*Journal of Intelligent System and Telecommunications 1(*1) (2024) 13-24

`

Journal of Intelligent System and Telecommunications

Journal homepage: https://journal.unesa.ac.id/index.php/jistel/index

# SURFACE DETECTION FOR QUADRUPED ROBOT USING YOLO-V3 TINY

*Muhammad Azzaky Rizki F*[1*],*Lilik Anifah.*[2], *Aye Aye Mon* [3]

[12]Universitas Negeri Surabaya, Surabaya, Indonesia
[3]Department of Mechanical Engineering Department Technical Promotion Training Centre, Baelin, Myanmar
E-mail: muhammadazzaky.21041@mhs.unesa.id

## ARTICLE INFO

***Article History:***
*Submitted/Received 6 November 2024*
*First Revised 20 December 2024*
*Accepted 20 December 2024*
*First Available 21 December 2024*
*Publication 21 December 2024*

***Keyword:***
*Artificial Intelligence, Deep Learning, Machine Learning, Quadruped Robot, YOLO, Confusion Matrix*

## ABSTRACT

Robotics has become a very important field for engineers, as robots can perform various assigned tasks quickly and efficiently. specifically in the field of robotic with legs. In the Indonesian robot competition there are many categories, SAR is the one of them. In the SAR (Search and Rescue) category there are many surfaces that robot has to pass through, such as coral, and marbles. Different surfaces have different movements to pass through. The researchers have designed a quadruped with auto changed movement when pass through the surfaces, with YOLOv3-Tiny model on a raspberry pi 5 was placed on a quadruped robot. YOLOv3-Tiny model can detect surfaces according to dataset that has been trained. The model that has been trained with YOLOv3-Tiny is capable of detecting surfaces. In this research has an accuracy of detecting up to 100% in 80 pictures. The precision value up to 1, the recall value up to 1, and the F1-score is up to 1.

## 1. INTRODUCTION

Robotics has become a very important field for engineers, as robots can perform various assigned tasks quickly and efficiently. specifically in the field of robotic with legs[1]. Technology and Higher Education of the Republic of Indonesia organizes the Indonesia Robot Competition (KRI) annually, featuring various categories, with SAR (Search and Rescue) being one of them[2].

In SAR category, according to the rules that were released couple of months before the competition started. participants must adhere to specific guidelines. Robot must navigate various surfaces, by using a quadruped robot. Quadruped is a four-legged robot that consists of a servo arrangement that forms like legs. The pros of Quadruped robots are reliability for any kind of surfaces even uneven floor. Quadruped robots also have high maneuverability. There are 2 types of Quadruped Robot, sprawling type and mammal's type. Sprawling type has a shaped like spider's legs, for the mammal's type has a shaped like dogs, cats, and pigs[3]. Quadruped robot with sprawling type is solution to make it easy to pass through any surfaces. This quadruped robot is using inverse kinematics to make the robot's movement more smoother, and faster to define legs pose[4].

The robot is used to detect surfaces by using Arducam 16mp camera who's connected to Raspberry PI 5 board. With the expectation that the robot can detect the surfaces it needs to pass through. In this research the robot walk pass through 2 surfaces such as marbles and coral. By using YOLO algorithm as an object detector, the researcher expecting that YOLO algorithm is an excellent model that can be implemented in this robot[5]. YOLOv3 is a light model to implementing in low performance PC it also have a decent response time than other object detection method[6]. YOLOv3-tiny's model is a great match for Raspberry Pi 5 due to its specification, although Raspberry Pi 5 doesn't have an AI chip, this model still has a decent performance[7]. The researcher hopes that in the future, there will be an autonomous quadruped robot capable. of detecting and adjusting to the surfaces it needs to pass through.

## 2. METHODS

Design a system on this research, it takes a several step to work on it. The step that required can be seen on the **Figure 1**.

