Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 45 No. 3
March 2024

Obstacle Detection and Avoidance Using Sensors and Deep Learning Models

Rishika Soni[1], Tina Chaudhary[2*], Mansi Mandal[3], Shilpee Gupta[4], Ankita Singh[5]

[1,2,3,4,5] Department of Mechanical and Automation Engineering, Indira Gandhi Delhi Technical University For Women, New Delhi, Delhi110006

Abstract: To study various technologies used in order to solve the problem of obstacle detection and avoidance and develop a system capable of efficiently detecting and avoiding obstacles and selecting a path that is free from congestion. This system integrates smart applications with computer vision, Arduino and sensor technology obstacle detection. It autonomouslyidentifies obstacles and avoids them by using the ultrasonic sensors. For the CNN implementation for Level 2 Autonomous Vehicle, a CNNapproach with mapping of camera input pixels to steering commandsand testing on CARLA open-source driving simulator, utilizing an ultrasonic sensor and an RGBD camera for obstacle detection and real- time position monitoring at 10Hz. It incorporates the usage of RRT-Connect algorithm for path planning, employing Arduino Mega and Raspberry Pi for motor control and processing. Perception Module Development for Obstacle Detection in Farm Equipment involves usage of multiple sensors for obstacle detection and motion control in farm equipment with a combination of camera, mini Lidarmodule, and ultrasonic sensors for accurate obstacle detection and distance measurement.

Keywords: CNN, RCNN, Arduino, Sensor

Introduction:

With the increase in the use of technology with the rapid and advancement of technology and usage of automated devices, the usage of automobile vehicles has also increased. The efficient and accurate navigation of the autonomous vehicles is challenged by the real world issues such as static and dynamicobstacles and selection of a free path for movement. To counter this the development of a system that detects and avoids obstacles efficiently is required. By using artificial intelligence, a technology capable of replicating human intelligence along with effective hardware control, the far of smooth, effortless and efficient autonomous navigation can be realised. Use of machine learning models that works on R-CNN network along with Arduino and ultrasonic sensor increases the efficiency of the obstacle detection and avoidance system. This system will not only ensure smooth navigation but also help in decreasing the accidents or collisions caused by automobiles. Use of machine learning model R-CNN will enable the system to intelligently work without any human intervention . Such system will prove to contribute in building safer, more efficient and self - sufficient automobile or mobile devices.

1.1 Literature Review:

Mohd Sani et.al(2012)[1] constructed a robot that will make up for time lost avoiding obstacles and arrive at the destination at the designated time. It also demonstrated how the robot determined the deviation point and created a new course to avoid the obstruction. In order to provide the robot better control over its movements, this study used a polynomial curve to ensure that it passed through each control point.

Shusmita M. Rathod et.al (2019) [2] presented a project aimed at creating an autonomous vehicle obstacleavoidance system utilizing infrared (IR) sensors and Arduino-based technologies. The design and implementation of the system were deliberated, encompassing hardware components such the L298N motor driver, dual-shaft DC geared motors, short and long-range infrared sensors, and Arduino Mega board. Additionally, the studysheds light on how these parts work together and are integrated.

Jiechao Liu et. al.(2016)[3] In order for a Model Predictive Control (MPC) based obstacle avoidance algorithm for safe and efficient obstacle avoidance even when the vehicle is about to exceed its dynamic limit, this work investigates the level of model accuracy required. The author concludes that a two-DoF representation that takes into account tire nonlinearity and longitudinal load transfer is necessary for an MPC-based algorithm to operate the vehicle as efficiently as possible in environments containing obstacles.

Akhil Agnihotri et. al.[4] (2019) The research study uses a convolutional neural network CNN techniqueof mapping the camera input pixels to the steering commands in order to build a level 2 autonomous car. Because the network derives the many changeable features from the camera input. When real time framesare provided as input, the driving strategy that NVIDIA and Udacity trained on the datasetmay be adjusted to simulate real-world driving conditions. CNN is tested using the CARLA driving simulator. A comprehensive description is also given of a beta-testing platform that includes an ultrasonic sensor for obstacle identification and an RGBD camera for 10Hz real-time position monitoring.

