Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)

RECKLESS DRIVING DETECTION USING DEEP LEARNING

Submitted in partial fulfillment of the requirement
of the degree of

Bachelor of Technology in Information Technology

By

VARUN BHOSALE SAPID : 60003200068
JAINAM SHAH SAPID : 60003200070
PREM DOSHI SAPID : 60003200164

Under the guidance of

Dr. RAMCHANDRA MANGRULKAR
Prof ANUSHREE PATKAR PATIL

शीलवृत्तफला विद्या

University of Mumbai

A.Y. 2023 – 2024

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)

CERTIFICATE

This is to certify that the project entitled, **"RECKLESS DRIVING DETECTION USING DEEP LEARNING "** is a bonafide work of **Varun Bhosale (60003200068), Jainam Shah (60003200070), Prem Doshi (60003200164)** submitted in the partial fulfillment of the requirement for the award of the Bachelor of Technology in Information Technology.

Name of the Guide
Dr Ramchandra Mangulkar

Dr. Vinaya Sawant
Head of the Department

Dr. Hari Vasudevan
Principal

Place:Mumbai

Date:2/5/2024

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)

APPROVAL SHEET

Project entitled, **"RECKLESS DRIVING DETECTION USING DEEP LEARNING"**, submitted by **Varun Bhosale (6000320068), Jainam Shah (60003200070), Prem Doshi (60003200164)** is approved for the award of the Bachelor of Technology in Information Technology.

Signature of Internal Examiner **Signature of External Examiner**

Dr Ramchandra Mangulkar
Name of the Guide

Dr. Vinaya Sawant
Head of the Department

Dr. Hari Vasudevan
Principal

Place:Mumbai

Date: 2/5/2024

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all the principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources, which have thus not been properly cited or from whom proper permission has not been taken, when needed.

Varun Bhosale (60003200068)

Jainam Shah (60003200070)

Prem Doshi (60003200164)

Place:Mumbai

Date: 2/5/2024

Acknowledgement

We would like to share our sincere gratitude to all those who helped us in the completion of this project.

We would like to thank our college, Dwarkadas J. Sanghvi College of Engineering, for providing us with such an opportunity to learn from these experiences.

We would also like to show our gratitude to our Project Coordinator Dr. Ramchandra Mangrulkar and Prof. Anushree Patkar Patil for their continuous help and monitoring during the project work.

All our team is thankful to Dr. Vinaya Sawant and all the Faculties and Staff of the Department of Information Technology Engineering, Dwarkadas J. Sanghvi College of Engineering, for their help and support towards this project and our team.

We are also thankful to our whole class and most of all to our parents.

Varun Bhosale 60003200068
Jainam Shah 60003200070
Prem Doshi 60003200164

Abstract

This research study provides an advanced driver monitoring system that combines deep learning and computer vision techniques to detect and identify unsafe driving behaviors. The suggested solution combines DeepSort for object tracking across frames, YOLO for real-time object recognition, and OpenCV for image processing. The system preprocesses input video streams using OpenCV, identifying pertinent characteristics and improving image quality for further analysis. Modern object detection method YOLO is used to recognize and categorize different items in every frame, with particular emphasis on cars and their actions.DeepSort is used to overcome the difficulty of tracking objects between frames in order to keep precise and consistent vehicle tracking across time. This makes it possible for the system to examine the dynamics of every object it detects, which makes it easier to identify poor driving habits.The paper also presents a self-devised algorithm that is intended to identify driving actions that are risky. This algorithm accurately identifies instances of reckless driving by taking into account a variety of factors, such as abrupt accelerations, aggressive manoeuvres, and unpredictable lane changes.

Keywords: Lousy driver detection, OpenCV, YOLO, DeepSort, Reckless driving, Computer vision, Deep learning, Object detection, Driver monitoring system.

Table of Contents

List of Figures

List of Tables

Chapter 1 - Introduction

Road safety remains a significant global concern, with a startling number of fatalities and injuries occurring due to traffic accidents. The World Health Organization highlights India's alarming statistic, estimating that one in ten global traffic accident-related deaths originates from this region. The multifaceted nature of road accidents encompasses various factors, such as speeding, distracted driving, impaired operation, vehicle defects, weather conditions, and disregard for traffic regulations. Negligence by drivers and pedestrians accounts for a considerable portion of these incidents, leading to both human and financial losses. Addressing this issue requires proactive identification and intervention in real-time to curb reckless driving practices.

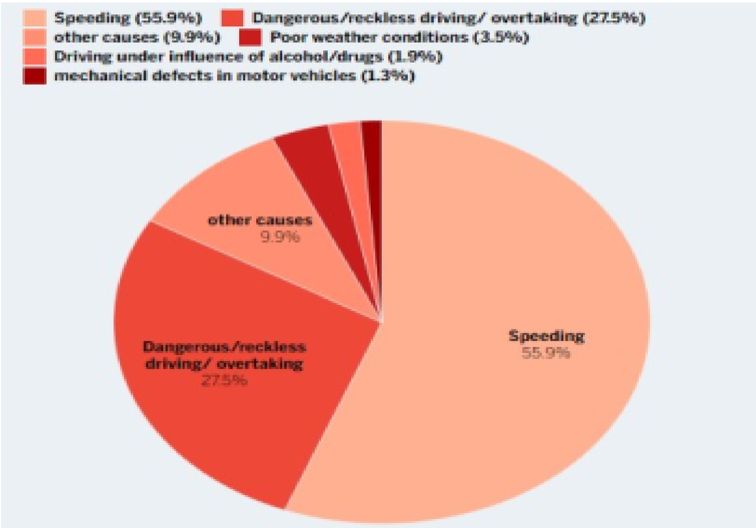

Fig 1.1 Reasons for Road Accident

The proposed system's framework encompasses the integration of these technologies to enhance real-time identification and prediction of reckless driving practices, contributing significantly to the enforcement of traffic safety measures.

1.1. Motivation / Objective

The imperative need to improve road safety and handle the mounting issues related to reckless driving behaviors is what inspired the creation of a reckless driving prediction system. The principal aim is to capitalize on the progress made in computer vision technology to provide a proactive and all-encompassing solution capable of promptly recognizing and forecasting instances of imprudent driving. Public safety is seriously threatened by irresponsible driving, which is defined by forceful movements, fast speeds, and sudden lane changes. The objective of this project is to create an intelligent system that offers a more dynamic and proactive approach to identifying and reducing possible threats on the road, going beyond conventional monitoring techniques.

To summarize, the motivating force behind the reckless driving prediction system is the necessity of utilising technology to tackle an urgent social issue. In order to develop a system that not only recognizes irresponsible driving behaviors but also anticipates and mitigates future problems, it is necessary to leverage the power of sophisticated computer vision models and proprietary algorithms. This will ultimately contribute to the development of a more secure and safe road infrastructure.

1.2. Major Challenges

Developing a predictive system for reckless driving that incorporates YOLO, DEEPSORT, lane detection models, and a proprietary algorithm is a challenging task that involves various obstacles ranging from technical to data-related, operational, and ethical.

From a technical perspective, careful synchronization is necessary to integrate different computer vision models while taking computational efficiency and real-time processing requirements into account. Finding the right mix between model accuracy and processing speed is essential for quickly detecting and forecasting risky driving behaviors. Developing an adaptable custom algorithm introduces additional complexity, necessitating a finely balanced approach between avoiding false positives and sensitivity, as well as careful consideration of dynamic traffic circumstances. Obtaining and getting ready datasets for training are the main data-related problems. To guarantee that models learn and generalize well, a thorough dataset that represents a variety of risky driving events must be created.

Putting the technology into use in real-world settings, integrating it with the current infrastructure, and handling privacy issues are operational hurdles. Achieving a successful deployment requires careful consideration of scalability, privacy restrictions, and seamless connection with traffic control systems. An additional layer is added by ethical and legal concerns, which necessitate actions to safeguard user consent, preserve individual privacy, and guarantee system openness. The issues of adaptability and maintenance entail continuous efforts to maintain the system's efficacy. Sustained accuracy requires regular updates, retraining of the model, and adaption to changing driving behaviors and technologies.

Long-term success depends on striking a balance between system stability and ongoing improvement.