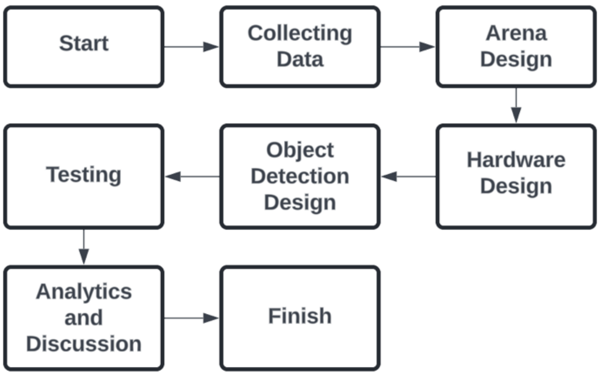


**Figure** 1. Research Step

In the first stage, the researcher identifies gaps in existing journals and searches for references from available journals. Then, design the arena that will be created for the robot arena. Designing the appropriate hardware required for the robot system. This robot using

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

object detection method for detecting surfaces. After designing arena, hardware, and object detection system, testing is a must step to do, then analyze from the testing and do some discussion each other at the last step.

### 2.1. Material

For this research, software development and design were implemented to enable object detection for the robot. A reference was essential during the implementation phase.

#### 2.1.1 Quadruped Robot

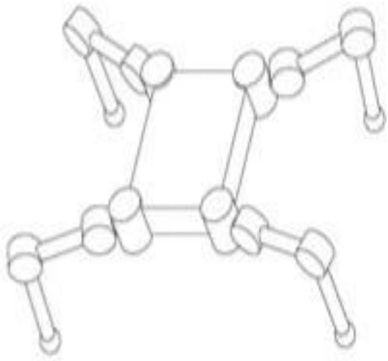

**Figure 2.** Quadruped Robot Sprawling Type

Quadruped Robots have four legs or limbs and move in a similar way to four-legged animals **(Figure 2)**, Quadruped robot also decent type for any surfaces that needs to pass through. It has three servos for each leg, with inverse kinematics calculation to make it easier to motion control[4].

Quadruped robot has types of legs configuration. The Mammals type and the Sprawling type. Mammals type using straight line vertically down for each servo, and for the Sprawling type at the first servo that close to the body of its robot using horizontal position, and the farthest servo using vertical position[8].

#### 2.1.2 STM32F4 Discovery

STM32F4 Discovery is a one of the best microcontroller device made by STMicroelectronics. It has ARM Cortex IC with 32 bit ARM Architecture microcontrollers (**Figure 3**). The speed is up to 168 MHz. STM32F4 Discovery has about 100 input output pins that have respective characteristics USART, UART, TIMER, ADC, DAC, AUDIO and i2c.

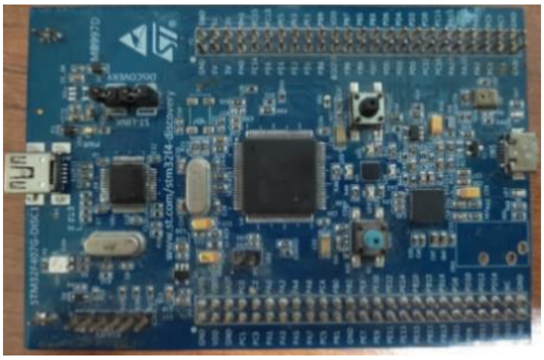

**Figure 3.** STM32F4 Discovery

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

The STM32F4 Discovery requires voltage around 3-5V DC. STM32F4 use C as the main programming language with Co IDE or STM CUBE IDE. The researcher using CoCoox Co IDE as a main IDE for this research[9].

### 2.1.3 Arduino Nano

The Arduino Nano V3 is part of arduino family, it has compact size for a microcontroller **(Figure 4)**. Arduino Nano V3 is manufactured by Gravitech. Arduino Nano V3 use Atmega328P IC [10]. this microcontroller's speed is up to 16MHz, The Arduino Nano has 30 pins, eight of which (19-26 pins) are Analog pins. The other 14 pins are digital [11]. In this research, arduino nano is used to send data from Raspberry Pi 5 to STM32F4.

