Application of The Yolo Algorithm for Human Detection in a Robot Receptionist

Aulia Nurhaliza[1], Annisa Ayu Marchanda Mangkey[2], Nasya Sunia Tubuon[3*], Ade Yusupa[4], Victor Tarigan[5]

[1,2,3,4,5] Sam Ratulangi University, Manado, Indonesia
[*]E-mail : nasyatubuon03@gmail.com

Abtract

The receptionist robot is designed to enhance interaction with users in various environments, such as hotels, office buildings, and shopping centers. One of the primary challenges in the development of such robots is the fast and accurate detection of humans to enable real-time operation. This study applies the You Only Look Once (YOLO) method, specifically YOLOv5, for human object detection in receptionist robots. YOLOv5 was chosen due to its advantages in detection speed and accuracy. The research stages include system design, data acquisition, image processing, model implementation, and performance analysis using metrics such as mean Average Precision (mAP) and frames per second (FPS). Based on the results, it can be concluded that the YOLOv5 method is capable of detecting human objects with high accuracy and fast inference time. The model demonstrated optimal performance under bright lighting conditions, achieving a precision of 95% and a recall of 92%, with an inference speed of 30 FPS. Although there was a decrease in accuracy of approximately 4% under low-light conditions, the model remained consistent, achieving a precision of 88% and a recall of 85%. Additionally, the YOLOv5 model was tested under various scenarios, including changes in lighting, object distance, and crowd density within the frame. The results showed a slight decline in accuracy when objects were more than 5 meters away or when there were more than three people in a single frame, which led to an increase in false positives. Architecturally, YOLOv5 divides the input image into a 7x7 grid, where each grid cell predicts class probabilities and bounding boxes, enabling efficient and accurate detection. Therefore, YOLOv5 proves to be an effective solution for human object detection under various environmental conditions, although there remains room for improvement in specific scenarios such as low-light environments and high-density crowds.

Keywords : *Object Detection, Robot, YOLO, Artificial Intelligence*

1. INTRODUCTION

One aspect of human intelligence is the ability to recognize objects in the surrounding environment, such as human figures. Humans are able to recognize objects by using their eyes as visual sensors to capture an image, which is then recorded and stored in the brain's memory. Technological advancements today support the development of machines that can learn to recognize objects in a similar way to humans [1].

Object detection has numerous practical applications, including security surveillance, autonomous vehicles, video analysis, education, facial recognition, fire detection, pest identification in agriculture, color recognition, and many more. Methods such as YOLO (You Only Look Once) have been developed to improve the speed and efficiency of real-time object detection [2]. The main task in object detection is to find the location of objects in an image and classify the type of each object. In other words, an image is used as input, and the algorithm will produce bounding box vectors and predict the object classes [3].

YOLO is a unified detection method that uses a single neural network to predict bounding boxes and class probabilities directly from a full image in one capture. The YOLO model can process input images at up to 45 frames per second (FPS), and with a smaller version of the neural network—Fast YOLO—it can reach processing speeds of up to 155 FPS, making it the fastest algorithm when compared to other real-time detection algorithms. In contrast, non-real-time methods such as Fast R-CNN and Faster R-CNN operate at approximately 0.5 FPS and 7 FPS, respectively. This performance advantage is due to YOLO's use of the single-shot detection technique, in which the CNN is run only once for the entire object detection process. This is different from other methods, such as R-CNN and its variants, which run the CNN multiple times for each region proposal [4].

In the era of automation and artificial intelligence, receptionist robots have emerged as one of the most widely developed innovations to enhance efficiency and convenience in various settings, such as hotels, office buildings, hospitals, and shopping centers. These robots are designed to recognize human presence, greet guests, and interact with users in a more intelligent manner. A key technology that supports the functionality of receptionist robots is an object detection system—particularly human detection—which enables the robot to accurately recognize and respond to the presence of individuals. To operate optimally, the robot must be capable of detecting humans accurately and in real-time, allowing it to respond to interactions intelligently.

One of the main challenges in developing receptionist robots is designing a fast and accurate human detection system. Vision-based object detection technology has become a crucial solution, enabling robots to simulate human visual perception and recognize their surroundings. By integrating computer vision into robotic control systems, these robots can interact intelligently with their environment.

This study implements the YOLO method for human object detection in receptionist robots. The goal of this implementation is to enhance the robot's ability to recognize guests in real-time, thereby enabling more interactive and effective responses. The study also analyzes YOLO's detection performance under various lighting conditions and viewpoints to ensure the system's reliability across different environmental scenarios.

With the application of this technology, it is expected that receptionist robots can operate more optimally in supporting automation-based services, improving user experience, and accelerating the adoption of artificial intelligence technologies across various industrial sectors.

