Depth Camera-Based Human Detection Using Yolov5

Wati P S Simanjuntak[1] and Anugerah Wibisana[2]

[12] Electrical Engineering Department, Politeknik Negeri Batam, Kepulauan Riau, Indonesia
watisimanjuntak123@gmail.com

Abstract. This research develops a depth camera-based human detection system using the YOLOv5n algorithm. The system is designed to address the challenges of object detection in various environmental and lighting conditions, as well as in real-time applications with hardware constraints. Testing results show the system achieves high accuracy in detecting distances and angles during the day, maintaining a combined error rate of approximately 2.439%. However, the system's performance declined at night with the combined error rate increasing to about 10.042%, indicating vulnerability to low lighting. Evaluation using the mean Average Precision (mAP) metric showed the model achieved a mAP value of 0.99 at an IoU threshold of 0.5 and an average mAP value of 0.9 at various thresholds from 0.5 to 0.95, indicating a high level of accuracy in object detection and classification. The integration of depth information from the RealSense camera and the real-time detection capability of YOLOv5n proves to be highly effective in human detection.

Keywords: Deep learning, YOLOv5n, Mean average precision (MAP), Human Detection.

1 Introduction

Image recognition is an important part of the field of computer vision. Digital image recognition aims to understand image information so that computers can recognize objects in images like humans[1]. Many technologies utilize computer vision for various applications such as security systems, behavior analysis, and human-computer interaction. In this context, human detection is one of the significant challenges, especially under various environmental and lighting conditions.

To address these challenges, deep learning-based object detection technologies, such as You Only Look Once (Yolo), have developed rapidly and become popular solutions. Yolo is an object detection method that is well-known for its fast and accurate ability to recognize multiple objects in a single pass. Recent versions of yolo, such as yoloV7 and yoloV8, offer significant performance improvements over previous versions, both in terms of speed and accuracy. However, this research chose to use the yoloV5n (nano) variant as it has a smaller model size and higher inference speed without sacrificing too much accuracy. YoloV5n is designed for devices with limited computing resources, making it an ideal choice for real-time applications and hardware-constrained environments[2]. According to research by Jocher et al[3], yoloV5n exhibits high efficiency

L. Lumombo et al. (eds.), *Proceedings of the 7th International Conference on Applied Engineering (ICAE 2024)*,
Advances in Engineering Research 251,
https://doi.org/10.2991/978-94-6463-620-8_12

with low latency, which is particularly useful in real-time detection applications on devices with limited computing capacity.

In addition, previous research conducted by Jing Wang et al. showed that yoloV5n has achieved a relatively stable version in performing object detection in various lighting conditions. This research shows that yoloV5 performs better than other algorithms in terms of object detection in low-light environments, achieving a mAP0.5 of 71.9% on the Mine_Exdark dataset, which is 4.4% higher[4].

While the yoloV5 has shown outstanding performance in various applications, challenges remain when applied under varying lighting conditions, especially in outdoor environments. In addition, the use of depth cameras such as Realsense can provide additional information that is useful in improving the accuracy of human detection by utilizing depth data. Depth cameras make it possible to obtain human body shape information more accurately by utilizing depth data, which can be helpful in poor lighting conditions or complex environments. Research by Zhang et al shows that the use of depth data can improve the accuracy of object detection in poor lighting conditions[5]. In addition, research by Wang et al. Examined the integration of RGB and depth data to improve the performance of human detection systems[6].

This research will combine YOLOv5n and Realsense depth-camera to address the challenges of human detection in varying lighting conditions. The proposed methodology will be evaluated through a series of experiments designed to test the performance of the system in various practical scenarios that are expected to improve the accuracy and reliability of detection under varying lighting conditions.

2 Methods

The research begins by connecting a depth camera to a computer and setting up a test environment that includes various lighting conditions. Data is collected through video capture converted to images, then preprocessed annotations, and then the YOLOv5 detection model is implemented by customizing the architecture and training it using the preprocessed dataset. Model evaluation is performed to measure the performance in detecting humans, including distance accuracy, and detection angle.