**Figure 4.** Arduino Nano V3

### 2.1.4 Dynamixel MX-28

The Dynamixel MX28 is a servo series from the company Dynamixel (**Figure 5**). This servo differs from PWM servos; Dynamixel servos use serial communication for communication between the microcontroller and the servo, with communication speeds of up to 80.000 bps–2 Mbps. The Dynamixel MX series is a metal gear servo series available in various sizes. The Dynamixel MX28 uses a 12VDC power input [12].


**Figure 5.** Dynamixel MX28

### 2.1.5 Gyroscope BNO055

The BNO055 gyroscope is a module that includes three sensors: an accelerometer, gyroscope, and magnetometer **(Figure 6)**. This gyroscope sensor has 9 degrees of freedom (DoF), with a 3 axis of accelerometer, 3 axis of gyroscope, and 3 axis of magnetometer. It uses a 32-bit ARM Cortex M0 in a single module and operates with a power supply of 3.3V to 5V. The researcher is only using this sensor for the gyroscope function. The sensor can operate in 9DoF mode or IMU mode. In 9DoF mode, the yaw axis of the gyroscope automatically faces north, but it requires calibration before use [13]. In contrast, in IMU mode, the yaw reading points forward from the sensor's initial orientation when it is powered on. BNO055 is fully compatible with Arduino and Raspberry Pi [14].

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

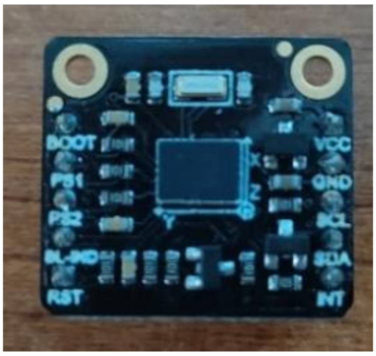


**Figure 6.** BNO055 Gyroscope Sensor

### 2.1.6 Raspberry Pi 5

The Raspberry Pi 5 is powered by a 64-bit quad-core with Arm Architecture. Raspberry Pi use ARM Cortex-A76, this processor can run up to 2.4GHz, offers a 2-3x improvement in CPU performance over its predecessor, the Raspberry Pi 4[15]. It also features an 800MHz VideoCore VII GPU, making it a solid choice for tasks like object detection, where it transmits data containing the coordinates of detected objects. The researcher use 8GB of ram on this Raspberry Pi 5 (**Figure 7**).


**Figure 7.** Raspberry Pi 5
Source (www.raspberrypi.com)

### 2.1.7 Arducam 16MP

The ArduCam 16MP is a camera module with a 16-megapixel resolution (**Figure 8**). It features a Sony IMX 519 autofocus camera sensor, supporting a maximum resolution of 1080p at 30 fps and 720p at 60 fps. This camera module is lightweight, weighing only 3 grams[16].


**Figure 8.** Arducam
Source (www.arducam.com)

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

### 2.1.8 Arena Design

The arena is based on the rules of the Indonesian Robot Competition

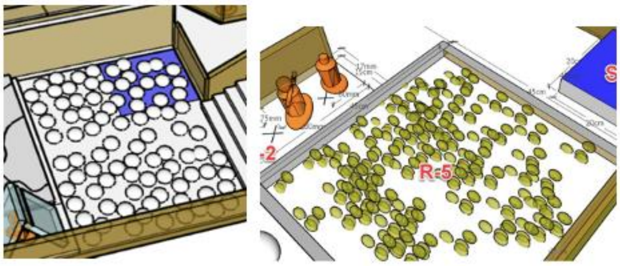

**Figure 9**. Surface Arena

The coral obstacle zone spans an area of 51 cm in length and 73 cm in width, utilizing coral stones as debris, covering approximately 90% of the area. These coral stones range in size from 3 to 5 cm and are distributed across the entire zone. The coral obstacle is illustrated in **Figure 9**. The marble obstacle has an area of 45 cm x 45 cm, with a density of marbles arranged in two layers and a 2 cm-high barrier to contain them[2].

