Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 9, No. 5, October, 2025

Ball Detection System for a Soccer on Wheeled Robot Using the MobileNetV2 SSD Method

Riky D. Puriyanto [1], Isro D. Yunandha [1], Hari Maghfiroh [2, 3*], Alfian Ma'arif [1], Furizal [4], Iswanto Suwarno [5]

[1] *Department of Electrical Engineering, Universitas Ahmad Dahlan, Yogyakarta 55191, Indonesia.*

[2] *Department of Electrical Engineering, Universitas Sebelas Maret, Surakarta 57126, Indonesia.*

[3] *Department of Electrical and Information Engineering, Universitas Gadjah Mada, Yogyakarta 55281, Indonesia.*

[4] *Department of Research and Development, Peneliti Teknologi Teknik Indonesia, Sleman 55281, Indonesia.*

[5] *Department of Electrical Engineering, Universitas Muhammadiyah Yogyakarta, Yogyakarta, 55183, Indonesia.*

Abstract

This paper discusses the research on the use of Artificial Intelligence in autonomous robot object identification. The specific focus of this research is on a wheeled soccer playing robot. The goal is to recognize a ball as an object using the Single Shot MultiBox Detector MobileNetV2 model. This system has multi-vision inputs such as distance measurements and angle values for object detection. This methodology is based on deep learning with the TensorFlow Object Detection API with the MobileNetV2 SSD model. This model is trained with a dataset of 3707 ball images over 617 thousand steps on Google Collaboratory. It was found that the average measurement error of the ball object is 6.58% for the distance when viewed through the robot's front camera. In addition, the omnidirectional camera is able to detect the ball object and angle values from the front of the robot. What makes this research different is the use of distance and angle measurements for detection and the omnidirectional camera for system performance in dynamic environments. This research aims to address the improvement of AI-based object detection systems for autonomous robotics in the context of real-world use cases.

Keywords:

Artificial Intelligence;
SSD MobileNetV2;
Soccer Robot; TensorFlow;
Distance Measurement.

Article History:

Received:	23	December	2024
Revised:	21	July	2025
Accepted:	28	July	2025
Published:	01	October	2025

1- Introduction

The development of technology is currently rapid, which is shown by the existence of technology developed by humans aimed at carrying out various kinds of work and making it easier to complete all activities [1-3]. Currently, technological developments, especially in the field of robotics [4, 5], are growing very quickly, as seen from the many applications that are based on the fields of computer vision and artificial intelligence, which are applied in the world of education, industry, art, and everyday life [6-9]. With the increasing use of technological devices, it can be seen that the digitalization of technology has experienced significant development [10]. Digitalization is a digital transformation process that involves the use of computerized technology to simplify various human tasks [11]. One of the computerization technologies that is currently being widely discussed is artificial intelligence [12, 13]. This technology is the ability possessed by computer systems to interact with the world like humans [14]. The term Artificial Intelligence

* **CONTACT**: hari.maghfiroh@staff.uns.ac.id

DOI: http://dx.doi.org/10.28991/ESJ-2025-09-05-028

(AI) has been discussed extensively in various disciplines for the development of advanced technologies to solve real-life problems [15, 16]. However, AI systems have a variety of models and methodologies that can be used in different ways depending on the situation [17, 18]. Having knowledge of these methods and their uses plays a vital role in improving the efficiency and effectiveness of AI applications in real-world settings [19-21].

The application of technology in the field of robotics, which is currently being developed by universities and hobby enthusiasts who are closely related to the field of computer vision, to competitions, namely the Indonesian Robot Contest [22-24], one of which is the Wheeled Indonesian Football Robot Contest [25]. One of the design and engineering competitions in the field of robotics at regional and national levels, which is held annually by the Ministry of Research, Technology, and Higher Education, Directorate General of Learning and Student Affairs [26, 27]. The wheeled soccer robot must be able to play the ball on the field like a soccer player [28]. Robots need an object detection system that has a high level of accuracy so that the robot can identify objects correctly and carry out its tasks like a football player without errors [29].

Developments in this detection are in the traditional era and the era of using deep learning. In the traditional era, object detection was carried out manually with human involvement in providing input to the system regarding the objects to be detected [30, 31], whereas in this era, the use of deep learning is part of the machine learning method, which allows system algorithms to learn and develop independently through data that has been designed and experience gained, without requiring significant human assistance [32, 33]. Wheeled soccer robots have tasks, for example, being able to find the position of the ball, goal, and avoiding obstacles [34]. To complete this task by using a camera as a sensor, the robot is able to recognize ball objects to help the wheeled soccer robot. Object detection, or what is often called object detection, aims to detect and identify certain objects [35, 36]. With this technology, robots can see using a form of vision similar to the human eye [37]. This makes real-time identification of balls, objects, and even goalposts much more realistic. In detecting objects in the field, a computer vision approach is used so that the robot can gain an understanding of every object it successfully captures [38, 39]. As well as obtaining the measured distance value and the angular value of the ball's direction in order to know where the object is located on the field.

Detecting the ball in a soccer robot has special difficulties that conventional object recognition systems cannot fully detect [40]. The ball is spherical and has no angle, making it even more difficult to identify accurately. In addition, the ball moves quickly and, combined with the ever-changing field settings, makes real-time detection, tracking, and position angle estimation very challenging, especially when the ball is in motion. Basic models tend to be slow to handle drastic changes and shifts in the location and position of the ball. Several methods used in detecting spherical objects have been carried out by previous researchers, namely the use of Local Binary Pattern (LBP) [41], YOLO V3 [42], YOLO V3 tiny [43], and Neural Network [44], where this method still requires capable hardware to obtain the desired performance with a computing level that adapts to the high specifications of the device.

Barry et al. [45] utilized YOLOv3 for object detection in a soccer robot and achieved a mAP of 87.07%. mAP stands for mean Average Precision which is a metric used to measure the performance of an object detection model [46, 47]. However, their configuration used a Shuttle Xl Mini PC with an Intel Core i7 and NVIDIA GeForce GTX 1060 GPU, which is high-end equipment, and may not be practical for low-cost devices in real-world scenarios. Similarly, Soebhakti et al. [48] introduced the XYOLO model, which outperformed Tiny-YOLO in detection speed by almost 70 times. Unfortunately, this improvement came at the expense of accuracy, as XYOLO produced FPS values that were insufficient to meet the real-time needs of soccer. Additionally, Irfan & Widodo [49] were able to detect soccer balls in real-time using a CNN-based architecture, but the system could only deliver a maximum accuracy of 67% and 13 FPS, limiting its effectiveness in faster environments. Sanubari & Puriyanto [43] used YOLOv3 and YOLOv3-Tiny for ball and goal detection, which performed quite well with a mAP of 87.5% for YOLOv3, but their approach struggled with multiple object speed and accuracy.

In this study, spherical object detection is performed using the MobileNetV2 Single Shot MultiBox Detector technique [50], which has the advantage over previously explored approaches of being more suitable for lower specification hardware. Unlike methods using YOLO [51] and R-CNN [52] (derivative of CNN [53]), which usually require high-end GPUs and large computing resources to obtain desired results [54], the SSD-MobileNetV2 technique is more efficient in terms of both speed and accuracy [55]. The choice of SSD MobileNetV2 in this research is based on its ability to operate on low-end machines while maintaining high detection accuracy, making it ideal for practical implementation in economical robotic systems [56].

In this research, the TensorFlow framework is used [57], which is a popular deep learning library written in the Python programming language and with a high-performance framework [58]. The aim of this research is to create an object detection system for a wheeled soccer robot so that the robot can detect objects accurately and can determine the position of the object and the angle of the object. This method provides results, it can identify a detected object such as a ball and there is the actual distance of the object and the angle value to the front of the robot which is the position of the ball object. In addition, this study also aims to analyze the performance of the SSD-MobileNetV2 model and compare it with the use of YOLO used in previous studies with an emphasis on its detection accuracy and frames.

