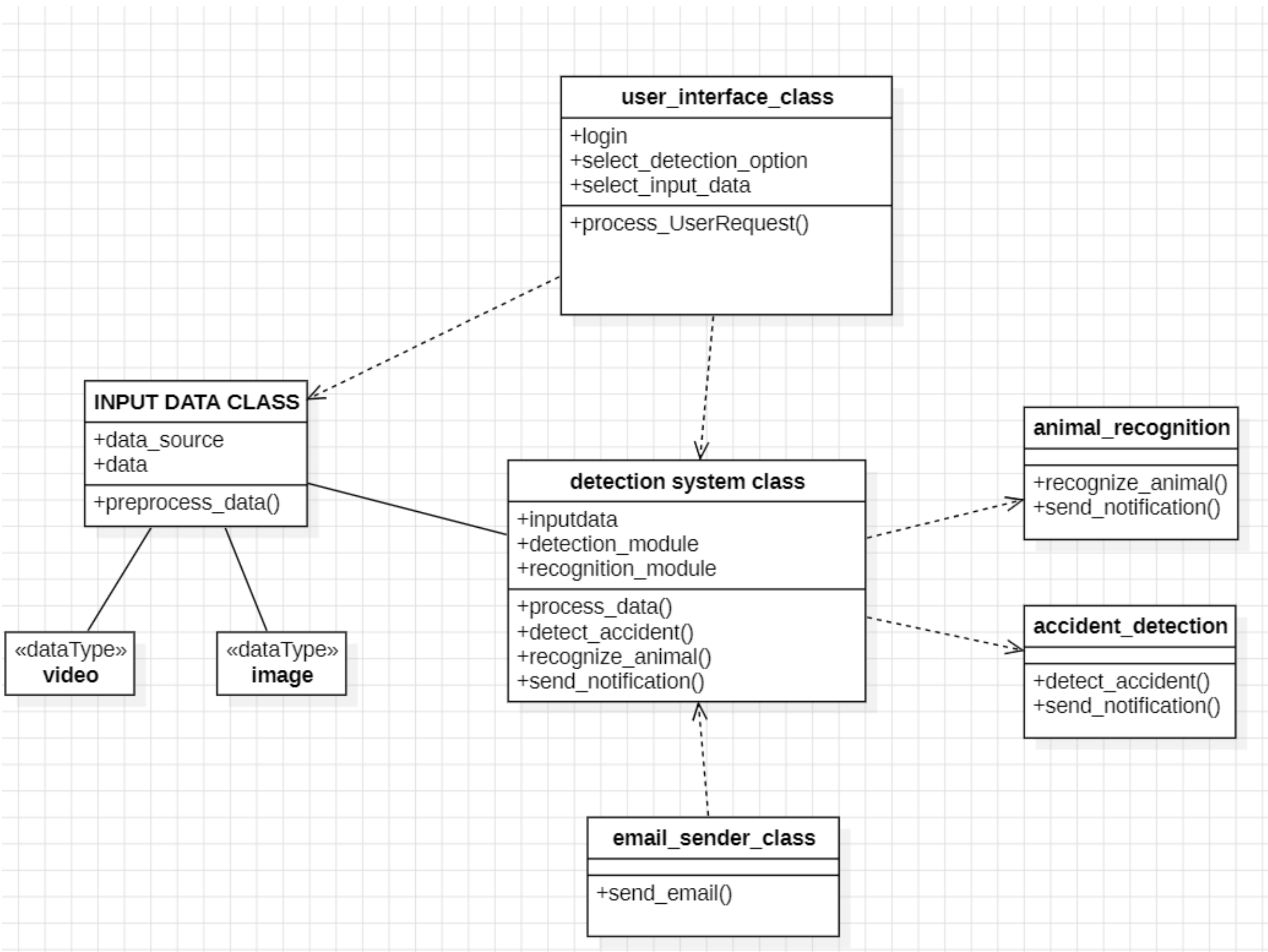


Figure 6. Class Diagram

1. **Core Modules**:

   The core modules contain functionalities of fetching and execution of the machine learning models. It also defines a schema for data storage and data handling. The below data flow figures describe this process.

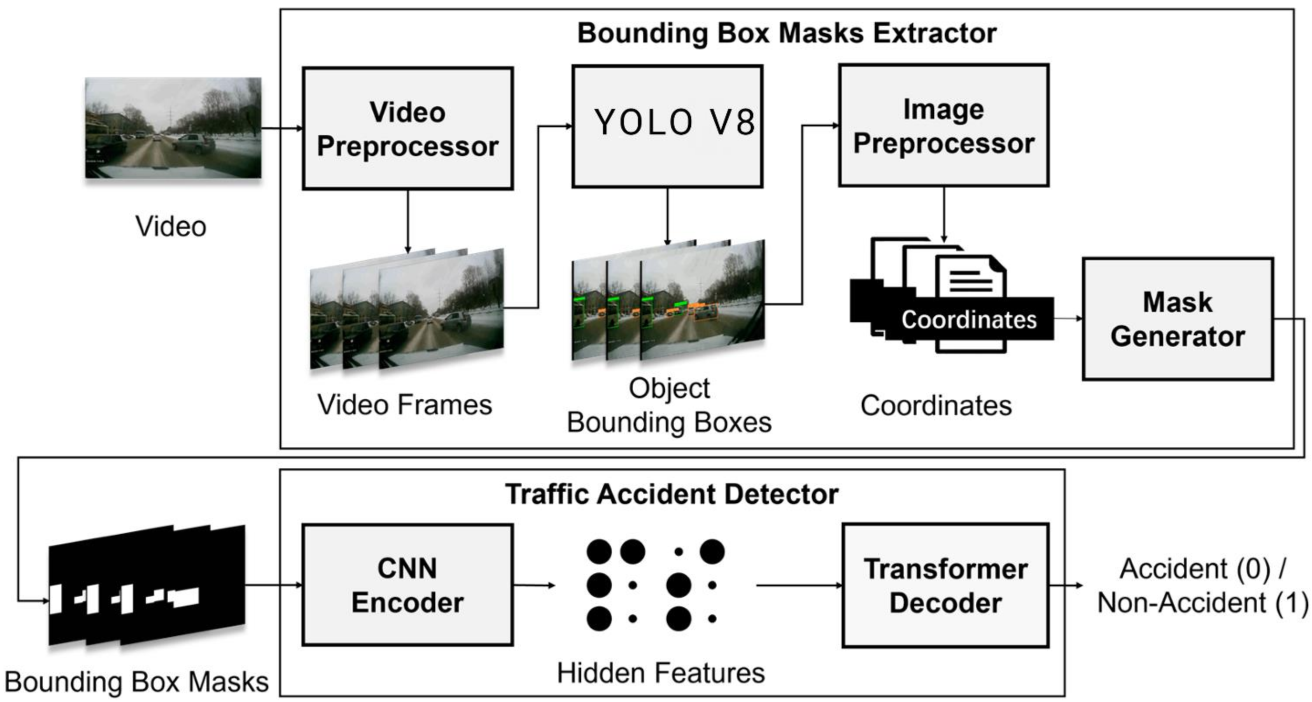


Figure 7. Core Modules

4.2.2 Use Case Diagram

The use case diagram represents the interactions between actors (users and external systems) and the system itself. Here's a description of the components of the use case diagram:

1. Actors:

- User: Represents a user interacting with the system. The user can log in, select detection options, process data, and log out.
- Emergency Services: Represents external services such as police and hospital staff who may be notified in case of incidents.
- Family Members: Represents family members of the user who may also be notified in case of incidents.

2. Use Cases:

- Login: Allows the user to log in to the system.
- Select Detection Option: Allows the user to choose between detecting car accidents or recognizing wildlife.
- Select Input Type: Allows the user to select the input type (live CCTV feed, photo, or video).
- Process Data: Initiates the processing of input data for detection or recognition.
- Detect Car Accident: Triggers the car accident detection process.
- Recognize Wildlife: Triggers the wildlife recognition process.
- Send Email Notification: Sends email notifications to emergency services and family members.
- Logout: Allows the user to log out of the system.

3. Associations:

- The User actor is associated with several use cases, including Login, Select Detection Option, Select Input Type, Process Data, and Logout. This indicates that the user initiates these actions.
- The Emergency Services actor is associated with the Send Email Notification use case, indicating that they are recipients of email notifications in case of incidents.
- The Family Members actor is also associated with the Send Email Notification use case for

the same reason.

- The Process Data use case is associated with the Detect Car Accident and Recognize Wildlife use cases, indicating that data processing is a prerequisite for these actions.
- The Send Email Notification use case is associated with the Detect Car Accident and Recognize Wildlife use cases, indicating that email notifications are triggered based on detection events.

