Design and Implementation of Advanced Driver Assistance Systems

Waleed Abd-Elshafi Ali
Department of Communications and Computer Engineering
Higher Institute of Engineering, EL-Shorouk Academy
Cairo, Egypt
w.abdelshafi@sha.edu.eg

Hatem Abdelhamid Saleh
Department of Electronics and Communications Engineering
Misr University for Science and Technology
Giza, Egypt
91145@must.edu.eg

Haneen Salah Eid
Department of Electronics and Communications Engineering
Misr University for Science and Technology
Giza, Egypt
85555@must.edu.eg

Ahmed Mohamed Fathi
Department of Electronics and Communications Engineering
Misr University for Science and Technology
Giza, Egypt
95960@must.edu.eg

Abd-El Rahman Amr Mohamed
Department of Electronics and Communications Engineering
Misr University for Science and Technology
Giza, Egypt
86519@must.edu.eg

***Abstract*— Advanced Driver Assistance Systems (ADAS) are a pivotal advancement in automotive technology, enhancing vehicle safety and driving experience. In our project, we have successfully designed and built a robotic prototype car. We have implemented a comprehensive control system for the car's movement, and the following features of ADAS systems: Bump Detection, Precise Lane Tracking, Adaptive Cruise Control, Traffic Sign Recognition, and Blind Spot Detection. Bump detection systems are equipped to detect and alert drivers of speed bumps and other road irregularities, in which we use NVIDIA Jetson Nano, Astra Pro Plus Depth camera as RGB and Depth sensor and use Custom AI Model based on YOLOv5 to detect different Bumps on the road and we use Python, TensorRT, and OpenCV. Lane Departure Warning (LDW) alerts drivers when their vehicle unintentionally drifts out of their lane, we use NVIDIA Jetson Nano as our main microcontroller unit (MCU) responsible for our Machine learning and Astra Pro Plus Depth camera as our RGB sensor, we use Python, Python's libraries such as OpenCV, and NumPy to implement LDW. Adaptive Cruise Control (ACC) adjusts the vehicle's speed automatically to maintain a safe distance from the vehicle ahead, we use NVIDIA Jetson Nano, Astra Pro Plus Depth camera as RGB and Depth sensor and use Custom AI Model based on YOLOv5 to detect different vehicles on the road, we use Python, TensorRT, and OpenCV. Traffic Sign Recognition can interpret and display traffic signs to keep drivers informed about speed limits, no-entry signs, and more, we use NVIDIA Jetson Nano, and Astra Pro Plus Depth camera as our RGB sensor and use Custom AI Model based on YOLOv5 to detect different Traffic Signs on the road, we use Python, TensorRT, and OpenCV. Blind Spot Detection (BSD) technology helps drivers identify vehicles in their blind spots and alerts them to avoid dangerous maneuvers, we use ESP32 as an MCU and Ultrasonic sensor, and use C++ to program the BSD feature.**

Keywords—ADAS, Robotic car, Bump Detection, Traffic Sign Recognition, Lane Tracing, Adaptive Cruise control, Blind Spot Detection. Object recognition using AI.

I. INTRODUCTION

The tremendous growth of technology in recent years has resulted in a considerable shift of the automobile sector. Advanced Driver Assistance Systems (ADAS) are one noteworthy advancement in this discipline. ADAS is a group of technologies that help drivers with different parts of their journey, improving safety, comfort, and the driving experience. These systems make use of artificial intelligence (AI), cameras, radar, sensors, and cameras to keep an eye on the environment around the car, identify any threats, and send out timely alerts or interventions to stop collisions. Reducing the possibility of human mistake is the main goal of ADAS, as it plays a significant role in most traffic incidents. The World Health Organization (WHO) states that traffic accidents are one of the top causes of mortality globally, with human error being a major contributing factor in almost 90% of these incidents. By supplementing the driver's abilities and awareness and offering extra safety and support, ADAS seeks to overcome this problem [1].