1.3.Report Overview

This report presents a comprehensive approach utilizing OpenCV for video frame modification, YOLO for vehicle recognition through bounding boxes, and DeepSort for continuous tracking of vehicles across frames. A bespoke algorithm predicts potentially dangerous driving actions based on behavioral patterns and data derived from YOLO and DeepSort. Real-time identification and proactive prediction of reckless driving aim to strengthen traffic safety measures. The dataset used for model training and validation is custom-built using the Carla simulation software, providing a tailored foundation for robust algorithm development.

Chapter 2- Literature Review

The literature review provided offers an extensive and insightful overview of the current landscape in road safety technology and analysis. It delves into several crucial aspects, shedding light on advancements, methodologies, and challenges encountered in enhancing road safety through technological interventions.

2.1.1 Literature Related to Existing Systems

[1] **Vehicle Detection:**

The study under review compares its methodology with noteworthy papers and examines recent developments in UAV-based object recognition and tracking. Notably, DroNet achieved impressive onboard frames-per-second at the expense of accuracy and image resolution by concentrating on real-time vehicle recognition, much like a simplified YOLO network. R3 made improvements by adding bounding boxes that were rotated, and Mask R-CNN made instance segmentation possible. The evaluated work, on the other hand, focuses on high-precision position estimation up to 100 meters, setting itself apart with a reference sensor that has an accuracy of 1 cm. Unlike previous studies, the accuracy of the research presented is excellent, as it addresses standstill mistakes and lighting changes. Superior shape estimation and position accuracy are ensured, outperforming a study that used non-rotated bounding boxes. The method is noteworthy for taking relief displacement into account, which lowers errors as one gets farther away from the central location. This work is noteworthy because it places a high priority on accurate sensor location and positioning and provides open-source code for future development and dataset production. Its unique connection to widely-accepted datasets, such as highD, inD, INTERACTION, and Stanford's dataset, highlights the importance of comparing vehicle positions. This contributes to the field of UAV-based object detection and tracking.

[2]**Vehicle Trajectory Prediction Overview:**

In vehicle prediction, highway trajectory analysis primarily considers the ego-vehicle and its adjacent equivalents. While adjacent cars may have inadequate sensor information, ego-vehicles have exact data. The process of trajectory prediction uses timestamped sequences, which are frequently represented numerically, to estimate future positions. Earlier techniques used models such as Kalman Filter or Neural Networks, utilizing kinematic and dynamic data. Contextual variables were then added, like lane information and vehicle interactions. Different strategies were used to integrate interactions, such as adding Time To Collision (TTC) or modeling the effect of neighboring cars on the ego-vehicle using spatial grids. Some models took into account how the ego-vehicle's future trajectories would affect other nearby vehicles. Prediction techniques varied from focusing on a single car to forecasting multiple vehicles at once. Public and private datasets for trajectory prediction, such as NGSIM, HighD, Argoverse, and Waymo, offer a variety of settings for model building and comprehending vehicle dynamics in various contexts.

[3]**Abnormal Driving Behaviour Detection**

Modern research on car surveillance uses sophisticated algorithms to address a variety of problems. In order to detect abnormal vehicle behavior in traffic camera footage, Wang et al. employ YOLO and Kalman filters. This allows them to effectively identify problems such as halted or speeding automobiles. In order to identify potential traffic dangers based on unfavorable feelings, Kumar et al. use sentiment analysis on Twitter data, enhancing conventional techniques with social media knowledge.Song et al. concentrate on identifying small automobiles on highways with the use of the ORB algorithm and YOLOv3 for accurate identification. Using Kalman and particle filters, Sudha and Priyadarshini improve YOLOv3 for multiple vehicle recognition and tracking. The enhanced YOLOv2 for vehicle detection at different scales by Sang et al. is one example of an object detection advancement. Du et al. emphasize striking a balance between accuracy and speed when optimizing YOLOv3 for real-time vehicle and traffic light identification.Liu et al. compare YOLOv3 and YOLOv5 and suggest 3-D constrained multiple kernels and Kalman filters for enhanced accuracy in vehicle tracking. Neupane et al. create a multi-vehicle tracking system with the goal of increasing accuracy across a range of parameters. They do this by using several YOLO networks for count, categorization, and speed estimation. Through the resolution of particular issues in detection and tracking scenarios, these research together contribute to the evolution of vehicle surveillance.

[4]**Lane Detection**

Three main approaches are used in lane detection: feature-based, model-based, and learning-based techniques. The Feature-Based Approach is more sensitive to changes in illumination but less susceptible to fluctuations in road form because it depends on visual features like edges and gradients. The Model-Based Approach makes use of global road models, which are sensitive to differences in road shape but resilient to changes in illumination. The Learning-Based Approach offers flexibility and adaptability by building a model during training and categorization phases. Inverse perspective mapping and Hough transforms are used in the feature-based approach, which combines image and sensor-based methods for lane detection. It has difficulties in tunnels or with different environmental elements, even though it is accurate in different settings. Although edge extraction, clustering, and geometric model estimation are all included in the Model-Based Approach for robust lane detection, it may have trouble with poorly visible road markers. Real-time adaptation is achieved by integrating deep neural networks and reinforcement learning in the Learning-Based Approach for predictive controller lane detection. It continuously enhances decision-making skills by placing a premium on precision in dynamic situations. On the other hand, a large amount of training data is needed, and differences in road shape could present difficulties.Every strategy has advantages and disadvantages. The feature-based approach suffers with changes in the environment but performs well under certain conditions. Although resilient, the model-based approach has problems with poorly drawn road markers. The learning-based method is quite flexible in real-time, but it needs a lot of training data and could have trouble with changes in road shape. In lane detecting applications, it is important to weigh their advantages and disadvantages fairly.

[5] **Carla Simulation Software**

Constructed from the ground up, CARLA is intended to facilitate the creation, instruction, and verification of autonomous driving systems. Apart from open-source technology and protocols, CARLA offers freely reusable digital assets (buildings, cars, and urban layouts) that were made specifically for this application. The simulation platform allows for the flexible definition of environmental conditions, sensor suites, complete control over both static and dynamic actors, the creation of maps, and much more.

[6]CNN(Convolutional Neural Networks)

Convolutional neural networks are different from other types of artificial neural networks in that they utilize knowledge about a particular type of input rather than concentrating on the problem domain. Consequently, this makes it possible to put up a considerably simpler network design. The fundamental ideas of convolutional neural networks have been presented in this study. Describing the layers needed to construct one and outlining the ideal network architecture for the majority of image analysis jobs. Neural network-based image analysis research has slightly slowed down recently. This is partially caused by the false perception of the degree of expertise and complexity needed to start modeling these incredibly potent machine learning algorithms.

2.1.2 Literature Related to Methodology / Approaches/ Algorithms

Vehicle trajectory prediction emerges as a significant area of study, emphasizing the utilization of historical motion data and integration of various variables, such as kinematics, dynamics, and contextual information. The role of computer vision in accident detection for autonomous vehicles is highlighted, along with challenges in predicting accidents under adverse conditions. Additionally, unique methods such as abnormal vehicle behavior detection using YOLO and sentiment analysis on social media data showcase innovative approaches to supplement traditional detection methods.

2.1.3 Literature Related to Technology / Tools / Frameworks

Recent developments in YOLO (You Only Look Once) models have shown a noteworthy emphasis on striking an ideal balance between speed and precision, especially when it comes to tracking, traffic light detection, and vehicle detection. YOLO, which is renowned for its real-time object detection abilities, has undergone enhancements to address issues unique to driving situations.

A significant advancement is the incorporation of Deep SORT (Simple Online and Realtime Tracking), an advanced tracking algorithm intended to improve the tracking precision of identified items, such as automobiles. The integration of deep learning methodologies with cutting-edge tracking algorithms, known as Deep SORT, substantially enhances the dependability of automobile tracking systems. Deep SORT performs exceptionally well at preserving consistent tracks between frames by utilizing deep embeddings and association measures, which raises tracking precision overall.

Furthermore, another significant improvement in vehicle tracking with YOLO models is the use of Kalman filters. Recursive and effective, Kalman filters help track objects forecast their future states while accounting for faults. The capacity to forecast the future improves the stability of vehicle tracking systems, particularly in situations where objects are obscured or move in an unpredictable manner.

Furthermore, specific tracking techniques developed to address the difficulties presented by automotive situations have surfaced. These techniques frequently consider the special qualities of automobiles, including their unusual sizes, shapes, and motion patterns. In real-time identification scenarios, these techniques help improve accuracy and dependability by customizing tracking algorithms to the unique characteristics of vehicle behavior.