Magda Skoczen et.al(2021) [5] The model was fine-tuned using the EDEN dataset, which contains pictures from 20 distinct artificial gardens. There are five image sequences for every scene, and a total of49,762 sets of data were used to assess the suggested detection system. 52,502 pairs of RGB photos and the matching image segmentation mask were used to train the models.

Bhasha Pydala et.al(2023)[6] This study suggested using deep learning algorithms, smartphone-based solutions, infrared, ultrasonic, and sensor-based systems. This research is dual-purposed: A smart eye model that helps people with vision impairments overcome obstacles and incorporates AI and sensor technologyinto a smart application to support these people. This was done in order to create a voice-basedsystem and wearable gadget that is both affordable and lightweight. The created system had a 95% detection accuracy during the dayand a 76% detection accuracy at night.

Jianying Yuan et.al(2023) [9] presented a novel method for dynamic obstacle identification that is essential for driverless driving. The technique uses residual optical flow and U-V discrepancy to identifythe drivable area and any obstructions within it. It efficiently detects barriers by using U-V disparity images. Notably, it greatly improves detection accuracy and efficiency by reducing the search range for moving impediments inside the driveable region. High efficiency, low missed rate of detection and low time consumption were proved by the approach through testing on datasets from the KITTI and self- acquired scene data.

Amitabh Das et. al.[8] (2021) Creating a perception module with a variety of sensors— including ultrasonic sensors for close-range detection—is the main goal of this project. The system makes use of ultrasonic sensors for obstacle identification, a tiny Lidar module for a restricted view and simultaneous distance measurement, and a camera for object detection and measurement using OpenCV DNNs. Actuators that regulate steering, braking, and acceleration are utilized for the motion of the vehicle based on data gathered by the system.

Anand Kumar Kyatsandra et al. (2021) [9] developed TRINETRA, a device that would address operational challenges encountered by Indian Railways, using cameras, imaging sensors, ML algorithms, and real-time processing technologies. When drivers constantly glance outside the locomotive to evaluate the weather for improved visibility of the obstacle, they can employ LIDAR, RADAR, ultrasonic sensors,and infrared sensors. This is helpful on foggy nights and in smog circumstances. The device's accuracy was 6.9 km for human view distance and 12.6 km for vehicle view distance.

Ravpreet Kaur et. al.(2022)[10] The various facets of object detection have been thoroughly examined inthis work. The performance of object identification models has significantly improved as a result of the continuous development of deep learning techniques for object recognition. This review paper begins witha quick introduction to object detection before delving into frameworks for object identification, commondatasets, and evaluation techniques. Comprehensive coverage is also given to object detection applications and issues. Results are presented after assessing object detection models' performance usingthe MS COCO and PASCAL VOC datasets.

2 Method:

2.1 EXPERIMENTAL SETUP:

An autonomous car outfitted with a motor driver, a DC motor, an Arduino UNO microprocessor, and an ultrasonic sensor was used to conduct the experiment. A 9V HW battery serves as the model's power supply. It also has an actuator that consists of a DC motor, a servo motor, and a sensor that can be moved from 0 to 180 degrees. The wheelis powered by a DC motor, which enables it to spin in the desired directions. By controlling the input and output signals from the ultrasonic sensor, obstacle detection can be achieved. Signal processing, signal conditioning, parameterization, and integration make up the input and output signal management. Moreover, Arduino IDE 2.2.1 is utilized to code and debug the program for the Arduino control (as shown in Fig.no. 1)

