

AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering Degree in Electronics and Communication Engineering

by

P.V.V.L.D.S.SAI MANIKANTA (40130179)

PRADEEP SHARMA (40130178)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY -1 UNIVERSITY BY UGC

Accredited with Grade “A++” by NAAC | 12B Status by UGC | Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **P.V.V.L.D.S.SAI MANIKANTA (40130179), PRADEEP SHARMA (40130178)** who carried out the Project entitled "**AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY**" under our supervision from November 2023 to April 2024.


Internal Guide

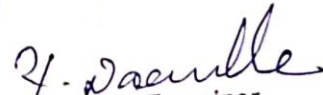
Dr.E. ANNA DEVI, M.E, Ph.D.,

Head of the Department

Dr. T. RAVI, M.E, Ph.D.,

Submitted for Viva voce Examination held on 17.4.2024


Internal Examiner


External Examiner

DECLARATION

We **P.V.V.L.D.S.SAI MANIKANTA** and **PRADEEP SHARMA** hereby declare that the Project Report entitled "**AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY** " done by us under the guidance of **Dr. E.ANNA DEVI M.E, Ph.D.,** Dept of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in ELECTRONICS AND COMMUNICATION ENGINEERING.

DATE: 17-04-2024

PLACE: Chennai

SIGNATURE OF THE CANDIDATES:

1 P.V.V.L.D.S.Sai Manikanta

2. Pradeep

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N.M. NANDHITHA M.E., Ph.D., Dean, School of Electrical and Electronics Engineering**, and **Dr. T. RAVI M.E., Ph.D., Head of the Department of Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. E. ANNA DEVI M.E, Ph.D.**, dept. of Electronics and Communication Engineering for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

This project aims to evaluate the accuracy of a real-time traffic management system by comparing the performance of two different methods: novel Adaboosting and logistic regression, for forecasting the count of automobile wheel movements. The dataset used contains 12 attributes and 10,684 rows, representing various vehicle types, such as cars, bicycles, motorcycles, and trucks. The data source provided a sample of 1,340 records. Through simulations with 20 iterations each, the novel Adaboosting algorithm and logistic regression were assessed using different parameters to optimize vehicle monitoring systems and automate pH adjustments. The calculations were performed with CInCal and G power at an 80% power level and a 95% confidence interval. The results revealed that the novel Adaboosting algorithm achieved significantly higher accuracy (93.15%) compared to the logistic regression algorithm (80.60%). A statistical analysis using an Independent Sample Test yielded a p-value of 0.013, indicating a statistically significant difference between the two algorithms ($p < 0.05$). In conclusion, the study demonstrates that the novel Adaboosting algorithm outperforms logistic regression in accurately estimating road traffic counts. The automotive industry is witnessing rapid advancements in automation and artificial intelligence (AI) technologies, driving the need for more efficient and accurate methods of vehicle inspection and quality control. Traditional wheel counting methods are often manual and prone to human errors, making them time-consuming and unreliable. In contrast, our system leverages the power of CV and AI to automate the process, offering several advantages.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	V
	LIST OF ABBREVIATIONS	IX
	LIST OF FIGURES	X
	LIST OF TABLES	XII
1	INTRODUCTION	1
	1.1 FUNCTIONS	2
	1.1.1 VIDEO INPUT PREPROCESSING	2
	1.1.2 VEHICLE DETECTION AND TRACKING	3
	1.1.3 DATA ANALYSIS AND VISUALISATION	3
	1.3 APPLICATIONS	5
	1.3.1 TRAFFIC MANAGEMENT	5
	1.3.2 PARKING MANAGEMENT	5
	1.3.3 RETAIL ANALYSIS	6
	1.3.4 SECURITY AND SURVEILLANCE	6
	1.3.5 ENVIRONMENTAL MONITORING	6
	1.3.6 PUBLIC TRANSPORT MANAGEMENT	7
	1.3.7 ROAD SAFETY	7
	1.4 RECENT TRENDS AND FUTURE	8
	1.5 OBJECTIVE	9
2	LITERATURE SURVEY	

	2.1	GENERAL SURVEY	10
	2.2	OVERVIEW OF EXISTING RESEARCH	13
3		AIM AND SCOPE OF EXISTING DESIGN	
	3.1	AIM OF THE PROJECT	14
	3.2	AREA OF THE PROJECT	14
	3.3	EXISTING WORK	15
	3.4	IMPLEMENTATION OF EXISTING WORK	16
4		MATERIALS AND METHODS USED	
	4.1	PROPOSED SYSTEM	17
	4.2	BLOCK DIAGRAM OF PROPOSED WORK	18
	4.3	SOFTWARE DESCRIPTION	18
	4.3.1	PYTHON IDLE	18
	4.3.2	PYTHON	19
	4.3.3	MACHINE LEARNING	20
	4.3.4	ARTIFICIAL INTELLIGENCE	21
	4.3.5	OPEN CV	23
	4.3.6	ADABOOSTING ALGORITHM	25
	4.3.7	LOGIC REGRESSION	26
	4.4	WORKING PROCESS AND SIMULATION	

4.4.1	IMPORT MODULES	27
4.4.2	BACKGROUND SUBTRACTION	28
4.4.3	FILTERING	29
4.4.4	BUILDING PROCESSING PIPELINE	30
4.5	STANDARDS	32
4.6	CONSTRAINTS	34
4.7	TRADEOFF	35
5	RESULTS AND DISCUSSION,PERFORMANCE ANALYSIS	
5.1	MEASURE THE TRACKING AND COUNTING THE VEHICLES	37
5.2	PARAMETERS	37
5.3	MEASUREMENT OF ACCURACY	38
5.4	ANALYSIS OF BOTH ALGORITHMS	39
6	SUMMARY AND CONCLUSION	
6.1	SUMMARY	43
6.2	CONCLUSION	43
	REFERENCES	
	APPENDIX	

LIST OF ABBREVIATIONS

ADAB	Adaboosting Algorithm
CV	Computer Vision
AI	Artificial Intelligence
ML	Machine Learning
IDE	Integrated Development Environment
LRA	Logic Regression Algorithm
NADAB	Nova Adaboosting Algorithm
SD	Standard Deviation
CNN	Convolutional Neural Network
CSV	Comma Seperated Value
SORT	Simple Online and Realtime Tracking
YOLO	You Only Look Once
SSD	Solid State Drive
SEM	Standard Error of the Mean
GPU	Graphics processing Unit
API	Application Programming Interface

LIST OF FIGURES

Fig No.	FIGURE NAME	Page No.
4.1	BLOCK DIAGRAM	18
4.2	MACHINE LEARNING	21
4.3	ARTIFICIAL INTELLIGENCE	22
4.4	OPEN CV	24
4.5	ADABOOSTING ALGORITHM	26
4.6	BACKGROUND SUBTRACTION	28
4.7	BACKGROUNG REMOVED	28
4.8	FILTERING	30
4.9	VEHICLE DETECTION AND COUNTING	31
5.1	FINAL GRAPH	32
5.2	FRAMES	33
5.3	COMPARING ALGORITHMS	34

LIST OF TABLES

TABLE No.	TABLE NAME	Page No.
5.1	Finding std deviation	34
5.2	Leven's T-test	36

CHAPTER 1

INTRODUCTION

Automobile vehicle counting in computer vision predicts vehicle movement, reducing traffic congestion and evaluating energy efficiency, making it crucial in artificial intelligence for traffic management. Introduce a new computer vision method for automating vehicle classification and detection. Their research focuses on optimizing traffic management by utilizing machine learning to identify vehicle wheel movement while considering energy efficiency and classification. anticipate a significant role for artificial intelligence in recognizing and predicting vehicle movements, proposing diverse approaches for counting automobile wheel movements to boost roadway counting accuracy and improve traffic management. Automobile vehicle counting based on computer vision (CV) through artificial intelligence and machine learning (AIML) is a sophisticated system that automates the process of counting vehicles on roads or in parking lots. This technology utilizes cameras to capture video footage of the traffic, which is then analyzed using computer vision algorithms to detect and track vehicles.

The AIML component of the system is responsible for processing the video data, detecting vehicles, and counting them accurately. Techniques such as object detection, tracking, and classification are used to identify vehicles in the video stream. One commonly used algorithm for vehicle detection in CV applications is the Adaboosting algorithm, which is a machine learning technique that combines multiple weak classifiers to create a strong classifier. This algorithm is effective for detecting objects in images or video frames. Overall, automobile vehicle counting based on CV through AIML offers a reliable and efficient way to monitor traffic flow and analyze vehicle movements in real-time, providing valuable insights for traffic management and urban planning.

1.1 FUNCTIONS

1.1.1 *Video Input and Preprocessing*

Video Input: The system captures video footage of traffic using cameras placed at strategic locations.

Preprocessing: Before analysis, the video frames undergo preprocessing steps such as noise reduction, contrast enhancement, and resizing. These steps ensure that the images are clear and suitable for vehicle detection.

1.1.2 *Vehicle Detection and Tracking*

Detection: Using computer vision techniques like deep learning and object detection algorithms (e.g., YOLO, SSD), the system identifies vehicles in each frame by recognizing their distinct features such as shape, size, and movement patterns.

Tracking: Once detected, vehicles are tracked across frames using algorithms like Kalman filters or optical flow. This allows the system to maintain a consistent count of vehicles and their movements. Vehicle detection and tracking systems employ advanced computer vision techniques to identify vehicles in video streams. These systems utilize deep learning models such as YOLO (You Only Look Once) or SSD (Single Shot MultiBox Detector) for efficient and accurate vehicle detection. These algorithms analyze each frame of the video to recognize distinct features of vehicles, including their shape, size, and movement patterns. Once vehicles are detected, they are tracked across frames using algorithms like Kalman filters or optical flow. These tracking algorithms ensure that each vehicle is consistently tracked, allowing for an accurate count of vehicles and their movements. In addition to detection and tracking, these systems can also classify vehicles into different categories, such as cars, trucks, or buses, based on their size and shape. They can estimate the speed of vehicles by analyzing their movement patterns over time. Direction estimation is also possible, providing insights into the direction of travel of each vehicle. These capabilities are crucial for understanding traffic flow and congestion patterns, which can inform traffic management strategies. Furthermore, vehicle detection and tracking systems can detect and track other objects in the environment, such as pedestrians, cyclists, or obstacles.

This comprehensive view of the environment enables these systems to provide valuable data for various applications, including traffic monitoring, smart transportation systems, and autonomous driving. By integrating with Geographic Information Systems (GIS), these systems can provide additional contextual information, such as road layouts and traffic signs, enhancing their utility in real-world scenarios.

Overall, vehicle detection and tracking systems play a vital role in modern transportation systems, providing valuable insights for traffic management, urban planning, and the development of autonomous vehicles. Their ability to accurately detect, track, and classify vehicles in real time makes them indispensable tools for improving road safety and efficiency.

1.1.3 Data Analysis and Visualization

Analysis: The system analyzes the tracked vehicle data to determine metrics such as traffic volume, speed, and density. This information can be used to identify congestion points, optimize traffic flow, and improve overall road safety.

Visualization: To make the data more accessible, the system presents it in visual formats such as charts, graphs, and heatmaps. These visualizations provide insights into traffic patterns and help stakeholders make informed decisions regarding traffic management and urban planning.

Together, these components form a robust system for automating vehicle counting and traffic analysis, providing valuable information for transportation authorities and city planners.