2. METHODOLOGY

In performing object detection, an algorithm capable of extracting features from images is required to recognize objects within them. One algorithm that has been proven to be both fast and accurate in previous studies is Faster R-CNN (Faster Region-based Convolutional Neural Network) [5].

The research framework in this study is structured according to the following stages:

1. Literature Review
   This stage collects data from various sources such as national journals, national and international proceedings and books related to the development of human object detection research.
2. Human Image Collection
   Researchers took and collected human image samples with various sizes, skin colors, shapes, positions obtained from various websites, namely KITTI, kaggle and others.
3. Determining Data for Training and Testing
   The human data samples that have been obtained are then grouped into two parts, namely testing and training data. The training data has a percentage of up to 70% of the total data while testing is 30%.
4. Designing a YOLO Program
   In order for the YOLO network/architecture to be smart and able to recognize human objects well, the computer is given artificial intelligence in the form of the YOLO program.
5. Conducting Training
   Before the YOLO architecture is applied, training is first carried out to change the YOLO network weights. This training will stop when the training time is met or the expected error value has been achieved.
6. Conducting Testing
   After the training is complete, the testing phase for the YOLO architecture continues to see whether the YOLO architecture can truly recognize objects according to the training data.

7. Conclusion
   From the test results, it will be known whether the research objectives have been achieved [6].

This study uses the You Only Look Once (YOLO) method to detect human objects in real-time on a receptionist robot. The stages of the method include system design, data acquisition and processing, implementation, and performance analysis.

2.1 System Design

The system consists of several main components:

1. Hardware
   Hardware is the physical components that assist the reception robot in operating.
   a. Intel RealSense Depth Camera D455
      Used to capture images or videos of the robot's surroundings. These images will then be input for the human object detection system. Object detection testing is carried out in real- time using the RealSense D455 camera [7].
   b. Jetson Nano
      The Jetson Nano is a minicomputer that utilizes a 128-core Maxwell GPU for fast processing of artificial intelligence models. Applications such as image classification, object detection, and segmentation are well suited to this device responsible for image data processing and for running object detection models (NVIDIA Jetson Nano/Raspberry Pi) for data processing [8].
   c. Speakers are used to provide voice responses, such as greetings or instructions, to guests.
   d. Motor drivers act as an interface between microcontrollers and power motors, providing voltage/current amplification while ensuring efficient PWM-based speed control and directional management. A drive motor that allows the robot to move autonomously or semi-autonomously [9].
2. Software
   The software includes algorithms and frameworks used for image processing and object detection.
   a. YOLOv5 Model
      YOLOv5 is used to detect human objects in images or videos in real-time. YOLOv5 is designed to perform object detection in a single image processing (single shot detection), making it much faster than two-stage methods such as Faster R-CNN. YOLO detects objects using a unified model where a single convolutional network predicts multiple bounding boxes and class probabilities within those boxes simultaneously. First, the YOLO system divides the input image into an S × S grid. If the center of an object falls within one of the grid cells, then that grid cell is responsible for detecting the object. Each grid cell predicts bounding boxes and a confidence score for each bounding box. The confidence score reflects how confident and accurate the model is that there is an object within that box. Each bounding box consists of 5 predictions: x, y, w, h, and confidence [10]. YOLOv5 is capable of achieving over 140 FPS (Frames Per Second) on NVIDIA Tesla V100 GPUs, making it ideal for real-time applications such as receptionist robots.

Table 1. Comparison of algorithm speeds (detect/frame) [11], [12]

Methods	UTI	UCF101	HMDB5 1	CASIA
YOLO v3	0.785	0.909	0.881	0.912
YOLO v4	0.652	0.791	0.789	0.806
YOLO v5	0.646	0.854	0.033	0.005

b. PyTorch Library
PyTorch is a library in a programming language commonly used in machine learning, deep learning and even natural language processing (NLP) developed by the Facebook AI research team (now changed to Meta AI) in 2016. With the PyTorch library, users can run and test parts of the code in real-time according to its open-source nature. Although developed using c++, PyTorch has a python-based API frontend that makes it very easy for users to use the library. PyTorch also has several features that provide many advantages, including dynamic computational graphs, different backend support, imperative style, has an intuitive and easy-to-use API and has large community support in this study. The PyTorch library is used to train and implement the YOLOv5 model, which is one of the deep learning-based object detection methods. PyTorch also supports GPU (Graphics Processing Unit) computing, which allows for accelerated data processing and model training more efficiently, so that it can produce more accurate detection in real time [13].

c. OpenCV
OpenCV (Open Computer Vision) is an API (Application Programming Interface) Library that is very familiar in Computer Vision Image Processing. Computer Vision itself is a branch of the Image Processing Science Field that allows computers to see like humans. With this vision, computers can make decisions, take action, and recognize an object [14]. The OpenCV library analyze photographs and videos to recognize artifacts, faces, and even human handwriting. When paired with many other libraries, like Numpy, a high- performance library for turning machines, achieve a good performance; that is, all services that can be performed in Numpy canalso be integrated with OpenCV.It is written based on C++ and has a C++ interface as its main interface, but it also has a less robust but still detailed older Language training. Both the latest technologies and algorithms are visible in the C++ GUI. Python, Java, and MATLAB/OCTAVE bindings are available [15].

