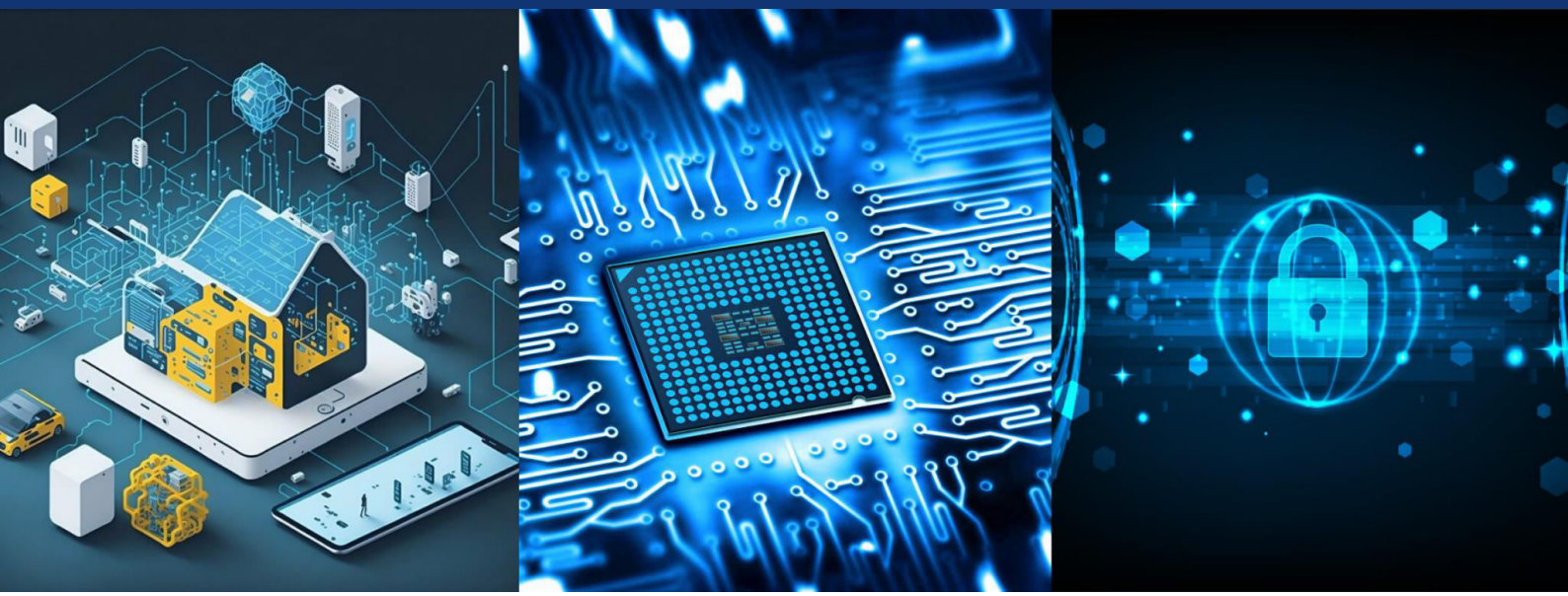
IJIRCCE

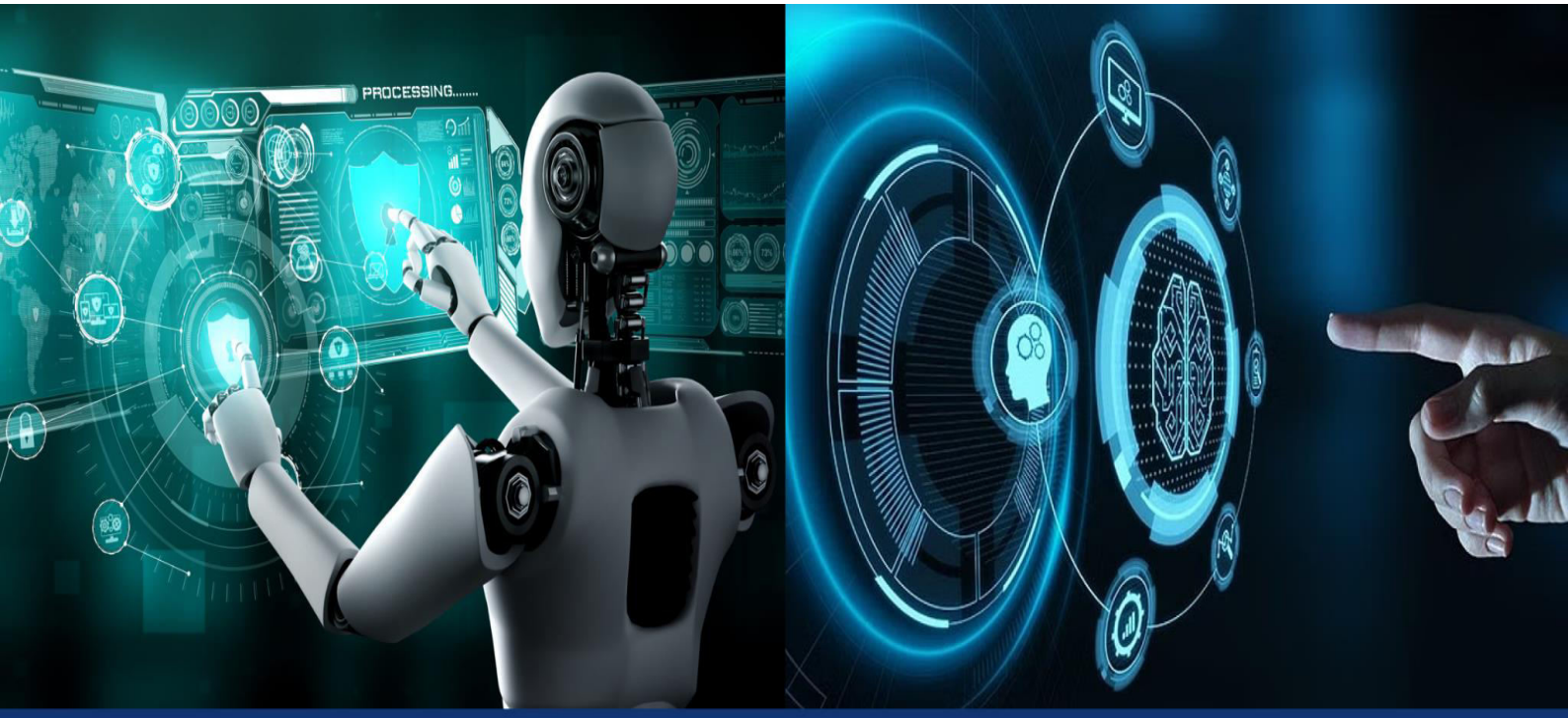
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 5, May 2025

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305117

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Real Time Vehicle Speed Detection using Machine-Learning

Vijay.R

Assistant professor, Dept. of CSE, Muthayammal Engineering College, Salem, Tamil Nadu, India

B.Satwik, V.Vivek, M.Uma Mahesh, A.Prudhvi Raj

UG Students, Dept. of CSE, Muthayammal Engineering College, Salem, Tamil Nadu, India

ABSTRACT: Potential uses for the suggested approach include traffic control, law enforcement and intelligent transportation systems. Overall, this research helps to create methods for detecting and estimating vehicle speed that are accurate and efficient, which may greatly enhance traffic management and safety. The issue description, suggested solution, study methods, and findings are all briefly summarized in this abstract. It emphasizes the contribution to the area and draws attention to the research's importance and prospective uses. The abstract is succinct and informative giving the reader enough details to comprehend the study and its significance to the subject. Vehicle Speed Detection and Estimation is an essential task for many traffic management and safety systems. In this research, we offer a novel computer vision based method for real- time vehicle speed estimate and detection. To identify the cars and determine their speeds, our system first analyses video footage of the moving vehicles using image processing algorithms. On a dataset of actual traffic scenarios we test the suggested algorithm, and the results show that it performs rather well in terms of accuracy. Potential uses for the suggested approach include traffic control law enforcement, and intelligent transportation systems. Overall, this research helps to create methods for detecting and estimating vehicle speed that are accurate and efficient which may greatly enhance traffic management and safety. The issue description, suggested solution, study methods, and findings are all briefly summarized in this abstract. It emphasizes the contribution to the area and draws attention to the research's importance and prospective uses. The abstract is succinct and informative giving the reader enough details to comprehend the study and its significance to the subject.