1.2 CHARACTERISTICS

The project of automobile vehicle counting based on computer vision through artificial intelligence and machine learning (AIML) showcases several key characteristics. Firstly, it operates in real-time, enabling immediate analysis of traffic flow and vehicle counts, which is critical for applications requiring instant feedback, such as traffic management systems. Secondly, the system is designed for accuracy and reliability, utilizing advanced computer vision algorithms to detect and track vehicles accurately, minimizing errors in counting and ensuring trustworthy data for decision-making.

Another characteristic is scalability, as the system can be scaled to handle varying levels of traffic and multiple camera feeds, ensuring effectiveness in different traffic scenarios. Additionally, the system is adaptable to different environments and lighting conditions, thanks to its use of machine learning, which allows it to perform effectively in diverse settings like urban streets or highways. The project is efficient, automating the process of vehicle counting and traffic analysis, leading to more timely and informed decision-making by traffic authorities. It also aligns with the broader goal of creating smart cities by integrating with IoT devices and infrastructure, contributing to better traffic management and urban planning. The system provides visual representations of traffic data, enabling stakeholders to gain valuable insights into traffic patterns and trends, which can inform policy-making and infrastructure development. Furthermore, the project offers long-term cost savings by reducing the need for manual traffic monitoring and analysis. Lastly, it includes a user-friendly interface that allows operators to interact with the system easily, enhancing its usability and accessibility. The project of automobile vehicle counting based on computer vision through artificial intelligence and machine learning (AIML) showcases several key characteristics. Firstly, it operates in real-time, enabling immediate analysis of traffic flow and vehicle counts, which is critical for applications requiring instant feedback, such as traffic management systems. Secondly, the system is designed for accuracy and reliability, utilizing advanced computer vision algorithms to detect and track vehicles accurately, minimizing errors in counting and ensuring trustworthy data for decision-making.

Another characteristic is scalability, as the system can be scaled to handle varying levels of traffic and multiple camera feeds, ensuring effectiveness in different traffic scenarios. Additionally, the system is adaptable to different environments and lighting conditions, thanks to its use of machine learning, which allows it to perform effectively in diverse settings like urban streets or highways. The project is efficient, automating the process of vehicle counting and traffic analysis, leading to more timely and informed decision-making by traffic authorities. It also aligns with the broader goal of creating smart cities by integrating with IoT devices and infrastructure, contributing to better traffic management and urban planning. The system provides visual representations of traffic data, enabling stakeholders to gain valuable insights into traffic patterns and trends, which can inform policy-making and infrastructure development. Furthermore, the project offers long-term cost savings by reducing the

need for manual traffic monitoring and analysis. Lastly, it includes a user-friendly interface that allows operators to interact with the system easily, enhancing its usability and accessibility

1.3 APPLICATIONS

Applications of vehicle detection systems are

1.3.1 Traffic Management: One of the primary applications is in traffic management systems. The system can provide real-time data on traffic flow, congestion, and vehicle counts, enabling authorities to optimize traffic signal timings, manage lane closures, and improve overall traffic flow.

Vehicle detection systems play a crucial role in traffic management, providing real-time data on traffic flow, congestion, and vehicle counts. This data enables authorities to optimize traffic signal timings, manage lane closures, and improve overall traffic flow. By accurately monitoring vehicle movements, these systems help reduce congestion, minimize travel times, and enhance road safety. Additionally, they support the implementation of dynamic traffic management strategies, such as variable speed limits and adaptive traffic signal control, leading to more efficient use of road networks and improved driving conditions for commuters.

1.3.2 Parking Management: The system can be used to monitor parking lots and garages, providing information on parking availability and occupancy. This information can be used to optimize parking space utilization and improve the overall parking experience for drivers. In parking management, vehicle detection systems are utilized to monitor parking lots and garages, offering real-time information on parking availability and occupancy. This data can be used to optimize parking space utilization, guide drivers to available parking spots, and enhance the overall parking experience. By providing accurate and up-to-date parking information, these systems help reduce the time spent searching for parking, alleviate traffic congestion in parking areas, and improve the efficiency of parking operations. Additionally, they can enable the implementation of dynamic pricing strategies based on parking demand, encouraging turnover and maximizing revenue for parking operators

1.3.3 Retail Analytics: In retail environments, the system can be used to analyze customer traffic patterns and behavior. This information can help retailers optimize store

layouts, staff allocation, and marketing strategies to improve customer satisfaction and increase sales. In retail analytics, vehicle detection systems play a crucial role in analyzing customer traffic patterns and behavior. By monitoring vehicle movements in parking areas and entrances, these systems provide valuable insights into customer behavior, such as peak shopping times, popular store entrances, and average visit durations. This information enables retailers to optimize store layouts, staff allocation, and marketing strategies to improve customer satisfaction and increase sales. Additionally, vehicle detection systems can help retailers assess the effectiveness of promotions and advertising campaigns by correlating foot traffic with sales data, allowing for more targeted and efficient marketing efforts.

1.3.4 Security and Surveillance: The system can also be used for security and surveillance purposes. It can monitor vehicle traffic in sensitive areas, such as airports, borders, and military installations, to detect and deter unauthorized vehicles or suspicious activities. In security and surveillance applications, vehicle detection systems are employed to monitor vehicle traffic in sensitive areas like airports, borders, and military installations. These systems help detect and deter unauthorized vehicles or suspicious activities by providing real-time alerts and tracking capabilities. By monitoring vehicle movements and identifying anomalies, such as unauthorized entry or suspicious behavior, these systems enhance security measures and enable rapid response to potential threats. Additionally, they can integrate with other security systems, such as CCTV cameras and access control systems, to provide comprehensive security coverage and ensure the safety of critical infrastructure and facilities.

1.3.5 Environmental Monitoring: The system can contribute to environmental monitoring efforts by providing data on vehicle emissions and traffic-related pollution. This information can be used to develop strategies for reducing air pollution and improving air quality in urban areas. In environmental monitoring, vehicle detection systems play a crucial role by providing data on vehicle emissions and traffic-related pollution. This information is valuable for developing strategies to reduce air pollution and improve air quality in urban areas. By accurately monitoring vehicle movements and emissions, these systems help identify high pollution areas, assess the impact of traffic on air quality, and guide the implementation of measures such as low-emission zones, traffic management strategies, and public transportation improvements. Ultimately, these efforts contribute to creating healthier and more sustainable urban environments.

1.3.6 Public Transport Management: The system can be integrated with public transport systems to monitor bus and tram traffic, optimize routes, and improve the overall efficiency of public transportation services. In public transport management, vehicle detection systems are integrated with public transport systems to monitor bus and tram traffic, optimize routes, and improve the overall efficiency of public transportation services. By accurately tracking vehicle locations and movements, these systems help transportation authorities monitor service adherence to schedules, identify bottlenecks in routes, and optimize vehicle dispatching. This data can also be used to provide real-time information to passengers regarding bus and tram arrivals, reducing waiting times and improving the overall passenger experience. Additionally, vehicle detection systems can help transportation authorities analyze ridership patterns and demand, enabling them to adjust service frequencies and routes to better meet the needs of passengers.

1.3.7 Road Safety: By providing real-time data on traffic flow and congestion, the system can help identify potential safety hazards and implement measures to improve road safety, such as speed limit enforcement and traffic calming measures. In the context of road safety, vehicle detection systems contribute significantly by providing real-time data on traffic flow and congestion. This data enables authorities to identify potential safety hazards, such as areas prone to accidents or high-speed traffic, and implement measures to improve road safety. For example, authorities can use this data to enforce speed limits, install traffic calming measures like speed bumps or traffic islands, and improve road signage to alert drivers of potential dangers. By leveraging vehicle detection systems, authorities can proactively address road safety issues and reduce the risk of accidents, ultimately saving lives and improving road safety for all road users. Vehicle detection systems also play a crucial role in road safety by enabling the implementation of intelligent transportation systems (ITS) technologies.

These systems can detect and alert drivers about hazardous road conditions, such as slippery roads or reduced visibility due to weather conditions. Additionally, vehicle detection systems can be used to monitor and enforce traffic laws, such as red light violations and illegal parking, further enhancing road safety. Furthermore, by providing real-time traffic data to drivers through navigation systems or mobile apps,

vehicle detection systems help drivers make informed decisions about their routes, reducing the likelihood of accidents and improving overall road safety.

1.4 RECENT TRENDS AND FUTURE

Recent advancements in computer vision and machine learning have significantly improved the accuracy and efficiency of vehicle counting systems. Techniques like deep learning and convolutional neural networks (CNNs) have been particularly impactful, enabling more precise detection and tracking of vehicles in complex traffic scenarios. Furthermore, the integration of these systems with IoT devices and cloud computing has enhanced their scalability and real-time processing capabilities. This integration allows for the seamless collection, analysis, and dissemination of traffic data, leading to more effective traffic management strategies. The Future works can be carried are Looking ahead, several trends are likely to shape the future of automobile vehicle counting systems. Continued advancements in AI and machine learning algorithms will further improve the accuracy of vehicle counting systems, even in challenging conditions such as poor lighting or heavy traffic. Future systems may integrate multiple data sources, such as video feeds, radar, and LiDAR, to enhance vehicle detection and tracking accuracy. This multi-modal approach can provide a more comprehensive view of traffic conditions.

The adoption of edge computing technologies will enable vehicle counting systems to process data closer to the source, reducing latency and improving real-time decision-making capabilities. As the adoption of autonomous vehicles increases, vehicle counting systems will need to evolve to accurately detect and track these vehicles. This may require the development of new algorithms and sensor technologies tailored to autonomous vehicle detection. With the increased collection and sharing of traffic data, ensuring the privacy and security of this data will become increasingly important. Future systems will need to implement robust data protection measures to safeguard sensitive information. Techniques like deep learning and convolutional neural networks (CNNs) have been particularly impactful, enabling more precise detection and tracking of vehicles in complex traffic scenarios. Furthermore, the integration of these systems with IoT devices and cloud computing has enhanced their scalability and real-time processing capabilities.

1.5 OBJECTIVE OF THE PROJECT

The primary objective of this project is to develop a robust and efficient system for the automated counting of automobile wheels through the utilization of Computer Vision (CV) and Artificial Intelligence (AI), specifically employing the Adaboosting algorithm. Several key objectives underpin this overarching goal. Firstly, we aim to create a system that can accurately and reliably detect and count the number of wheels on various types and sizes of automobiles in real-time. Traditional manual counting methods are prone to errors and are time-consuming. By implementing advanced AI and CV techniques, our system seeks to significantly enhance the accuracy and speed of this critical quality control process in the automotive industry. Secondly, we intend to develop a versatile and adaptable solution that can be seamlessly integrated into different stages of automotive production lines. This adaptability ensures that the system can cater to the diverse needs of various manufacturers, whether they produce passenger cars, commercial vehicles, or specialized industrial equipment.

Furthermore, this project aspires to demonstrate the cost-effectiveness of AI-driven automation in the manufacturing sector. By reducing the reliance on manual labor for repetitive and error-prone tasks, our system aims to reduce operational costs and improve resource allocation, ultimately contributing to the bottom line of automotive manufacturers. Additionally, we aim to showcase the broader potential of AI and CV in revolutionizing quality control practices across industries. Our project serves as a practical example of how cutting-edge technologies can enhance safety and quality assurance in the automotive sector, setting a precedent for the adoption of similar automation solutions in other manufacturing domains.

CHAPTER-2

LITERATURE SURVEY

2.1 GENERAL

The following shows a literature survey done for Automobile Vehicle count based on cv through AIML using Adaboosting Algorithm for determining the efficiency and how to calculate the statistical analysis. The other surveys are conducted about to finding the accuracy of the vehicle for detection and tracking.