2- Research Methodology

Research into the detection system for this wheeled soccer robot is in stages according to the plot. A general overview of the research stages that will be carried out in this research is shown in Figure 1. In the initial design stage, identify the problem, then carry out a literature study, which is used as a reference source for detecting ball objects on wheeled soccer robots. Then proceed with planning which analyses are needed, such as planning the hardware and software that will be used.

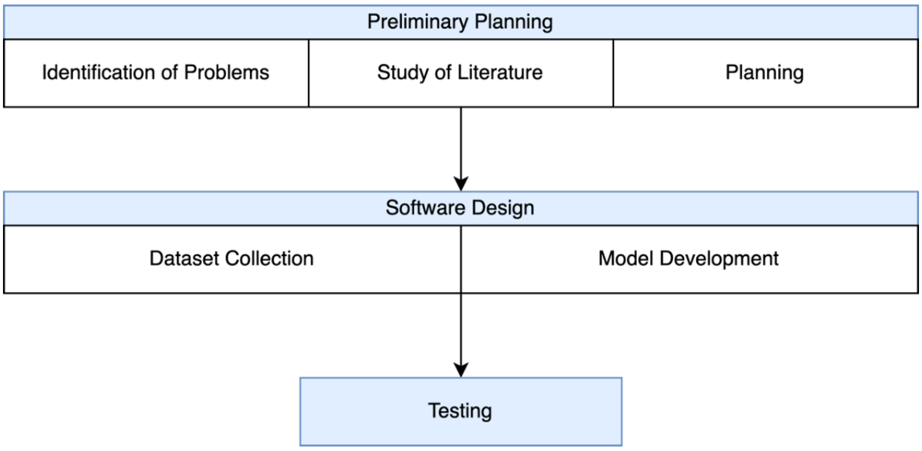


Figure 1. Research method flow

At the software design stage, dataset collection is carried out; the data used in the research focuses on spherical objects. Next is the development of an object detection model, which includes data processing, architecture development, and data training. The testing will be carried out in real time by manually testing the measurement results and calculations obtained from the detection results of the multi-vision application. The results of the test are the distance of the object detected on the robot's front camera, the object class, and the object angle value on the omnidirectional camera.

2-1- Single Shot MultiBox Detector

Single Shot MultiBox Detector, or commonly known as SSD, is a well-known computer vision algorithm for fairly fast target detection and is used in real time [59]. This identification can be described in terms of a bounding box, where the bounding box predicts the class specified/created according to the object [60]. SSDs are also part of the development of deep learning, or methods such as object recognition [61]. The SSD method is based on a feedforward convolutional network that produces a finite set of sizes and values indicating the presence of a detected class [62]. The SSD model is the basis of artificial intelligence for soccer robots. The advantage of this model is that it is fast and has high target detection accuracy. The depth and layers of the SSD-MobileNet model can be seen in Figure 2.

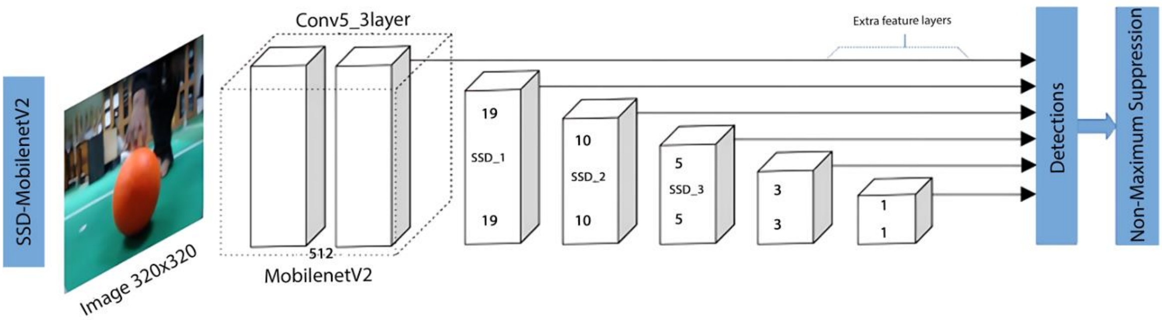


Figure 2. Single Shot MultiBox Detector MobileNetV2 architecture

2-2- MobileNet

MobileNet is a Convolutional Neural Network (CNN) architecture that is used to overcome the excessive computing research needs of mobile platforms [63]. MobileNet itself is a small-scale, low-latency, low-power design that meets the resource constraints of a variety of use cases [64]. Stating that MobileNet divides convolution into depth wise convolution and pointwise convolution, MobileNet itself released its newest version, namely version 2, which is different from the previous version by adding two new features, namely: 1) line bottlenecks and 2) shortening connections between bottlenecks [65]. Bottlenecks contain inputs and outputs between models, while inner layers encapsulate the model's ability to transform inputs from low-level concepts to higher-level descriptors. Finally, like traditional residual connections in CNNs, shortening between bottlenecks allows for faster training with greater accuracy [66].

2-3- Software Planning

This stage is the preprocess stage of the multi-vision application object detection system, which will be implemented on a wheeled soccer robot, including training data collection, data processing, model development, and data training, as shown in Figure 3.

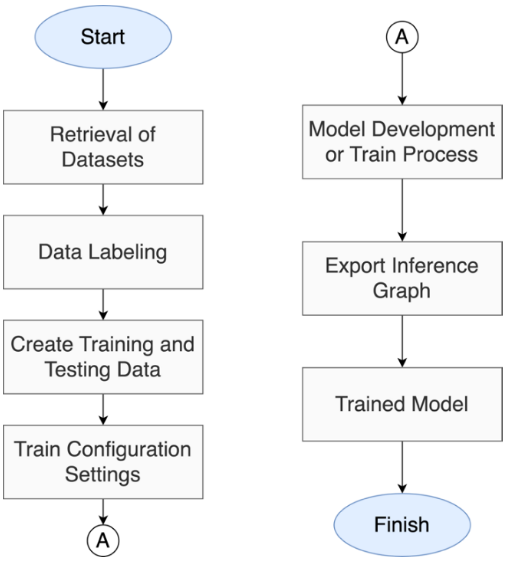


Figure 3. Software planning flowchart

2-4- Description of Software Design

2-4-1- Dataset Retrieval

In this step, the data, tools, and materials needed for the research are collected. The data is recorded using an omnidirectional camera and a Logitech C922 webcam camera by converting the video into images. The data is recorded data of a ball object with a 640480 frame. The input data in this research is video in .mp4 and .mkv format, which is needed for the training data in the form of images; therefore, it needs to be converted into image form so that it can be processed using a video conversion program into per-frame images and resized to 320×320 results. These changes are shown in Figure 4.

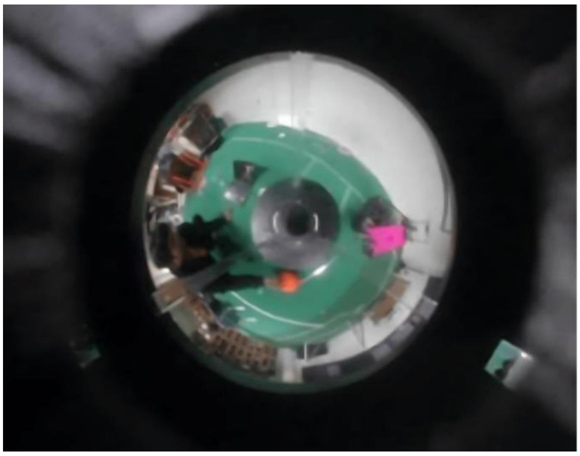


Figure 4. Video to image conversion results and resize

2-4-2- Labelling Data

Labelling objects (images) is the process of creating labels on images by assigning bounding boxes along with class names to objects [67]. The object labelling process is carried out with image data using the software shown in Figure 5. This labelling process provides bounding boxes for the data that has previously been collected [68]. When the labelling is complete, there will be an image file and an .xml file where the data has been labelled and will be processed to the next stage. This dataset will be divided into training data and test data for ball objects that have been taken using the front camera on the robot and an omnidirectional camera. The distribution of the data is shown in Table 1.