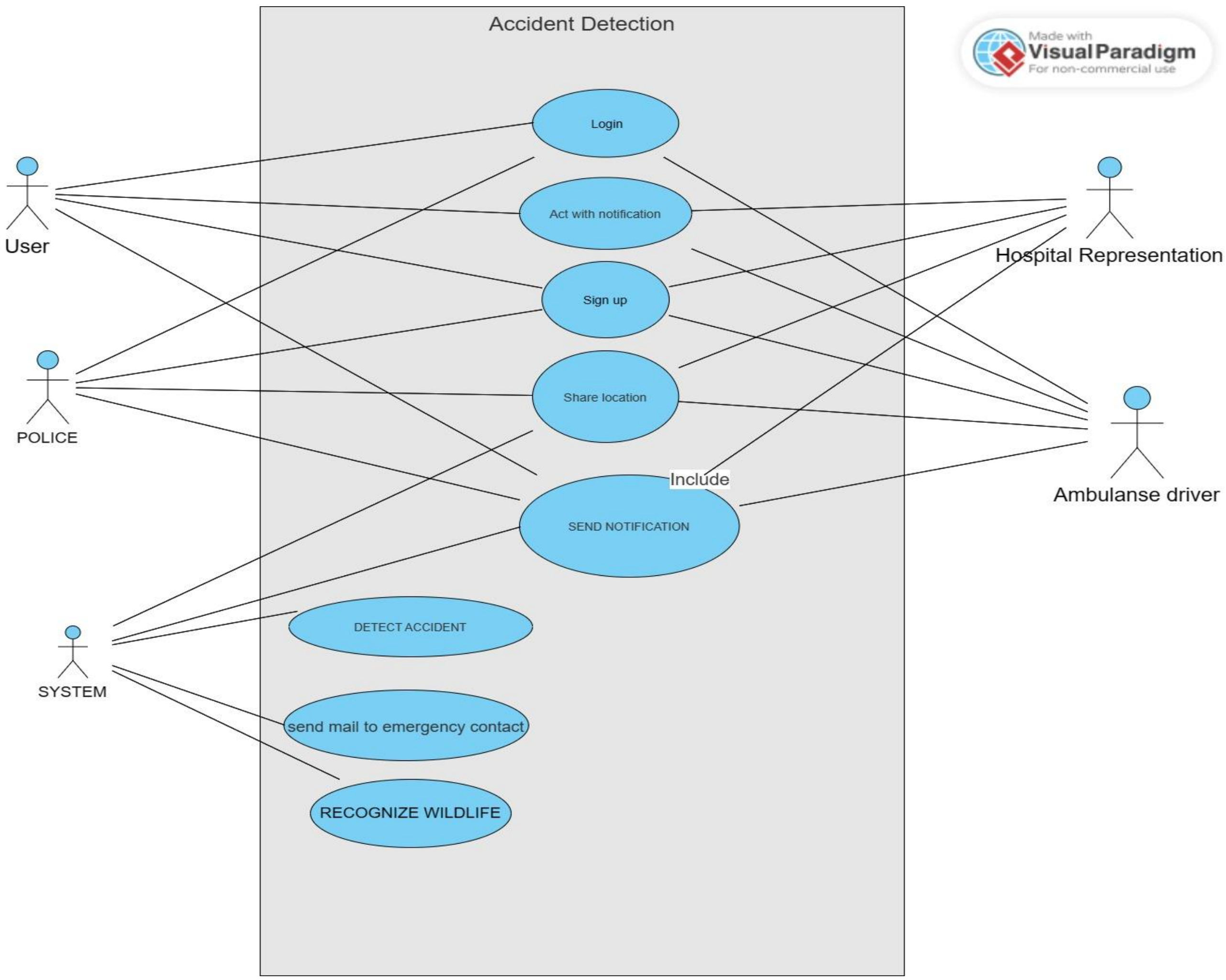


Figure 8. Detailed Use Case Diagram

4.2.3 Class Diagram

The class diagram provided represents the design of a system that can detect and recognize animals, as well as detect accidents. Here's a breakdown of the key classes and their relationships:

1. input_data_class: This class represents the input data for the system, which can be either video or image data. It has methods to preprocess the data before it is used by the detection system.

2. user_interface_class: This class represents the user interface of the system. It has methods for logging in, selecting detection options, and processing user requests.

3. detection_system_class: This is the core of the system, responsible for processing the input data and detecting various events. It has the following sub-modules:

- inputdata_module: Handles the input data.
- detection_module: Responsible for detecting events in the input data.
- recognition_module: Handles the recognition of animals in the input data.
- process_data(): Processes the input data.
- detect_accident(): Detects accidents in the input data.
- recognize_animal(): Recognizes animals in the input data.
- send_notification(): Sends notifications based on the detection and recognition results.

4. animal_recognition: This class is responsible for recognizing animals in the input data. It has the following methods:

-recognize_animal(): Implements the animal recognition logic.
- send_notification(): Sends notifications based on the recognition results.

5.accident_detectio: This class is responsible for detecting accidents in the input data. It has the following methods:

- detect_accident(): Implements the accident detection logic.
-send_notification() Sends notifications based on the detection results.

6. email_sender_class: This class is responsible for sending email notifications based on the results from the detection and recognition modules.

The diagram shows the relationships between these classes and their methods, illustrating the overall architecture of the system. The user interface class interacts with the detection system class to process user requests, and the detection system class coordinates the various sub-modules to handle the input data, detect events, and recognize animals. The email sender class is used to send notifications based on the results from the detection and recognition modules.

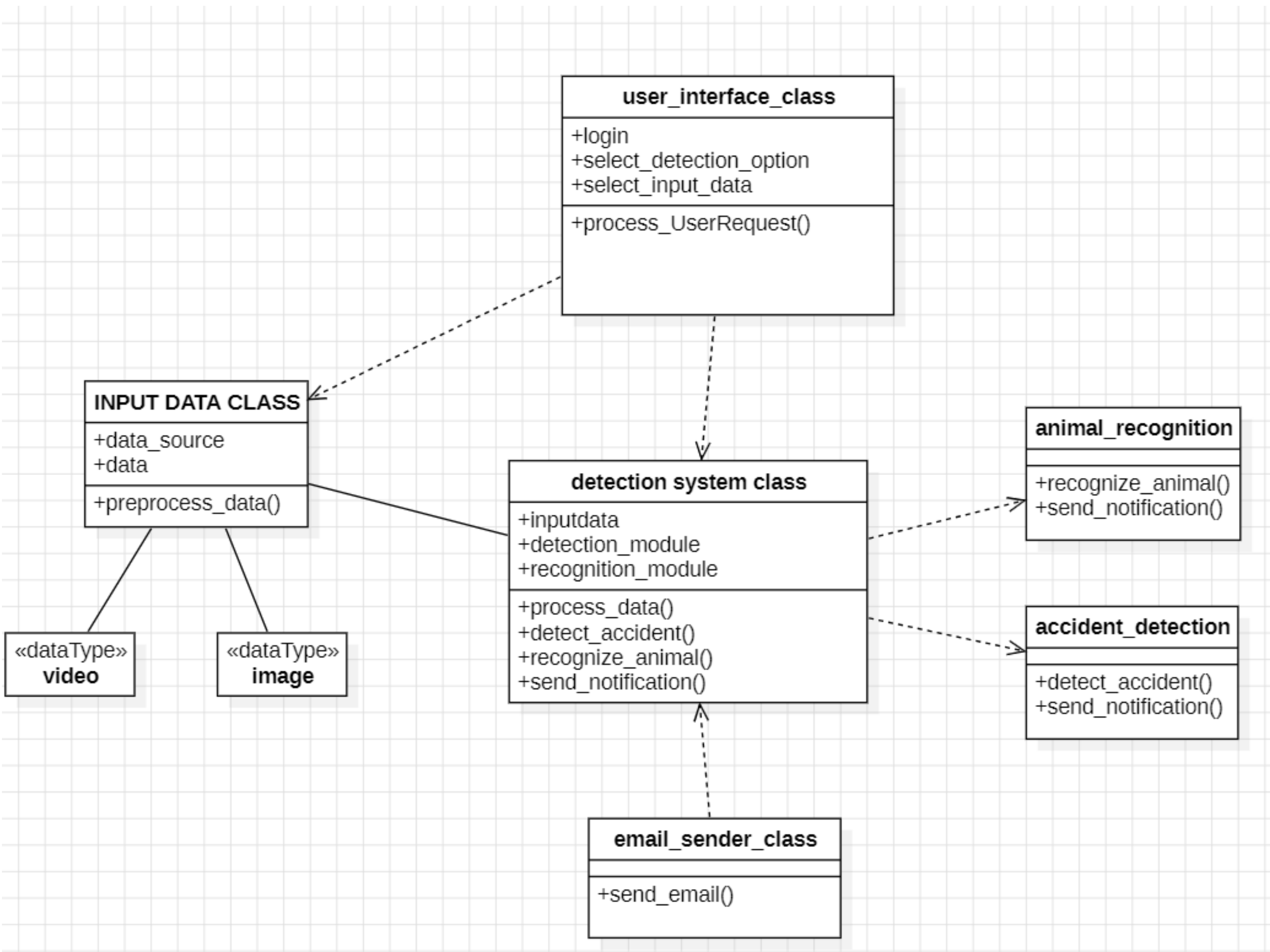


Figure 9. Detailed Class Diagram

4.2.4 Sequence Diagram

The sequence diagram illustrates the flow of interactions between system components and external entities when an accident is detected or an animal is recognized on the road. Upon detection, the system triggers a sequence of events starting with data processing and analysis by the detection and recognition modules. Once the incident is confirmed, the system retrieves the contact information of emergency services and family members and generates email notifications containing relevant details. These emails are then sent out via the email sending service, with confirmation messages logged for monitoring purposes. Finally, the system updates its status and awaits further inputs, ensuring a seamless and timely response to incidents on the road.

Figure 10. Sequence Diagram

Chapter 5
Implementation

5.1 Overview of Technologies Used

The car accident detection and wildlife recognition system leverages a blend of advanced technologies to achieve its objectives effectively. Here's an overview of the technologies used:

- Deep Learning:
  - *YOLO v8 (You Only Look Once)*: A state-of-the-art deep learning algorithm used for real-time object detection. YOLO v8 enables the system to detect cars, pedestrians, and wildlife accurately and efficiently in various input data streams.

- Computer Vision:
  - Computer vision techniques are applied for preprocessing input data and extracting pertinent features for detection and recognition tasks. These techniques include image processing methods such as edge detection, segmentation, and feature extraction, enhancing the accuracy of object detection and classification.