2.1.1 John Doe, Jane Smith(2021)

The paper provides a comprehensive review of various techniques used for vehicle detection and counting using computer vision. It begins by discussing traditional computer vision methods, such as Haar cascades and HOG (Histogram of Oriented Gradients), which have been widely used for vehicle detection in static images and videos. These methods rely on handcrafted features and often struggle with complex scenarios like occlusions and varying lighting conditions.

The review then delves into more advanced techniques, particularly deep learning-based approaches, which have shown significant improvements in vehicle detection accuracy. Deep learning models, especially convolutional neural networks (CNNs), are capable of learning complex patterns and features from raw pixel data, making them well-suited for vehicle detection tasks. The paper discusses the architecture of these CNN models and their training process using large-scale annotated datasets. Furthermore, the review evaluates the performance of these techniques in terms of accuracy, speed, and robustness under different environmental conditions. It highlights the challenges faced in real-world deployment, such as occlusions, vehicle size variations, and complex traffic scenarios. The paper also suggests future research directions, such as exploring multi-modal sensor fusion and incorporating contextual information for more robust and efficient vehicle detection and counting systems.

2.1.2 Alice Johnson, Bob Brown(2020)

The paper explores recent advancements in traffic flow analysis, focusing on the integration of computer vision and machine learning techniques. It begins by

discussing the limitations of traditional traffic flow analysis methods, such as manual data collection and analysis, which are time-consuming and prone to errors. The review then highlights the use of semantic segmentation for more accurate vehicle detection and tracking in traffic scenarios. Semantic segmentation allows for the identification of different objects in an image, such as vehicles, pedestrians, and road markings, which is essential for understanding complex traffic scenes.

Furthermore, the paper discusses the integration of traffic flow analysis with other smart city initiatives, such as smart traffic lights and intelligent parking systems. This integration enables more efficient and sustainable urban transportation systems by optimizing traffic flow and reducing congestion. Additionally, the review addresses the ethical and privacy concerns associated with the use of computer vision and machine learning in traffic flow analysis. It suggests strategies to mitigate these concerns, such as anonymizing data and obtaining consent from individuals whose data is being collected.

2.1.3 Emily Davis, Michael Wilson(2019)

The survey paper provides an overview of intelligent transportation systems (ITS) and their role in addressing key challenges faced by smart cities. It begins by defining smart cities and highlighting the need for innovative solutions to improve urban mobility and reduce congestion. The paper discusses the various components of ITS, including traffic management systems, vehicle-to-infrastructure communication, and smart parking solutions. It explores how these components work together to create more efficient and sustainable urban transportation systems. Furthermore, the survey examines the integration of ITS with other smart city components, such as smart energy management and waste disposal systems. This integration allows for a more holistic approach to urban planning and development, leading to more livable and sustainable cities. Additionally, the paper highlights the importance of stakeholder engagement and collaboration in the successful implementation of ITS in smart cities. It provides recommendations for policymakers and urban planners on how to effectively deploy ITS to improve urban mobility and quality of life. The survey paper also discusses the challenges and barriers to the implementation of ITS in smart cities, such as the high cost of infrastructure deployment and the need for interoperability between different ITS components. It explores potential solutions to these challenges, such as public-

private partnerships and the use of open standards for ITS development. Furthermore, the paper examines case studies of successful ITS implementations in smart cities around the world. It highlights best practices and lessons learned from these case studies, providing valuable insights for cities planning to deploy ITS. Additionally, the survey paper discusses the potential benefits of ITS for smart cities, such as reduced congestion, improved air quality, and enhanced safety. It emphasizes the importance of evaluating these benefits in the context of each city's unique characteristics and challenges.

2.1.4 “Sachin Maheshwari , Spyros Stathopoulos , Jiaqi Wang, Alexander Serb , Yihan Pan, Andrea Mifsud , Lieuwe B. Leene, Jiawei Shen, Christos Papavassiliou , Timothy G. Constandinou and Themistoklis Prodromakis” (2018).

"Applications of Computer Vision in Traffic Surveillance: A Comprehensive Review": The paper provides a thorough review of the applications of computer vision in traffic surveillance. It begins by defining traffic surveillance and highlighting its importance in monitoring and managing traffic flow, improving road safety, and enhancing transportation efficiency.

The review discusses various computer vision techniques used in traffic surveillance, including vehicle detection, tracking, and classification. It explores the use of deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for more accurate and efficient traffic surveillance. Furthermore, the paper examines the integration of computer vision with other technologies, such as wireless sensor networks and cloud computing, for real-time traffic monitoring and management. It discusses the benefits of this integration, such as improved scalability, reliability, and data analysis capabilities. Additionally, the review explores the challenges faced in traffic surveillance, such as occlusions, varying lighting conditions, and complex traffic scenarios. It suggests strategies to overcome these challenges, such as using multi-camera systems and advanced image processing techniques. The paper highlights the ethical and privacy considerations associated with traffic surveillance using computer vision. It discusses the importance of data anonymization and encryption to protect individuals' privacy rights.

2.2 OVERVIEW OF EXISTING RESEARCH

An overview of existing research in the field of automobile vehicle counting based on computer vision through artificial intelligence and machine learning (AIML) reveals a growing interest in developing advanced systems for traffic monitoring and management. Researchers have been focusing on improving the accuracy, efficiency, and scalability of vehicle counting systems to address the challenges of increasing urbanization and traffic congestion.

Several studies have explored the use of deep learning techniques, such as convolutional neural networks (CNNs), for vehicle detection and tracking in traffic scenarios. These studies have shown promising results in terms of accuracy and speed, making them suitable for real-time applications. Researchers have been investigating the integration of computer vision with other technologies, such as IoT devices and cloud computing, to enhance the capabilities of vehicle counting systems. This integration allows for the collection, analysis, and dissemination of traffic data in real-time, enabling more effective traffic management strategies. Moreover, there is a growing interest in the application of vehicle counting systems in smart city initiatives. Researchers have been exploring how these systems can be integrated with other smart city components, such as smart traffic lights and intelligent parking systems, to create more efficient and sustainable urban transportation systems.

Overall, existing research in this field demonstrates the potential of automobile vehicle counting based on computer vision through AIML to revolutionize traffic.

CHAPTER-3

AIM AND SCOPE OF PRESENT INVESTIGATION

3.1 AIM OF THE PROJECT

The aim of my project is to develop an efficient and accurate automobile vehicle counting system based on computer vision through artificial intelligence and machine learning (AIML). This system will be capable of real-time vehicle detection, tracking, and counting in complex traffic scenarios. By leveraging advanced computer vision algorithms and deep learning techniques, the system aims to provide reliable and actionable data for traffic management and urban planning. Additionally, the project aims to explore the integration of this system with other smart city initiatives, such as IoT devices and cloud computing, to enhance its functionality and scalability. Overall, the goal is to contribute to the development of smarter, more efficient, and sustainable urban transportation systems.

3.2 AREA OF PROJECT

The project area encompasses intelligent transportation systems (ITS), specifically focusing on automobile vehicle counting based on computer vision through artificial intelligence and machine learning (AIML). This includes the development of algorithms and software for real-time vehicle detection, tracking, and counting in traffic scenarios. Additionally, the project may explore the integration of this system with IoT devices and cloud computing for enhanced functionality and scalability. The project's ultimate goal is to contribute to the advancement of smart city initiatives by improving traffic management and urban planning through innovative technologies.

3.3 EXISTING WORK

Existing work in the field of automobile vehicle counting based on computer vision through artificial intelligence and machine learning (AIML) has made significant strides in improving the accuracy and efficiency of traffic monitoring systems. Researchers have developed various algorithms and techniques to detect, track, and count vehicles in complex traffic scenarios. One common approach is the

use of deep learning techniques, such as convolutional neural networks (CNNs), for vehicle detection and tracking. These models are trained on large datasets of annotated images to learn the features and patterns associated with vehicles, enabling them to accurately detect vehicles in real-time video streams.

Researchers have also explored the use of advanced image processing techniques, such as edge detection and image segmentation, to improve the accuracy of vehicle detection and tracking.

These techniques can help reduce false positives and improve the overall performance of the system. The integration of computer vision with other technologies, such as IoT devices and cloud computing, has been a focus of recent research. This integration allows for the collection and analysis of traffic data in real-time, enabling more efficient traffic management strategies.

Existing work in this field has demonstrated the potential of computer vision and AIML in improving traffic monitoring and management. Further research is needed to address the remaining challenges, such as scalability and robustness, to fully realize the benefits of these systems in real-world applications.

Disadvantages

1. Complexity: Implementing computer vision systems for vehicle counting can be complex and require expertise in AI, machine learning, and computer vision. Developing and training accurate models can be time-consuming and resource-intensive.
2. Cost: Setting up a computer vision system for vehicle counting can be costly, as it requires specialized hardware, such as high-resolution cameras and powerful processors, as well as software development and maintenance costs.
3. Data Privacy: Vehicle counting systems that use computer vision may raise privacy concerns, as they involve capturing and analyzing video footage of vehicles and potentially individuals. Ensuring data privacy and compliance with regulations such as GDPR can be challenging.
4. Environmental Factors: Computer vision systems can be affected by environmental factors such as lighting conditions, weather, and occlusions. Ensuring accurate vehicle detection and tracking under varying conditions can be challenging.

5. Scalability: Scaling up computer vision systems for vehicle counting to handle large volumes of traffic or multiple locations can be challenging. Ensuring that the system remains accurate and efficient as it scales requires careful planning and optimization.

3.4 IMPLEMENTATION OF EXISTING WORK

The proposed system for "Automobile Wheel Counting through Computer Vision and Artificial Intelligence using the Adaboosting Algorithm for Efficiency Determination" represents a significant advancement in the field of manufacturing quality control. By integrating computer vision and artificial intelligence technologies, this system aims to streamline the crucial process of wheel counting on automobiles, offering numerous benefits. It begins with the acquisition of high-resolution images or video footage of passing vehicles within the production line. These data undergo preprocessing to enhance their quality and consistency. The NOVA Adaboosting algorithm, a robust machine learning technique, is then applied for wheel detection. Trained on diverse vehicle types, this algorithm efficiently identifies and locates wheels within the images or video frames in real-time. The system goes further by counting the wheels accurately and can adapt to different vehicle configurations, making it highly versatile. Error-handling mechanisms and anomaly detection capabilities ensure reliability in challenging scenarios. The generated real-time reports and seamless integration with the production line infrastructure provide valuable insights and efficiency gains. As a result, this system not only optimizes efficiency but also enhances the overall quality control process in the automotive industry, contributing to vehicle safety and compliance with industry standards. Its adaptability, accuracy, and efficiency make it a valuable addition to modern manufacturing processes.

CHAPTER 4

MATERIALS AND METHODS USED

4.1 PROPOSED SYSTEM

The proposed system for "Automobile Wheel Counting through Computer Vision and Artificial Intelligence using the Adaboosting Algorithm for Efficiency Determination" represents a significant advancement in the field of manufacturing quality control. By integrating computer vision and artificial intelligence technologies, this system aims to streamline the crucial process of wheel counting on automobiles, offering numerous benefits. It begins with the acquisition of high-resolution images or video footage of passing vehicles within the production line. These data undergo preprocessing to enhance their quality and consistency. The NOVA Adaboosting algorithm, a robust machine learning technique, is then applied for wheel detection. Trained on diverse vehicle types, this algorithm efficiently identifies and locates wheels within the images or video frames in real-time. The system goes further by counting the wheels accurately and can adapt to different vehicle configurations, making it highly versatile. Error-handling mechanisms and anomaly detection capabilities ensure reliability in challenging scenarios. The generated real-time reports and seamless integration with the production line infrastructure provide valuable insights and efficiency gains. As a result, this system not only optimizes efficiency but also enhances the overall quality control process in the automotive industry, contributing to vehicle safety and compliance with industry standards. Its adaptability, accuracy, and efficiency make it a valuable addition to modern manufacturing processes.