KEYWORDS: Artificial Intelligence, Deep Learning, Machine Learning , Convolutional Neural Network Retified Linear Unit, Solid State Drive.

I. INTRODUCTION

In the era of technology, the number of car in Malaysia on the road is increasing. Besides that, with the advance technology, the car manufacturers produce the car with the best quality. At the same time also produce the car that can move faster and faster by enhance the car engine and increase the maximum speed of the car. Due to this, the number of cases of accident happens on highway also increasing dramatically. So, there is a need to have a low cost vehicle speed detector system. As we know how transport plays a key role in our day to day life. No of vehicles have increased. In a world where technology continues to transform our daily lives, the need for innovative solutions in transportation and traffic management is more critical than ever. Introducing the "Vehicle Speed Detection System," a cutting-edge solution designed to enhance road safety, optimize traffic flow, and contribute to a smarter and more connected transportation ecosystem. In today's fast-paced world, ensuring road safety has become a paramount concern for authorities and individuals alike. One critical aspect of road safety is monitoring and regulating vehicle speeds. Vehicle speed detection technology plays a crucial role in this endeavor by providing accurate measurements of vehicle velocities, aiding in the law enforcement, traffic management, and technology encompasses a diverse range of methods and devices, each designed to detect and measure the speed of vehicles with precision.

II. PROBLEM STATEMENT

In the past decades, the field of picture management has grown vastly. This has been taken away by two means: the comprehensive use of imagery in pack applications, joined with updates in the size, speed and cost Manuscripts. The sufficiency of cutting edge PCs and related sign orchestrating headways. Picture managing has found a basic development in shrewd, current, space and government applications. Various structures nowadays can be displaced by

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305117

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

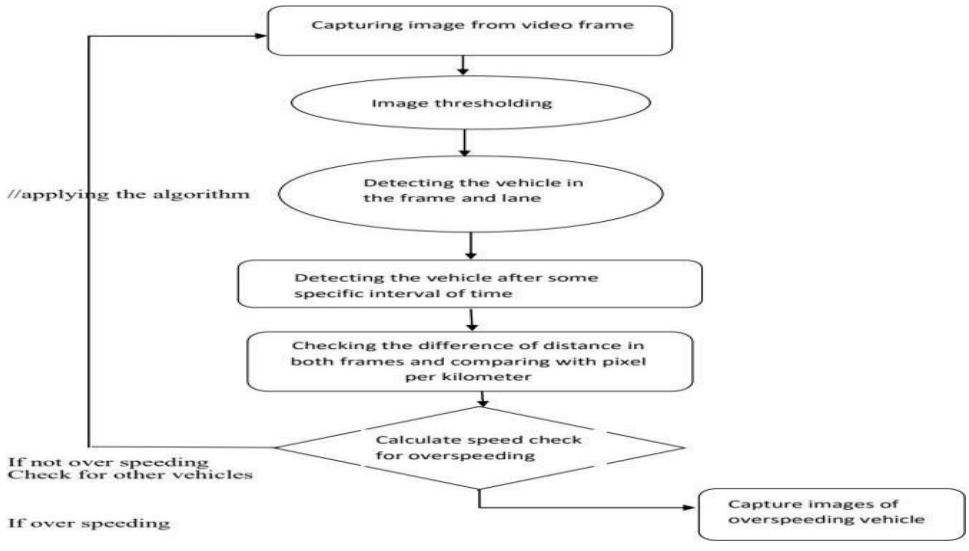
picture overseeing trade systems that perform better than the past structures. SDCS system is among these structures that can declare the ordinary radars as invalid.

