VISION-BASED ROAD SIGNAGE RECOGNITION FOR AUTONOMOUS VEHICLE IN AGRICULTURAL PLANTATION

Ayman Ahmed Hashem Salem Almashwali
Faculty of Technology and Electrical Engineering
Universiti Teknikal Malaysia Melaka (UTeM)
76100 Durian Tunggal, Melaka, Malaysia
Email: m112310026@student.utem.edu.my

Anuar Bin Mohamed Kassim
Center of Robotics and Industrial Automation
Faculty of Technology and Electrical Engineering
Universiti Teknikal Malaysia Melaka (UTeM)
76100 Durian Tunggal, Melaka, Malaysia
Email: anuar@utem.edu.my

Mohd Rusdy Bin Yaacob
Center of Robotics and Industrial Automation
Faculty of Technology and Electrical Engineering
Universiti Teknikal Malaysia Melaka (UTeM)
76100 Durian Tunggal, Melaka, Malaysia
Email: rusdy@utem.edu.my

Tan Kim Loong
Center of Robotics and Industrial Automation,
Faculty of Technology and Electrical Engineering,
Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya,
76100 Durian Tunggal, Melaka, Malaysia,
Email: m012210016@student.utem.edu.my

Awangku Khairul Ridzwan bin Awangku Jaya
IngeniousCity Engineering Solutions Sdn Bhd.,
81-1 Jalan Satu, Taman Satu Krubong,
Kampung Sungai Badau,
75260 Krubong, Malacca, Malaysia
Email: khairul@ices.my

Zuraida Ngadiron
Industry Centre of Excellence for Railway,
Institute Integrated Engineering,
Universiti Tun Hussein Onn Malaysia,
86400 Parit Raja, Batu Pahat, Johor, Malaysia
Email: zuraidahn@uthm.edu.my

Takashi Yasuno
Graduate School of Technology,
Industrial and Social Sciences,
Tokushima University, 2-1 Minami Jousanjima-cho
Tokushima 770-8506, Japan
Email: yasuno.takashi@tokushima-u.ac.jp

ABSTRACT

The growth of self-driving vehicles demands dependable vision-based traffic sign recognition systems to maintain safety and efficiency in agricultural plantations. This research aims to create an enhanced traffic sign identification system based on the YOLOv3 algorithm, which will solve the limitations of standard human vision-based approaches, notably in low-light circumstances and with occlusions. The system leverages computer vision and machine learning techniques, requiring extensive training on diverse datasets to ensure robustness against environmental variations and regional signage differences. Implemented on platforms like Google Colab, the system was trained and tested using a comprehensive dataset, achieving a mean average precision (mAP) of 96.96%, precision of 94%, and recall of 95%. Despite its high accuracy and effective real-time processing capabilities, challenges like handling similar signs and occlusions persist. Future work will concentrate on increasing the dataset, refining the model, enhancing occlusion management approaches, and allowing real-time processing on edge devices like the Jetson Nano and Raspberry Pi, boosting system dependability and developing autonomous driving technologies.

Keywords: Vision-Based, Road Sign Recognition, Precision Agriculture, Autonomous Vehicle.

INTRODUCTION

The growth of self-driving vehicles demands dependable vision-based traffic sign recognition systems to maintain safe and efficient highway operations (K. Muhammad, A. Ullah, et al. 2021). The autonomous vehicle could also be beneficial for agricultural purposes for reducing the human burden of monitoring the plantation at all times, including pest monitoring (Kassim, A.M., Said, et al. 2022). Vision-based road sign recognition systems have received a lot of interest because they can potentially improve the safety and reliability of autonomous driving (M. I. Pavel, S. Y. Tan, and A. Abdullah 2022). These systems, which use powerful computer vision and machine learning techniques, are critical for autonomous cars' accurate navigation and decision-making capabilities (Kassim, A.M., et. al. 2011). They must withstand environmental variables such as weather, illumination, and occlusions and adapt to local and regional signage standards (D. Tabernik and D. Skocaj 2020). Creating such systems requires training on massive, diverse datasets, which is time-consuming yet necessary for high-quality performance (Kassim, A.M., Yasuno, et. al 2015).