2.2 Observations on Existing Work

The review provides comprehensive insights into lane detection approaches, including feature-based, model-based, and learning-based methodologies. Each approach exhibits specific strengths and limitations, emphasizing the necessity of a nuanced understanding and application in diverse scenarios. The focus on advancing model robustness, real-time adaptability, and machine learning techniques signals a future direction aimed at overcoming challenges posed by varying environmental conditions in lane detection.

Overall, the literature review portrays a comprehensive landscape of research endeavors in road safety technology, highlighting key advancements, methodologies, tools, and challenges. It underlines the collaborative and innovative efforts aimed at ensuring safer and more efficient roadways through technological advancements and thorough testing methodologies.

Chapter 3 -Proposed Methodology

The proposed methodology revolves around the development of a sophisticated driver monitoring system employing a sequence of advanced technologies and a meticulously constructed dataset. Let's delve deeper into the various components and their functionalities within this methodology

3.1.Problem Definition

The inclusion of YOLO models for real-time object identification, including cars and road boundaries, is essential to building an all-encompassing autonomous vehicle tracking system with road boundary detection. More sophisticated tracking algorithms, such as Deep SORT, can improve vehicle tracking accuracy and guarantee ongoing observation. Furthermore, the integration of sophisticated algorithms designed to detect reckless driving behavior—such as those that evaluate relative distances and speeds—assists in detecting veering off course, irresponsible overtaking, hazardous closeness to other vehicles, and instances of brake checks. These technologies blend together seamlessly to create a complex system that can analyze and intervene to guarantee safe and responsible autonomous driving.

3.2. Scope

The autonomous vehicle tracking system with road boundary detection offers a wide and significant range of applications that address many facets of safe and effective driving. Real-time item tracking and detection are included in the project, which enhances traffic control and road safety in general. The system's ability to monitor and react to potentially hazardous driving behaviors is improved by the unique features that detect deviations from intended routes, reckless overtaking, risky proximity to other vehicles, and brake checking.
The scope includes applications in intelligent transportation systems and autonomous driving, where the system's real-time analysis and intervention capabilities facilitate safer navigation. The project tackles significant obstacles in the deployment of autonomous vehicles by incorporating cutting-edge algorithms and sensor technology, guaranteeing compliance with traffic regulations and improving overall road safety. The scope of the system also includes

use cases in fleet management, where risk reduction and operational efficiency can be greatly enhanced by monitoring driver behavior.

Additionally, the initiative supports the increased emphasis on safety requirements in the development of self-driving technologies and adds to the conversation on safe and responsible autonomous mobility. The project's scope puts it at the vanguard of intelligent transportation system innovations as the field of technology advances, opening the door to autonomous driving experiences that are more dependable and safe.

3.2.1.Assumptions and Constraints

The methodology assumes a consistent video feed quality and real-time processing capabilities. The suggested system is limited and functions under certain presumptions. The system is predicated on the idea that the integration of OpenCV, YOLO, and DeepSort enables accurate object detection and tracking in a variety of environmental settings. It also presumes that data transmission between components occurs over a reliable and steady network connection.

Constraints might include challenges in detecting obscured or partially visible vehicles due to environmental conditions or other factors. The system's performance may be limited in unfavorable weather circumstances, such as intense rain or snow, where sensor accuracy may be jeopardized. Additionally, the system is predicated on the assumption that the monitored vehicles follow accepted communication standards and have operational onboard systems. Additionally, in some hardware setups, it could be limited by processing power constraints that affect the system's real-time responsiveness. Last but not least, moral and legal issues might make it impossible to implement the system in different jurisdictions, requiring adherence to local laws and privacy rules regarding the use of surveillance technologies.

3.3. Proposed Approach to build

3.3.1 Features of proposed system

The suggested system offers continuous vehicle tracking, careful examination of spatial and temporal data, and precise prediction of reckless driving occurrences, making it a state-of-the-art method for the real-time identification of irresponsible driving behaviors. The system's design consists of a proprietary algorithm, OpenCV, YOLO (You Only Look Once), and DeepSort (Simple Online and Realtime Tracking) seamlessly integrated. The synergistic combination of these components guarantees extensive driver monitoring capabilities by capitalizing on their own strengths. OpenCV offers a strong foundation for computer vision, but YOLO is best at quickly and precisely identifying objects, especially those related to cars. DeepSort improves tracking accuracy by preserving steady tracks between frames. The unique algorithm further optimizes the system for particular driving behavior analysis, tackling erratically overtaking, braking checking, dangerously close proximity to other cars, and departures from planned routes. All things considered, this system is a smart and practical way to integrate cutting edge technology to encourage safe driving and improve traffic safety.

3.3.2 Proposed System Architecture:

The architecture of the driver monitoring system comprises interconnected modules, each contributing to the system's overall functionality and efficiency. The seamless integration of these modules ensures a cohesive approach to real-time identification and prediction of reckless driving behaviors.

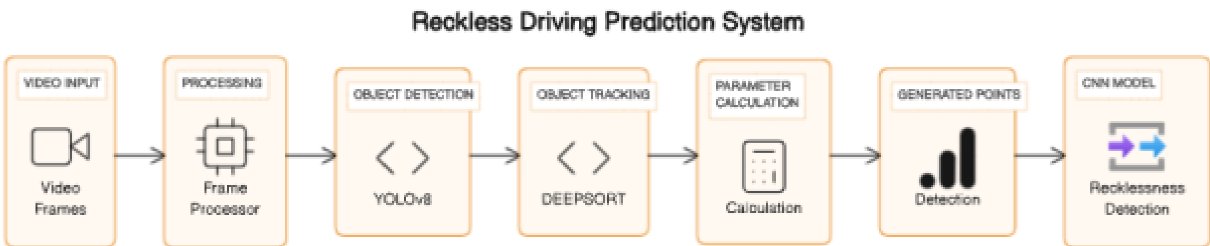

Fig 3.3.2.1 System Architecture

- **Input Module (Video Frames):**

  The system starts by ingesting video frames captured from traffic cameras or vehicle-mounted cameras.

  These frames serve as the input data for subsequent processing and analysis.

- **OpenCV Preprocessing Module:**

  The video frames undergo preprocessing using OpenCV to ensure optimal quality and relevance.

Enhancements include noise reduction, image enhancement, and standardization for consistent analysis.

- **YOLO Object Recognition Module:**

  Once preprocessed, the frames pass through the YOLO (You Only Look Once) algorithm for real-time object recognition.

  YOLO identifies and categorizes vehicles within the frames, providing critical spatial data such as coordinates and dimensions.

- **DeepSort Vehicle Tracking Module:**

  The system employs DeepSort, a tracking algorithm based on deep learning, to maintain stable vehicle tracking across frames.

  DeepSort associates distinct identities with detected vehicles, tracking their motion, velocity, and behavior continuously.

- **Custom Algorithm Analysis Module:**

  A bespoke algorithm analyzes behavior patterns of identified vehicles using data from YOLO and DeepSort.

  This module focuses on detecting specific driving maneuvers indicative of reckless behavior, such as abrupt lane changes or aggressive accelerations.

- **Integration Module:**

  Data from YOLO, DeepSort, and the custom algorithm are integrated and processed collectively to generate predictions related to potential instances of reckless driving.

- **Output Module (Alerts/Notifications):**

  Based on the predictions and analysis, the system generates alerts or notifications in real-time.

  These alerts can be sent to relevant authorities, on-board systems in autonomous vehicles, or traffic management centers for immediate action or further evaluation.

Key Features of the Proposed System

- **Continuous Tracking:** Continuous tracking of vehicles across frames enables thorough behavior analysis.
- **Behavior Pattern Analysis:** The system scrutinizes behavior patterns to detect and predict episodes of irresponsible driving.
- **Integration of Technologies:** The integration of OpenCV, YOLO, DeepSort, and a custom algorithm ensures a comprehensive and cohesive approach to driver monitoring.

This proposed system architecture combines various state-of-the-art technologies and algorithms to create an efficient and reliable driver monitoring solution, fostering enhanced road safety and effective enforcement of traffic regulations.

