SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

Project Report on

"Image Processing based Traffic Surveillance"

submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS & COMMUNICATION ENGINEERING

Submitted by

D K Chandrakanth	(1SI20EC020)
Siddharth Prabhu	(1SI20EC090)
Preetham G M	(1SI20EC121)
Mithun J B	(1SI20EC124)

under the guidance of

A N Mukunda Rao

Associate Professor

Department of E&CE

SIT, Tumakuru-03

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

2023-24

SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU-572103

(An Autonomous Institute under Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

Certified that the project work entitled "IMAGE PROCESSING BASED TRAFFIC SURVEILLANCE" is a bonafide work carried out by D K Chandrakanth (1SI20EC020), Siddharth Prabhu(1SI20EC090), Preetham G M(1SI20EC121) & Mithun J B(1SI20EC124) in partial fulfillment for the award of degree of Bachelor of Engineering in Electronics & Communication Engineering from Siddaganga Institute of Technology, an autonomous institute under Visvesvaraya Technological University, Belagavi during the academic year 2023-24. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The Project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering degree.

A N Mukunda Rao	Head of the Department	Principal
Associate Professor	Dept. of E&CE	SIT,Tumakuru-03
Dept. of E&CE	SIT,Tumakuru-03	
SIT,Tumakuru-03		

External viva:

Names of the Examiners **Signature with date**

1.

2.

ACKNOWLEDGEMENT

We offer our humble pranams at the lotus feet of **His** Holiness, **Dr. Sree Sree Sivakumara Swamigalu**, Founder President and **His** Holiness, **Sree Sree Siddalinga Swamigalu**, President, Sree Siddaganga Education Society, Sree Siddaganga Math for bestowing upon their blessings.

We deem it as a privilege to thank **Dr. M N Channabasappa**, Director, SIT, Tumakuru, **Dr. Shivakumaraiah**, CEO, SIT, Tumakuru, and **Dr. S V Dinesh**, Principal, SIT, Tumakuru for fostering an excellent academic environment in this institution, which made this endeavor fruitful.

We would like to express our sincere gratitude to **Dr. K V Suresh**, Professor and Head, Department of E&CE, SIT, Tumakuru for his encouragement and valuable suggestions.

We thank our guide **A N Mukunda Rao**, Associate Professor, Department of Electronics & Communication Engineering, SIT, Tumakuru for the valuable guidance, advice and encouragement.

D K Chandrakanth	(1SI20EC020)
Siddharth Prabhu	(1SI20EC090)
Preetham G M	(1SI20EC121)
Mithun J B	(1SI20EC124)

Course Outcomes

After successful completion of major project, graduates will be able to

CO1: To identify a problem through literature survey and knowledge of contemporary engineering technology.

CO2: To consolidate the literature search to identify issues/gaps and formulate the engineering problem.

CO3: To prepare project schedule for the identified design methodology and engage in budget analysis, and share responsibility for every member in the team.

CO4: To provide sustainable engineering solution considering health, safety, legal, cultural issues and also demonstrate concern for environment.

CO5: To identify and apply the mathematical concepts, science concepts, engineering and management concepts necessary to implement the identified engineering problem.

CO6: To select the engineering tools/components required to implement the proposed solution for the identified engineering problem.

CO7: To analyze, design, and implement optimal design solution, interpret results of experiments and draw valid conclusion.

CO8: To demonstrate effective written communication through the project report, the one-page poster presentation, and preparation of the video about the project and the four page IEEE/Springer/ paper format of the work.

CO9: To engage in effective oral communication through power point presentation and demonstration of the project work.

CO10: To demonstrate compliance to the prescribed standards/ safety norms and abide by the norms of professional ethics.

CO11: To perform in the team, contribute to the team and mentor/lead the team.

PSOs:

PSO1: The ability to analyze and design systems in the areas related to microelectronics, communication, signal processing and embedded systems for solving real world problems **(Professional Skills)**.

PSO2: The ability to identify problems in the areas of communication and embedded systems and provide efficient solutions using modern tools/algorithms individually or working in a team **(Problem-Solving Skills)**.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO-1												3		3
CO-2	3												3	
CO-3											3			3
CO-4						3	3							3
CO-5	3	3											3	
CO-6					3									3
CO-7			3	3									3	
CO-8										3				3
CO-9										3				3
CO-10								3						3
CO-11									3					3
Average	3	3	3	3	3	3	3	3	3	3	3	3	3	3

Attainment level: - 1: Slight (low) 2: Moderate (medium) 3: Substantial (high)

POs: PO1: Engineering Knowledge, PO2: Problem analysis, PO3: Design/Development of solutions, PO4: Conduct investigations of complex problems, PO5: Modern tool usage, PO6: Engineer and society, PO7: Environment and sustainability, PO8: Ethics, PO9: Individual and team work, PO10: Communication, PO11: Project management and finance, PO12: Lifelong learning.

Abstract

Motorcycles and cars being primary modes of road transportation globally, face significant risks when traffic violations such as failure to wear helmets, engaging in triple riding, using mobile phones while driving and surpassing zebra cross lane happens, increasing the likelihood of accidents and injuries, posing threats not only to those directly involved but to the entire community sharing the road. Traditional traffic law enforcement encounters challenges like resource constraints, inconsistent enforcement, limited coverage, and safety concerns for law enforcement officers. To tackle these issues, efforts are underway to revolutionize traffic law enforcement through an automated surveillance system driven by image processing technology. This proposed system extracts the license plate information of the vehicle when traffic rules violation such as failure to wear helmet, triple riding, mobile phone usage in case of bikes and surpassing the zebra crossing lane for both bikes and cars. This system utilizes traffic videos as inputs and employs YOLOv8, an object detection model, to accurately identify bike, car and subsequently is trained to detect violations. Upon violation detection, an OpenALPR model extracts license plate of the corresponding vehicle and stores this secure data in the Firebase Cloud. This system is tested on Tumkuru city road traffic and in SIT campus for 50 videos of total duration around 100 hours. The system achieves an accuracy of 99% in detecting bike riders, helmeted riders with 99% accuracy, no helmeted riders with 98% accuracy and detecting the instance of mobile phone usage with an accuracy of 98%, triple riding instance with 96%, and detects the instances of zebra cross lane by an accuracy of 90%. The system achieved an overall accuracy in detecting the traffic offenders with 97%. Also the system detects the license plate information of the violated vehicle with an accuracy of 93%. Such technology aids law enforcement in consistently enforcing traffic rules, thereby promoting enhanced road safety and orderliness, with the ultimate goal of creating safer roads for everyone.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

India witnessed a substantial 12% surge in road accidents while concurrently observing an 18% increase in vehicle ownership during the year 2023 [Courtesy: indiatoday.in]. As the global population grows, the number of bikes and cars is also increasing. To reduce injury risks, it's mandatory for riders to adhere to traffic rules. Motorcyclists and car drivers are among the most vulnerable to road accidents, posing threats not only to those directly involved but to the entire community sharing the road worldwide, with reckless driving being a major cause. Head injuries are a leading cause of fatalities in these accidents. Studies show that more than half of motorcyclists and car occupants who died in road accidents could have survived if they were wearing helmets, refrained from triple riding, and avoided using mobile phones while riding. Section 129 of the Motor Vehicles Act, 1988, mandates that motorcyclists wear helmets, refrain from triple riding, avoid using mobile phones while riding, and that cars and bikes do not violate the zebra crossing rules. Helmets should have a thickness ranging from 20 to 25 mm along with an ISI mark [Courtesy: indiacode.nic.in]. However, people often do not follow these regulations.

There are existing techniques that use sensors installed on motorcycles to detect the presence of helmets, but they are very expensive to install on every vehicle and are also very inefficient. Along with this, various approaches have been proposed that use video processing techniques to detect traffic rule violations by motorcyclists and car drivers. Determination of helmeted and non-helmeted riders is done by identifying visual features using a binary classifier [1]. A deep neural network was proposed to identify multiple riders on motorcycles using the YOLOv3 model and also to detect helmets [3]. An automatic approach to detect triple riding and speed rule violation using faster-CNN trained on the COCO dataset was proposed in [6]. A technique to detect helmets on motorcycle riders based on the K-Nearest Neighbour (KNN) approach was presented in [7]. The YOLOv5 model is used to extract the license plate location and cropping of the license plate image and the recognition network uses a Gated Recurrent Units(GRU) [8]. The drawbacks of the technologies include less accuracy, lower speed computation, and increased complexity.

Therefore, YOLO models are chosen for the traffic violation detection, as they offer good speed and accuracy. Specifically, YOLOv8 model is selected as it can be trained on a larger dataset and has a Multi-Scale Feature Fusion to detect small objects. Additionally, this model has an anchor-free bounding box detection capacity, which helps in non-maximum suppression (NMS) for faster processing, improving both speed and accuracy.

By leveraging these techniques, the system aims to significantly enhance road safety, diminish traffic rule violations and foster the creation of a more secure and orderly traffic environment.

1.1 Motivation

The motivation for this project comes from the challenges faced by traffic police officers in the past. Their methods required a significant number of officers, vehicles and financial resources, straining their capacity. Moreover, their enforcement was inconsistent because each officer had their own interpretation of the rules. Coverage was limited, with many violations slipping through the cracks. Additionally, there were safety concerns for officers when stopping vehicles, especially in busy or fast traffic roads. To address these problems, this project, based on image processing for traffic surveillance, is designed to provide solutions that automatically manage various traffic violations eliminating the need for manual monitoring and to enhance road safety, streamline enforcement, prioritize public welfare, promote pedestrian safety. By doing so, this system aims to create a more efficient and safer traffic environment while addressing the limitations of traditional law enforcement methods.