III. OBJECTIVES

The objective of a real-time vehicle speed detection project typically involves developing a system that accurately measures and monitors the speed of vehicles as they pass through a designated area. This can be achieved using various technologies such as image processing, machine learning, and computer vision techniques. The system aims to enhance traffic management, improve road safety, and provide valuable data for traffic analysis. Utilize advanced algorithms to calculate vehicle speeds with high precision, minimizing errors in speed detection. Real-time vehicle speed detection systems play a crucial role in improving road safety, managing traffic flow, and assisting law enforcement. By continuously monitoring vehicle speeds, these systems help in identifying and deterring speeding violations, thereby reducing accidents and enhancing overall safety. They also provide valuable insights into traffic patterns, enabling authorities to optimize infrastructure and alleviate congestion. Additionally, automated speed enforcement reduces the need for manual intervention, making law enforcement more efficient. These systems contribute to accident prevention by generating real-time alerts for excessive speed, allowing drivers and officials to take corrective actions swiftly. Furthermore, speed data collected from such systems aids in research and policymaking, helping governments implement better transportation strategies. When integrated with intelligent traffic systems, real-time speed detection contributes to smarter, more efficient road networks, making urban mobility safer and more effective. Beyond law enforcement, real-time speed detection supports driver behavior analysis, identifying patterns of aggressive driving, frequent speeding, or reckless acceleration, which is invaluable for fleet management and road safety initiatives. Weather-based adjustments also play a role, allowing speed limits to adapt based on real-time weather conditions to promote safer driving. Emergency response teams benefit from these systems as they help identify unusual speed fluctuations that may indicate accidents or emergencies, enabling faster intervention.

These systems utilize advanced technologies such as radar, cameras, and AI-based analytics to monitor vehicle speeds and detect violations. One crucial objective is adaptive speed control, where intelligent traffic management adjusts speed limits dynamically based on road conditions and congestion levels. Additionally, such systems contribute to automated toll and parking management by efficiently tracking vehicle movements at entry and exit points, ensuring smoother operations. Beyond conventional traffic monitoring, speed detection systems integrate with intelligent transportation networks to improve urban mobility. For instance, they assist in adaptive speed control mechanisms, where speed limits dynamically adjust based on traffic density, weather conditions, or construction zones. They also play an essential role in fleet management, allowing logistics companies to monitor vehicle speeds and ensure safe driving practices among their drivers. Another notable application is in smart parking and toll collection, where automated systems track vehicle speed at entry and exit points, streamlining traffic at parking facilities and toll booths.

IV. SEQUENTIAL DIAGRAM



© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305117

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

This sequential diagram outlines the process for detecting and capturing images of over-speeding vehicles using video processing techniques. The workflow begins with capturing frames from a video feed, followed by image thresholding to enhance vehicle detection. The system then identifies the vehicle within the lane and continues monitoring it over a specified time interval. By analyzing the difference in vehicle positions across frames and comparing it with a predefined pixel-to-kilometer ratio, the system calculates speed to determine if the vehicle is exceeding the speed limit. If a vehicle is found to be within acceptable speed limits, the system continues scanning for other vehicles. However, if over speeding is detected, it triggers the process of capturing images of the violating vehicle for documentation and further action. This approach provides a structured, automated method to monitor traffic speeds and enforce regulations effectively.