A vast array of features and functionalities, each with a distinct role and addressing various elements of driving, are included in ADAS. Adaptive Cruise Control (ACC), Lane Departure Warning System (LDW), Blind Spot Detection (BSD), Traffic Sign Recognition (TSR), and Bump Detection are a some of the most popular and well-known ADAS technologies. Utilizing sensors or cameras, BSD systems keep an eye on the car's blind zones and notify the driver when another car approaches them via audio or visual cues. This lessens the possibility of crashes when changing lanes or making other maneuvers. LDW systems assist minimize accidents resulting from lane drifting, inattention, or sleepiness by tracking the location of the vehicle inside the lane and informing the driver if an unintentional departure from the lane is detected.

ACC modifies the speed of the vehicle to keep a safe following distance behind the leading vehicle. To maintain a safe space, ACC can automatically apply the brakes or accelerate based on constant monitoring of the distance and speed of the vehicle ahead., enhancing driving comfort and reducing the risk of rear-end collisions.

To recognize and understand traffic signs and give drivers pertinent information regarding stop signs, speed limits, and other traffic rules, TSR systems employ cameras and image processing algorithms. This increases general road safety by helping drivers keep aware of changing road conditions.

A recently introduced feature called "Bump Detection" aims to shield the car and its occupants from any harm from bumps Using sensors and cameras, the system can identify objects or obstructions close to the car and trigger safety features like applying the brakes or alerting the driver audibly. This adds an extra degree of security when parking or in confined spaces where visibility may be restricted.

The main contributions of the paper are as follows:

Design and Implementation of prototype robotic car, the Car Movements was achieved using a DC motor and the steering system, and control of the car was done through a PlayStation-4(PS4) controller.

Design and Implementation of Blind Spot System, utilizing an ultrasonic sensor. A flowchart was created to measure the levels of danger based on the distance between the car and other objects and provide warnings on the PS4 controller.

Research was done to find the best model for Lane Departure detection, using an Astra pro plus Depth camera to provide video input, and using Canny edge detection to determine the slope of lane departure.

The YOLO5 model was trained to detect other cars and measure the distance between the car and other vehicles using the Astra pro plus Depth camera, providing warnings on the PS4 controller.

The YOLO5 model was trained to detect speed bumps and measure the distance between the car and the speed bump, providing warnings on the PS4 controller.

The YOLO5 model was trained to detect a set of traffic signs and provide the driver with notifications of the traffic signs.

The remainder of this paper is arranged as follows. In Section II, the related work is discussed. In Section III, the proposed system and its features are introduced. and the main conclusions are drawn in Section IV.

II. RELATED WORK

Prior studies that are detailed in these papers [2,], [3], [4], [5], and [6] employ GPS and accelerometers to identify speed bumps. These methods analyze the accelerometer data after the car passes the speed bump to detect speed bumps; they are not real-time. Cloud storage is used in methods outlined in [7], and [8] to store data on speed bumps. The vehicle's driver will only get alerts for speed bumps known to exist and for which cloud storage has the necessary information. The devices cannot detect unknown speed bumps, and problems with network access might cause the motorist to get a warning at the incorrect place.

There are several studies on traffic sign recognition and hundreds of training datasets. The German Traffic Sign Recognition Benchmark (GTSRB) [9] is a well-known dataset including 39,209 training photos and 12,630 test images from 43 traffic sign classes. The dataset is known for its historical significance. in [10], a model for traffic sign identification was created that combines Deep Neural Networks (DNNs) trained on diverse preprocessed data, making it immune to differences in contrast and lighting. The final prediction is derived by averaging the outputs of each DNN. They discovered four different preprocessing methods, and as a result, five nets—one for each of the five datasets—were trained, for a total of twenty-five nets. We didn't think this was the best course of action to take, considering the significant real-time limitations, even if each network is extremely tiny and light in terms of computing. The prior method's invariance to translation is another flaw. Our working environment is highly dynamic; thus, we have adopted an invariant approach to brightness, contrast, and translation.