1.2 Objective of the project

The objective of this project is to extract the License plate information of the bike and car which violate the traffic rules such as,

- Riding bike without a helmet.
- Triple riding on a bike.
- Mobile phone usage while riding the bike.
- Bike and car surpassing over the zebra crossing lane in a traffic signal.

1.3 Organisation of the report

The project report is divided into 6 chapters.

Chapter 1 mainly discusses about introduction and the motivation of the project which leads to the development of this project and also it lists-out the main objectives of the project.

Chapter 2 contains the literature survey.

Chapter 3 proposes the system overview implemented in the project.

Chapter 4 encompasses hardware.

Chapter 5 gives the flowchart, algorithm and software implemented.

Chapter 6 gives the training. testing, results and further discussions of the project.

Chapter 7 gives the conclusion and scope for future work.

Chapter 2

Literature Survey

This chapter includes the various literature survey carried out related to Traffic Surveillance and a brief summary of the literature review.

• A technique for automatically detecting motorcycle riders not wearing helmets in security and inspection videos was proposed. This involved the detection of motorcyclists from surveillance video using object segmentation and background subtraction techniques. Then, determination of helmeted and non-helmeted riders was achieved by identifying visual features using a binary classifier. To compare the approach with previously presented methods, the technique was evaluated against three commonly used feature representations: Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and Local Binary Patterns (LBP). The introduced approach proved to be more affordable with fewer calculations and performed well in all cases [1].

• Machine learning based system has been developed to automatically detect bike riders without helmets and those engaged in triple riding from traffic surveillance videos. The system utilizes YOLOv3, a real-time object detection model, to identify bikes and riders. Following bike detection, it employs bounding boxes to distinguish riders with or without helmets, and also counts the number of riders on each bike to detect triple rides. The methodology encompasses video collection, image classification, vehicle and rider detection, testing the detection model, and analyzing results [2].

• A deep neural network is introduced to identify multiple riders on motorcycles using the YOLOv3 model, as well as to detect helmets. Motorcycle riders are detected using bounding boxes to localize the object for center coordinates and for width and height prediction. The overlapping area is calculated to determine whether the person is on the bike or not. The Euclidean distance between the center coordinates of two bounding boxes of a person and a motorcycle within the bounding box determines whether they are the rider or not. A softmax classifier is employed to classify the object and predict whether a helmet is present or not [3].

• The block-based Hough transform method is utilized for recognizing zebra crossing lanes in natural scene images. This method entails segmenting the region of interest into overlapping blocks, conducting preprocessing for edge detection, and employing the Hough transform for the detection of parallel and straight lines. The proposed technique furnishes both the position and direction of zebra crossings in nearby regions. The performance of this method was evaluated using numerous images, demonstrating high precision and recall [4].

• An automated system for the detection of triple riding and speed rule violations was developed, employing the YOLOv3 model for identifying violators. The YOLOv3 model was trained using the COCO dataset to enhance its accuracy in detecting rule violations. Traffic image data, collected at junctions, underwent rigorous testing to evaluate the system's performance under various scenarios of traffic violations by the rider. Upon detection of a traffic rule violation, the system automatically captured and saved snapshots of the motorcycles involved in the violation [5].

• A proposed a method for automatically detecting motorcyclists wearing helmets using a Convolutional Neural Network (CNN). This involved a two-step process with YOLOv2 to enhance helmet detection accuracy. Initially, YOLOv2 identified various objects in test images from the COCO dataset. Then, it focused on individuals to refine helmet detection precision. Cropped images of detected individuals were fed into a second YOLOv2 stage trained on helmeted images. Additionally, non-helmeted images were processed for license plate extraction using OpenALPR. This approach integrated COCO and helmet datasets to improve detection [6].

• A comprehensive approach employing Convolutional Neural Networks (CNN) has been proposed for helmet use detection on tracked motorcycles. This innovative framework integrates multi-task learning to address visual similarity learning and patch-based helmet classification. Initially, a pre-trained RetinaNet model identifies active motorcycles, defined as those with at least one rider, at a single-frame level. Subsequently, motorcycles are tracked across consecutive frames using both their motion state and visual similarities, leveraging techniques such as optical flow and feature matching. Finally, upon tracking termination, the system predicts helmet use class, including rider number, positions, and helmet identification, employing sophisticated classification algorithms like softmax regression. This approach seamlessly combines advanced techniques in object detection, motion tracking, and classification for real-time helmet use detection on tracked motorcycles [7].

•A novel deep learning framework is introduced for the precise identification and extraction of license plates within natural environments. Leveraging the YOLOv5 model, the system adeptly detects license plate locations and performs image cropping, optimizing computational resources by minimizing parameters and classes. Subsequently, a recognition network is deployed, integrating Gated Recurrent Units (GRU) for effective sequence labeling and decoding. The training regimen employs the Connectionist Temporal Classification (CTC) loss function, ensuring robust model optimization. This holistic approach not only enhances efficiency but also fosters accuracy in license plate localization and recognition tasks, thereby advancing the capabilities of automated surveillance and vehicular identification systems [8].

• An automatic license plate recognition model is proposed utilizing the AOLP (Application Oriented License Plate) dataset for license plate recognition. The study primarily focuses on license plate detection, character extraction, and character recognition. The YOLOv3 model is trained to extract license plates. Following extraction, a character segmentation algorithm is trained using license plates with margins and coordinates for character extraction. The YOLO model, known for its speed and accuracy, can process up to 45 frames per second. Padding is applied to enhance character prediction. For text recognition, CRNN is employed [9].

• Addressing the prevalent safety concerns associated with motorcycle riders by automating the detection of rule violations using object detection model is proposed. Using YOLOv3, the system accurately identifies motorcycles in traffic scenes and further employs a second model for helmet detection. To tackle the issue of triple riding, the system uses YOLOv3 for person detection, counting the riders to identify rule violations. Upon detection of triple riding, an alarm is triggered, and images are captured for documentation. This work demonstrated with high accuracy, with 99% for motorcycle detection, 92% for helmet detection, and 97% for no-helmet detection [10].

2.1 Summary

In a series of studies addressing the critical issue of enforcing traffic safety regulations, various researchers have proposed innovative approaches utilizing image processing techniques. After reviewing the survey, it was noticed that some techniques, such as CNNs used for classification. Object segmentation and background subtraction methods are used to identify motorcyclists from surveillance videos. Binary classifiers differentiate between helmeted and non-helmeted riders by analyzing visual features. Bounding boxes help localize the motorcycle and individuals, calculating the overlapping area between them to differentiate if a person is on the bike.

A deep learning model for license plate localization and recognition in the real world was also identified. It employs a recognition network using Gated Recurrent Units (GRU) for sequence labeling and decoding, along with the corresponding ground truth text. This network is fed into a Connectionist Temporal Classification (CTC) loss function, achieving a less recognition accuracy in meeting real-time processing requirements for license plate localization and text recognition within natural scenarios.

However, these technologies have drawbacks including less accuracy, lower speed computation, and increased complexity. Additionally, each paper individually develops models for specific tasks. To overcome these limitations, the proposed project integrates these models to create a unified system capable of extracting license plate information from the vehicles of traffic rule violators.

Therefore, YOLO models are chosen for traffic violation detection, as they offer good speed and accuracy. Specifically, the YOLOv8 model is selected for its ability to be trained on a larger dataset and its Multi-Scale Feature Fusion, which aids in detecting small objects. Additionally, this model's anchor-free bounding box detection capacity helps in non-maximum suppression (NMS) for faster processing, improving both speed and accuracy.

Chapter 3

System Overview

This chapter presents an overview of the Image Processing based Traffic Surveillance system.

3.1 Block diagram

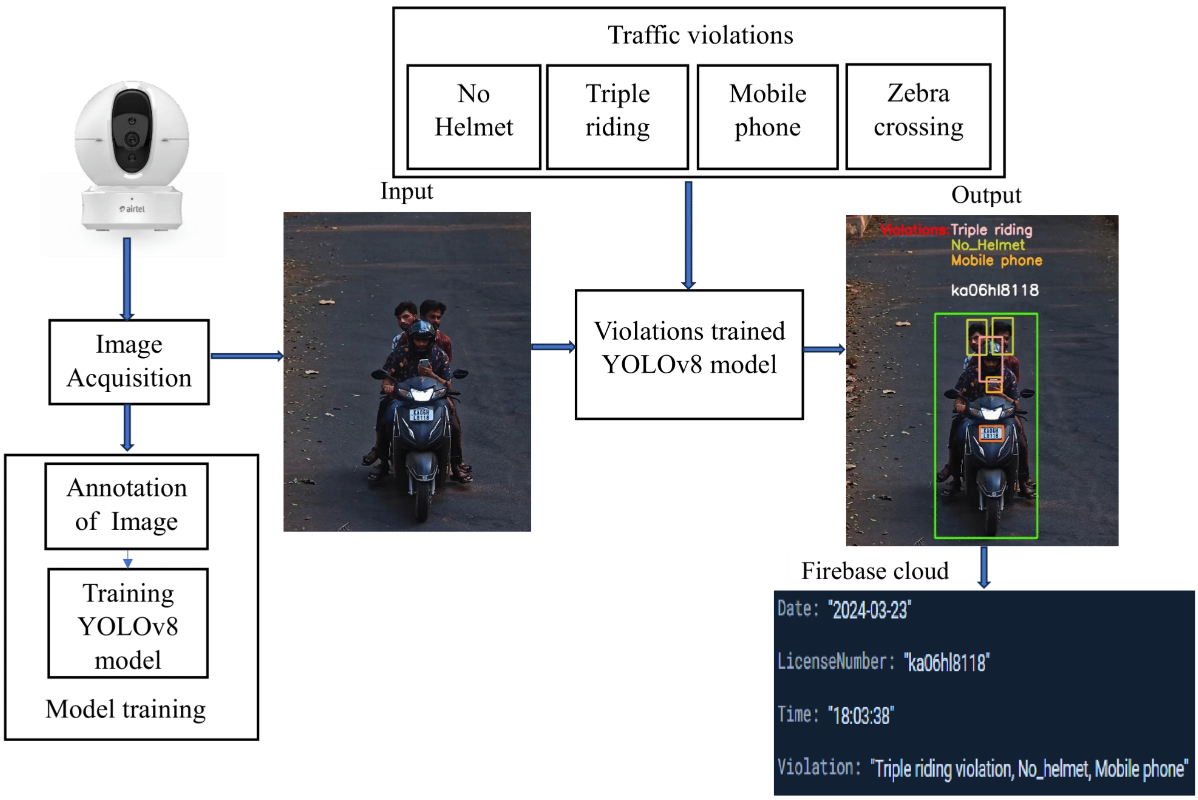


Figure 3.1: Block diagram of the proposed system

The block diagram in Figure 3.1 illustrates the design of the Image Processing Based Traffic Surveillance system, aimed at efficiently monitoring traffic scenarios and enforcing safety regulations. Here's an in-depth explanation of each stage:

1. Camera Setup: Three strategically positioned cameras capture high-resolution video feeds of traffic scenes from different angles. Each camera records at a resolution of 4MP at 30 frames per second (fps). The cameras are mounted at a height of 12 feet to provide a comprehensive view of the traffic flow.

2. Image Acquisition: The high-resolution video feeds are processed frame by frame to extract images. These images serve as the input data for the surveillance system.

3. Annotation with Roboflow: Before training the model, the images are annotated using Roboflow. Annotation involves labeling objects of interest in the images, such as motorcycles, cars, persons, helmets, and zebra crossings. This step is crucial for training the model to recognize and differentiate between different objects.

4. Image Resizing: To streamline processing time without compromising accuracy and resolution, the annotated images are resized to a standard size, such as 640x640 pixels. This standardization ensures that the model receives consistent input data during training.

5. Model Training with Google Colab: The resized and annotated images are used to train the model using Google Colab. The proposed algorithm utilizes the YOLOv8 model, which is pre-trained on a dataset of traffic violations. YOLOv8 is chosen for its speed, accuracy, and ability to detect small objects like helmets and license plates.

6. Vehicle Detection: Once the model is trained, it is deployed to detect vehicles in the video feed of the traffic scene, with a primary focus on motorcycles and cars. The algorithm employs person detection for motorcycles to identify violations such as no helmet, triple riding, mobile phone usage while riding, and zebra crossing lane violation. Similarly, for cars, the algorithm checks for violations such as surpassing the zebra crossing lane. If a violation is detected, the model sends the information of the violated vehicle to the OpenALPR model for license plate extraction.

7. License Plate Extraction: When a violation is detected, the system captures the violator and extracts the license plate information using the OpenALPR model. OpenALPR is a software library for license plate recognition.

8. Data Storage in Firebase: The extracted data, including the license plate number, violation type, fine amount for the violation, along with the date and time stamp, is stored in the Firebase cloud for future analysis. This allows for the tracking and analysis of traffic violations over time.

By dividing the process into these stages, the Image Processing Based Traffic Surveillance system can efficiently monitor traffic scenarios, enforce safety regulations, and contribute to improved road safety.

Chapter 4

System Hardware

This chapter outlines the hardware requirement for capturing videos in the traffic.

4.1 Surveillance camera

Figure 4.1: Surveillance camera [12].

A 360° camera with an resolution of 1920 x 1080 pixels is shown in Figure 4.1, this camera excels in capturing intricate panoramic views, facilitating comprehensive traffic monitoring. Its expandable memory, supporting Micro SD cards up to 32 GB, ensures prolonged and uninterrupted recording of traffic data. Operated by a stable 5V/2A input, the camera guarantees reliable performance while capturing expansive perspectives through its wide-angle lens, offering a 110° field of view, ideal for monitoring traffic flow. With night vision capabilities and a robust weatherproof design, the camera excels in effective surveillance across diverse environmental conditions. Compatibility enhances operational flexibility, enabling remote monitoring and control. The cameras seamless integration with different protocols optimizes overall system efficiency, establishing it as a pivotal asset in improving traffic management and ensuring the safety and security.

Chapter 5

System Software

This chapter outlines the flowchart, Algorithm and software requirement for the traffic surveillance system.

5.1 Flowchart

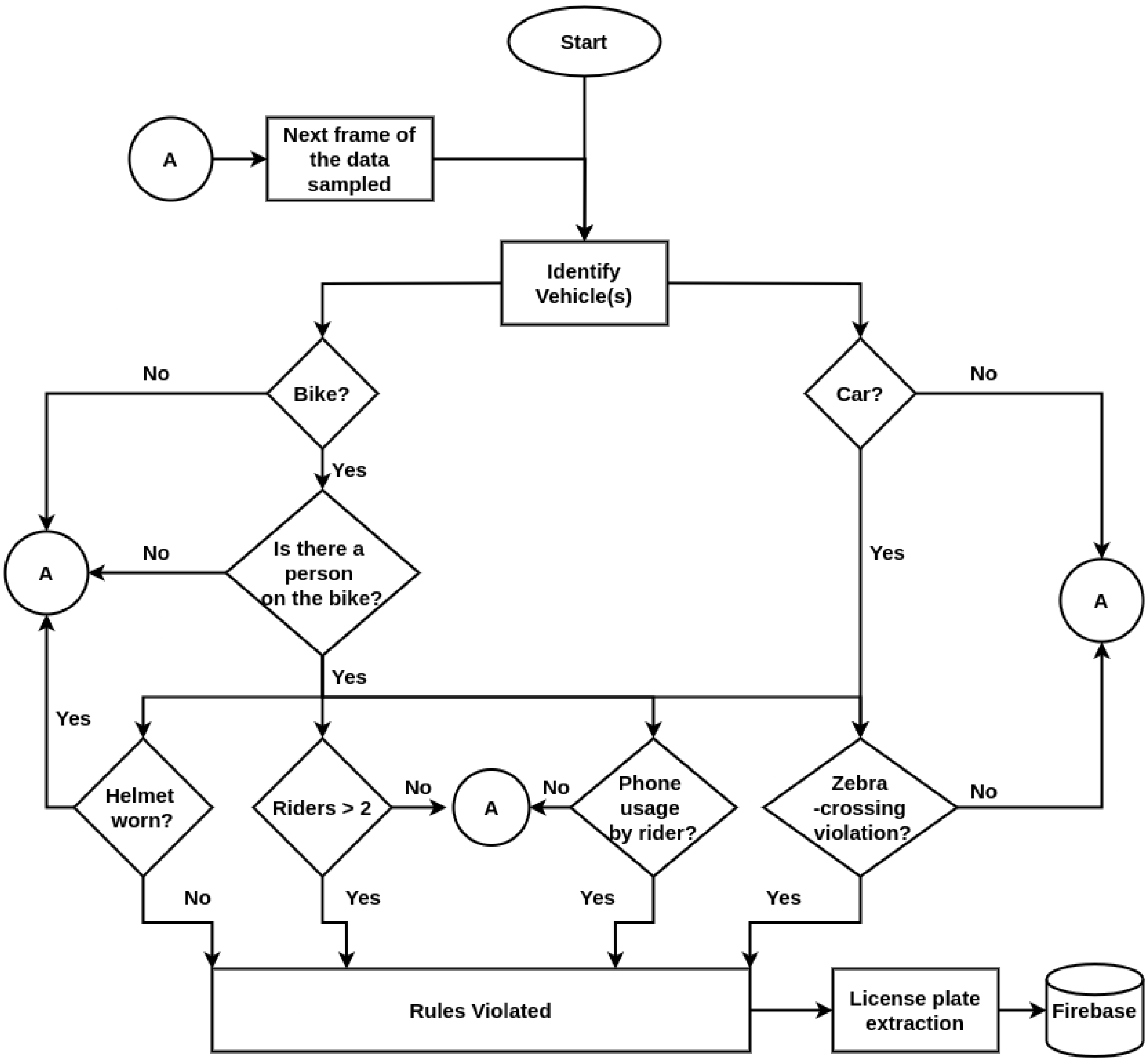


Figure 5.1: Flowchart of the proposed system

Figure 5.1 shows the comprehensive workflow of the Image Processing based Traffic Surveillance system. The process commences with input video of traffic, where the system differentiates between bikes and cars within the video stream. In the case of bikes, the model proceeds to identify violations such as riding without a helmet, triple riding, mobile phone usage, and zebra crossing lane violation by the rider. If any violations are detected, the rule violators are captured. Similarly, the algorithm utilizes YOLOv8 to detect cars for zebra crossing rule violations. For both bikes and cars violators, an OpenALPR model is applied to extract the license plate information. Following the extraction of license plate data, the information is sent to Firebase Cloud for secure storage. This integrated approach ensures a thorough analysis of traffic violations and the efficient management of data for subsequent actions.