2.2 Data Acquisition and Processing

The data acquisition and processing process is a critical stage in the human object detection system using YOLOv5 on a reception robot. This stage includes capturing images or videos from the surrounding environment, image pre-processing, and object detection using the YOLOv5 model. The following is a detailed explanation of each stage:

1. Data Acquisition (Image/Video Capture),
   Image processing is a technique for processing images, either moving or still images, into useful information. In addition to searching for information, image processing can be used in several applications such as image recognition and object detection[16].
2. Image Pre-Processing, is done after the image is taken and several pre-processing stages are carried out to ensure the image is ready to be processed by the YOLOv5 model. These stages include:
   a. Resize, resize the image with the aim of standardizing the image size [17]. The image is resized to 640x640 pixels (the default size of YOLOv5) or according to the model's needs.
   b. Color Normalization, by equalizing the color distribution of the image to match the characteristics of the data used during model training. The pixel values of the image are normalized to the range [0, 1] by dividing each pixel value by 255. Normalization can also involve subtracting the mean and dividing by the standard deviation.
   c. Noise Reduction, which removes noise or disturbances in the image that can interfere with the detection process. Techniques such as Gaussian Blur or Median Filtering can be used to reduce noise. An image that is free from noise makes it easier for the YOLOv5 model to detect objects more accurately.

3. Object Detection Using YOLOv5, after the image is processed, the YOLOv5 model is used to detect human objects. These stages include:
   a. Model Inference
   The YOLOv5 model divides the image into a grid of cells (e.g., 19x19 or 38x38 depending on the YOLOv5 version). Each grid cell predicts a bounding box and a confidence score for the detected object. The model also predicts a class probability (in this case, human) for each bounding box.
   b. Bounding Box and Confidence Score
   The bounding box is used to mark the location of the human object, while the confidence score indicates the model's level of confidence in the detection. The bounding box is represented by coordinates (x, y, width, height), where (x, y) is the center point of the box, and (width, height) is the width and height of the box. The confidence score is a value between 0 and 1 that indicates how confident the model is that the detected object is a human. If the confidence score exceeds a certain threshold (for example, 0.5), the object is considered valid.
   c. Non-Maximum Suppression (NMS)
   Non-Maximum Suppression (NMS) is an indispensable post-processing step in object detection. With the continuous optimization of network models, NMS has become the “last mile” to improve object detection efficiency [18]. NMS compares bounding boxes with high overlap and selects the bounding box with the highest confidence score. Overlapping bounding boxes are removed to avoid double detection of the same object.
   d. Detection Output
   If a human object is detected, the robot can provide a voice response (e.g., greeting) or move the motor to approach the guest. The detection result can also be displayed visually on the robot's screen (if available) to provide feedback to the user.

2.3 Implementation on Guest Reception Robot

(a) Object Detection System Integration The camera installed on the robot is tasked with capturing real-time images or videos from the surrounding environment. These images are then sent to a microprocessor, such as NVIDIA Jetson Nano or Raspberry Pi, for further processing. The YOLOv5 model that has been trained to detect human objects is integrated into the robot system. When the camera captures an image, the YOLOv5 model will process the image to detect the presence of humans. The detection results in the form of a bounding box and confidence score are used to determine the location and accuracy of the detection. If a human object is detected with a confidence score that exceeds a certain threshold, the system will send a signal to the robot's response and navigation module.

(b) Robot Response Once a human object is detected, the robot provides an interactive response in the form of a voice greeting or a text message on the screen. The voice response is generated through a speaker integrated into the system, while the text message is displayed on an LCD screen or monitor attached to the robot. These responses are designed to providea friendly and enjoyable experience for the guest. For example, the robot can say “Welcome” or display a similar message on the screen. In addition, the robot can also adjust the response based on additional information, such as the time of day (morning, afternoon, evening) or the guest’s identity if the system is equipped with facial recognition technology.

(c) Robot Navigation Towards Guests To move towards a detected guest, the robot uses bounding box data generated by the YOLOv5 model. This data provides information about the relative position of the human to the robot. If the human is out of range, the robot will activate its navigation system to move closer to the guest. Environmental Optimization and Adjustment, the implementation of this system also considers various environmental conditions, such as different lighting or the presence of other objects around the guest. To overcome these challenges, the YOLOv5 model can be optimized by fine-tuning using datasets that cover various environmental scenarios. In addition, the navigation system can be adjusted according to the robot's movement speed and sensor sensitivity to ensure consistent performance. For example, in low-light conditions, the camera can be equipped with a night vision feature, or the YOLOv5 model can be retrained with a dataset that includes images in low-light conditions.