the distance from the previous vehicle is frequently calculated using a geometric technique using a monocular camera. Most known research on calculating distance from a preceding vehicle considers two categories of objects: the vehicle and its plate. After identifying and localizing one of these two items in our image, we determine the size in pixels, followed by the distance from the object. For example, [11] makes extensive use of preprocessing to find the license plate edges of a car, primarily using Canny edge detectors [12] and the Probabilistic Hough Transform Method [13]. [14] instead takes use of the fact that the background color of a Chinese license plate is blue for larger cars and yellow for smaller ones. Using these properties, they determine the precise angles of the plate and, from there, its size in pixels. These approaches are the most accurate because they are based on the size of the plate, which is known a priori for most countries, but they rely on strong assumptions: the size of the plate is fixed, but it may vary slightly from country to country, it is assumed that the method returns exactly the bounding box of the plate, and these methods do not work if the previous vehicle is a long distance away. NVIDIA's DRIVE Labs developed a deep neural network model that uses automatically labeled radar and LiDAR data to estimate distances from objects. This is undoubtedly a more novel technique, but it is also far more computationally costly than any previous geometric alternatives. To achieve a real-time and universally applicable solution, we used a distance-calculating technique based on the previous vehicle's bounding box.

III. PROPOSED SYSTEM

We will discuss each feature of the proposed system, its operation, the sensor used, algorithms, and the block diagram and flowchart for each feature.

A. Car Movement:

The movement of the car is orchestrated by a confluence of mechanical, electrical, and software systems working in harmony. At the heart of the car movement lie the following key components:

- Chassis and Drivetrain

The chassis is the foundation that supports the car and provides structural integrity. The drivetrain, consisting of the motor, gearbox, and wheels, converts electrical energy into mechanical motion. In our project, a compact and efficient drivetrain system is crucial for smooth and controlled movement. The chassis is designed to handle the weight of the components. It features a rear wheel drive, with the front wheels moving right and left for steering, replicating realistic car movement.

- Motor Control System

The motor control system oversees the motors that propel the car, handling speed, direction, and torque for optimal movement dynamics. It must respond swiftly to input signals from sensors and control algorithms for precise maneuvering. Our project employs two motor types: a DC motor drives the rear wheels forward and backward, simulating an engine, while a servo motor steers the front wheels. These motors are carefully selected to handle the chassis weight. The DC Motor we chose has a torque of 8.8kg/cm and 250 rpm. Because of the car's weight, this torque is more than enough for our needs. The Servo Motor has a torque of 8.5kg/cm and a speed of 0.2s/60°. That torque is enough to make smooth steering for the front wheels.

- ESP32 Microcontroller Unit (MCU)

The brain that controls all the movement is the ESP32 microcontroller. The reason for choosing this MCU is that it has over 20 Pulse Width Modulation (PWM) pins, Wi-Fi, and Bluetooth connectivity, is easy to use, and its cost is very low. The ESP32 acts as a slave for the Master (Nvidia Jetson Nano MCU). The Jetson Nano will give the ESP32 signals of any alarm system in our project, and the ESP32 must take appropriate action to avoid the cushion.

B. Blind Spot Detection System (BSD):

The BSD system increases driver awareness and reduces the risk of accidents brought on by poor vision in blind spot zones. To monitor blind areas and identify objects or vehicles, it makes use of sensors such as radar, ultrasonic waves, and cameras. The system uses haptic, aural, or visual signals to notify the driver when a possible threat is detected. The efficiency of integration with other safety measures is increased. But bad weather can still impact system operation, so drivers should continue to rely on mirrors and eye inspections. BSD systems are supplementary tools; they shouldn't take the place of alertness and careful driving.

- BSD System Working Principle

Because ultrasonic sensors can measure near distances, we have chosen them for blind spot identification. To identify blind areas, these sensors are angled and placed on the right and left sides next to the mirror. The ESP32 microprocessor, which gets distance information from the ultrasonic sensors, is attached to them. Decisions are made based on the readings, and notifications are given out via vibrations and flashing lights.

- Ultrasonic Sensor

There are two ultrasonic transducers in the HC-SR04 Ultrasonic Distance Sensor. One functions as a transmitter, producing ultrasonic sound pulses at 40 KHz from an electrical signal. The transmitted pulses are heard by the receiver. Should it obtain them, it generates an output pulse whose width may be utilized to ascertain the pulse's distance traveled. The sensor provides excellent non-contact range detection between 2 cm and 400 cm with an accuracy of 3mm. It is tiny and simple to utilize in any robotics project. It may be connected straight to an ESP32 or any other 5V logic microcontroller because it runs on 5 volts [15].