5.2 Algorithm

The algorithm of the proposed model is as follows:

1. Receive a video feed of a traffic scene as the algorithm input with a resolution of 4MP at 30 fps, with the camera placed at a height of 12 feet.
2. Process the video to reduce the resolution to 1850 x 1046 and pass it to the model.
3. Use the trained YOLOv8 model for vehicle detection, specifically focusing on identifying motorcycles and cars within the scene.
4. If a motorcycle is detected, proceed to further analysis for potential violations. Employ the YOLOv8 model for person detection to identify individuals on the motorcycle.
5. Check for violations related to motorcycle riders, such as no-helmet, triple riding, mobile phone usage while riding, and surpassing zebra crossings. Employ the trained YOLOv8 model for each specific violation detection.
6. If a car is detected, employ the YOLOv8 model to detect zebra cross violations.
7. Extract the license plate information of the vehicle using the OpenALPR model to identify rule violators.
8. Store the extracted license plate information along with the violation type, fine amount for the violation, and timestamp in the Firebase Cloud database for further processing and analysis.
9. Proceed to the next frame for continuous monitoring and analysis.

This algorithm integrates YOLOv8 models for vehicle, person, and violation detection, along with an OpenALPR model for license plate recognition, ensuring comprehensive monitoring and enforcement of traffic regulations.

5.3 Roboflow

Roboflow is a versatile platform tailored to streamline the complex process of handling image datasets for machine learning purposes. By providing a suite of tools, it simplifies data annotation, enabling users to label and categorize images effectively, crucial for training supervised learning models. Roboflow facilitates data augmentation through various techniques, empowering users to generate diverse versions of their datasets to enhance model robustness. It simplifies data transformation, converting images into formats compatible with different machine learning frameworks.

5.4 Google colab

Google Colab, part of the Google Cloud ecosystem, offers a powerful environment for developing and deploying machine learning models for traffic surveillance. With its integration with Google Drive, everyone can easily access and share datasets, code, and trained models across team members. Furthermore, Colab's ability to execute code on powerful GPUs and TPUs accelerates model training and inference, making it an ideal platform for handling the computational demands of image processing tasks in real-time traffic monitoring systems. Additionally, its support for Markdown and LaTeX allows for the creation of rich, interactive documentation within Jupyter notebooks, facilitating comprehensive project documentation and reporting.

5.5 PyCharm

PyCharm is a robust integrated development environment (IDE) for Python, offering intelligent code assistance, powerful tools for debugging and testing, and seamless integration with popular version control systems. Its user-friendly interface enhances productivity, making it a preferred choice for Python developers.

5.6 Ultralytics

Ultralytics is a renowned platform dedicated to advancing computer vision research and development through its innovative tools and resources. It offers a comprehensive suite of solutions tailored to empower researchers, developers, and data scientists in their pursuit of cutting-edge machine learning models for image analysis tasks. From data preprocessing and model training to deployment and inference, Ultralytics offers a seamless experience. Its focus on open-source frameworks ensures accessibility and flexibility, allowing users to customize and extend their models according to specific project requirements. Moreover, Ultralytics emphasizes performance optimization, leveraging techniques such as mixed precision training and distributed training to maximize efficiency and scalability. Its open source frameworks, such as PyTorch and TensorFlow, combined with pre-trained models, enable users to kickstart their projects with ease.

5.6.1 You Only Look Once(YOLO)

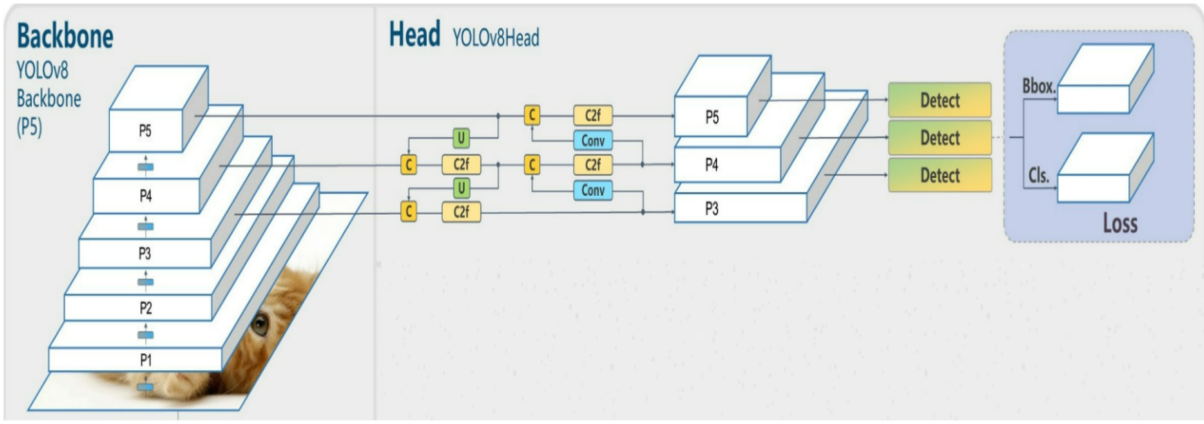


Figure 5.2: YOLOv8 network architecture [13].

Object detection represents a formidable challenge in the field of computer vision. Despite the existence of several algorithms dedicated to this task, YOLO stands out as one of the swiftest and most accurate options. YOLO, an acronym for “YOU ONLY LOOK ONCE,” is a cutting-edge, real-time object detection algorithm employed for identifying objects in images, videos, or live feeds. What sets YOLO apart is its ability to provide rapid and precise object recognition. The YOLO models are trained on complete images, directly optimizing the efficiency and performance of object detection. YOLO goes beyond mere object detection in images, it excels in identifying the precise locations of objects within those images. Illustrated in Figure 5.2, the network architecture of YOLOv8 underlines its innovative approach to addressing object detection challenges.

YOLOv8 streamlines object detection with a single neural network. Input images are processed directly by the deep convolutional neural network (CNN), which predicts bounding boxes and class probabilities for multiple objects in the image grid as in Figure 5.3. Filtering out low-confidence predictions, it employs non-maximum suppression to remove redundant bounding boxes, delivering high-confidence detections for object locations and classes.

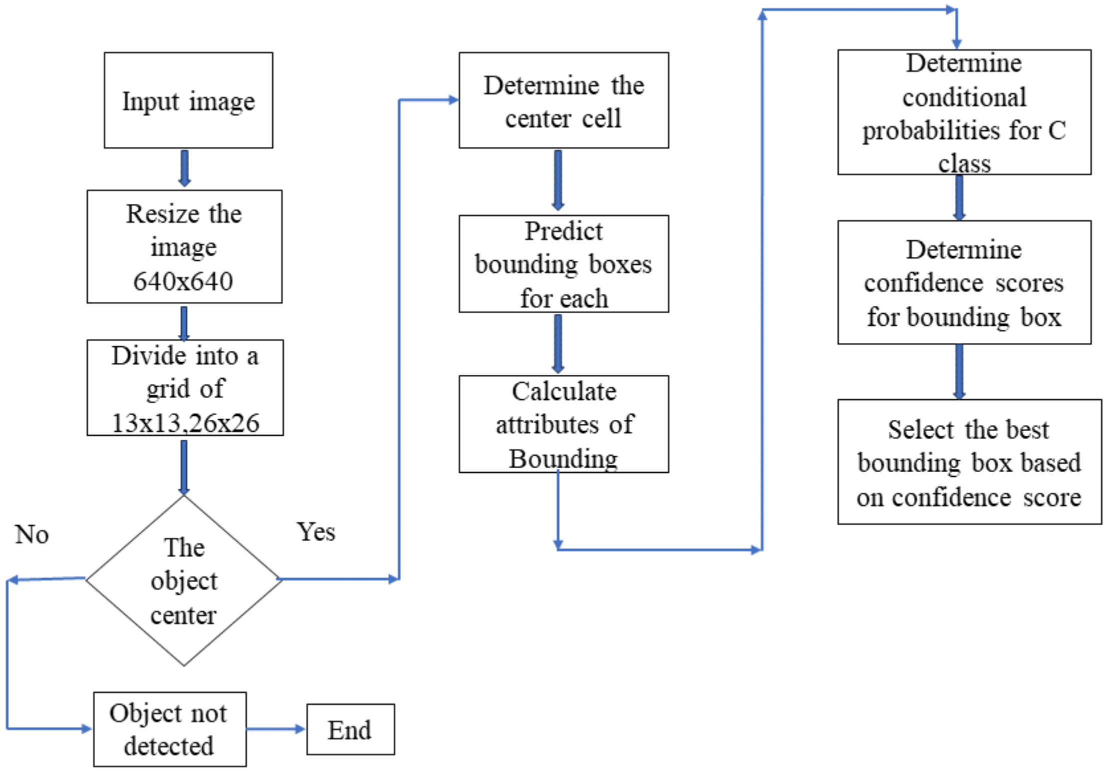


Figure 5.3: The workflow of YOLOv8.

5.7 OpenALPR

OpenALPR begins by acquiring images or video frames through cameras strategically placed in areas like traffic intersections or parking lots. These images undergo preprocessing to enhance quality and standardize inputs. Preprocessing steps include resizing, noise reduction, and contrast enhancement to optimize images for subsequent analysis. Following preprocessing, features such as edge detection and character segmentation are extracted. Edge detection algorithms identify object edges, while character segmentation isolates license plate characters. License plate localization in OpenALPR involves utilizing computer vision algorithms to detect and extract license plate regions from images or video frames, facilitating automatic identification and recognition of vehicle license plates. This process employs techniques such as edge detection, contour analysis, and pattern recognition to accurately locate license plates within the given input. These features are then fed into the character recognition module, which employs machine learning techniques like convolutional neural networks (CNNs) or optical character recognition (OCR) to accurately identify alphanumeric characters.

