Revolutionizing Road Safety: Raspberry Pi based driver assistance system for Indian Roads using LIDAR and Camera Sensors

Kaustubha Sahu
Department of Computer Engg.
VES Institute of Technology
Mumbai,India-400074
2020.kaustubha.sahu@ves.ac.in

Pranav Ubarhande
Department of Computer Engg.
VES Institute of Technology
Mumbai,India-400074
2020.pranav.ubarhande@ves.ac.in

Aaryan Manawat
Department of Computer Engg.
VES Institute of Technology
Mumbai,India-400074
2020.aaryan.manawat@ves.ac.in

Abhiman Singh
Department of Computer Engg.
VES Institute of Technology
Mumbai, India-400074
2020.abhiman.rajput@ves.ac.in

Sanjay Mirchandani
Department of Computer Engg.
VES Institute of Technology
Mumbai, India-400074
sanjay.mirchandani@ves.ac.in

***Abstract*—A driver assistance system is implemented on Raspberry Pi which processes real-time data from the LIDAR for distance measurement and camera data for object detection and lane detection, enabling smart surround-view and collision avoidance to enhance road safety. The object detection system is based on MobileNetSSD v2 and achieves a mean average precision (mAP) of 22.1%. The distance calculation is performed using the TF-LUNA Micro LIDAR sensor, which operates within a range of 8 meters and has an error margin of ±6 cm for distances less than 3 meters, and an error margin of ±2% for distances between 3 meters and 8 meters. Thus, the system is designed to handle the unique challenges of Indian roads, to prevent accidents, and ultimately save lives.**

***Index Terms*—Road Safety, Driver Assistance System, LIDAR, Camera, Object Detection, Lane Detection.**

I. Introduction

The need for strong and precise object detection systems has increased across various industries, such as autonomous vehicles, robotics, surveillance, and industrial automation, in today's fast-changing technological landscape. However, it is still a difficult challenge to achieve accurate object detection in complex and dynamic environments. Traditional single-sensor solutions often fail to deal with the complexities of real-world scenarios, such as bad weather conditions, low-light environments, and the presence of occlusions. A promising solution is the integration of multiple sensors, which enhances the capabilities of object detection systems[11][13][15]. The concept of multi-sensor fusion, amalgamating data from diverse sensors, offers a complete approach to perceiving and recognising in complex environments. By combining the strengths of different sensor modalities, such as cameras, LIDAR, sensors, etc., multi-sensor fusion systems can significantly improve the accuracy, reliability, and robustness of object detection. A LIDAR excels at distance estimation, functioning as a Time-Of-Flight sensor. Meanwhile, a camera is adapted to object classification and scene comprehension. Cameras are commonly used to generate bounding boxes, identify lane line positions, recognize traffic light colours, and detect traffic signs, among other applications. The demand for accurate and dependable object detection in complex and dynamic environments has increased in an era of rapid technological progress and the spread of automation and autonomous systems.

The study of the current system is summarized in the related work section II. The methodology section, section III provides a succinct overview of the main research methods and data gathering techniques employed in the project. The implementation of the system is detailed in the Implementation in section IV, which includes an overview of the hardware and software, the integration of sensors and the operation of the entire system. This is followed by a presentation of the results obtained from the system in section V.

II. Related Work

Kyongsu Yi, Heungseok Chae, and Hojoon Lee [1] introduce a novel sensor fusion system for autonomous vehicles, merging Radar's large Field of View (FOV) with LIDAR's great distance accuracy, utilizing a multiple hypothesis tracking (MHT) technique. Xiaoting Yan, Deguang Li, and Bo Wang [2] examine classification systems for Sensor Fusion, emphasizing its critical role in autonomous vehicles and supporting dependable autonomous driving. An IIT B alumnus and KLA-Tencor CSE propose [3] the significance of Sensor Fusion in overcoming individual sensor limitations for autonomous cars, advocating for reasonably priced driver assistance systems. [4] discusses object identification and location regression for autonomous driving, proposing a LIDAR-camera fusion approach using a Siamese network.Ponn, T., Kröger, T. and Diermeyer in [5] conducts a thorough analysis of object detectors in automated driving using meta-information. Machine learning-based detectors' behavior is modeled using random forests, showing high accuracy in representing detection performance. W. Deng and C. Wang [6] propose an instance segmentation-based approach to lane identification, offering significant advancements in the creation of dependable lane-keeping systems. [7] presents an improved method for real-time object tracking and detection in autonomous transportation systems, advocating for multi-sensor fusion and collaboration between tracking and detection algorithms to enhance applicability and reliability in dynamic situations.