The description of BSD operation is indicated by the flow chart in Fig.1. and the test result analysis in Fig.2.

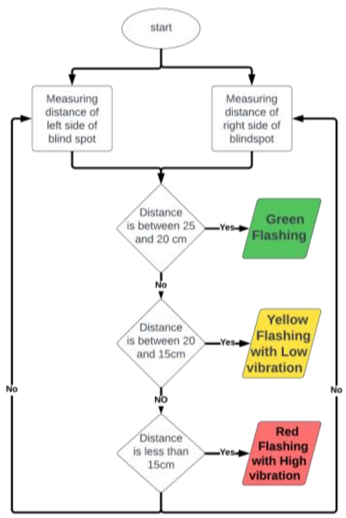

Fig. 1. Flow Chart for BSD System.

- Our BSD System Test Result Analysis

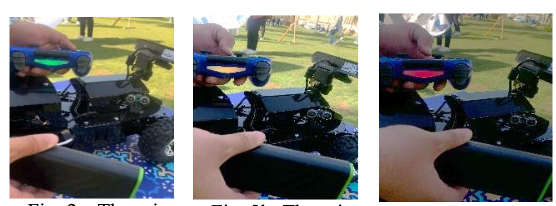

Fig. 2a. There is a car within a safe distance.

Fig. 2b. There is a car, distance between 20 cm to 15 cm.

Fig. 2c. There is a car within danger zone.

Fig. 2. BSD system Test results.

C. Lane Departure Warning System (LDW):

On highways and comparable roadways with one- or two-lane markings, LDW systems notify drivers when they inadvertently stray from their intended lane. The direction of the vehicle is outside the control of LDW systems. Rather, they notify the motorist and encourage them to take the necessary precautions to avoid unexpected movements. Two warning zones are usually defined by LDW systems. The position detecting unit, which identifies lane markers and assesses whether a warning should be sent based on the lateral departure from the lane boundary, is one of the essential parts of LDW systems. Although they are not required, the rate of departure and curvature of the next road section might be considered [16].

- Warning elements

The driver must be made aware of a warning feature by using one of their senses. The visual and audible channels are utilized for LDW systems.

- Methodology

1. Input Video Selection

   Input video from the Camera mounted on the Car.

2. Canny Edge Detection

   Using a 2-D FIR filter, canny edge detection is done to a chosen picture before the auto thresholding value is applied. The edge-detected picture is obtained. Edge detection aims to drastically decrease a picture's data while maintaining its structural integrity, which may be utilized for additional image processing.

3. Hough Transform

   The Hough transform is the method used to recognize the vertical lines in the filtered picture and retain the most likely lane boundary [17].

4. Our LDW system Test Result Analysis

   The video player on display shows the type and color of the lane markers. It also shows the left and right lane markers and warning messages as indicated in Fig.3.

Fig.3a. original image. Fig.3b. blurred gray image. Fig.3c. canny edge detection.

Fig.3d. Lower Portion of Input Video Frame and Hough transform.

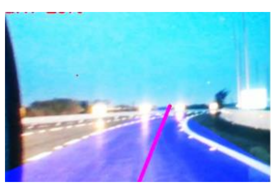

Fig.3e. Lane departure.

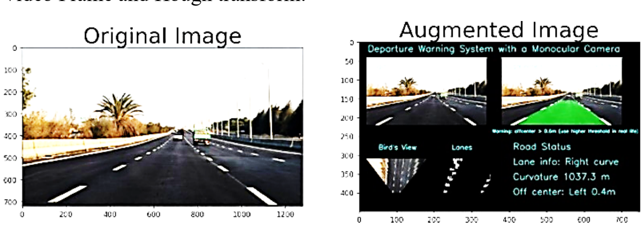

Fig.3f. Comparison between original image and augmented image.

Fig. 3. LDW system Test Result.