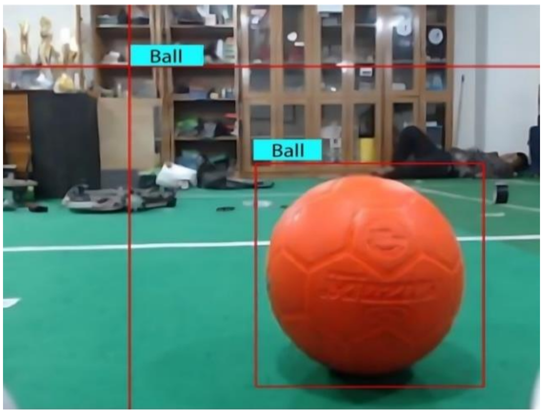


Figure 5. Labeling images

Table 1. Training and testing dataset division

Data Sharing	Frequency
Training data	2966
Test data	741
Total	3707

In Table 1, the training data and test data are divided into 80% training data and 20% test data. The amount of ball image data being in the training data is divided into whole or far ball images and half ball images taken using the front camera and ball images from the omnidirectional camera. The front camera images are where the half-ball images are 775, while the far or whole-ball images are 861, and for the omnidirectional camera ball images, they are 2071.

2-4-3- Data Training

Training data, or "train data," includes the input TFRecord and config files used, and all the executions we run will be saved on the Google Drive used. In the config configuration, repetition is also specified for training data and entering the model before it is trained and entering the program file for the training process. The training process is carried out using GPU on Google Collaboratory [69]. The training results have been completed and then exported so that the training data can be tested. In this research, we used a pre-trained SSD MobileNetV2 model using commands in the program that we previously created on Google Collaboratory [70]. The data training process can be seen in Figure 6. Next, after the data training process is complete, we export it using the program. In export, good parameters have been set, and a model can be created. The model will be exported and saved as a file in the form of a protocol buffer (.pb) and label map (.pbtxt), which is used in the ball object detection system on a wheeled soccer robot.

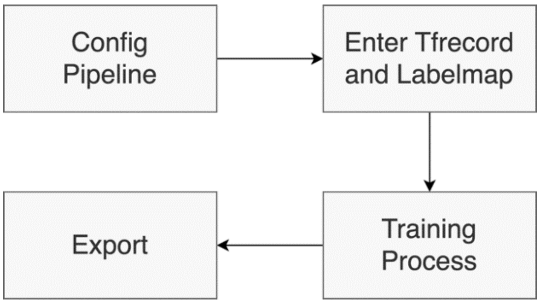


Figure 6. Data training process

2-5- System Testing

The testing process was carried out in order to find out the results of the detection system on a real wheeled soccer robot. This testing process used a combination of two omnidirectional cameras and a Logitech C922 webcam camera as input, as shown in Figure 7. In the block diagram in Figure 7, there is training data, which is the data resulting from our training. Then we enter the pre-process stage, where at this stage we continue our training model, which will be tested with the SSD MobileNetV2 model. Then, after the model has been successfully loaded using the trained SSD-MobileNetV2 method, it will be carried out. spherical object detection testing. The detection results are in the form of measured distance values from the detected object. Input from the webcam camera and the angle value when the object is detected on the omnidirectional camera. The method used in this research is the Single Shot MultiBox Detector MobileNetV2 algorithm [71].

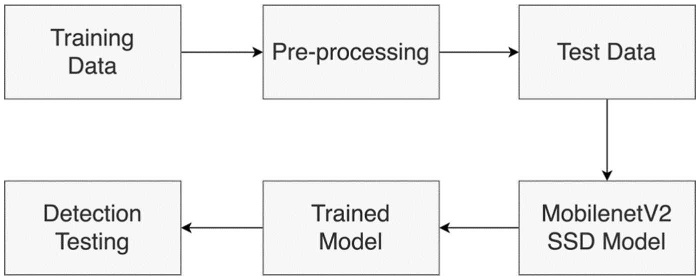


Figure 7. Detection system block diagram

In this case, the researchers made a ball object detectable using this method, and produced the actual distance value detected on the front camera of the wheeled soccer robot, the expected detection results were also the angle value of the object detected by the omnidirectional camera. In the system flow diagram in Figure 8, the image process from multi-vision input is captured directly where the multi-vision input is an omnidirectional camera and a front camera on a wheeled soccer robot. Then the image processing process will continue, namely the process where there is a MobileNetV2 SSD model and a trained model. The combination of SSD and MobileNet will simplify the process of creating a detection system for wheeled soccer robots. In the detection system, SSD is needed to determine the target location, while MobileNet is used to classify targets, namely spherical objects. Then is the ball object detected in multi-vision, namely the omnidirectional camera and the front camera on the wheeled soccer robot.

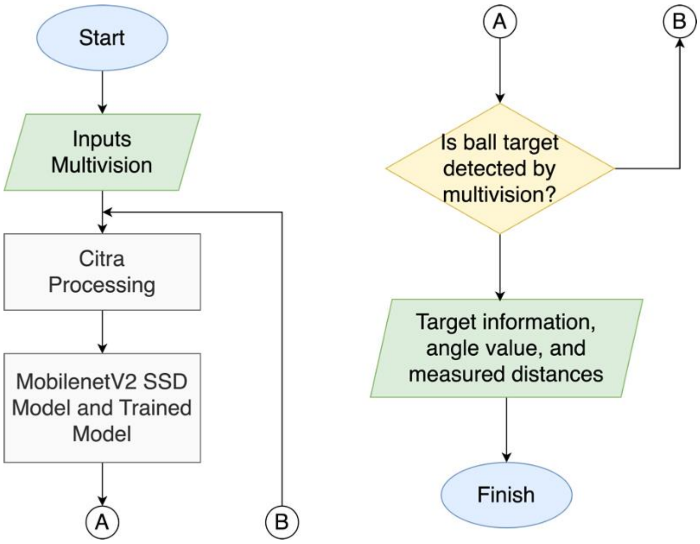


Figure 8. Detection system flowchart

Next, the target information, angle value and measured distance where the object is a ball, if the ball is detected on the omnidirectional camera, then the angle value and target information, namely the ball, will be displayed, but if on the front camera the ball object is detected then a distance calibration process/calculation process will occur. distance and will display the actual distance and pixel distance information.

2-6- Distance Determination

The target image is used for the process of determining the coordinates of the target position according to calculating the parts in the image. Then the target identification process uses the Single Shot MultiBox Detector MobileNetV2 algorithm method in this research. An illustration of the distance calculation on the front camera of the soccer robot is shown in Figure 9.

The position of the ball relative to the robot by the image is expressed in polar coordinates derived from distance and angle. The original point of coordinates is located at the center point of the image or target, which is the center point of the robot. The coordinates of the spherical object in the image can be obtained using Equations 1 and 2 to obtain the angle value of the detected target. t_x is the center point of the image frame on the x-axis, t_y is the center point of the image frame on the y-axis, c_x is the center point of the object's coordination on the x-axis, c_y is the center point of the object's coordination on the y-axis, h is the height of the frame, and J_{px} is the Pixel Distance.