Traditional road sign identification methods relying on human vision are inadequate, especially under challenging conditions like nighttime or occlusions caused by environmental factors such as weather and physical obstructions (Bin Mohamed Kassim et al, 2021). These limitations necessitate an innovative and automated road sign recognition system for autonomous driving (Kassim, A.M., Termezai, et. al 2020). Current methods fail to accurately differentiate between various traffic signs and their meanings, posing safety risks due to visual similarities and obstructions (Kassim, A.M., Jaafar, et. al 2013). This project aims to develop a vision-based traffic sign recognition system using advanced computer vision techniques to identify, categorize, and interpret traffic signs reliably under diverse conditions, thereby enhancing traffic safety and supporting the advancement of autonomous driving technology (Kassim, A.M., et. al 2012).

Mashrukh Zayed, Al Amin, and Shohanur Rahman developed and implemented a real-time traffic sign detection and recognition system using the YOLOv3 algorithm to address challenges in traditional methods and improve road safety in Bangladesh. Huibing Zhan et al. (2020) study investigates the effectiveness of different components of MSA_YOLOv3, comparing its performance with other models and highlighting the computational complexity of certain aspects of the proposed algorithm. Some studies explore the implementation of traffic sign recognition using deep learning in autonomous vehicles, focusing on the YOLO method, dataset combination, training image implementation, and achieving robust and accurate real-time results (S. B. Wali et. al. 2019). There are also investigations on traffic sign recognition and distance estimation using deep learning models such as YOLOv3 and Disnet (G. S. R. Nath et. al 2021).

The study of control systems in autonomous vehicles focuses on designing and implementing algorithms and hardware that enable vehicles to operate without human intervention (C. Bila et al. 2017). This involves sensor fusion, where data from various sensors such as cameras, LIDAR, and radar are integrated to perceive the environment and decision-making algorithms that plan safe and efficient paths (Kassim, A.M., et. al 2021). The control system then translates these decisions into precise vehicle maneuvers using actuators for steering, acceleration, and braking (Kassim, A.B.M., Yasuno, et. al 2010). Advanced control techniques, such as Model Predictive Control (MPC) and Reinforcement Learning (RL), are often employed to handle the complexities and uncertainties of real-world driving scenarios (Hasim, N., et. al 2012). Research in this field aims to improve autonomous systems' reliability, safety, and efficiency, ultimately leading to the widespread adoption of self-driving technologies (Mohamed Kassim A. et al. 2016) along with battery performances (Azam, M.A. et al. 2021).

The Velibor Ilic focuses on developing classification and road sign detection algorithms for autonomous cars, the development of combining CNNs with YOLOv3 for real-time usage in ADAS development, traffic sign recognition with other sensor readings, and the academic nature of the research, leaving room for future advancements (B. Novak, V. Ilić, et. al 2020). Another study on traffic sign recognition with YOLOv3 focuses on building a solution, testing its performance on a specific collection of traffic signs, analyzing its real-time applicability, and evaluating its performance under various weather circumstances (E. Yurtsever, et. al. 2020). The project focuses on designing, developing, and testing a system for recognizing and categorizing traffic signs, emphasizing accuracy and reaction speed.

MATERIALS AND METHODS

Figure 1 shows a flowchart of training a YOLOv3 model for object detection. The process starts with collecting a dataset of images. Then, the images are labeled using Make Sense, a data labeling platform. Next, the images are divided into training and test datasets. The training dataset is used to train the YOLOv3 model, and the test dataset is used to evaluate the performance of the trained model. The YOLOv3 model is trained using the Google Colab platform. Once the model is trained, its accuracy is evaluated. The model is retrained with different parameters if the accuracy is not accepted. The process continues until the desired accuracy is achieved.