D. Adaptive Cruise Control (ACC)

One crucial component of ADAS is ACC. To maintain a safe following distance, it automatically modifies a vehicle's speed when it encounters slower-moving traffic. The ACC automatically picks up speed to your preset speed when there is clear traffic ahead [18].

- ACC System Working Principle:

1. using RADAR or LIDAR technology to determine the speed and distance of the cars ahead.
2. The key to measuring distance is the time it takes for transmission and reception.
3. To determine the speed, the Doppler Effect's shift in the reflected beam's frequency is observed.
4. The brake and throttle controls are set based on this speed to maintain the vehicle in a safe position [19].

- Vehicle Distance Estimation

We needed to calculate our distance from the car in front to use adaptive cruise control effectively. Therefore, our goal was to use an image from a single camera to calculate the distance between the camera and an item. Considering the Pinhole camera model, this may be accomplished by making use of the triangles' resemblance. It is simple to calculate the distance between the object and the camera using the formula in equation (1) if we know the focal length of the camera and the actual height of the item from which we need to compute the distance [20].

$$Distance = \frac{focal\ length \cdot real\ vehicle\ height}{image\ vehicle\ height} \tag{1}$$

- Our ACC System Result Analysis

Our project results due to different tests of ACC system implementation are indicated in Fig.4.

Fig. 4a. Oure Implementation of ACC system test1.

Fig.4b. Oure Implementation of ACC system test2.

Fig. 4c. Oure Implementation of ACC system test3.

Fig. 4. ACC Test results.

E. Bump Detection System:

Bump detection makes use of sensitive sensors to find obstructions and abnormalities in the road. It enhances safety and comfort while driving by measuring abrupt changes in acceleration and alerting the driver in real time.

- Bump Detection System Working Principle

The bump detection system utilizes an Astra Pro Plus depth camera and Nvidia Jetson Nano microcontroller. The depth camera captures the surrounding environment and generates a depth map based on infrared projections. The depth data is processed by the Jetson Nano, which analyzes the changes in depth values to detect bumps on the road. The system then triggers alerts or adjusts vehicle settings to mitigate the impact of the bump on the driver and passengers.

- Our Bump Detection System Result Analysis

Fig.5a and Fig.5b indicate our results of Bump Detection System.

Fig.5a. Bump Detection test1 Fig. 5b. Bump Detection test2

Fig. 5. Bump Detection Test results.

F. Traffic Sign Recognition System:

An essential part of ADAS is traffic sign recognition, which helps cars understand and react to traffic signs efficiently. Road signs, including stop signs, speed restrictions, and directional markers, may be reliably detected and classified by ADAS systems, which improve driver awareness, lessen cognitive burden, and support decision-making. Vehicles can adjust to changing road conditions, maneuver through intricate traffic scenarios, and follow traffic laws thanks to real-time traffic sign recognition. Therefore, traffic identification is essential for improving ADAS capabilities and encouraging safer driving practices [21].

- Use of AI and Machine Learning

In ADAS, strong traffic sign recognition is made possible by AI and machine learning. Convolutional neural networks use big datasets and real-time analysis of onboard video data to precisely recognize symptoms. With its superior real-time object-detecting capabilities, YOLO and SSD versions are perfect for ADAS. They promote driver safety and progress in the field of autonomous driving [22].

To improve real-time processing and accuracy in traffic sign identification systems, scholarly research in AI for traffic recognition focuses on investigating several object detection models. Scholars assess many models, including YOLO, SSD, Squeezed, MobileNet, and Efficient, based on how well they identify and categorize traffic indicators in dynamic environments. Comparative evaluations draw attention to the advantages and disadvantages of each model and shed light on which ones are best suited for certain uses in intelligent transportation systems [23].

1. Dataset

The Working principle depends on large-scale datasets that are essential for creating and assessing models for traffic sign recognition. These datasets guarantee the efficacy of models in practical situations by providing the basis for their training and testing. The selection of datasets is based on several factors, such as geographical variety, representation of different types of traffic signs, differences in lighting, size of the dataset, quality of the annotations, and the inclusion of tough scenarios to improve the resilience of the model.