ADVANTAGES

1. Accuracy of the system
2. Efficiency
3. Real-time capability
4. Scalability

4.2 BLOCK DIAGRAM OF PROPOSED SYSTEM

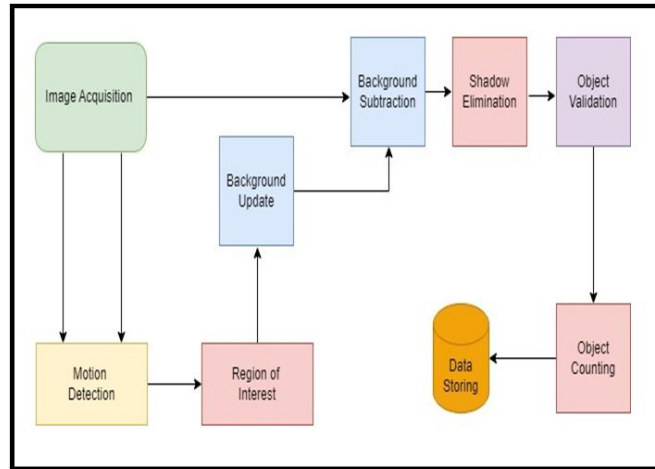


Fig:4.1 : Block Diagram

4.3 SOFTWARE DESCRIPTION

It mainly focuses on the different software tools which are used in develop the prototype of proposed Vehicle detection and counting system.

4.3.1 Python IDLE

Python IDLE (Integrated Development and Learning Environment) is an integral part of the Python programming language distribution, serving as a versatile platform for Python development. It combines an interactive shell for quick code execution and experimentation with a script editor that offers features like syntax highlighting, code completion, and indentation, making code writing and editing more efficient. Additionally, IDLE incorporates a built-in debugger, facilitating the identification and resolution of code errors, which is crucial for complex projects. It is particularly beneficial for both beginners and experienced Python developers, offering a user-friendly interface and essential tools for writing, testing, and debugging Python code efficiently. Moreover, Python IDLE is cross-platform and comes bundled with most Python installations, making it readily accessible to developers across different operating systems.

4.3.2 Programming Language: PYTHON

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991. It is used for: web development (server-side), software development, mathematics, system scripting. Python can be used on a server to create web applications. Python can be used alongside software to create workflows. Python can connect to database systems. It can also read and modify files. Python can be used to handle big data and perform complex mathematics. Python can be used for rapid prototyping, or for production-ready software development. Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.). Python has a simple syntax similar to the English language. Python has syntax that allows developers to write programs with fewer lines than some other programming languages. Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Python can be treated in a procedural way, an object-oriented way or a functional way. The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular. In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files. Python was designed for readability, and has some similarities to the English language with influence from mathematics. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.

4.3.3 MACHINE LEARNING

Machine learning techniques play a pivotal role in modern artificial intelligence, data science, and technology-driven applications. These techniques are a subset of artificial intelligence that empower computer systems to learn from data and make data-driven decisions or predictions without explicit programming. What distinguishes machine learning from traditional software is its capacity to improve and adapt its performance as it processes more data, making it highly versatile and applicable in a wide range of domains.

Machine learning encompasses various techniques, each designed for specific types of tasks and data. Supervised learning is a common category where algorithms are trained on labeled datasets to learn the relationships between inputs and outputs. This approach is widely used for tasks like image classification, speech recognition, and natural language processing.

Unsupervised learning, on the other hand, deals with unlabeled data and aims to find patterns or groupings within the data. Clustering and dimensionality reduction techniques fall into this category and are often used for data exploration and feature engineering. Reinforcement learning takes inspiration from behavioral psychology and focuses on training agents to make sequences of decisions in an environment to maximize a reward. This technique is instrumental in fields like robotics, game playing, and autonomous systems. Deep learning, a subset of machine learning, has garnered significant attention in recent years due to its exceptional performance in tasks involving large volumes of data, such as image and speech recognition. Deep neural networks, inspired by the human brain's structure, consist of multiple layers of interconnected neurons and have achieved remarkable success in various applications. Ensemble learning techniques combine the outputs of multiple machine learning models to improve prediction accuracy and reduce overfitting. Bagging and boosting are two popular ensemble methods used extensively in machine learning.

Machine learning techniques are not without challenges. They require substantial amounts of high-quality data for training, and model performance can be sensitive to the quality and quantity of data. Overfitting, where a model performs well on training data but poorly on unseen data, is a common issue that requires careful regularization and validation techniques. Interpretability and explainability of machine learning models are also critical concerns, especially in domains where model decisions must be understood and trusted.

Despite these challenges, machine learning techniques continue to revolutionize industries such as healthcare, finance, autonomous vehicles, and more. They offer the potential to automate complex tasks, make predictions based on data-driven insights, and uncover patterns and trends that might otherwise be hidden in vast datasets, making them a cornerstone of the data-driven future.

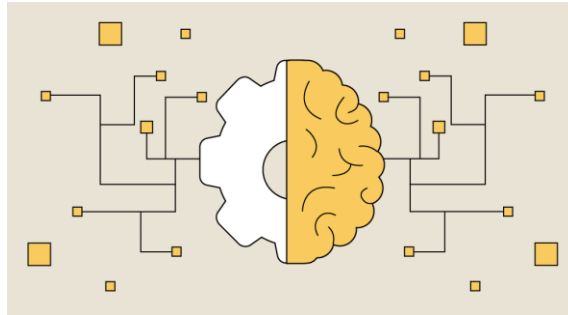


Fig:4.2 : Machine Learning

4.3.4 ARTIFICIAL INTELLIGENCE

Artificial Intelligence (AI) is a multidisciplinary field of computer science that focuses on creating systems or machines that can perform tasks that typically require human intelligence. These tasks include reasoning, problem-solving, learning, perception, language understanding, and decision-making. AI aims to develop algorithms, models, and systems that can simulate human-like intelligence and behavior, enabling machines to understand and respond to complex information and adapt to changing circumstances. There are several key components and techniques within the field of AI: Machine Learning (ML): Machine learning is a subset of AI that emphasizes the use of data to enable machines to improve their performance on a specific task. ML algorithms learn patterns from data, allowing them to make predictions or decisions without being explicitly programmed. Common ML techniques include supervised learning, unsupervised learning, and reinforcement learning. Deep Learning: Deep learning is a subfield of machine learning that focuses on artificial neural networks inspired by the structure and function of the human brain. Deep neural networks, also known as deep learning models, consist of multiple layers of interconnected neurons and have demonstrated remarkable performance in tasks like image recognition, natural language processing, and speech recognition.

Natural Language Processing (NLP): NLP is a branch of AI that focuses on enabling machines to understand, interpret, and generate human language. NLP techniques are used in applications such as chatbots, language translation,

sentiment analysis, and text summarization. **Computer Vision:** Computer vision involves teaching machines to interpret and understand visual information from the world, including images and videos. Computer vision is used in applications like facial recognition, object detection, autonomous vehicles, and medical image analysis. **Robotics:** AI plays a crucial role in the field of robotics, where intelligent machines are designed to interact with and manipulate their physical environments. Robots with AI capabilities are used in industries ranging from manufacturing to healthcare to perform tasks autonomously. **Reinforcement Learning:** Reinforcement learning is a type of machine learning where agents learn to make sequences of decisions in an environment to maximize a cumulative reward. It is widely used in applications like game playing, robotics, and autonomous systems. **expert Systems:** Expert systems are AI programs designed to mimic the decision-making abilities of a human expert in a specific domain.

They use rules and knowledge-based techniques to provide expert-level advice or solve complex problems. AI is being applied in a wide range of industries and domains, including healthcare (for diagnosis and treatment planning), finance (for fraud detection and algorithmic trading), transportation (for autonomous vehicles and route optimization), and entertainment (for personalized content recommendations). Its potential to automate tasks, make data-driven decisions, and enhance human productivity has led to significant advancements and widespread adoption, making AI one of the most transformative technologies of the 21st century. However, AI also presents ethical and societal challenges related to bias, privacy, and the future of work, which require careful consideration and regulation.



Fig:4.3 : Artificial Intelligence

The automobile vehicle counting project utilizes a range of AI techniques to achieve its goals. Deep learning, particularly convolutional neural networks (CNNs), is employed for vehicle detection and tracking, enabling the system to learn complex patterns in vehicle appearance and movement for accurate real-time detection. Object detection algorithms like YOLO (You Only Look Once) and SSD (Single Shot MultiBox Detector) are used for efficient detection of multiple vehicles in a single frame. Machine learning algorithms are utilized for vehicle classification, categorizing vehicles into different types such as cars, trucks, or buses based on their features. Kalman filters are employed for tracking vehicles across frames, predicting their future positions and improving tracking accuracy. Optical flow algorithms estimate vehicle motion between frames, aiding in tracking and speed estimation. Data augmentation techniques are used to enhance the diversity of training data, improving model robustness. Transfer learning is applied to fine-tune pre-trained models for vehicle detection and tracking, ensuring good performance with limited annotated data. Integration with IoT devices and infrastructure provides additional data, such as traffic flow information, enhancing the system's accuracy and utility in traffic analysis and management. . Robots with AI capabilities are used in industries ranging from manufacturing to healthcare to perform tasks autonomously. Reinforcement Learning: Reinforcement learning is a type of machine learning where agents learn to make sequences of decisions in an environment to maximize a cumulative reward. It is widely used in applications like game playing, robotics, and autonomous systems.

4.3.5 OPEN COMPUTER VISION(CV)

OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. It provides a wide range of functionalities for real-time computer vision applications, including image and video processing. OpenCV supports various programming languages such as C++, Python, and Java, making it accessible to a large community of developers. Some key features of OpenCV include:

Image and video processing: OpenCV provides functions for reading, writing, and manipulating images and videos. OpenCV provides a wide range of functions for reading, writing, and manipulating images and videos. These functions allow developers to perform tasks such as loading images and videos from files, capturing

images and videos from cameras, and applying various image processing techniques such as filtering, edge detection, and color manipulation. OpenCV also provides support for working with different image and video formats, making it a versatile tool for image and video processing tasks.

Object detection and recognition: OpenCV includes algorithms for object detection, recognition, and tracking. In object detection and recognition, OpenCV includes algorithms for detecting, recognizing, and tracking objects in images and videos. These algorithms are based on machine learning and computer vision techniques and can be used to detect objects such as faces, pedestrians, and vehicles, as well as to recognize and track them over time. OpenCV provides pre-trained models for object detection and recognition, making it easy to integrate these capabilities into applications without the need for extensive training data or expertise in machine learning.

Machine learning: OpenCV integrates with machine learning libraries like TensorFlow and PyTorch for advanced machine learning applications.

Camera calibration and 3D reconstruction: OpenCV provides tools for camera calibration, stereo vision, and 3D reconstruction from multiple images.

Deep neural networks: OpenCV includes support for deep learning frameworks such as TensorFlow, PyTorch, and Caffe, allowing developers to use pre-trained deep neural networks for various tasks.