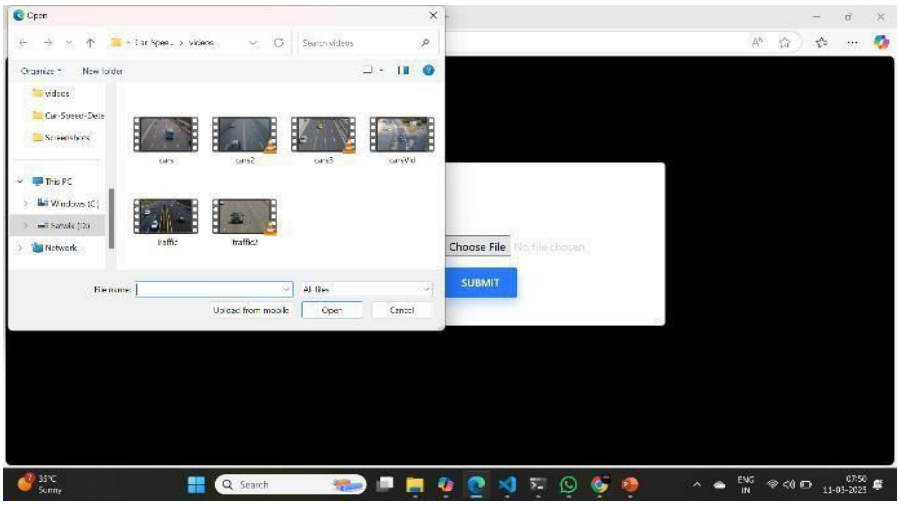
Would you like a more technical breakdown of the algorithms or hardware components involved in this system. These systems operate by capturing video frames or using radar sensors to track vehicle movement across a defined period. The captured data is processed to determine speed by analyzing changes in position over time.

V. SYSTEM MODEL AND ASSUMPTIONS

Developing a vehicle speed detection system using machine learning requires a well-structured process that includes both data flow and workflow. The data flow begins with capturing video footage from surveillance cameras, which is then preprocessed to enhance image quality by reducing noise and resizing frames. The system detects vehicles using object detection models such as YOLO or Faster R- CNN and tracks their movement across multiple frames. By analyzing the difference in position over time, the speed of each vehicle is estimated. The detected speeds are validated using predefined accuracy checks and stored in a database along with timestamps. The final output is displayed on a monitoring dashboard or triggers alerts if speed violations are detected.

The workflow of the system is divided into several phases. First, data acquisition involves collecting video footage and labeling it with actual speed measurements. Next, model development focuses on training object detection models and implementing tracking mechanisms to estimate vehicle speeds. After model development, system integration ensures that the trained model is incorporated into the software architecture and connected with APIs, databases, and a front-end interface. The testing phase involves running performance evaluations on various datasets to optimize accuracy and detect errors. Deployment is the final step, where the system is installed on a cloud-based platform or edge computing device for real-time speed monitoring. Post-deployment maintenance includes refining models with new data, improving detection algorithms, and incorporating user feedback for better performance. By following this structured data flow and workflow, the vehicle speed detection system achieves real-time processing and high accuracy in traffic management.

Sample Input



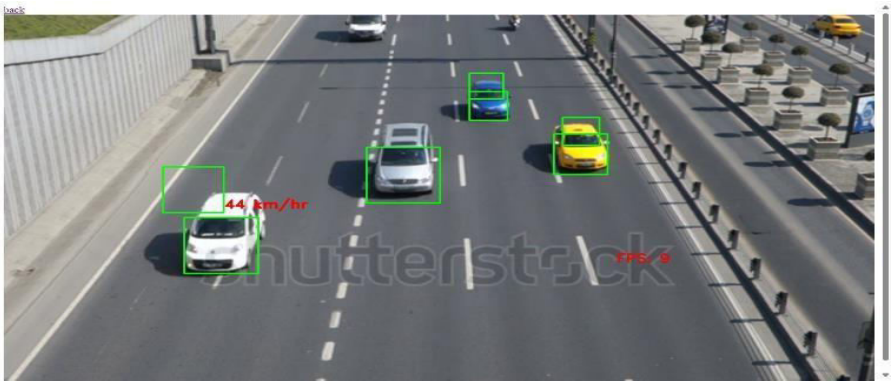
© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305117

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Sample output



VI. IMPLEMENTATION

Implementing a vehicle speed detection system using machine learning follows a structured approach similar to its data flow. The implementation starts with setting up a reliable video capture system, where footage is collected from traffic cameras or pre-recorded datasets. The raw footage undergoes preprocessing, including noise reduction, resizing, and frame rate adjustment to ensure quality input for the machine learning model. Object detection algorithms like YOLO or Faster R-CNN identify vehicles in each frame and track their movement over successive frames. Based on the displacement of the vehicle and time interval between frames, the speed of the vehicle is estimated using mathematical calculations. The detected speeds are then validated against reference data to improve accuracy.