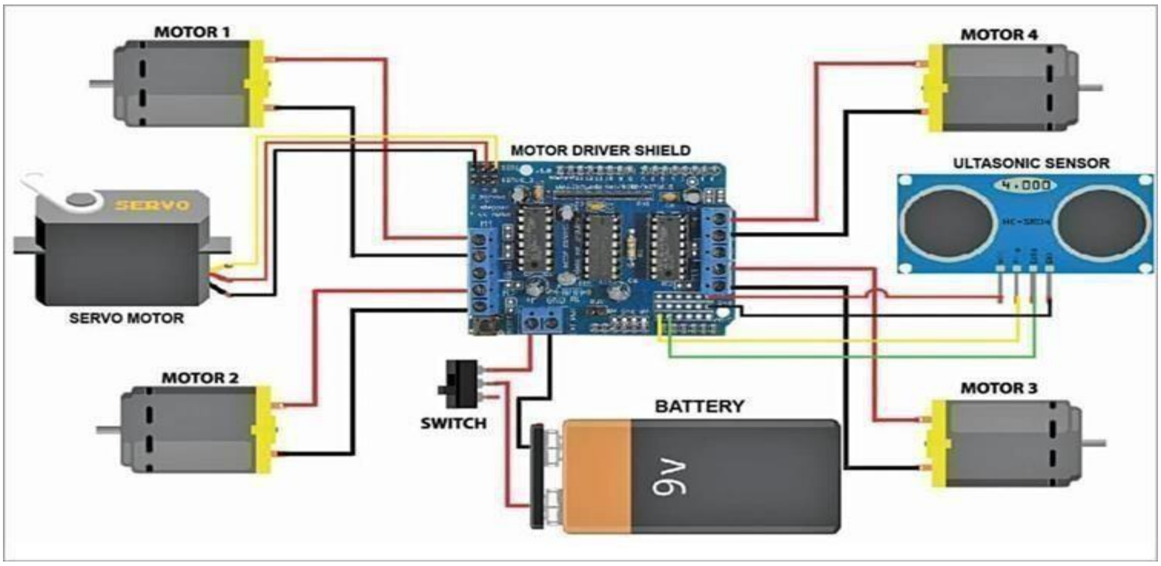


Fig1: Car Model Setup Diagram

2.2 METHODOLOGY:

Within the field of artificial intelligence (AI), machine learning is the branch that focuses on creating models and algorithms that let computers learn from data and make judgments or predictions without explicit programming. Giving computers the ability to automatically learn from experience and get better at it is the fundamental principle of machine learning.

2.2.1 MACHINE LEARNING MODELS

Computational algorithms or systems known as machine learning models are created to recognize patterns and automatically generate predictions or judgment calls without the need for explicitprogramming.

Through the application of statistical approaches, these models allow computers to perform better overtime when theyare exposed tomore data on a particular activity.

PRE-TRAINED MODELS:

Models which we have used for our project are as follows:-

i. <u>MobileNet:</u>

MobileNet is a model that filters images using convolution in a similar manner to CNN, but in adifferent method than CNN's previous model.

In contrast to the standard convolution carried out by standard CNNs, it makes use of theconcepts of depth convolution andpoint convolution.

This makes CNN more effective at predicting images, enabling them to compete in mobilesystems as well.

ii. <u>ResNet:</u>

Residual Network, is a type of deep neural network architecture that was introduced to address the challenges of training very deepneural networks.

For a variety of purposes, pre-trained ResNet models are frequently available. These models are trained on extensive datasets, like ImageNet, toextract general features from a wide range ofimages.

A variety of computer vision tasks, including segmentation, object identification, and image classification, can be started with these pre-trained models.

iii. <u>VGG16:</u>

VGG16 (Visual Geometry Group 16) is a convolutional neural network (CNN) architecture.

It serves as a strong baseline for various computer vision tasks and has been used as a featureextractor or a pre-trained model for transfer learning.

The model is often pre-trained on large datasets like ImageNet, where it learns to recognize awide variety of visual patterns.

iv. <u>INCEPTION:</u>

Utilizing Inception modules—blocks comprising several parallel convolutional filters of varying sizes—is the main concept behind the program. Information is captured by these parallel filters at various spatial scales.

