

V-CAS: A Realtime Vehicle Collision Avoidance System Using Deep Learning on Multiple Camera Streams



By

Muhammad Waqas Ashraf

(Registration No.: 00000431965)

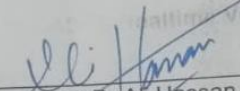
Supervisor

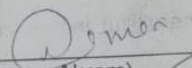
Dr. Ali Hassan

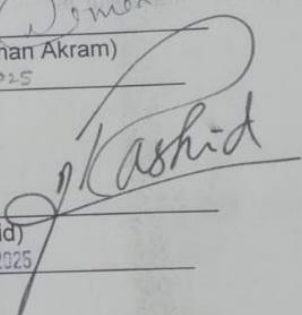
DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING
COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY
ISLAMABAD, PAKISTAN

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis entitled "V-CAS: A Realtime Vehicle Collision Avoidance System Using Deep Learning on Multiple Camera Streams" written by Muhammad Waqas Ashraf Registration No. 00000431965, of College of E&ME has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature : 
Name of Supervisor: Dr Ali Hassan
Date: 21-01-2025

Signature of HoD: 
(Dr Muhammad Usman Akram)
Date: 21-01-2025

Signature of Dean: 
(Brig Dr Nasir Rashid)
Date: 21 JAN 2025

V-CAS: A Realtime Vehicle Collision Avoidance System Using Deep Learning on Multiple Camera Streams

By

Muhammad Waqas Ashraf

(Registration No.: 00000431965)

A thesis submitted to the National University of Sciences and Technology,

Islamabad

in partial fulfillment of the requirements for the degree of

Master of Sciences in Computer Engineering

Supervisor

Dr. Ali Hassan

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING,

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING,

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD, PAKISTAN

January, 2025

Dedication

Dedicated to my exceptional parents and adored wife whose tremendous support and cooperation led me to this wonderful accomplishment.

Acknowledgements

I am thankful to my Creator Allah Almighty for guiding me throughout this work at every step. Indeed, I could have done nothing without Your priceless help and guidance. I am empowered to Read and Write only by Him, who has bestowed upon me the knowledge I carry forward.

I would also like to express special thanks to my supervisor Dr. Ali Hassan for his help throughout my thesis and knowledge imparted through the courses he taught me. I would also like to pay special thanks to Dr. Muhammad Yasin and Dr. Muhammad Salman for their tremendous support and cooperation.

I sincerely express my solemn gratitude with earnest sense of reverence to my parents for their encouragement, heartfelt prayers and kind wishes for successful completion of my studies along with this research work. Specially, I want to thank my spouse and children from all my heart for always supporting and motivating me at every point of my research journey.

Finally, I would like to express my appreciation to all the individuals who have rendered valuable assistance to my study.

Abstract

This research introduces a robust real-time Vehicle Collision Avoidance System (V-CAS) aimed at enhancing vehicle safety through environmental perception-based adaptive braking. V-CAS utilizes the advanced vision-based transformer model RT-DETR, DeepSORT tracking, speed estimation, brake light detection, and an adaptive braking mechanism. It computes a composite collision risk score from vehicles' relative accelerations, distances, and detected braking actions, leveraging brake light signals and trajectory data through multiple camera streams for improved scene perception. Implemented on the Jetson Orin Nano, V-CAS enables real-time collision risk assessment and proactive mitigation via adaptive braking. A comprehensive training process was conducted on various datasets for comparative analysis, followed by fine-tuning the selected object detection model using transfer learning. The system's effectiveness was rigorously evaluated on the Car Crash Dataset (CCD) from YouTube and through real-time experiments, achieving over 98% accuracy with an average proactive alert time of 1.13 seconds. Results show significant improvements in object detection and tracking, enhancing collision avoidance compared to traditional single-camera methods. This research highlights the potential of low cost, multi-camera embedded vision transformer systems to advance automotive safety through enhanced environmental perception and proactive collision avoidance mechanisms.

Index Terms—vehicle collision avoidance, Jetson Orin, object detection, multiple camera fusion, RT-DETR.

Contents

Chapter 1	1
Introduction	1
1.1 Background.....	1
1.2 Problem Statement.....	4
1.3 Research Objectives.....	4
1.4 Contribution.....	4
1.5 Thesis Organization	5
Chapter 2	6
Literature Review	6
2.1 Collection of Related Articles	6
2.1.1 Year Wise Publications Analysis.....	8
2.1.2 Country Wise Publications Analysis	9
2.1.3 Journal Wise Publications Analysis.....	11
2.1.4 Institutional Analysis	13
2.2 Existing Approaches for Collision Prediction	14
2.3 An Overview of DL Based Methods for VCA	17
2.5 Threat Assessment Approaches	30
2.5.1 Logic Based Approaches	30
2.5.2 Set Based Approaches	30
2.5.3 Probability Based Approaches	31
Chapter 3	33
Methodology	33
3.1 Introduction.....	33

3.2 Working Principle.....	34
3.2.1 Object Detection – RT-DETR.....	35
3.2.2 Object Tracking – DeepSORT	37
3.2.3 Integrating DeepSORT with RT-DETR and PyTorch	38
3.2.4 Speed Estimation.....	39
3.2.5 Calculating Relative Rate of Acceleration	40
3.2.6 Brake Light Detection	41
3.2.7 Fusion of Multiple Camera Sensors	42
Chapter 4	46
Experimental Results.....	46
4.1 Datasets.....	46
4.1.1 For Object Detection.....	46
4.1.2 For Collision Avoidance Evaluation.....	47
4.2 System Resources and Training Setup	48
4.3 Evaluation Metrics	50
4.4 Results.....	52
Chapter 5	57
Conclusion and Future Work	57
5.1 Conclusion.....	57
5.2 Future Work.....	58
References	59

List of Figures

Figure 1.1 Schematic diagram of CV environment	2
Figure 2.1 Selection process of related articles	7
Figure 2.2 Year wise actual and predicted trend for related research	8
Figure 2.3 Country wise distribution based on density of publications.....	10
Figure 2.4 Co-citation network of journals in VCA	12
Figure 2.5 Collaborative network between institutions in VCA	14
Figure 3.1 Basic architecture of VCAS	35
Figure 3.2 Architecture of RT - DETR.....	36
Figure 3.3 Architecture of DeepSORT	38
Figure 3.4 Input and output matching of DeepSORT with RT-DETR.....	38
Figure 3.5 Simulated view of velocity from car dash camera view.....	39
Figure 3.6 Schematic view of relative acceleration of vehicles.....	40
Figure 3.7 Brake light ON / OFF detection.....	42
Figure 3.8 Workflow of multi-camera stream fusion.....	43
Figure 3.9 Horizontal stacking and resize of input streams.....	45
Figure 4.1 Experimental setup.....	49
Figure 4.2 Precision and recall comparison on vehicle i2 and brake light detection datasets .	54
Figure 4.3 Multi-Camera Fused Object Detection using Custom Trained RT-DETR	54
Figure 4.4 Day and Night Collision Prediction using VCAS on Car Crash Dataset.....	55

List of Tables

Table 2. 1 Most contributing countries in the field of VCA.....	9
Table 2. 2 Top 10 journals in the field of VCA and vehicle safety related publications.....	11
Table 2. 3 Top 16 institutions in the field of VCA and vehicle safety related publications....	13
Table 2. 4 A comparison between DL architectures used for vehicle safety.....	18
Table 4. 1 System resources for training and inference	49
Table 4. 2 Training parameters for both OD datasets	50
Table 4. 3 Comparison of different real-time object detectors on vehicle i2 public dataset ...	53
Table 4. 4 Comparison of different real-time object detectors on brake light detection dataset	53
Table 4. 5 V-CAS overall performance evaluation on Car Crash dataset	55

List of Abbreviations

ADAS	Advanced Driver Assistance Systems
AEB	Autonomous Emergency Braking
CCD	Car Crash Dataset
DAS	Driver Assistance Systems
V2X	Vehicle-to-Everything
CVs	Connected Vehicles
TTC	Time-to-Collide
ABS	Anti-locking Brake Systems
FOV	Field of View
OD	Object Detection
DL	Deep Learning
DRL	Deep Reinforcement Learning
CV	Computer Vision
LCD	Liquid Crystal Display
VCA	Vehicle Collision Avoidance
YOLO	You Only Look Once
CNN	Convolution Neural Network
RNN	Recursive Neural Network
PWM	Pulse Width Modulation
IEEE	Institute of Electrical and Electronics Engineers
LiDAR	Light Detection and Ranging
RC-NN	Electromotive Force
SORT	Simple Online and Realtime Tracking
RPM	Revolutions Per Minute
DNN	Deep Neural Network
FC	Fully Connected
SVM	Support Vector Machine
LSTM	Long Short Term Memory
m/s	Meters Per Seconds
RMS	Root Mean Square
GPU	Graphics Processing Unit

GPS	Global Positioning System
VCAS	Vehicle Collision Avoidance System
WHO	World Health Organization
FSD	Full Self-driving
SSD	Single Shot Multibox Detector

Chapter 1

Introduction

1.1 Background

The increase in car ownership, driven by economic growth and the desire for convenience, has resulted in a rise in traffic accidents, leading to significant loss of life. Motor vehicle collisions continue to be a major public health issue, leading to a high number of casualties around the globe. As per WHO (World Health Organization), about 1.35 million people die yearly in traffic accidents. Moreover, between 20 to 50 million suffer from non-fatal injuries, causing long-term disabilities. The economic impact of these incidents is substantial, with losses amounting to 3% of most countries' gross domestic product. Research shows that approximately 77% of these accidents are caused by drivers [1]. This statistic shows the urgent need for better vehicle safety technologies. To overcome this alarming situation, we really need better car safety technology. Therefore, the development and implementation of advanced safety warning systems have become a prime focus of academic research and industrial practices. This concerning trend underscores the urgent need for intelligent road safety systems that can perceive surrounding traffic objects and prevent collisions. These systems utilize various data sources, including vehicle speed, accelerometers, and video feeds. Recent advancements have seen researchers incorporating Light Detection and Ranging (LiDAR) sensor inputs and monocular camera images to enhance the performance of collision avoidance systems.

To help prevent accidents, researchers and car companies are working on new safety warning systems. These systems are important because they can spot danger quickly. Studies show that if a driver receives just half a second before crash warning, 60% of them could be avoided [2] – [4]. Connected vehicles (CVs) are helping with this. They use special technology to talk to each other and to things around them. Figure 1.1 shows how this works. This talking between cars and road signs or traffic lights is called Vehicle-to-Everything (V2X) communication [5].

It helps drivers know more about what's happening around them. Researchers are looking at how these CV systems can make driving safer. The systems use information from sensors and V2X to spot dangers on the road. They might warn drivers with sounds or lights. In really dangerous situations, they might even brake or steer the car automatically.

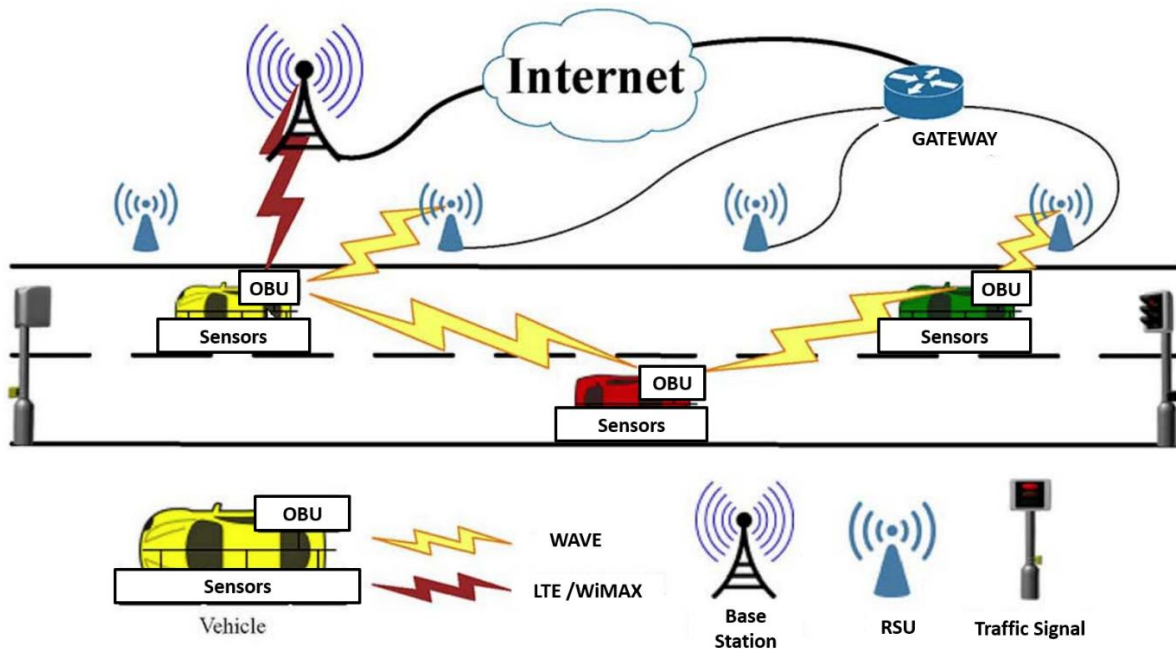


Figure 1.1 Schematic diagram of CV environment

Self-driving cars have changed how we think about safety on the road. Researchers are now using smart computer programs to make these cars safer. Two types of programs are really important: deep learning (DL) and reinforcement learning (RL). DL assist cars see and understand what's around them. It's like teaching a computer to recognize things in pictures, but for cars. This helps self-driving cars spot other vehicles, people, and signs on the road. RL is different. It helps cars figure out the best way to drive. It's like the car practices driving over and over, learning from its mistakes. This helps cars make good choices in tricky driving situations. When you combine these two, you get deep reinforcement learning (DRL). This super-smart system helps cars navigate better. It's like the car has both good eyes and a good brain for driving. These new ways of teaching cars to drive themselves are making big improvements in how safe they can be on the road. While DRL and sensor fusion techniques show their robustness, vision-based approaches offer a promising alternative due to their cost-effectiveness and ease of integration. Monocular vision proved to be valuable by estimating Time-to-Collide (TTC) and addressing the issue of collision rarity. Vehicle Detection achieved higher accuracy in classifying and counting vehicles across various highway videos [6].

Another important sub field in the integration of intelligent vehicle systems, is multi-sensor fusion. These sensors are like different senses for the car. Cameras act like eyes, showing the car things like lane lines and traffic lights. LiDAR is like super-accurate 3D vision for the car. It can measure exactly how far away things are. There's also radar and GPS. By using all these tools together, cars can get a much better picture of what's happening around them. It's like how we use our eyes, ears, and sense of touch to understand our surroundings. This helps cars spot dangers more accurately and avoid crashes more reliably. This approach makes self-driving cars much better at staying safe on the road. It helps them see and understand their surroundings more like a human would do. In contrast, the Hall effect speedometer sensors as the name indicates works on the Hall effect phenomenon. According to this phenomenon, a voltage difference in a semiconductor is produced when exposed to a magnetic field. In this configuration, a magnet is fastened to the moving part of the vehicle, while a Hall effect sensor is positioned in close proximity. Whenever the magnet passes the sensor during rotation, its magnetic field is disturbed which results in variation of the voltage in the sensor. The fluctuation in the voltage is captured and translated into a digital signal. This signal is subsequently utilized to ascertain the speed of the wheel and is finally visualized on the speedometer.

ADAS can be categorized into two main types: (1) Passive Safety focuses on reducing injuries during a crash through high production safety standards, while (2) Active Safety systems proactively prevent accidents by using sensors such as radar, cameras, and ultrasonic devices to detect potential hazards like nearby vehicles or sudden braking. When a threat is identified, these systems alert the driver with visual or audio warnings or initiate automatic braking to avert collisions. Modern systems often integrate cameras and radars, providing distinct advantages. However, the addition of sensors can increase vehicle costs and design complexity. To address this, researchers are investigating computer vision insights, particularly in object detection (OD) techniques that utilize either depth-based or camera-based sensors. The proposed system utilizes spatial feature extraction from RGB feeds captured by three cameras, facilitating enhanced scene interpretation and a broader field of view (FOV). It integrates object detection and tracking algorithms to predict collision scores based on relative motion, all executed efficiently in real-time on edge devices like the Jetson Orin Nano. This method promises a more robust and computationally efficient recognition of surrounding traffic objects.

This study synthesizes the latest research on safety warnings and threat assessment for autonomous driving technologies. By focusing on the integration of cutting-edge technologies and methodologies using DL and DRL based vision only and multi sensor fusion approaches, this paper aims to contribute to the ongoing efforts to improve vehicle safety and reduce the global burden of traffic accidents. The conclusion and future development at the end of this review highlight the potential areas for further investigation, ensuring to enhance the effectiveness and reliability of collision avoidance systems for vehicle safety.

1.2 Problem Statement

Traffic accidents are a major concern, with a significant portion caused by driver error. Existing collision avoidance systems often rely on single cameras and outdated deep learning techniques, limiting their effectiveness. That's why we tried to propose a method to predict collision proactively in real-time and with better accuracy and environmental perception.

“To develop a robust and efficient real-time collision avoidance system using a multi-camera and deep learning approach to improve vehicle safety.”

1.3 Research Objectives

The objective of this research undertaking are as follows:

1. To carryout a comprehensive survey of vehicle collision avoidance techniques and further investigate the effectiveness of Deep Learning (DL) techniques for enhancing the performance and reliability of collision avoidance systems, especially for vehicles.
2. Devise a novel real time DL based model using multiple camera streams and computational compatibility of low power embedded systems like Jetson Orin Nano for autonomously generating various alerts and applying adaptive braking action in case of emerging collision threats.
3. Evaluate the proposed model on publicly available traffic datasets and real-world scenes captured from various sources to assess its detection accuracy and robustness.
4. Compare the performance of proposed model with existing collision avoidance and prediction methods to identify strengths, weaknesses, and areas for improvement.

1.4 Contribution

Keeping the research objectives in mind, we have made following contributions:

1. **Publication** Muhammad Waqas Ashraf, Ali Hassan, Imad Ali Shah. “V-CAS: A Realtime Vehicle Anti Collision System Using Vision Transformer on Multi-Camera

Streams”, 23rd IEEE International Conference of Machine Learning and Applications (ICMLA) <https://doi.org/10.48550/arXiv.2411.01963>, 2024 IEEE DOI 10.1109/ICMLA61862.2024.00138 (Published).

2. **Publication** Muhammad Waqas Ashraf, Imran Shafi, Ali Hassan, Imad Ali Shah, Muhammad Murad Khan “A Survey on Contemporary Collision Avoidance Techniques for Ground Vehicles”, ACM Computing Surveys (under review).

1.5 Thesis Organization

Chapter 1 is an introduction that acquaints with the proposed topic and research objective. Chapter 2 includes literature review done related to the proposed topic. This literature review also incorporated the systematic selection of related articles method and an in-depth discussion of related field articles. Chapter 3 explains the detailed methodology adopted to achieve the objective of the research. Chapter 4 includes experimental setup, results and their analysis. Chapter 5 includes the conclusion, and future directions.