2. AI Model Development and Training

AI models that go through iterative development and training include YOLO, SSD, MobileNet, TinyYOLO, and SqueezeDet. Models are trained using labeled datasets so they can pick up pertinent patterns and characteristics. Preprocessing operations are carried out to maximize model performance for real-time applications, such as data augmentation, resizing, and normalization. Thorough testing is conducted utilizing different datasets to assess metrics like accuracy, precision, and recall after the training process.

3. Evaluation Metrics

When evaluating how well AI models perform in traffic sign recognition, evaluation measures are essential. Several metrics that provide light on the efficacy and accuracy of model predictions are True Positive (TP), False Positive (FP), False Negative (FN), Intersection over Union (IoU) Precision, Recall, Average Precision (AP), and Mean Average Precision (mAP). These metrics enable the assessment of multi-class object detectors in a variety of classes, guaranteeing the efficacy of AI models in precisely identifying and detecting traffic signs [24].

- Performance on Traffic Recognition

Fig.6 indicates sample of the traffic sign from long distance in the real life after 40 iterations, while Fig.7 indicates sample of the traffic sign from short distance in the real life after 10 iterations.

Fig. 6. sample of the traffic sign from long distance in the real life

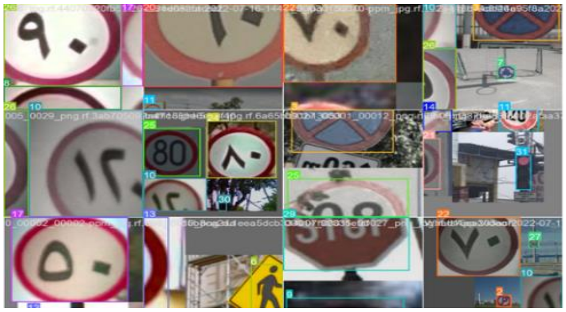

Fig.7. sample of the traffic sign from short distance in the real life.

Fig.8 indicates "Model Performance Plot" to convey its purpose of assessing the relationship between predicted and true values.

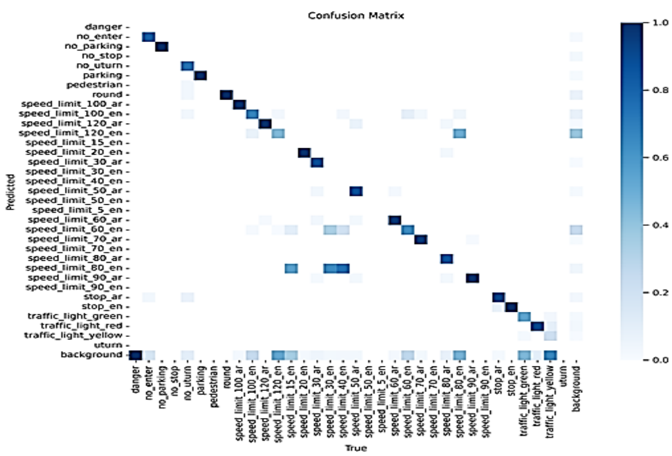

Fig.8. Predicted versus. True (Actual).

IV. CONCLUSIONS

We have successfully designed and built a robotic car. We have implemented a comprehensive control system for the car's movement using an ESP32 microcontroller in conjunction with a PS4 controller. Additionally, we have integrated two ultrasonic sensors, one on the right and one on the left, at specific angles to detect blind spots for the driver. When an object or another car approaches the blind spot area, an alert is triggered. Also, we Successfully designed and implemented Lane Departure Warning, Adaptive Cruise Control, Bump Detection, and Traffic Sign Recognition Systems. Our project underscores the potential of technology to revolutionize road safety and improve the lives of drivers worldwide. As we continue to refine and expand upon our work, we are committed to making driving safer, more efficient, and more enjoyable for everyone.

REFERENCES

[1] Kathiresh, M., and R. Neelaveni. *Automotive Embedded Systems.* Springer International Publishing, 2021.

[2] A. Mednis, et al., “Real time pothole detection using Android smartphones with accelerometers, " 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), 1-6 (2011).

[3] V. Rishiwal, H. Khan, “Automatic pothole and speed breaker detection using android system, " 2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 1270-1273 (2016).