Inception usually employs global average pooling at the end of the network instead of fully connected layers.

In order to do this, the spatial dimensions are reduced to one value per channel by averaging each feature map. This aids in lowering the network's parameter and computation count.

v. <u>XCEPTION:</u>

Extreme Inception, is a deep convolutional neural network (CNN) architecture that represents an extreme version of the Inceptionarchitecture.

Xception uses depthwise separable convolutions in place of the conventional convolutional layers in traditional architectures. Pointwise and depthwise convolutions are the two steps that make up depthwise separable convolutions. By applying a singleconvolutional filter to each input channel, the depthwise convolution independently captures spatial information.

3. RESULTS:

Results obtained from different model implementations are as below:

i. MobileNet: By adding extra layers for computation the pre-trained MobilNetV2 model was able to achieve a validation accuracy of 82%. Fig2(a) shows the training and validation accuracy and fig2(b) shows the training and validation loss for MobileNet. Fig3 shows the pictorial representation for the true positive,true negative ,false positive and false negative values.Table 1 compares the different characteristic measures for obstacle and no obstacle.

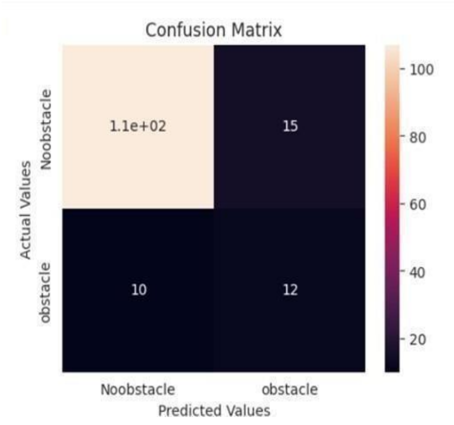


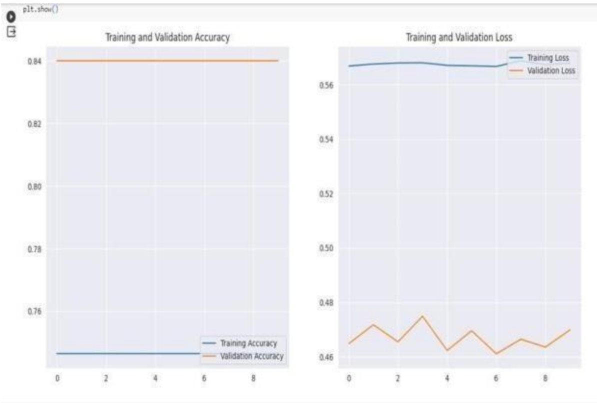
Fig.2.(a)AccuracyLossforMobileNetV2Fig.3 Confusion Matrix of MobileNetV2 model

Table .1 Value of metrics evaluated for MobileNetV2 model

	Precision	Recall	F1-Score	support
Obstacle	0.91453	0.87705	0.89540	122
No_Obstacle	0.44444	0.54545	0.48980	22
Accuracy			0.82639	144
Macro Avg.	0.67949	0.71125	0.69260	144
Weighted Avg.	0.84271	0.82639	0.8334	144

ii. ResNet: By adding extra layers for computation the pre-trained resnet model was able to achieve a validation accuracy of 84 %. The Recall-score is 1 for obstacle. Fig4(a) shows the training and validation accuracy and fig4(b) shows the training and validation loss for ResNet.Fig5 shows the pictorial representation for the true positive,true negative ,false positive and false negative values.Table2 compares the different characteristic measures for obstacle and noobstacle.