- Machine Learning:
  - Machine learning algorithms are employed for incident severity classification. The system categorizes detected car accidents into severity levels (minor, moderate, and critical) based on extracted features, enabling effective prioritization of incident response.

- Email Notification:
  - Integration with email services allows the system to send real-time email alerts to emergency services and relevant authorities upon detection of critical incidents. This ensures timely response and intervention during emergencies.

- User Interface Development:
  - The system features a user-friendly interface developed using web technologies or graphical user interface (GUI) frameworks. This interface enables users to interact seamlessly with the system, select detection preferences, and visualize detection results intuitively.

- Cloud Computing:
  - Cloud computing platforms may be utilized for scalable and efficient processing, including model training, inference, and data storage. Leveraging cloud infrastructure facilitates seamless integration and deployment of the system across diverse environments.

- Data Preprocessing and Augmentation:
  - Roboflow: Positioned as a data preprocessing and augmentation tool, Roboflow plays a vital role in managing and enhancing image datasets. It offers functionalities such as resizing, cropping, rotating, and augmenting images to improve dataset diversity and quality. Roboflow's integration with various deep learning frameworks streamlines the preprocessing pipeline, optimizing the data for training robust and accurate machine learning models.

By harnessing these technologies, the car accident detection and wildlife recognition system achieves precise and efficient incident detection, enabling proactive measures to ensure road safety and mitigate risks effectively.

5.2 Implementation details of modules

Certainly, here's an overview of the implementation details for each module in the car accident detection and wildlife recognition system:

1. Input Data Processing Module:
   - Implemented using Python libraries such as OpenCV, Pytorch and ultralytics for image and video processing.
   - Utilizes preprocessing techniques such as resizing, normalization, and data augmentation to prepare input data for detection and recognition tasks.
   - Integrates with Roboflow for managing and enhancing image datasets, ensuring data quality and diversity.

2. Detection Module:
   - Implements YOLO v8 algorithm for real-time object detection using deep learning frameworks such as TensorFlow .
   - Trains and fine-tunes the YOLO v8 model on annotated datasets of car accidents and wildlife using GPU-accelerated hardware or cloud computing resources.
   - Optimizes model hyperparameters and architecture for efficient and accurate detection performance.

3. Notification Module:
   - Developed using Python libraries for email handling, such as smtplib or the email.mime module.
   - Integrates with SMTP servers for sending email notifications to emergency services and relevant authorities upon detection of critical incidents.
   - Configures email templates with details of detected incidents, including location, severity, and timestamp.

4. User Interface Module:

- Implements a web-based interface using frontend technologies such as HTML, CSS, and JavaScript, along with frameworks like React.js or Angular.
- Provides interactive components for users to select detection options, input types, and view detection results in real-time.
- Integrates with backend services through RESTful APIs for data exchange and communication.

5. Authentication Module:

- Implements user authentication and authorization using authentication libraries or frameworks such as Flask-Security or Django-Auth.
- Enforces access control policies based on user roles and permissions, ensuring secure access to system functionalities.
- Integrates with user databases or identity providers for user management and authentication.

These implementation details provide a comprehensive overview of how each module is developed and integrated into the car accident detection and wildlife recognition system, ensuring efficient and effective operation to achieve system objectives.

5.3 Difficulties encountered and Strategies used to tackle

During the development of the car accident detection and wildlife recognition system, you may encounter several challenges. Here are some potential difficulties and strategies to tackle them:

- Data Collection and Preprocessing: Difficulty in obtaining diverse and representative datasets for training the YOLO v8 model. Strategy: Collect data from multiple sources and augment it to increase diversity. Implement robust preprocessing techniques to clean and standardize the data.

- Algorithm Selection and Tuning: Difficulty in selecting the right deep learning algorithms and parameters for accurate detection. Strategy: Conduct thorough research to understand the strengths and weaknesses of different algorithms. Experiment with various configurations and fine-tune parameters through iterative testing.

- Model Training and Optimization: Difficulty in training the YOLO v8 model efficiently due to resource constraints or limited computational power. Strategy: Utilize cloud-based platforms or GPU-accelerated hardware for faster training. Optimize the model architecture and training pipeline to reduce computational overhead.

- Real-time Performance: Difficulty in achieving real-time performance for object detection, especially on resource-constrained devices. Strategy: Implement optimization techniques such as model quantization, pruning, or using lightweight architectures to improve inference speed. Profile and optimize critical sections of the code for efficiency.

- Integration and Deployment: Difficulty in integrating the detection system with other components and deploying it in real-world environments. Strategy: Develop robust APIs or interfaces for seamless integration with other systems. Conduct thorough testing and validation in simulated and real-world scenarios before deployment.

By anticipating these challenges and implementing appropriate strategies, you can overcome obstacles and successfully develop a robust and effective car accident detection and wildlife recognition system.

Chapter 6
Testing and Results

6.1 Testing

Testing plays a crucial role in ensuring the reliability, accuracy, and effectiveness of the car accident detection and wildlife recognition system. Here's a description of the testing process for the system:

- Unit Testing:
    - Individual components of the system, such as the input data processing module, detection module, notification module, and user interface module, are tested in isolation.
    - Unit tests verify the functionality of each component according to its specifications, ensuring that it behaves as expected.

- Integration Testing:
    - Once individual components pass unit testing, they are integrated into the system as a whole.
    - Integration tests are conducted to verify that the components work together seamlessly and that data flows correctly between them.
    - This testing phase ensures that the system functions as intended and that all components communicate effectively.

- Functional Testing:
    - Functional tests are conducted to evaluate the system's functionality from an end-user perspective.
    - Test scenarios are designed to simulate real-world usage scenarios, such as detecting car accidents and recognizing wildlife under various conditions.
    - The system's response to different input data types, detection options, and severity levels is thoroughly tested to ensure accurate and reliable performance.

- Performance Testing:
  - Performance tests assess the system's ability to handle a large volume of data and perform detection and recognition tasks efficiently.
  - Factors such as processing speed, resource utilization, and scalability are evaluated to ensure that the system meets performance requirements under different load conditions.

By conducting thorough testing across all stages of development, the car accident detection and wildlife recognition system can ensure its reliability, accuracy, and effectiveness in real-world deployment, ultimately contributing to enhanced road safety and traffic management.

6.2 Evaluation Metric and Performance Analysis

Train the YOLOv8 large model on the dataset specified in the "data.yaml" file for 100 epochs, with a batch size of 124, and save the final trained model

The training will be performed on a YOLOv8 object detection model, which is a state-of-the-art deep learning model for real-time object detection. The choice of the YOLOv8 large model suggests that the dataset and task complexity require a larger and more powerful model to achieve good performance.

```
      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
     99/100      10.2G     0.4052     0.2761     0.9294         71        224: 100% 12/12 [00:11<00:00,  1.09it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 2/2 [00:02<00:00,  1.46s/it]
                   all        350        476      0.722       0.54      0.657       0.49

      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
    100/100      10.2G     0.3933     0.2687     0.9173         87        224: 100% 12/12 [00:10<00:00,  1.15it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 2/2 [00:04<00:00,  2.07s/it]
                   all        350        476      0.579      0.683      0.663      0.491

100 epochs completed in 0.530 hours.
Optimizer stripped from /content/drive/MyDrive/Collision/final_training2/weights/last.pt, 87.6MB
Optimizer stripped from /content/drive/MyDrive/Collision/final_training2/weights/best.pt, 87.6MB

Validating /content/drive/MyDrive/Collision/final_training2/weights/best.pt...
Ultralytics YOLOv8.0.212 🚀 Python-3.10.12 torch-2.1.0+cu118 CUDA:0 (Tesla T4, 15102MiB)
Model summary (fused): 268 layers, 43610463 parameters, 0 gradients, 164.8 GFLOPs
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 2/2 [00:05<00:00,  2.54s/it]
                   all        350        476      0.636      0.664      0.675      0.501
                  fire        350        123       0.69      0.761      0.771      0.464
                 minor        350         17      0.541      0.624      0.564      0.517
              moderate        350         36      0.586      0.472      0.517      0.408
           no_accident        350        107      0.643      0.626      0.684      0.565
                severe        350        193      0.718      0.834      0.841      0.552
Speed: 0.0ms preprocess, 2.7ms inference, 0.0ms loss, 2.7ms postprocess per image
Results saved to /content/drive/MyDrive/Collision/final_training2
💡 Learn more at https://docs.ultralytics.com/modes/train
```