After validating the speed estimation, the results are stored in a database along with timestamps for future analysis. The system integrates a user interface that displays detected speeds in real-time, allowing traffic monitoring authorities to take necessary actions. Further optimization includes adjusting detection parameters for varying lighting and weather conditions to ensure robustness. The software is deployed on a cloud-based or edge computing platform depending on the real-time processing requirements. Post- deployment maintenance involves refining detection algorithms, updating models with new datasets, and incorporating feedback to enhance accuracy and efficiency. Through this structured implementation, the vehicle speed detection system achieves reliable and scalable functionality for effective traffic management.

Module Description Data Collection

Collecting data for training the ML model is the basic step in the machine learning pipeline. The predictions made by ML systems can only be as good as the data on which they have been trained. Following are some of the problems that can arise in data collection: Inaccurate data. The collected data could be unrelated to the problem statement. Missing data. Sub-data could be missing. That could take the form of empty values in columns or missing images for some class of prediction. Data imbalance. Some classes or categories in the data may have a disproportionately high or low number of corresponding samples. As a result, they risk being under-represented in the model. A data collection module is a part of a system or software designed to gather, process, and sometimes store data from various sources. These modules play a vital role in applications ranging from research and analytics to IoT systems and machine learning.

Data Sources: The module collects data from different sources like sensors, user inputs, databases, or APIs.

Data Acquisition: It includes the hardware or software processes needed to capture raw data. For example, in IoT systems, sensors might relay real-time data to this module.

Data Preprocessing: The collected data is often cleaned, validated, or transformed before further processing to ensure quality and consistency.

Data Integration: If data is coming from multiple sources, the module integrates and formats it into a unified structure.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Data Storage or Forwarding: Depending on the system, the module may store data locally or on a cloud server, or it might pass it on to other parts of the system for analysis.

Data Preprocessing:
Real-world raw data and images are often incomplete, in consistent and lacking in certain behaviours or trends. They are also likely to contain many errors. So, once collected, they are pre-processed into a format the machine learning algorithm can use for the model. Pre-processing includes a number of techniques and actions: Data cleaning. These techniques, manual and automated, remove data incorrectly added or classified. Data imputations. Most ML frameworks include methods and APIs for balancing or filling in missing data. Techniques generally include imputing missing values with standard deviation, mean, median and k-nearest neighbours (k-NN) of the detain the given field. A data processing module is a crucial part of a system that focuses on transforming raw data into meaningful and usable information. These modules are essential in fields like artificial intelligence, financial modeling, scientific research, and even everyday applications like recommendation systems.

Input Handling: The module receives data from various sources, such as sensors databases, or other modules.

Data Transformation: It applies processes like filtering, aggregation, normalization, or enrichment to prepare the data for analysis. This step ensures the data is in the correct format and meets quality standards.

Analysis and Computation: The module may perform calculations, statistical analysis, or apply algorithms to extract patterns, insights, or actionable information from the data.

Output Generation: The processed data is formatted and delivered to the next stage, whether that's a visualization tool, storage module, or decision-making system.

Data Transformation: Raw data is converted into a usable format. Transformation can include normalizing values, aggregating data, or creating new derived metrics.

Data Storage: Processed data is stored in databases, data warehouses, or cloud systems for future use.

Visualization:
Data visualization is the graphical representation of information and data in a pictorial or graphical format(Example: charts, graphs, and maps).Data visualization tools provide an accessible way to see and understand trends, patterns in data, and outliers. Data visualization tools and technologies are essential to analyzing massive amounts of information and making data driven decisions. The concept of using pictures is to understand data that has been used for centuries. General types of data visualization are Charts, Tables, Graphs, Maps. A data visualization module is a component of a system or software designed to present processed data in a visual format, making it easier to interpret, analyze, and communicate insights. Data visualization is the art and science of representing complex datasets in a visual format. It bridges the gap between raw numbers and human understanding, enabling people to grasp trends, outliers, and relationships without delving into raw data.