2.1 Dataset Collection

In the image data collection stage carried out at four main periods of the day, namely morning, afternoon, evening, and night, the human body shape dataset is taken using a depth camera that represents the shape of the human body as in Figure 1, which is capable of producing depth output resolutions up to 1280 × 720[7].

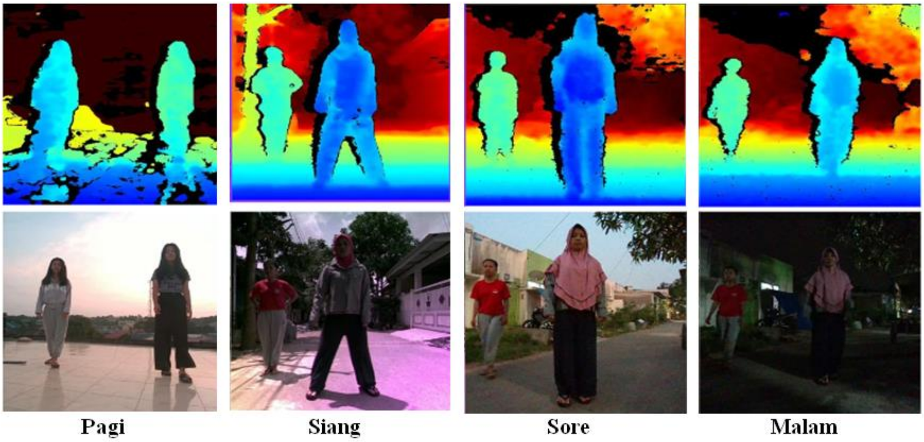


Fig. 1. Dataset Collection

This dataset is documented in detail in Table 1, which records a total dataset of 4000 images collected, with each period contributing 1000 images. Subsequently, the image dataset was sorted into 2800 images for training data, 500 images for data validation, and 400 images for testing, ensuring a balanced and representative distribution for the subsequent analysis and evaluation process[8].

Table 1. Summary Of Dataset Collection and Processing

Period	Total Images
Morning	1000
Afternoon	1000
Evening	1000
Night	1000
Total	4000

The next step is to annotate the images to mark the objects to be detected, where each object is given a rectangular box as a bounding box. In addition, each bounding box is given a class identification to indicate the type of object to be detected, in this case, the marked object is a human. It can be seen in Figure 2, where the annotation process is performed using Roboflow software[9][10]. This annotation process is important to prepare the right training data for training the object detection model.

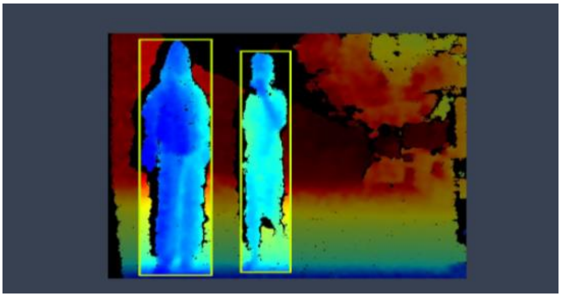


Fig. 2. Image Annotation Process

2.2 Deep Learning Architecture

In this research, object detection is performed on images using the YoloV5n object detector. YoloV5n is well known for being able to perform object detection quickly and accurately, even in different variations of size, position, overlap and has a smaller model size with a higher inference speed that does not have too much accuracy. YoloV5n is also designed for devices with limited computing resources which makes it an ideal choice for real-time applications and hardware-constrained environments. The process starts by initializing the YoloV5n detector to train the model using the YoloV5n architecture in Figure 3 where the YoloV5n architecture consists of four main components: Input, Backbone, Neck, and Head.