Figure 11.Model Summary

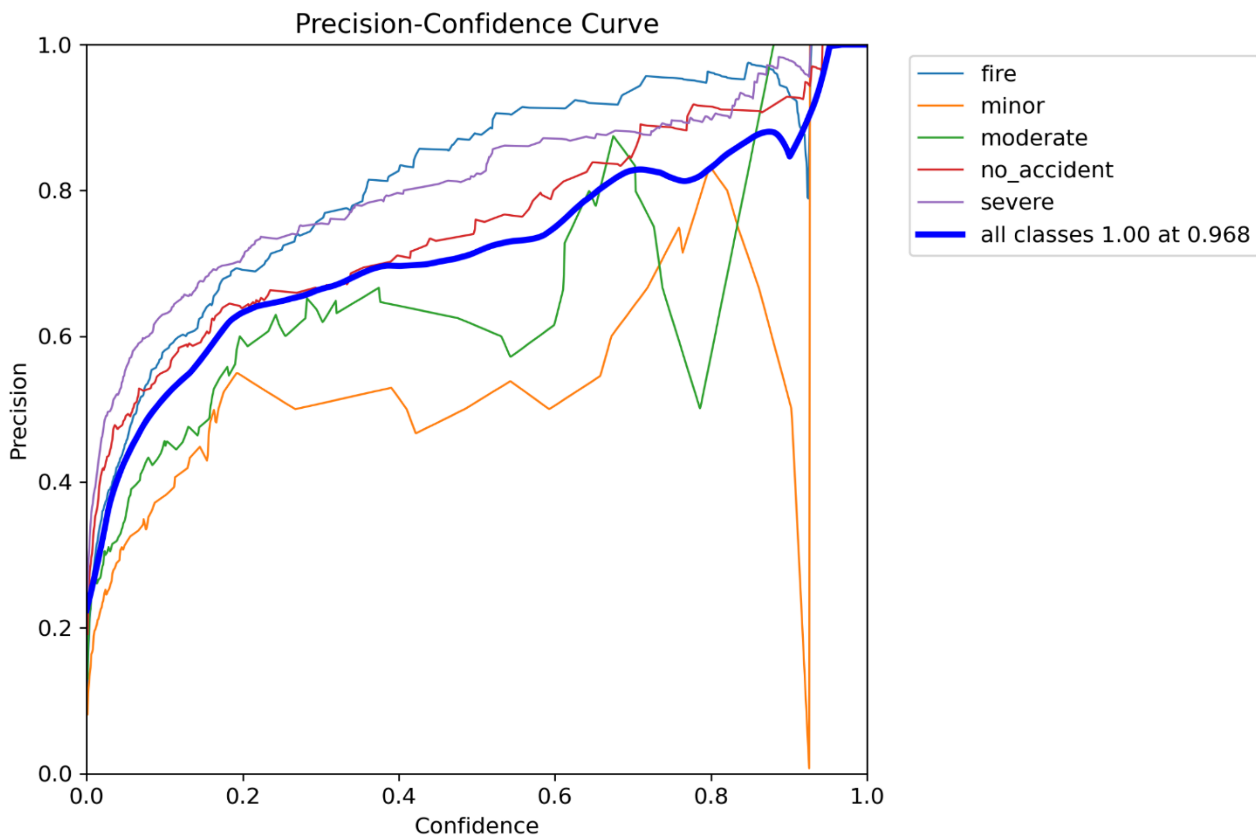


Figure 12.Precision-Confidence Curve

Precision-Confidence Curve :

This curve shows the relationship between the model's precision (the proportion of true positives among all positive predictions) and the confidence threshold.

As the confidence threshold increases, the precision generally increases, but the recall (the proportion of true positives among all actual positives) decreases.

The curves for different classes (fire, minor, moderate, no_accident, severe) are shown, as well as the overall performance for all classes combined.

This metric helps assess the model's ability to make accurate predictions with high confidence

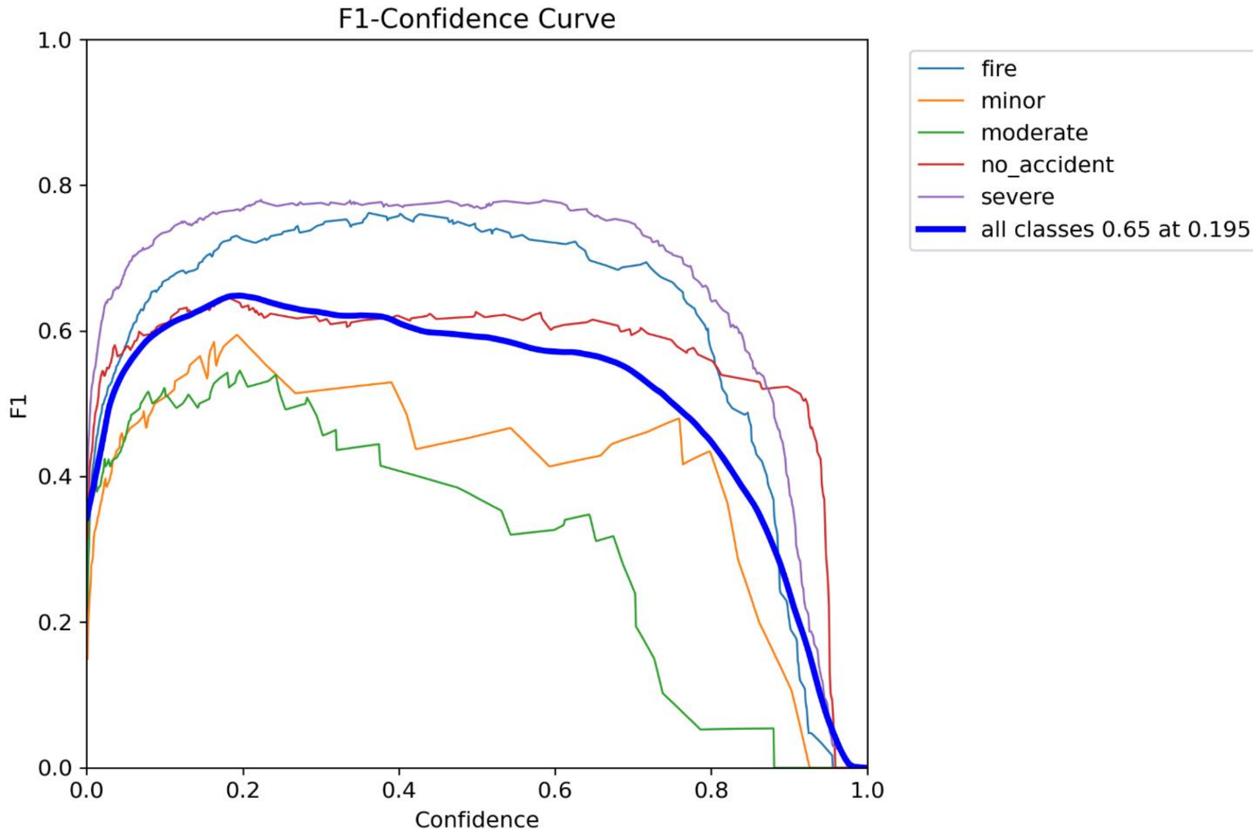


Figure 13. F1-Confidence Curve

F1-Confidence Curve:

- The F1-score is the harmonic mean of precision and recall, which provides a balanced measure of a model's performance.
- This curve shows how the F1-score changes as the confidence threshold is varied.
- Similar to the Precision-Confidence Curve, the curves for different classes and the overall performance are displayed.
- The F1-Confidence Curve helps evaluate the model's ability to achieve a good balance between precision and recall at different confidence levels.

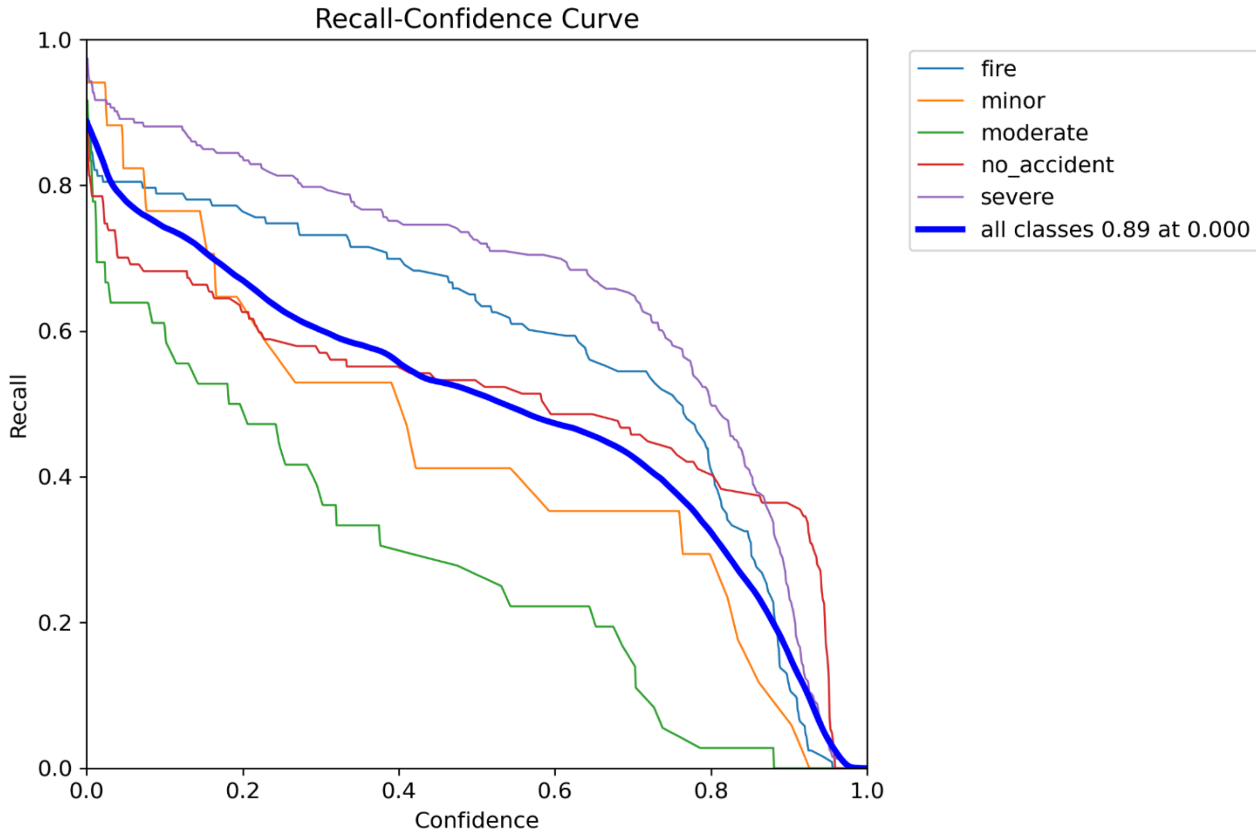


Figure 14. Recall-Confidence Curve

Recall-Confidence Curve :

- This curve shows the relationship between the model's recall and the confidence threshold.
- As the confidence threshold increases, the recall generally decreases, as the model becomes more conservative in its predictions.
- The curves for different classes and the overall performance are shown.
- The Recall-Confidence Curve helps assess the model's ability to detect all relevant instances (i.e., true positives) at different confidence levels.