[4] Q. Chen, et al., “A Speed Hump Sensing Approach to Global Positioning in Urban Cities without GPS Signals," 2017 IEEE International Conference on Smart Computing (SMARTCOMP),1-8 (2017).

[5] P. Mohan, V. Padmanabhan, R. Ramjee.Nericell, “ Rich monitoring of road and traffic conditions using mobile smartphones, " Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems 323-336 (2008).

[6] F. E. Gunawan, Yanfi, B. Soewito, “A vibratory-based method for road damage classification, " 2015 International Seminar on Intelligent Technology and Its Applications (ISITIA), 1-4 (2015).

[7] M. V. K. Kiran, K. Vimalkumar, R. E. Vinodhini, R. Archanaa., “An Early Detection Warning System to Identify Speed Breakers and Bumpy Roads using Sensors In Smartphones, " International Journal of Electrical and Computer Engineering, 1377-1384 (2017).

[8] P. R. Pooja, B. Hariharan, “An early warning system for traffic and road safety hazards using collaborative crowd sourcing, " 2017 International Conference on Communication and Signal Processing (ICCSP), 1203-1206 (2017).

[9] Sebastian Houben, et al., “Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark, " In International Joint Conference on Neural Networks, 2013.

[10] Dan Cires¸an, et al., “Multi-column deep neural network for traffic sign classification. Neural networks, " the official journal of the International Neural Network Society, 32:333–8, 02 2012.

[11] Purnama Sari Nur Mellinda, Febryanti Sthevanie, and Kurniawan Nur Ramadhani, “Detection of vehicle number plate using probabilistic hough transform, " eProceedings of Engineering, 7(2), 2020.

[12] John Canny, “A computational approach to edge detection, " IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6):679–698, 1986.

[13] Jiri Matas, Charles Galambos, and Josef Kittler, “Robust detection of lines using the progressive probabilistic hough transform, " Computer vision and image understanding, 78(1):119–137, 2000.

[14] Wenfeng Wang, Shujun Yang, Yong Li, and Weili Ding, “ A rough vehicle distance measurement method using monocular vision and license plate, " In 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pages 426–430. IEEE, 2015.

[15] Augustine, Stephen Tipa, “Design and Implementation of a Blind Spot Detection and Monitoring System for The Kayoola Buses, " Diss. 2022.

[16] Azim Eskandarian, *Handbook of Intelligent Vehicles*. First Edition, Springer-Verlag London Ltd. 2012.

[17] Mahajan, Rashmi N., and A. Patil. "Lane departure warning system." *International Journal of Engineering and Technical Research* 3.1 (2015): 120-123.

[18] Piccinini, Giulio Francesco Bianchi, et al. "Driver's behavioral adaptation to Adaptive Cruise Control (ACC): The case of speed and time headway." *Journal of safety research* 49 (2014): 77-e1.

[19] Marsden, Greg, Mike McDonald, and Mark Brackstone. "Towards an understanding of adaptive cruise control." *Transportation Research Part C: Emerging Technologies* 9.1 (2001): 33-51.

[20] Zhou, Chong. *Yolact++ Better Real-Time Instance Segmentation*. University of California, Davis, 2020.

[21] Khan, Muneeb A., Heemin Park, and Jinseok Chae. "A lightweight convolutional neural network (CNN) architecture for traffic sign recognition in urban road networks." *Electronics* 12.8 (2023): 1802.

[22] Kukkala, Vipin Kumar, Sooryaa Vignesh Thiruloga, and Sudeep Pasricha. "Machine Learning for Anomaly Detection in Automotive Cyber-Physical Systems." *Embedded Machine Learning for Cyber-Physical, IoT, and Edge Computing: Use Cases and Emerging Challenges*. Cham: Springer Nature Switzerland, 2023. 253-283.

[23] P. Azevedo, "What is the best YOLO," 8 Jun 2022. [Online]. Available: https://medium.com/@pedroazevedo6/what-is-the-best-yolo-8526b53414af. [Accessed 2023].

[24] "dagshub," 28 6 2022. [Online]. Available: https://dagshub.com/blog/yolov6/.