Fig. 6(a) Accuracy(b)Loss for Inception model

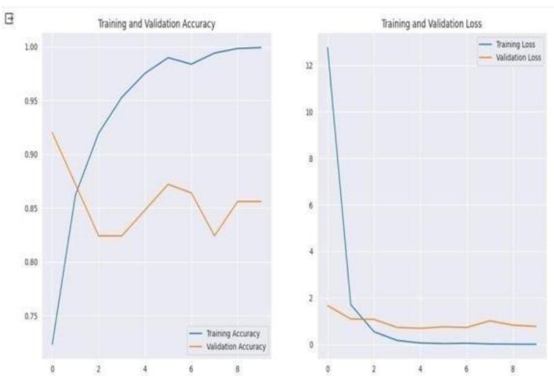


	Precision	Recall	F1-Score	support
Obstacle	0.8400	1.00000	0.91304	105
No_Obstacle	0.00000	0.00000	0.00000	20
Accuracy			0.84000	125
Macro Avg.	0.42000	0.50000	0.45652	125
Weighted Avg.	0.70560	0.84000	0.76696	125

(b) Loss for ResNet **Fig. 5 Confusion Matrix for ResNet model**

<u>Inception:</u> By adding extra layers for computation the pre-trained Inception model was able to achieve a validation accuracy of 85%. Fig6(a) shows the training and validation accuracy and fig6(b) shows the training and validation loss for inception. Fig7 shows the

iii. pictorial representation for the true positive,true negative ,false positive and false negative values.Table3 compares the different characteristic measures for obstacle and no obstacle.



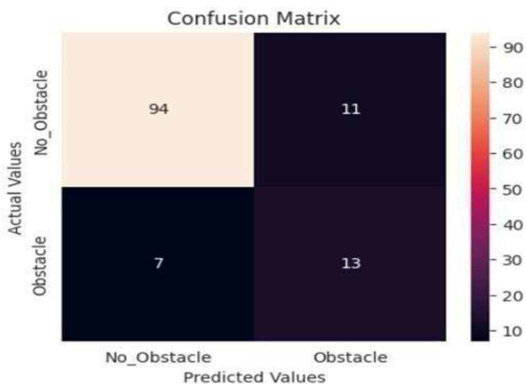


Table 3.Value of metrics evaluated for Xception model

	Precision	Recall	F1-Score	support
Obstacle	0.93069	0.89524	0.91262	105
No_Obstacle	0.54167	0.65000	0.59091	20
Accuracy			0.85600	125
Macro Avg.	0.73618	0.77262	0.75177	125
Weighted Avg.	0.86845	0.85600	0.86115	125

iii. Xception: By adding extra layers for computation and fine-tuning the hyperparameters the pre- trained model was able to achieve a validation accuracy of 84% . The Recall-score is 1 for obstacle. Fig8(a) shows the training and validation accuracy and fig8(b) shows the training and validation loss for xception. Fig9 shows the . pictorial representation for the true positive,true negative ,false positive and false negative values.Table4 compares the different characteristic measures for obstacle and no obstacle.

Fig 8 (a)Accuracy (b)Loss for Xception model

	Precision	Recall	F1-Score	support
Obstacle	0.84000	1.00000	0.91304	105
No_Obstacle	0.00000	0.00000	0.00000	20
Accuracy			0.84000	125
Macro Avg.	0.42000	0.50000	0.45652	125
Weighted Avg.	0.70560	0.84000	0.76696	125

	Precision	Recall	F1-Score	support
Obstacle	0.93684	0.84762	0.89000	105
No_Obstacle	0.46667	0.70000	0.56000	20
Accuracy			0.82400	125
Macro Avg.	0.70175	0.77381	0.72500	125
Weighted Avg.	0.86161	0.82400	0.83720	125

VGG16:By adding extra layers for computation and fine-tuning the hyperparameters the pre - trained model was able to achieve a validation accuracy of 82% . Fig10(a) shows the training and validation accuracyand fig10(b) shows the trainingand validation loss forVGG16. Fig11 shows the pictorial representation for the true positive,true negative ,false positive and false negative values.Table5 compares the different characteristic measures for obstacle and no obstacle.