Figure 2 illustrates the system's item detection process. Initially, a camera or video clip captures the photograph. These images are then uploaded to MAKE SENSE, where they are annotated with bounding boxes around the objects. The annotated images are then used to train the YOLO model using the YOLO training script with Python and OpenCV. The trained model can detect real-time objects in any computing platform, such as Jetson Nano and Raspberry Pi. The dataset, comprising a collection of related data managed by a computer, is crucial for training the Convolutional Neural Network (CNN) on road sign properties (Aras, M.S.M., et. al 2013). Images used to train the algorithm are extracted from a video featuring various rotations and angles and divided into 80% training and 20% testing images. Each image undergoes pre-processing to adjust its size and file type.

Building an AI or deep learning model that functions like a human requires extensive training data. Accurate annotation of data is a critical step in training the model. Annotation involves marking data available in various forms, such as text, video, or images. "Make Sense," an online tool, allows users to name and annotate images based on desired classifications manually. The platform's UI and labeling process are illustrated in Figure 3.4. After labeling, users can export the dataset in the appropriate format. For this study, the dataset was exported in the YOLO format, with each image accompanied by a text file (.txt) containing labeling information and coordinates. The datasets were further processed using the RoboFlow web tool, which matches annotations with images and divides the dataset into training, testing, and validation subsets.

Figure 1. System flowchart

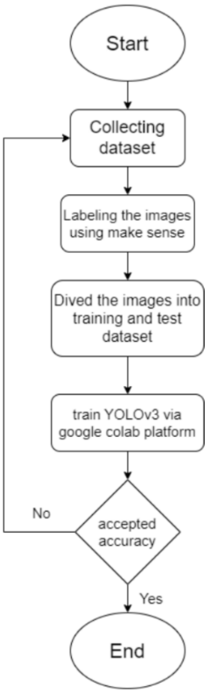
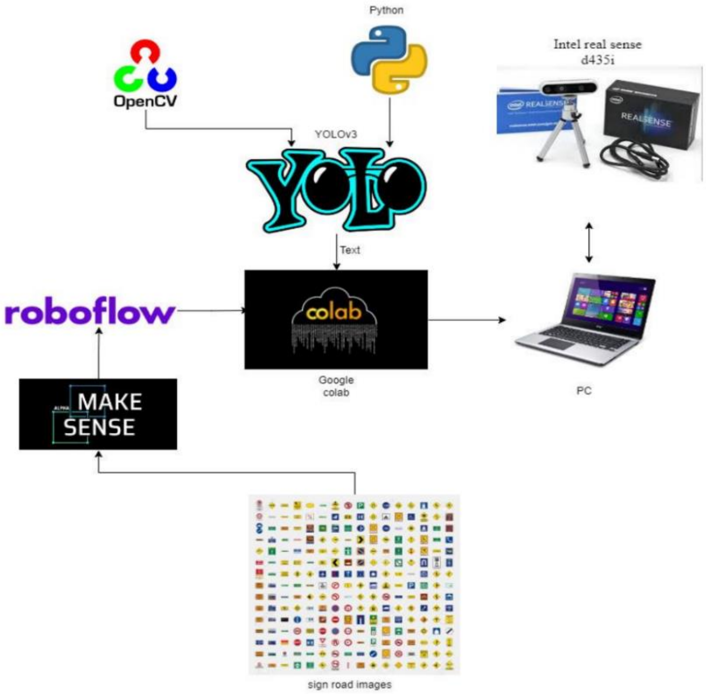