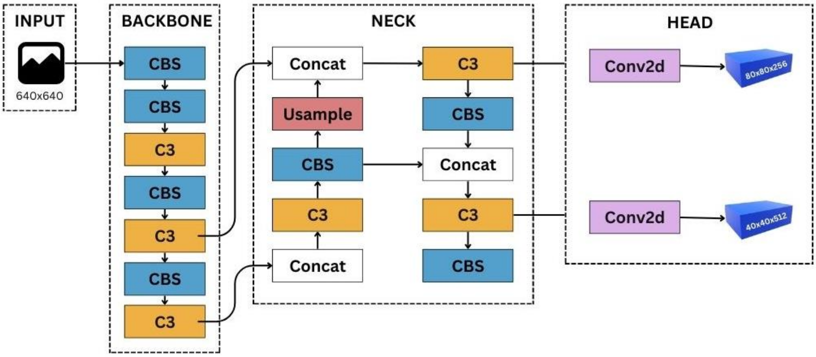


Fig. 3. Deep Learning Architecture

The input component is in charge of receiving and preparing 640x640 pixel input images for processing, often involving steps such as image adaptive scaling and mosaic data enhancement. The backbone serves as the network's main feature extractor, using a series of convolutional layers consisting of CBS (Convolutional Block with SiLU) blocks and a C3 module based on CSPNet to enhance the feature representation. The Neck component combines features from the Backbone and improves their representational power through Feature Pyramid Network (FPN) and Pixel Aggregation Network

(PAN) structures, which include operations such as concatenation and upsampling. Finally, the Head component predicts bounding boxes and class probabilities for each object in the input image using the Conv2d layer. The loss function used is CIOU Loss, and Non-Maximum Suppression (NMS) is applied to filter multi-object boxes, resulting in accurate prediction output.

2.3 YOLOv5 Architecture Evaluation Method

This research was conducted using the Mean Average Precision (mAP) evaluation method. mAP is a parameter used to evaluate object detection[10]. Precision (P) is the level of accuracy between the information requested by the user and the answer given by the system. Recall (R) is the success rate of the system in retrieving information. Accuracy (A) is an accurate calculation between predictions and actual results. Average Precision is calculated for each class and averaged to get mAP. Equations 1 – 3 are used to calculate the value of recall, precision, and accuracy.

$$P = \frac{TP}{TP + FP} \tag{1}$$

$$R = \frac{TP}{TP + FN} \tag{2}$$

$$A = \frac{TP + FN}{TP + FP + FN + TN} \tag{3}$$

True Positive (TP) is the condition when the system detects humans while the actual is human and False Positive (FP) is the condition when the system detects humans while the actual is not human. For the True Negative (TN) condition is when the system detects non-humans while the actual is not human and False Negative (FN) condition is when the system detects non-humans while the actual is human [12].

Furthermore, to get the mAP value, we must first find the IoU value. IoU (Intersection Over Union) is an evaluation metric to measure the accuracy of object detectors on a particular dataset [4]. IoU aims to determine the bounding box between the predicted bounding box and the bounding box at the time of annotation. IoU value can be calculated using Equation 4.

$$IoU = \frac{Area\ Overlap}{Area\ Union} \tag{4}$$

Furthermore, in object detection evaluation, the confidence score is used to assess how confident the system is in predicting the presence of objects in the image. The confidence score is obtained from the object detection model and indicates the confidence level of the model in the presence of the object as well as the accuracy of the prediction made. To Confidence Score using Equation 5.

$$\text{Confidence Score} = P(Object) \times IOU\frac{Truth}{Pred} \quad (5)$$

Each AP that has been calculated will be averaged to get the final value or mAP value. In addition to dividing the AP value by the number of classes, the AP is also calculated based on the threshold value. Therefore, mAP is also calculated by averaging different IoU threshold values. The equation to get the mAP value is shown in Equation 6. The error of the detection system can be calculated using Equation 7.

$$mAP = \frac{1}{N}\sum_{Recall(i)}^{N} Precision(recall(i)) \quad (6)$$

$$Error = \frac{Prediction - Actual}{Actual} \times 100\%$$
(7)