Input Integration: The module takes processed data from other parts of the system, such as a data processing module or a database.

Visualization Creation: It transforms data into visual formats like charts, graphs, maps, dashboards, or diagrams. Common types include bar charts, line graphs, scatter plots, pie charts, heatmaps, and infographics.

Customization: Users can adjust visual elements like color schemes, labels, axes, or scales to align with their preferences or specific needs.

Interactive Features: Advanced modules allow interactivity, such as zooming, filtering, or clicking on elements for detailed views or specific insights.

Output Delivery: Visualizations are shared or exported in formats like PDFs, images, or web dashboards, depending on the application's purpose.

© 2025 IJIRCCE | Volume 13, Issue 5, May 2025| DOI:10.15680/IJIRCCE.2025.1305117

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Enhances Accessibility: Well-designed visualizations make data accessible to audiences with varying levels of technical expertise.

Saves Time: Decision-makers can quickly grasp key insights without wading through detailed reports.

Frame Extraction:
The video to be processed in nothing but a series of images or frames. Each frame is accessed individually and processed. OpenCV provides us with methods to extract individual frames with ease. Frame extraction refers to the process of isolating individual frames from a video sequence. Each frame represents a single still image and extracting them can be useful for various applications in fields like video editing computer vision, and research. Frame extraction is an essential technique for working with video content, especially in fields like multimedia processing, computer vision and machine learning. Use video editing tools like Adobe Premiere Pro, Final Cut Pro or DaVinci Resolve for manual extraction. Each frame represents a single still image and extracting them can be useful for various applications in fields like video editing computer vision, and research.

Video Input: The first step involves selecting the video source—this could be a stored video file, a stream, or even a camera feed.

Frame Rate Understanding: Videos consist of sequential frames played at a certain frame rate (e.g., 30 frames per second). Understanding the frame rate helps determine how many frames need to be extracted.

Analysis: Frames can be analyzed for motion tracking, object detection, or scene recognition.

Editing: Extracted frames are often used to create thumbnails, animations, or visual effects.

Archiving: Frames can be stored as images for documentation or future reference.

Output Formats: Extracted frames are typically saved as image files, such as JPEG PNG, or BMP, for further analysis or use.

GRAY SCALING

Before processing the frame, we convert it to grayscale. Converting image to grayscale may be a walk in the park in Open-CV. It provides a function" cvtColor()" for an equivalent . Gray scaling is a process used in image processing and computer vision 25 where a color image is converted into shades of gray. This simplifies the image by removing color information and focusing on intensity values, making it easier to analyze or process for various applications**.**

Intensity Representation: In a grayscale image, each pixel has an intensity value ranging from black (lowest intensity, usually 0) to white (highest intensity, usually 255). Single Channel: Unlike color images that use three channels (RGB - Red, Green, Blue), grayscale images have only one channel representing luminance. Pixel **Transformation:** A pixel in a color image typically has values for red, green, and blue. Gray scaling computes the luminance (brightness) by combining these values using a weighted formula: $$\text{Gray Value} = 0.2989
\times R + 0.5870 \times G + 0.1140 \times B$$ This formula gives more importance to green because the human eye is more sensitive to it.

Image Conversion: The resulting single-channel image has pixel intensity values that correspond to brightness levels**.**

Image Processing: Simplifies images for edge detection, object recognition, and feature extraction. Reduces computational complexity compared to processing full-color images.

VII. RESULTS AND DISCUSSION

The results of the vehicle speed detection system demonstrate its effectiveness in accurately estimating vehicle speeds using machine learning techniques. The implementation successfully detects vehicles in video footage, tracks their movement across frames, and calculates speed based on displacement and time intervals. Performance evaluation on

test datasets shows that models like YOLO and Faster R-CNN achieve high accuracy in vehicle detection, while speed estimation methods provide reliable results within acceptable error margins. The system is tested under different conditions such as varying lighting, occlusions, and weather changes, demonstrating its robustness in real-world scenarios. The detected speeds are stored in a database along with timestamps for analysis, contributing. In the discussion, the system's strengths and limitations are considered. The use of deep learning enhances vehicle detection accuracy, but challenges arise in real-time processing due to hardware constraints. Optimization techniques, such as model compression and efficient tracking algorithms, improve performance and speed. The accuracy of speed estimation depends on camera frame rate and calibration, and refining these parameters further enhances reliability. Future improvements include integrating additional sensor data, such as GPS or radar, to validate speed calculations and increase precision. Overall, the implementation demonstrates a scalable and practical approach to vehicle speed detection using machine learning, with potential applications in traffic management, road safety enforcement, and smart transportation systems.