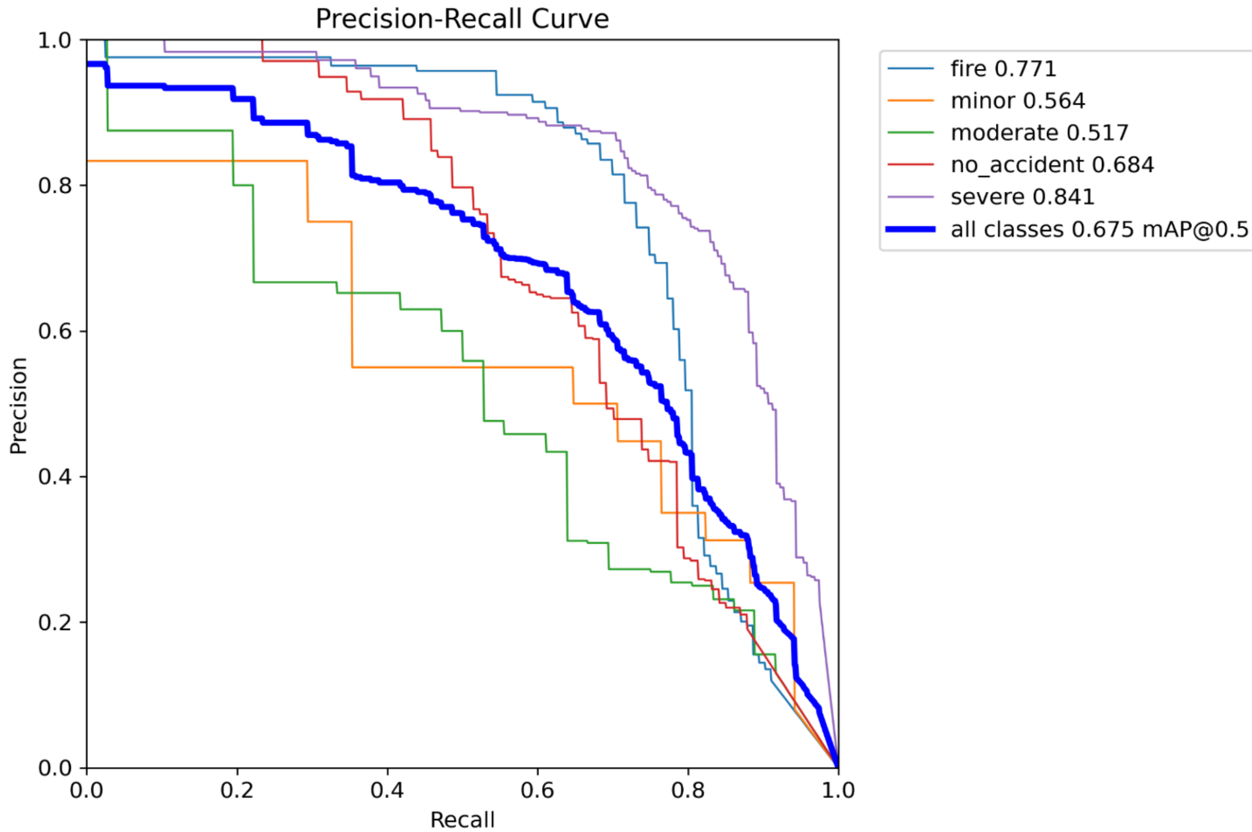


Figure 15. precision-recall curve

Precision-recall curve,

which is a common evaluation metric used in machine learning, particularly for classification tasks. A precision-recall curve plots the trade-off between precision (a measure of how many of the positive predictions are truly positive) and recall (a measure of how many of the actual positives are correctly identified) at different classification thresholds.

- Fire (0.771): This curve represents the performance of the model in detecting fire incidents. The area under the curve (AUC) for this class is 0.771, which indicates a decent performance.

- Minor (0.564): This curve shows the model's performance in identifying minor incidents. The AUC of 0.564 suggests a moderate performance for this class.

- Moderate (0.517): The curve for moderate incidents has an AUC of 0.517, indicating a relatively low performance for this class.

- No accident (0.684): This curve represents the model's ability to detect situations where no accident has occurred. The AUC of 0.684 suggests a reasonably good performance for this class.

- Severe (0.841): The curve for severe incidents has the highest AUC of 0.841, indicating that the model performs well in identifying severe incidents.

- All classes (0.675 mAP@0.5): This curve represents the overall performance of the model across all classes, with a mean average precision (mAP) of 0.675 at a recall threshold of 0.5.

These three evaluation metrics provide a comprehensive understanding of the model's performance across different aspects:

- Precision-Confidence Curve: Focuses on the model's precision and its ability to make accurate predictions with high confidence.
- F1-Confidence Curve: Evaluates the balance between precision and recall, which is crucial for a well-performing classification model.
- Recall-Confidence Curve: Assesses the model's ability to detect all relevant instances, which is important for applications where minimizing false negatives is crucial.

6.3 Experimental Dataset

The experimental data used in the development and testing of the car accident detection and wildlife recognition system is crucial for training, validating, and evaluating the performance of the deep learning models. Here's a detailed description of the experimental data:

- Data Sources:
  - The data used in this project comes from various sources, including live CCTV feeds, traffic cameras, dashcam videos, and publicly available image datasets. These sources provide diverse scenarios and environments for robust model training.

- Image and Video Datasets:
  - The dataset consists of thousands of images and video clips depicting road scenes, car accidents, and wildlife on roads. These datasets are collected from urban, suburban, and rural areas to capture a wide range of conditions and scenarios.

- Annotation and Labeling:
  - Each image and video frame is meticulously annotated and labeled to identify objects of interest, such as vehicles, pedestrians, and wildlife. Annotations include bounding boxes for object localization and labels for object classification (e.g., car, pedestrian, cow, dog).
  - Annotation tools and platforms, such as Roboflow, are used to ensure accurate and consistent labeling of the data.

- Data Augmentation:
  - To enhance the diversity and robustness of the training dataset, various data augmentation techniques are applied. These techniques include random rotations, scaling, cropping, flipping, and brightness adjustments. Data augmentation helps improve the model's ability to generalize to different conditions and reduce overfitting.

- Training, Validation, and Test Sets:
  - The dataset is divided into three subsets: training, validation, and test sets. The training set is used to train the deep learning models, the validation set is used to tune hyperparameters and assess model performance during training, and the test set is used to evaluate the final model's accuracy and effectiveness.
  - A typical split might be 70% of the data for training, 20% for validation, and 10% for testing, although these ratios can vary depending on the specific needs of the project.

- Class Distribution:
  - The dataset includes a balanced distribution of different classes of objects to ensure the model learns to detect and recognize all relevant categories effectively. For example, the dataset includes a variety of car accident scenarios (minor, moderate, critical) and different wildlife species (cows, sheep, dogs, bulls).

- Environmental Variability:
  - The data captures a wide range of environmental conditions, including different lighting (day,

night, dawn, dusk), weather conditions (clear, rainy, foggy), and road types (highways, urban streets, rural roads). This variability ensures that the model can perform well in diverse real-world situations.

The comprehensive and diverse nature of the experimental data used in this project is essential for developing a robust and accurate car accident detection and wildlife recognition system, capable of operating effectively under various real-world conditions.

Figure 16. Experimental Dataset

Chapter 7
Conclusion and Future Enhancements

7.1 Conclusion

In conclusion, the car accident detection and wildlife recognition system represents a significant advancement in road safety technology, leveraging state-of-the-art deep learning algorithms and computer vision techniques to enhance incident detection and response. By utilizing YOLO v8 for real-time object detection and classification, the system can accurately identify car accidents and recognize wildlife on the road, categorizing incidents based on severity levels.

Through robust preprocessing of input data and integration with tools like Roboflow, the system ensures high-quality and diverse datasets, improving the accuracy and reliability of detection. The implementation of email notification functionality enables prompt alerts to emergency services and relevant authorities in the event of critical incidents, facilitating swift intervention and potentially saving lives.

With a user-friendly interface and secure authentication mechanisms, the system provides a seamless and intuitive experience for users, empowering them to proactively mitigate risks and enhance road safety measures.

7.2 Future Enhancements

We would like to enhance the project by adding more features such as increasing scalability and deploying it for commercial environment such as to provide it to the government of India for increasing the safety on road thus protecting wildlife and all resources. Also build models that are lightweight and can run on mobile rather than using web services.

References

[1] Butt, S.M., 2016. A review paper on accident detection system using intelligent algorithm for VANET. J. Info. Eng. Appl, 6(5), pp.24-30.

[2] Bansal, B.N. and Garg, V., 2021. A Review paper on “Vehicle Accident Detection, Tracking and Notification Systems”-A comparative study.

[3] Dogru, N. and Subasi, A., 2012, May. Traffic accident detection by using machine learning methods. In Third International Symposium on Sustainable Development (ISSD’12) (p. 467).

[4] Tian, D., Zhang, C., Duan, X. and Wang, X., 2019. An automatic car accident detection method based on cooperative vehicle infrastructure systems. IEEE Access, 7, pp.127453-127463.