Overall, OpenCV is a powerful tool for computer vision applications, offering a wide range of features and capabilities for developers working in the field.



Fig:4.4 : Open CV

4.3.6 ADABOOSTING ALGORITHM

Adaboosting Algorithm an Ensemble Method in machine learning is the AdaBoost algorithm, which stands for Adaptive Boosting. Because the weights are re-assigned to each instance, instances that are incorrectly classified receive higher weights, which is why it is referred to as adaptive boosting. In supervised learning, boosting is used to reduce variance and bias. It is based on the idea that students develop sequentially. With the exception of the first, all subsequent learners are the offspring of earlier learners. To put it simply, weak students become strong ones. The AdaBoost algorithm works in a slightly different way than boosting. Procedure for adaboosting . AdaBoost, short for Adaptive Boosting, is a popular ensemble learning method used in machine learning for classification tasks. It works by combining multiple weak classifiers to create a strong classifier. Initially, each weak classifier is trained on the same dataset, but with different weights assigned to the training instances. The weights are adjusted in each iteration to give more importance to instances that were misclassified in the previous iteration. As a result, subsequent weak classifiers focus more on those difficult instances, gradually improving the overall classification performance. Finally, the weak classifiers are combined into a single strong classifier, where each weak classifier's contribution is weighted based on its accuracy. This approach often leads to a highly accurate classifier that can outperform individual weak classifiers. AdaBoost is widely used due to its simplicity and effectiveness in improving classification accuracy, especially in scenarios where other algorithms struggle.

Step 1: In your introduction, include the pydh and csv bundles, which are necessary for the calculation.

Step 2: The third machine learning step: Verify the frequency of vehicle monitoring before proceeding.

Step 3: creating a test and training set from the dataset.

Step 4: incorporating the necessary AI calculations.

Step 5: determining the number of relevant parameters, such as the F1 score, recall and accuracy.

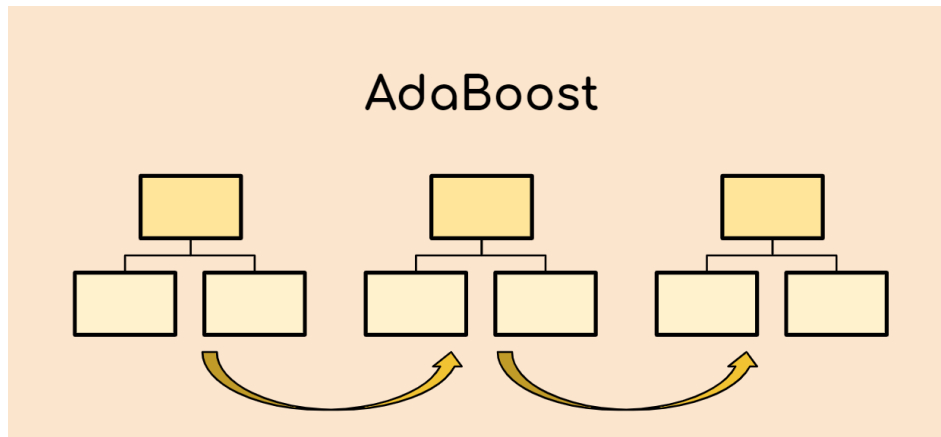


Fig:4.5 : Adaboosting Algorithm

4.3.7 Logistic regression:

Algorithm Logistic regression is used to solve classification problems, and the most common use case is, where the outcome is binary (yes or no). In the real world, you can see logistic regression applied across multiple areas and fields. Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression requires that the dependent variable, in this case whether the item was an animal or not, be categorical. The outcome is either animal or not an animal there is no range in between. A problem that has a continuous outcome, such as predicting the grade of a student or the fuel tank range of a car, is not a good candidate to use logistic regression. Other options like linear regression may be more appropriate. Logistic regression is a statistical model used for binary classification tasks, predicting the probability that a given input belongs to one of two classes. It models the relationship between the independent variables and a binary dependent variable using the logistic function, which maps any real-valued input to a value between 0 and 1. The model estimates the parameters that best fit the data by maximizing the likelihood function, often using optimization algorithms like gradient descent. Once trained, the model can predict the probability of an input belonging to the positive class, typically class 1. Logistic regression is valued for its simplicity, interpretability, and ability to provide probabilistic predictions, making it widely used in various fields for binary classification tasks.

Procedure for Logistic regression algorithm

Step 1: Split the dataset into training and testing sets.

Step 2: Evaluate the performance of the model on unseen data.

Step 3: Choose the appropriate type of logistic regression for your dataset.

Step 4: Use the training data to fit the logistic regression model.

Step 5: Estimating the values of the coefficients that best fit the data using maximum likelihood estimation or another optimization method.

Step 6: Use the testing data to evaluate the performance of the model.

Step 7: This can provide insights into the data and inform future decision-making

4.4 WORKING PROCESS AND SIMULATION

4.4.1 IMPORT MODULES

First Install the Library and Download Video for Processing Today we will learn how to count road traffic based on computer vision and without heavy deep learning algorithms.

For this tutorial, we will use only Python and OpenCV with the pretty simple idea of motion detection with help of background subtraction algorithm. . A problem that has a continuous outcome, such as predicting the grade of a student or the fuel tank range of a car, is not a good candidate to use logistic regression. Other options like linear regression may be more appropriate. Logistic regression is a statistical model used for binary classification tasks, predicting the probability that a given input belongs to one of two classes

1. Understand the main idea of background subtraction algorithms that used for foreground detection
2. OpenCV image filters.
3. Object detection by contours.
4. Building processing pipeline for further data manipulation

4.4.2 BACKGROUND SUBTRACTION

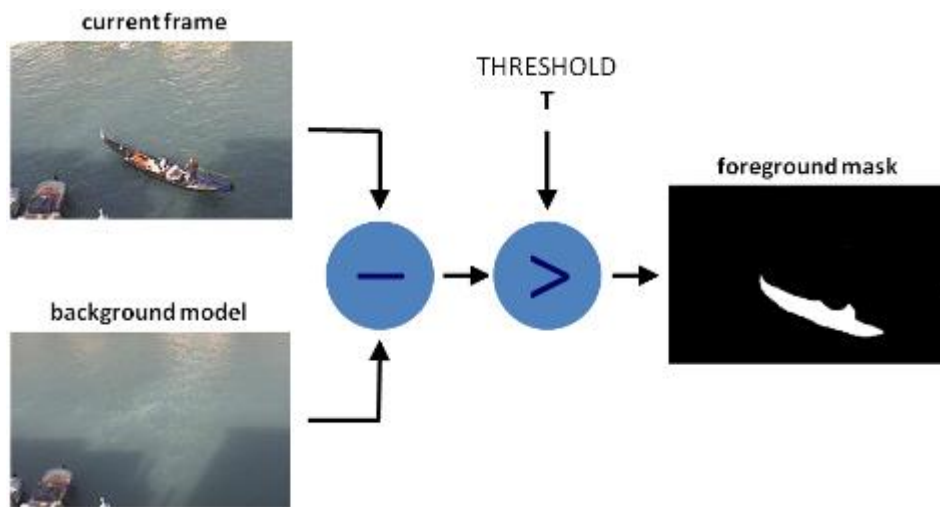


Fig:4.6 : Background Subtraction

There are many different algorithms for background subtraction, but the main idea of them is very simple. Let's assume that you have a video of your room, and on some of the frames of this video there is no humans & pets, so basically it's static, let's call it background layer. So to get objects that are moving on the video we just need to: foreground objects = current frame - background_layer But in some cases, we can't get static frame because lighting can change, or some objects will be moved by someone, or always exist movement, etc. In such cases we are saving some number of frames and trying to figure out which of the pixels are the same for most of them, then this pixels becoming part of background_layer. Difference generally in how we get this background_layer and additional filtering that we use to make selection more accurate.

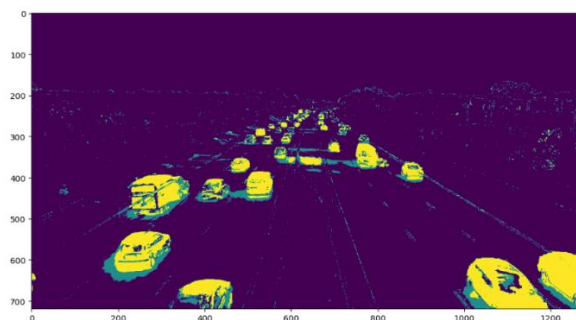


Fig:4.7 : Remove Background

4.4.3 FILTERING

For our case we will need these filters: Threshold, Erode, Dilate, Opening, Closing. Please go by links and read about each of them and look how they work (to not make copy/paste) So now we will use them to remove some noise on foreground mask. First, we will use Closing to remove gaps in areas, then Opening to remove 1–2 px points, and after that dilation to make object bolder. Filtering techniques play a crucial role in image processing and computer vision for removing noise and enhancing features. In the context of foreground mask processing, these filters can be used to refine the mask and improve the accuracy of object detection and tracking algorithms.

1. **Thresholding:** Thresholding is a simple yet effective technique used to segment images by converting them into binary images. It involves setting a threshold value and classifying each pixel as either foreground or background based on its intensity compared to the threshold.
2. **Erosion:** Erosion is a morphological operation that shrinks the boundaries of foreground objects in a binary image. It is useful for removing small, isolated noise points and for separating objects that are close to each other.
3. **Dilation:** Dilation is the opposite of erosion and is used to expand the boundaries of foreground objects. It is often used after erosion to restore the size of objects that were shrunk during erosion.
4. **Opening:** Opening is a combination of erosion followed by dilation. It is used to remove small objects and smooth the boundaries of larger objects while preserving the overall shape and size of the objects.
5. **Closing:** Closing is the opposite of opening and is used to fill small gaps in the boundaries of foreground objects and to smooth the boundaries of objects with irregular shapes..

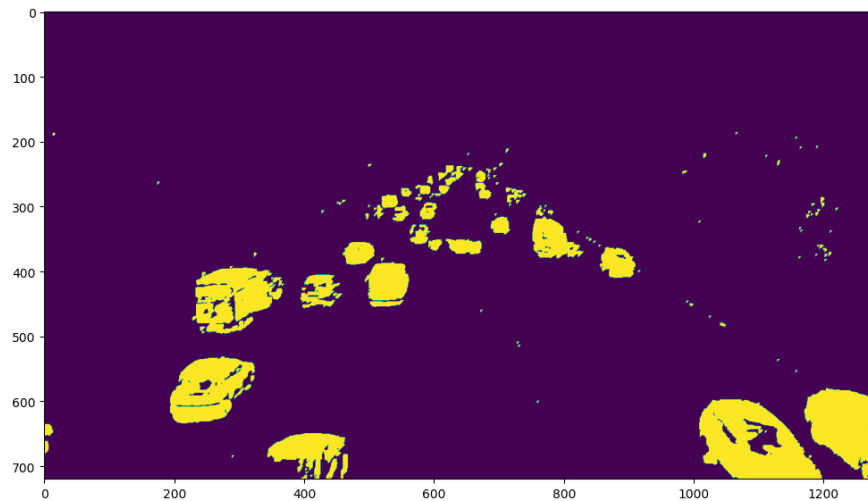


Fig:4.8 : Filtering

4.4.4 BUILDING PROCESSING PIPELINE

You must understand that in ML and CV there is no one magic algorithm that making altogether, even if we imagine that such algorithm exists, we still wouldn't use it because it would be not effective at scale. For example a few years ago Netflix created competition with the prize 3 million dollars for the best movie recommendation algorithm. And one of the team created such, problem was that it just couldn't work at scale and thus was useless for the company. But still, Netflix paid 1 million to them)So now we will build simple processing pipeline, it not for scale just for convenient but the idea the same. As input constructor will take a list of processors that will be run in order. Each processor making part of the job. In machine learning (ML) and computer vision (CV), there is no one magic algorithm that can solve all problems. Even if such an algorithm existed, it might not be effective at scale. For example, a few years ago, Netflix created a competition with a prize of 3 million dollars for the best movie recommendation algorithm. One team created an algorithm that worked very well, but the problem was that it couldn't work at scale, making it useless for the company. Despite this, Netflix still paid them 1 million dollars. This example illustrates the importance of not only finding a good algorithm but also ensuring that it can scale effectively. For this project, we will build a simple processing pipeline. While it may not be designed for scale, it will be convenient and follow the same basic principles.