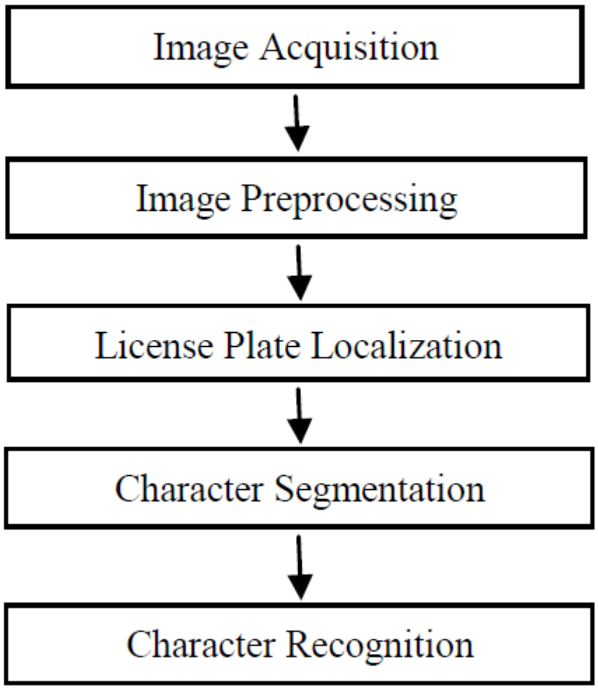


Figure 5.4: Working of OpenALPR [14].

After character recognition, post-processing techniques refine results and enhance accuracy. Error correction, pattern matching, and contextual analysis are applied to validate license plate numbers. Post-processing algorithms consider spatial relationships between characters to filter out false positives, ensuring reliable identification as in Figure 5.4. Finally, recognized license plate numbers, along with metadata like timestamps and confidence scores, are generated as output. OpenALPR offers flexible integration options, providing APIs and SDKs for seamless integration into existing software applications. It supports deployment on various hardware platforms, including embedded systems and cloud servers, making it adaptable to diverse use cases.

5.8 Firebase

Firebase-admin is the official library to interact with Firebase from server-side environments. Firebase offers a suite of cloud-based services including a database, user authentication, and hosting, which can be utilized by applications for scalable and flexible development. In traffic surveillance, Firebase-admin can be utilized to store and manage real-time data collected from surveillance, such as vehicle counts, timestamps of traffic violations, and even stream video data. It facilitates the development of a backend infrastructure where data can be accessed and managed securely from anywhere.

5.9 Libraries Used

The libraries used to excecute the project are listed below.

5.9.1 OpenCV

OpenCV is an acronym for the Open-source Software Library for Computer Vision, offering compatibility with operating systems like Windows, Linux, Mac, and Android. Featuring over 2000 optimized algorithms, OpenCV excels in tasks such as facial recognition, object detection, and tracking camera movements, making it a preferred choice over MATLAB. Its extensive open-source library for image processing, computer vision, and machine learning plays a pivotal role in real-time performance, a critical aspect in modern systems. With capabilities to process videos and photos, OpenCV proves invaluable in identifying elements for real-life applications. When combined with libraries like Numpy, Python becomes a potent tool for analyzing OpenCV array structure.

5.9.2 Datetime

The datetime module in Python provides classes for manipulating dates and times in both simple and complex ways. It's essential for handling time-related tasks, such as timestamping events or scheduling. In traffic surveillance, the datetime module can be used to timestamp traffic incidents, log the time vehicles enter or exit specific zones, and correlate traffic patterns with time variables. This temporal data is crucial for analyzing traffic flow, predicting peak hours, and scheduling maintenance or security measures.

5.9.3 Torch

Torch, a prominent library in the Python ecosystem, is renowned for its versatility and efficiency in deep learning and scientific computing tasks. Torch offers a rich set of tools and modules tailored for machine learning research and application development. Its seamless integration with GPUs accelerates computation, making it a preferred choice for training large-scale models. With an active community and extensive documentation, Torch continues to be a cornerstone in the deep learning landscape, facilitating advancements in various domains such as computer vision, natural language processing, and reinforcement learning.The torch library is utilized for loading the YOLOv8 model, performing inference on input images or videos, and processing the predictions.

Chapter 6

Results and Discussion

This chapter includes the training, testing ,results obtained from the YOLOv8 model and model detection analysis for various traffic violations.

6.1 Training phase

1. Dataset gathering: A comprehensive dataset comprising 50 videos totaling 100 hours was collected from traffic video footage captured in Tumkuru city main roads and SIT campus. From this footage, a total of 2300 images were extracted, ensuring a diverse set of scenarios and angles, totaling 20 possibilities for training a custom object detection model to detect traffic violations. This approach provides a robust training and testing dataset, enhancing the model's ability to accurately detect violations in real-world traffic conditions.

2. Annotation of Images:

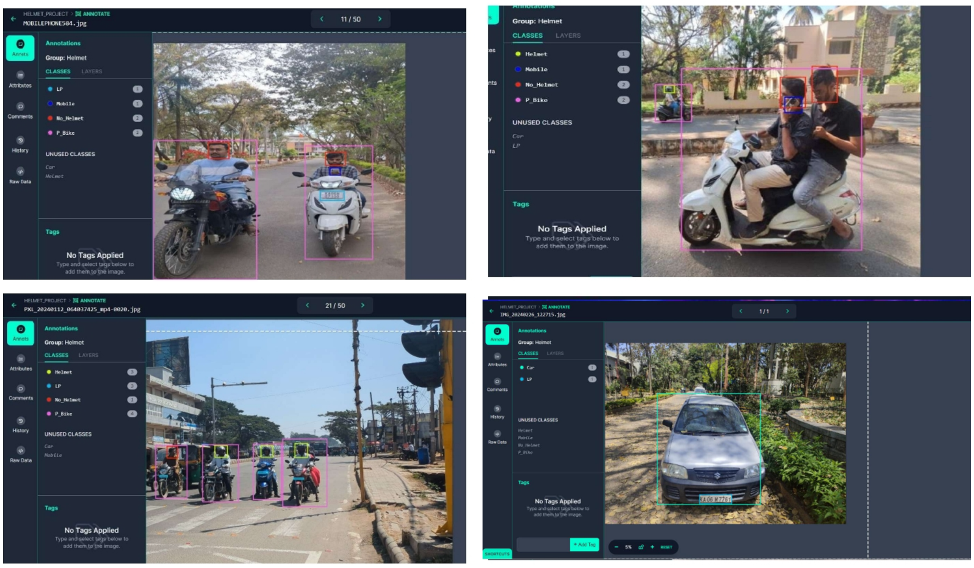


Figure 6.1: Labeling on Roboflow for P_Bike, No_helmet, LP, Mobile, Helmet, Zebra & Car.

The dataset was annotated into seven classes: 'Car', 'Helmet', 'LP' (license plate), 'Mobile', 'No Helmet', 'Zebra', and 'P Bike' as in Figure 6.1. Bounding boxes were manually drawn around objects of interest, such as persons, bikes, mobile phones, helmets, and zebras in an image. Annotation was done using the YOLO format with the assistance of the Roboflow tool, resulting in YOLOv8 formatted labels. The images were resized to 640 x 640 and divided into 80% training, 18% validation, and 2% testing sets for training the YOLOv8 model, which was used to detect all seven classes for comprehensive traffic violation detection.

3. Model Training: The training process involved developing specialized YOLOv8 models for multiple detection tasks, including identifying individuals, bikes, helmets, mobile phone, car, zebra, and license plates. Images and their associated labels were carefully arranged and then uploaded to Google Drive for training the models using Google Colab. The training was conducted over 100 epochs using the YOLOv8n model. After completing the training, configuration files containing crucial parameters and weights files were obtained. These files are essential for executing prediction tasks, ensuring the effective deployment and utilization of the trained models.

6.2 Testing phase

1. Prediction:

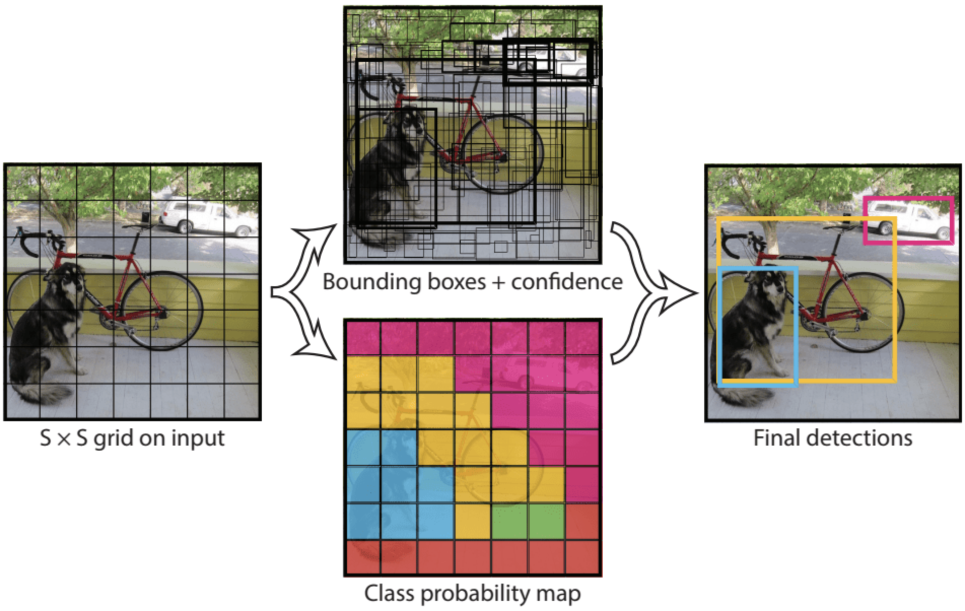


Figure 6.2: Processing and prediction of raw image [15].