### 2.1.9 Prototype Design

The design of this robot features a quadrupedal structure, with each of its four legs equipped with a configuration of three servos, allowing for enhanced mobility and precise movement control (**Figure 10**). Weighing approximately 5 kg, the robot is built with durability in mind: its chassis is constructed from a robust 2mm thick aluminum plate, ensuring structural integrity during operation. Additionally, the body is crafted using PLA++ filament through 3D printing, combining lightweight properties with sufficient strength. The compact design yields a robot with dimensions of 27 cm in both length and width, and a height of 29 cm, making it a well-proportioned and stable unit suited for various obstacle navigation tasks.

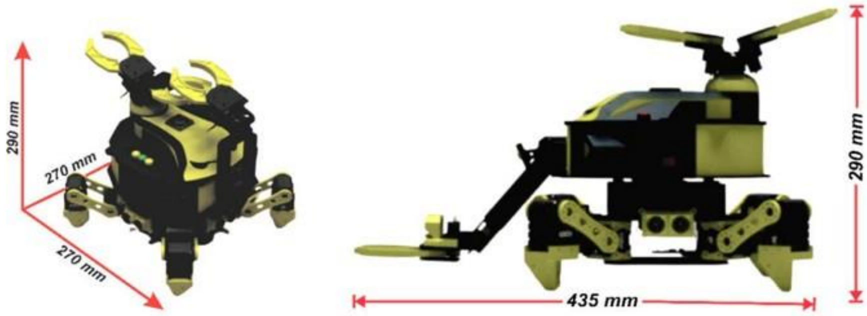


**Figure 10.** Robot Prototype

## 2.2 Method

### 2.3.1 Convolutional Neural Networks

CNN also known as Convolutional Neural Network is one of the Deep Learning for computer vision method[5]. Convolutional Neural Networks have recently gained significant attention and popularity[17]. Convolutional Neural Networks (CNNs), widely utilized in image and signal processing domains, effectively transform input image features into abstract representations, thereby enhancing accuracy in regression, classification, and segmentation tasks across a broad range of applications[18]. CNN has a multiple layer: convolution layer, pooling layer, fully connected layer, and nonlinearity layer [19]. CNN are commonly used

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

methods for image classification, an example is R-CNN (Region-Based CNN), which works by creating bounding boxes around regions of an image then performing classification on each these boxes[5]. Convolutional Layer in CNN's operates by comparing patterns within an image (**Figure 11**). Filters, typically small is spatial dimensions (width and height), are applied across the entire depth of the input image, capturing intricate details across different channels or color spaces.