$$J_{px} = \sqrt{(t_x - c_x)^2 + (t_y - c_y)^2} \quad (1)$$

$$\theta = \tan^{-1}\frac{(h - c_y)}{(t_x - c_x)} \quad (2)$$

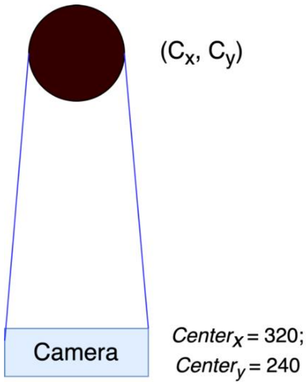


Figure 9. Illustration of calculating distance in pixel units

3- Results and Discussion

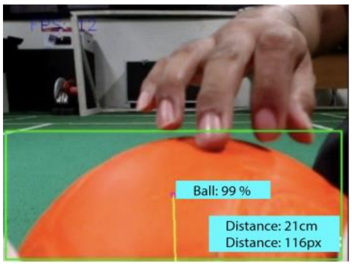
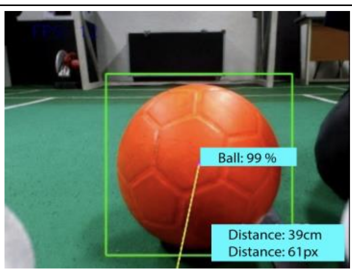
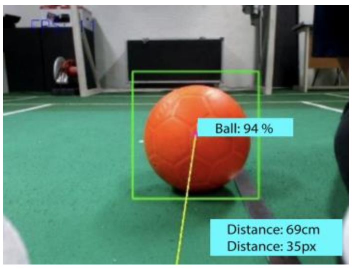
The results of this research are a spherical object detection system as well as the measured distance values and angular values of the spherical object when detected by the system with the application of multi-vision. In this research, several tests were carried out to obtain results and analysis of the system that has been created, including testing the detection of spherical objects in multi-vision, measuring the actual distance with the calculated distance and spherical objects detected by the system, and reading the angle value of the object when detected on an omnidirectional camera and testing on the front of a soccer robot.

3-1- System Testing

3-1-1- Results of Measured Distance Values in Pixels and Actual Distance

The main focus in testing this detection system is to evaluate the distance measured in pixels compared to the actual distance measured in centimeters for objects detected in front of the robot using the front camera. Testing was carried out at various distances, specifically at: 20 cm, 40 cm, 60 cm, 80 cm, 100 cm, 120 cm, 140 cm, 160 cm, 180 cm, and 200 cm, all of which were measured from the robot's front camera. The results of these distance measurements are then presented in Table 2.

Table 2. Measured distance test results with actual distance

Number	Detection Results	Pixel Distance (Px)	Actual Distance (cm)	Detection Distance (cm)	Distance Difference (cm)	Error (%)
1		116	20	21	1	5
2		61	40	39	1	2.5
3		35	60	69	9	15

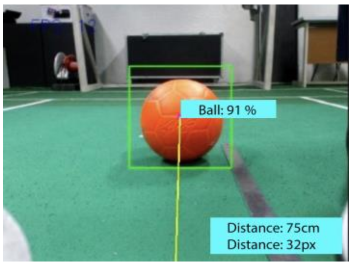
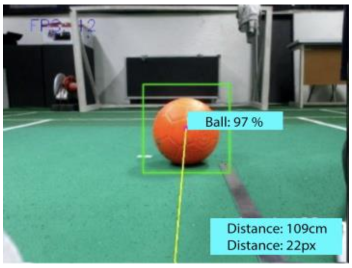
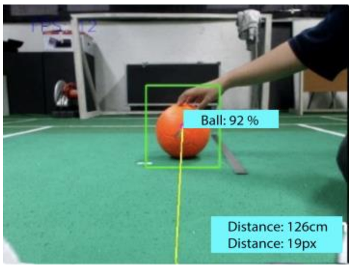
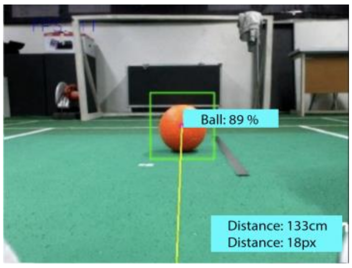
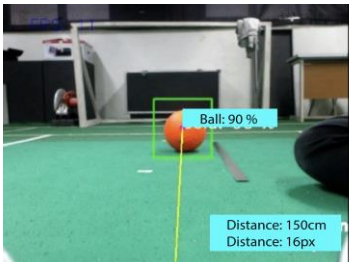
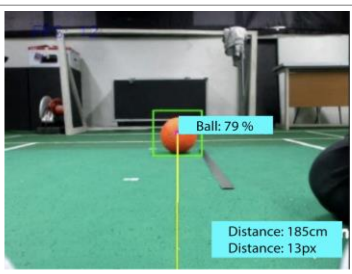
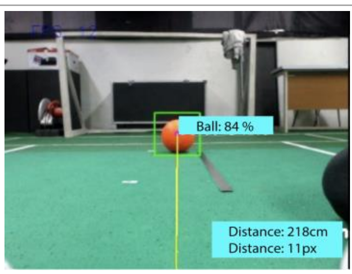
4		32	80	75	5	6.25
5		22	100	109	9	9
6		19	120	126	6	5
7		18	140	133	7	5
8		16	160	150	10	6.25
9		13	180	185	5	2.78
10		11	200	218	18	9

Table 2 presents the results of the distance test by measuring the distance in pixels calculated by the system and the actual distance known. In this table, the Detection distance as predicted by the system (the distance calculated by the system based on the pixel data captured by the camera) and the difference between the detection distance and the actual distance are also recorded. The percentage error indicating the difference between the two values is used to indicate the accuracy of the system in measuring the distance.

Referring to the results shown in the table, we observe that the system error generally varies depending on the measurement distance. At a distance of 40 cm, the error is relatively small, only 2.5%, but quickly increases to 15% at 60 cm. This shows that the system is more accurate at closer measurement distances, there is a tendency for the error of this system to be greater with increasing detection distance, although not absolute. The largest error is 18 centimeters at 200 centimeters with an error rate of 9 percent. The average percentage error is 6.58% in the percentage distance difference. The error value is obtained from calculations using calculations, $\left|\frac{\text{Actual distance}-\textit{Measured}\text{ distance}}{\textit{Actual distance}}\right| \times 100\%$ [72, 73], where the distance difference is the reduction between the detection distance and the actual distance. And for the average error value obtained from, $\left(\frac{\Delta error}{\text{number of tests}}\right) \times 100\% = \left(\frac{65.78}{10}\right) \times 100\% = 6.58\%$. The measured distance results were obtained using Equation 1. For more details, it is shown in Figure 10 which displays the distance difference that is getting bigger and bigger along with the shift in the actual distance value which is bigger.

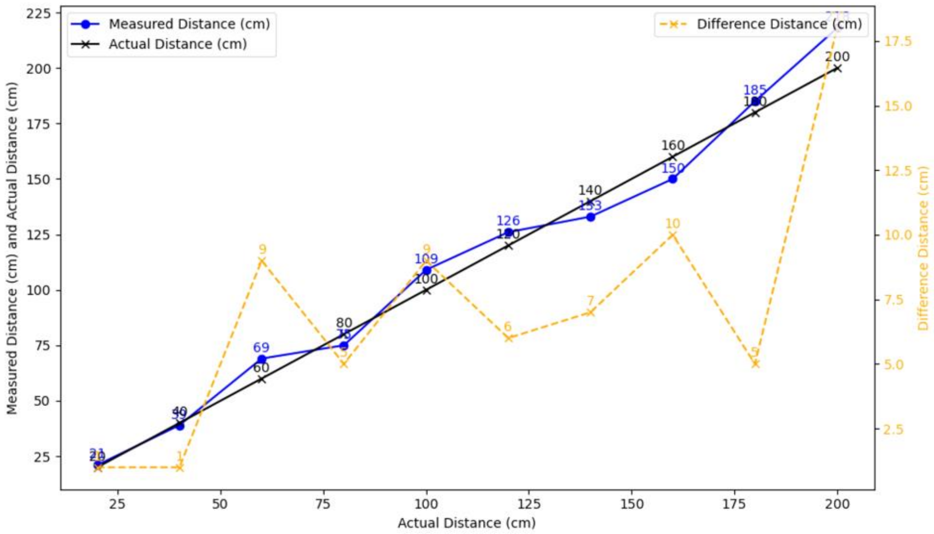


Figure 10. **The distance difference that is getting bigger and bigger along with the shift in the actual distance value which is bigger**

Where at the detection distance there is a pixel distance which indicates changes in the actual distance as well. These results show that the system has measurement accuracy where the smaller the pixel value, the actual distance value also increases to be farther away from the measured value. The results of system testing that have been carried out state that the detection system on the front camera of the wheeled soccer robot is able to detect the object of the ball and the distance between the ball and the actual distance.