2.4 Distance Object Detection

The distance is calculated to estimate the distance between the detected object and the camera. This distance measurement process is important to know how far the object is from the camera, which is useful in various applications such as autonomous vehicle navigation, object size calculation, or security monitoring[14]. To obtain distance data from the RealSense depth camera, a few key steps need to be taken. First, install the Intel RealSense SDK and the pyrealsense2 library to enable access and interaction with data from the depth camera using Python. After that, initialize the RealSense depth camera and start streaming depth data in real-time by configuring the right pipeline. Then detecting the human object using YOLOv5n, we extract the bounding box surrounding the object. The center position of the object in the image (CenterX, CenterY) is calculated from the bounding box. Next, the distance from the camera to the object is calculated using the depth information provided by the RealSense camera. We retrieve the depth value from the center point of the object in the depth image using the depth_frame.get_distance(CenterX, CenterY) function. The actual distance is calculated in meters by using the obtained depth information using Equation 8.

$$\text{Distance} = \text{depth\_frame.get\_distance(CenterX, CenterY)} \quad (8)$$

2.5 Angle Object Detection

In the implementation of object detection using YoloV5n and the use of RealSense to measure object distance and orientation, the angle calculation method is used to determine the relative position of the object concerning the image center point in the image plane. This method provides additional information that is useful for visual object orientation analysis.

The calculated angle is the angle between a straight line connecting the image center point and the horizontal position of the detected object. This angle is calculated using the trigonometric function arctangent (atan) to determine the relative orientation of the object concerning the camera's center of view [15]. To calculate the angle θ based on the position of the object concerning the image center point using Equation 9.

$$\theta = atan2 \times \left(\frac{\text{CenterX}-\text{image\_center\_x}}{480-\text{CenterY}}\right) \times \frac{180}{\pi} \tag{9}$$

The values *CenterX* and *CenterY* are the coordinates of the center of the detected object, while *image_center _x* is the horizontal coordinate of the image center point. The value *480* is the height of the image in pixels, while *atan2* is the atan function that accepts two arguments to generate angles in all quadrants.

3 Results and Discussion

Testing was carried out using hardware with specifications in Table 2.

Table 2. Hardware Specifications

Processor	Intel(R) Core (TM) i7-10750H CPU @ 2.60GHz (Up to 5.0GHz)
RAM	16 GB DDR4 2933MHz
Storage	512 GB NVMe PCIe SSD
Graphic Card	NVIDIA GeForce GTX 1650 4GB GDDR5
Depth Camera	Intel RealSense D455

The tests carried out were the results of training data sets, object detection, calculation of object distances, and calculation angles of human objects carried out outdoors in 4 periods of the day (morning, afternoon, evening, and night) with GPU usage of 74%.

3.1 Training Dataset

Object detection testing is done in real-time using a depth RealSense D455 camera. The results of data training testing are shown in Figure 4 which is a graph when training data with 500 epochs. The training graph is obtained when the model learns the image data set after labeling.