Figure 2. Components used



YOLOv3 represents an improvement over its predecessors, YOLOv1 and YOLOv2. Unlike exclusive labeling, YOLOv3 uses multi-label classification with a logistic classifier to determine the probability of an object fitting a label. It employs binary cross-entropy loss for classification instead of the mean square error used in previous versions. YOLOv3 makes multiple bounding box predictions and assigns an object value of 1 to the bounding box anchor that most overlaps with a ground truth object, ignoring other anchors with sufficient overlap. Utilizing the concept of feature pyramid networks, YOLOv3 predicts boxes at three scales, extracting features at each scale. The Darknet-53, a 53-layer CNN with skip connections inspired by ResNet, is the feature extractor. Darknet-53 starts with an ImageNet-trained network and adds 53 layers for detection, resulting in a 106-layer convolutional architecture. Convolution layers capture information from the input image, maintaining pixel relationships through mathematical operations. Pooling layers integrate feature points and resample feature maps to create new features. Before producing the classification output, the fully connected layer flattens the matrix into a vector.

The training procedure combines labels and images to teach the YOLOv3 module to detect and recognize traffic sign features. Google Colab facilitated the training process, utilizing Python. Out of 552 images, 414 were used for training, 83 for validation, and 55 for testing. The dataset and annotation files were uploaded to Google Drive and linked to Google Colab. Darknet was cloned to Google Drive using the command, (git clone https://github.com/AlexeyAB/darknet.git), and the YOLOv3 configuration file was adjusted to meet the desired specifications, setting the maximum number of patches to 6000 and steps to 4800 and 5400, representing 80% and 95% of the total patches, respectively. YOLOv3's core architecture is based on a 53-layer Darknet network, expanded to 106 layers for detection, and operates in Google Colab after downloading from GitHub and connecting to the cloud.

OPTIMIZATION OF RIS PLACEMENT AND CLUSTERING

This study will use the Intersection over Union (IoU) concept to identify objects. IoU computes the intersection of two bounding boxes: the red box represents the ground truth, while the green box represents the anticipated bounding box shown in Figure 3. To assess the validity of object detection, an IoU threshold value may be defined by confirming that the predicted and real bounding boxes are consistent. The picture of a car, with the predicted bounding box in red and the ground-truth bounding box in green shown in Figure 4.

Figure 3. IoU concept.

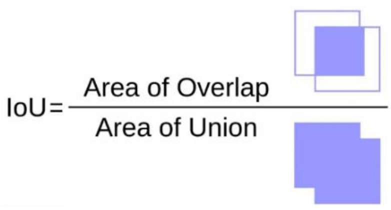
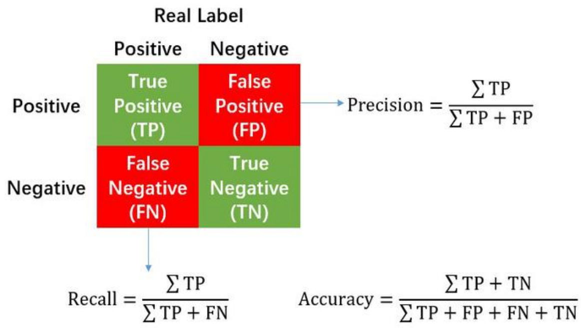


Figure 4. Predicted Label.



To evaluate performance, precision and recall metrics will be utilized, involving true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN). The confidence score for each object detected by the model must be considered. Bounding boxes with a confidence score exceeding a certain threshold are regarded as positive, while those below the threshold are considered negative.¬