3-1-2- Results of Angle Values on the Omnidirectional Camera towards the Front of the Robot

In testing, inserting an omnidirectional camera into this wheeled soccer robot uses a resolution of 640×480 divided into 4 positions. The omnidirectional camera will work to detect spherical objects in front of the robot, an overview of the detection system on the camera is shown in Figure 11.

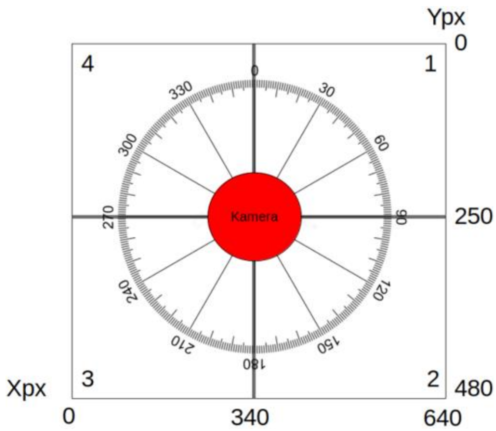
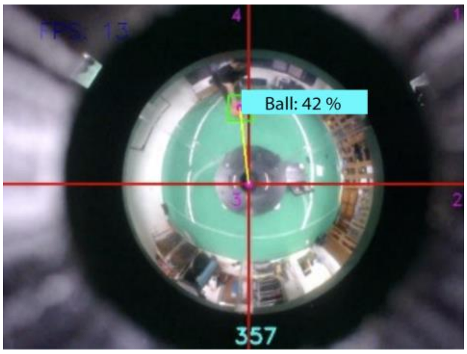
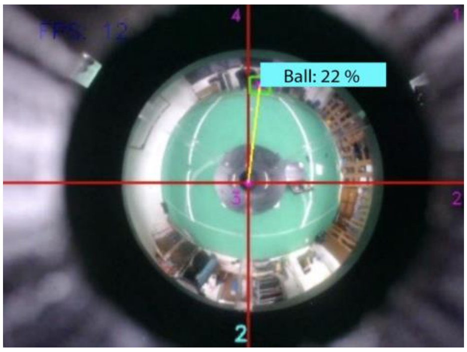


Figure 11. **Omnidirectional Camera Detection System Scenario**

This detection system utilizes a camera at the bottom facing upwards towards a convex mirror so that it can see in all directions or 360° utilizing an omnidirectional camera system. Testing the detection results on the omnidirectional camera focuses on the spherical object that was successfully detected and the angle value read by the system. The angle value is obtained using the calculation in Equation 2. The results of this test are shown in Table 3.

Table 3. Results of reading angle values on an omnidirectional camera

Number	X $_{pixel}$	Y $_{pixel}$	Angle	Result
1	0 – 340	0 – 250	$270^{\circ} < \theta < 360^{\circ}$	
2	340 - 640	0 – 250	$0^{\circ} < \theta < 90^{\circ}$	

4- Comparison with Previous Studies

Comparison of several studies shows significant progress in object detection algorithms on autonomous soccer robots. Barry et al. [45] applied CNN-YOLOv3 (XYOLO) and obtained an FPS of 9.66 with an accuracy of 68%, using a limited dataset and frames. Soebhakti et al. [48] used YOLOv3 with a large dataset containing 52000 images and obtained a higher FPS of 28.3 and an accuracy of 87.7%, indicating better efficiency on powerful hardware. Irfan & Widodo [49] also obtained an accuracy of 67%, but had a lower frame resolution (128×128) using CNN with an FPS of 13. Meanwhile, Sanubari & Puriyanto [43] did not mention FPS, but their accuracy with YOLOv3 was very high at 93.2%. In the same study, YOLOv3-Tiny achieved an accuracy of 81.8% but with a higher speed. This shows that the number of FPS is not always directly proportional to the level of accuracy obtained by the system. Not only talking about the performance of the designed model, FPS is also greatly influenced by the specifications of the hardware used such as the CPU, this is in line with what was stated in Santos et al. [74] and Jaiswal et al. [75].

On the other hand, the research we conducted in this paper, implemented SSD-MobileNetV2 with FPS 12 and accuracy of 93.42%, which is a good compromise between speed and accuracy; in this system, an omnidirectional camera is used to detect the ball in a dynamic environment, which makes this system different from other studies. A summary of the comparison of this study with previous studies is summarized in Table 4 and Figure 12.

Table 4. Summary of comparison with previous studies

Ref.	Year	Algorithm	Dataset		Frame Size	FPS	Accuracy (%)
			Training	Testing			
[45]	2019	CNN-YOLOv3 (XYOLO)	(Not mentioned) 90%	(Not mentioned) 10%	256×256	9.66	68.00
[48]	2019	YOLOv3	52000		416×416	28.3	87.70
[49]	2020	CNN	2000	2000	128×128	13	67.00
[43]	2022	YOLOv3	7000	1000	416×416	-	93.20
[43]	2022	YOLOv3-Tiny	7000	1000	416×416	-	81.80
This research	2024	SSD-MobileNetV2	2966	741	320×320	12	93.42

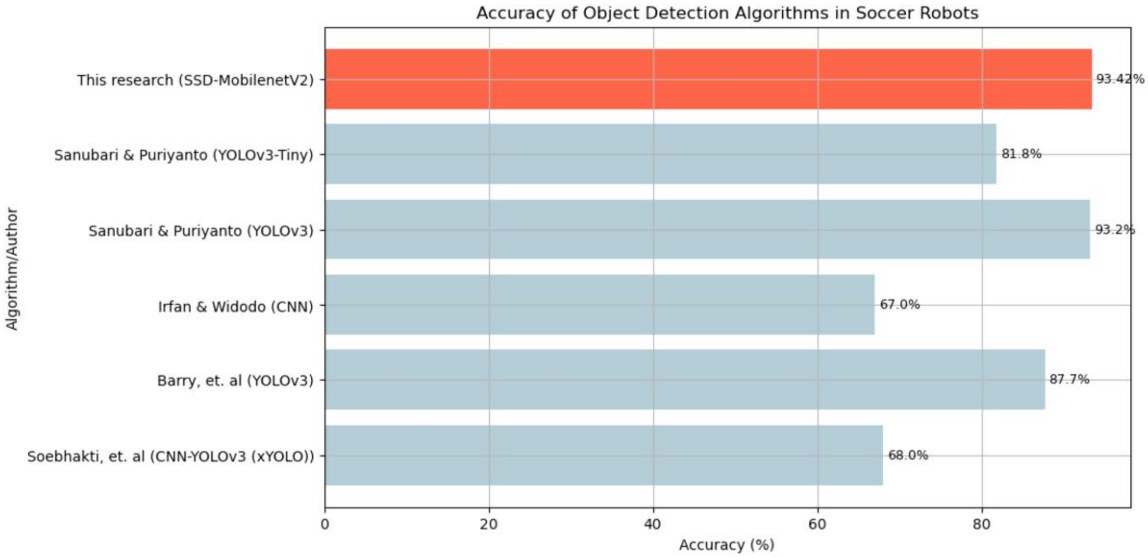


Figure 12. Comparison of object detection algorithm accuracy in Soccer Robots with previous studies

5- Conclusion

Based on the test results of the ball object detection system in front of the wheeled soccer robot, there is the actual distance value detected on the front camera and the angle value read on the omnidirectional camera. The system is able to detect spherical objects well. The omnidirectional camera is able to detect objects up to 4m, while the front camera detects objects up to 2m using the single shot multi-box detector MobileNetV2 method. During the test, the fps produced was classified as good, where the fps obtained was around 8-12 FPS (Frames Per Second). The test results of the measured distance and the actual distance also have an average error value of 6.58%. Things that influence the difference between the detected distance and the actual distance are the placement of the front camera on the wheeled soccer robot. For the improvement of the object detection system related to ball recognition in the soccer player robot in this study, several major issues need to be addressed. First of all, the currently available FPS between 8 and 12 FPS seems too low to accommodate the fast movements that occur during competitive scenarios. Future work should focus on increasing the FPS through hyperparameter optimization and more sophisticated computing hardware for the system. Hyperparameter tuning will improve the performance of the MobileNetV2 SSD model in terms of accuracy and detection speed, and make it suitable for real-time use. Furthermore, assessing the performance of higher computing systems will help to understand their impact on FPS and overall system performance. Adding more changes in lighting, ball color, and background to the dataset will improve its performance in highly dynamic environments. These improvements will increase the reliability and effectiveness of the competitive robotics system, where fast and accurate object recognition is critical.