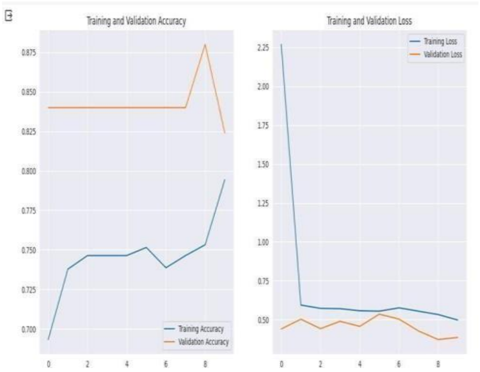


Fig .10(a)Accuracy (b)Loss for VGG16

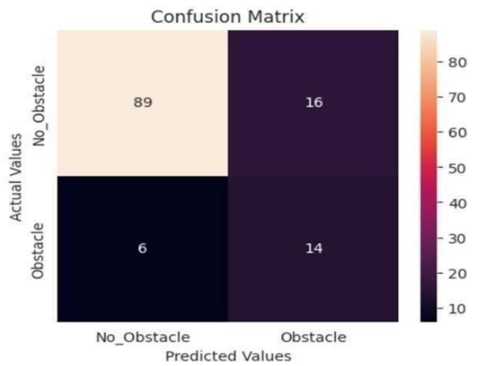
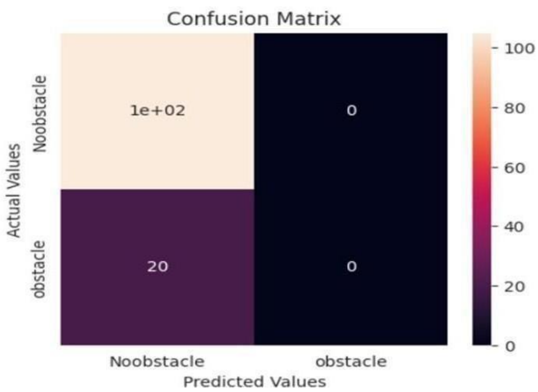


Fig. 11 Confusion Matrix for VGG16 model

420

Table 5. Value of metrics evaluated for VGG16 mode



420

The following table.6 compares the results obtained from all the machine learning models used in thisstudy.

Table .6: Comparison of all the models implemented:-

MODELS	LEARNING RATE ANDEPOCHS	OPTIMIZATIO	RESULT
MobileNetV2	Epochs=10 LR=0.001	Adam	Val acc=82%
ResNet	Epochs=10 LR=0.001	Adam	Val acc=84%
VGG16	Epochs=10 LR=0.001	Adam	Val acc=82%
Inception	Epochs=10 LR=0.001	Adam	Val acc=85%
Xception	Epochs=10 LR=0.001	Adam	Val acc=84%

It can be further validated on different obstacle detection datasets in the future. Its performance with otheroptimizers and pre-processing techniques such as CLAHE, Morphological gradient, Adaptive Histogram Equalization, Gaussian Blur, Weighted Gaussian Blur etc. can be studied.

4. Conclusion:

In this study, Inception model achieved the highest accuracy of 85% due to its ability to capture multi-scale features efficiently with minimum training and validation loss .Inception model proved to work well on several obstacle detection and avoidance cases by capturing information from parallel filters at various spatial scales and then returning the global average pooling at the end of the network. ResNet and Xception model achieved the second highest accuracy of 84% each. ResNet is one of the strong choice for computer vision tasks to ease the training of deep neural networks.This received a recall score of 1 for obstacle which indicates a perfect performance where every positive instance in the dataset has been correctly identified by the model with no false negatives or omissions.Whereas Xception used depthwise separable convolutions to capture complex pattern in the data,similar to deeperand more parameter- heavy networks might achieve. This used Xception as benchmark to have a computationally efficient model with fewer parameters that can also deliver us comparable results withhigher accuracy.