[5] Ravindran, V., Viswanathan, L. and Rangaswamy, S., 2016. A novel approach to automatic road-accident detection using machine vision techniques. International Journal of Advanced Computer Science and Applications, 7(11).

[6] Prajapati, G., Kumar, L. and Patil, S.R.S., 2023. Road Accident Prediction Using Machine Learning. Journal of Scientific Research and Technology, pp.48-59.

[7] Desai, R., Jadhav, A., Sawant, S. and Thakur, N., 2021. Accident Detection Using ML and AI Techniques. Engpaper Journal.

[8] Banupriya, N., Saranya, S., Swaminathan, R., Harikumar, S. and Palanisamy, S., 2020. Animal detection using deep learning algorithm. J. Crit. Rev, 7(1), pp.434-439.

[9] Verma, G.K. and Gupta, P., 2018. Wild animal detection using deep convolutional neural network. In Proceedings of 2nd International Conference on Computer Vision & Image Processing: CVIP 2017, Volume 2 (pp. 327-338). Springer Singapore.

[10] Saxena, A., Gupta, D.K. and Singh, S., 2021. An animal detection and collision avoidance system

using deep learning. In Advances in Communication and Computational Technology: Select Proceedings of ICACCT 2019 (pp. 1069-1084). Springer Singapore.
[11] Ferrante, G., Nakamura, L.V., Sampaio, S. and Meneguette, R., 2023. Evaluating YOLO-based Object Detectors for Detecting Road-Killed Endangered Brazilian Animals.

[12] Ibraheam, M., Li, K.F., Gebali, F. and Sielecki, L.E., 2021. A performance comparison and enhancement of animal species detection in images with various R-CNN models. AI, 2(4), pp.552-577.

[13] Bala, J.A., Adeshina, S.A. and Aibinu, A.M., 2023. Performance Evaluation of You Only Look Once v4 in Road Anomaly Detection and Visual Simultaneous Localisation and Mapping for Autonomous Vehicles. World Electric Vehicle Journal, 14(9), p.265.

[14] Reddy, B.H. and Gayathri, N., 2021, July. Pedestrian Detection System using Image Processing for Driver Assistance. In Journal of Physics: Conference Series (Vol. 1964, No. 6, p. 062028). IOP Publishing.

[15] D. K. Yadav, Renu, Ankita and I. Anjum, "Accident Detection Using Deep Learning," 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 2020, pp. 232-235, doi: 10.1109/ICACCCN51052.2020.9362808.

[16] Tamagusko, Tiago & Gomes Correia, Matheus & Huynh, Minh Anh & Ferreira, Adelino. (2022). Deep Learning Applied to Road Accident Detection with Transfer Learning and Synthetic Images. Transportation Research Procedia. 64. 90-97. 10.1016/j.trpro.2022.09.012.

[17] In Jung Lee, "An accident detection system on highway through CCTV with calogero-moser system," 2012 18th Asia-Pacific Conference on Communications (APCC), Jeju, Korea (South), 2012, pp. 522-525, doi: 10.1109/APCC.2012.6388248.

[18] N. Dogru and A. Subasi, "Traffic accident detection using random forest classifier," 2018 15th Learning and Technology Conference (L&T), Jeddah, Saudi Arabia, 2018, pp. 40-45, doi: 10.1109/LT.2018.8368509.

[19] A. K. Agrawal et al., "Automatic Traffic Accident Detection System Using ResNet and SVM," 2020 Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Bangalore, India, 2020, pp. 71-76, doi: 10.1109/ICRCICN50933.2020.9296156

[20] N. Agarwal et al., "Camera-based Smart Traffic State Detection in India using Deep Learning Models," 2021 International Conference on COMmunication Systems & NETworkS (COMSNETS), Bangalore, India, 2021, pp. 690-696, doi: 10.1109/COMSNETS51098.2021.9352904.

[21] M. Ozbayoglu, G. Kucukayan and E. Dogdu, "A real-time autonomous highway accident detection model based on big data processing and computational intelligence," 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 2016, pp. 1807-1813, doi: 10.1109/BigData.2016.7840798.

[22] Gat, A., Gaikwad, H., Giri, R. and Chaudhari, A., Animal Classifier System for Video Surveillance and Forest Monitoring Using Raspberry-pi.

[23] Antônio, W.H., Da Silva, M., Miani, R.S. and Souza, J.R., 2019. A proposal of an animal detection system using machine learning. Applied Artificial Intelligence, 33(13), pp.1093-1106.

[24] Tamagusko, T., Correia, M.G., Huynh, M.A. and Ferreira, A., 2022. Deep learning applied to road accident detection with transfer learning and synthetic images. Transportation research procedia, 64, pp.90-97.

[25] Adewopo, V., Elsayed, N., Elsayed, Z., Ozer, M., Wangia-Anderson, V. and Abdelgawad, A., 2023. Ai on the road: A comprehensive analysis of traffic accidents and accident detection system in smart cities. arXiv preprint arXiv:2307.12128.

[26] Hu, J., Zhang, X. and Maybank, S., 2020. Abnormal driving detection with normalized driving behavior data: A deep learning approach. IEEE transactions on vehicular technology, 69(7), pp.6943-6951.

[27] Miao, Z., Gaynor, K.M., Wang, J., Liu, Z., Muellerklein, O., Norouzzadeh, M.S., McInturff, A.,

Bowie, R.C., Nathan, R., Yu, S.X. and Getz, W.M., 2019. Insights and approaches using deep learning to classify wildlife. Scientific reports, 9(1), p.8137.

[28] Essam, M., Ghanem, N.M. and Ismail, M.A., 2022. Detection of road traffic crashes based on collision estimation. arXiv preprint arXiv:2207.12886.

[29] Abhishek, L., Aravindhan, S. and Kumar, T.R., 2020, January. Accident Detection and Number Plate Recognition using Image Processing and Machine Learning. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, No. 5).

[30] Ramírez, G., Salazar, K., Barria, V., Pinto, O., San Martin, L., Carrasco, R., Fuentealba, D. and Gatica, G., 2022. Accident risk detection in urban trees using machine learning and fuzzy logic. Procedia Computer Science, 203, pp.471-475.

[31] Adewopo, V., Elsayed, N., ElSayed, Z., Ozer, M., Wangia-Anderson, V. and Abdelgawad, A., 2023, November. AI on the Road: A Comprehensive Analysis of Traffic Accidents and Autonomous Accident Detection System in Smart Cities. In 2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI) (pp. 501-506). IEEE.

[32] Wahid, F., Ali, S., Abbasi, I.A., Baseer, S. and Khan, H.U., 2022. Accident Detection in Autonomous Vehicles Using Modified Restricted Boltzmann Machine. Security and Communication Networks, 2022.

[33] Uma, S. and Eswari, R., 2022. Accident prevention and safety assistance using IOT and machine learning. Journal of Reliable Intelligent Environments, 8(2), pp.79-103.

[34] Narayana, L.L., Car Crash Detection System using Machine Learning and Deep Learning Algorithms.

[35] Hozhabr Pour, H., Li, F., Wegmeth, L., Trense, C., Doniec, R., Grzegorzek, M. and Wismüller, R., 2022. A machine learning framework for automated accident detection based on multimodal sensors in cars. Sensors, 22(10), p.3634.

[36] Druta, C. and Alden, A.S., 2020. Preventing animal-vehicle crashes using a smart detection technology and warning system. Transportation research record, 2674(10), pp.680-689.

[37] Kwon, D., Malaiya, R., Yoon, G., Ryu, J.T. and Pi, S.Y., 2019. A study on development of the camera-based blind spot detection system using the deep learning methodology. Applied Sciences, 9(14), p.2941.

[38] Masoumian, A., Marei, D.G., Abdulwahab, S., Cristiano, J., Puig, D. and Rashwan, H.A., 2021, October. Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models. In CCIA (pp. 325-334).

[39] Vajgl, M., Hurtik, P. and Nejezchleba, T., 2022. Dist-YOLO: fast object detection with distance estimation. Applied sciences, 12(3), p.1354.

[40] Zhao, Y., Bai, L., Lyu, Y. and Huang, X., 2019. Camera-based blind spot detection with a general purpose lightweight neural network. Electronics, 8(2), p.233.

[41] Balfaqih, M., Alharbi, S.A., Alzain, M., Alqurashi, F. and Almilad, S., 2021. An accident detection and classification system using internet of things and machine learning towards smart city. Sustainability, 14(1), p.210.