6- Declarations

6-1- Author Contributions

Conceptualization, R.D.P.; methodology, I.D.Y.; validation, R.D.P.; formal analysis, I.D.Y. and F.F.; writing—original draft preparation, I.D.Y. and F.F; writing—review and editing, A.M.; supervision, I.S. and H.M. All authors have read and agreed to the published version of the manuscript.

6-2- Data Availability Statement

The data presented in this study are available in the article.

6-3- Funding and Acknowledgements

This work was supported by Indonesia Endowment Fund for Education (LPDP), Ministry of Finance of the Republic of Indonesia.

6-4- Institutional Review Board Statement

Not applicable.

6-5- Informed Consent Statement

Not applicable.

6-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7- References

[1] Javaid, M., Haleem, A., Singh, R. P., & Suman, R. (2021). Substantial capabilities of robotics in enhancing industry 4.0 implementation. Cognitive Robotics, 1, 58–75. doi:10.1016/j.cogr.2021.06.001.

[2] Purwono, P., Wulandari, A. N. E., Ma'arif, A., & Salah, W. A. (2025). Understanding Generative Adversarial Networks (GANs): A Review. Control Systems and Optimization Letters, 3(1), 36–45. doi:10.59247/csol.v3i1.170.

[3] Furizal, Ma'arif, A., & Rifaldi, D. (2023). Application of Machine Learning in Healthcare and Medicine: A Review. Journal of Robotics and Control (JRC), 4(5), 621–631. doi:10.18196/jrc.v4i5.19640.

[4] Sharkawy, A.-N. (2025). Impact of Inertial and External Forces on Joint Dynamics of Robotic Manipulator: Experimental Insights. Control Systems and Optimization Letters, 3(1), 1–7. doi:10.59247/csol.v3i1.163.

[5] Soori, M., Arezoo, B., & Dastres, R. (2023). Artificial intelligence, machine learning and deep learning in advanced robotics, a review. Cognitive Robotics, 3, 54–70. doi:10.1016/j.cogr.2023.04.001.

[6] Oliveira, L. F. P., Moreira, A. P., & Silva, M. F. (2021). Advances in agriculture robotics: A state-of-the-art review and challenges ahead. Robotics, 10(2), 52. doi:10.3390/robotics10020052.

[7] Grau, A., Indri, M., Lo Bello, L., & Sauter, T. (2021). Robots in Industry: The Past, Present, and Future of a Growing Collaboration with Humans. IEEE Industrial Electronics Magazine, 15(1), 50–61. doi:10.1109/MIE.2020.3008136.

[8] Dzedzickis, A., Subačiūtė-Žemaitienė, J., Šutinys, E., Samukaitė-Bubnienė, U., & Bučinskas, V. (2021). Advanced Applications of Industrial Robotics: New Trends and Possibilities. Applied Sciences, 12(1), 135. doi:10.3390/app12010135.

[9] Sitompul, E., Teguh, M., Yaqin, I., Jaya Tarigan, H., Tampubolon, G. M., Samsuri, F., & Galina, M. (2025). Integration of Pixy2 Camera Sensor and Coordinate Transformation for Automatic Color-Based Implementation of a Pick-and-Place Arm Robot. Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika, 11(1), 92–109. doi:10.26555/JITEKI.V11I1.30717.

[10] Furizal, Ma'arif, A., Firdaus, A. A., & Rahmaniar, W. (2023). Future Potential of E-Nose Technology: A Review. International Journal of Robotics and Control Systems, 3(3), 449–469. doi:10.31763/ijrcs.v3i3.1091.

[11] Solikhah, E. W., Asih, H. M., Astuti, F. H., Ghazali, I., & Mohammad, E. Bin. (2024). Industry 4.0 Readiness Trends: A Bibliometric and Visualization Analysis. International Journal of Robotics and Control Systems, 4(1), 105–124. doi:10.31763/ijrcs.v4i1.1247.

[12] Górriz, J. M., Álvarez-Illán, I., Álvarez-Marquina, A., Arco, J. E., Atzmueller, M., Ballarini, F., Barakova, E., Bologna, G., Bonomini, P., Castellanos-Dominguez, G., Castillo-Barnes, D., Cho, S. B., Contreras, R., Cuadra, J. M., Domínguez, E., Domínguez-Mateos, F., Duro, R. J., Elizondo, D., Fernández-Caballero, A., … Ferrández-Vicente, J. M. (2023). Computational approaches to Explainable Artificial Intelligence: Advances in theory, applications and trends. Information Fusion, 100, 101945. doi:10.1016/j.inffus.2023.101945.

[13] Wijaya, S. A., Famuji, T. S., Mu'min, M. A., Safitri, Y., Tristanti, N., Dahmani, A., Driss, Z., Sharkawy, A.-N., & Al-Sabur, R. (2025). Trends and Impact of the Viola-Jones Algorithm: A Bibliometric Analysis of Face Detection Research (2001-2024). Scientific Journal of Engineering Research, 1(1), 33–42. doi:10.64539/sjer.v1i1.2025.8.

[14] Semeraro, F., Griffiths, A., & Cangelosi, A. (2022). Human–robot collaboration and machine learning: A systematic review of recent research. Robotics and Computer-Integrated Manufacturing, 79, 102432. doi:10.1016/j.rcim.2022.102432.

[15] Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. SN Computer Science, 2(3), 160. doi:10.1007/s42979-021-00592-x.

[16] Fahmi, M., Yudhana, A., Sunardi, Abdel-Nasser Sharkawy, & Furizal. (2025). Classification for Waste Image in Convolutional Neural Network Using Morph-HSV Color Model. Scientific Journal of Engineering Research, 1(1), 18–25. doi:10.64539/sjer.v1i1.2025.12.

[17] Ali, A. K., & Mahmoud, M. S. (2022). Methodologies and Applications of Artificial Intelligence in Systems Engineering. International Journal of Robotics and Control Systems, 2(1), 201–229. doi:10.31763/ijrcs.v2i1.532.

[18] Kumar, G. S., Kumar, M. D., Reddy, S. V. R., Kumari, B. V. S., & Reddy, C. R. (2024). Injury Prediction in Sports using Artificial Intelligence Applications: A Brief Review. Journal of Robotics and Control (JRC), 5(1), 16–26. doi:10.18196/jrc.v5i1.20814.

[19] Saranya, A., & Subhashini, R. (2023). A systematic review of Explainable Artificial Intelligence models and applications: Recent developments and future trends. Decision Analytics Journal, 7, 100230. doi:10.1016/j.dajour.2023.100230.

[20] Sarker, I. H. (2022). AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems. SN Computer Science, 3(2), 1–20. doi:10.1007/s42979-022-01043-x.

[21] Rashid, A. Bin, & Kausik, M. A. K. (2024). AI revolutionizing industries worldwide: A comprehensive overview of its diverse applications. Hybrid Advances, 7, 100277. doi:10.1016/j.hybadv.2024.100277.