MobileNetV2 and VGG16 yields an accuracy of 82% which is slightly lesser than our other models taken under consideration for this study. Despite the downsides, can use MobileNetV2 for using the real-time detection and avoidance through web and mobile applications due to its light-weight architecture. And VGG16 for interpretability tasks and visualization of learned features. Though it is known for its simple and uniform architecture of 16 weight layers, it is computationally more expensiveas compared to other models. Thus the selection of model depends on the project requirements. For higher accuracy, go with Inception or ResNet. For the model to be computationally efficient and optimal model size, choose Xception that offers less parameters with a slight compromise in the accuracy.

However, for integration and deployment into mobile or web applications which will allow to show a real-time demonstration. For this opt MobileNet.For learning purposes, and a simple architecture and explainability of the learned features over the accuracy of the model, then opt for VGG16 model.

Declarations

Conflict of interest: The author has no **conflicts of interest** to declare.

REFERENCES

[1] M. S. M. Hashim, T. -F. Lu and H. H. Basri, "Dynamic obstacle avoidance approach for car- like robots indynamic environments," *2012 International Symposium on Computer Applications and Industrial Electronics (ISCAIE)*, Kota Kinabalu, Malaysia, 2012, pp. 130-135, Doi: 10.1109/ISCAIE.2012.6482083.

[2] Sushmita M. Rathod and S. K. Apte (2019) OBSTACLE DETECTION AND AVOIDANCE

[3] FOR AUTONOMOUS ELECTRIC VEHICLE USING ARDUINO. International Journal of Engineering Applied Sciences and Technology, 2019 Vol. 4.

[4] J. Liu, P. Jayakumar, J. L. Stein, and T. Ersal, "A study on model fidelity for model predictive control-basedobstacle avoidance in high-speed autonomous ground vehicles," Vehicle System Dynamics, 2016.http://dx.doi.org/10.1080/00423114.2 016.1223863

[5] Agnihotri, Akhil, Prathamesh Saraf and Kriti Rajesh Bapnad. "A Convolutional Neural Network Approach Towards Self-Driving Cars." *201D IEEE 1Sť" I→ıdia Cou→ıcil I→ıťeí→ıaťio→ıal Co→ıreíe→ıce (INKICON)* (2019): 1-4.

[6] Skoczeń M, Ochman M, Spyra K, Nikodem M, Krata D, Panek M, Pawłowski A. Obstacle Detection System forAgricultural Mobile Robot Application Using RGB-D Cameras. Sensors (Basel). 2021 Aug 5;21(16):5292. Doi: 10.3390/s21165292. PMID: 34450732; PMCID:PMC839991

[7] Pydala, B., Kumar, T. P., & Baseer, K. K. (2023). Smart Eye: A Navigation and Obstacle Detection for Visually Impaired People through Smart App . *Journal of Applied Engineering and Technological Science (JAETS)*, *4*(2), 992–1011. https://doi.org/10.37385/jaets.v4i2.2013

[8] Jiaying Yuan , Tao Jiang , Xi He , Sidong Wu, Jiajia Liu and Dequan Guo (2023) Dynamic obstacle detectionmethod based on U–V disparity and residual optical flow for autonomous driving. Nature Portfolio , scientific Reports.

[9] Pant, Kumar & Das, Amitabh & Agrewale, Mohammad & BHATESHVAR, YOGESH & Vora, Kamalkishore. (2021). Obstacle Detection Using Sensor Fusion and Deep Neural Network for Motion Control of Smart Electric Tractor. 1-6. 10.1109/ITEC-India53713.2021.9932516.

[10] A.K. Kyatsandra, R. K. Saket, S. Kumar, K. Sarita, A. S. S. Vardhan and A. S. S. Vardhan, "Development ofTRINETRA: A Sensor Based Vision Enhancement System for Obstacle Detection on Railway Tracks," in IEEESensors Journal, vol. 22, no. 4, pp. 3147-3156, 15 Feb.15, 2022, Doi: 10.1109/JSEN.2021.3140032.

[11] Ravpreet Kaur, Sarbjeet Singh, A comprehensive review of object detection with deep learning, Digital SignalProcessing, Volume132,2023,103812,ISSN 1051-

[12] 2004,https://doi.org/10.1016/j.dsp.