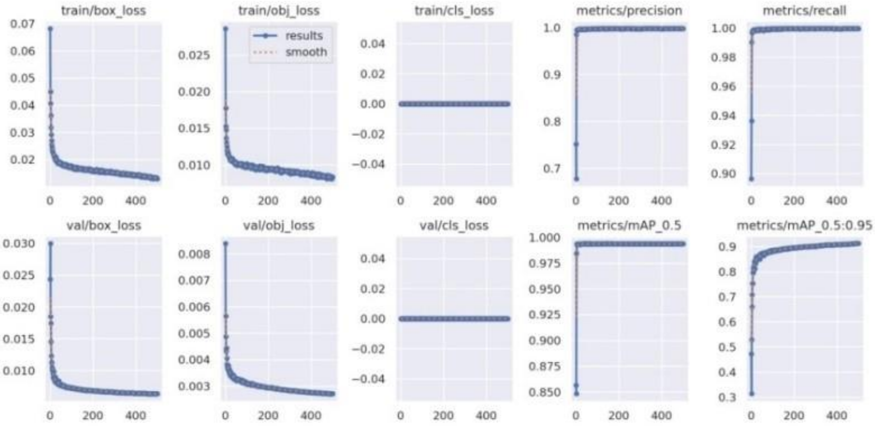


Fig. 4. Dataset Training Graph

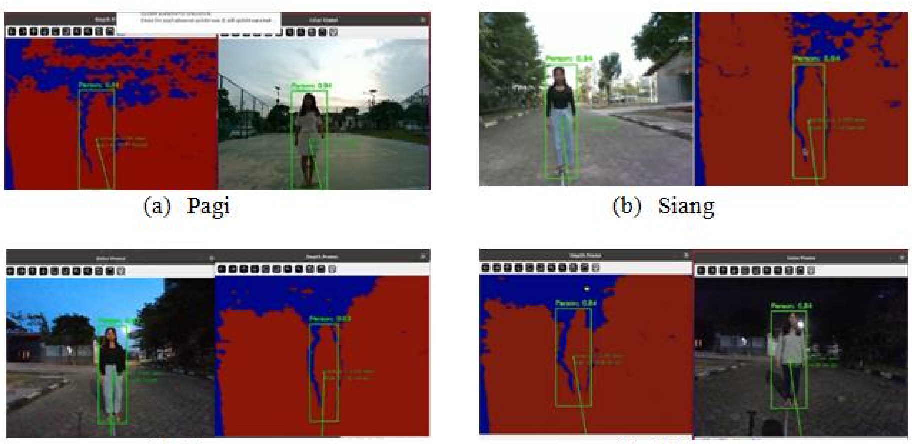
Figure 4 shows the training evaluation results of the YoloV5n model. The graphs reflect some of the key metrics used to evaluate model performance, specifically precision, recall, and mean Average Precision (mAP). Precision and recall are the main metrics generated from the confusion matrix. A high precision indicates that the model rarely gives false detections (False Positives), while a high recall indicates that the model can detect most of the objects present (True Positives). The metrics/precision and metrics/recall graphs show that both metrics reach values close to 1 indicating excellent detection performance.

Mean Average Precision (mAP) is a metric that combines precision and recall at various IoU (Intersection over Union) thresholds. The metrics/mAP_0.5 graph shows the mAP at an IoU threshold of 0.5 reaching about 0.99 while the metrics/mAP_0.5:0.95 graph shows the average mAP at various thresholds from 0.5 to 0.95 reaching about 0.9. This high mAP value indicates that the model is highly accurate in detecting and classifying objects.

Once the training process is complete, the next step is to perform testing to evaluate the performance of the model. In this section, the model is tested on human subjects to determine their distance and angle. Human detection is performed at four main periods throughout the day: morning, afternoon, evening, and night. This test aims to evaluate the accuracy of the model in detecting visible human objects in the color frames taken by the camera. At this stage, the system automatically detects and identifies humans in each frame received from the camera. Each human object detection comes with rectangular coordinate information, Confidence Score, and depth data retrieved from the RealSense sensor. Next, the system calculates the relative distance of the human object to the camera using the available depth information, as well as calculates the relative orientation angle of the object to the center of the camera image. The results of these calculations are then visualized in a color and depth frame, displaying a rectangle as a human object marker, distance information in meters, orientation angle in degrees, as well as visual elements such as lines and dots to visualize the relative position of the object.

3.2 Distance Evaluation

Distance detection measurements were tested on objects at 4 periods a day (morning, afternoon, evening, and night) from a distance of 1 to 6 meters to evaluate how accurate the distance detection generated by the system is to the actual distance in various time conditions as shown in Figure 5.



(a) Pagi

(b) Siang

(c) Sore

(d) Malam

Fig. 5. Distance Evaluation (a) Morning (b) Afternoon (c) Evening (d) Night

The test results obtained in Table 3 show the distance detection system tested can measure distance with relatively high accuracy in various periods (morning, afternoon, evening, and night). However, there is a tendency for the system to produce negative error values, meaning that it tends to measure distances that are slightly further than the actual distance, with an average error value of around -0.11935. Variations in error values were also observed between periods with performance tending to be better in the morning and afternoon than in the evening and night which may be influenced by lighting conditions or other environmental factors.