[22] Diantoro, R., & Aji, W. S. (2020). Design of Blessing Ball Throwing Techniques for KRAI 2018 Robot. Buletin Ilmiah Sarjana Teknik Elektro, 2(3), 119. doi:10.12928/biste.v2i3.2623.

[23] Sukaswanto, P., & Fadlil, A. (2020). Humanoid Dancer Robot Movement Synchronization System with Zigbee. Buletin Ilmiah Sarjana Teknik Elektro, 2(3), 103. doi:10.12928/biste.v2i3.2695.

[24] Wajiansyah, A., Malani, R., Supriadi, & Gaffar, A. F. O. (2022). Optimization of Humanoid Robot Leg Movement Using Open CM 9.04. Journal of Robotics and Control (JRC), 3(5), 699–709. doi:10.18196/jrc.v3i5.15071.

[25] Setiawardhana, Kurnianto Wibowo, I., & Achmad Husein Bernardt, N. (2025). Prediction of Ball Position Using CNN Methods With Zed Camera on Goalkeeper Robot Application. IEEE Access, 13, 41559–41570. doi:10.1109/ACCESS.2025.3546346.

[26] Putra, P. M., & Widodo, N. S. (2023). Development of the Design and Control of a Hexapod Robot for Uneven Terrain. Buletin Ilmiah Sarjana Teknik Elektro, 5(4), 552–566. doi:10.12928/biste.v5i4.9426.

[27] Prasetio, Y., & Widodo, N. S. (2023). Hexapod Robot Movement Control for Uneven Terrain. Control Systems and Optimization Letters, 1(2), 82–86. doi:10.59247/csol.v1i2.23.

[28] Antonioni, E., Suriani, V., Riccio, F., & Nardi, D. (2021). Game Strategies for Physical Robot Soccer Players: A Survey. IEEE Transactions on Games, 13(4), 342–357. doi:10.1109/TG.2021.3075065.

[29] Jati, H., Ilyasa, N. A., & Dominic, D. D. (2024). Enhancing Humanoid Robot Soccer Ball Tracking, Goal Alignment, and Robot Avoidance Using YOLO-NAS. Journal of Robotics and Control (JRC), 5(3), 829–838. doi:10.18196/jrc.v5i3.21839.

[30] Handayani, A. N., Pusparani, F. A., Lestari, D., Wirawan, I. M., Wibawa, A. P., & Fukuda, O. (2023). Real-Time Obstacle Detection for Unmanned Surface Vehicle Maneuver. International Journal of Robotics and Control Systems, 3(4), 765–779. doi:10.31763/ijrcs.v3i4.1147.

[31] Ghasemi, Y., Jeong, H., Choi, S. H., Park, K. B., & Lee, J. Y. (2022). Deep learning-based object detection in augmented reality: A systematic review. Computers in Industry, 139, 103661. doi:10.1016/j.compind.2022.103661.

[32] Bai, Q., Li, S., Yang, J., Song, Q., Li, Z., & Zhang, X. (2020). Object detection recognition and robot grasping based on machine learning: A survey. IEEE Access, 8, 181855–181879. doi:10.1109/ACCESS.2020.3028740.

[33] Manakitsa, N., Maraslidis, G. S., Moysis, L., & Fragulis, G. F. (2024). A Review of Machine Learning and Deep Learning for Object Detection, Semantic Segmentation, and Human Action Recognition in Machine and Robotic Vision. Technologies, 12(2), 15. doi:10.3390/technologies12020015.

[34] Dewantara, B. S. B., & Ariyadi, B. N. D. (2021). Adaptive Behavior Control for Robot Soccer Navigation Using Fuzzy-based Social Force Model. Smart Science, 9(1), 14–29. doi:10.1080/23080477.2021.1871799.

[35] Muwardi, R., Nugroho, I. P., Salamah, K. S., Yunita, M., Rahmatullah, R., & Chung, G. J. (2024). Optimization of YOLOv4-Tiny Algorithm for Vehicle Detection and Vehicle Count Detection Embedded System. Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika, 10(3), 639–648. doi:10.26555/jiteki.v10i3.29693.

[36] Chotikunnan, P., Puttasakul, T., Chotikunnan, R., Panomruttanarug, B., Sangworasil, M., & Srisiriwat, A. (2023). Evaluation of Single and Dual Image Object Detection through Image Segmentation Using ResNet18 in Robotic Vision Applications. Journal of Robotics and Control (JRC), 4(3), 263–277. doi:10.18196/jrc.v4i3.17932.

[37] Lee, T. H., Shair, E. F., Abdullah, A. R., Rahman, K. A., Ali, N. M., Saharuddin, N. Z., & Nazmi, N. (2025). Comparative Analysis of 1D – CNN, GRU, and LSTM for Classifying Step Duration in Elderly and Adolescents Using Computer Vision. International Journal of Robotics and Control Systems, 5(1), 426–439. doi:10.31763/ijrcs.v5i1.1588.

[38] Putra, R. K., Alfarisy, G. A. F., Nugraha, F. W., & Nuryono, A. A. (2024). Automatic Plant Disease Classification with Unknown Class Rejection using Siamese Networks. Buletin Ilmiah Sarjana Teknik Elektro, 6(3), 308–316. doi:10.12928/biste.v6i3.11619.

[39] Banjaransari, M., & Prahara, A. (2023). Image Classification of Wayang Using Transfer Learning and Fine-Tuning of CNN Models. Buletin Ilmiah Sarjana Teknik Elektro, 5(4), 632–641. doi:10.12928/biste.v5i4.9977.

[40] Szemenyei, M., & Estivill-Castro, V. (2022). Fully neural object detection solutions for robot soccer. Neural Computing and Applications, 34(24), 21419–21432. doi:10.1007/s00521-021-05972-1.

[41] Wibowo, I. K., Haq, M. A., Bachtiar, M. M., Dewantara, B. S. B., & Ihsan, F. L. H. (2019). Ball Detection using Local Binary Pattern in Middle Size Robot Soccer (ERSOW). 2019 2nd International Conference of Computer and Informatics Engineering (IC2IE), 29–32. doi:10.1109/ic2ie47452.2019.8940835.

[42] Le, V. H., & Pham, T. L. (2024). Ovarian Tumors Detection and Classification on Ultrasound Images Using One-stage Convolutional Neural Networks. Journal of Robotics and Control (JRC), 5(2), 561–581. doi:10.18196/jrc.v5i2.20589.

[43] Sanubari, F. F., & Puriyanto, R. D. (2022). Ball and Goal Detection with the YOLO Method Using an Omnidirectional Camera on the KRSBI-B Robot. Buletin Ilmiah Sarjana Teknik Elektro, 4(2), 76–85. doi:10.12928/biste.v4i2.6712.

[44] Cruz, N., Leiva, F., & Ruiz-del-Solar, J. (2021). Deep learning applied to humanoid soccer robotics: playing without using any color information. Autonomous Robots, 45(3), 335–350. doi:10.1007/s10514-021-09966-9.

[45] Barry, D., Shah, M., Keijsers, M., Khan, H., & Hopman, B. (2019). xYOLO: A Model For Real-Time Object Detection In Humanoid Soccer on Low-End Hardware. 2019 International Conference on Image and Vision Computing New Zealand (IVCNZ), 1–6. doi:10.1109/ivcnz48456.2019.8960963.

[46] Fathurrahman, H. I. K., & Li-Yi, C. (2022). Character Translation on Plate Recognition with Intelligence Approaches. Buletin Ilmiah Sarjana Teknik Elektro, 4(3), 105–110. doi:10.12928/biste.v4i3.7161.

[47] Chuang, G., & Li-Jia, C. (2023). Bridge Crack Detection Based on Attention Mechanism. International Journal of Robotics and Control Systems, 3(2), 259–269. doi:10.31763/ijrcs.v3i2.929.

[48] Soebhakti, H., Prayoga, S., Fatekha, R. A., & Fashla, M. B. (2019). The Real-Time Object Detection System on Mobile Soccer Robot using YOLO v3. 2019 2nd International Conference on Applied Engineering (ICAE), 1–6. doi:10.1109/icae47758.2019.9221734.