Chapter 2

Literature Review

2.1 Collection of Related Articles

When conducting a comprehensive review of the Vehicle Collision Avoidance (VCA) field, the selection of a suitable database is crucial for ensuring the quality and comprehensiveness of the literature analysis. Among the various available options, Web of Science (WoS) stands out as the best choice for several reasons. Firstly, WoS is renowned for its rigorous selection criteria, ensuring that only high-quality, peer-reviewed publications are included in its database [7]. This is particularly important in a rapidly evolving field like VCA, where the quality of research is paramount. Secondly, WoS offers extensive coverage across multiple disciplines, which is essential given the interdisciplinary nature of collision avoidance technologies, spanning areas such as computer science, engineering, and transportation [8]. The database's robust citation tracking capabilities enable researchers to identify seminal works and trace the evolution of ideas within the field [9]. Additionally, WoS provides comprehensive metadata and standardized indexing, facilitating more accurate and consistent bibliometric analyses [10]. While other databases like Scopus and IEEE Xplore also offer valuable resources, WoS's unique combination of quality control, interdisciplinary coverage, and analytical tools makes it particularly well-suited for a thorough review of the VCA literature. Furthermore, WoS's integration with other research tools and its ability to export data in formats compatible with various bibliometric software enhances the diversity of possible analyses. These factors collectively justify the selection of Web of Science as the primary database for this review, ensuring a comprehensive and high-quality foundation for the bibliometric analysis of the Vehicle Collision Avoidance field.

The VCA field encompasses various sub-categories based on various road and traffic situations, requiring a carefully designed index string to show different analysis accordingly. Drawing from the work of [11] and [12], a comprehensive set of keywords was developed to

capture the multifaceted nature of VCA research. This index string includes terms related to vehicle collision avoidance, vehicle safety, autonomous driving, DAS, Automatic Emergency Braking (AEB), Advanced Driver Assistance System (ADAS), and path planning, among others. The final search string consists of 14 keywords linked by the logical operator "OR", designed to retrieve relevant publications from the Web of Science (WoS) Core Collection [13], [14]. The literature search covered the period from January 1, 2000, to June 30, 2024, a timespan of over 20 years, which, according to [12] and [15], provides sufficient statistical data to summarize the overall landscape of a research field. The search was conducted using the Science Citation Index Expanded (SCIE) and Social Sciences Citation Index (SSCI) databases, known for their renowned coverage of scientific literature. Initially, 457,407 publication records were retrieved, which were then subjected to a rigorous screening process to ensure relevance and quality.

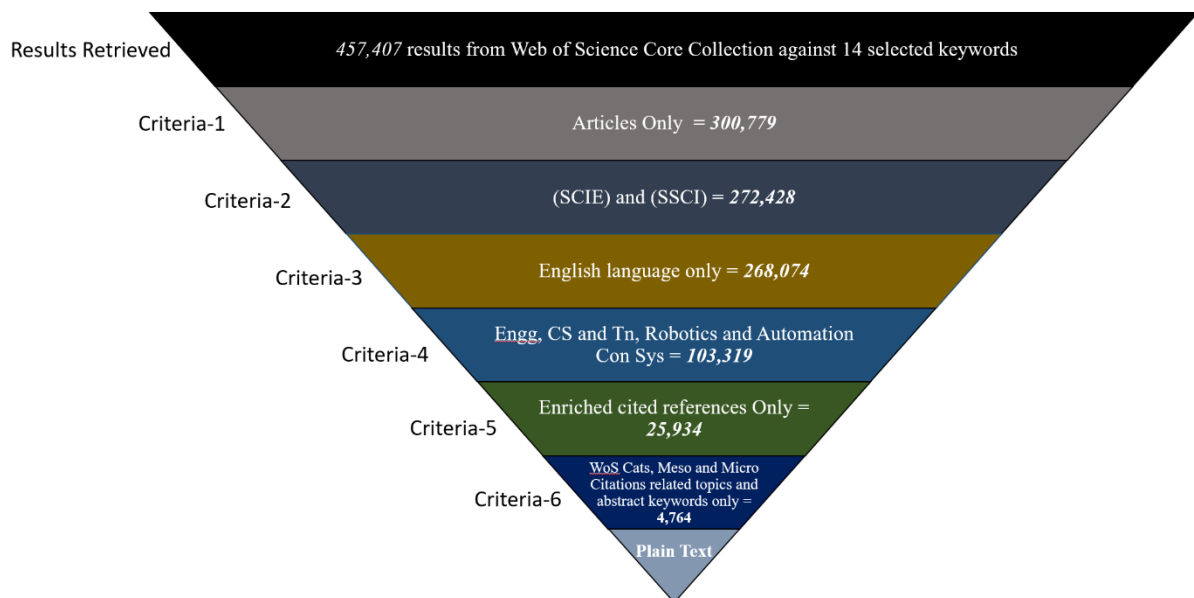


Figure 2.1 Selection process of related articles

After applying evaluation criteria to drop irrelevant records and focusing solely on high-quality articles in English language having enriched cited references only, the number of publications was reduced to 25,934. Further manual verification of titles and abstracts on macro and micro level led to the exclusion of 21,170 articles that primarily not related with vehicle collision, ADAS or autonomous driving systems or unrelated to VCA in general. The final dataset for bibliometric analysis consisted of 4,764 documents, authored by 15,323 researchers from 3,358 institutions across 95 countries, published in 247 different journals. This carefully curated

dataset forms the basis for a comprehensive analysis of the VCA field, providing insights into its development, key contributors, and emerging trends. The relevant publication data was exported to plain text files for analysis. Figure 2.1 illustrates a flowchart of the detailed screening process.

2.1.1 Year Wise Publications Analysis

Research in vehicle safety and collision avoidance has grown significantly over the years. We made a chart to show how many papers were published each year, focusing particularly on the last five years from 2019 to 2023. This recent data reveals a clear upward trend in publication numbers. In the early years of the field, not many papers were published - only a handful per year on average. This was probably because the technology and theories weren't advanced enough yet. But in recent years, things have really taken off. The growth got even faster after 2019, showing how important this topic has become. Given the strong upward trend in the last five years, we used linear regression to predict how many papers might be published in the future. This mathematical approach suggests the number will keep growing steadily from 2025 to 2030 as shown in Figure 2.2. However, it is highlighted that while we expect continuous growth, the actual rate might vary as the field evolves.

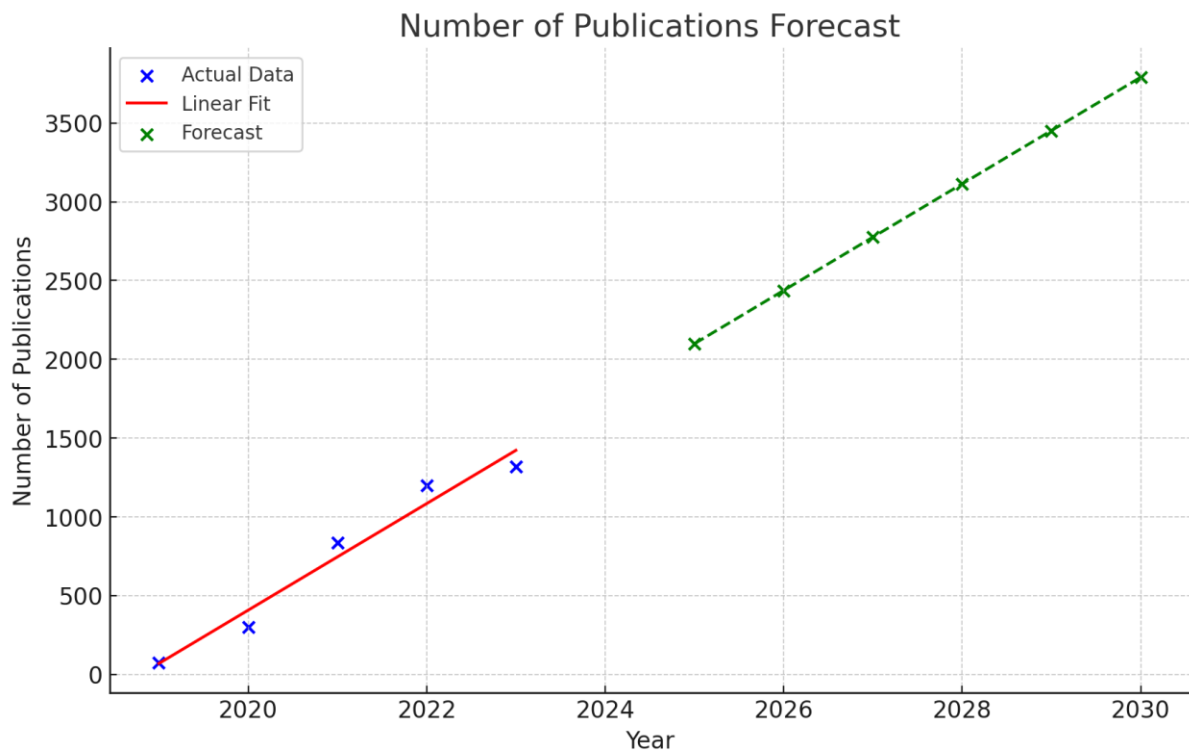


Figure 2.2 Year wise actual and predicted trend for related research

This pattern - starting slow, then growing fast - is common in research fields. It usually means that as technology advances and more researchers get involved, the number of studies increases. Our analysis helps show where this field of vehicle safety and collision avoidance is heading and why it's a crucial area to review right now. By focusing on the most recent five years and using linear regression for future predictions, we aim to capture the current momentum in the field and provide a reasonable estimate of its short-term trajectory. This approach allows us to highlight the growing importance of vehicle safety and collision avoidance research in recent years and its potential for a growing trend in near future.

2.1.2 Country Wise Publications Analysis

The field of vehicular safety and collision avoidance systems has seen contributions from a diverse global community, spanning 95 nations. An examination of publication output reveals interesting patterns in research productivity and impact across different countries. Table 2.1 summarizes top 10 countries number of publications wise along with their citations. At the forefront of this field, two nations stand out prominently, Peoples Republic of China and USA. The country, China, with the highest publication count has produced 2573 articles, followed by the second-ranking nation, USA with 879 publications.

This leading country's strong showing can be attributed in part to governmental initiatives promoting vehicle-infrastructure integration, exemplified by projects like TTIC-VG [16] and [17]. This project has funded several significant studies, including investigations into communication topologies for uniform vehicle groups, control systems for mixed vehicle types, and cooperative strategies for cyclical vehicle formations enhancing vehicle safety and autonomous driving capabilities.

Table 2. 1 Most contributing countries in the field of VCA

Rank	Country	No of Publications	No of Citations	Rank	Country	No of Publications	No of Citations
1.	China	2573	16406	6.	England	159	1781
2.	USA	879	6942	7.	Germany	139	1022
3.	South Korea	301	1724	8.	Italy	117	648
4.	Canada	222	1773	9.	Australia	115	1075
5.	India	178	1232	10.	Spain	105	751

Interestingly, when we analyze qualitatively, it is seen that while the top-producing country leads in quantity, the runner-up shows superior impact in terms of citations. The second-ranked nation's works have garnered 6942 citations for 879 publications which is almost 8 citations per article, outpacing the leader's 16406 citations against 2573 publications making 6.4 citations per article by a substantial 25 % more citations per document. This disparity suggests potential areas for improvement in the quality and influence of research from the leading country. The third and fifth positions are held by two Asian nations, South Korea and India, contributing 301 and 178 works respectively. Their citation counts are 1,724 and 1,232. It's noteworthy that among the top ten contributing countries, only one country India is classified as a developing nation, underscoring its significant role in advancing this field of study.

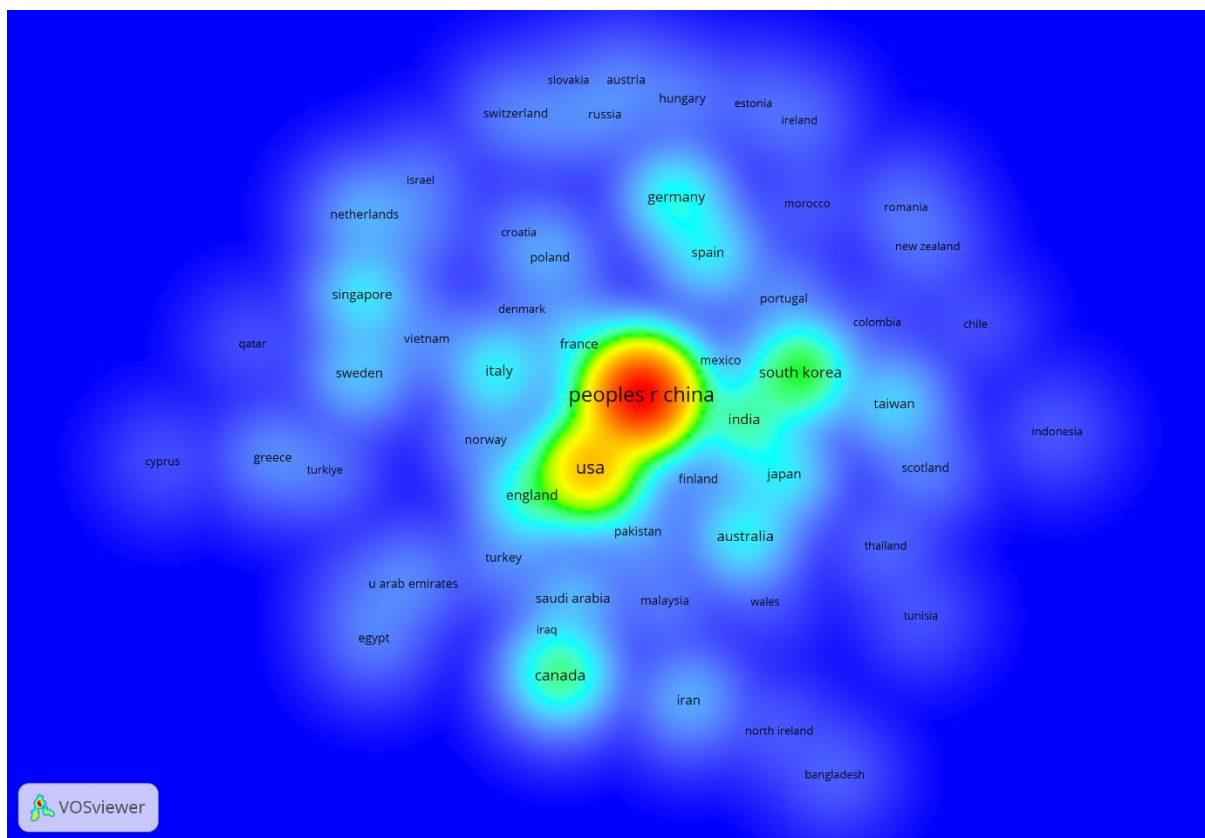


Figure 2.3 Country wise distribution based on density of publications

To visualize the global distribution of research activity, we employed citation analysis tools to create a knowledge map (Figure 2.3). This map focuses on 61 countries, excluding those with fewer than five publications. The resulting visualization consists of a heatmap, country labels, and connecting color blocks. The intensity of red in the heatmap correlates with higher publication volumes, while blue indicates lower output. Concentric circles within the heatmap

suggest strong collaborative networks. The map clearly identifies the two leading nations as research powerhouses in this domain, with one showing a darker red, indicating a higher publication count. This visual representation aligns with the numerical data presented earlier.

Moreover, the map reveals intricate patterns of international collaboration. The background color blocks connecting different countries highlight these cooperative relationships. For instance, the top two nations show strong collaborative ties. Additionally, some countries, such as England, India, South Korea and France, maintain collaborative links with both of the top-producing nations. This analysis not only highlights the dominant players in the field but also illuminates the complex web of international cooperation driving advancements in vehicular safety and collision avoidance research. Such collaborations are crucial for enhancing the academic value and impact of research in this vital area.

2.1.3 Journal Wise Publications Analysis

The Scientific journals play a significant role in sharing research findings. We looked at which journals published the most about smart car systems. Table II shows the top 10 Journals in the VCA field along with their citations of related articles and impact factor. IEEE Xplore stands out, publishing half of these top journals. The rest come from different publishers like MDP, Elsevier and Wiley. This shows IEEE Xplore is a leader in this field.

Table 2. 2 Top 10 journals in the field of VCA and vehicle safety related publications

Rank	Journal Name	No of Publications	No of Citations	Impact Factor	Publisher
1.	Sensors	655	4014	3.7	MDPI
2.	Transportation Research Record (TRR)	330	1016	1.6	SAGE-Journals
3.	IEEE Transactions on Intelligent Transportation Systems (T-ITS)	316	2654	9.5	IEEE Xplore
4.	Electronics	238	1155	2.6	MDPI
5.	IEEE Access	225	802	3.9	IEEE Xplore

6.	Journal of Advanced Transportation (JAT)	186	867	2.3	Hindawi
7.	Journal of Automobile Engineering	183	574	1.7	SAGE-Journals
8.	IEEE Robotics and Automation Letters	154	1546	4.6	IEEE Xplore
9.	IET Intelligent Transportation System (ITS)	134	621	2.5	Wiley Online Library
10.	Traffic Injury Prevention	121	555	2.2	Taylor and Francis

The star performer is Sensors by MDPI. It has the most 655 articles, gets cited the most 4,014 times. However, IEEE Xplore published Journals like IEEE T-ITS has a strong impact factor of 9.5 and a very good Cite Score. IEEE Xplore is leading publishers with three Journals in top 10 most related articles Journals with decent citations per article. We also looked at how often these journals are cited together, which according to [18] is an effective way to judge a journal's impact. We made a network map of these connections using VOSviewer, shown in Figure 2.4. It includes 777 journals that have at least 20 articles in this field.

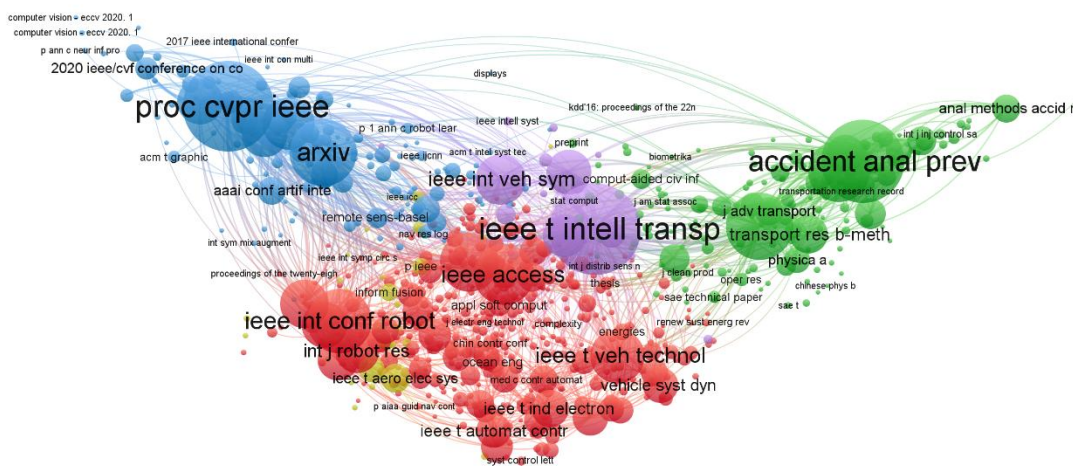


Figure 2.4 Co-citation network of journals in VCA

The connectivity map shows six main groups. IEEE TITS, IEEE CVPR and IEEE Access are the big players in three of these groups. Interestingly, there are lots of connections between these main journals across different groups. This analysis helps us see which journals are the most important for research on smart car collision avoidance systems. It also shows how different areas of this research are connected.