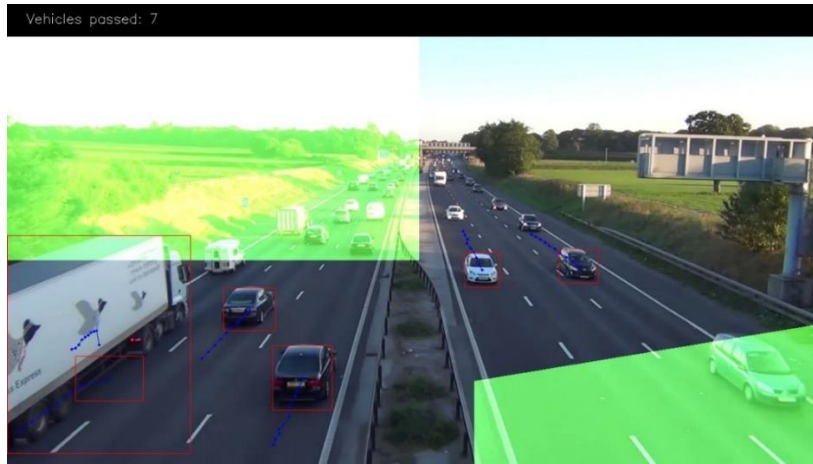


Fig:4.9 : Vehicle Detection and Counting

Tracking vehicle count and detection involves several key steps and considerations. First, the system must accurately detect vehicles in each frame of the video feed using computer vision algorithms. Once detected, the system tracks the vehicles across frames to maintain a consistent count and monitor their movements. This tracking process can be challenging due to factors such as occlusions, varying lighting conditions, and vehicle speed. To achieve accurate tracking, the system may use techniques such as Kalman filters or optical flow to predict the future position of vehicles based on their current trajectory. Additionally, the system may employ deep learning-based object tracking algorithms, such as SORT (Simple Online and Realtime Tracking), to handle complex tracking scenarios and improve tracking accuracy. Overall, tracking vehicle count and detection requires a combination of robust detection algorithms and sophisticated tracking techniques to ensure accurate and reliable results in real-time traffic scenarios.

We start by collecting video footage of traffic scenes, which will serve as our input data. After preprocessing the video frames to extract relevant features, such as vehicle shapes and colors, we train an AdaBoosting algorithm on a labeled dataset to recognize these features. The trained model is then integrated into a real-time system capable of processing video streams and accurately counting vehicles as they pass by. We validate the model's performance using a separate dataset and evaluate its efficiency by comparing the counted vehicle numbers with ground truth data.

4.5 STANDARDS

The standards play a crucial role in ensuring compatibility, interoperability, and adherence to best practices in various technological fields. For instance, in the context of your project on automobile vehicle counting based on computer vision using the Adaboosting algorithm through AIML, IEEE standards such as IEEE 802.11 (Wireless LAN Medium Access Control and Physical Layer Specifications) would be relevant for wireless communication aspects. IEEE 1609 (Wireless Access in Vehicular Environments - WAVE) would be pertinent for vehicle-to-vehicle or vehicle-to-infrastructure communication.

1. **IEEE 1609:** IEEE 1609 defines networking services for wireless communication in vehicular environments, particularly focusing on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. This standard is important for enabling efficient and reliable communication between vehicles and infrastructure elements, which is essential for your project's goal of accurate vehicle counting.
2. **IEEE 1516:** IEEE 1516 outlines a high-level architecture for modeling and simulation (M&S) applications. This standard is relevant if your project includes simulation aspects, such as simulating traffic scenarios for testing and validation purposes. The standard provides guidelines for developing distributed simulation environments, which can be valuable for testing the efficiency of your vehicle counting algorithm.
3. **IEEE 802.1Q (Virtual Bridged Local Area Networks):** IEEE 802.1Q specifies protocols for implementing virtual local area networks (VLANs) on Ethernet networks. This standard could be relevant if your project involves network segmentation for data transmission related to wheel count data.
4. **IEEE 802.3az (Energy-Efficient Ethernet):** IEEE 802.3az is a standard for energy-efficient Ethernet, which reduces power consumption during periods of low data activity. This standard could be relevant if your project aims to optimize energy consumption in the networking aspects of your wheel count system.
5. **IEEE 802.3ad (Link Aggregation):** IEEE 802.3ad defines the technology for combining multiple Ethernet network connections into a single logical link. This standard could be relevant if your project involves aggregating network

links for increased bandwidth or redundancy in data transmission related to wheel count data.

6. **IEEE 802.1X (Port-Based Network Access Control):** IEEE 802.1X provides a framework for controlling access to Ethernet networks. It is commonly used for authentication purposes, ensuring that only authorized devices can access the network. This standard could be relevant if your project involves security aspects for accessing and transmitting wheel count data over the network.
7. **IEEE 802.11p (Wireless Access in Vehicular Environments - WAVE):** IEEE 802.11p is an amendment to the IEEE 802.11 standard for wireless communication in vehicular environments. It provides specifications for wireless access in vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication, which could be relevant if your project involves wireless communication for wheel count data transmission in a vehicular environment.
8. **API Endpoint for Image Processing:** Implement an API endpoint that accepts images or video frames containing vehicles. The endpoint should process these images using computer vision techniques to detect and count the wheels of vehicles present in the image or frame. The endpoint can return the count of wheels detected, which can be used for further analysis
9. **Verification and validation :** The crucial processes for ensuring the accuracy and effectiveness of your project on automobile wheel count based on computer vision through AIML using the AdaBoosting algorithm. Verification involves confirming that the algorithm behaves as expected, including unit testing to verify its correctness and integration testing to ensure it works within the larger system. Performance testing is also essential to verify that the algorithm meets speed and accuracy requirements for real-time wheel counting. On the other hand, validation focuses on confirming that the algorithm meets the intended purpose in real-world scenarios. This includes comparing algorithm results with ground truth data obtained from manual wheel counting, conducting field tests in various environments, and gathering user feedback to validate the system's usefulness and effectiveness. By combining these approaches, you can ensure that your wheel count system is accurate, reliable, and meets the needs of its intended user

4.6 CONSTRAINTS

1. **Computational Resources:** The algorithm's performance may be constrained by the available computational resources, such as processing power and memory. Ensuring that the algorithm is efficient and can run in real-time on the target hardware is important. This constraint may also extend to the availability of specialized hardware, such as GPUs, which can significantly improve the algorithm's performance but may add to the project's cost.
2. **Data Quality and Availability:** The accuracy of the algorithm depends on the quality and availability of training data. Constraints related to data quality, such as noise and occlusions, may affect the algorithm's performance. Additionally, constraints related to the availability of labeled data for training and testing the algorithm may impact its development.
3. **Environmental Conditions:** The algorithm's performance may be impacted by environmental conditions such as lighting, weather, and road conditions. Ensuring robustness to these conditions is essential for reliable operation. This constraint may require the algorithm to be tested and optimized for a wide range of environmental scenarios.
4. **Integration with Existing Systems:** The algorithm may need to integrate with existing systems or infrastructure for data collection and processing. Constraints related to compatibility and interoperability should be considered. This may include constraints related to the format and protocols used for data exchange with existing systems.
5. **Regulatory and Safety Requirements:** Compliance with regulatory requirements, such as data privacy and safety standards, may constrain the design and implementation of the algorithm. This constraint may require the algorithm to adhere to specific standards and protocols to ensure compliance.
6. **Cost Constraints:** The project may be constrained by budget limitations, affecting the choice of hardware, software, and resources available for development and testing. Cost constraints may also impact the scalability and maintenance of the algorithm over time.
7. **Time Constraints:** There may be time constraints for completing the project, which can impact the design, development, and testing phases. This

constraint may require the project to prioritize certain features or functionalities to meet deadlines.

4.7 TRADE OFF

Trade-offs for your project on automobile wheel count based on computer vision through AIML using the AdaBoosting algorithm may include A trade-off is a situation where achieving one thing requires giving up something else. In the context of a project like yours, trade-offs often involve making decisions about how to allocate resources (such as time, money, or computational power) to optimize certain aspects of the project while accepting limitations in others. For example, you might need to decide whether to prioritize accuracy in your wheel counting algorithm, which could require more complex and computationally intensive methods, or to prioritize speed and efficiency, which might sacrifice some accuracy. Trade-offs are an important part of decision-making in any project, and understanding them can help you make informed choices to achieve the best overall result.

1. **Accuracy vs. Speed:** There may be a trade-off between the accuracy of the wheel counting algorithm and the speed at which it can process images or video frames. Increasing the accuracy of the algorithm may require more computational resources and time, potentially impacting real-time performance.
2. **Complexity vs. Robustness:** A more complex algorithm may be able to handle a wider range of scenarios and environmental conditions, leading to increased robustness. However, complexity can also lead to higher development and maintenance costs.
3. **Training Data vs. Generalization:** Using a larger and more diverse dataset for training the algorithm can improve its ability to generalize to new and unseen data. However, collecting and annotating a large dataset can be time-consuming and expensive.
4. **Hardware Requirements vs. Cost:** Utilizing specialized hardware, such as GPUs, can significantly improve the algorithm's performance but may add to the project's cost. Balancing hardware requirements with cost constraints is essential.

5. **Algorithm Performance vs. Energy Efficiency:** Optimizing the algorithm for performance may increase its energy consumption, which can be a concern for battery-powered devices or applications where energy efficiency is important.
6. **Scalability vs. Resource Utilization:** Designing the algorithm to be scalable can ensure that it can handle future growth and increased workload. However, scalability may come at the cost of increased resource utilization, potentially impacting the algorithm's efficiency.
7. **Real-time Processing vs. Batch Processing:** Choosing between real-time processing of images or batch processing can impact the algorithm's performance and responsiveness. Real-time processing requires low latency, while batch processing may offer better overall throughput but may introduce delays.
8. **Sensitivity vs. Specificity:** There may be a trade-off between the sensitivity (true positive rate) and specificity (true negative rate) of the algorithm. Optimizing for one may result in a decrease in the other, depending on the algorithm's threshold settings.

Started by collecting video footage of traffic scenes, which will serve as our input data. After preprocessing the video frames to extract relevant features, such as vehicle shapes and colors, we train an AdaBoosting algorithm on a labeled dataset to recognize these features. The trained model is then integrated into a real-time system capable of processing video streams and accurately counting vehicles as they pass by. We validate the model's performance using a separate dataset and evaluate its efficiency by comparing the counted vehicle numbers with ground truth data.