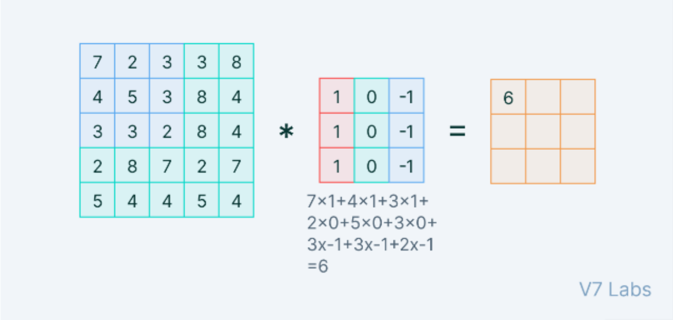


**Figure 11**. Convolutional Neural Network
Source (www.v7labs.com)

### 2.3.2 YOLO (You Only Look Once)

There are many type of pretrained model, for example ImageNet, COCO, PASCAL VOC, BDD100k, and DOTA v2.0[20]. R-CNN, SSD, and YOLO become popular because of its accuracy, and output performance for many object or classes[21]. YOLO is one of the object detection methods. In practice, YOLO tends to make more errors, though it has the capability to detect objects even in complex background regions. This feature allows YOLO to predict objects within varied backgrounds, but it may increase the likelihood of false positives or misclassifications[5]. YOLO has proven capable of classifying objects and holds great potential for building damage monitoring due to its performance in detecting and classifying objects from a distance [22]. YOLO system is dividing input picture become S x S grid. When the center point of an object falls within a grid cell, that cell takes responsibility for detecting the object. Each grid cell predicts B bounding boxes and assigns a confidence score to each box. If there is no object in a grid cell, the confidence score is set to zero. Otherwise, the confidence score equals the Intersection over Union (IoU) between the predicted bounding box and the actual object. Each bounding box prediction includes five value : x, y, w, h, and confidence rate[23]. YOLOv3 model's strength versatility and speed, also can detect the smaller object[24]. YOLOv3 has a smaller version, it called YOLOv3-Tiny. YOLOv3-Tiny have a smaller config file.

### 2.3.3 Hardware Design

**Figure 12** illustrates the hardware block system within the robot. It uses two 12V LiPo batteries allocated to power the Raspberry Pi, STM32F4, servos, Arduino, and other peripherals. These batteries have their voltage stepped down to 5V using a UBEC to ensure compatibility with the components. The Raspberry Pi is equipped with a 16MP Arducam camera for object detection. Detected object data is sent to an Arduino Nano, which then relays the data to the STM32, enabling further processing and response from the robot's systems.

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

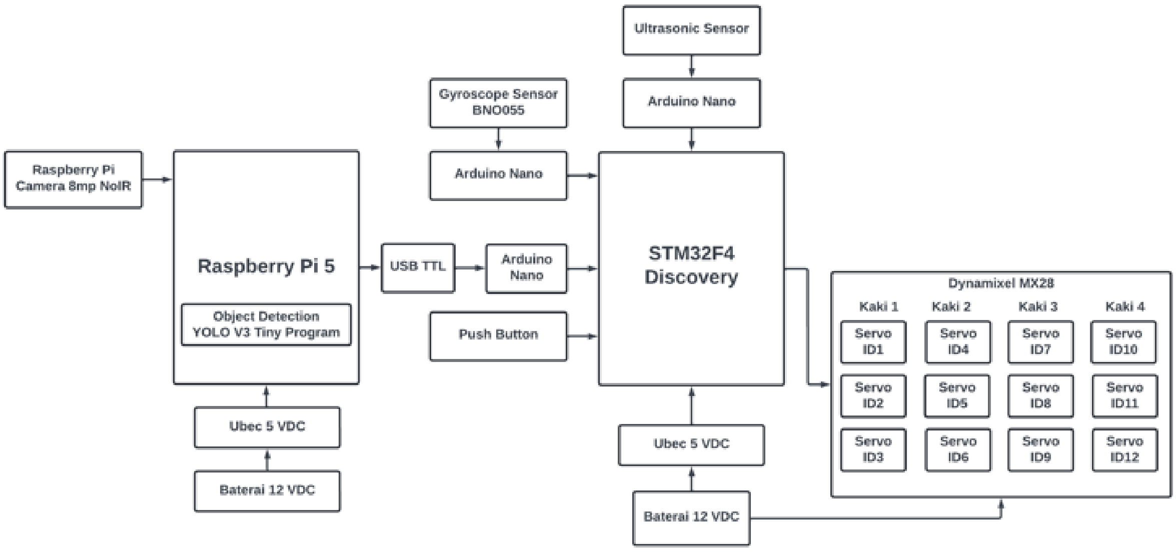


**Figure 12.** Hardware Block System

### 2.3.4 System Design

This research utilizes Python programming on the Raspberry Pi 5 through the Thonny software, where Python is applied for object detection tasks. Meanwhile, C++ is used in the CooCox CoIDE environment and Arduino software for data transmission and robot motion programming. The vision sensor relies on a Raspberry Pi camera module mounted on the Raspberry Pi 5, which sends data including X and Y coordinates, Width, Height, and Confidence Score. The Y coordinates will be parameter for robot to switch the robot movement.