2.1.4 Institutional Analysis

Quantitative statistics enable comparisons of scientific outputs between institutions to identify which are the leading contributors and their quality of work can also be accessed through the linkages and citations. We looked at which universities and research centers are doing the most work on smart car systems. Table 2.3 shows the top 16 universities in the related field. Six of these are in China, with the others in the US, Netherlands, and Singapore. Tsinghua University in China is the big leader, with 176 papers that have been cited 4,137 times. Southeast University, also in China, comes second with 137 papers. Together, these top 10 places account for nearly a third of all the research in this area. This shows that China is really pushing this field forward, and that a small number of places are doing a lot of the work.

Table 2. 3 Top 16 institutions in the field of VCA and vehicle safety related publications

Rank	Institution	Country	Publications	Citations
1.	Tsinghua Univ	China	179	1440
2.	Tongji Univ	China	154	994
3.	Southeast Univ	China	121	654
4.	Beijing Inst Tech	China	104	797
5.	Jilin Univ	China	103	532
6.	Beihang Univ	China	84	644
7.	Changan Univ	China	78	519
8.	Zhejiang Univ	China	78	593
9.	Chinese acad SCI	China	76	725
10.	Beijing Jiaotong Univ	China	65	350
11.	Shanghai Jiaotong Univ	China	62	416
12.	Harbin Inst Tech	China	62	344
13.	Chongqing Univ	China	60	496
14.	Nanyang Tech Univ	Singapore	58	697
15.	Hunan Univ	China	58	425
16.	Univ Michigan	USA	54	478

How these institutions work together was also analyzed using VOSviewer. We focused on 114 places that have published at least 10 papers. In the map (Figure 2.5), each place is shown as a dot, with bigger dots meaning more papers. Lines between dots show teamwork, with thicker lines meaning more collaboration.

The map shows 11 main groups and 1,089 connections. Tsinghua University has the biggest dot, showing it's the most active. Southeast University and Beihang University are the next biggest. These main players have lots of thick lines connecting them to others, which means they're working with lots of different places. This analysis helps us see not just who's doing the most work, but how different research centers are working together to advance smart car technology.

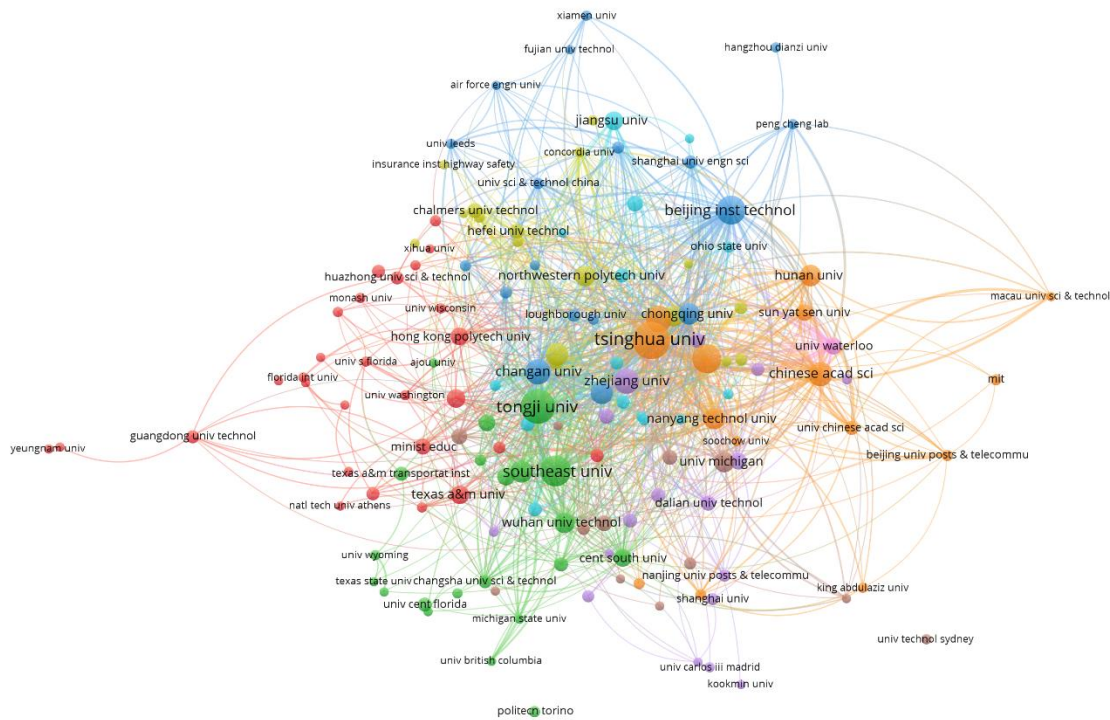


Figure 2.5 Collaborative network between institutions in VCA

2.2 Existing Approaches for Collision Prediction

Existing approaches for collision prediction and avoidance systems can be broadly categorized into three main groups: (1) motion trajectory prediction-based models using Deep Reinforcement Learning (DRL), (2) radar-camera sensor fusion techniques, and (3) vision-based approaches with Deep Learning (DL).

2.2.1 Motion Trajectory Prediction using DRL

An efficient collision detection system relies on confident prediction of vehicle motion and trajectory. Lefevre et al. [19] explores various motion prediction approaches, analysing the trade-off between model complexity and real-time implementation and categorized motion prediction models into three main types: physics-based, maneuver-based and interaction-aware. DRL, where algorithms learn from trial and error, has shown promise in navigation systems. Kahn et al. [20] proposes a collision avoidance mechanism using a standard stereo camera with a navigation model outperformed Double Q-learning in achieving fully autonomous navigation. Chen et al. [21] shows a decentralized collision avoidance algorithm using DRL employing a value network that predicts best paths with minimal collision risk, considering the positions and velocities of surrounding vehicles. Kim et al. [22] designed intelligent self-driving policies using DRL to reduce intersection collision risk. While DRL holds promise for real-time collision avoidance systems, limitations do exist as these algorithms require vast amounts of training data and limit real-world generalizability. Additionally, their computational demands can lead to delays in critical moments where fast reactions are crucial, making them hard to be implemented in real-time for passenger vehicles.

2.2.2 Radar-Camera Sensor Fusion Approaches

Several DL architectures have been proposed for radar-camera and LiDAR sensor fusion in collision avoidance systems. Radar offers all-weather functionality, detecting objects using electromagnetic waves. However, detailed information about object size and shape is absent. Camera-based sensors provide rich visual data like lane markings, traffic signals, and object shapes. LiDAR (Light Detection and Ranging) creates a 3D point cloud representation of the environment, offering precise distance and shape information but is costly. Sensor fusion by Kim et al. [23] combines the strengths of all the sensors, leading to more robust and reliable perception for collision avoidance systems. Some of the approaches are: Early Fusion which merges raw radar data and camera images at the beginning of the network and processed by a single DL model for such as proposed by Xu et al. [24]. This approach is computationally efficient but requires careful pre-processing. Late Fusion separates DL models process for radar and camera data independently, extracting features which are then fused at a later stage for final decision-making. Kim et al. [25] proposed late fusion of camera with LiDAR data for pedestrian detection. This approach allows for independent optimization of each sensor model but might lose some information. Feature Level Fusion involves processing both sensors through individual feature extraction layers which are then concatenated before feeding them

into a final classification layer. Zhu et al. [26] introduced multi-sensor-based feature level fusion. This approach leverages the strengths of both sensors while maintaining some level of independence in the feature extraction process. Attention-based Fusion focuses on the most relevant features, dynamically allocating weights for improved robustness but is computationally expensive as seen in the proposed method by Huang et al. [27] for vehicle detection.

2.2.3 Vision-based Approaches with Deep Learning (DL)

While DRL and sensor fusion techniques show their robustness, vision-based approaches offer a promising alternative due to their cost-effectiveness and ease of integration. Direct perception focuses solely on steering angle prediction as done by Chen et al. [28] and lack real-world testing under complex situations. Monocular vision proved to be valuable by estimating Time-to-Collide (TTC) and addresses the issue of collision rarity. Existing methods such as by Shi et al. [29] for TTC estimation include feature tracking, motion divergence analysis, and optical flow techniques. These approaches have limitations like lack of hardware efficiency, robustness, and reliance on additional data like lane markings. Azimjonov's method leverages YOLO, a DL model, for vehicle recognition and tracking in traffic videos as proposed by Datondji et al. [30]. Vehicle Detection achieved higher accuracy in classifying and counting vehicles across various highway videos [31]. Redmon et al. [32], proposed YOLO which revolutionized object detection with a real-time, single pass approach by using regression for both bounding box coordinates and object probabilities. Chen et al. [33] proposed YOLO v3-live, a modification of YOLO v3-tiny designed for real-time vehicle detection on embedded devices. It prioritizes faster processing by minimizing down-sampling feature maps, resulting a slight decrease in accuracy but maintaining acceptable detection speed. However, further research is needed to improve accuracy without sacrificing speed. Keeping in view the cost effectiveness, ease of integration, and optimal performance for real-time predictions, our focus is towards vision-based methods acquiring input data from multiple camera streams. The scope of achieving improving the accuracy of object detection over multiple cameras by gathering a more comprehensive view of the scene was already discussed as future works by Sharma [34]. Researchers are using extensively **YOLO with DeepSORT**, including Ngeni et al. [35] and Lin et al. [36], for real-time traffic occlusion and tracking related models. Similarly, various models of SSD models like MobileNets v1 to v3 [37, 38, 39] were also used for real time object detection specially for low power embedded devices, however, their detection performance in terms of precision and accuracy was not very promising specially to be used in real world traffic

safety scenarios where accuracy is the most vital metrics. Transformers, on the other hand showed very promising results in terms of contextual relationship and accuracy for NLP but for a considerable period from their first model in 2018 till late 2023, due to their lack of real-time ability, were not being used in vision related tasks. *RT-DETR* [40] proposed by Zhao et al in CVPR 2024 and proved their vision-based transformer model as the new contender to become SOTA real-time object detector where it has beaten YOLOs in performance and speed. Many leading autonomous car manufacturers like *Tesla and Kia* have also shifted towards totally vision based object detection systems. Tesla's *Full Self-Driving (FSD)* software [41] relies solely on cameras, abjuring radar and lidar. This approach leverages vast amounts of video data from Tesla's fleet to improve its AI systems for depth estimation and object detection. The company believes that the cost-effectiveness and scalability of cameras will accelerate their path to full autonomy, despite the lack of redundancy that comes with using multiple sensor types.

2.3 An Overview of DL Based Methods for VCA

Deep learning is a sophisticated computational paradigm that enables feature extraction and representation learning across multiple levels of abstraction [42]. As a branch of machine learning, it autonomously identifies patterns and features from raw data, making predictions or taking actions based on predefined reward functions [44]. This field encompasses various techniques, including neural networks, hierarchical probabilistic methods, supervised and unsupervised learning models, and deep reinforcement learning (DRL).

Autonomous vehicles have attracted substantial interest and investment, largely due to breakthroughs in deep learning, convolutional neural networks (CNN), and deep neural networks (DNN) [45]. Unsupervised learning in autonomous driving aims to interpret the driving environment with minimal human input [46]. Unlike support vector machines (SVM), deep learning can tackle complex, non-linear problems without resorting to higher-dimensional projections [43]. It uses many hyper-parameters and layers to solve intricate problems. To achieve human-like driving capabilities from a computer vision perspective, autonomous vehicles must recognize their environment, interpret 3D representations, discern object and pedestrian movements, and navigate human emotions [47]. While deep learning algorithms excel in perception-control learning from data, the high costs associated with LiDAR technology and manual map annotation pose challenges to widespread adoption in autonomous driving [48]. Therefore, the techniques of computer vision only using camera sensors are widely adopted by many car making giants as a cheaper and realistic alternative. Manufacturers

have introduced features such as collision warning systems, blind spot monitors, lane departure alerts, rear-view cameras, and autonomous braking mechanisms [49] using deep learning techniques. For road infrastructure design, simulation-based tools like Site3D, RoadEng [50], and OpenRoads Designer [51] have become increasingly prevalent. Intelligent traffic management has seen the rise of GPS-enabled navigation systems, including sophisticated software capable of interpreting road user behavior to optimize route decisions [52].

Numerous automotive companies have embraced AI-driven safety innovations. Prominent manufacturers like Tesla, Audi, and BMW have pioneered sensor and vision-based perception systems, enabling drivers to assess road conditions more accurately and utilize partial autonomy features [53, 54]. The rapid advancement of deep learning techniques in video processing, coupled with the availability of cost-effective, high-performance computational platforms such as GPUs and TPUs, has accelerated the development of AI-based functionalities at both vehicle and infrastructure levels [55].

Table 2. 4 A comparison between DL architectures used for vehicle safety

Architecture	MLP	CNN	RNN / LSTM	Transformers
Pros	<ul style="list-style-type: none"> • Straightforward design • Easy to implement on low power hardware 	<ul style="list-style-type: none"> • Appropriate for high dimensional data • Learns features with locality shift-invariance • Powerful for CV tasks 	<ul style="list-style-type: none"> • Appropriate for limited level sequential data • Can learn previous mid-term dependencies 	<ul style="list-style-type: none"> • Good for really long-term sequences • Process sequences in parallel • Enables self-attention • Very good for NLP and CV (but not in real time)
Cons	<ul style="list-style-type: none"> • Cannot handle complex situations having large real-world parameters 	<ul style="list-style-type: none"> • Weak on long sequence data • Vanishing Gradients (solved by a complex architecture ResNet) 	<ul style="list-style-type: none"> • Unable to perform on large long-term dependencies • Hard to train the gradients issue • Handles on serially fed data 	<ul style="list-style-type: none"> • Very long and extensive resource required training • Over complicated for short sequences • Poor in real-time tasks

At the core of vision-based driving safety analysis lies the application of deep learning methods for image and video processing. It's worth noting that recent progress in DL has been largely propelled by advancements in computer vision and natural language processing, which represent key areas of visual and sequential data analysis. State-of-the-art DL platforms typically leverage architectures such as multilayer perceptron, convolutional neural networks, recurrent neural networks, and transformers as fundamental components. Table 2.4 offers a concise yet informative comparison of these methodologies. CNNs, due to their real-time performance, are the best choice for on road vehicle safety and collision avoidance tasks whereas, Transformers, being the new player, are good for contextual and long-term analysis where the time constraints are not too much strict. Few important DL techniques being used in our area of interest are being discussed briefly.

2.3.1 Convolutional Neural Networks (CNN)

Convolutional neural networks (CNN) have emerged as a powerful tool in image classification and computer vision, achieving remarkable success with perfect classification rates on datasets like ImageNet [56]. The CNN architecture is characterized by its ability to learn progressively complex features through successive neural layers in a supervised manner, utilizing backpropagation of classification errors to refine its performance [57]. A key distinction of CNN is their integrated approach to feature extraction and classification. Unlike traditional methods, CNN do not rely on separate modules for these tasks, nor do they require unsupervised pre-training. Instead, they learn input representations implicitly through supervised training, eliminating the need for manual feature description and extraction [58]. This allows CNNs to derive features directly from raw pixel data, culminating in final object categories.

In the context of autonomous vehicles, CNN exhibit remarkable versatility in input processing, capable of handling various data types such as images, video, text, and audio. These inputs can have one-to-one, one-to-many, or many-to-many relationships with output classes. The depth of a CNN, determined by its number of layers, is analogous to its feature-learning capacity. Through backpropagation, the network optimizes its feature extraction across filters of various sizes [59]. Recent advancements in CNN architecture design leverage transfer learning, where pre-defined convolutional layers are combined with fully connected layers, obviating the need to train networks from scratch [60]. In a typical CNN pipeline for autonomous driving, input images undergo convolution with activating functions to generate feature maps, which can be further refined to identify salient patterns. CNN exhibit robustness to translational and rotational variances due to their convolutional nature, applying consistent weights across the

input. Each successive layer in the network identifies increasingly complex features, starting from simple elements in the initial layer and progressing to more intricate patterns in deeper layers. The final stage of processing typically involves fully connected neural networks (FCNN) that operate on the extracted feature maps.

The ultimate goal of CNN application in autonomous vehicles extends beyond current semi-autonomous models with ADAS. It aims to dramatically reduce driver responsibilities and engagement, potentially eliminating the need for active human involvement in the driving process. This vision represents a significant leap forward in automotive technology and has far-reaching implications for transportation and society at large.

2.3.2 Recurrent Neural Networks (RNN)

Recurrent neural networks (RNN) are specialized architectures designed to recognize sequences and patterns in data through recurrent computations, enabling sequential processing of input information [61]. This design allows RNN to maintain an internal state that can capture temporal dependencies in the data. Long short-term memory (LSTM) networks, a specific type of RNN, have gained prominence due to their ability to handle long-range dependencies more effectively. LSTMs utilize a sophisticated gating mechanism, incorporating input, output, and forget gates to control the flow of information through the network [62]. This structure enables LSTMs to selectively remember or forget information from previous time steps, making decisions based on both current inputs and relevant historical context [63]. The defining feature of RNN is their cyclic connection structure, where outputs from one-time step serve as inputs for the next. This creates a directed cycle within the network, allowing information to persist and influence future computations [64]. This recursive nature makes RNN particularly well-suited for tasks involving sequential or time-series data.

In the context of autonomous vehicles, RNN have demonstrated their utility in visual tracking tasks, especially under constrained scenarios [65]. Their ability to maintain temporal context allows for more robust and accurate tracking of objects across video frames. The temporal correlation capabilities of RNN enable predictive modelling for object tracking. By using the region of interest (ROI) from one frame to predict an object's position in the subsequent frame, RNN can create a continuous tracking model [66]. This approach mimics a prediction-correction cycle, where each new frame's input is informed by the predictions made from previous frames.

This predictive capacity of RNN is particularly valuable in autonomous driving scenarios, where anticipating the movement of other vehicles, pedestrians, and objects is crucial for safe navigation. By leveraging historical information and current inputs, RNN-based systems can make more informed decisions about likely future states, potentially improving the overall safety and efficiency of autonomous vehicles. The application of RNN and LSTMs in autonomous driving extends beyond mere object tracking. These architectures can also be employed in trajectory prediction, behavior modelling of other road users, and even in the decision-making processes of the autonomous system itself. Their ability to handle sequential data makes them powerful tools for understanding and predicting the dynamic environment in which autonomous vehicles operate.