3.4 Benefits of Proposed Solution

The envisioned system promises heightened accuracy in identifying reckless driving, enabling real-time detection and behavior analysis to enforce road safety measures effectively. By promoting safer driving practices, it aims to contribute significantly to enhancing road safety and upholding traffic law adherence.When YOLO, DeepSort, and the proprietary algorithm are combined, a reliable solution for extensive driver monitoring is produced. Through the integration of object identification, tracking, and behavior analysis, the system is able to forecast instances of reckless driving in real-time in addition to identifying poor driving habits. The system's accuracy and efficacy in promoting road safety and enforcing traffic laws are improved by this methodical approach.

1. **Enhanced Security**:
Instantaneous Threat Identification Rapid detection of potentially dangerous driving behaviors by the system, such as sudden lane changes and high-speed maneuvers, enables prompt intervention or notification of appropriate authorities.
2. **Precise Monitoring and Evaluation**:
Object Tracking with DEEPSORT: DEEPSORT makes it possible to follow cars precisely over time, giving useful information for examining traffic patterns, spotting bottlenecks, and comprehending driver behavior.
3. **Improved Traffic Control:**
Data-Driven Decision Making: By providing traffic management authorities with up-to-date information on traffic flow, speeds, and lane changes, the system helps them make well-informed decisions about how best to arrange traffic signals and routes.
4. **Accident Prevention**
Lane Change Prediction: By anticipating potentially dangerous lane-changing behaviors and generating alarms or interventions, the unique algorithm for lane change prediction can help prevent accidents.

5. Visualizations and Reports:

The system's user interface can offer reports and visualizations, which, particularly combined with campaigns and education, can help raise public awareness of traffic patterns and promote safer driving practices.

6. Assistance for Law Enforcement:

The technology can help law enforcement by giving time-stamped, location-based proof of events involving reckless driving, which can speed up court cases.

Chapter 4 - Project Management

4.1. Project Schedule

Fig 4.1.1 Timeline Chart

4.2Feasibility Study

4.2.1Technical Feasibility

The availability of well-established computer vision models like YOLO and DEEPSORT lends technical promise to the suggested reckless driving prediction system. These models work well for tracking and detecting objects , which is crucial for quickly identifying instances of irresponsible driving. To guarantee flawless compatibility and peak performance, the integration of these models with a unique lane change prediction algorithm might need to be implemented carefully. Furthermore, even though YOLO and DEEPSORT provide solid bases, much testing and fine-tuning will be required to create a dependable and accurate custom algorithm for lane shift prediction.

4.2.2Operational Feasibility

Operationally, it is possible to integrate the system with current traffic control systems and provide an easy user interface. Weighing the possible advantages of improving monitoring capabilities through integration against potential compatibility and communication protocol issues is important. The operational success of the system will be influenced by user acceptance and convenience of use factors made during the design phase. It is also technically possible to construct an alert system based on predetermined criteria; however, reducing false positives and negatives is essential to preserving the credibility of the system.

4.2.3Economical Feasibility

The project's economic viability takes into account both the project's start-up and ongoing costs. Open-source versions of YOLO and DEEPSORT save money on software, but the hardware needed for real-time processing can add to the initial cost. Purchasing and preparing unique datasets to train the models may result in extra costs. It is economically reasonable to do routine maintenance, but if significant improvements or regular updates are required, prices may rise. Once implemented, operational costs can be effectively managed; nevertheless, long-term budgeting should take false alerts and system failures into account.

To sum up, the suggested approach for predicting reckless driving shows potential in terms of technology, operational efficiency, and financial sustainability. To create a reliable and efficient system for detecting and forecasting risky driving behaviors, successful deployment will necessitate careful consideration of technical integration, user interface design, and economic issues.

4.3Project Resources

4.3.1.Hardware Requirements

In order to provide real-time object identification and tracking, the reckless driving prediction system's hardware needs include a powerful processing unit with a strong CPU/GPU combination. A GPU (especially one from the NVIDIA series) should be added for faster calculations. To meet the simultaneous processing demands of many frames and models, a minimum of 16GB of RAM is required. SSD storage is also advised in order to speed up data retrieval and processing of large video streams. To capture clear and detailed input data, the system needs high-resolution cameras or sensors. Installing and configuring these components is crucial to getting real-time video feeds from the monitored region.

4.3.2.Software Requirements

Software-wise, the operating system needs to work with both Windows- and Linux/Unix-based computers, with the latter being favored due to its aptitude for deep learning applications. Deep learning frameworks require certain drivers and libraries, which must be supported by the operating system of choice. Models like YOLO and DEEPSORT require the installation of deep learning frameworks like TensorFlow or PyTorch, in addition to certain dependencies and libraries for model training and inference. In addition, the integration of lane detection models requires the use of appropriate libraries or frameworks, like OpenCV. Code development and experimentation also require development tools, such as deep learning-specific ones or integrated development environments like Jupyter Notebooks.

4.3.3.Operating Requirements

In terms of operational needs, for efficient tracking and detection, the system must guarantee continuous real-time processing with low latency. In order to keep monitoring from being interrupted, a dependable power source and backup systems are essential. Data transmission requires network access, particularly when remote monitoring or data storage are required. For systems to operate as efficiently and accurately as possible, routine maintenance is essential. This includes hardware inspections, software upgrades, and model retraining. To avoid hardware overheating and ensure reliable operation, environmental factors like ventilation and temperature management should be taken into mind. It is crucial that the many hardware, software, and operating requirements be fulfilled for the reckless driving prediction system to be deployed successfully and continue to work over time.

Chapter 5 - System Design

5.1Design Diagrams

5.1.1Data Flow Diagram

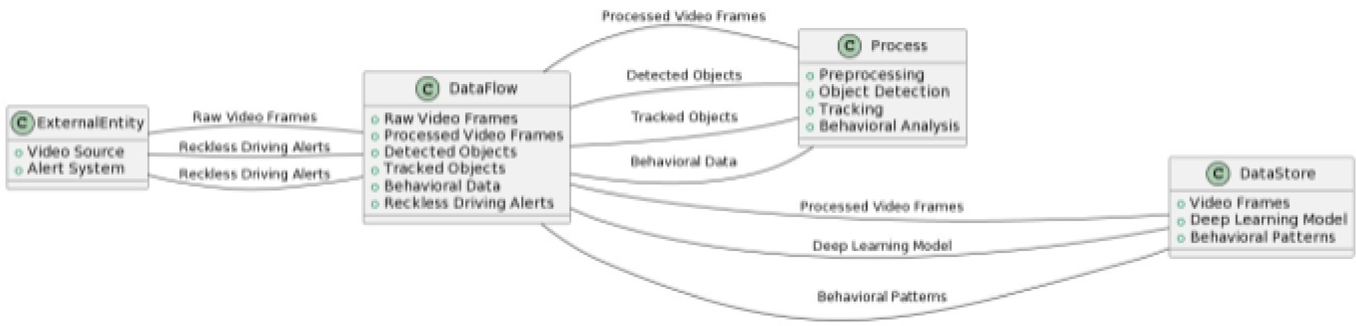

Fig 5.1.1 DFD Diagram

5.1.2UML Diagrams - Appropriate UML Diagrams for the proposed system

Sequence Diagram:

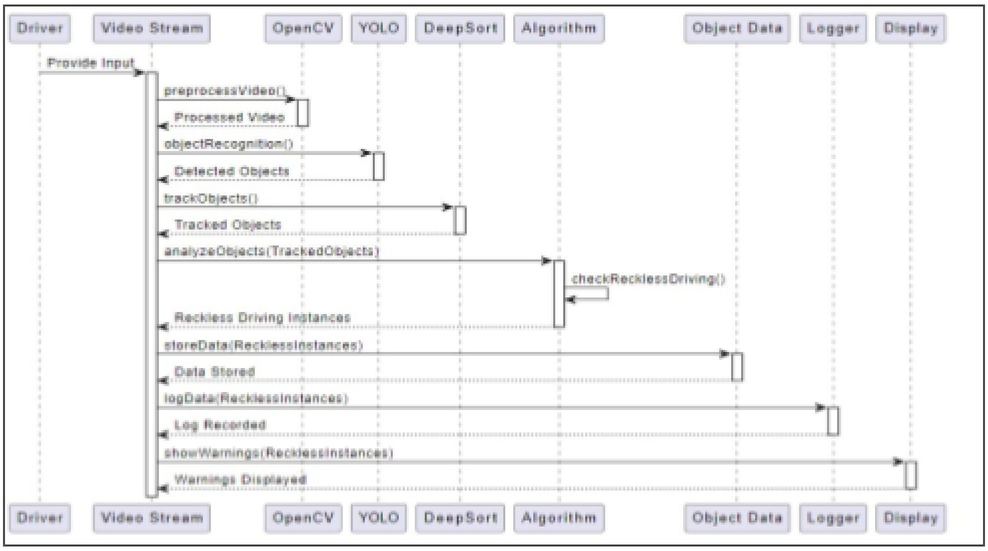

Fig 5.1.2 Sequence Diagram

Class Diagram:

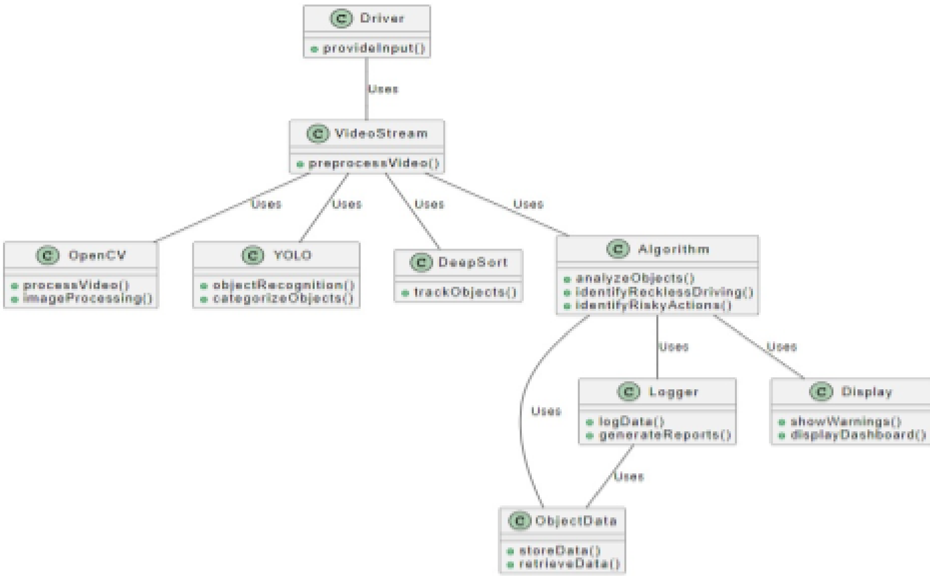

Fig 5.1.3 Class Diagram

Activity Diagram:

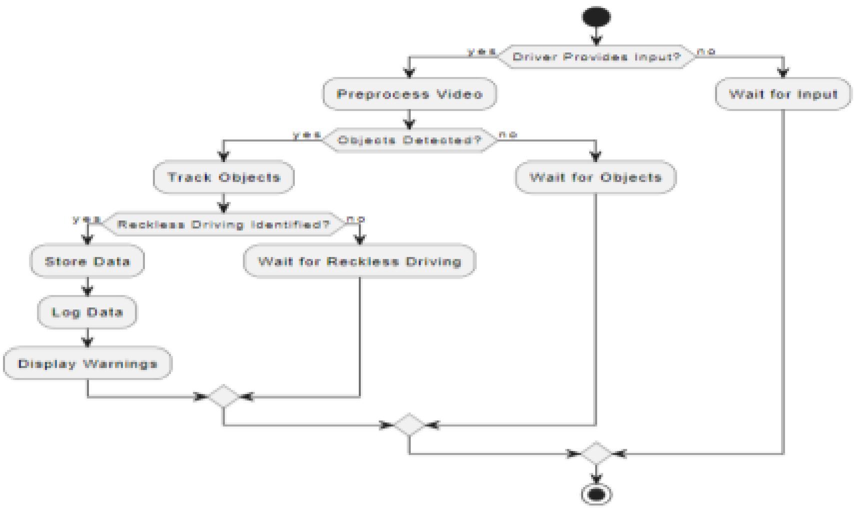

Fig 5.1.4 Activity Diagram

Chapter 6 - Implementation

6.1. Tools used for data collection, size of the sample and limitations

For the purpose of doing research on autonomous driving, the open-source Carla simulation program offers a high-fidelity and realistic simulation environment. Carla gives researchers access to comprehensive 3D models of cars, people, and urban landscapes. This allows researchers to test and validate different algorithms under dynamic weather situations like rain, fog, and different times of day. Carla's ability to simulate sensors like radars, lidars, and cameras makes it easier to create and test perception algorithms, which are essential for driverless cars. Because of its open-source foundation, users can add new features and engage with the community while also encouraging modularity and customisation. Through a Python API, researchers can design the behavior of autonomous agents, allowing the development of unique situations to mimic certain difficulties faced by self-driving automobiles. Carla is an important tool for growing autonomous driving technology because of its integration with reinforcement learning, multi-agent simulation capabilities, and features like data recording and analysis. Carla offers an extensive documentation system, an engaged community, and a useful platform for field research and development collaboration.

6.2 Description of Datasets

The dataset used for training and testing the system is created by us using the Carla simulation software. The Vehicle Tracking Dataset is an extensive series of almost two thousand high-quality photos that were painstakingly created using the Carla simulator program. Each photograph, which was taken from four different perspectives, offers a different viewpoint on the spatial interactions between the cars. The implementation of a multi-angle strategy guarantees a resilient dataset that emulates the intricacy of actual traffic situations, hence facilitating the advancement and assessment of vehicle tracking algorithms that exhibit heightened precision and dependability. Vehicle tracking models can be trained and tested in a variety of scenarios using these photos, from the bright light of a sunny day to the difficulties presented by low light or bad weather. This helps to ensure that the algorithms are resilient and flexible.

This dataset can be utilized by scholars and professionals working in the domains of autonomous systems and computer vision to promote the creation of intelligent transportation systems. The dataset is well-suited for training machine learning models that can precisely monitor cars in a variety of real-world circumstances because it includes a variety of angles and lighting conditions. This will ultimately improve the efficiency and safety of future transportation systems.

Fig 6.2.1 Carla Normal view

Fig 6.2.2 Carla Top View

Fig 6.2.3 Carla Back View

6.3. Preprocessing

1. Frame Extraction and Standardization: From the video data that was acquired from the Carla simulation, extract individual frames. Resize each frame to a uniform resolution during training to guarantee efficiency and consistency. This reduces the possibility of information loss during further investigation in addition to streamlining computational procedures.

2. Manually annotate the frames by drawing bounding boxes around the cars as part of the object annotation for ground truth. Ground truth data is established by this method, which is essential for the object detection model's training. Precise annotations give the model points of reference from which to understand the spatial properties of the cars in the frames.
3. Data Augmentation Techniques: To artificially diversity the training dataset, use data augmentation techniques. Methods like flips, random rotations, brightness variations, and small translations increase the variety of the dataset. Through data augmentation, the model performs better on unknown data and is able to generalize better across many contexts.
4. Handling Unbalanced Data: Make sure that the dataset accurately represents both positive (vehicle) and negative (non-car) samples. Unbalanced data might cause model predictions to be skewed. In order to rectify this imbalance and guarantee that the model is trained on an equitable proportion of both classes, apply strategies like oversampling or undersampling.
5. Partitioning and Coding the Dataset for Model Input: Divide the dataset into sets for testing, validation, and training. Testing data assesses the model's performance on unobserved data, validation data helps with hyperparameter tuning, and training data is used to train the model. To prepare the data for input into the deep learning model, use one-hot encoding to encode categorical information, such as class labels (vehicle, non-car).

Through careful implementation of these procedures, you create a ready-made dataset that facilitates training and helps identify automobiles with bounding boxes in Carla simulated video frames. By addressing distinct aspects of data quality and model generalization, each stage enhances the deep learning model's overall robustness for car identification.

6.4 Description of algorithm to be used :

A complex combination of YOLO (You Only Look Once), DeepSort (Simple Online and Realtime Tracking), and a unique Lane Change Detection Algorithm is used in the bespoke algorithm to identify reckless driving. The method starts by identifying automobiles in the area of vision by utilizing YOLO's quick and precise object identification capabilities. Then, DeepSort is used to guarantee accurate tracking of these cars over time. The program includes a special Lane Change Detection program to evaluate hazardous driving. This specialized part tracks the motion of cars in space and examines lane position variations. A threshold for comprehensible variations in velocity is set such that only notable deviances warrant additional examination. The computer marks an occurrence as possibly being reckless driving when it notices a significant shift in speed together with a lane change.