The process of predicting with YOLOv8, as illustrated in Figure 6.2, can be broken down into the following steps:

1. Feature Extraction: The image is resized to 640 x 640 and fed through the network backbone architecture to extract features from the input image, such as edges and textures, patterns and shapes, object parts, scale, and orientation.

2. Multi-Scale Feature Fusion: YOLOv8 incorporates multi-scale feature fusion to combine features from different layers of the backbone network. This allows the model to capture both low-level and high-level features, which is crucial for detecting objects of different sizes and shapes.

3. Grid Division: The images are divided into S x S grids (e.g., 13x13, 26x26 cells).

4. Bounding Box Center and Size Prediction: For each cell, the network predicts a set number of bounding boxes with:

Class probabilities: The likelihood of each class for each box.

Bounding box coordinates: These represent the box's location and size relative to the cell, calculated using equations:

- Center coordinates (tx, ty):

$$bx = \sigma(tx) + cx$$

$$by = \sigma(ty) + cy$$

- Width and height (tw, th):

$$bw = pw \cdot e^{tw}$$

$$bh = ph \cdot e^{th}$$

where:

- bx, by are the center coordinates of the bounding box.
- bw, bh are the width and height of the bounding box.
- tx, ty, tw, th are the outputs of the neural network.
- cx, cy are the coordinates of the top-left corner of the anchor box.
- pw, ph are the dimensions of the anchor box.

5. Decoding Predictions: Predicted attributes are decoded to obtain absolute coordinates and dimensions within the original image.

6. Non-Maximum Suppression (NMS): If multiple boxes predict the same object, NMS removes redundant boxes, keeping only the one with the highest confidence score.

7. Confidence Score Calculation: It is used to indicate the model's certainty that a bounding box contains an object and that object belongs to a specific class is shown in

Equation 6.1

$$\text{Confidence Score} = \sigma(\Pr(\text{Object}) \times \text{IoU}) \tag{6.1}$$

where:

- $Pr(Object)$ is the probability that an object exists in the bounding box.

- IoU(Intersection over Union) formula is used to calculate the overlap between two bounding boxes. It is shown in Equation 6.2.

$$\text{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{6.2}$$

These equations and steps are used in YOLOv8 to predict bounding boxes, object-ness scores, and class probabilities for objects detected in an image, contributing to its efficiency and accuracy.

2. Evaluation:

The evaluation of the performance of the models is done by comparing the Precision, Recall, and mAP (Mean Average Precision). The evaluation metrics can be understood by knowing the concept of the Confusion matrix.

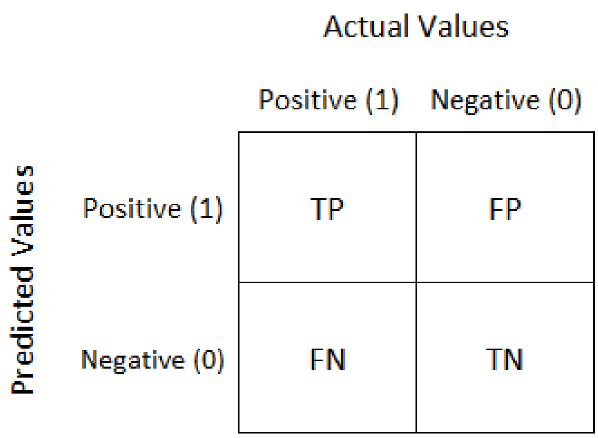


Figure 6.3: Confusion matrix [Courtesy: www.google.com].

Confusion matrix is a performance measurement for classification problem with two or more classes. Figure 6.3 shows the confusion matrix, it has 4 different combinations of predicted and actual values. They are,

- True Positive : Predicted value is positive and the actual value is true given by TP.
- True Negative : Predicted value is negative and the actual value is true given by TN.

- False Positive : Predicted value is positive and actual value is false given by FP.
- False Negative : Predicted value is negative and actual value is false given by FN.

Equations 6.3 to 6.7 use TP, TN, FP, FN, N, and APi to calculate the Precision, Recall, and mAP. APi represents Average Precision of class i and N is the number of classes.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6.3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6.4}$$

$$AP = \int_0^1 P(R)\, dR \tag{6.5}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i \tag{6.6}$$

$$mAP@50\% = \frac{1}{N} \sum_{i=1}^{N} \text{AP@0.5}_i \tag{6.7}$$

where AP is the area under the precision-recall curve for a category at different confidence thresholds. mAP is the mean average precision, obtained by averaging AP for each category. mAP@50% is the mAP with an Intersection over the Union (IoU) threshold of 0.5.

3. Detection: After evaluating the input image, the model detects all object classes. It filters out only the required classes, each represented by a bounding box, a class label, and a confidence score. The model then identifies the 'P_bike' class and checks for other classes inside that class's bounding box for violation detection. If the 'Helmet' and 'No_Helmet' classes are greater than 2, it's considered triple riding. If the 'Mobile' class is found, or if the 'No_Helmet' class is found, or if the 'P_bike' class and the 'Zebra' class bounding boxes touch each other, it's considered a zebra violation. If any violation is detected, the model sends the 'P_bike' image to the openALPR model for license plate detection. If a car is detected, it finds the bounding box of the car and the zebra crossing and sends the car image to the openALPR model to extract the license plate information. If no vehicle is detected, it moves to the next frame for processing. These metrics can help assess the system's effectiveness in detecting violations and can be used to optimize its performance.

6.3 Results

The confusion matrix, as depicted in Figure 6.4, gives a detailed breakdown of the YOLOv8 models performance, precision, recall, and overall accuracy across various categories. The matrix encompasses seven classes: Car, Mobile, P_bike, Helmet, No_helmet, Zebra, and LP[License Plate]. It showcases true labels along the rows and predicted categories along the columns, with correct detection's indicated by diagonal elements.

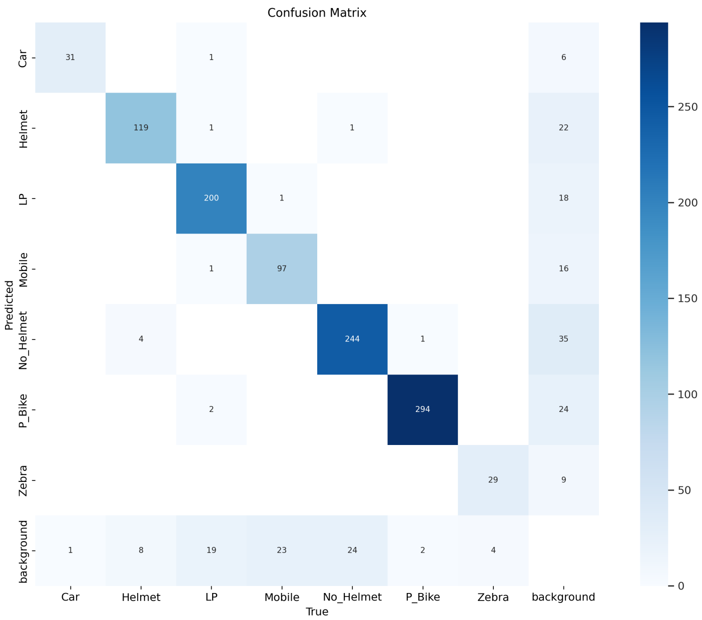


Figure 6.4: The confusion matrix

Table 6.1: Validation results

Classes	Instances	Precision	Recall	mAP score
All	4078	0.901	0.89	0.913
Car	212	0.845	0.943	0.934
Helmet	569	0.94	0.885	0.937
License Plate	900	0.932	0.861	0.928
Mobile	221	0.877	0.8	0.847
No_Helmet	769	0.924	0.874	0.93
P_bike	1297	0.94	0.99	0.985
Zebra	110	0.832	0.788	0.832

Table 6.1 shows the validation outcomes for multiple metrics such as precision, recall,

and mAP50, illustrating the model's efficacy in object detection. The model obtained a good accuracy across all classes, with a total of 4078 instances in the validation dataset. These results serve as strong evidence for the model's effectiveness in object detection. Additionally, the table presents the metrics for each class alongside the corresponding instance counts, emphasizing the model's accurate object detection across diverse classes.

6.4 Model detection analysis

The model was tested on approximately 50 videos with total duration of 100 hours, depicting traffic scenarios in Tumakuru and on the scenarios in SIT campus, considering 20 possibilities in the videos. The models performance was as follows: Motorcycle detection accuracy is 99%. Helmet detection with 99% accuracy. No_helmet detection accuracy is 98%. Mobile phone usage detection accuracy is 98%. Triple riding detection accuracy is 96%. License plate extraction accuracy for both bikes and cars is 93%. Zebra Crossing violation detection accuracy is 90%. The system has achieved an overall accuracy of 97% in detecting the traffic offenders. This system provides a sophisticated approach for the betterment and safety towards road safety.

Figure 6.5: Detecting Helmeted and No helmeted riders in a traffic scenario.

(a) (b)

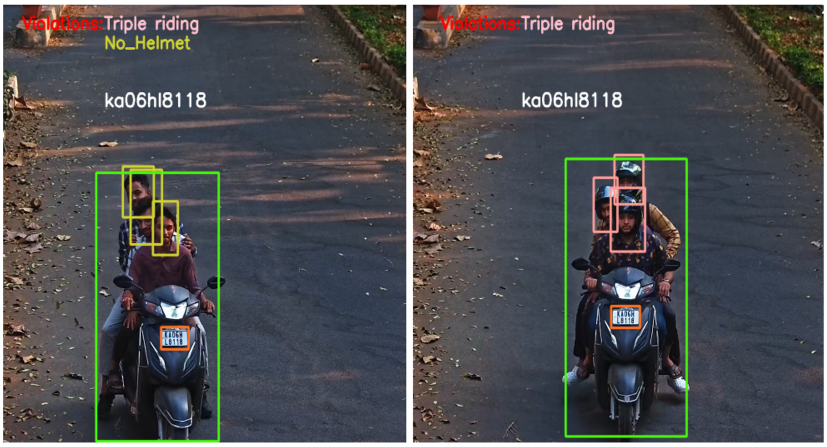
Figure 6.6: No violation (a) & extracting the license plate for no helmet violation (b).