2.3.3 Transformers

The evolution of DL methods for vision-based traffic video analysis has been marked by significant milestones in network architectures. These developments have primarily centred around fully connected (FC) layers, convolutional neural networks (CNNs), and recurrent neural networks (RNN). Parallel to computer vision advancements, deep learning methods for sequential learning, particularly in natural language processing (NLP), have seen remarkable progress. Long-standing dominance of gated recurrent units (GRU) [67] and long short-term memory networks (LSTM) [68] in sequential learning has been challenged by the introduction of the transformer architecture [69].

Transformers, introduced by Google researchers, feature an encoder-decoder structure utilizing multi-head self-attention modules. This design allows for capturing longer internal dependencies in addition to input-output relationships in sequential data. Position embedding enables parallel training and the ability to capture dependencies beyond sequential relations [69, 70]. The success of transformers in both NLP and computer vision tasks has been noteworthy. In NLP, Google's implementation [71] outperformed competitors across 11 tasks, marking a potential end to the LSTM era in this field. Whereas, in computer vision, transformers are challenging CNN dominance. Google's implementation [72] achieved an unprecedented 88.55% accuracy on ImageNet through transfer learning, while another study [73] attained 83.3% top-1 accuracy, surpassing ResNet50 with comparable parameters.

Transformers have demonstrated versatility in downstream tasks such as detection and semantic segmentation. Their potential extends beyond traditional CV tasks to processing sequential data, including trajectory extraction, tracing, and modelling individual and

collective behaviors of pedestrians and autonomous vehicles. The field anticipates increased adoption of transformer-based safety analysis frameworks in the coming years, not only for computer vision tasks but also for a wide range of sequential data processing applications in the autonomous driving domain. This shift represents a significant evolution in the approach to complex data analysis and prediction tasks in vehicular safety and collision avoidance systems.

2.3.4 DL Based Object Detection (OD) Techniques for Vehicle Safety Applications

Object detection (OD) is a crucial component in DL based techniques for driving safety analysis. This process involves locating and identifying various objects within images or video frames, often in complex environments, by drawing bounding boxes around objects of interest. Object detection can be integrated with or exist alongside object classification and labelling tasks. In driving safety analysis, object detection is applied to identify key elements such as vehicles, pedestrians, road traffic signs and potential obstacles. The applications of object detection extend to more complex tasks like: -

1. Traffic distribution and composition analysis
2. Detecting improper lane crossing events
3. Trajectory extraction
4. Speed estimation
5. Moving object tracking
6. Path planning
7. Identifying vehicles on road shoulders

An additional application of object detection involves privacy protection, such as masking personally identifiable information like human faces and license plate numbers before publishing traffic video footage. While there are existing datasets for traffic analysis from roadside cameras [74, 75], there remains a critical need for more comprehensive datasets covering diverse scenarios comprising urban, suburban, rural settings and various environmental conditions. Compared to conventional object detection algorithms like the Viola-Jones detector [76], histogram of oriented gradients (HOG detector) [77], and deformable part-based models (DPM) [78], CNN-based methods have significantly improved recognition success rates. From an implementation perspective, DL based object detection

algorithms can be categorized into two main approaches: Single-stage methods and Two-stage methods. Here is a brief detail about both these methods: -

2.3.4.1 Two-Stage OD Method

The evolution of two-stage object detection, also known as the region-based approach, has significantly affected vehicle detection and classification in recent years. This method, characterized by high localization accuracy but slower processing speed, involves generating candidate frames from a scene and then classifying and refining these proposals to enhance detection precision. Its basic architecture starts with R-CNN, proposed by Girshick et al. [79,80], which utilizes AlexNet as its backbone and employed selective search [81] for region proposals. R-CNN marked a substantial improvement over traditional object detection algorithms like HOG, Haar [84], and LBP [85]. However, its computational intensity during training prompted further refinements.

Fast R-CNN, introduced by Girshick [82], addressed some of these limitations by processing the entire image to generate convolutional feature maps and introducing Region of Interest (ROI) pooling layers. This approach streamlined the training process and increased efficiency, though it still relied on selective search for region proposals. [83] took the next step with Faster R-CNN, replacing selective search with a Region Proposal Network (RPN). This innovation, along with the use of anchor boxes at various scales and aspect ratios, significantly improved both detection speed and accuracy. The R-FCN architecture, proposed by Dai et al. [86], further refined the concept by addressing position sensitivity and variance issues. It introduced "position-sensitive score maps" and increased the sharing of convolutional parameters, leading to enhanced performance.

Comparative studies have shown the progressive improvements of these models. Wang et al. [89] showed that Faster R-CNN improved detection accuracy by 3.2% over Fast R-CNN on the COCO dataset [87]. Furthermore, R-FCN outperformed Faster R-CNN in both accuracy and processing speed on multiple datasets, including PASCAL VOC 07 [88]. These advancements in two-step object detection have played a crucial role in enhancing vehicle detection and classification capabilities. As autonomous driving technologies continue to evolve, these algorithms form the backbone of many advanced driving safety systems, pushing the boundaries of what's possible in computer vision for automotive applications.

2.3.4.2 Single-Stage OD Method

The evolution of single-stage OD algorithms has revolutionized real-time vehicle detection and recognition. These methods, unlike their two-stage counterparts, drop the region proposal phase, directly obtaining prediction results from the input image. This approach has led to significant improvements in both speed and accuracy, crucial for autonomous driving applications.

Both Single Shot Multibox detector (SSD) and YOLO (You Only Look Once) are the major families of single stage OD techniques. SSD have gained significant traction in vehicle collision avoidance applications due to their real-time processing capabilities and efficient detection performance. Unlike two-stage detectors that first propose regions and then classify objects, single-stage detectors streamline the process by directly predicting bounding boxes and class probabilities in a single network pass. SSD stands out in this domain because of its balance between speed and accuracy, making it particularly suitable for time-sensitive applications like collision avoidance. It utilizes a series of convolutional layers to predict a fixed number of bounding boxes and their associated scores for multiple object categories, enabling the rapid detection of vehicles, pedestrians, and other obstacles. This efficiency is crucial for automotive systems, where timely detection and response can significantly reduce the likelihood of accidents. Additionally, SSD's ability to detect objects at different scales through its multi-scale feature maps enhances its robustness in varying traffic scenarios. As autonomous driving technology advances, the integration of SSD into vehicle systems exemplifies how modern deep learning techniques can contribute to safer roadways by enabling quicker and more accurate obstacle detection.

The YOLO family of detectors has been at the forefront of this evolution. YOLOv2 addressed limitations of its predecessor by introducing batch normalization, high-resolution classifiers, and multi-scale training. It employs a high-resolution classifier backbone, maximizing input resolution to 448x448, and uses convolution anchor boxes to improve region proposals. The introduction of K-means clustering for anchor box sizing and multi-scale training further enhanced its performance across various object sizes.

YOLOv3 built upon these improvements, utilizing the DarkNet53 model for feature extraction and employing multi-label classification with overlapping patterns. It's particularly notable for object detection in complex scenes, using three feature maps of multiple sizes for bounding box prediction. YOLOv4 represents a significant leap forward, combining the strengths of its

predecessors to achieve an optimal balance of accuracy and speed. It introduces a three-part structure: "Neck," "Backbone," and "Prediction." The neck, composed of SPPNet and PANet, enhances feature fusion and compression. The CSPDarkNet53 backbone extracts features, which are then processed through the prediction scheme and filtered using Non-maximal Suppression (NMS). YOLOv5, further refines this approach. It uses CSPDarkNet as its backbone, offering improved small object detection, higher accuracy, and faster processing. The model employs Bottleneck CSP instead of residual shortcut links to enhance image feature description. Its neck system produces feature pyramids, enabling the network to detect objects of various sizes more effectively.

YOLOv6 [185] offers a good balance between speed (frames per second) and accuracy (mAP) compared to previous versions. This makes it ideal for real-time applications. It introduces new features like Bi-directional Concatenation (BiC) module. YOLOv6 adopted an anchor-free detector that enhances performance without sacrificing speed significantly. YOLOv8 was released in January 2023 by Ultralytics, the company that developed YOLOv5. YOLOv8 is an anchor-free architecture, reducing the number of box predictions and speeding up the Non-maximum suppression (NMS). In addition, YOLOv8 uses mosaic augmentation during training. Evaluated on MS COCO dataset test-dev 2017, YOLOv8x achieved an AP of 53.9% with an image size of 640 pixels (compared to 50.7% of YOLOv5 on the same input size) with a speed of 280 FPS on an NVIDIA A100 and TensorRT.

These advancements have led to wide-ranging applications of CNN-based object detectors, from face mask recognition to vehicle classification, pedestrian detection, and even medical image classification. Recent studies have demonstrated the effectiveness of both single and two-step detectors in vehicle detection and classification tasks. However, it's crucial to understand the strengths and limitations of these algorithms. Detection and classification performance can be affected by various factors, and ongoing research aims to minimize errors in object class prediction and improve overall algorithm performance. As the field continues to evolve, we can expect further refinements and innovations in single-step object detection, pushing the boundaries of what's possible in computer vision for automotive applications and beyond.

2.4 Embedded System for Autonomous Vehicles

The evolution of embedded systems in autonomous vehicles represents a pivotal advancement in automotive technology, marking a transition from basic engine control to sophisticated, AI-

driven decision-making systems. This progression has been instrumental in shaping the landscape of modern autonomous driving. The journey began with the introduction of Engine Control Units (ECUs), which revolutionized the automotive industry by providing basic control over engine, transmission, and other critical systems. As technology advanced, the integration of microcontrollers and sensors led to more sophisticated functionalities, enhancing vehicle performance and safety.

Today's embedded systems in autonomous vehicles serve as the technological nerve centre, enabling perception, decision-making, and control. These systems comprise both hardware and software components working in harmony. Sensors like LiDAR, radar, and cameras capture real-time environmental data, while actuators translate decisions into actions. The software interprets sensor data, processes it, and generates control commands, often utilizing AI and machine learning algorithms for complex decision-making in real-time. The efficacy of these systems lies in their ability to process information in real-time, requiring a symbiotic relationship between high-performance hardware and efficient software. This constant exchange of information between hardware and software components ensures a continuous feedback loop for autonomous decision-making.

Recent technological advancements have further propelled the field. The migration from single core to multi-core architectures in engine ECU software has necessitated new methodologies. Multi-core processors now enable high computing performance with low thermal dissipation, optimizing task-intensive real-time applications. Open-source-based peripherals for automotive ECUs have also emerged as educational platforms. Safety and security have become paramount concerns, leading to the implementation of systematic methodologies for functional testing of automotive embedded software. Simulation-based testing and inspection of engine control units during manufacturing have contributed to improved quality and reliability.

However, the increasing connectivity of vehicles has raised new security concerns. As the industry continues to evolve, addressing these challenges while integrating new technologies remains a priority. This progression has significantly enhanced vehicle functionalities, performance, and safety, paving the way for the future of autonomous driving. As technology continues to advance, further refinements in these foundational concepts will play a crucial role in shaping the automotive landscape, fostering safety, efficiency, and transformative user experiences

2.4.1 Types of Embedded Systems used in Vehicles

The Embedded systems in automotive vehicles have become integral to enhancing performance, safety, and user experience. These systems can be broadly categorized into three main types: engine and transmission control systems, in-car entertainment and infotainment systems, and ADAS.

Engine and transmission control systems play a crucial role in optimizing fuel efficiency and overall vehicle performance. These sophisticated systems are designed to fine-tune engine speed, manage transmission gears, and balance workload across various driving conditions. The automotive industry has recognized the importance of safety in these systems, leading to the integration of safety analysis tools into the model-based development toolchain for embedded systems.

In-car entertainment and infotainment systems have evolved significantly, focusing on connectivity features and multimedia integration. Modern vehicles now incorporate phone-car connected systems and In-Vehicle Infotainment (IVI) systems, which have been shown to positively influence user adoption through improved facilitating conditions and technographics. However, the increasing complexity of these systems, particularly Android-based infotainment apps, has raised concerns about security vulnerabilities that need to be addressed.

Advanced Driver Assistance Systems (ADAS) represent a significant leap in automotive technology, evolving from basic safety features to autonomous driving capabilities. ADAS has played a crucial role in enhancing vehicle safety, as evidenced by the development of integrated engine-hydro-mechanical transmission control algorithms for tractors. These algorithms utilize artificial intelligence to adapt engine speed and improve overall performance. To ensure the reliability and safety of ADAS and other embedded systems, researchers have developed integrated virtual execution platforms for large-scale distributed embedded systems, facilitating thorough validation processes.

2.4.2 Latest Trends and Role of Embedded Systems for VCA

The field of embedded systems in autonomous vehicles is rapidly evolving, driven by three key trends: the increasing integration of artificial intelligence (AI) and machine learning, the advancement of connectivity and edge computing, and the development of adaptive and learning capabilities. AI and machine learning are revolutionizing embedded systems in autonomous vehicles. These technologies enable sophisticated sensor fusion, combining data

from LiDAR, radar, and cameras to create a comprehensive understanding of the vehicle's surroundings. Machine learning models are enhancing path planning and decision-making, allowing vehicles to navigate complex environments more efficiently. Deep learning techniques, particularly CNN, are improving object recognition, enabling vehicles to accurately identify and classify objects in their vicinity. Additionally, predictive analytics powered by machine learning are helping vehicles anticipate the behavior of other road users, enhancing safety and efficiency.

Controllers are the brain of collision avoidance systems, executing the necessary computations to process sensor data and make real-time decisions. The most common controllers used in these systems are microcontrollers (MCUs) and digital signal processors (DSPs). MCUs, such as those from the STM32 family by STMicroelectronics, are popular due to their low power consumption, affordability, and adequate processing power for handling basic collision avoidance tasks. These controllers are often integrated with various peripherals and communication interfaces, making them suitable for automotive applications. On the other hand, DSPs are designed for high-performance real-time processing, making them ideal for more complex tasks such as image and signal processing required in advanced collision avoidance systems. The TMS320C6000 series by Texas Instruments is a notable example, providing robust performance and flexibility. These controllers can efficiently handle tasks such as object detection and classification, lane departure warning, and adaptive cruise control by processing data from cameras, radar, and LiDAR sensors.

GPUs have become indispensable in the development of sophisticated collision avoidance systems. Their parallel processing capabilities allow them to handle the immense computational load required for tasks such as image recognition, sensor fusion, and path planning. Nvidia GPUs, in particular, are widely used in automotive applications due to their superior performance and support for AI frameworks. The Nvidia Jetson platform has revolutionized embedded systems in automotive applications, particularly in collision avoidance systems. The Jetson platform, including models like Jetson Nano, Jetson TX2, Jetson Xavier and Jetso Orin nano provides unparalleled computational power for DL and AI applications in a compact form factor. These platforms leverage Nvidia's GPU technology, enabling the deployment of complex neural networks for real-time object detection, tracking, and decision-making. The Jetson TX2, for instance, is equipped with a 256-core Pascal GPU and a powerful ARM Cortex-A57 CPU, providing the necessary horsepower to process high-resolution images and perform deep learning inference at the edge. This capability is crucial

for collision avoidance systems that require quick and accurate responses to dynamic driving environments. Jetson Xavier, with its Volta GPU architecture and integrated tensor cores, takes it a step further by supporting more advanced AI models and providing higher throughput and efficiency. Nvidia's software stack, including the Jetpack SDK and TensorRT, further enhances the capabilities of the Jetson platform. These tools simplify the development and deployment of AI models, ensuring that developers can optimize their applications for real-time performance and low latency, which are critical for collision avoidance systems. The Jetson platform's versatility and scalability make it a preferred choice for automotive manufacturers aiming to integrate cutting-edge AI capabilities into their vehicles.

The role of GPUs extends beyond mere computation. They also facilitate the development and training of AI models. With platforms like Nvidia's CUDA and cuDNN, developers can leverage GPU acceleration to train complex models faster, reducing the time to market for advanced collision avoidance systems. Moreover, the flexibility of GPUs allows for the integration of new AI models and algorithms, ensuring that the collision avoidance system can evolve and improve over time.

FPGAs are increasingly being used in collision avoidance systems due to their flexibility, low latency, and ability to handle parallel processing tasks efficiently. Unlike fixed-function ASICs, FPGAs can be reprogrammed to adapt to new requirements and algorithms, making them ideal for rapidly evolving fields like autonomous driving and collision avoidance. One of the key advantages of FPGAs is their ability to process data in real-time with minimal latency. This is crucial for collision avoidance systems that need to react instantly to dynamic driving conditions. FPGAs can be programmed to perform specific tasks such as sensor fusion, object detection, and path planning with high efficiency. For example, the Xilinx Zynq Ultra Scale + MPSoC combines programmable logic with ARM Cortex-A53 processors, providing a powerful platform for developing collision avoidance systems. They can handle tasks such as image processing, radar signal processing, and data aggregation simultaneously, ensuring that the collision avoidance system has a comprehensive understanding of the vehicle's surroundings. Moreover, the reconfigurability of FPGAs allows for continuous optimization and integration of new features, ensuring that the system remains up-to-date with the latest advancements in AI and sensor technology.

2.5 Threat Assessment Approaches

2.5.1 Logic Based Approaches

Logic-based methods are being used to make sure self-driving car systems are safe. Instead of writing complicated rules, they turn safety requirements into logical statements that computers can check. One team created a system for safe highway driving where cars use adaptive cruise control. They proved it was safe using special mathematical models. Their proof was cleverly broken into parts, making it easier to check each piece separately. Other researchers used something called Multi-Lane Spatial Logic to prove safety on different types of roads. This method separates thinking about space from thinking about how cars move. It's like making sure certain spots on the road are always empty to avoid crashes. Logic-based threat assessment often uses Boolean algebra and operators like AND (\wedge), OR (\vee) etc.

Another study looked at how cars can work together safely. They used math to check if cars could complete driving tasks without crashing. Some researchers [20] are even using advanced logic to turn traffic rules into a language computer can understand and follow. All these approaches are different ways of using math and logic to make sure self-driving cars will be safe on the road.

2.5.2 Set Based Approaches

In contrast Set-based approaches focus on specifying acceptable or unacceptable behaviors or system configurations. One group created a system for intersections where a central computer assigns time slots to cars. The cars then figure out if they can safely cross in that time. Another team predicts all the places a car and other vehicles might be, accounting for things like inaccurate sensors. They tested this on real self-driving cars. Some researchers developed a method to check for potential collisions on various road types. They look at where the main car and other objects might be, and if these areas overlap, it could mean a crash. A tool called SPOT was created to predict where other cars might go. It considers all possible moves, physical limits, and traffic rules to help plan safe routes. Recently, researchers came up with a way to calculate how much time a car has to react in different traffic situations.