To prevent false positives, the Lane Change Detection Algorithm takes into account contextual data including the kind of road and traffic patterns. With the YOLO and DeepSort components offering real-time object detection and tracking and the Lane Change Detection Algorithm providing a nuanced understanding of driving actions, the integration of these components ensures a robust and context-aware system for identifying reckless driving behaviors. By improving the system's capacity to identify instances of irresponsible driving, this algorithmic synergy encourages drivers to drive more sensibly and safely.

Chapter 7 - Testing and Results

7.1 Test Plan

1. Objective:

- To verify the functionality, performance, usability, and security of the Advanced Driver Monitoring System.

2. Scope:

- The test plan covers the following aspects:

  - Video preprocessing using OpenCV.

  - Object detection using YOLO.

  - Object tracking using DeepSort.

  - Identification of unsafe driving behaviors using a self-devised algorithm.

3. **Test Environment**:

- Hardware:

  - Computer system with sufficient processing power (minimum requirements: quad-core CPU, 8 GB RAM).

  - Webcam or video input device.

- Software:

  - Operating system: Windows 10.

  - Development environment: Python 3.8.

  - Libraries: OpenCV, YOLO, DeepSort.

- **Test Data:**

  - Collection of video clips depicting various driving scenarios (normal and unsafe behaviors).

  - Sample input videos with different resolutions and frame rates

4. Test Cases:

a. Video Preprocessing:

- Verify that OpenCV preprocesses input video streams correctly.

- Check if pertinent characteristics are identified and image quality is improved.

b. Object Detection with YOLO:

- Confirm YOLO accurately detects cars and their actions in real-time.

- Check for correct categorization of different items in each frame.

c. Object Tracking with DeepSort:

- Validate DeepSort's ability to track objects between frames accurately.

- Ensure consistent and precise vehicle tracking across time.

d. Unsafe Driving Behavior Identification:

- Test the self-devised algorithm's capability to identify risky driving actions.

- Verify accurate identification of factors such as abrupt accelerations, aggressive maneuvers, and unpredictable lane changes.

5. Performance Testing:

- Measure the system's processing speed for different video resolutions and frame rates.

- Monitor CPU and memory usage during system operation.

- Ensure real-time performance and resource consumption within acceptable limits.

6. Integration Testing:

- Verify proper integration of OpenCV, YOLO, and DeepSort.

- Test compatibility and functionality across different versions of libraries.

- Ensure seamless communication between modules without data loss.

7. User Acceptance Testing (UAT):

- Collaborate with end-users to validate the system's effectiveness and usability.

- Gather feedback on system performance and overall satisfaction.

8.. Scalability Testing:

- Assess the system's ability to handle increased workload or data volume.

- Test scalability across different hardware configurations.

9. Post-Deployment Monitoring:

- Implement monitoring mechanisms for real-world usage.

- Gather user feedback and address issues or improvement opportunities.

7.2 Test Cases

Functionality Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
FT01	Verify that the reckless driving detection model correctly identifies instances of reckless driving.	1. Input video footage of a vehicle exhibiting reckless driving behaviour. 2. Process the video footage through the detection model.	Video footage showing reckless driving behaviour.	The model correctly identifies instances of reckless driving.	The model correctly identifies instances of reckless driving.	Pass
FT02	Verify that the reckless driving detection model correctly identifies instances of safe driving.	1. Input video footage of a vehicle exhibiting safe driving behaviour. 2. Process the video footage through the detection model.	Video footage showing safe driving behaviour.	The model correctly identifies instances of safe driving.	The model correctly identifies instances of safe driving.	Pass

User Interface Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
UI01	Ensure the user interface displays relevant information about detected reckless driving incidents.	1. Launch the reckless driving detection application. 2. Observe the user interface elements for clarity and relevance. 3. Trigger a detection event and check for real-time updates on the interface.	Video footage showing reckless driving behaviour.	The user interface displays relevant information about detected reckless driving incidents in a clear and understandable manner.	The user interface displays relevant information about detected reckless driving incidents.	Pass

Performance Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
PT01	Assess the processing speed and efficiency of the reckless driving detection model.	1. Input various lengths of video footage containing reckless driving instances. 2. Measure the	Video footage of varying lengths with reckless driving instances.	The model processes video footage within acceptable time limits, maintaining a high FPS rate.	The model processes video footage within acceptable time limits, maintaining a high FPS rate.	Pass

		time taken by the model to process each video. 3. Record the performance metrics such as frames per second (FPS) or processing time per frame.				

Integration Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
IT01	Ensure the reckless driving detection model integrates seamlessly with other components of the application.	1. Integrate the detection model with the video input module. 2. Verify that the model accurately processes video streams from different sources. 3. Test the integration with alert/notification systems to ensure timely response to detected incidents.	-	The detection model integrates seamlessly with other components, and the overall system functions cohesively.	The detection model integrates seamlessly with other components.	Pass

Usability Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
UT01	Evaluate the ease of use and user-friendliness of the reckless driving detection application.	1. Conduct a survey with potential users to gather feedback on the application's usability. 2. Ask users to perform common tasks such as initiating a detection process and reviewing detected incidents. 3. Analyze user feedback and observations to identify any usability issues. Test Data: Survey responses, user feedback.	Survey responses, user feedback	The application is intuitive and easy to use, with minimal learning curve for users.	The application is intuitive and easy to use, with positive feedback from users.	Pass

Database Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
DT01	Verify the storage and retrieval of reckless driving incident data from the database.	1. Trigger a detection event and ensure that relevant data is stored in the database. 2. Retrieve stored data from the database and compare it with the original detection results. 3. Test database performance with a large volume of data to ensure scalability.	Reckless driving incident data.	The database accurately stores and retrieves reckless driving incident data without loss or corruption.	The database accurately stores and retrieves reckless driving incident data.	Pass

Security Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
ST01	Assess the security measures implemented in the reckless driving	1. Conduct penetration testing to identify vulnerabilities in the system.	Penetration testing results, security logs.	The system withstands security attacks and ensures the confidentiality, integrity, and	The system withstands security attacks and ensures the confidentiality, integrity, and	Pass

	detection system.	2. Verify that user authentication and authorization mechanisms are robust. 3. Test data encryption during transmission and storage to prevent unauthorized access.		availability of data.	availability of data.	

User Acceptance Test Cases:

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
UA01	Validate the reckless driving detection system against user requirements and expectations.	1. Present the system to a group of end-users or stakeholders. 2. Have users interact with the system and provide feedback on its functionality and performance. 3. Ensure that the system meets the specified requirements and addresses user needs effectively.	User feedback, requirements documentation.	Users are satisfied with the system's performance and believe it meets their needs.	Users are satisfied with the system's performance and believe it meets their needs.	Pass

7.3Testing methods

Unit testing is disassembling the program into its smallest testable components, or units, and testing each one separately to make sure it functions as intended. These tests, which are usually automated, check the behavior of particular classes, methods, or functions.

Validation of Input/Output: Determined whether the model generates the desired result for the supplied input data. This entails providing predetermined input samples to the model and determining if the resultant output aligns with the predicted outcomes.

Testing Accuracy: We assessed the accuracy of the model on the test dataset after fine-tuning it based on validation performance. This offers a fair assessment of the model's effectiveness using hypothetical data.

Boundary testing: examined how the model behaved when input range boundaries were reached.
Testing the model's ability to handle errors, such as erroneous input data or unexpected exceptions, is known as error handling testing. This guarantees that failures are handled by the model gently, preventing crashes and erroneous output.

Testing the website's ability to accurately check user inputs prior to sending them to the CNN model was done. By doing this, the model is guaranteed to process only legitimate input data.

Mocking is the process of simulating the behavior of external dependencies, such as the CNN model, by using mock objects or functions. As a result, we were able to test the functionality of the website independently of the CNN model.
Performance testing: Evaluate how well the website functions when it is integrated with the CNN model, taking into account variables like resource consumption, throughput, and response time. This guarantees that the website will continue to be scalable and responsive across a range of loads and usage circumstances.

Integration Examination :

Verifying the interactions and data flow between the frontend, backend, and machine learning model of a website is a crucial step in integration testing for the integration of a CNN model. This is how the integration testing was carried out:
Input Validation: Made sure that user inputs are correctly validated by the website's frontend before being sent to the backend for processing. To stop incorrect or malicious inputs from getting to the CNN model, this involves verifying the input format, data type, and range validation.