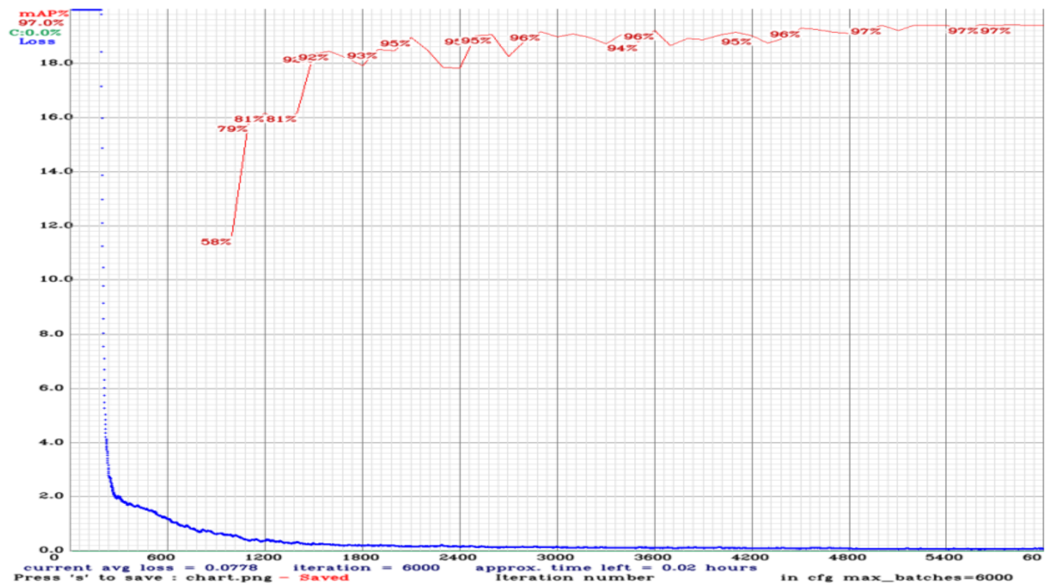
The YOLOv3 testing technique involves a systematic seven-step process. First, the image is uploaded and scaled to the correct dimensions. It is then divided into three grids of 13x13, 26x26, and 52x52 scales, where each grid unit predicts the object when its center point falls within it. K-means clustering is applied to each grid unit to examine bounding box priors, resulting in three clusters per grid unit, totaling nine clusters. A 13x13 small-scale feature map is constructed by feeding the image into the network for feature extraction. This feature map undergoes convolutional set and sampling twice, then combines with the 26x26 feature map to produce the prediction map response. Similarly, the 26x26 feature map is processed and combined with the 52x52 feature map. Finally, the three-scale predictive performance functions are combined, and a probability score threshold is used to filter out most low-scale anchors. YOLOv3 then employs Non-Maximum Suppression (NMS) for post-processing, resulting in more accurate bounding boxes and enabling real-time detection facilitated by the GPU's efficient computing capacity.

RESULT AND DISCUSSION

The trained module was used to test a total of 600 photos (200 images for each class), and the average confidence rate of the results is remarkably accurate, with the Berhenti sign having a little higher average confidence rate than the other signs. However, the module has shown itself to be both extremely accurate and trustworthy. Since the study didn't explicitly state the average confidence rate for each class, there are no findings for the confidence rate of the research algorithm from the previous year.

The graphic in Figure 5 depicts the training results for a YOLOv3 model on a traffic sign detection challenge. The red line represents the model's mean average accuracy (mAP), which measures how effectively it can recognize traffic signs. The blue line represents the model's loss, which is a measure of how effectively the model can learn the job. The mAP is initially low, but it increases as the model trains. This indicates that the model is learning to detect traffic signs more accurately. The loss is initially high, but it decreases as the model trains. This indicates that the model is learning to make better predictions. The model's performance reaches a peak after about 6000 iterations. This indicates that the model has learned to detect traffic signs very well.

Figure 5. Output chart



High-performance deep learning programs frequently make use of high-level graphics processing units (GPU). Computers have grown into hardware that can work nearly endlessly as technology has advanced. Small, portable, low-cost, highp erformance computers with relatively high features are preferred over huge, expensive computers. In these cases, single-board computers are ideal. Single-board computers are made up of a single circuit board including input/output, memory, microprocessors, and other essential parts. FPS (Frames Per Second) is the measure of the number of full-screen pictures that are displayed consecutively in a second.