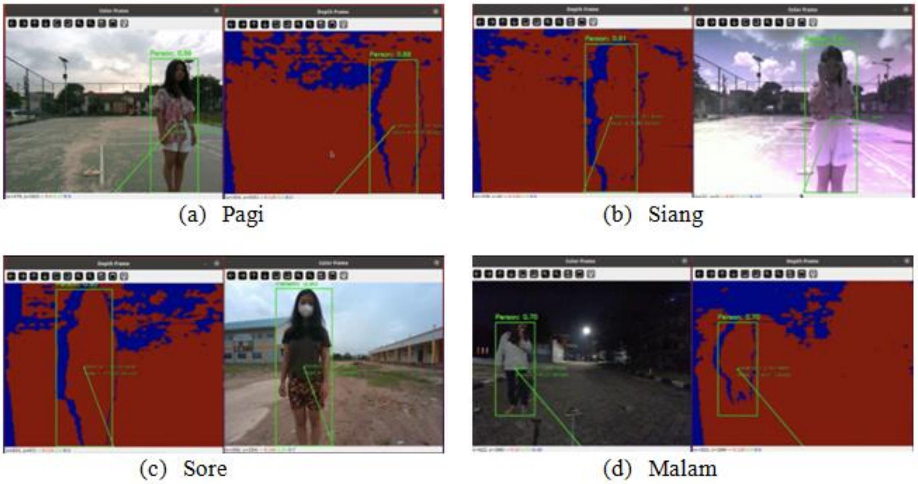
Table 3. Distance Evaluation

Periods	Actual Distance (m)	Distance Detection (m)	Error
Morning	1	1.063	-0.063
	2	2.044	-0.044
	3	3.048	-0.048
	4	4.211	-0.211
	5	5.274	-0.274
Afternoon	1	1.023	-0.023
	2	2.055	-0.055
	3	3.170	-0.17
	4	4.257	-0.257
	5	5.274	-0.274
Evening	1	1.027	-0.027
	2	2.034	-0.034
	3	3.153	-0.153
	4	4.211	-0.211
	5	5.274	-0.274
Night	1	1.000	0
	2	2.000	0

3	3.002	-0.002
4	4.040	-0.04
5	5.227	-0.227
Average		-0.11935

3.3 Angle Evaluation

The actual detection angle and the angle detected by the system on the object in 4 periods of the day (Morning, Afternoon, Evening, and Night) from 5 specified point angles (-40°, -20°, 0°, 20°, 40°) were tested to evaluate the performance of the detection model and find out how consistent and accurate the system is as shown in Fig. 6.


(a) Pagi

(b) Siang

(c) Sore

(d) Malam

Fig. 6. Angle Evaluation (a) Morning (b) Afternoon (c) Evening (d) Night

From the test results obtained in Table 4, it can be concluded that the angle detection system shows good ability in consistency and accuracy in measuring object angles at various periods (Morning, Afternoon, Evening, and Night). In general, the average error value of -0.117° indicates that the system tends to detect angles that are slightly larger than the actual angle. However, there are variations in the error value between the actual and detected angles, with some cases of larger error values especially in low lighting conditions such as at night. Further evaluation and adjustments may be required to improve the consistency and reduce the variation of error values in the angle measurement of this detection system.