Data Transmission: Verify that user input data is correctly sent to the backend for processing by the CNN model by testing the communication between the frontend and backend of the website. This required confirming the accuracy and completeness of the data that was sent back and forth between the two components.
Model Inference: Verify that the backend and CNN model are integrated properly to make sure that input data is processed appropriately and that predictions are produced as

anticipated. Confirmed that the predictions made by the model are sent back to the backend in a format that the website can easily use.

Error Handling: The website's ability to manage errors, such as broken communications, erroneous model replies, or unanticipated exceptions that arise during the integration process was tested. Made sure the system gracefully resolves problems without crashing and that users see the relevant error messages.
Performance Testing: Evaluated the performance of the integrated system under different loads and usage scenarios. Measured factors such as response time, throughput, and resource utilization to ensure that the website remains responsive and scalable when integrated with the CNN model.

7.4 Experimental Results

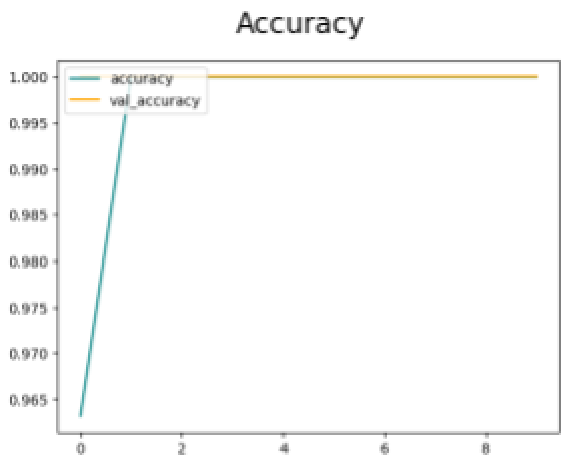

Fig 7.4.1 Accuracy

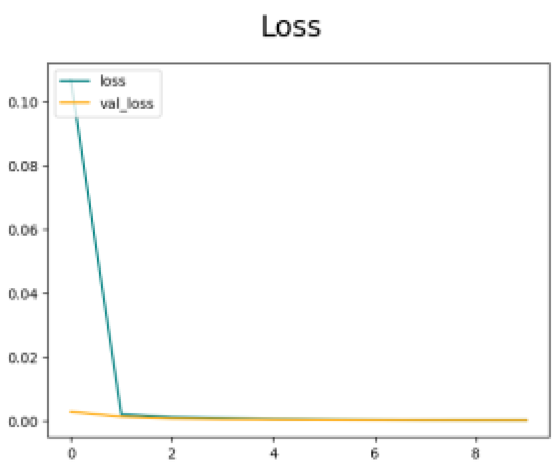

Fig 7.4.2 Loss

In the two following graphs the line represents the path followed by the vehicle.

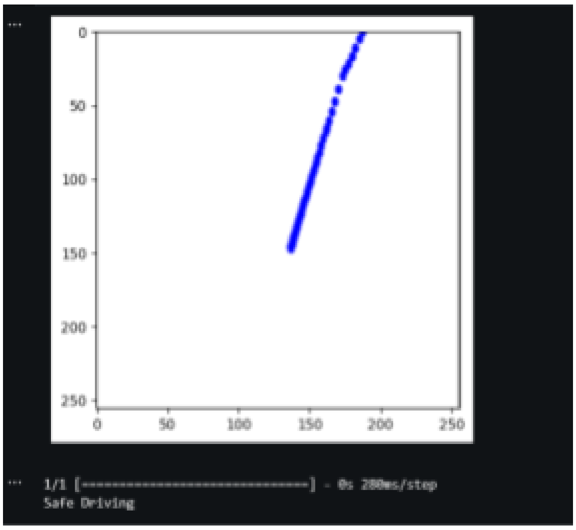

Fig 7.4.3 Safe Driving

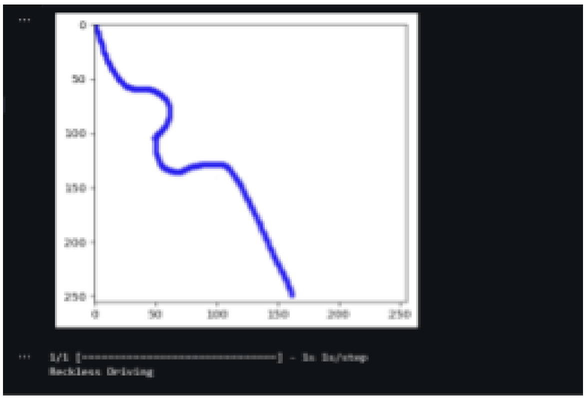

Fig 7.4.4 Reckless driving

In the two following graphs middle line represents the path followed by the vehicle and the side line represents the lane in which the vehicle is moving across.

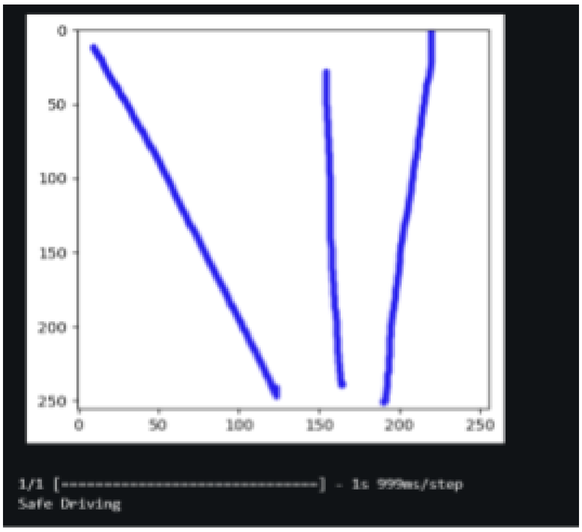

Fig 7.4.5 Vehicle driving safely wrt lanes

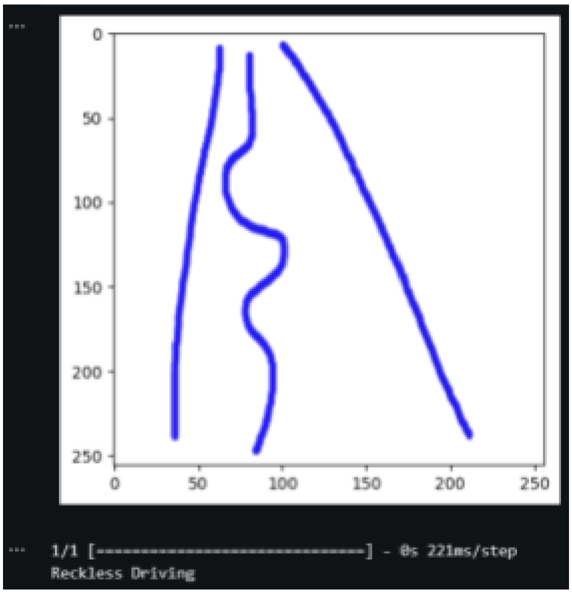

Fig 7.4.6 Vehicle Driving recklessly

CHAPTER 8 - CONCLUSION

To sum up, the suggested method for predicting reckless driving demonstrates a strong integration of cutting-edge computer vision algorithms, including YOLOv8, YOLO, and DEEPSORT, to evaluate driver behavior by analyzing video frames. With the ability to precisely identify cars, follow their paths, compute speeds, and identify lane changes, the system offers a thorough grasp of the dynamics on the road. The combination of object detection and tracking algorithms enables accurate data extraction, which in turn allows for the analysis of driving behaviors. An important layer of functionality is added to the system by incorporating a specific model for recklessness determination. This all-encompassing method not only improves the predictive accuracy of reckless driving but also lays the groundwork for future developments in driver safety and intelligent transportation systems. The system's performance, which has been confirmed by comprehensive testing, highlights its potential as an effective tool for raising driving standards and improving road safety.

PUBLICATIONS

Paper accepted at Springer International Conference on Intelligent Computing and Big Data Analytics, 2024, (ICICBDA2024)

Proof :

Acceptance Notification - Springer ICICBDA 2024 Inbox ×

Microsoft CMT <email@msr-cmt.org>
to me

Tue, 14 May, 10:04 (14 hours ago)

Dear Varun Baliram Bhosale,
Paper Id: 68
Paper Title: Enhancing Road Safety: Reckless Driver Detection via OpenCV in Simulated Environments

CONGRATULATIONS!