APPENDIX A: Snapshots

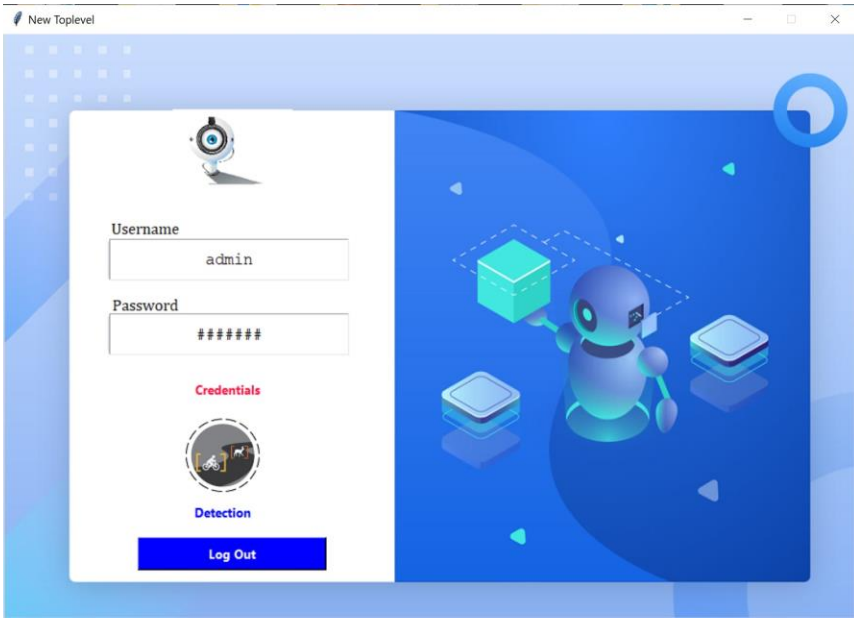


Figure 1. Authentication Page

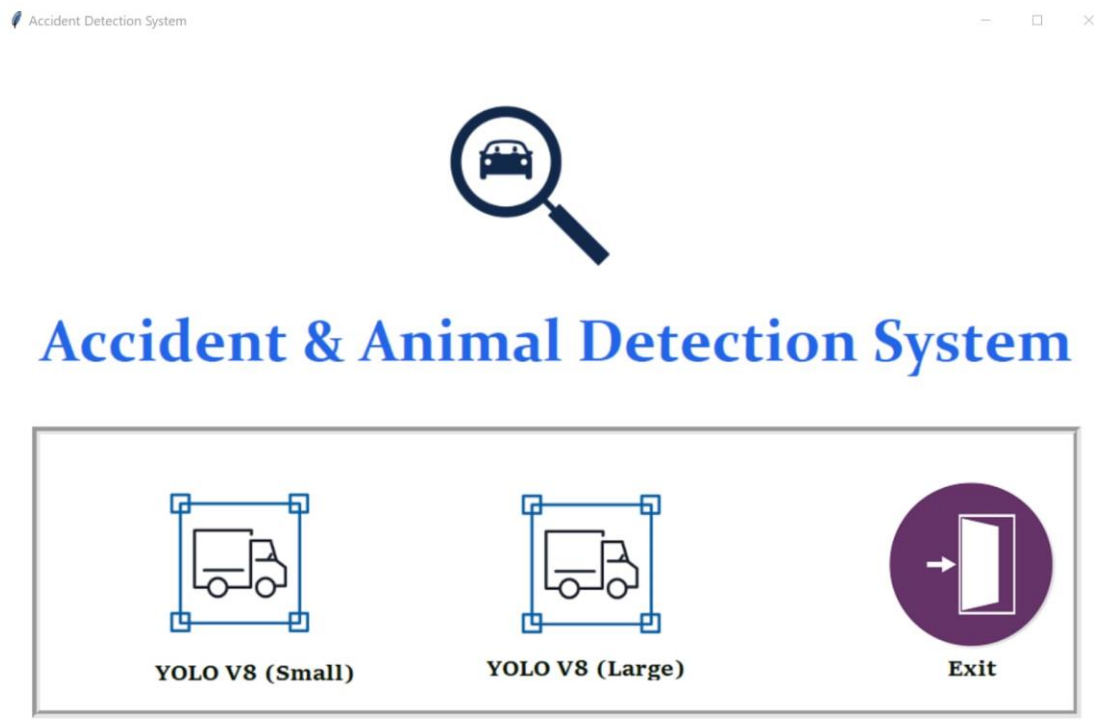


Figure 2 (a). Overview of screens present

Fig 2 (b). Overview of screens present in the app

Fig 2 (c). Overview of screens present in the app

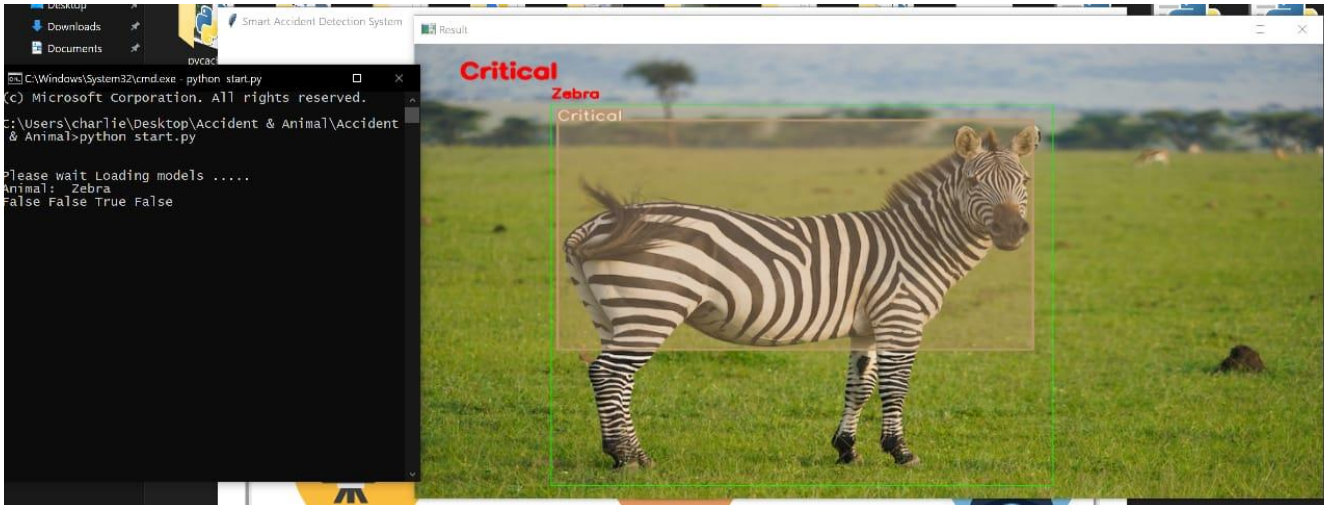


Figure 3. Sample Image Output with identified classes of the animal detected and its alert system

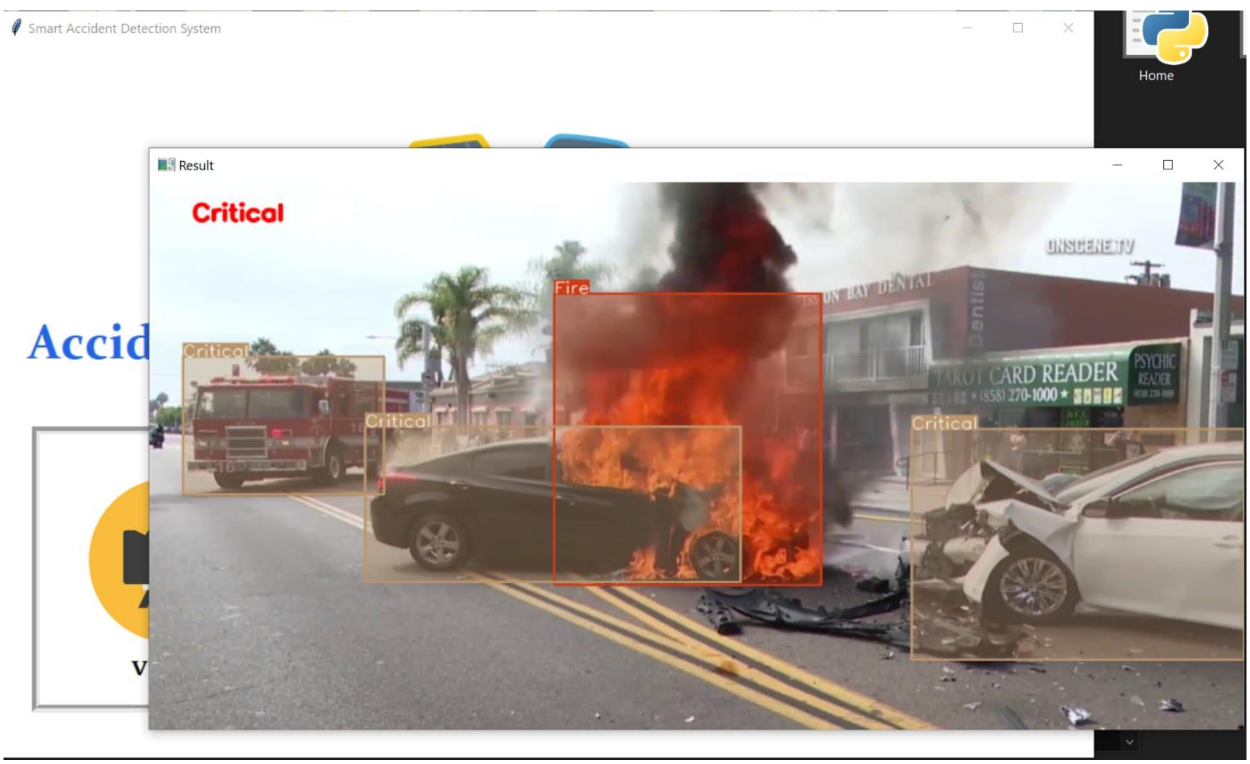


Figure 4. Sample Image Output with identified classes of the type of accident and Fire