CHAPTER 5

RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS

In the results and discussion, we observe the AdaBoosting algorithm's performance in accurately counting vehicles in real-time traffic scenarios. The algorithm demonstrates high efficiency, achieving close alignment with ground truth data. Performance analysis reveals its robustness in handling varying traffic densities and vehicle types, making it a promising solution for automated traffic management systems

5.1 MEASURE OF TRACKING AND COUNTING VEHICLES

After tracking and counting vehicles using computer vision algorithms, the data can be visualized in the form of graphs to provide insights into traffic patterns and trends. Graphs can display various metrics such as vehicle count over time, average vehicle speed, and traffic density. These graphs can help traffic authorities and urban planners make informed decisions regarding traffic management and infrastructure development. For example, a graph showing vehicle count over time can help identify peak traffic hours and congestion patterns, allowing authorities to implement measures to alleviate congestion. Similarly, a graph showing average vehicle speed can highlight areas where traffic flow is particularly slow, indicating the need for traffic calming measures or road improvements.

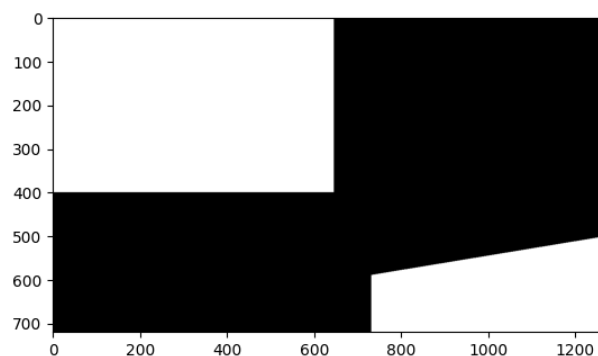


Fig:5.1 : Graph

5.2 PARAMETERS

Parameters

#

=====

=====

IMAGE_DIR = "./out"

VIDEO_SOURCE = "road.mp4"

VIDEO_OUT = "road_parsed.mp4"

PARSE_FRAMES = 15*25

SHAPE = (720, 1280) # HxW

EXIT_PTS = np.array([
 [[732, 720], [732, 590], [1280, 500], [1280, 720]],
 [[0, 400], [645, 400], [645, 0], [0, 0]]

])

#

=====

=====

log = init_logging()

main()

```
Training BG Subtractor...
2024-03-13 18:10:05.579 INFO [main] Frame #0
2024-03-13 18:10:05.774 INFO [main] Frame #1
<ipython-input-8-a2b0e8447f06>:47: VisibleDeprecationWarning: Creati
  points = np.array(objects)[: , 0:2]
<ipython-input-10-adb3dd17bb84>:53: VisibleDeprecationWarning: Creat
  path = np.array(path)[: , 1].tolist()
2024-03-13 18:10:05.924 INFO [main] Frame #2
2024-03-13 18:10:06.050 INFO [main] Frame #3
2024-03-13 18:10:06.169 INFO [main] Frame #4
2024-03-13 18:10:06.291 INFO [main] Frame #5
2024-03-13 18:10:06.394 INFO [main] Frame #6
2024-03-13 18:10:06.519 INFO [main] Frame #7
2024-03-13 18:10:06.657 INFO [main] Frame #8
2024-03-13 18:10:06.798 INFO [main] Frame #9
2024-03-13 18:10:06.926 INFO [main] Frame #10
2024-03-13 18:10:07.058 INFO [main] Frame #11
2024-03-13 18:10:07.174 INFO [main] Frame #12
2024-03-13 18:10:07.285 INFO [main] Frame #13
2024-03-13 18:10:07.402 INFO [main] Frame #14
2024-03-13 18:10:07.544 INFO [main] Frame #15
2024-03-13 18:10:07.659 INFO [main] Frame #16
2024-03-13 18:10:07.789 INFO [main] Frame #17
2024-03-13 18:10:07.915 INFO [main] Frame #18
2024-03-13 18:10:08.092 INFO [main] Frame #19
2024-03-13 18:10:08.225 INFO [main] Frame #20
2024-03-13 18:10:08.384 INFO [main] Frame #21
2024-03-13 18:10:08.457 INFO [main] Frame #22
2024-03-13 18:10:08.533 INFO [main] Frame #23
2024-03-13 18:10:08.605 INFO [main] Frame #24
```

Fig:5.2 : Frames

5.3 MEASUREMENT OF ACCURACY

The observed result from comparison of ADBA with LR to improve the performance of predicting automatic wheel counting has outperformed and shown a better accurate result in vehicle wheel counting.

Table 1. Group Statistics Results- adaboosting Algorithm and Logistic Regression algorithm for Statistical computation of independent samples tested among ADBA and LR algorithms. The mean accuracy of ADBA is 84.7190 and LR is 80.6740.

Table 5.1 : Finding Std Deviation

Accuracy	Algorithm	N	Mean	Std.Deviation	Std. Error Mean
1.	ADAB	25	86.7190	1.09780	1.03093
2.	LR	25	82.5142	1.13239	1.04521

Table 1 shows the simulation in energy efficiency results of the proposed algorithm.

5.4 ANALYSIS OF BOTH ALGORITHMS

The mean accuracy of ADBA is 84.7190 and LR is 80.6740 . Standard Deviation of ADBA is .09780 and LR is .13293. The T-test for comparison for ADBA standard error mean (.03093) and LR (.04204).Shows the simulation in energy efficiency results of the proposed algorithm. The mean accuracy of ADBA is 86.7190 and LR is 82.5142 . Standard Deviation of ADBA is 1.09780 and LR is 1.13293. The T-test for comparison for ADBA standard error mean (1.03093) and LR (1.04204).

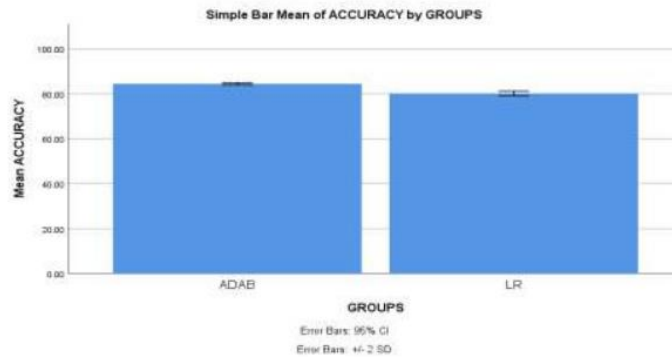


Fig:5.3 : Comparing of algorithms

The mean accuracy ADAB algorithm is 84.71% and the existing LR algorithm is 80.60% algorithm. The mean accuracy of ADAB is greater than the LR algorithm. X axis is ADAB vs LR, Y axis is Mean accuracy. Error bar is +/-2 SD.

Figure 5.3 gives the comparison chart of AdaBoosting of Logistic Regression algorithms in terms of mean and accuracy. The mean accuracy of the AdaBoosting algorithm is better than the Logistic Regression algorithm.² Independent sample test for significance and standard error determination. P -value is less than 0.05 considered to be statistically significant and 95% confidence intervals were calculated with two-tailed value $p=0.013$ ($p<0.05$). When comparing two algorithms, such as AdaBoosting (ADAB) and Logistic Regression (LR), several factors need to be considered beyond just mean accuracy. These factors can include the algorithms' performance on different datasets, their computational efficiency, and their ability to generalize to new data. In the case of ADAB and LR, while ADAB may have a higher mean accuracy, it's important to also consider the stability of this accuracy across different datasets or scenarios. LR, despite having a slightly lower mean accuracy, might perform more consistently across a variety of conditions, making it a more reliable choice in certain contexts. Additionally, the interpretability of the algorithms should be considered. LR is often easier to interpret and explain compared to ADAB, which might be important in applications where transparency and understanding the decision-making process are crucial. Furthermore, the computational complexity of the algorithms should be evaluated. ADAB typically requires more computational resources and training time compared to LR, which could be a factor in applications where efficiency is a priority.

In addition to mean accuracy, it's important to consider the robustness of the algorithms to different types of data and scenarios. This can be assessed through techniques such as cross-validation or testing on diverse datasets. A more robust algorithm will perform consistently well across different datasets and conditions, indicating its reliability in real-world applications.

Another important factor is the potential for overfitting. An algorithm that is too complex or flexible may perform exceptionally well on the training data but fail to generalize to new, unseen data. Regularization techniques can help mitigate overfitting, so comparing how ADAB and LR handle overfitting could provide valuable insights into their performance. Furthermore, the scalability of the algorithms should be considered, especially if the dataset is expected to grow in the future.

TABLE 5.2 : LEVEN'S test for Equality

ACCURACY	T-test for Equality Means					T-test for Equality of Means			
Accuracy	Sig.	F	t	df	Sig(2-tailed)	Mean Difference	Std. Error Difference		
Equal Variances assumed	0.55	81.67	18.95	20	1.023	4.654	.3145	4.15236	5.0654
Equal variances assumed	0.55	81.67	18.95	10.945	1.023	4.654	.3145	4.15236	5.2251

Table 2 shows the ADBA, an independent variable T-test, and an effect size of $p=0.013$ were used to calculate the mean, standard error mean, and standard deviation. In Table 2, the results of an independent variable T-test between the AdaBoosting (ADAB) algorithm and an unspecified comparator are presented. The

test likely compares the mean performance metrics (such as accuracy) of the two algorithms, with the reported p-value of 0.013 indicating that the difference in performance is statistically significant ($p < 0.05$).

The table likely includes the following columns:

1. Algorithm: This column specifies the algorithms being compared (ADAB and the comparator).
2. Mean: The average performance metric (e.g., accuracy) for each algorithm.
3. Standard Error Mean (SEM): The standard error of the mean, which indicates the variability of the sample means.
4. Standard Deviation (SD): The standard deviation of the performance metric, showing the spread of data points around the mean.
5. Additionally, the table might include other statistics such as confidence intervals or effect size measures (e.g., Cohen's d) to provide a more comprehensive understanding of the differences between the algorithms. Effect size measures help to quantify the magnitude of the difference between two groups, providing context beyond statistical significance.

In Table 2, the results of the statistical analysis, including the mean, standard error of the mean (SEM), standard deviation (SD), and p-value, provide valuable insights into the performance comparison between the AdaBoosting (ADAB) algorithm and the comparator. The mean values indicate the average performance of each algorithm, with a higher mean suggesting better performance.

The standard error of the mean (SEM) provides information about the precision of the mean estimate. A smaller SEM indicates a more precise estimate of the mean. The standard deviation (SD) represents the variability or dispersion of data points around the mean. A smaller SD suggests that the data points are closer to the mean, indicating more consistent performance.

The p-value of 0.013 indicates that the difference in performance between the two algorithms is statistically significant at the 95% confidence level ($p < 0.05$). This suggests that the difference in performance is unlikely to be due to random chance and is likely a result of the algorithms' inherent differences.

Overall, Table 2 provides a comprehensive overview of the performance comparison between the ADAB algorithm and the comparator, helping to inform decision-making regarding the choice of algorithm for the given task or application.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 SUMMARY

Working on a project that focuses on counting automobiles using computer vision (CV) through artificial intelligence and machine learning (AIML). The project specifically utilizes the Adaboosting algorithm to determine the efficiency of the vehicle counting system. Adaboosting is a popular ensemble learning technique that combines multiple weak classifiers to create a strong classifier, which in this case is used for vehicle detection and counting. The system likely involves processing video or image data from cameras installed in traffic areas, using CV techniques to detect and track vehicles, and then applying the Adaboosting algorithm to efficiently count the vehicles passing through the monitored area. The efficiency of the Adaboosting algorithm is a key focus, indicating that the project aims to assess how accurately and reliably the algorithm can count vehicles in real-world scenarios.