III. Methodology

This brief methodology section summarizes the key research methods and data collection techniques used in the system.

1) **Sensor Fusion:** Sensor fusion techniques combine data from multiple sensors, such as LIDAR and cameras, to improve object detection and depth estimation.
2) **Object Detection Algorithms:** Utilizing MobilenetSSDv2 for real-time detection and classification of objects in camera images and OpenCV for lane detection.
3) **Depth Estimation:** Techniques for estimating object distance from the sensor (TF-Luna micro LIDAR), including stereo vision, structure from motion (SfM), and point cloud processing, have been used to provide depth information.
4) **Machine Learning:** Utilizing machine learning models, such as neural networks and deep learning, for training object detection and classification algorithms.
5) **Calibration and Alignment:** Methods for sensor calibration and alignment have been crucial for ensuring accurate sensor fusion and depth estimation.
6) **Real-time Processing:** Systems have focused on optimizing real-time data processing to provide timely alerts and assistance to drivers.

IV. Implementation

This study focuses on developing an innovative object detection system through the fusion of 2D LIDAR and camera sensors. By combining cost-effective 2D LIDAR technology with cameras, the system achieves real-time object detection, classification, and depth estimation. The integration of data from the 2D LIDAR sensor and the Raspberry Pi camera aims to create a driver assistance system that enhances road safety and driver awareness. Implementation will involve sensor fusion algorithms, object detection models, and user-friendly interfaces to provide timely visual alerts to the driver[14].

A. Overview of Hardware Components

1) **Sensor Integration**
   - 2D LIDAR Sensor: The TF-Luna as shown in Figure 1, is a Light Detection And Ranging (LIDAR) module. It can accurately measure distances up to 8m.

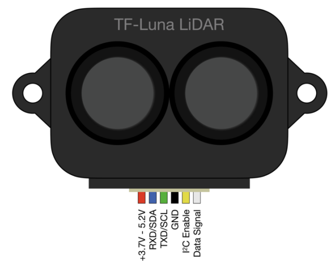


Fig. 1. TF-Luna Mini LIDAR

   - Raspberry Pi: A Raspberry Pi 4 (8GB RAM) single-board computer for data processing and integration.
   - Raspberry Pi Camera: As shown in Figure 2, the camera serves for object classification and scene understanding. This Pi camera supports 1080p resolution with a frame rate of 30 fps with a 5MP sensor.

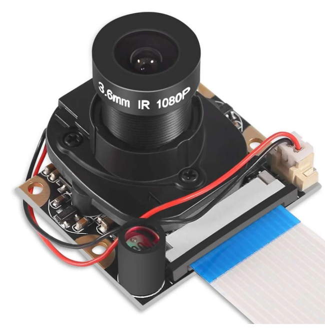


Fig. 2. Raspberry PI Camera Module 5MP 1080p

2) **Power Supply:** A 5V 3A 10,000 mAh battery for powering the system.
3) **Connectivity:**
   - Type C-USB Cable: A Type C-USB cable to connect the power supply to the Raspberry Pi.
   - FFC cable to connect the camera to Raspberry Pi.
   - JST Jumper 2 cable to connect the Lidar sensor to Raspberry Pi.

B. Overview of Software Components

Operating System: The system runs on Raspberry Pi OS (formerly Raspbian).

Sensor Interface

- LIDAR Interface: The software facilitates communication and data reception from the 2D LIDAR sensor via a USB connection. To connect with the Raspberry Pi, the TF-Luna utilizes the Universal Asynchronous Receiver-Transmitter (UART) serial port, specifically, through GPIO pins. Before utilizing the LIDAR sensor with the Raspberry Pi, the mini UART port is enabled in the boot configuration file.

- Camera Interface: Software for capturing and processing images from the Raspberry Pi Camera.

Preprocessing: In this section, essential Preprocessing steps for Camera and LIDAR are outlined:

1) Camera Preprocessing:

- Grayscale Conversion: As shown in Figure 3, the RGB image is converted to grayscale. This step reduces dimensionality while preserving essential edge and gradient information.