We are delighted to convey that your paper has been accepted for presentation at the International Conference on Intelligent Computing and Big Data Analytics, 2024, (ICICBDA2024), 15-16 June 2024.

The proceedings of the International Conference on Intelligent Computing and Big Data Analytics, 2024, (ICICBDA2024) is being published by Springer's Communications in Computer and Information Science series (CCIS). This series is indexed in SCOPUS. ICICBDA 2024 received the confirmed approval for the publication of proceedings in the CCIS series. All registered and presented papers only will be published in the conference proceedings.

The review process of ICICBDA 2024 was double-blind. We have received the review comments from the reviewers and they are available at CMT. You need to address each review comment and prepare a Camera Ready Copy.

References

1. A Real-Time Computer Vision Based Approach to Detection and Classification of Traffic Incidents

2023

https://www.mdpi.com/2504-2289/7/1/22

2. Traffic Accident Detection Method Using Trajectory Tracking and Influence Maps

2023

https://www.mdpi.com/2227-7390/11/7/1743

3. Advances and applications of computer vision techniques in vehicle trajectory generation and surrogate traffic safety indicators

2023

https://www.sciencedirect.com/science/article/abs/pii/S0001457523002385

4. 3D-Net: Monocular 3D object recognition for traffic monitoring

2023

https://www.sciencedirect.com/science/article/pii/S0957417423007558

5. Vehicle trajectory prediction on highways using bird eye view representations and deep learning

https://ieeexplore.ieee.org/document/9342226

6. Innovative Research of Trajectory Prediction Algorithm Based on Deep Learning in Car Network Collision Detection and Early Warning System

2021

https://www.hindawi.com/journals/misy/2021/3773688/

7. Vehicle Position Estimation with Aerial Imagery from Unmanned Aerial Vehicles

https://www.researchgate.net/publication/348368287_Vehicle_Position_Estimation_with_Aerial_Imagery_from_Unmanned_Aerial_Vehicles

8. Vehicle Trajectory Prediction and Cut-In Collision Warning Model in a Connected Vehicle Environment

2020

https://ieeexplore.ieee.org/document/9186817

9.CARLA+: An Evolutionof the CARLA Simulator for ComplexEnvironment Using a ProbabilisticGraphical Model

2023

PAPER NAME

final black book (1).pdf

WORD COUNT

8678 Words

CHARACTER COUNT

53594 Characters

PAGE COUNT

50 Pages

FILE SIZE

1.6MB

SUBMISSION DATE

May 16, 2024 3:48 PM GMT+5:30

REPORT DATE

May 16, 2024 3:49 PM GMT+5:30

3% Overall Similarity

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

- 1% Internet database
- 0% Publications database
- Crossref database
- Crossref Posted Content database
- 2% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 8 words)
- Manually excluded text blocks

3% Overall Similarity

Top sources found in the following databases:

- 1% Internet database
- 0% Publications database
- Crossref database
- Crossref Posted Content database
- 2% Submitted Works database

TOP SOURCES

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

1	University of Hertfordshire on 2023-05-01 Submitted works	<1%
2	National College of Ireland on 2022-08-12 Submitted works	<1%
3	University of Pretoria on 2020-11-02 Submitted works	<1%
4	Leeds Beckett University on 2021-06-18 Submitted works	<1%
5	"Artificial Intelligence and Smart Vehicles", Springer Science and Busin... Crossref	<1%
6	University of Bedfordshire on 2020-09-28 Submitted works	<1%
7	University of Alabama at Birmingham on 2024-02-15 Submitted works	<1%
8	usermanual.wiki Internet	<1%

9	coursehero.com Internet	<1%
10	NCG on 2024-03-08 Submitted works	<1%
11	University of Ghana on 2023-07-28 Submitted works	<1%
12	University of Westminster on 2024-04-10 Submitted works	<1%
13	srmap on 2024-05-12 Submitted works	<1%
14	tnsroindia.org.in Internet	<1%
15	mdpi.com Internet	<1%

Excluded from Similarity Report

- Bibliographic material
- Cited material
- Manually excluded text blocks
- Quoted material
- Small Matches (Less then 8 words)

EXCLUDED TEXT BLOCKS

LEARNINGSubmitted in partial fulfillment of the requirementof the degree ofBache...

www.coursehero.com

Signature of Internal ExaminerSignature of External Examiner

www.coursehero.com

iiiDECLARATIONWe declare that this written submission represents our ideas in o...

svkm on 2024-05-10

AcknowledgementWe would like to

ir.aiktclibrary.org:8080

We would like to

svkm on 2024-05-10

for their continuous

mctrajiv on 2024-04-21

Departmentof Information Technology Engineering, Dwarkadas J. Sanghvi College...

link.springer.com

viTable of ContentsAcknowledgement ………………………………………………………… vAbs...

dspace.iiuc.ac.bd:8080

1. Introduction ……………………………………………………………… 11.1. Motivation / Objec...

www.coursehero.com

4. Project Management
Coventry University on 2013-01-11

4.2. Feasibility Study ……………………………………………………… 154.2.1. Technical Feas...
www.coursehero.com

5. System Design
www.slideshare.net

257.1. Test Plan ………………………………………………………………257.2. Test
Macquarie University on 2024-04-04

Figure 5.1.1
Rochester Institute of Technology on 2005-04-14

List of FiguresFigure 1.1
lib.buet.ac.bd:8080

Table 7.2.2
University of Central England in Birmingham on 2017-05-05

Test Case
University of Hull on 2022-09-15

2.1.1 Literature Related to Existing Systems
www.coursehero.com

2.1.2 Literature Related to Methodology / Approaches
www.coursehero.com

3.1 Features of proposed systemThe
The Hong Kong Polytechnic University on 2006-04-01

YOLO (You Only Look Once
National College of Ireland on 2022-08-12

4 - Project Management4.1. Project Schedule4

wiki.smu.edu.sg

4.3.1.Hardware RequirementsIn order to

London School of Commerce on 2012-11-23

4.3.2.Software

Atsuko Kaga, Kimio Nakahama, Shiro Hamada, Syuichi Yamaguchi, Tsuyoshi Sasada. "Collaborative design ...

the dataset into sets for testing,validation, and training

srmap on 2024-05-12

data is used to train the model

Muhammad Waseem Sabir, Muhammad Farhan, Nabil Sharaf Almalki, Mrim M. Alnfiai, Gabriel Avelino Sam...

Chapter 7 - Testing and Results7.1 Test

University of Southampton on 2009-05-10

minimum requirements: quad-coreCPU, 8 GB RAM

www.novell.com

Operating system: Windows 10.- Development environment: Python 3

assets.researchsquare.com

User Acceptance Testing (UAT):- Collaborate with end-users to validate the system

Southern New Hampshire University - Continuing Education on 2023-10-09

Scalability Testing:- Assess the system's ability to handle increased workload

Bradford College, West Yorkshire on 2023-07-05

Test Cases:Test CaseTestScenario

www.coursehero.com

Test Cases:TestCaseIDTestScenarioTest StepsTest DataExpectedResultsActualR...

www.coursehero.com

Test Cases:TestCase IDTestScenarioTest StepsTest DataExpectedResultsActualR...

www.coursehero.com

Test Cases:TestCaseIDTestScenarioTest StepsTestDataExpectedResultsActualRe...

www.coursehero.com

modelintegratesseamlesslywith othercomponentsof

University of Illinois at Urbana-Champaign on 2024-05-01

Test Cases:TestCase IDTestScenarioTest StepsTest DataExpectedResultsActualR...

www.coursehero.com

theease of useanduser

University of Westminster on 2024-04-10

Test Cases:Test CaseTestScenario

www.coursehero.com

toensure

University of Wales Institute, Cardiff on 2024-05-02

Test Cases:TestCaseIDTestScenarioTest StepsTest DataExpectedResultsActualR...

www.coursehero.com

Test Cases:TestCaseIDTestScenarioTest StepsTest DataExpectedResultsActualR...

www.coursehero.com

Usersaresatisfiedwiththesystem'sperformanceand

University of Greenwich on 2010-06-02

International Conference on Intelligent Computing and Big DataAnalytics, 2024

1login.easychair.org