All these methods aim to make self-driving cars safer by predicting and avoiding potential dangers on the road. Using similar concepts, tackled the tricky problem of what happens when other drivers do unexpected things. They came up with a clever system that splits a car's path into safe zones and risky areas. They called the last safe point the "Point of No Return" and the

first completely safe point the "Point of Guaranteed Arrival." This helps the car figure out which parts of its journey might be dangerous. They also created a way to give each possible path a safety score. By using this score in their calculations, the car can choose the safest route possible, even thinking far ahead into the future. They tested this idea using SPOT, checking how well it worked when a self-driving car tries to pass another vehicle on a two-way road. Set theory can be used to model threat landscapes. Basic operations include Union (\cup), Intersection (\cap) and Complement ($'$). If R , T and V are risk, threat and vulnerability set respectively, we can define the risk as the intersection of threats and vulnerabilities.

2.5.3 Probability Based Approaches

Probabilistic threat-assessment (TA) methods use system uncertainties to make decisions with confidence. These methods assign probabilities to events, like the likelihood of a collision given certain uncertainties. Probabilistic TA assigns "how likely" answers to events, like an autonomous car potentially colliding soon. Uncertainties like imperfect vehicle models, sensor noise, and driver intent make this crucial in self-driving cars.

Drivers have countless options on the road, making their actions difficult to predict. However, these actions can often be categorized into a finite set of common maneuvers, like lane changes or overtaking. They have reduced computational complexity by linking driver actions to high-level maneuvers and using Monte Carlo simulations to compute collision probabilities. This method was improved by considering the driver's awareness of other objects and further refined in with a better vehicle model. It was introduced a curved coordinate system to simplify modelling on curved roads, while assessed risk using probabilistic Time-to-Collision (TTC) levels. A probabilistic approach using a Markov chain abstraction was proposed for predicting traffic participant occupancy, extended in by comparing Markov chains and Monte Carlo simulations. An algorithm combining an Unscented Kalman Filter with reachability analysis for emergency interventions was presented. Bayesian approaches, like those in [50] and [51], used Dynamic Bayesian Networks (DBNs) to compute collision risks, further developed in with a Partially Observable Markov Decision Process (POMDP) to model driver behavior combined network-level and vehicle-level collision predictions using a DBN, while used Bayesian Occupancy Filtering to estimate future occupancy, addressing occluded objects with prior map knowledge. Computational efficiency was addressed in by mixing set theoretical and probabilistic methods, dividing the threat-assessment problem into preliminary and specialized parts.

For probabilistic decision-making, some have use hypothesis testing to derive decision rules for automated braking, generalized in for any stochastic TA algorithm. A two-level threat-assessment approach was discussed in and, focusing on physical system threats and driver perception. Finally, it was focused on behavior generation for automated vehicles using a POMDP algorithm to handle uncertainties at intersections.

Chapter 3

Methodology

3.1 Introduction

The Conventional vision-based vehicle collision prediction and avoidance systems typically rely on a single monocular camera paired with a deep learning (DL)-based object detection (OD) model. However, such systems often fall short in either accuracy or real-time performance, limiting their effectiveness in dynamic and fast-paced driving environments. Most of these systems operate by generating passive alerts based solely on the detected object's displacement within the spatio-temporal domain, which involves tracking the object's position and movement over time.

This approach, while functional, fails to account for multiple other contributing factors that influence collision prediction. For instance, it does not consider environmental complexities, varying lighting conditions, or the interplay between multiple objects in the scene, which are critical for robust and reliable decision-making. As a result, the system's predictive capabilities are restricted, often leading to delays or inaccuracies in generating alerts or initiating preventive actions.

Moreover, the use of a single camera limits the field of view (FOV), potentially causing blind spots that reduce the system's ability to detect and respond to threats in time. Such limitations make traditional systems less suited for real-world applications where split-second decisions are crucial for ensuring vehicle safety and collision avoidance.

3.2 Working Principle

The proposed Vehicle Collision Avoidance System (V-CAS) integrates multiple advanced components to achieve robust, real-time performance for vehicle safety. An overview of this system, along with its key building blocks and their integration, is provided here. The block diagram of the V-CAS architecture is illustrated in Figure 3.1, showcasing how various modules are combined into a cohesive system.

To enhance the interpretation of real-world objects, an array of three cameras was employed, providing a wide field of view (FOV). This setup enables the system to capture a more comprehensive scene, critical for detecting potential hazards. At the core of the detection pipeline is the state-of-the-art (SOTA) real-time object detector RT-DETR, capable of identifying moving or potentially moving objects such as vehicles, pedestrians, and other relevant entities. For tracking, the system employs DeepSORT, a robust tracking algorithm that combines the predictive capabilities of the Kalman filter with the strengths of deep learning. This hybrid approach ensures reliable tracking of objects even in dynamic environments.

By analyzing the tracked objects' positions, speeds, and rates of acceleration, the system calculates a collision score for each detected entity. These computations are performed on the NVIDIA Jetson Orin Nano, a compact yet powerful embedded platform optimized for edge AI applications. When the predicted collision score of any object surpasses a predefined threshold, a braking signal is generated. This signal is transmitted through the Jetson device's 40-pin expansion header to the vehicle's adaptive braking mechanism, enabling timely and proportional application of brakes based on the collision risk.

To further enhance collision prediction capabilities, a supplementary method was integrated into V-CAS. This method focuses on detecting the brake lights of frontal vehicles, providing an additional layer of safety. The system interprets the activation of brake lights as an indication that the vehicle ahead is decelerating or coming to a halt, which could lead to a collision if not addressed. This brake light detection mechanism proves especially valuable at night, where visibility is reduced, and conventional object detection may falter. Even if a vehicle remains undetected by the primary object detector, its brake lights are likely to be identified, triggering a cautionary response.

Moreover, if the detected brake lights are in close proximity to the host vehicle, the system bypasses the standard speed estimation process and directly initiates emergency braking. This dual-layer approach ensures that the system remains effective in a wide range of conditions,

from daylight to challenging nighttime scenarios, thereby significantly improving vehicle safety and collision avoidance.

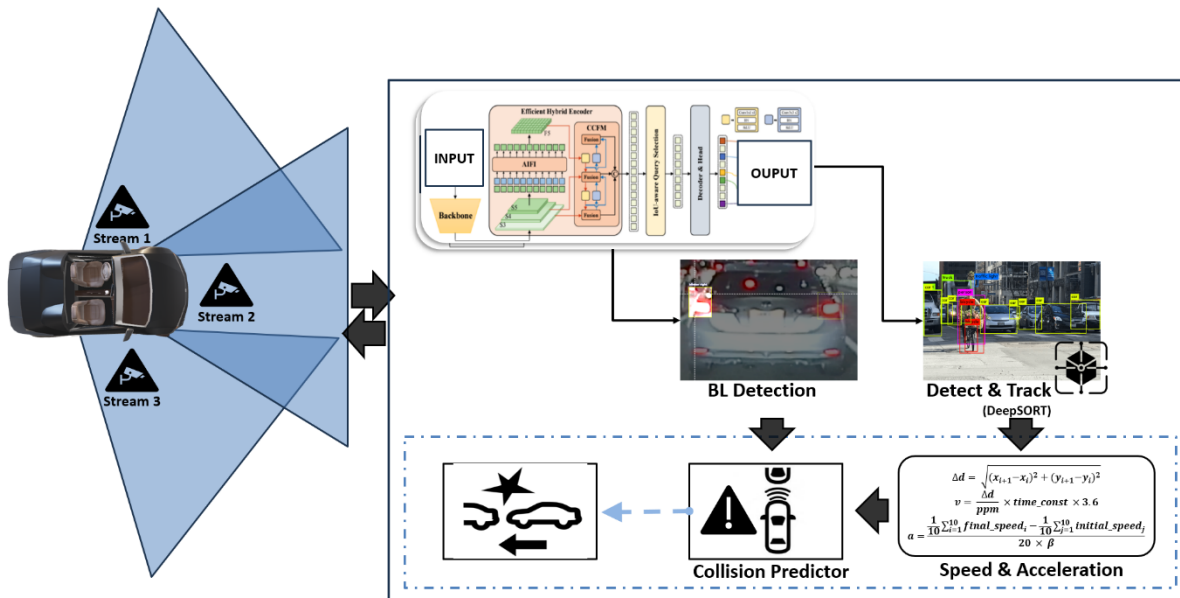


Figure 3.1 Basic architecture of VCAS

3.2.1 Object Detection – RT-DETR

Object detectors are broadly classified into two major categories, each with distinct approaches and characteristics: 1) **Two-Stage Object Detectors**. These detectors operate by first generating candidate regions for potential objects through a region proposal network (RPN). Each of these regions is then individually classified to achieve precise object detection. While this approach results in higher accuracy, it comes at the cost of slower processing speeds, making it less suitable for real-time applications. Common examples of two-stage algorithms include Faster R-CNN and R-FCN, both of which are widely recognized for their precision in complex object detection tasks. 2) **One-Stage Object Detectors**. In contrast, one-stage detectors directly predict bounding boxes and class probabilities in a single step, bypassing the region proposal process. This design prioritizes speed over accuracy, enabling faster processing but with a trade-off in detection precision. These models are often preferred in scenarios where real-time performance is critical, such as autonomous systems and video surveillance.

The backbone of our Vehicle Collision Avoidance System (V-CAS) integrates the strengths of modern object detection advancements by utilizing a vision-based transformer model, specifically RT-DETR (Real-Time Detection Transformer). This model, a competitor to traditional single-stage object detection methods, has been pre-trained on the COCO dataset. Through transfer learning, the architecture is tailored to include only the necessary parameters and layers for classifying desired classes, effectively optimizing the model for our use case.

The main architecture of RT-DETR is shown in figure 3.2. It is designed to balance speed and accuracy for real-time object detection tasks. At its core, the model integrates a transformer-based backbone with a lightweight encoder-decoder structure. The backbone extracts essential features from input images, efficiently processing visual data using multi-scale feature maps. These features are then fed into a transformer encoder that enhances contextual understanding by modeling long-range dependencies across spatial and channel dimensions. The decoder, equipped with query embeddings, refines these features to generate precise object predictions, including bounding box coordinates and class labels. RT-DETR also incorporates a one-to-many assignment strategy, which links ground truth objects to multiple predictions, ensuring more robust training and reducing false negatives. Unlike traditional object detection models, which often rely on region proposals or anchor-based methods, RT-DETR eliminates these components for a streamlined approach, improving processing speed without compromising detection quality. Its architecture is further optimized by leveraging parallel computing and dynamic attention mechanisms, enabling efficient inference on both high-performance GPUs and resource-constrained embedded platforms. These design choices make RT-DETR a suitable choice for deployment in scenarios requiring rapid and accurate detection, such as autonomous driving and real-time surveillance systems.

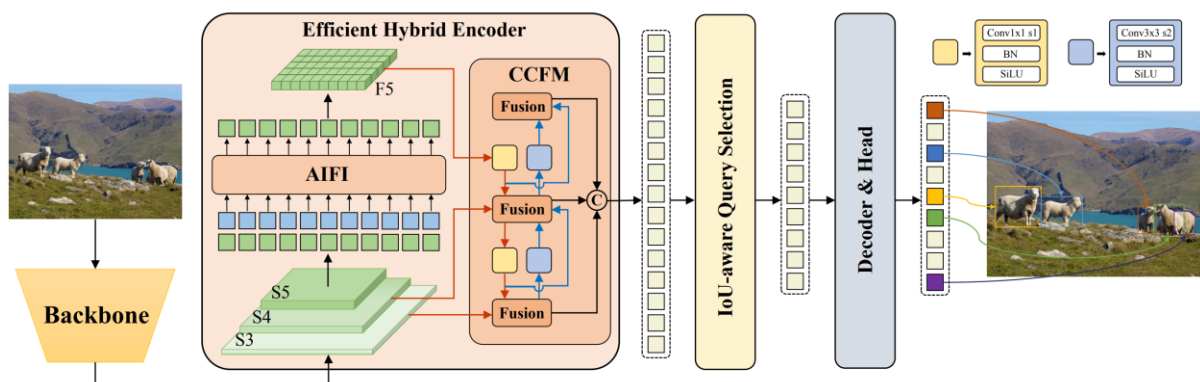


Figure 3.3 Architecture of RT-DETR, cited from [40]

This approach offers a balanced trade-off between speed and accuracy, making it highly suitable for deployment in V-CAS. By leveraging RT-DETR's capabilities, the system achieves real-time performance without significant compromises on precision, ensuring reliable object detection and collision prediction in dynamic driving environments.

3.2.2 Object Tracking – DeepSORT

We incorporated **DeepSORT**, one of the most effective real-time multi-object tracking algorithms, into our system. DeepSORT leverages the strengths of deep learning for feature extraction and combines them with the predictive power of a classic Kalman filter for robust data association. This hybrid approach ensures precise and efficient tracking of multiple objects, even in challenging scenarios involving occlusions or missed detections. DeepSORT operates through two primary modules:

Deep Appearance Descriptor: This module utilizes a pre-trained deep convolutional neural network (CNN) to extract high-level features from cropped object images in each video frame. These features represent the unique characteristics of objects, enabling the system to differentiate between them, even when they appear similar or move across frames.

Kalman Filter and Hungarian Algorithm: The Kalman filter is employed to predict the state of each detected object across consecutive frames, providing estimates for object locations and velocities. This predictive capability is crucial for maintaining object tracks during occlusions or temporary missed detections. The Hungarian algorithm, on the other hand, is used to associate detections in the current frame with existing tracks or initiate new tracks. This association is performed based on the similarity between the predicted states (from the Kalman filter) and the current detections, often measured using the Mahalanobis distance.

This robust combination allows DeepSORT to maintain high accuracy and reliability in multi-object tracking. Figure 3.3 illustrates the fundamental architecture of DeepSORT, highlighting the interaction between the CNN-based feature extraction, the Kalman filter, and the data association mechanisms.

By integrating DeepSORT into our system, we ensured seamless and precise tracking of multiple objects, a critical requirement for real-time vehicle collision avoidance and monitoring in dynamic driving environments. This methodology enhances the system's ability to predict and respond to potential collisions effectively.

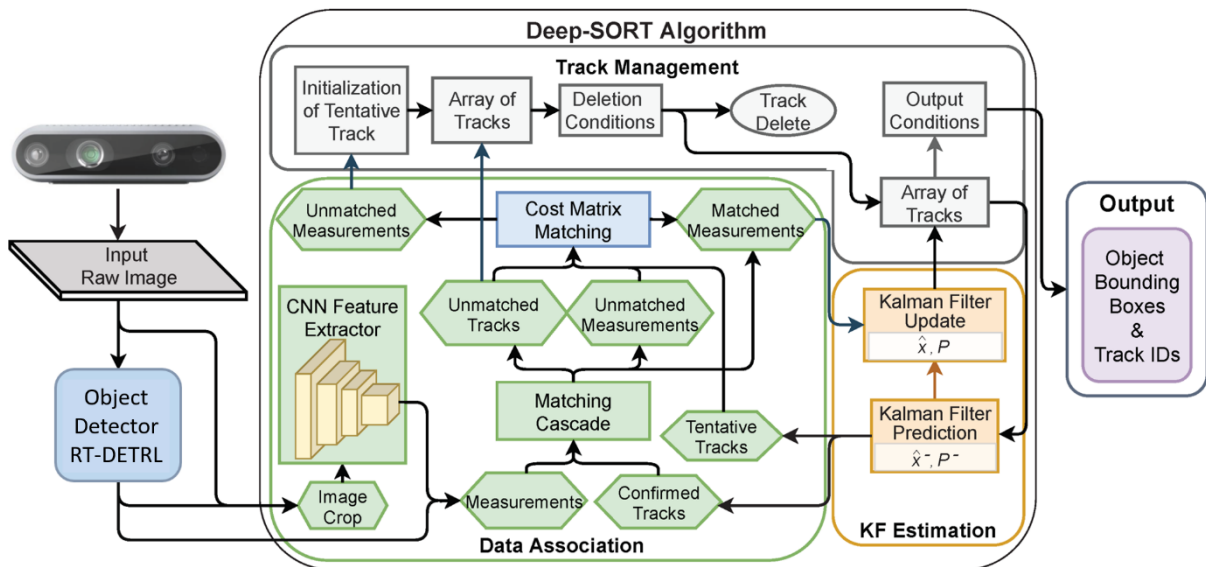


Figure 3.3 Architecture of DeepSORT, cited from [94]

3.2.3 Integrating DeepSORT with RT-DETR and PyTorch

DeepSORT integration with RT-DETR needs some output conversions because RT-DETR delivers output values as top left and right bottom coordinate values of the bounding box while DeepSORT expects inputs in the form of center coordinates (C_x, C_y), width and height of the bounding box as shown in Figure 3.4. A separate function was defined to convert the bounding box output values coming from RT-DETR into the desired input format of DeepSORT.

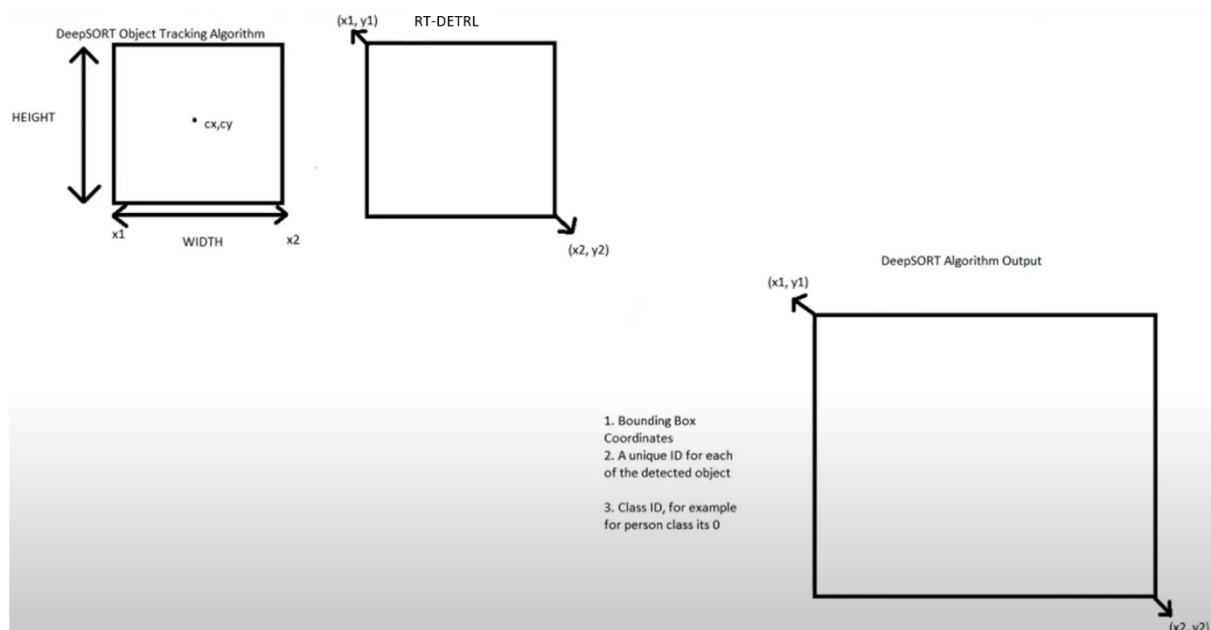


Figure 3.4 Input and output matching of DeepSORT with RT-DETR

The original Deep SORT implementation was based on TensorFlow [94]. But we have more interest in PyTorch due to its fast iterations speed, flexibility and more pythonic approach. So, we have to find a version of DeepSORT that supports implementation with PyTorch.