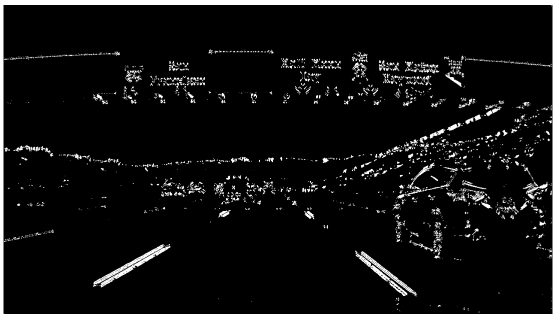


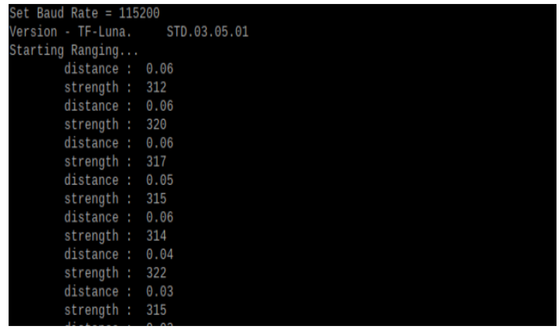
Fig. 3. Grayscaling and Contour Detection

- Undistortion: Camera images undergo undistortion using calibration parameters derived from chessboard images. This correction addresses lens distortions, ensuring an accurate representation of objects.
- Colour Space Conversion: The RGB colour space is converted to the HLS colour space. This conversion facilitates the extraction of lane line information, which is crucial for object detection.
- Gradient Thresholding: Sobel gradient thresholding is applied to the lightness channel (L-channel). This process enhances edge detection, improving the contrast between lane lines and the road surface.
- Color Thresholding: Colour thresholding is performed on the saturation channel (S-channel). It further isolates lane lines from the background, addressing challenges posed by varying lighting conditions and shadows.
- Perspective Transformation: Achieved through perspective warping, this step provides a bird's-eye view of the road. Lane lines are rendered as straight lines parallel to the image's vertical axis, simplifying lane detection.

These preprocessing steps collectively refine the quality and representation of camera images, establishing a solid foundation for accurate and reliable object detection in computer vision applications.

2) LIDAR preprocessing: LIDAR preprocessing for TF-Luna data involves parsing raw sensor output and extracting relevant metrics. The TF-Luna sensor outputs the following data:

- Distance: Measured in centimetres by default.
- Signal strength (Amp): The reliability of the distance value is affected when the received signal is either overexposed (Amp = 0xFFFF) or too low Amp < 100.
- Chip Temperature (Temp): The temperature in degrees Celsius is calculated as Temp/8 - 256°C.



```
Set Baud Rate = 115200
Version - TF-Luna.     STD.03.05.01
Starting Ranging...
        distance :  0.06
        strength :  312
        distance :  0.06
        strength :  320
        distance :  0.06
        strength :  317
        distance :  0.05
        strength :  315
        distance :  0.06
        strength :  314
        distance :  0.04
        strength :  322
        distance :  0.03
        strength :  315
```

Fig. 4. Data obtained from TF-Luna LIDAR sensor.

As shown in Figure 4, the sensor prints out a test range and signal amplitude at the module's default baud rate (115200) and sample rate (100Hz). The processed data is then suitable for applications such as obstacle detection, mapping, and object recognition.

Lane detection: The lane detection pipeline initiates with the conversion of the input RGB image to grayscale. This step effectively reduces dimensionality while preserving crucial edge and gradient information. Following this, contour detection is applied to identify potential road surface obstructions or irregularities. Next, a perspective transformation grants us a top-down view of the road scene. This transformation significantly aids in accurate curved lane line detection. Simultaneously, a histogram of pixel intensities along the transformed image's vertical axis is computed. These intensity peaks correspond to lane lines.

To iteratively search for lane pixels, the sliding window algorithm is employed within vertical windows of the transformed image. Additionally, polynomial curve fitting models the detected lane pixels, providing a reliable estimation of lane boundaries. Finally, the detected lane lines are visualized on the original undistorted image. This visualization offers a clear depiction of the vehicle's trajectory relative to the lane boundaries. The integrated approach combines various preprocessing and detection techniques, contributing to advancements in autonomous driving technology. Figures 5, 6 represent the results of various steps used in lane detection.