(a) (b)

Figure 6.7: No violation (a) & extracting the license plate for a two people violating the helmet rule (b).

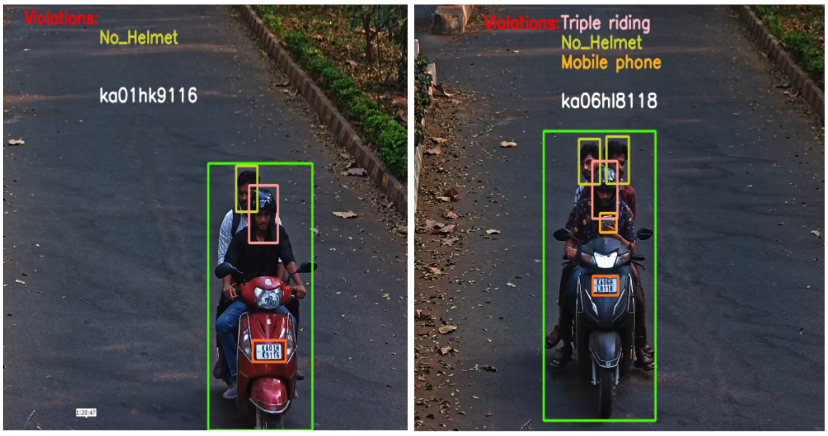
(a) (b)

Figure 6.8: Extracting license plate for mobile phone violation (a) and no helmet with mobile phone violation (b).



(a) (b)

Figure 6.9: Extracting the license plate for triple riding & no helmet (a) and triple riding violation (b).



(a) (b)

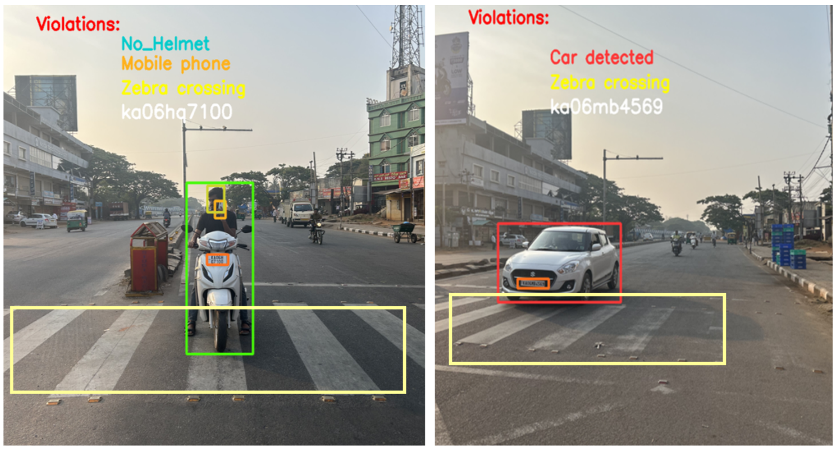
Figure 6.10: Extracting the license plate for no helmet (a) & for triple riding, no helmet with mobile phone violation (b).

(a) (b)

Figure 6.11: Extracting license plate for car (a) (b)& No violation of bike (b).

(a) (b)

Figure 6.12: No Zebra cross lane violation by car (a) & bike (b).



(a) (b)

Figure 6.13: Extracting License plate for Zebra cross lane violation by bike (a) & car (b).

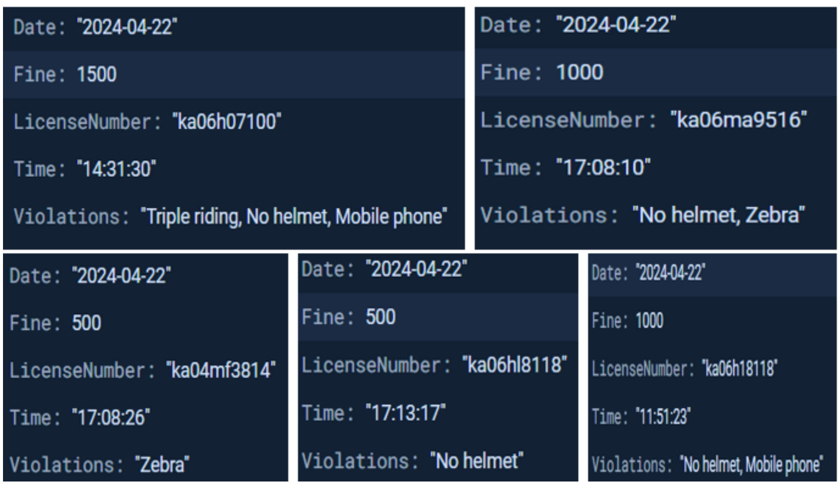


Figure 6.14: Rule violators information in Firebase cloud.

Case 1: Multiple vehicles with helmet and no helmet riders in traffic scenario (Figure 6.5).
case 2: No violation- single rider with helmet (Figure 6.6(a), Figure 6.11(b)), two riders with helmet (Figure 6.7(a)).
Case 3: Violation & license plate extraction- no helmet (Figure 6.6(b), Figure 6.7(b), Figure 6.10(a)), mobile usage (Figure 6.8(a)), mobile usage & without helmet (Figure 6.8(b)), triple riding & without helmet (Figure 6.9(a)), triple riding (Figure 6.9(b)), triple riding, no helmet & mobile (Figure 6.10(b)).
Case 4: License plate extraction of car (Figure 6.11(a), Figure 6.11(b)).
Case 5: No zebra cross lane violation by car (Figure 6.12(a)) & bike (Figure 6.12(b)).
Case 6: Extracting the license plate for the zebra cross violation by bike (Figure 6.13(a)) and car (Figure 6.13(b)).
The traffic rule violators' information such as license plate of the vehicles, violation type, date & time along with the fine amount is recorded in Firebase cloud as shown in Figure 6.14.

Table 6.2: Penalty for traffic rules violators.

SL. No	**Violations**	**Fine amount in rupees**
1	No Helmet	500
2	Mobile phone	500
3	Triple riding	500
4	Zebra-crossing offence	500

Table 6.2 shows the penalty amount for traffic rule violations. Using the license plate information a penalty can be enforced on the rule violators informing them the details of violation such as date, time,type of violation committed.
The confusion matrix for motorcycle detection is presented in Table 6.3. Table 6.4 shows the confusion matrix for helmet and no helmet detection. Table 6.5 displays the confusion matrix for mobile phone usage detection. Table 6.6 illustrates the confusion matrix for triple riding detection. The confusion matrix for violations detection is depicted in Table 6.7 and Table 6.8 showcases the confusion matrix for license plate extraction.

Table 6.3: Confusion Matrix for Motorcycle Detection

Actual Class %	**Predicted Class %**	
	Motorcycle	Other
Motorcycle	99	1
Other	1	99

Table 6.4: Confusion Matrix for Helmet and No_helmet Detection

Actual Class %	**Predicted Class %**	
	Helmet	No Helmet
Helmet	99	1
No Helmet	2	98

Table 6.5: Confusion Matrix for Mobile Phone Detection

Actual Class %	**Predicted Class %**	
	Mobile Phone	No mobile phone
Mobile Phone	98	2
No mobile phone	2	98

To evaluate the model's performance, the confusion matrix was calculated using the output from testing the model on 50 videos, totaling approximately 100 hours of footage and 2300 photos were extracted in the due process. The evaluation considered all possible violation scenarios, including motorcycle detection, helmet and no helmet detection, mobile phone usage detection, triple riding detection, violations in the rule with no violations, and license plate extraction. Each case was analyzed to determine how accurately the model predicted the correct output. The confusion matrix provides a detailed breakdown of the model's performance, showing the number of true positives, true negatives, false positives, and false negatives for each class, allowing for a comprehensive assessment of the model's accuracy and effectiveness in detecting violations and extracting license plate information.

Table 6.6: Confusion Matrix for Triple Riding Detection

Actual Class %	**Predicted Class %**		
	1 rider	2 riders	More than 2 riders
1 rider	99	1	0
2 riders	2	98	1
More than 2 riders	1	3	96

Table 6.7: Confusion Matrix for Violation Detection

Actual Class %	**Predicted Class %**	
	Violation	No Violation
Violation	97	3
No Violation	3	97

Table 6.8: Confusion Matrix for License Plate Extraction

Actual Class %	**Predicted Class %**	
	LP match	1 to 2 Character Mismatch
LP match	93	7
1 to 2 Character Mismatch	7	93

Chapter 7

Conclusion

The proposed project "Image Processing based Traffic Surveillance" is designed to identify various offenses including lack of helmet usage, triple riding, mobile phone usage while riding motorcycles, and violations at zebra crossings has been successfully implemented. This marks a significant advancement in improving road safety and traffic management. Utilizing advanced technologies such as YOLOv8 for object detection and OpenALPR for license plate recognition, in combination with image processing and Firebase-admin for data management, the project offers a comprehensive solution for monitoring and enforcing traffic regulations.