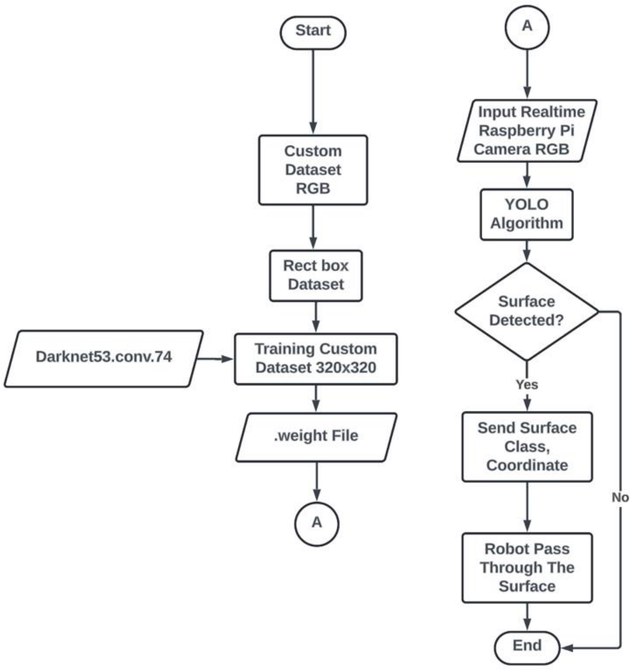


**Figure 13**. Flowchart System Design

The flowchart provided in **Figure 13** outlines the steps of the experimental process in detail. Initially, the dataset acquisition phase involves gathering a collection of 480 images

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

categorized into two distinct classes: coral and marbles. Each image undergoes an annotation process where bounding boxes are applied, generating a corresponding .txt file that contains the coordinates for each annotated box. For the training process, an input resolution of 320 x 320 is used, with Darknet53.conv.74 serving as the backbone network. This results in a .weight file that will later be essential for the object detection process. Leveraging the YOLO v3 Tiny algorithm for object detection, the robot employs a structured if-else logic to assess whether any surface previously included in the training data has been detected. Upon detection, the robot dynamically adjusts its movement mode to align with the specified settings preconfigured for that particular surface type, thereby adapting its response based on the nature of the detected obstacle

### 2.3 Evaluation

#### 2.3.1 Accuracy

This metric quantifies the accuracy of a model by calculating the fraction of correctly predicted instances (both true positives and true negatives) out of the entire dataset. It's determined by dividing the sum of true positives (TP) and true negatives (TN) by the total number of samples[25].

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} x100\% \quad (1)$$

#### 2.3.2 Precision

Precision indicates the percentage of correct positive predictions out of all instances predicted as positive by the model. It is computed by dividing the number of true positives (TP) by the total of true positives (TP) and false positives (FP).

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

#### 2.3.3 Recall

Recall, also called sensitivity or the true positive rate, represents the percentage of actual positive instances that the model correctly identifies. It's calculated by dividing the number of true positives (TP) by the total of true positives (TP) and false negatives (FN).

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

#### 2.3.4 F1-Score

The F1 Score is a metric that provides a balanced assessment of precision and recall by calculating their harmonic mean. This metric is especially helpful when aiming to achieve a trade-off between high precision and high recall, as it penalizes cases where one metric is significantly lower than the other.

$$F1 - Score = 2x \left(\frac{Pecision\ x\ Recall}{Precision\ x\ Recall}\right) \quad (4)$$

## 3. RESULTS AND DISCUSSION

DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

In this experiment, a confusion matrix is used to evaluate the performance of the trained model. The test data consists of 80 images, each containing instances of either coral or marbles. This matrix provides a breakdown of the model's predictions, allowing for a clear assessment of accuracy and error rates in detecting and classifying these two classes based on the trained dataset.

**Table 1. Confusion Matrix**

| Confusion Matrix | | |
|---|---|---|
| Actual / Predicted | Marbles | Coral |
| Marbles | 80 | 0 |
| Coral | 0 | 80 |

In the confusion matrix **Table 1**, it is shown that, out of 80 test images, the model achieved a True Positive (TP) count of 80, indicating that all target surfaces were correctly detected. Additionally, the False Negative (FN) and True Negative (TN) values are both 0, meaning there were no missed detections and no instances of the other class detected incorrectly. This result can indicate perfect accuracy across both classes in this test set.