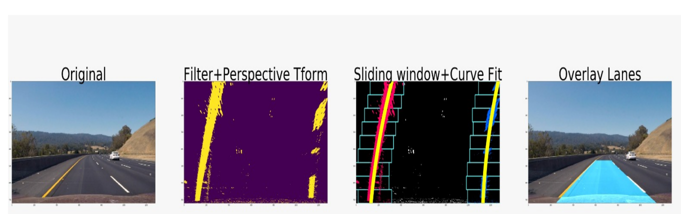


Fig. 5. Warping pipeline for curve detection

Object Detection: Object Detection Algorithm: The system will utilize object detection algorithms, such as SSD

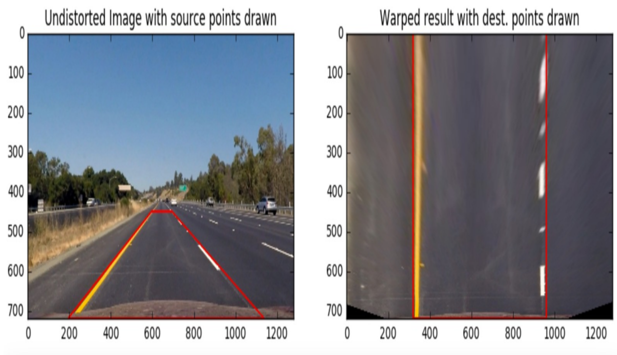


Fig. 6. Warping Results

MobileNet, to identify and classify objects within camera images. SSD (Single Shot MultiBox Detector) employs a convolutional network that generates bounding boxes and class scores, refined through non-maximum suppression to produce final detections. This network is constructed upon a base architecture designed for high-quality image classification, augmented with layers dedicated to detection. These additional layers, progressively decreasing in size, facilitate multi-scale detection. Each layer generates detection predictions using convolutional filters, with a small kernel predicting parameters for potential detections, yielding either a category score or a shape offset. This process significantly enhances the system's perception and decision-making capabilities, thereby improving overall accuracy. In the realm of computer vision, YOLO (You Only Look Once) and MobileNet SSD (Single Shot MultiBox Detector) stand out as widely used object detection models. YOLO conducts object detection in a single forward pass by dividing the input image into a grid and making predictions for bounding boxes along with confidence scores. Conversely, MobileNet SSD also performs a single forward pass but employs convolutional filters on feature maps at various scales to detect objects of different sizes. Although YOLO achieves marginally higher mean Average Precision (mAP), MobileNet SSD is optimized for real-time applications, running efficiently even on mobile devices[12]. Thus, we have chosen MobileNet SSD v2, the latest iteration, which sacrifices some accuracy compared to MobileNet SSD v1 but offers improved performance.

TABLE I
YOLO AND MOBILENETSSD V2 COMPARISON

Parameter	YOLO	MobileNetSSD v2
Params	Typically larger	Typically smaller
Multiply-Adds	Higher	Lower
mAP	Varies (depends on variant)	Generally moderate to high
Hardware Platform	Requires powerful GPU	Suitable for mobile CPUs

Sensor Fusion: Algorithms and software for fusing data from the LIDAR sensor and camera to provide depth and direction information.

C. Flowchart

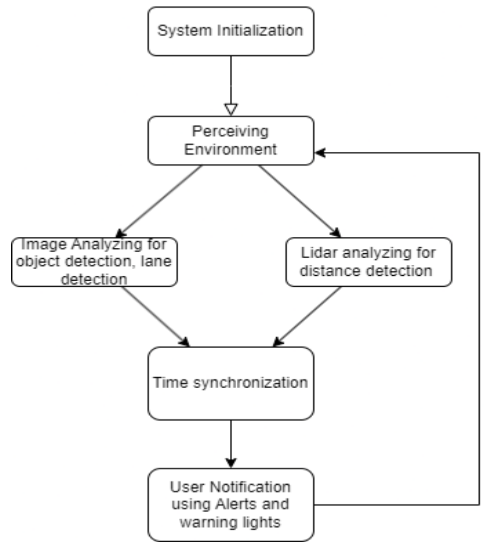


Fig. 7. Flowchart of the system

Raspberry Pi acts as the underlying hardware platform for hosting the system. As shown in Figure 7, the system comprises various processes including initializing system components, capturing images using the Raspberry Pi camera, detecting objects within captured images, and measuring distances to those objects using LIDAR. Additionally, it displays detected objects and their distances to the user, evaluates if detected objects are at a safe distance, and generates alerts if they are too close.

Data is stored in separate data stores: The Image Data Store temporarily holds captured images, the Detected Object Data Store stores information about detected objects, and the Measured Distance Data Store contains distance measurements. Data flows include transferring captured image data for processing, conveying information about detected objects, and transferring distance measurements.

The system operates through different states: Idle, Object Detection, and Distance Measurement. In the Idle state, the system awaits user activation. Upon activation, it transitions to the Object Detection state, which actively identifies and locates objects. If no objects are detected, it returns to the Idle state; otherwise, it moves to the Distance Measurement state to measure distances to the detected objects. If distances are within a safe range, it returns to the Object Detection state. Otherwise, it triggers alert generation due to proximity.