The model demonstrates proficiency in automatically detecting traffic violations, achieving an overall accuracy rate of 97%. Specifically, it excels in identifying helmeted riders with 99% accuracy, non-helmeted riders with 98% accuracy, instances of triple riding with 96% accuracy, mobile phone usage with 98% accuracy, and zebra crossing violations with 90% accuracy, while also recognizing license plate information with 93% accuracy.

By harnessing the capabilities of these technologies, the system efficiently identifies and records instances of traffic violations, facilitating swift intervention by traffic authorities. Timestamps generated using date and time ensure accurate documentation of incidents, enabling further analysis and reporting, all information are stored securely in Firebase cloud.

7.1 Scope for future work

The project can be optimised in the future by the following ways.

1. The offences committed could be identified and an SMS can be directly sent to the traffic offender with the fine amount and payment link.

2. This project lays the foundation for future advancements in smart transportation systems like traffic flow monitoring, anomaly detection and parking management.

Bibliography

[1] K. Dahiya, D. Singh and C. K. Mohan, "*Automatic detection of bike-riders without helmet using surveillance videos in real-time*" 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, pp. 3046-3051, 2016.

[2] A. Saumya, V. Gayathri, K. Venkateswaran, S. Kale and N. Sridhar, "*Machine Learning based Surveillance System for Detection of Bike Riders without Helmet and Triple Rides*" 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, pp. 347-352, 2020.

[3] M. Dasgupta, O. Bandyopadhyay and S. Chatterji,"*Automated Helmet Detection for Multiple Motorcycle Riders using CNN*" 2019 IEEE Conference on Information and Communication Technology, Allahabad, India, pp. 1-4, 2019.

[4] X. -H. Wu, R. Hu and Y. -Q. Bao, "*Block-Based Hough Transform for Recognition of Zebra Crossing in Natural Scene Images*" in IEEE Access, pp. 59895-59902, 2019.

[5] Mallela Nikhil Chakravarthy, Rohit Volety, and Nadesh R.K., "*Detection of the triple riding and speed violation on two-wheelers using deep learning algorithms*", Multimedia Tools and Applications, Springe, pp. 8175-8187, 2021.

[6] Mistry, Jimit, Aashishk and kishor P. Upla. "*An automatic detection of helemeted and non helemeted motorcyclists with license plate extraction using CNN*", in International Electrical Engineering Congress (iEECON), Chonburi, Thailand, pp. 1-6, 2014.

[7] H. Lin, J. D. Deng, D. Albers and F. W. Siebert, "*Helmet Use Detection of Tracked Motorcycles Using CNN-Based Multi-Task Learning*" in IEEE Access, pp. 162073-162084, 2020.

[8] H. Shi and D. Zhao, "*License Plate Recognition System Based on Improved YOLOv5 and GRU*" in IEEE Access, pp. 10429-10439, 2023.

[9] W. Riaz, A. Azeem, G. Chenqiang, Z. Yuxi, Saifullah and W. Khalid, "*YOLO Based Recognition Method for Automatic License Plate Recognition*" IEEE Interna-

tional Conference on Advances in Electrical Engineering and Computer Applications(AEECA), Dalian, China, pp. 87-90, 2020.

[10] A. N. Mukunda Rao, Nikhil Kumar, Gaurav Kumar Sahu, Ravi M, Sahil Kumar, Sukruth V, "*Triple Riding and No-Helmet Detection*", 2022.

[11] H. Yasar, "*Detection of Driver's mobile phone usage*" IEEE 9th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), Manila, Philippines, pp. 1-4, 2017.

[12] Camera image: `https://www.cctvcamerapros.com/4K-AI-camera-systems-s/1597.htm`

[13] Yolov8 architecture: `https://blog.roboflow.com/whats-new-in-yolov8/`

[14] OpenALPR workflow: `https://shorturl.at/nuAI8`

[15] Processing and prediction raw image: `https://shorturl.at/kquKW`

Appendices

Appendix A

Data Sheet of Surveillance camera

Figure A.1: Surveillance camera [Courtesy: Google.com].

Specification	Details
Resolution	4K (3840 x 2160 pixels)
Frame Rate	Up to 30 frames per second
Lens Type	Varifocal lens
Field of View	90 degrees
Night Vision	Infrared LEDs, up to 30 meters
Connectivity	Wi-Fi, Ethernet
Power Supply	PoE (Power over Ethernet)
Dimensions	150mm x 80mm x 80mm
Weatherproof Rating	IP66

Table 1.1: AI Vision Camera Specifications

Self Assessment

Table 7.2: Self Assessment of the project

Self Assessment of the project			
1	Engineering Knowledge: Knowledge of mathematics, engineering fundamentals engineering specialization to form of complex engineering problems	Learning mathematical expressions, analyzing the confusion matrix for model performance, and understanding the Non-Maximum Suppression (NMS) algorithm for reducing multiple bounding box detections are important aspects of image processing.	4
2	System Analysis: Identify, formulate, research literarture and analyse engineering problems to derive substaintiate conclusions by first principles of mathematics, natural and engineering science	The project aims to apply principles from mathematics, natural sciences, and engineering to automatically identify various traffic offenders and extract the license plate information of the violated vehicle.	5
3	Design/development of solutions: Design solutions of complex engineering problems, design system components or process that meet the specified process with appropriate consideration for the public health, safety and the cultural and environmental considerations.	The project contributes to designing solutions for complex engineering problems by developing a system that enhances public health and safety in traffic management. It considers cultural and environmental factors, aiming to improve road safety and compliance with traffic regulations.	5

Continued on next page

Table 7.2 Continued from previous page

4	Conduct investigations of complex problems: Use research based knowledge and research methods including design experiments, analysis and interpretation of data, and synthesis of information to provide valid conclusions.	The project, based on a thesis, employs an in-depth understanding of image processing and the YOLOv8 model. It uses a dataset of 2300 images to detect traffic violations, storing results for analysis, showcasing research-based problem-solving.	5
5	Modern tool usage: Create, insert and apply appropriate techniques, resources and modern engineering and tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	The project utilizes modern engineering tools, including YOLOv8 for object detection and the OpenALPR model for license plate extraction, demonstrating an understanding of tool limitations in complex engineering activities.	4
6	The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.	The project, through cost-effective automation, enhances road safety, contributing to a safer environment, demonstrating awareness of societal and safety concerns in engineering practice.	5
7	Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.	The project enhances road safety, reduces congestion, thus mitigating environmental impacts.The data generated enables informed decisions, promoting sustainable development and resilient, livable cities within smart city frameworks.	5

Continued on next page

Table 7.2 Continued from previous page

8	Ethics: Apply ethical principles and commit to professional ethics and norms of the engineering practice.	The project adheres to ethical principles and professional norms by managing teamwork effectively and completing work within provided time slots.	4
9	Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams, and in multidisciplinary settings.	Team members actively participate equally in the project, demonstrating effective individual and collaborative work skills in diverse and multidisciplinary settings.	5
10	Communication: communicate effectively on complex engineering activities with the engineering community and with the society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.	Effective communication is demonstrated through the use of LaTeX for documentation, plagiarism checks for reports, and clear presentation of findings to the engineering community and society at large.	4
11	Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	Applying engineering and management principles ensures projects stay on budget and schedule while effectively coordinating teams, demonstrating proficiency in project management and multidisciplinary environments.	5

Continued on next page

Table 7.2 Continued from previous page

12	Life-long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in broadcast context of technological change.	As the project is about the ongoing and upcoming technologies, further image processing along with YOLOv8 is applied to detect the traffic offenders like no helmet usage, triple riding, mobile phone usage and surpassing zebra lane.	5
13	PSO1: Apply the concepts of electronic circuits and systems to analyses and design systems related to Microelectronics, Communication, Signal processing and Embedded systems for solving real world problems	The project applied concepts of image processing, utilizing the NMS algorithm to remove multiple detections, showcasing practical application in solving real-world problems related to signal processing and embedded systems.	4
14	PSO2: To identify problems in the area of communication and embedded systems and provide efficient solutions using modern tools/algorithms working in a team	The project identifies and addresses problems in communication and embedded systems, providing efficient solutions using modern tools/algorithms. Results are stored in Firebase Cloud, including license plate numbers, violations, and timestamps, demonstrating teamwork and use of modern tools.	4

Level	Grade
poor	1
average	2
good	3
vgood	4
excellent	5

DrillBit Plagiarism Report

DrillBit

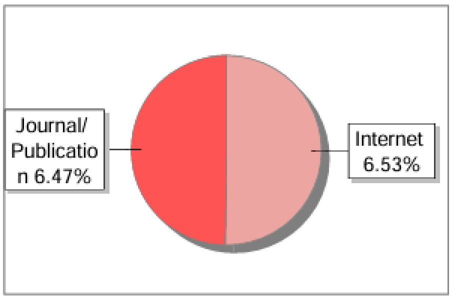
The Report is Generated by DrillBit Plagiarism Detection Software

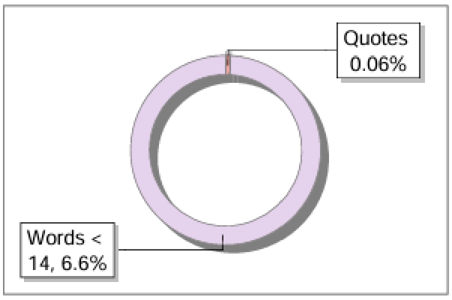
Submission Information

Author Name	chandrakanth
Title	Image Processing Based Traffic Surveillance
Paper/Submission ID	1700961
Submitted by	anmukund@sit.ac.in
Submission Date	2024-04-26 14:00:07
Total Pages	31
Document type	Project Work

Result Information

Similarity **13 %**





Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Sources: Less than 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	Yes

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