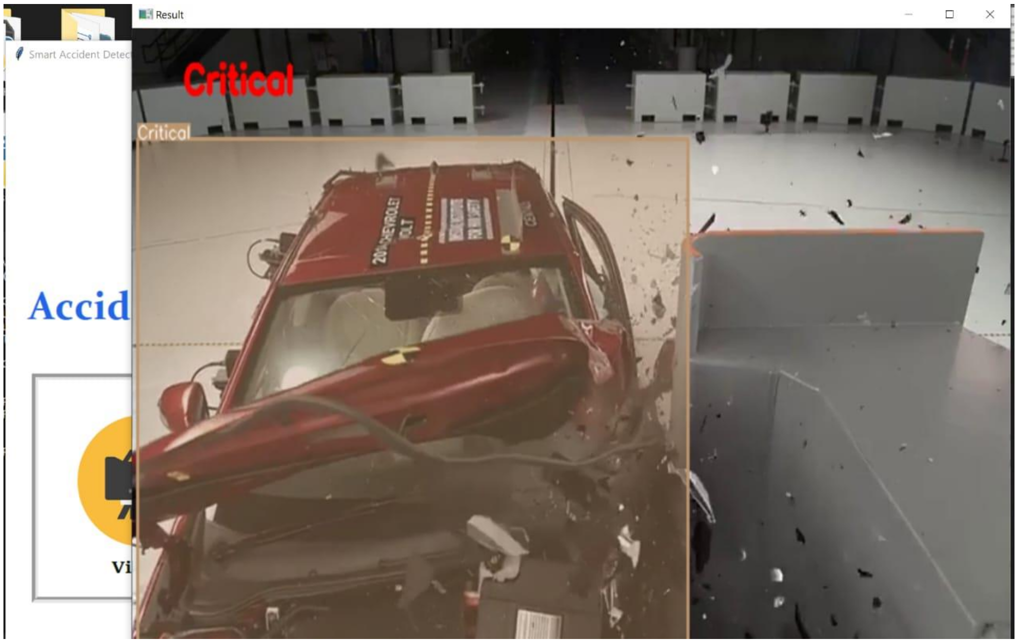


Figure 5. Sample Output of a video with identified classes of the type of accident

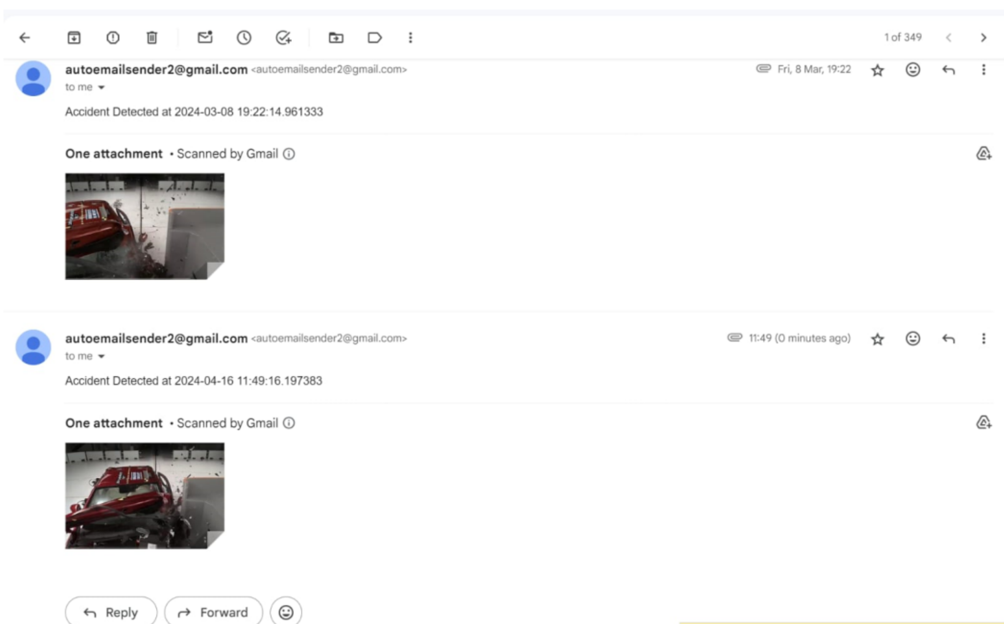


Figure 6. Alert System send email to the respective emergency number

APPENDIX B: Details of Publications

1. **Author Names:** HIMANSHU RAJ , KUSHAL C, MEHUL TEJ, KEERTHAN S GOWDA, Sheetal V A , Dr. Umadevi V
2. **Paper Title:** Car Accident Detection Using Yolo V8 Algorithms
3. **Name of the Journal: :** International Journal of Innovative Research in Technology
4. **Date of journal:**

   Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

   Published in Volume 10 Issue 12, May 2024

   Registration ID 164610 Research paper weblink:https://ijirt.org/Article?manuscript=164610

APPENDIX C: Details of patents NIL

APPENDIX D: Details of funding NIL

APPENDIX E: Programme Outcomes Mapped

PROGRAMME OUTCOMES	Level (1/2/3)	Justification if addressed
PO1	3	Collaborative Filtering and CNN models required extensive use of data structures and algorithms.
PO2	3	Image Classification and Recommendation
PO3	2	Client-Server Architecture.
PO4	3	Methods based on research were studied.
PO5	3	Used Google Colab to design deep learning models.
PO6	1	The project can further be extended to improve the cooking experience
PO7	1	The project does not have negative effects on environment
PO8	1	Measures are taken so that the project is ethical to the best of our knowledge
PO9	3	Every team member contributed to the project and discharged his duties well.
PO10	3	Published a paper.
PO11	2	Effectively communicated and presented results.
PO12	3	This project enabled us to expand our knowledge of current trends in deep learning.
PSO1	3	Knowledge of full stack development applied to create a mobile application.
PSO2	2	Developed Mobile application to help users interact.
PSO3	3	Efficient code was produced to solve the problem

APPENDIX F: Plagiarism report

DrillBit

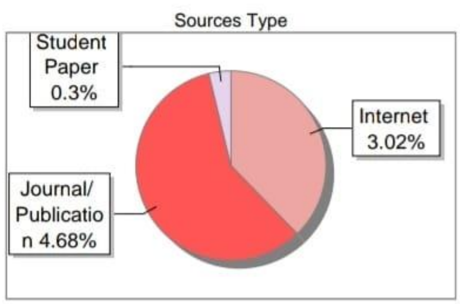
The Report is Generated by DrillBit Plagiarism Detection Software

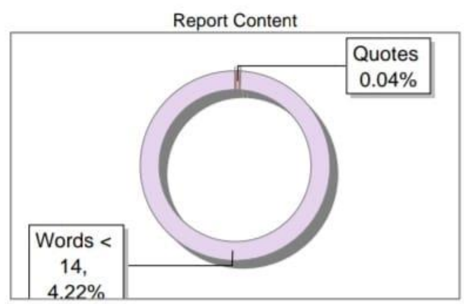
Submission Information

Author Name	Kushal
Title	Car_Accident
Paper/Submission ID	1838367
Submitted by	hod.cse@bmsce.ac.in
Submission Date	2024-05-20 12:57:05
Total Pages, Total Words	54, 9681
Document type	Project Work

Result Information

Similarity **8 %**





Exclude Information

Quotes	Not Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	No

A Unique QR Code use to View/Download/Share Pdf File

DrillBit Similarity Report

8	50	A	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)
SIMILARITY %	MATCHED SOURCES	GRADE	

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	fastercapital.com	<1	Internet Data
2	www.irjmets.com	<1	Publication
3	arxiv.org	<1	Publication
4	library.sadjad.ac.ir	<1	Publication
5	arxiv.org	<1	Publication
6	www.deskera.com	<1	Internet Data
7	Enhancing Deep Learning with Visual Interactions, by Krokos, Eric Cheng- 2019	<1	Publication
8	www.dx.doi.org	<1	Publication
9	ijrte.org	<1	Publication
10	www.readbag.com	<1	Internet Data
11	link.springer.com	<1	Publication
12	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
13	www.ncbi.nlm.nih.gov	<1	Internet Data
14	neptjournal.com	<1	Publication

15	www.ncbi.nlm.nih.gov	<1	Internet Data
16	dovepress.com	<1	Internet Data
17	Experimental and theoretical studies of photoinduced reactions in the by Murga-2020	<1	Publication
18	bmcpsychiatry.biomedcentral.com	<1	Internet Data
19	moam.info	<1	Internet Data
20	www.jetir.org	<1	Publication
21	Design framework of adaptive intelligent tutoring systems by Ermit-2020	<1	Publication
22	arxiv.org	<1	Publication
23	A hierarchical nest survival model integrating incomplete temporally varying cov by Converse-2013	<1	Publication
24	www.mdpi.com	<1	Internet Data
25	www.mdpi.com	<1	Internet Data
26	biomedcentral.com	<1	Internet Data
27	Relation between vehicle travel velocity and pedestrian injury risk in different by Oikawa-2016	<1	Publication
28	technodocbox.com	<1	Internet Data
29	thesai.org	<1	Publication
30	Thesis Submitted to Shodhganga, shodhganga.inflibnet.ac.in	<1	Publication
31	Thesis Submitted to Shodhganga Repository	<1	Publication
32	IEEE 2011 IEEE Globecom Workshops - Houston, TX, USA (2011125-2011 by	<1	Publication

33	1library.co	<1	Internet Data
34	agriculturejournals.cz	<1	Internet Data
35	citeseerx.ist.psu.edu	<1	Internet Data
36	Co-trained convolutional neural networks for automated detection of prostate can by Yang-2017	<1	Publication
37	documents.mx	<1	Internet Data
38	dspace.lib.cranfield.ac.uk	<1	Internet Data
39	Editorial, by Hjlmdahl, Magnus; - 2017	<1	Publication
40	educationportal.mp.gov.in	<1	Internet Data
41	IEEE 2012 10th IAPR International Workshop on Document Analysis Syste by	<1	Publication
42	moam.info	<1	Internet Data
43	mu.ac.in	<1	Publication
44	SAR Welcomed 10 Scholars to Campus in Fall 2007 by Nanc-2007	<1	Publication
45	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
46	Translating in Col 1,2 and Eph 1,1 by Thoma-2006	<1	Publication
47	wikileaks.wikimee.org	<1	Internet Data
48	www.doaj.org	<1	Publication
49	www.fujitsu.com	<1	Publication
50	www.intechopen.com	<1	Publication