V. RESULTS

The system, which was designed for Indian roads, underwent implementation on different roads. Following this, an analysis was conducted, and the results obtained from the system are presented in this section.

Fig. 8. Real-time object, lane detection and distance estimation on Indian road

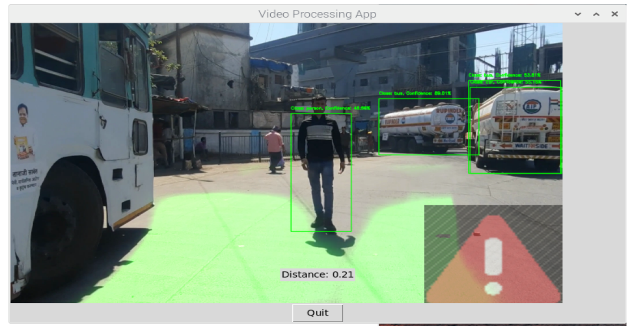


Fig. 10. Giving Alert Signals when the object is near

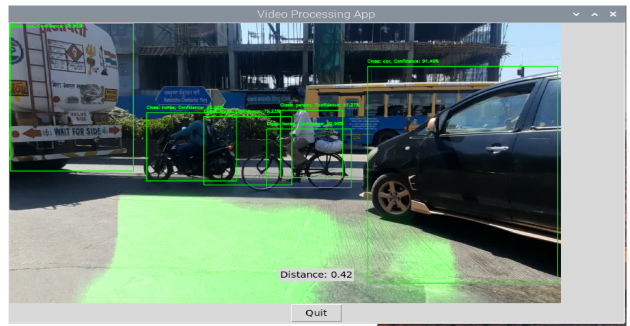


Fig. 11. lane and object detection on a curved road

Fig. 12. Final Result

1) Camera Outcome Illustration: As shown in Figure 8, the objects detected with accuracy were car (98.01%), motorbike(94.64%), and person(60.29%). The green patch shows the detected lane.

LIDAR Visual Representation

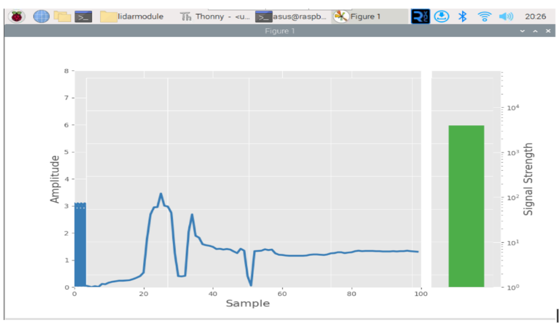


Fig. 9. Live distance measurements provided by the TF-Luna LIDAR

As illustrated in Figure 9, the information obtained from the TF-Luna sensor is presented graphically as signal strength (amplitude) on a bar chart and distance measurements on a time-series graph. Users can discover instances of weak signal strength in certain items or scan routines with the help of this graphical representation.

An alert is given to the user if the distance drops below a predetermined threshold, of 0.40 meters.

Integrated System:

As Figure 10. shows the lane detection and object detection having accuracy of person(56.6%), truck1(89.01%), truck2(55.16%). Since the distance (0.21m) is less than the threshold value an alert is generated.

As Figure 11. shows the lane detection and object detection having accuracy of person1(41.27%), truck(93%), car(91.49%).The distance is 0.42m.

As Figure 12. shows the lane detection and object detection accuracy of the car(99.49%). The distance is 2.33m.

The system code can process video input from a camera or video file, use OpenCV to detect lane lines and train a MobileNetSSDv2 model for object detection. To calculate the distance to the closest object in front of the system, it additionally combines the TF-Luna LIDAR sensor. The processed footage is shown in a graphical user interface window with real-time object and lane line overlays. It also saves the video after processing it into a new file. A real-time plot showing the LIDAR sensor's distance data is also incorporated into the system. An alert icon is incorporated into the video feed to alert the user if the distance drops below a predetermined threshold. The script runs indefinitely to add new data to the plot and update the video feed. Lastly, it can gently close the program and release the video files when you close it.

Object detection model performance metrics:

During road testing in Mumbai, India, the object detection model— MobileNet SSD demonstrated an accuracy range of approximately 70% to 90%.