As shown in Figure 6 the outcomes are obtained by using (!./darknet detector map cfg/traffic_light.data cfg/yolov3x-traffic.cfg backup/yolov3x traffic_best.weights -map -dont_show 2>&1 > map_accuracy.txt) this code will produce the output of the evaluation of a deep-learning model for object detection. The model was trained on 545 images and was evaluated on 55 images.

Figure 6. Coding in YOLOv3

```
 seen 64, trained: 360 K-images (5 Kilo-batches_64)

 calculation mAP (mean average precision)...
 Detection layer: 82 - type = 28
 Detection layer: 94 - type = 28
 Detection layer: 106 - type = 28

 detections_count = 140, unique_truth_count = 108
 rank = 0 of ranks = 140
 rank = 100 of ranks = 140
class_id = 0, name = 1-SPEED BUMB, ap = 95.34%          (TP = 28, FP = 2)
class_id = 1, name = 2-CROSSWALK, ap = 95.74%           (TP = 27, FP = 4)
class_id = 2, name = 3-BERHENTI, ap = 99.81%     (TP = 48, FP = 0)

 for conf_thresh = 0.25, precision = 0.94, recall = 0.95, F1-score = 0.95
 for conf_thresh = 0.25, TP = 103, FP = 6, FN = 5, average IoU = 79.76 %

 IoU threshold = 50 %, used Area-Under-Curve for each unique Recall
 mean average precision (mAP@0.50) = 0.969625, or 96.96 %
```

Figure 7 indicates that the model has a mean average accuracy (mAP) of 96.96%. The mAP measures the model's overall performance and takes into consideration both accuracy and recall. Precision is defined as the proportion of accurate detections, whereas recall is the fraction of ground truth objects discovered. The model attained a 94% accuracy and a 95% recall rate. The F1 score is a weighted average of accuracy and recall, and it measures the model's overall performance. The model obtained an F1 score of 95%.

Figure 7 Achieved model

	True Positive	False Positive	Precision %
Speed bump	28	2	95.34%
Crosswalk	27	4	95.74%
Berhenti	48	0	100%
Avrage Precision			97%

Total Precision %	Recall %	F1 score %
94	95	95
IoU %	Map %	Iteration
79.76	97	6000

Figure 8 Image recognition results

CONCLUSION AND FUTURE WORKS

The use of the YOLOv3 algorithm to create a vision-based traffic sign recognition system improves the safety and reliability of autonomous driving by reaching a mean average precision (mAP) of 96.96%, precision of 94%, and recall of 95%. While the system demonstrates high accuracy and effective real-time processing, challenges like handling similar signs and occlusions remain. Future work should focus on expanding and diversifying the dataset, optimizing the model with advanced algorithms, improving techniques for handling occlusions and similar signs, and implementing real-time processing on edge devices like Jetson Nano and Raspberry Pi.

ACKNOWLEDGEMENT

This research is financially supported by Ministry of Higher Education Malaysia through PRGS 2024-1 (PRGS/1/2024/FTKE/T00031). Deeply acknowledgement goes to Universiti Teknikal Malaysia Melaka, UTeM for providing equipment and facilities.

REFERENCES

Aras, M.S.M., et. al (2013), ‘Tuning process of single input fuzzy logic controller based on linear control surface approximation method for depth control of underwater remotely operated vehicle’, Journal of Engineering and Applied Sciences, 8(6), pp. 208–214

Azam, M.A., et. al (2021), ‘Structural characterization and electrochemical performance of nitrogen doped graphene supercapacitor electrode fabricated by hydrothermal method’, International Journal of Nanoelectronics and Materials, 14(2), pp. 127 136

B. Novak, V. Ilić, et. al (2020), ‘YOLOv3 Algorithm with additional convolutional neural network trained for traffic sign recognition,’ 2020 Zooming Innovation in Consumer Technologies Conference, ZINC 2020, pp. 165–168.