## 4. CONCLUSION

From the research and testing conducted, the following conclusions have been drawn that the surface detection system for the quadruped robot was successfully designed and implemented using the YOLO method for object detection. This success is demonstrated by the confusion matrix results, which show perfect scores for accuracy, precision, recall, and F1-score. The YOLOv3 Tiny it's a perfect fit for low performance Mini PC, due to its light object detection method. Not just because of its light, YOLOv3 Tiny also have a good performance in detecting object, it's proven by Accuracy value is up to 100%.

## 5. ACKNOWLEDGMENT

Thank you to the Faculty of Engineering for providing the Robotics Laboratory facilities.

## 6. AUTHORS' NOTE

The authors declare that there is no conflict of interest regarding the publication of this article. The authors confirmed that the paper was free of plagiarism.

## 7. AUTHOR'S CONSTRIBUTION

Muhammad Azzaky Rizki F: Conceptualization, Methodology, Writing Original Draft, Investigation; Lilik Anifah: Formal Analysis, Writing Original Draft, Supervision; Aye Aye Mon: Data Curation, Review & Editing, Investigation.

## 8. REFERENCES


[1] R. Cahyo Prayogo, A. Triwiyatno, and D. Sumardi, “Perancangan Robot Berkaki 4 (Quadruped) Dengan Stabilization Algorithm Pada Uneven Floor Menggunakan 6-Dof Imu Berbasis Invers Kinematic,” *Transient J. Ilm. Tek. Elektro*, vol. 7, no. 2, pp. 543–551, 2018.

[2] “PEDOMAN KONTES ROBOT INDONESIA ( KRI ) PENDIDIKAN TINGGI TAHUN 2024,” 2024.

[3] Y. Zhong, R. Wang, H. Feng, and Y. Chen, “Analysis and research of quadruped robot’s legs: A comprehensive review,” *Int. J. Adv. Robot. Syst.*, vol. 16, no. 3, pp. 1–15, 2019, doi: 10.1177/1729881419844148.

[4] N. E. Khomariah and S. Huda, “Udiebot: Implementasi Inverse Kinematic pada Quadruped Robot,” *J. Ilmu Komput. dan Desain Komun. Vis.*, vol. 3, no. 1, pp. 12–21, 2018.

[5] K. Khairunnas, E. M. Yuniarno, and A. Zaini, “Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk Mobile Robot,” *J. Tek. ITS*, vol. 10, no. 1, 2021, doi: 10.12962/j23373539.v10i1.61622.

[6] J. Redmon and A. Farhadi, “YOLO v.3,” *Tech Rep.*, pp. 1–6, 2018, [Online]. Available: https://pjreddie.com/media/files/papers/YOLOv3.pdf%0Ahttps://pjreddie.com/yolo/.

[7] A. Amperawan, D. Andika, M. Anisah, S. Rasyad, and P. Handayani, *Confusion Matrix Using Yolo V3-Tiny on Quadruped Robot Based Raspberry PI 3B +* . Atlantis Press International BV, 2024. doi: 10.2991/978-94-6463-386-3_56.

[8] H. E. Saputro, N. A. S. Putro, S. Hartati, and I. Usuman, “Implementasi Sistem Kendali Keseimbangan Statis Pada Robot Quadruped Menggunakan Reinforcement Learning,” *IJEIS (Indonesian J. Electron. Instrum. Syst.*, vol. 13, no. 1, 2023.

[9] J. Billingsley and J. Billingsley, “Implementasi Inverse Kinematics Pada Pergerakan Robot Quadruped,” *Its*, pp. 1–110, 2017.

[10] M. ALFAYET, “Perancangan Sistem Pengontrolan Pengisian Air Siap Minum Berbasis Arduino (Studi Kasus Bintang Asih),” 2022.