TABLE II
OBJECT DETECTION MODEL PERFORMANCE METRICS

Model	Params	Multiply-Adds	mAP	Mobile CPU
MobileNetV2 + SSDLite	4.3M	0.8B	22.1%	200ms

The TF-Luna Micro LIDAR sensor has an operating range of 0.2m to 8m. TF-Luna Micro LIDAR sensor error rate:

TABLE III
TF-LUNA MICRO LIDAR SENSOR ERROR RATE

Distance Range (meters)	Error Rate
0 - 3m	± 6cm
3m - 8m	± 2%

VI. CONCLUSION

The system introduces an innovative approach to object detection, combining 2D LIDAR and camera data. The goal is to create a cost-effective and reliable solution that significantly enhances road safety and driver awareness. By leveraging the capabilities of a 2D LIDAR sensor and a Raspberry Pi camera, the system provides real-time visual alerts to drivers, making driving safer. Sensor fusion techniques and robust object detection algorithms are employed to achieve this. The user-friendly interface ensures seamless interaction for drivers. Despite some limitations, such as the constraints of 2D LIDAR and regulatory compliance gaps, the system strives to offer a valuable, low-cost driver assistance solution.

As the system evolves, it remains committed to fine-tuning its performance across diverse environmental conditions and optimizing real-time data processing. With accessibility and effectiveness at the forefront, the driver assistance system aims to meet the dynamic needs of the automotive industry and contribute to the broader goal of road safety and accident prevention. The system is implemented on a Raspberry Pi, processing real-time data from LIDAR for distance measurement and a camera for object detection. This enables smart surround-view and collision avoidance, significantly enhancing road safety.

REFERENCES

[1] Lee, H., Chae, H. and Yi, K. (2019) "A geometric model based 2D LIDAR/radar sensor fusion for tracking surrounding vehicles", *IFAC-PapersOnLine*, 52(8).

[2] Wang, B., Yan, X. and Li, D. (2022) "An end-to-end lane detection model with attention and residual block", *Computational Intelligence and Neuroscience*, 2022.

[3] Ramachandra, A.C. *et al.* (2022) ‘Advanced Driving Assistance System for cars using Raspberry Pi’, *Indian Journal Of Science And Technology*, 15(40).

[4] Liu, H., Wu, C. and Wang, H. (2023) ‘Real time object detection using LIDAR and camera fusion for autonomous driving’, *Scientific Reports*, 13(1).

[5] Ponn, T., Kröger, T. and Diermeyer, F. (2020) ‘Identification and explanation of challenging conditions for camera-based object detection of Automated Vehicles’,*Sensors*, 20(13), p. 3699.

[6] Wang, W., Lin, H., & Wang, J. (2020). "CNN based lane detection with instance segmentation in edge-cloud computing". *Journal of Cloud Computing*, *9*(1)

[7] Haghbayan, Hashem, et al., "An Efficient Multi-sensor Fusion Approach for Object Detection in Maritime Environments," *Advanced Science and Technology Letters*, ISSN: 2287-1233, Vol. 66.

[8] Lee, J. and Hwang, K. (2021) ‘Yolo with adaptive frame control for real-time object detection applications’, *Multimedia Tools and Applications,* 81(25), pp. 36375–36396.

[9] Parekh, D. et al. (2022) ‘A review on Autonomous Vehicles: Progress, methods and challenges’, *Electronics*, 11(14), p. 2162.

[10] Dang, X., Rong, Z. and Liang, X. (2021) ‘Sensor fusion-based approach to eliminating moving objects for slam in Dynamic Environments’,*Sensors*, 21(1), p. 230.

[11] [2] A. Miglani and N. Kumar, "Deep learning models for traffic flow prediction in Autonomous Vehicles: A review, solutions, and challenges,"*Vehicular Communications*, vol. 20, p. 100184, Dec. 2019.

[12] Meyer, G.P.*et al*. (2019) ‘LaserNet: An efficient probabilistic 3D object detector for autonomous driving’, 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[13] Korium, M.S., Abdelsalam, A.A. and Perepelkina, S.Yu. (2021) ‘Lane detection algorithm’, *Izvestiâ vysših učebnyh zavedenij. Priborostroenie,* .

[14] Dai, X. et al. (2020) ‘Lane detection combining details and integrity: An advanced method for Lane Detection’,2020 3rd International Conference on Algorithms, *Computing and Artificial Intelligence*

[15] Kanchana, B. *et al*. (2021) ‘Computer Vision for autonomous driving’, 2021 3rd International Conference on Advancements in Computing (ICAC)