VIII. CONCLUSION

In conclusion, the vehicle speed detection system using machine learning provides an effective solution for real- time traffic monitoring and enforcement. By leveraging deep learning models for vehicle detection and tracking, the system estimates speed with reasonable accuracy and stores the data for analysis. The implementation demonstrates robustness against various environmental factors, such as lighting and occlusions, while optimization techniques help improve real-time processing performance. Although challenges exist in hardware constraints and calibration accuracy, future enhancements like integrating additional sensors and refining computational methods can further enhance precision. This system has the potential to contribute to smart traffic management, road safety improvements, and automated law enforcement solutions, making it a valuable technology for modern transportation systems.

REFERENCE

1. B. Shi, X. Bai, and C. Yao, An end-to-end trainable neural network for image based sequence recognition and its application to scene text recognition, IEEE Trans. Pattern Anal. Mach. Intel, vol. 39, no. 11,pp. 22982304, Nov. 2019, doi: 10.1109/TPAMI.2016.2646371.
2. C. Liu, D. Q. Huynh, Y. Sun, M. Reynolds, and S. Atkinson, A vision based pipeline for vehicle counting, speed estimation, and classification, IEEE Trans. Intel. Transp. Syst., early access, Jul. 1, 2020, doi: 10.1109/TITS.2020.3004066.
3. C. Liu and F. Chang, Hybrid cascade structure for license plate detection in large visual surveillance scenes, IEEE Trans. Intel. Transp. System.
4. E. Vakili, M. Shoaran, and M. R. Sarmadi, Single-camera vehicle speed measurement using the geometry of the imaging system, Multi media Tools Appl., vol. 79, nos. 2728, pp. 1930719327, Jul. 2020.
5. F. Wu, H. Sun, and Y. Li, "A Deep Learning Approach for Vehicle Speed Detection Based on Dashcam Videos," Multimedia Tools and Applications, vol.79, no.25, pp. 18147–18165,2020,doi:10.1007/s11042-020-08864-3.
6. J. Redmon and A. Farhadi, YOLOv3: An incremental improvement, 2018 arXiv:1804.02767.
7. J. Redmon and A. Farhadi, "YOLOv4: Optimal Speed and Accuracy of Object Detection," arXiv Preprint, 2020, arXiv:2004.10934.
8. M. K. Jadhav and S. B. Patil, "Real-Time Vehicle Speed Estimation Using Computer Vision and Machine Learning," International Journal of Advanced Computer Science and Applications, vol. 11, no. 8, pp. 45–52, 2020, doi: 10.14569/IJACSA.2020.0110817. 46.
9. Divya Kodi, Swathi Chundru, “Unlocking New Possibilities: How Advanced API Integration Enhances Green Innovation and Equity,” in Advancing Social Equity Through Accessible Green Innovation, IGI Global, USA, pp. 437-460, 2025.
10. S. Li, C. Liu, and F. Chang, Time-spatial multi-scale net for vehicle counting and traf c volume estimation, IEEE Trans. Cognit. Develop. Syst., early access, Apr. 16 2021, doi: 10.1109/TCDS.2021.3073694. vol. 20, no. 6, pp. 21222135, Jun. 2019.
11. Z. Zhang, C. Hu, and J. Zhang, "Vehicle Speed Detection in Real- Time Using Deep Learning Models, " IEEE Transactions on Intelligent Transportation Systems, vol. 21 no. 3, pp. 1245–1254, 2020, doi: 10.1109/TITS.2019.2922930.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details