3.2.4 Speed Estimation

To estimate the speed of detected and tracked objects, the system calculates the Euclidean distance between the object's positions in consecutive frames using the distance formula in two-dimensional space (Equation 1). Traveling distance of the vehicle between these two frames called pixel displacement (Δd) is represented by:

$$\Delta d = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (1)$$

where (x_i, y_i) and (x_{i+1}, y_{i+1}) are the horizontal and vertical pixel position of the target vehicle on frame i and $i + 1$ respectively. Pixel displacement is then converted to real-world meters using pre-calibrated pixel-per-meter (ppm), which was 20 in our case. Finally, the function calculates the speed (Equation 2) by dividing (Δd) with ppm and multiplying it by the time constant (1/FPS) and 3.6 (total seconds in an hour / 1000) to get our speed v in km/h which is represented mathematically as:

$$v = \frac{\Delta d}{ppm} \times time_const \times 3.6 \quad (2)$$



Figure 3.5 Simulated view of velocity from car dash camera view

Figure 3.5 shows a simulated view of how the change in positioning of an object (vehicle) in consecutive frames helps in finding the velocity / speed estimation of the target object and how it looks like from drivers dash mounted camera point of view.

3.2.5 Calculating Relative Rate of Acceleration

The relative rate of acceleration (Equation 3) for each tracked object which crosses the invisible grid around the subject vehicle is calculated to assess collision risk. A queue of 20 speed values is maintained for each object, divided into initial and final buffers of 10 values each. The relative acceleration is then divided by a variable β which was 0.0625 in our case (1/16fps), shown mathematically as:

$$a = \frac{\frac{1}{10} \sum_{i=1}^{10} final_speed_i - \frac{1}{10} \sum_{j=1}^{10} initial_speed_j}{20 \times \beta} \quad (3)$$

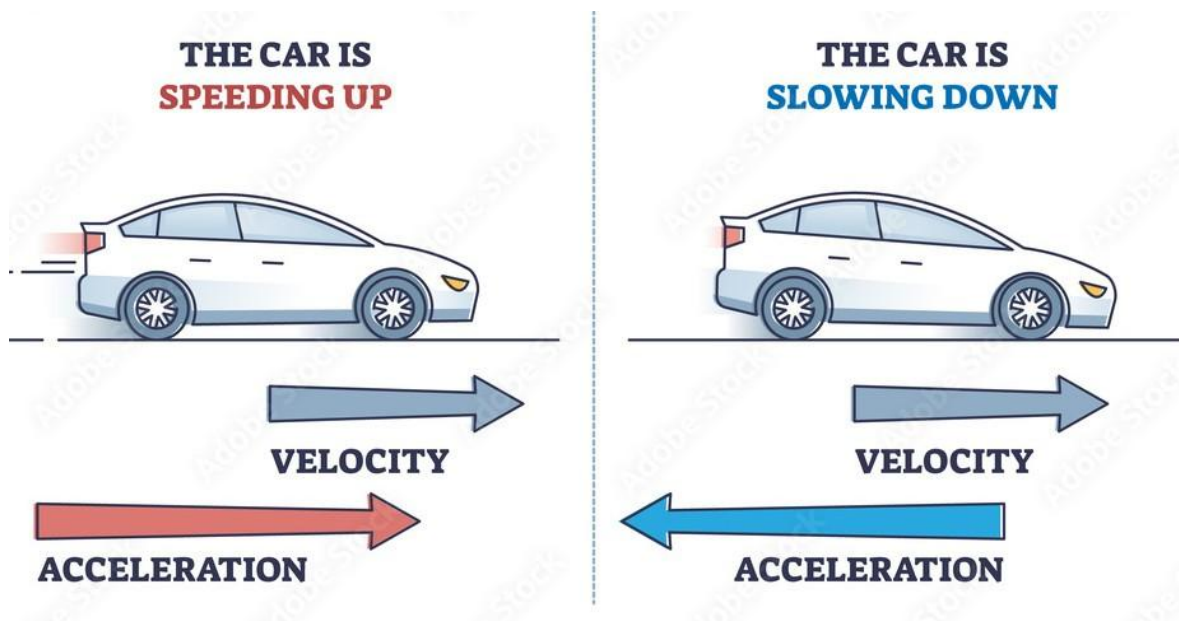


Figure 3.6 Schematic view of relative acceleration of vehicles

Figure 3.6 shows how the increase and decrease of velocities of two vehicles in same trajectory impacts their relative rate of acceleration and deceleration. If the frontal vehicle is also going in same or more speed then the host vehicle, their will be 0 or negative relative acceleration. However, if the frontal vehicle is slowing down, the host vehicle's relative acceleration will increase showing a chance of collision.

3.2.6 Brake Light Detection

The prediction of potential collisions through speed estimation is a relatively straightforward approach but presents several critical limitations that can impact its reliability in real-world scenarios:

Limitations in Pixel-Based Collision Prediction The approach relies heavily on the calculation of collision prediction based on the relative displacement of a single pixel. While this method simplifies the computation process, it lacks robustness in certain situations. Small inaccuracies or noise in pixel displacement measurements can lead to erroneous predictions, reducing the reliability of the system. Furthermore, environmental factors such as motion blur, lens distortion, or sensor noise can exacerbate the problem, making the approach insufficient in handling complex real-world scenarios.

Challenges in Low-Light Conditions. The system's reliance on object detection poses a significant issue, particularly in low-light or nighttime conditions. Failure to detect the objects of interest—such as vehicles or pedestrians—can result in the complete absence of collision predictions. This limitation underscores the need for a more reliable mechanism that can complement or replace speed estimation-based predictions under such adverse conditions.

To address these challenges, another widely adopted method was integrated into the Vehicle Collision Avoidance System (V-CAS): predicting collisions by detecting the brake lights of the frontal vehicle. This additional mechanism provides an enhanced layer of safety and addresses several shortcomings of speed estimation.

Brake Light Detection for Enhanced Collision Prediction. The detection of brake lights allows the system to anticipate a potential collision more effectively, even in scenarios where object detection may fail. For instance, at night, when vehicles might remain undetected due to poor lighting conditions or occlusion, brake lights can still be reliably identified. This approach leverages the higher visibility and distinctiveness of brake lights in low-light environments, providing an additional safety net for the system.

Emergency Braking in Close Proximity. Moreover, the integration of brake light detection bypasses the dependence on speed estimation calculations in critical situations. If the detected brake light is in close proximity to the host vehicle, the system immediately issues a warning and applies emergency brakes, preventing potential collisions. This proactive measure ensures that the system remains responsive and effective, even when traditional speed estimation calculations might not suffice.

By incorporating brake light detection into the collision prediction mechanism, V-CAS achieves a more robust and comprehensive approach to collision avoidance. This dual-layered strategy ensures that the system remains effective across a wide range of scenarios, enhancing safety for both the driver and other road users.



Figure 3.7 Brake light ON / OFF detection

Figure 3.7 is a depiction of real-world scenario in which vehicles brake lights are being detected as ON or OFF and bounding boxes with different colors are drawn respectively.

3.2.7 Fusion of Multiple Camera Sensors

There are few known video pipelines and SDK like Nvidia DeepStream [91] and GStreamer [92] for real-time multiple video / camera stream fusion, however, they are very difficult to integrate and not flexible enough to be used / modified for custom application easily. Therefore, a more simplistic and flexible approach was adopted using OpenCV, Numpy and multi-threading. The 'vSstream' class is designed to handle individual video streams, continuously

capturing frames from different sources in separate threads for efficient processing. These frames are resized to a uniform dimension and stored in a thread-safe manner. Then frames from all camera sources are read simultaneously, and if available, they are combined into a single frame using horizontal stacking provided by NumPy. This combined frame is then passed to an OD model, which identifies objects and returns their bounding boxes, confidence scores, and class IDs. The detections are subsequently processed by single tracker to maintain consistent identities of objects across frames. Finally, bounding boxes and labels are drawn on the combined frame to indicate tracked objects, and the frame is displayed using OpenCV. This approach enables real-time fusion and processing of multiple video streams in an efficient and simple way, facilitating tasks such as object detection and tracking across a unified video feed.

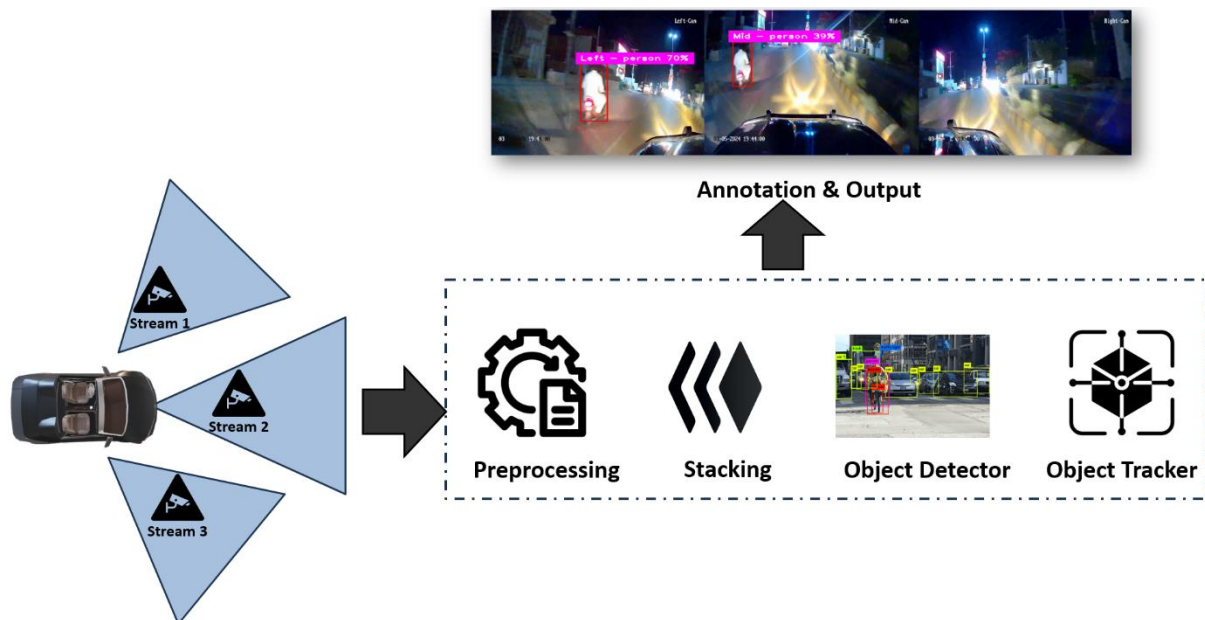


Figure 3.8 Workflow of multi-camera stream fusion

Figure 3.8 shows a workflow diagram of steps involved in fusion of multiple cameras streams so that a single OD and tracker can be applied on the combined window of concatenated window of all cameras. It is also pertinent to mention that, to keep the integrity of original camera streams, an identifier is being assigned to each camera stream depending upon the spatial coordinated of the combined window. In this case, any object detected is identifiable to which camera stream it belongs to.

Another main pre-processing step is to resize the input camera streams so that its individual camera resolution is resized in a way that total resolution / number of pixels of a single camera stream remains unchanged after concatenation of three camera inputs. While re-sizing of each input resolution, it is paramount to keep the aspect ratio either unchanged or preserve according to actual resolution of our dataset resolution on which our OD model was initially trained. We have, resized each stream to 640 x 640 size and stacked them horizontally, making a combined resolution of $(640 \times 3) \times 640 = 12,28,800$ pixels. Each camera individual stream was 960P $(1280 \times 960) = 12,28,800$ pixels. This overall pixel resolution matches our collision performance dataset each video resolution (1280×960) . In this way we have obtained same number of frames per second (FPS) on combined window of real-world cameras input as of individual video of dataset. Reason is that our OD model and tracker has to process same number of overall pixels in both cases. This can be illustrated in figure 3.9. One of the function code blocks of our vStream class snippet is also given below.

Code Snippet

```
class vStream:
    def __init__(self, src, width, height, identifier):
        self.width = width
        self.height = height
        self.capture = cv2.VideoCapture(src)
        self.frame = None
        self.identifier = identifier
        self.lock = Lock()
        self.stopped = False
        self.thread = Thread(target=self.update, args=())
        self.thread.daemon = True
    self.thread.start()
```

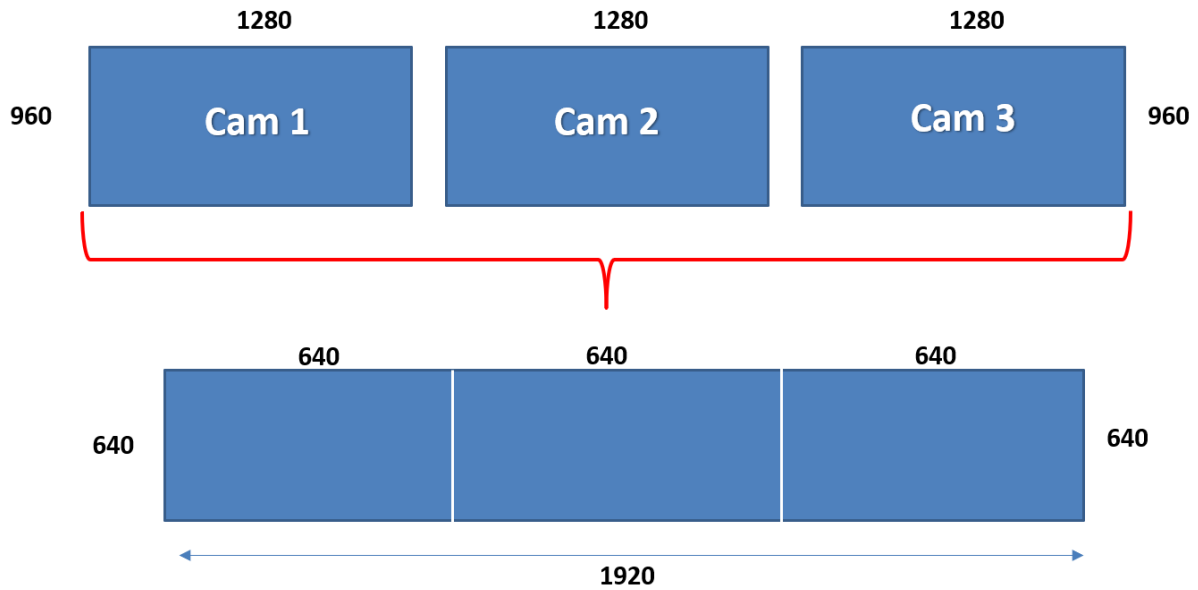



Figure 3.9 Horizontal stacking and resize of input streams

3.2.8 Calculating Collision Prediction Score

The relative rate of acceleration of the same object against its earlier values plays a pivotal role in our collision avoidance system. If it is increasing for detected object in the same trajectory as of our host vehicle, then it shows that our vehicle and the other object are closing with each other and vice versa. Depending upon these values, a confidence score was assigned. If it crosses a threshold (>60%), then an electric signal is generated from 40 pin Expansion Header of Jetson Orin Nano to the braking mechanism as pulse width modulation (pwm) signal. Where width of the pulse is proportional to the confidence score of collision prediction. Finally, it displays collision warnings on the monitor screen to the driver. Additionally, the custom trained Brake Light detection model also keeps on detecting the vehicles in the scene with brakes “ON” status. If any such vehicle comes into very near proximity of the host vehicle in the same line of trajectory or cross-sectional trajectory, it generates an emergency braking signal from our embedded device to the vehicle brake application system.

Chapter 4

Experimental Results

4.1 Datasets

4.1.1 For Object Detection

Two public datasets from Roboflow were utilized to fine-tune the pre-trained RT-DETR-L object detection model. These datasets were specifically chosen to cater to the diverse requirements of the target application and to ensure robust performance across various scenarios:

4.1.1.1 Vehicle i2 Dataset

This dataset comprises a total of 7,458 high-quality images, each with dimensions of 640 x 640 pixels. The dataset covers 25 distinct classes, representing an extensive range of vehicle types, including ambulances, buses, rickshaws, bicycles, and motorcycles. Additionally, the dataset encompasses other potential collision objects such as pedestrians, thereby making it highly versatile for real-world applications involving traffic and collision avoidance. As part of the preprocessing pipeline, auto-orientation of pixel data was performed with EXIF-orientation stripping, ensuring uniformity and consistency in the input data. This step was crucial for eliminating variations caused by camera orientation during image capture. The diverse set of classes and comprehensive coverage of vehicles make this dataset an invaluable resource for developing models capable of detecting a wide variety of traffic-related entities.

4.1.1.2 Brake Light Detection Dataset

This dataset contains an impressive 22,525 images, all uniformly resized to 640 x 640 pixels. It focuses on a binary classification task, distinguishing between "Brake Off" and "Brake On" states, making it highly relevant for applications in automotive safety and advanced driver-assistance systems (ADAS). The preprocessing pipeline included auto-orientation of pixel data with EXIF-orientation stripping to maintain data consistency. To further enhance model robustness, a series of data augmentation techniques were applied. These included:

1. A 50% probability of horizontal flipping to address variations in vehicle orientation.
2. Random cropping of between 0% and 20% of the image, introducing spatial variability.
3. Adjustments to brightness levels ranging from -25% to +25%, simulating different lighting conditions.
4. Application of Gaussian blur with a range of 0 to 1.5 pixels, mimicking potential lens distortions.
5. Addition of salt-and-pepper noise to 5% of the image pixels, creating resilience to sensor noise.

These augmentations not only increased the diversity of the training data but also prepared the model to handle real-world complexities effectively. The substantial size and targeted scope of this dataset make it highly suitable for fine-tuning models aimed at brake light detection, a critical feature in modern traffic safety systems.

4.1.2 For Collision Avoidance Evaluation

To date, as far as we know, no publicly available video dataset exists that provides multiple camera streams specifically for traffic data analysis and collision prediction. To address this gap, we utilized a combination of hybrid datasets to support our research. The first dataset is our own, recorded using three dash-mounted cameras installed in the same vehicle. This dataset spans over 10 hours of footage captured on highways and within city areas under a variety of conditions, including normal and rash driving scenarios during the day, cloudy weather, dusk, and night. The recordings cover vehicle speeds ranging from 10 to 120 km/hr, offering a diverse set of driving environments and behaviors.