Bin Mohamed Kassim, A., et. al (2021), ‘Performance analysis of wireless warning device for upper body level of deaf-blind person’, 2015 54th Annual Conference of the Society of Instrument and Control Engineers of Japan, SICE 2015, pp. 252–257

C. Bila, et. al (2017), ‘Vehicles of the Future: A Survey of Research on Safety Issues’, IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 5, pp. 1046–1065

D. Tabernik and D. Skocaj (2020), ‘Deep Learning for Large-Scale Traffic-Sign Detection and Recognition’, IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 4, pp. 1427–1440

E. Yurtsever, et. al. (2020), ‘A Survey of Autonomous Driving: Common Practices and Emerging Technologies’, IEEE Access, vol. 8, pp. 58443–58469

G. S. R. Nath, et. al (2021), ‘Traffic Sign Recognition and Distance Estimation with YOLOv3 model,’ 2021 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT 2021, pp. 154–159.

Hasim, N., et. al (2012), ‘Development of fuzzy logic water bath temperature controller using MATLAB’, Proceedings – 2012 IEEE International Conference on Control System, Computing and Engineering, ICCSCE 2012, pp. 11–16

H. Zhang et al. (2020), ‘Real-Time Detection Method for Small Traffic Signs Based on Yolov3’, IEEE Access, vol. 8, pp. 64145 64156

Kassim, A.M., et. al. (2011), ‘Design and development of MY 2nd EYE for a visually impaired person’, 2011 IEEE Symposium on Industrial Electronics and Applications, ISIEA 2011, pp. 700–703

Kassim, A.M., Termezai, et. al (2020), ‘Design and development of an autonomous pesticide sprayer robot for fertigation farm’, International Journal of Advanced Computer Science and Applications, (2), pp. 545–551

Kassim, A.M., et. al (2021), ‘Potential of on-the-go gamma-ray spectrometry for estimation and management of soil potassium site specifically’, Sustainability, 13(2), pp. 1–17, 661

Kassim, A.B.M., Yasuno, et. al (2010), ‘Moving control of quadruped hopping robot using adaptive CPG networks’, 2010 IEEE Conference on Robotics, Automation and Mechatronics, RAM 2010, pp. 581–588

Kassim, A.M., Jaafar, et. al (2013), 'Performances study of distance measurement sensor with different object materials and properties', Proceedings - 2013 IEEE 3rd International Conference on System Engineering and Technology, ICSET 2013, pp. 281–284

Kassim, A.M., Jamri, et. al (2012), 'Design and development of obstacle detection and warning device for above abdomen level, 2012 International Conference on Control, Automation and Systems, pp. 410–413

Kassim, A.M., et. al (2012), 'Design and development of vibration method for Vehicle Reverse System (VRS)', Procedia Engineering, 41, pp. 1114–1120

Kassim, A.M., Yasuno, et. al (2015), 'Vision based of tactile paving detection method in navigation system for blind person', Jurnal Teknologi, vol. 77, no. 20, pp. 25-32, 2015

Kassim, A.M., Said, et. al (2022), 'Performance Analysis of Spraying Coverage Rate by Using Mobile Robot for Pesticide Spraying Application on Chili Fertigation Farm, Lecture Notes in Electrical Engineering, vol 842, pp. 639-650

K. Muhammad, A. Ullah, et al. (2021), 'Deep Learning for Safe Autonomous Driving: Current Challenges and Future Direction' IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 7, pp. 4316–4336

Mohamed Kassim, A., et al. (2016), 'Conceptual design and implementation of electronic spectacle based obstacle detection for visually impaired persons', Journal of Advanced Mechanical Design, Systems and Manufacturing, 10(7)

M. I. Pavel, S. Y. Tan, and A. Abdullah (2022), 'Vision-Based Autonomous Vehicle Systems Based on Deep Learning: A Systematic Literature Review', Applied Sciences, vol. 12, no. 14.

S. B. Wali et. al. (2019), 'Vision-based traffic sign detection and recognition systems: Current trends and challenges', Sensors, vol. 19, no. 9.