Table 4. Data Angle Evaluation

Periods	Actual Angle (m)	Angle Detection (m)	Error
Morning	40	41.79	-1.79
	20	20.53	-0.53
	0	0.00	0

	-20	-20.98	0.98
	-40	-40.20	0.2
Afternoon	40	40.19	-0.19
	20	20.44	-0.44
	0	0.000	0
	-20	-20.09	0.09
	-40	-40.85	0.85
Evening	40	40.36	-0.36
	20	20.82	-0.82
	0	0.00	0
	-20	-20.56	0.56
	-40	-40.60	0.6
Night	40	40.52	-0.52
	20	21.49	-1.49
	0	0.49	-0.49
	-20	-20.70	0.7
	-40	-40.31	0.31
Average			-0.117

4 Conclusion and Future Work

This research successfully developed a human detection system using an Intel RealSense D455 depth camera and YOLOv5n algorithm. The test results show that the system has high accuracy in detecting distance and angle during the day with a combined error rate of about 2.439%. However, the system performance degrades at night with the combined error rate reaching about 10.042%, indicating that the system is more susceptible to environmental changes such as low lighting. Evaluation using the mAP metric showed that the model achieved a mAP value of 0.99 at an IoU threshold of 0.5 and an average mAP value of 0.9 at various thresholds from 0.5 to 0.95, indicating high accuracy in object detection and classification. Integrating depth information from the RealSense camera and the real-time detection capability of YOLOv5 shows good effectiveness in human detection. However, further enhancements are needed to improve performance in low-lighting conditions. Overall, the developed system shows significant potential in human detection applications, especially in bright lighting conditions, but requires further customization for consistency of detection in various lighting conditions.

Acknowledgments. This study was funded by Barelang Robotics Artificial and Intelligent Lab (BRAIL) under Delivery Robot Research.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Z. A. Fikriya, M. I. Irawan, and S. Soetrisno., "Implementasi Extreme Learning Machine untuk Pengenalan Objek Citra Digital," J. Sains dan Seni ITS, vol. 6, no. 1, 2017, doi: 10.12962/j23373520.v6i1.21754.
2. C. Fatichah and R. Dikairono, "Deteksi Objek Menggunakan Metode Yolo dan Implementasinya pada Robot Bawah Air," vol. 12, no. 3, 2023.
3. J. et Al, Ultralytics YOLO Docs. [Online]. Available: https://docs.ultralytics.com/yolov5/tutorials/architecture_description/
4. J. Wang et al., "Research on Improved YOLOv5 for Low-Light Environment Object Detection," Electron., vol. 12, no. 14, pp. 1–22, 2023, doi: 10.3390/electronics12143089.
5. R. Li, W. Si, M. Weinmann, and R. Klein, "Constraint-based optimized human skeleton extraction from single-depth camera," Sensors (Switzerland), vol. 19, no. 11, pp. 1–20, 2019, doi: 10.3390/s19112604.
6. 2∗ Yuming Fang 3 Aimin Hao 1 Hong Qin 4 Xuehao Wang 1 Shuai Li 1 Chenglizhao Chen 1, "Skema Rekombinasi Tingkat Data dan Fusi Ringan untuk Deteksi Objek Menonjol RGB-D", [Online]. Available: https://ar5iv.labs.arxiv.org/html/2009.05102
7. I. R. D455, Spesification D455, Intel Realsense. [Online]. Available: https://www.intelrealsense.com/depth-camera-d455/#:~:text=URL%3A https%3A%2F%2Fwww.intelrealsense.com%2Fdepth
8. Andre Kelana Perangin-Angin, "Interaksi Manusia Dengan Robot Menggunakan Bahasa Isyarat".
9. A. Helnawan, M. Attamimi, and A. N. Irfansyah, "Sistem Segmentasi Jalan dan Objek untuk Kendaraan Otonom Menggunakan Kamera RGB-D," J. Tek. ITS, vol. 12, no. 1, 2023, doi: 10.12962/j23373539.v12i1.110848.
10. R. Padilla, S. L. Netto, and E. A. B. Da Silva, "A Survey on Performance Metrics for Object-Detection Algorithms," Int. Conf. Syst. Signals, Image Process., vol. 2020-July, no. July, pp. 237–242, 2020, doi: 10.1109/IWSSIP48289.2020.9145130.
11. R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," Electron., vol. 10, no. 3, pp. 1–28, 2021, doi: 10.3390/electronics10030279.