The second dataset is the publicly available Car Crash Dataset (CCD) [93], specifically designed for traffic accident analysis. The CCD comprises real-world traffic accident videos sourced from YouTube channels. These videos have been carefully processed and split into 1,500 trimmed clips, with each video containing 50 frames recorded at 10 frames per second.

Additionally, 3,000 normal driving videos were included as a reference set, randomly sampled from the BDD100K dataset, which provides a wide range of non-accident driving scenarios.

By leveraging this combination of datasets, our approach ensures a comprehensive evaluation of traffic behavior and collision prediction under both controlled and real-world conditions. The inclusion of multi-camera recordings from our custom dataset and the diverse accident scenarios from CCD provides a unique dataset blend, enabling a more robust and realistic analysis of collision prediction and traffic data across varying contexts and driving situations.

4.2 System Resources and Training Setup

The training process for the Vehicle Collision Avoidance System (V-CAS) was executed on a high-performance PC powered by Nvidia RTX 4090 GPU, while inference tasks were carried out on the NVIDIA Jetson Orin Nano, an advanced embedded platform, using Ezviz H1C cameras. Python 3.10 was utilized as the core programming language for the implementation, with PyTorch serving as the deep learning framework due to its flexibility and efficiency in handling complex neural network operations.

The embedded system's specifications, as outlined in Table 4.1, emphasize the Jetson Orin Nano's critical role as the primary computational and monitoring resource. This embedded device is central to real-time operations within the V-CAS, enabling seamless processing and decision-making inside a moving vehicle. Numpy, a powerful numerical computing library, was employed for tasks such as horizontal stacking of multiple video streams, providing an efficient mechanism to manage and process video data.

In addition to hardware and programming specifics, Table 4.2 presents a detailed summary of the training hyperparameters and configurations tailored to the datasets used for fine-tuning the model. These datasets required precise tuning of parameters such as the optimizer, learning rate, momentum, and decay. The selection and adjustment of these parameters were carefully conducted to ensure they aligned with the dataset characteristics, including size and class distribution. This meticulous tuning was instrumental in achieving optimal model performance, allowing the system to handle the complexities of object detection and collision prediction effectively in diverse real-world scenarios. Through this integrated approach, the training and inference pipeline was optimized to support robust, real-time functionality for V-CAS.

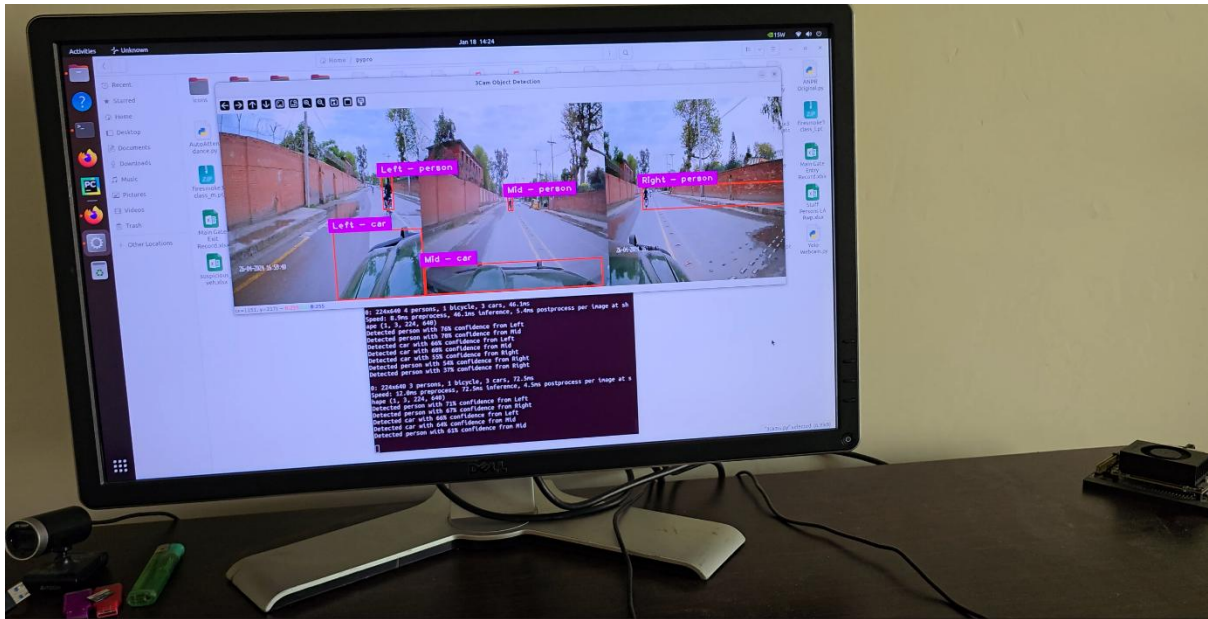


Figure 4.1 Experimental Setup

Figure 4.1 shows the experimental setup of multi camera dataset being inferred using Jetson Orin Nano in real-time.

Table 4. 1 System resources for training and inference

Training (Nvidia GeForce RTX 4090)			Inference (Jetson Orin Nano)		
Type	Object	Specifications	Type	Object	Specifications
Hardware	CPU	Intel core i11	Hardware	CPU	6-core Arm® Cortex®-A78AE v8.2 64-bit
	GPU	Nvidia Geforce RTX 4090 with 24564MiB		GPU	1024-core NVIDIA Ampere architecture GPU with 32 Tensor Cores
	RAM	64 GB		RAM	8GB 128-bit LPDDR5
	Power	450 W		Power	15 W
Software	OS	Windows 11 Professional 64 bit	Software	OS	Jetpack 6.0 Developer, Ubuntu 22.04
	Frame work &	PyTorch 2.3.1, CUDA 12.1, cuDNN 9.2.1, Anaconda, PyCharm IDE		Frame work &	PyTorch v2.2.0, CUDA 12.2.12, cuDNN 8.9.4, TensorRT 8.6.2

Training (Nvidia GeForce RTX 4090)			Inference (Jetson Orin Nano)		
Type	Object	Specifications	Type	Object	Specifications
	other tools			Other tools	

Table 4. 2 Training parameters for both OD datasets

Dataset	Training Images	Validation Images	No of Classes	Training Hyperparameters						
				No of Epochs	Batch Size	Image Size	Optimizer	Learning Rate	Momentum	Decay
Vehicle i2	6,638	820	25	100	32 / 16	640	AdamW	0.0003	0.9	0.0005
Brake Light Detection	18,939	3,586	2	100	32 / 16	640	SGD	0.01	0.5	0.005

4.3 Evaluation Metrics

Spatial evaluation of our object detection system is centered on the mean average precision (mAP), a widely recognized and extensively utilized metric in object detection tasks. Among its variants, mAP50 (mean Average Precision at an Intersection over Union [IoU] threshold of 0.5) serves as a straightforward benchmark. It evaluates the precision of the model by calculating how accurately it can identify objects when the overlap between the predicted bounding box and the ground truth bounding box meets or exceeds 50%. This metric offers a single, concise value that reflects the model’s competency in correctly identifying objects with a reasonable level of overlap, making it a popular choice for initial evaluations due to its simplicity and directness.

On the other hand, mAP50-95 (mean Average Precision across IoU thresholds from 0.5 to 0.95) provides a much more thorough assessment of the model's performance. It measures average precision over a range of IoU thresholds (0.5, 0.55, 0.6, ..., 0.95) in increments of 0.05, effectively requiring the model to demonstrate accuracy across varying levels of overlap. This

metric is significantly more rigorous, as it tests the robustness and consistency of the detection system under stricter conditions. By incorporating this range of thresholds, mAP50-95 delivers a nuanced picture of the model’s ability to generalize across diverse scenarios, making it an essential metric for advanced evaluations.

In addition to mAP metrics, we adopted a confusion matrix for a detailed evaluation of object detection and collision prediction performance, particularly when applied to the Vehicle i2, Brake Light Detection, and CCD datasets. The confusion matrix offers a comprehensive framework to assess the Degree of Completeness (Recall) and the Degree of Correctness (Precision). This analysis is based on four pivotal parameters: true positive (TP), true negative (TN), false positive (FP), and false negative (FN), each representing a specific classification outcome for detected and actual samples.

Accuracy, calculated using Equation 4, reflects the proportion of all correctly classified samples among the total number of samples, providing an overall measure of the system’s effectiveness. Precision, expressed through Equation 5, focuses specifically on the model’s ability to correctly identify positive samples, quantifying the ratio of true positives to all detected positives. Meanwhile, Recall, derived from Equation 6, measures the extent to which the detection system identifies all potential objects of interest. As Recall evaluates the quantum of objects covered by the object detection and collision prediction system, it holds the highest weight among our evaluation metrics, given the critical nature of ensuring comprehensive detection coverage in our use case.

This multi-faceted evaluation strategy combines the strengths of mAP metrics and confusion matrix-derived parameters, delivering a robust and detailed understanding of the detection system’s performance across diverse datasets and challenging conditions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

4.4 Results

Tables 4.3 and 4.4 present a detailed comparison of the most recent real-time object detection (OD) models applied to our training datasets for vehicle and brake light detection, respectively. For this analysis, we focused exclusively on real-time OD models released after 2022, prioritizing those that combine high accuracy with real-time performance. Models such as the vision-based end-to-end object detection transformer (DETR) or Dino-DETR, despite their impressive accuracy, were excluded due to their lack of real-time processing capabilities, which are crucial for our application. This selection criterion ensures that the evaluated models align with the performance demands of the Vehicle Collision Avoidance System (V-CAS), where real-time processing is paramount.

Figure 4.2 visually compares the precision and recall metrics of all trained models, evaluated on both the Vehicle i2 dataset and the Brake ON class of the Brake Light Detection dataset. The comparison highlights that RT-DETR consistently outperforms other models, particularly in terms of Recall, which measures the system's ability to detect and cover all relevant objects. Additionally, RT-DETR shows a strong performance in Precision across most cases, indicating its capability to minimize false positives while accurately identifying true positives.

This superior performance of RT-DETR in both metrics underscores its robustness and suitability for our use case. As a result, it has been selected as the backbone OD model for V-CAS, ensuring optimal detection accuracy and real-time responsiveness, which are critical for the effective implementation of our collision avoidance system. By choosing RT-DETR, we ensure that the system is equipped with a state-of-the-art detection framework capable of meeting the challenges of real-world scenarios.

Table 4. 3 Comparison of different real-time object detectors on vehicle i2 public dataset

Model	Size (MB)	Parameters (Mn)	Inference Time per Image (ms)	Evaluation Metrics			
				Precision	Recall	mAP50	mAP5 0-95
YOLOv8s	21.4	11.13	0.7	0.844	0.747	0.83	0.706
YOLOv8m	49.6	25.85	1.8	0.767	0.77	0.816	0.697
YOLOv8l	83.5	43.62	2.7	0.771	0.785	0.803	0.688
YOLOv9s	14.5	7.29	0.9	0.864	0.731	0.842	0.706
YOLOv9c	49.2	25.34	2.6	0.705	0.838	0.842	0.714
YOLOv9e	111	57.39	5.9	0.828	0.723	0.782	0.664
YOLOv10s	15.7	8.05	1	0.85	0.736	0.861	0.73
YOLOv10m	31.9	16.47	1.9	0.856	0.706	0.801	0.676
YOLOv10b	39.5	20.45	2.5	0.75	0.721	0.79	0.654
RT-DETR L	63.1	32.03	2.6	0.857	0.845	0.853	0.724

Table 4. 4 Comparison of different real-time object detectors on brake light detection dataset

Model	Size (MB)	GFLOP S	Brake OFF Class (-ve class)					Brake ON Class (+ve class)				
			TP	FP	FN	Precision	Recall	TP	FP	FN	Precision	Recall
YOLOv8s	21.4	28.5	1479	1308	640	0.530	0.697	1278	1026	586	0.554	0.685
YOLOv8m	49.6	78.8	1560	1238	559	0.557	0.736	1350	1007	514	0.572	0.724
YOLOv8l	83.5	164.9	1572	1245	547	0.558	0.741	1328	996	536	0.571	0.712
YOLOv9s	14.5	26.7	1575	1282	544	0.551	0.743	1325	1028	539	0.563	0.710
YOLOv9c	49.2	102.4	1597	1240	522	0.562	0.753	1337	962	527	0.581	0.717
YOLOv9e	111	189.2	1596	1249	523	0.560	0.753	1361	961	503	0.586	0.730
YOLOv10s	15.7	24.5	1528	1163	591	0.567	0.721	1307	867	557	0.601	0.701
YOLOv10m	31.9	63.6	1474	1321	645	0.527	0.696	1366	989	498	0.580	0.732
YOLOv10b	39.5	98.1	1391	1357	728	0.506	0.656	1362	1020	502	0.572	0.730
RT-DETR L	63.1	103.4	1629	850	490	0.657	0.769	1373	631	491	0.685	0.736

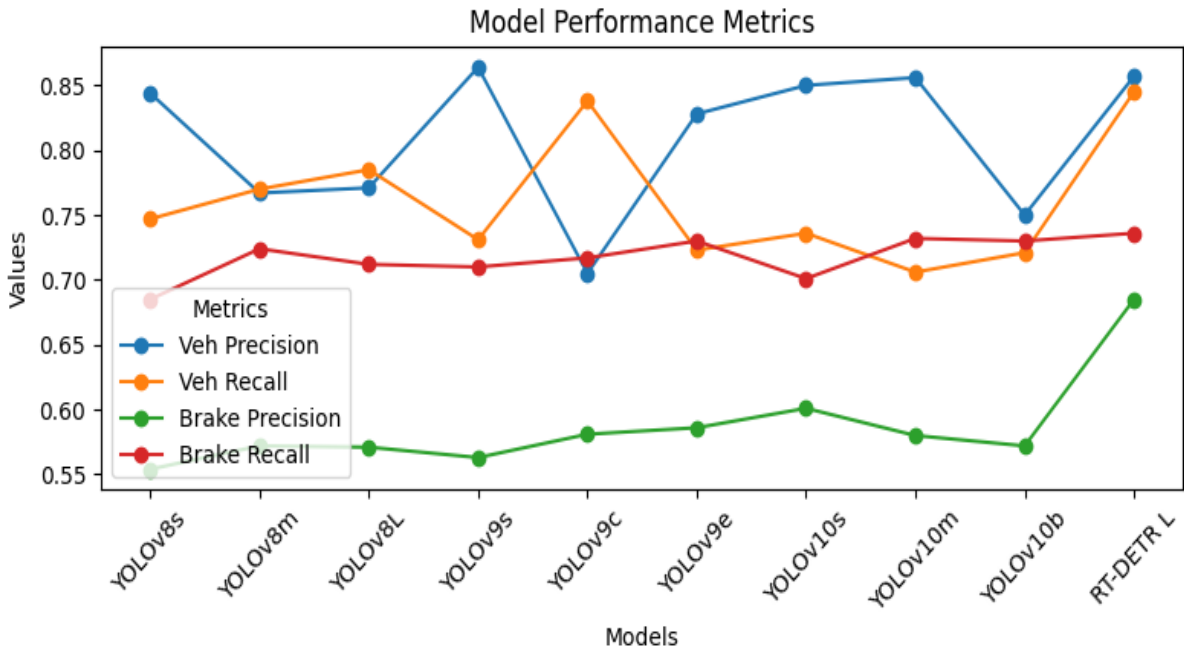


Figure 4.2 Precision and recall comparison on vehicle i2 and brake light detection datasets

Figure 4.2 illustrates real-time parallel object detection on three vehicle mounted dash cam streams having 1920 x 1080 resolution in both day and night. The advantage of having multiple camera streams to get a better understanding of scene is clearly visible. Few objects which are missed with middle camera (monocular approach) are detected with either left or right side camera's view. We have evaluated our own created dataset to see the performance of object detector-tracker results as well as rate of acceleration calculation. However, for collision prediction analysis, we have used the available CCD dataset having actual collision incidents since it was not possible to create actual collision scenario on ground.



Figure 4.3 Multi-Camera Fused Object Detection using Custom Trained RT-DETR

Figure 4.3 is a depiction of collision prediction of V-CAS on CCD Crash-1500 subset in both day and night conditions from the actual crash incident. From left to right we can see how the object was detected first, then its relative rate of acceleration and trajectory continuously being measured and basing upon that a collision warning was generated on screen proactively along with adaptive barking signal generated from 40 pin expansion header of our Jetson device.



Figure 4.4 Day and Night Collision Prediction using VCAS on Car Crash Dataset

Table 4. 5 V-CAS overall performance evaluation on Car Crash dataset

Category	Total	Ground Truth (TP+FN)	V-CAS without Brake Detection			V-CAS with Brake Detection			FPS on Nvidia GeForce RTX 4090	FPS on Jetson Orin Nano
			Predicted (TP+FP)	Precision	Accuracy	Predicted (TP+FP)	Precision	Accuracy		
Day	1062	764	759	98.68%	97.64%	760	98.94%	98.12%	62	15.6
Night	438	376	304	89.47%	68.95%	352	97.72%	90.87%	61.8	15.1

Table 4.5 highlights the performance of the Vehicle Collision Avoidance System (V-CAS) when evaluated on the Crash-1500 subset of the CCD dataset. This subset consists of 1500 crash videos, out of which 1140 depict actual crash incidents involving collisions with host vehicles. The dataset includes 764 daytime and 376 nighttime crash scenarios, providing a comprehensive evaluation under varying visibility conditions.

The results for daytime crashes are highly promising, demonstrating precision levels exceeding 98% and an almost equivalent accuracy, reflecting the robustness of the system during optimal visibility conditions. However, the performance during nighttime crashes reveals challenges, primarily due to poor visibility and the impact of high-beam lights, which degrade detection and tracking capabilities. Without incorporating the brake detection module, the accuracy during nighttime scenarios drops noticeably due to an increased number of false negatives, where collisions fail to be identified effectively.

When the brake detection module is integrated, the system's nighttime performance improves significantly, achieving an accuracy above 90%. This enhancement is attributed to the module's ability to detect brake lights, even in challenging lighting conditions, thereby reducing the false negatives and bolstering the system's reliability.

The embedded implementation of V-CAS maintains near-real-time performance, achieving frame rates of over 15 fps by leveraging detection on alternate frames. While there is a slight reduction in fps during nighttime scenarios, caused by difficulties in object tracking and intermittent detection losses due to challenging lighting conditions, the system still performs effectively.