[11] R. Ranjane, R. Chavan, and R. Mahajan, “Quadruped Robot with Speed and Direction control,” *Int. Res. J. Eng. Technol.*, vol. 08, no. 01, pp. 1965–1969, 2021.

[12] M. Haekal Maulana, A. Arifin, and R. Dikairono, “Design of Walking Movement on Bipedal Antropomorphic Robot Using Inverse Kinematics and Gait Data,” 2016.

[13] I. Kuric, “Inclination Stability Controller For Spot Micro Robot Based on Artificial Neural Networks,” no. December, 2022.

[14] Bosch Sensortec GmbH., “BNO055 Data sheet,” no. November, p. 105, 2014.

[15] Raspberry Pi, “Raspberry Pi OS,” *Raspberry Pi Ltd.*, no. October, 2023.

[16] U. S. B. C. Module, “Datasheet,” pp. 0–5.

[17] N. H. Harani, C. Prianto, and M. Hasanah, “Deteksi Objek Dan Pengenalan Karakter Plat Nomor Kendaraan Indonesia Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Python,” *J. Tek. Inform.*, vol. 11, no. 3, pp. 47–53, 2019, [Online]. Available: https://ejurnal.ulbi.ac.id/index.php/informatika/article/view/658


DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx

[18] R. Asadi *et al.*, “Process monitoring by deep neural networks in directed energy deposition: CNN-based detection, segmentation, and statistical analysis of melt pools,” *Robot. Comput. Integr. Manuf.*, vol. 87, no. December 2023, p. 102710, 2024, doi: 10.1016/j.rcim.2023.102710.

[19] Purwono, A. Ma’arif, W. Rahmaniar, H. I. K. Fathurrahman, A. Z. K. Frisky, and Q. M. U. Haq, “Understanding of Convolutional Neural Network (CNN): A Review,” *Int. J. Robot. Control Syst.*, vol. 2, no. 4, pp. 739–748, 2022, doi: 10.31763/ijrcs.v2i4.888.

[20] N. Aiman, A. Norizan, M. Razali, M. Tomari, W. Nurshazwani, and W. Zakaria, “Object Detection Using YOLO for Quadruped Robot Manipulation,” *Evol. Electr. Electron. Eng.*, vol. 4, no. 1, pp. 329–336, 2023, [Online]. Available: https://publisher.uthm.edu.my/periodicals/index.php/eeee/article/view/10778

[21] J. Zhang, Y. Zhang, J. Liu, Y. Lan, and T. Zhang, “Human figure detection in Han portrait stone images via enhanced YOLO-v5,” *Herit. Sci.*, vol. 12, no. 1, pp. 1–16, 2024, doi: 10.1186/s40494-024-01232-2.

[22] B. K. Pratama, Sri Lestanti, and Yusniarsi Primasari, “Implementasi Algoritma You Only Look Once (YOLO) untuk Mendeteksi Bahasa Isyarat SIBI,” *ProTekInfo(Pengembangan Ris. dan Obs. Tek. Inform.*, vol. 11, no. 2, pp. 7–14, 2024, doi: 10.30656/protekinfo.v11i2.9105.

[23] G. Plastiras, C. Kyrkou, and T. Theocharides, “Efficient convnet-based object detection for unmanned aerial vehicles by selective tile processing,” *ACM Int. Conf. Proceeding Ser.*, 2018, doi: 10.1145/3243394.3243692.

[24] E. Syahrudin, E. Utami, and A. D. Hartanto, “Augmentation for Accuracy Improvement of YOLOv8 in Blind Navigation,” vol. 5, no. 158, pp. 579–588, 2024.

[25] C. Niu, Y. Song, and X. Zhao, “SE-Lightweight YOLO: Higher Accuracy in YOLO Detection for Vehicle Inspection,” *Appl. Sci.*, vol. 13, no. 24, 2023, doi: 10.3390/app132413052.


DOI: http://dx.doi.org/10.xxxxx/jistel.vXiX
e- ISSN xxxx-xxxx