[49] Irfan, S. Al, & Widodo, N. S. (2020). Application of Deep Learning Convolution Neural Network Method on KRSBI Humanoid R-SCUAD Robot. Buletin Ilmiah Sarjana Teknik Elektro, 2(1), 40. doi:10.12928/biste.v2i1.985.

[50] Aulia, U., Hasanuddin, I., Dirhamsyah, M., & Nasaruddin, N. (2024). A new CNN-BASED object detection system for autonomous mobile robots based on real-world vehicle datasets. Heliyon, 10(15), e35247. doi:10.1016/j.heliyon.2024.e35247.

[51] Mohamed Ameerdin, M. I., Jamaluddin, M. H., Shukor, A. Z., & Mohamad, S. (2024). A Review of Deep Learning-Based Defect Detection and Panel Localization for Photovoltaic Panel Surveillance System. International Journal of Robotics and Control Systems, 4(4), 1746–1771. doi:10.31763/ijrcs.v4i4.1579.

[52] Rani, S., Ghai, D., & Kumar, S. (2022). Object detection and recognition using contour based edge detection and fast R-CNN. Multimedia Tools and Applications, 81(29), 42183–42207. doi:10.1007/s11042-021-11446-2.

[53] Handayani, A. N., Amaliya, S., Akbar, M. I., Wiryawan, M. Z., Liang, Y. W., & Kurniawan, W. C. (2025). Hand Keypoint-Based CNN for SIBI Sign Language Recognition. International Journal of Robotics and Control Systems, 5(2), 813–829. doi:10.31763/ijrcs.v5i2.1745.

[54] Allebosch, G., Van Hamme, D., Veelaert, P., & Philips, W. (2022). Efficient detection of crossing pedestrians from a moving vehicle with an array of cameras. Optical Engineering, 62(3), 031210-031210. doi:10.1117/1.oe.62.3.031210.

[55] Dang, T. V. (2023). Smart Attendance System based on Improved Facial Recognition. Journal of Robotics and Control (JRC), 4(1), 46–53. doi:10.18196/jrc.v4i1.16808.

[56] Aghaee, F., Fazl-Ersi, E., & Noori, H. (2024). MDSSD-MobV2: An embedded deconvolutional multispectral pedestrian detection based on SSD-MobileNetV2. Multimedia Tools and Applications, 83(15), 43801–43829. doi:10.1007/s11042-023-17188-7.

[57] Kirana, K. C., & Abdulrahman, S. A. K. (2024). Random Multi-Augmentation to Improve TensorFlow-Based Vehicle Plate Detection. Buletin Ilmiah Sarjana Teknik Elektro, 6(2), 113–125. doi:10.12928/biste.v6i2.10542.

[58] Biglari, A., & Tang, W. (2022). A Vision-Based Cattle Recognition System Using TensorFlow for Livestock Water Intake Monitoring. IEEE Sensors Letters, 6(11). doi:10.1109/LSENS.2022.3215699.

[59] Wang, L., Shoulin, Y., Alyami, H., Laghari, A. A., Rashid, M., Almotiri, J., Alyamani, H. J., & Alturise, F. (2024). A novel deep learning-based single shot multibox detector model for object detection in optical remote sensing images. Geoscience Data Journal, 11(3), 237–251. doi:10.1002/gdj3.162.

[60] Cai, J., Makita, Y., Zheng, Y., Takahashi, S., Hao, W., & Nakatoh, Y. (2022). Single shot multibox detector for honeybee detection. Computers and Electrical Engineering, 104, 108465. doi:10.1016/j.compeleceng.2022.108465.

[61] Bouazizi, O., Azroumahli, C., El Mourabit, A., & Oussouaddi, M. (2024). Road Object Detection using SSD-MobileNet Algorithm: Case Study for Real-Time ADAS Applications. Journal of Robotics and Control (JRC), 5(2), 551–560. doi:10.18196/jrc.v5i2.21145.

[62] Magalhães, S. A., Castro, L., Moreira, G., Dos Santos, F. N., Cunha, M., Dias, J., & Moreira, A. P. (2021). Evaluating the single-shot multibox detector and yolo deep learning models for the detection of tomatoes in a greenhouse. Sensors, 21(10), 3569. doi:10.3390/s21103569.

[63] Aldhahri, E., Aljuhani, R., Alfaidi, A., Alshehri, B., Alwadei, H., Aljojo, N., Alshutayri, A., & Almazroi, A. (2023). Arabic Sign Language Recognition Using Convolutional Neural Network and MobileNet. Arabian Journal for Science and Engineering, 48(2), 2147–2154. doi:10.1007/s13369-022-07144-2.

[64] Mohammed, O. N. (2024). Enhancing Pulmonary Disease Classification in Diseases: A Comparative Study of CNN and Optimized MobileNet Architectures. Journal of Robotics and Control (JRC), 5(2), 427–440. doi:10.18196/jrc.v5i2.21422.

[65] Dang, T. V. (2022). Smart home Management System with Face Recognition Based on ArcFace Model in Deep Convolutional Neural Network. Journal of Robotics and Control (JRC), 3(6), 754–761. doi:10.18196/jrc.v3i6.15978.

[66] Bi, C., Wang, J., Duan, Y., Fu, B., Kang, J. R., & Shi, Y. (2022). MobileNet Based Apple Leaf Diseases Identification. Mobile Networks and Applications, 27(1), 172–180. doi:10.1007/s11036-020-01640-1.

[67] Jia, S., Jiang, S., Lin, Z., Li, N., Xu, M., & Yu, S. (2021). A survey: Deep learning for hyperspectral image classification with few labeled samples. Neurocomputing, 448, 179–204. doi:10.1016/j.neucom.2021.03.035.

[68] Wiguna, I. W. A. A., Huizen, R. R., & Pradipta, G. A. (2024). Optimization of Vehicle Detection at Intersections Using the YOLOv5 Model. Scientific Journal of Electrical Engineering, Computer Science and Informatics (JITEKI), 10(4), 885–896. doi:10.26555/JITEKI.V10I4.29309.

[69] Julianda, R. R., & Puriyanto, R. D. (2024). Tracking Ball Using YOLOv8 Method on Wheeled Soccer Robot with Omnidirectional Camera. Buletin Ilmiah Sarjana Teknik Elektro, 6(2), 203–213. doi:10.12928/biste.v6i2.10816.

[70] Pérez, I., & Figueroa, M. (2021). A heterogeneous hardware accelerator for image classification in embedded systems. Sensors, 21(8), 2637. doi:10.3390/s21082637.

[71] Kamath, V., & Renuka, A. (2024). Investigation of MobileNet-Ssd on human follower robot for stand-alone object detection and tracking using Raspberry Pi. Cogent Engineering, 11(1). doi:10.1080/23311916.2024.2333208.

[72] Tseng, A. A., Tanaka, M., & Leeladharan, B. (2002). Laser-based internal profile measurement system. Automation in Construction, 11(6), 667–679. doi:10.1016/S0926-5805(02)00008-0.

[73] Chen, Y., Yi, H., & He, C. (2018). Introduction and investigation of magnetic location system for steam-assisted gravity drainage dual-horizontal heavy oil wells. Advances in Mechanical Engineering, 10(9), 1-8. doi:10.1177/1687814018798978.

[74] Santos, R. C. C. D. M., Coelho, M., & Oliveira, R. (2024). Real-time Object Detection Performance Analysis Using YOLOv7 on Edge Devices. IEEE Latin America Transactions, 22(10), 799–805. doi:10.1109/TLA.2024.10705971.

[75] Jaiswal, M., Sharma, A., & Saini, S. (2025). Hardware acceleration of Tiny YOLO deep neural networks for sign language recognition: A comprehensive performance analysis. Integration, 100, 102287. doi:10.1016/j.vlsi.2024.102287.