In summary, V-CAS achieves a final accuracy of 98.12% for daytime crash videos and 90.87% for nighttime crash videos, demonstrating its capability to handle a wide range of real-world conditions with a balance of precision, reliability, and real-time performance. This evaluation underscores the system's adaptability and robustness, particularly with the inclusion of the brake detection module, making it a viable solution for real-time collision avoidance applications.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this research, we have examined the latest techniques and technologies utilized in vehicle collision avoidance systems, focusing on threat assessment, deep learning and embedded systems. We have performed a comprehensive literature review for the vehicle collision avoidance techniques and proposed a real-time, multicamera, collision avoidance system V-CAS using custom trained vision-based transformer RT-DETR and DeepSORT. They were being compared for their performance and precision along with the integration technique for multicamera streams for a single object detector-tracker solution. RT-DETR is a balanced choice between inference speed and precision whereas DeepSORT is best for real-time multi-object tracking in diverse scenarios. Our proposed system showed promising results on the Car Crash Dataset in daytime scenarios with above 98% and 90% accurate results in daytime and nighttime scenarios. A combination of Brake light detection was used to further enhanced night time performance and robustness of our model. Our proposed system is quite precise, computationally efficient, and low-cost real-time solution systems that can be implemented on low-power-embedded platforms for vehicles in everyday life. As these technologies evolve, their integration into ADAS and autonomous vehicles is expected to revolutionize road safety and driving efficiency. The advancements in sensor technologies, computational power, and machine learning algorithms have collectively enhanced the accuracy and reliability of collision avoidance systems.

5.2 Future Work

Looking ahead, several key areas require further research and development to enhance the effectiveness and adoption of collision avoidance systems in autonomous vehicles and ADAS applications. Future work should focus on developing more sophisticated sensor fusion algorithms. They can have the capability to dynamically adjust to changing environmental and lighting conditions, improving the accuracy of sensor data fusion under bad weather conditions. The integration of additional sensors, such as thermal cameras, could provide valuable information in scenarios where traditional sensors may struggle. The computational demands of real-time processing and decision-making continue to be a significant challenge; thus, research should explore more efficient algorithms using the field of tinyML and hardware accelerators to reduce latency and improve the responsiveness of collision avoidance systems. The future of collision avoidance systems will depend on the public acceptability to integration of vehicle-to-everything (V2X) communication or using stand-alone emended systems to perform all tasks independently on vehicles without any foreign intervention. Ensuring the cybersecurity and reliability of these systems is paramount as vehicles become increasingly connected. Moreover, understanding how drivers respond to system alerts and interventions is essential for designing intuitive and effective interfaces. Policymakers must develop dynamic regulations that keep pace with technological advancements while ensuring safety and public acceptance. Collaborative initiatives between industry, academia, and government agencies will be crucial for addressing these challenges and fostering innovation.

References

- [1] Navarro W. W. Wierwille, R. J. Hanowski, J. M. Hankey, C. A. Kieliszewski, S. E. Lee, A. Medina, A. S. Keisler, and T. A. Dingus, "Identification and evaluation of driver errors: Overview and recommendations," U.S. Dept. Transp., Washington, DC, USA, Tech. Rep. FHWA-RD-02-003, 2002.
- [2] C. D. Wang and P. James Thompson, "Apparatus and method for motion detection and tracking of objects in a region for collision avoidance utilizing a real-time adaptive probabilistic neural network," U.S. Patent 005613039 A, Mar. 18, 1997.
- [3] D. R. Ankrum, "Learning from errors in a driving simulation: Effects on driving skill and self-confidence," *Traffic Saf.*, vol. 92, no. 3, pp. 6–9, 1992, doi: 10.1080/00140130050201427.
- [4] X. Chang, H. Li, J. Rong, Z. Huang, X. Chen, and Y. Zhang, "Effects of on-board unit on driving behavior in connected vehicle traffic flow," *J. Adv. Transp.*, vol. 2019, pp. 1–12, Feb. 2019.
- [5] S. Adnan Yusuf, A. Khan, and R. Souissi, 'Vehicle-to-everything (V2X) in the autonomous vehicles domain – A technical review of communication, sensor, and AI technologies for road user safety', *Transportation Research Interdisciplinary Perspectives*, vol. 23. Elsevier Ltd, Jan. 01, 2024.
- [6] Maity, Madhusri, Sriparna Banerjee, and Sheli Sinha Chaudhuri. "Faster r-cnn and yolo based vehicle detection: A survey." In 2021 5th international conference on computing methodologies and communication (ICCMC), pp. 1442- 1447. IEEE, 2021.
- [7] Clarivate. "How to Get a Journal Indexed in the Web of Science Core Collection: Updated Guide." *Scholastica HQ Blog*, Jul. 2021. [Online]. Available: <https://blog.scholasticahq.com>.
- [8] Web of Science. "Core Collection Full Record Details," 2024. [Online]. Available: <https://www.webofscience.com/wos/>.
- [9] M. Berkvens, "CiteSpace analysis of scientific research on the Web of Science database," *International Journal of Technology Management*, vol. 58, no. 4, pp. 356-370, 2012.
- [10] W. A. Richardson and D. King, "Tracking research development using the WoS database," *Research Evaluation*, vol. 12, no. 3, pp. 123-135, 2014.
- [11] T. Meyer, "Decarbonizing road freight transportation – A bibliometric and network analysis," *Transp. Res. Part D: Transport Environ.*, vol. 89, 2020, Art. no. 102619.

- [12] H. Li, J. Zhang, X. Sun, J. Niu, and X. Zhao, "A survey of vehicle group behaviors simulation under a connected vehicle environment," *Physica A: Stat. Mechanics Appl.*, vol. 603, 2022, Art. no. 127816.
- [13] X. Ding and Z. Yang, "Knowledge mapping of platform research: A visual analysis using VOSviewer and CiteSpace," *Electron. Commerce Res.*, vol. 22, no. 3, pp. 787–809, 2022.
- [14] X. Zou, H. L. Vu, and H. Huang, "Fifty years of Accident analysis & prevention: A bibliometric and scientometric overview," *Accident Anal. Prevention*, vol. 144, 2020, Art. no. 105568.
- [15] S.-H. Chung, "Applications of smart technologies in logistics and transport: A review," *Transp. Res. Part E: Logistics Transp. Rev.*, vol. 153, 2021, Art. no. 102455.
- [16] Y. Wu, S. Li, J. Cortes, and K. Poolla, "Distributed sliding mode control for nonlinear heterogeneous platoon systems with positive definite topologies," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 4, pp. 1272–1283, Jul. 2020.
- [17] Y. Bian et al., "Behavioral harmonization of a cyclic vehicular platoon in a closed road network," *IEEE Trans. Intell. Veh.*, vol. 6, no. 3, pp. 559–570, Sep. 2021.
- [18] X. Zou, W. L. Yue, and H. L. Vu, "Visualization and analysis of mapping knowledge domain of road safety studies," *Accident Anal. Prevention*, vol. 118, pp. 131–145, 2018.
- [19] S. Lefevre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH journal*, vol. 1, no. 1, pp. 1–14, 2014.
- [20] G. Kahn, A. Villaflor, B. Ding, P. Abbeel, and S. Levine, "Self supervised deep reinforcement learning with generalized computation graphs for robot navigation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8.
- [21] Y. F. Chen, M. Liu, M. Everett, and J. P. How, "Decentralized noncommunicating multiagent collision avoidance with deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 285–292.
- [22] M. Kim, S. Lee, J. Lim, J. Choi, and S. G. Kang, "Unexpected collision avoidance driving strategy using deep reinforcement learning," *IEEE Access*, vol. 8, pp. 17 243–17 252, 2020.
- [23] X. Li, P. Wang, and H. Xu, "Deep learning for real-time traffic sign detection and classification with environmental influence filtering," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3344–3353, 2020.
- [24] J. Xu, X. Li, and Z. Sun, "Early fusion of radar and camera for robust lane marking detection in adverse weather conditions," *IEEE Access*, vol. 7, pp. 77432–77442, 2019.

- [25] J. Kim, S. Hong, S. Yoon, and M. Kim, "Late fusion based deep learning approach for pedestrian detection using camera and LiDAR data," 2018 16th International Conference on Advanced Robotics (ICAR), pp. 1-6, 2018.
- [26] X. Zhu, Y. Li, X. Zhao, J. Huang, and S. Zhang, "Deep learning based feature fusion for multi-sensor object detection," 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 180-185, 2017.
- [27] J. Huang, Z. Sun, Y. Zhou, H. Bao, and X. Guo, "Attention-based multi-sensor fusion for vehicle detection in adverse weather conditions," *Neurocomputing*, vol. 469, pp. 71-82, 2021.
- [28] Chen Y, Palanisamy P, Mudalige P, Muelling K, Dolan JM. Learning on-road visual control for self-driving vehicles with auxiliary tasks. In: *IEEE Winter Conf Appl Comput Vis (WACV)*. 2019. pp. 331–8.
- [29] Shi C, Dong Z, Pundlik S, Luo G. A hardware-friendly optical flow-based time-to-collision estimation algorithm. *Sensors*. 2019;19(4).
- [30] Datondji, Sokemi Rene Emmanuel, Yohan Dupuis, Peggy Subirats, and Pascal Vasseur. "A survey of vision-based traffic monitoring of road intersections." *IEEE transactions on intelligent transportation systems* 17, no. 10 (2016): 2681-2698.
- [31] Maity, Madhusri, Sriparna Banerjee, and Sheli Sinha Chaudhuri. "Faster r-cnn and yolo based vehicle detection: A survey." In 2021 5th international conference on computing methodologies and communication (ICCMC), pp. 1442- 1447. IEEE, 2021.
- [32] Chen, Shaobin, and Wei Lin. "Embedded system real-time vehicle detection based on improved YOLO network." In 2019 IEEE 3rd advanced information management, communicates, electronic and automation control conference (IMCEC), pp. 1400-1403. IEEE, 2019.
- [33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.
- [34] N. Sharma, S. Baral, M. P. Paing, and R. Chawuthai, "Parking Time Violation Tracking Using YOLOv8 and Tracking Algorithms," *Sensors*, vol. 23, no. 13, 2023.
- [35] F. Ngeni, J. Mwakalonge, and S. Siuhi, "Solving traffic data occlusion problems in computer vision algorithms using DeepSORT and quantum computing," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 11, no. 1, 2024.
- [36] L. Lin, H. He, Z. Xu, and D. Wu, "Realtime Vehicle Tracking Method Based on YOLOv5 + DeepSORT," *Comput Intell Neurosci*, vol. 2023, 2023.

- [37] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 21–37.
- [38] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, arXiv:1704.04861.
- [39] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," 2018, arXiv:1801.04381.
- [40] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "DETRs Beat YOLOs on Real-time Object Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2024, pp. 16965-16974.
- [41] Tesla Places Big Bet on Vision-Only Self-Driving, IEEE Spectrum Available: [Tesla Places Big Bet on Vision-Only Self-Driving - IEEE Spectrum](#).
- [42] Yakovlev S, Borisov A. A synergy of the rosenblatt perceptron and the Jordan recurrence principle. *Automat Contr Comput Sci* 2009;43:31–9.
- [43] Goodfellow I, Heaton J, Bengio Y, Courville A. *Deep learning*. first ed. The MIT Press; 2016, ISBN 0262035618800pp.
- [44] S. Yun, D. Kum, The multilayer perceptron approach to lateral motion prediction of surrounding vehicles for autonomous vehicles, *IEEE Intelligent Vehicles Symposium (IV)*.
- [45] Van Brummelen J, OBrien M, Gruyer D, Najjaran H. Autonomous vehicle perception: the technology of today and tomorrow. *Transport Res Part C* 2018;89: 384–406.
- [46] Brownsword R. From erewhon to alphago: for the sake of human dignity, should we destroy the machines? *Law, Innovation and Technology* 2017;9:117–53.
- [47] Sugano Y, Matsushita Y, Sato Y. Appearance-based gaze estimation using visual saliency. *IEEE Trans Pattern Anal Mach Intell* 2013;35:329–41.
- [48] Li Y, Wang J, Xing T, Liu T, Li C, Su K. Tad16k: an enhanced benchmark for autonomous driving. *IEEE International Conference on Image Processing* 2017: 2344–8.
- [49] <https://www.statefarm.com/simple-insights/auto-and-vehicles/latest> car-safety-features-becoming-must-haves. Accessed 17 January 2021
- [50] Dinita, M.: Best road design software for PC [2020 Guide] (2019). <https://windowsreport.com/road-design-software/>. Accessed 17 January 2021
- [51] <https://www.bentley.com/en/solutions/road-design-and-analysis> Accessed 17 January 2021

- [52] Kirkland, G.: How new technologies have changed the automotive industry (2019) <https://www.oponeo.co.uk/blog/how-newtechnologies-have-changed-the-automotive-industry> Accessed 17 January 2021
- [53] [https://www.oponeo.co.uk/blog/how-new-technologies-have-changed-the-automotive-industry#:~:text= The Growth of Autonomous Technology&text= Most modern cars feature autonomous, and workout potential collisions.](https://www.oponeo.co.uk/blog/how-new-technologies-have-changed-the-automotive-industry#:~:text=The%20Growth%20of%20Autonomous%20Technology&text=Most%20modern%20cars%20feature%20autonomous,%20and%20workout%20potential%20collisions.) Accessed 20 June 2022.
- [54] Autopilot and Full Self-Driving Capability (2019). <https://cvpr2021.wad.vision/>. Accessed 20 June 2022.
- [55] Zheng, L., Ismail, K., Meng, X.: Traffic conflict techniques for road safety analysis: open questions and some insights. *Can. J. Civ. Eng.* 41(7), 633– 641 (2014).
- [56] Li Y, Cui F, Xue X, Cheung-Wai Chan J. Coarse-to-fine salient object detection based on deep convolutional neural networks. *Signal Process Image Commun* 2018;64:21–32.
- [57] Zhang K. Beyond a Gaussian denoiser: residual learning of deep cnn for image denoising. *IEEE Trans Image Process* 2017;26:3142–55.
- [58] Mahmoudi N, Ahadi S, Rahmati M. Multi-target tracking using cnn-based features: Cnnmtt. *Multimedia Tools and Applications*; 2018. p. 1–20.
- [59] Qian Y, Dong J, Wang W, Tan T. Feature learning for steganalysis using convolutional neural networks. *Multimed Tool Appl* 2018;77:19633–57.
- [60] Al-Qizwini M. Deep learning algorithm for autonomous driving using googlenet. *IEEE Intelligent Vehicles Symposium 2017;(IV):89–96* <https://doi.org/10.1109/IVS.2017.7995703>.
- [61] Chung J. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv.org*, <https://arxiv.org/pdf/1412.3555.pdf>.
- [62] Han M, Chen W, Moges A. Fast image captioning using lstm. *Cluster Comput* 2019; 22(S3):6143–55.
- [63] Liu A, Shao Z, Wong Y, Li J, Su Y, Kankanhalli M. Lstm-based multi-label video event detection. *Multimed Tool Appl* 2019;78(1):677–95.
- [64] Ullah A, Ahmad J, Muhammad K, Sajjad M, Baik SW. Action recognition in video sequences using deep bi-directional lstm with cnn features. *IEEE access* 2018;6: 1155–66.
- [65] Abdellaoui M, Douik A. Human action recognition in video sequences using deep belief networks. *Trait Du Signal* 2020;37(1):37–44.
- [66] Lahan GS, Talukdar AK, Sarma KK. Action recognition from depth video sequences using microsoft kinect. *IEEE*; 2019. p. 35–40.
- [67] Cho, K., Van Merriënboer, B., Bahdanau, D., Bengio, Y.: On the properties of neural machine translation: encoder-decoder approaches. *arXiv:14091259* (2014).

- [68] Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neur. Comput.* 9(8), 1735–1780 (1997).
- [69] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al.: Attention is all you need. *arXiv:1706.03762* (2017).
- [70] Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., et al.: A survey on vision transformer. *IEEE Trans. Pattern Anal. Mach. Intell.* (2022).
<https://doi.org/10.1109/TPAMI.2022.3152247>.
- [71] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805* (2018).
- [72] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., et al.: An image is worth 16x16 words: transformers for image recognition at scale. *arXiv:2010.11929* (2020).
- [73] Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., et al.: Token-to-token VIT: training vision transformers from scratch on imagenet. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 558–567. IEEE, Piscataway, NJ (2021).
- [74] Wang, M., Wang, X.: Automatic adaptation of a generic pedestrian detector to a specific traffic scene. In: *CVPR 2011*, pp. 3401–3408. IEEE, Piscataway, NJ (2011).
- [75] Hospedales, T., Gong, S., Xiang, T.: Video behaviour mining using a dynamic topic model. *Int. J. Comput. Vis.* 98(3), 303–323 (2012).
- [76] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Vol. 1*. pp. I–I. IEEE, Piscataway, NJ (2001).
- [77] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 1, pp. 886–893. IEEE, Piscataway, NJ (2005).
- [78] Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Anal. Mach. Intell.* 32(9), 1627–1645 (2009).
- [79] Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014*; pp. 580–587.
- [80] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 2017, 60 , 84–90. [CrossRef]

- [81] Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* 2013, 104, 154–171.
- [82] Girshick, R. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
- [83] Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* 2015, 28, 1137–1149.
- [84] Mita, T.; Kaneko, T.; Hori, O. Joint haar-like features for face detection. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)*, Beijing, China, 17–21 October 2005; Volume 2, pp. 1619–1626.
- [85] Zhang, G.; Huang, X.; Li, S.Z.; Wang, Y.; Wu, X. Boosting local binary pattern (LBP)-based face recognition. In *Proceedings of the Chinese Conference on Biometric Recognition*, Guangzhou, China, 13–14 December 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 179–186.
- [86] Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. *Adv. Neural Inf. Process. Syst.* 2016, 29. Available online: https://proceedings.neurips.cc/paper_files/paper/2016/file/577ef1154f3240ad5b9b413aa7346a1ePaper.pdf (accessed on 25 April 2023).
- [87] Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, Zurich, Switzerland, 6–12 September 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.
- [88] Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* 2010, 88, 303–338.
- [89] Wang, H.; Yu, Y.; Cai, Y.; Chen, X.; Chen, L.; Liu, Q. A comparative study of state-of-the-art deep learning algorithms for vehicle detection. *IEEE Intell. Transp. Syst. Mag.* 2019, 11, 82–95. [CrossRef].
- [90] N. Wojke, A. Bewley, and D. Paulus, “Simple Online and Realtime Tracking with a Deep Association Metric,” Mar. 2017, [Online] Available: <http://arxiv.org/abs/1703.07402>.
- [91] NVIDIA. (n.d.). Managing Video Streams in Runtime with the NVIDIA DeepStream SDK. NVIDIA Technical Blog. [Online]. Available: <https://developer.nvidia.com/blog/managing-video-streams-in-runtime-with-nvidia-deepstream-sdk>.

- [92] C. K. Tan, S. V. Rao, and K. Haribabu, "Optimizing Video Streaming Pipelines with NVIDIA GStreamer and CUDA," in *IEEE Transactions on Multimedia*, vol. 23, no. 7, pp. 1892-1904, Jul. 2021.
- [93] W. Bao, Q. Yu, and Y. Kong, "Uncertainty-based Traffic Accident Anticipation with Spatio-Temporal Relational Learning," in *MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia*, Association for Computing Machinery, Inc, Oct. 2020, pp. 2682–2690.
- [94] A. Pujara and M. Bhamare, "DeepSORT: Real Time & Multi-Object Detection and Tracking with YOLO and TensorFlow," in *Proceedings - International Conference on Augmented Intelligence and Sustainable Systems, ICAISS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 456–460.