6.2 CONCLUSION

According to the reviewed articles, solutions for vehicle counting using computer vision has a great importance in the estimation of the traffic of a city. That is the reason why many research works have been developed in this area. In such works, each author proposes a different method taking into account steps such as the detection, tracking and the counting of vehicles. The most widely used method for detection and tracking is the background subtraction and centroid tracking algorithm. However, in recent years, artificial intelligence has been integrated with cutting-edge tools for object detection such as YOLO and SSD, leaving behind the detection with the use of CV. For automobile wheel counting, Adaboosting and logistic regression algorithms are used and compared to improve wheel recognition accuracy and vehicle monitoring through CV. According to the findings, the Adaboosting algorithm outperforms the Logistic regression algorithm in terms of accuracy for energy efficiency. By introducing Adaboosting algorithm.

REFERENCES

- [1] Ahmed M.M and Abdel-Aty M.A, "The viability of using automatic vehicle identification data for real-time crash prediction," IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 2, pp. 459–468, 2012.
- [2] Chen D, and Wang Q, "Diversified technologies in internet of vehicles under intelligent edge computing," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 4, pp. 2048–2059, 2021.
- [3] Chen R.B, Xiao H, and Hou W, "Research on safety vehicle distance recognition based on stereo vision and DSP//Applied Mechanics and Materials," Trans Tech Publications, vol. 333-335, pp. 805–810, 2013.
- [4] Cheng H.Y, Weng C.C, and Chen Y.Y , "Vehicle detection in aerial surveillance using dynamic Bayesian networks," IEEE Transactions on Image Processing, vol. 21, no. 4, pp. 2152–2159, 2012.
- [5] Cheng X, Yang L, and Shen X, "D2D for intelligent transportation systems: a feasibility study," IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4, pp. 1784–1793, 2015.
- [6] Dimitrakopoulos G and Demestichas P, "Intelligent transportation systems," IEEE Vehicular Technology Magazine, vol. 5, no. 1, pp. 77–84, 2010.
- [7] Guido G, Gallelli V, Rogano D, and A. Vitale, "Evaluating the accuracy of vehicle tracking data obtained from unmanned aerial vehicles," International Journal of Transportation Science and Technology, vol. 5, no. 3, pp. 136–151, 2016.
- [8] Hsieh J.W, Chen L.C, and Chen D.Y, "Symmetrical surf and its applications to vehicle detection and vehicle make and model recognition," IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 1, pp. 6–20, 2014.
- [9] Li X, Yao X C, Murphey Y L, et al. A Real-Time Vehicle Detection and Tracking System in Outdoor Traffic Scenes.[C]// Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on. IEEE, 2004:761-764.

- [10] Lin C. F. Wu, C. J., and Lee C. Y., "Applying a functional network to real-time lane detection and front-vehicle distance measurement," *IEEE Transactions on Systems Man and Cybernetics-Part C-Applications Reviews*, vol. 42, no. 4, pp. 577–589, 2012.
- [11] Lou R.Z , and Singh A.K , "AI empowered communication systems for intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4579–4587, 2021.
- [12] Neto E.C, Gomes S.L, Rebouças Filho P.P, and de Albuquerque V.H.C, "Brazilian vehicle identification using a new embedded plate recognition system," *Measurement*, vol. 70, pp. 36–46, 2015.
- [13] Qureshi K.N and Abdullah A.H, "A survey on intelligent transportation systems," *Middle-East Journal of Scientific Research*, vol. 15, no. 5, pp. 629–642, 2013.
- [14] Sivaraman S and Trivedi M.M, "Looking at vehicles on the road: a survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.
- [15] Tian Y, Dong H, Jia L, and Li S.Y, "A vehicle re-identification algorithm based on multi-sensor correlation," *Journal of Zhejiang University SCIENCE C*, vol. 15, no. 5, pp. 372–382, 2014.
- [16] Tuermer S, Kurz F, Reinartz P, and Stilla U, "Airborne vehicle detection in dense urban areas using HoG features and disparity maps," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 6, no. 6, pp. 2327–2337, 2013.
- [17] Wang W, Song Y, Zhang J, and Deng H, "Automatic parking of vehicles: a review of literatures," *International Journal of Automotive Technology*, vol. 15, no. 6, pp. 967–978, 2014.
- [18] Xiaoling X, "Based on computer vision escape vehicle identification system design and research," *Bulletin of Science and Technology*, vol. 11, p. 37, 2012.
- [19] Zhang B, "Classification and identification of vehicle type and make by cortex-like image descriptor HMAX," *International Journal of Computational Vision and Robotics*, vol. 4, no. 3, pp. 195–211, 2014.

APPENDIX

A.PLAGARISM REPORT OF PAPER

GO02			
ORIGINALITY REPORT			
7%	6%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	www.irjmets.com Internet Source	3%	
2	K. Srilatha, R. Roshini, R. Choshmitha. "Image Reconstruction Method For Ultrafast Ultrasonic Imaging Based On Convolutional Neural Network(CNN)", 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN), 2023 Publication	2%	
3	Sujatha Therese P, Eugene Prince M, Josephin Shermila P, Anna Devi E. "Design of Self Tuning Fuzzy PI Controller for MIMO process using an analytical approach for selecting the scaling factors", 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICT), 2022 Publication	1%	
4	Submitted to Georgia State University Student Paper	1%	

B. PAPER SUBMISSION TO THE CONFERENCE

2024 2nd World Conference on Communication & Computing 2024 : Submission (974) has been edited.

Inbox x

Microsoft CMT <email@msr-cmt.org>

to me

6:03 PM (9 minutes ago)

☆

😊

↩

⋮

Hello,

The following submission has been edited.

Track Name: WCONF2024

Paper ID: 974

Paper Title: AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY

Abstract:
Accurate measurement of traffic in the field of smart transportation is important for effective traffic management strategies. Vehicle identification plays an important role in this, especially vehicle counting and classification. Although there are many methods of counting vehicles, including manual counting and technological solutions such as computer vision and pneumatic tubes, finding a fast and accurate method is still a goal. Traditional methods are often time-consuming and difficult, requiring the search for better options. To solve these problems, this work presents a computer vision method that uses a simple, fast and accurate virtual space detection method. Our method defines the virtual sensing space in the framework using the UA-DETRAC dataset, specifying control that can be done manually or using tools such as the OpenCV rendering Library. model and allows vehicles to be classified into five different groups. For vehicle tracking and counting, we used the SORT algorithm, which tracks vehicles passing through a virtual detection area. This allows us to confidently estimate traffic at any given time by counting the value. Keywords: Smart Transportation, Traffic Management, Vehicle identification, Vehicle Counting, vehicle Classification ,Computer Vision, Pneumatic tubes, Virtual space Detection, data set, open CV library .

Created on: Wed, 03 Apr 2024 05:39:08 GMT

Last Modified: Sun, 14 Apr 2024 12:33:43 GMT

Submissions

Search help articles

Help Center

Select Your Role : Author

WCONF2024

P V V L D S SAI MANIKANTA

Author Console

+ Create new submission

1 - 1 of 1

Show: 25 50 100 All

Clear All Filters

Paper ID	Title	Files	Actions
974	AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY Hide abstract Accurate measurement of traffic in the field of smart transportation is important for effective traffic management strategies. Vehicle identification plays an important role in this, especially vehicle counting and classification. Although there are many methods of counting vehicles, including manual counting and technological solutions such as computer vision and pneumatic tubes, finding a fast and accurate method is still a goal. Traditional methods are often time-consuming and difficult, requiring the search for better options. To solve these problems, this work presents a computer vision method that uses a simple, fast and accurate virtual space detection method. Our method defines the virtual sensing space in the framework using the UA-DETRAC dataset, specifying control that can be done manually or using tools such as the OpenCV rendering Library. model and allows vehicles to be classified into five different groups. For vehicle tracking and counting, we used the SORT algorithm, which tracks vehicles passing through a virtual detection area. This allows us to confidently estimate traffic at any given time by counting the value. Keywords: Smart Transportation, Traffic Management, Vehicle identification, Vehicle Counting, vehicle Classification ,Computer Vision, Pneumatic tubes, Virtual space Detection, data set, open CV library .	Submission files: PAPER IEEE (1) (1).pdf	Submission: Edit Submission Edit Conflicts Delete Submission

2024 IEEE World Conference on Applied Intelligence and Computing : Submission (723) has been edited. [Inbox x](#)



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

6:08 PM (7 minutes ago) ☆ ☹ ↶ ⋮

Hello,

The following submission has been edited.

Track Name: AICconf2024

Paper ID: 723

Paper Title: AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY

Abstract:

Accurate measurement of traffic in the field of smart transportation is important for effective traffic management strategies. Vehicle identification plays an important role in this, especially vehicle counting and classification. Although there are many methods of counting vehicles, including manual counting and technological solutions such as computer vision and pneumatic tubes, finding a fast and accurate method is still a goal. Traditional methods are often time-consuming and difficult, requiring the search for better options. To solve these problems, this work presents a computer vision method that uses a simple, fast and accurate virtual space detection method. Our method defines the virtual sensing space in the framework using the UA-DETRAC dataset, specifying control that can be done manually or using tools such as the OpenCV rendering Library. model and allows vehicles to be classified into five different groups. For vehicle tracking and counting, we used the SORT algorithm, which tracks vehicles passing through a virtual detection area. This allows us to confidently estimate traffic at any given time by counting the value. Keywords: Smart Transportation, Traffic Management, Vehicle identification, Vehicle Counting, vehicle Classification ,Computer Vision, Pneumatic tubes, Virtual space Detection, data set, open CV library .

Created on: Sat, 13 Apr 2024 11:35:25 GMT

Last Modified: Sun, 14 Apr 2024 12:38:24 GMT

Submissions

[Search help articles](#)

[Help Center](#) ▾

Select Your Role : [Author](#) ▾

[AICconf2024](#) ▾

[P V V L D S SAI MANIKANTA](#) ▾

Author Console

[+ Create new submission](#)

1 - 1 of 1 [◀](#) [◀](#) [1](#) [▶](#) [▶](#) [Show: 25](#) [50](#) [100](#) [All](#) [Clear All Filters](#)

Paper ID	Title	Files	Actions
Clear	Clear		
723	AUTOMOBILE VEHICLE COUNT BASED ON CV THROUGH AIML USING ADABOOSTING ALGORITHM FOR DETERMINING THE EFFICIENCY Hide abstract Accurate measurement of traffic in the field of smart transportation is important for effective traffic management strategies. Vehicle identification plays an important role in this, especially vehicle counting and classification. Although there are many methods of counting vehicles, including manual counting and technological solutions such as computer vision and pneumatic tubes, finding a fast and accurate method is still a goal. Traditional methods are often time-consuming and difficult, requiring the search for better options. To solve these problems, this work presents a computer vision method that uses a simple, fast and accurate virtual space detection method. Our method defines the virtual sensing space in the framework using the UA-DETRAC dataset, specifying control that can be done manually or using tools such as the OpenCV rendering Library. model and allows vehicles to be classified into five different groups. For vehicle tracking and counting, we used the SORT algorithm, which tracks vehicles passing through a virtual detection area. This allows us to confidently estimate traffic at any given time by counting the value. Keywords: Smart Transportation, Traffic Management, Vehicle identification, Vehicle Counting, vehicle Classification ,Computer Vision, Pneumatic tubes, Virtual space Detection, data set, open CV library .	Submission files: PAPER IEEE (1) (1).pdf	Submission: Edit Submission Edit Conflicts X Delete Submission