2.4 Performance Analysis

Evaluation is carried out based on:

a. Detection accuracy: Detection accuracy is one of the main metrics used to evaluate the performance of the YOLOv5 model. Accuracy is measured by comparing the detection results provided by the model with ground truth data, which is data that has been manually labeled and is considered a reference truth. A common metric used to measure accuracy is mean Average Precision (mAP), which combines precision (detection accuracy) and recall (the model's ability to find all objects that should be detected). The higher the mAP value, the better the model's performance in detecting human objects. For example, if the model achieves an mAP of 80%, it means that the model is able to detect 80% of human objects with a high level of accuracy. This evaluation was conducted on a test dataset that includes various scenarios, such as humans in standing, sitting, or moving positions.

b. Inference time: Inference time measures the speed at which the YOLOv5 model processes each frame of an image or video. This speed is critical for real-time applications such as reception robots, as they must be able to respond quickly to human presence. Inference time is measured in frames per second (FPS), which indicates how many frames the model can process in one second. For example, if the model achieves 30 FPS, it means that the model can process 30 frames per second, which is sufficient for real-time applications. Inference time is affected by factors such as hardware capacity (e.g., the GPU used) and model complexity. Testing was conducted on various devices, such as the NVIDIA Jetson Nano and Raspberry Pi, to ensure that the system can run smoothly on devices with limited resources.

c. Error rate (False Positive & False Negative): Error rate is an important aspect in evaluating the performance of an object detection system. False positive occurs when the model detects an object as a human when it is not, while false negative occurs when the model fails to detect a human when it should have been detected. Both types of errors can affect the effectiveness of the robot in interacting with guests. For example, false positives can cause the robot to respond unnecessarily, while false negatives can cause the robot to not respond to the presence of a guest. To reduce the error rate, the YOLOv5 model can be optimized by improving the quality of the training dataset, adjusting the confidence score threshold, or using data augmentation techniques to enrich the variety of images in the dataset.

3. RESULTS AND DISCUSSION

The results show that the YOLOv5 method is able to detect human objects with a high level of accuracy and fast inference time. This model works optimally in good lighting conditions, and only experiences a slight decrease in accuracy in low lighting.

Table 2. YOLO Accuracy Test Results under various conditions

Environmental Conditions	*Precision*	*Recall*	FPS
Environmental Conditions	*Precision*	*Recall*	FPS
Dim Lighting	88%	85%	28 FPS
Many People in the Frame	90%	87%	27 FPS

In addition to accuracy and speed, the YOLO model was tested in various scenarios to determine its strengths and weaknesses:

1. Changing lighting: accuracy decreases by about 4% in low lighting conditions compared to bright lighting.
2. Distance between object and camera: At distances greater than 5 meters, the model starts having difficulty in correctly detecting humans.
3. Density of people in a frame: if there are more than 3 people in a frame, false positives tend to increase.

YOLO divides the input image into a 7 x 7 grid. Each grid cell corresponds to 2 bounding boxes. Each grid cell predicts a set of class probabilities (including 20 classes) regardless of the number of boxes. And each bounding box consists of 5 predictions: x, y, w, h, and confidence. The coordinates (x, y) represent the center of the box relative to the grid cell boundaries. (w, h) represent the width and height of the box relative to the entire image.

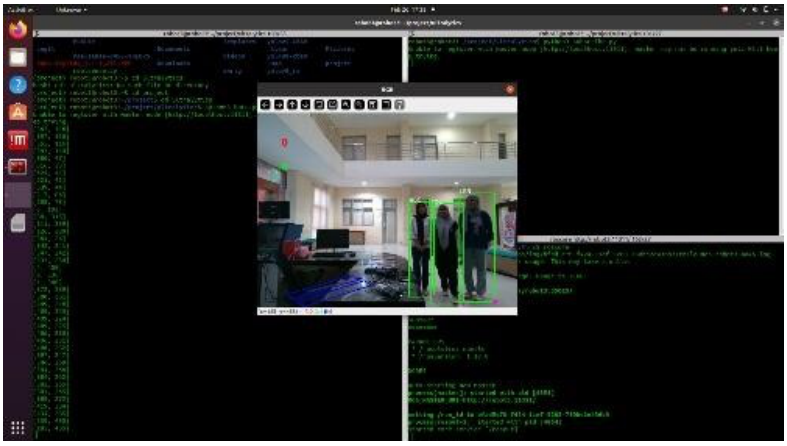

Figure 1. Three Person Object Detection

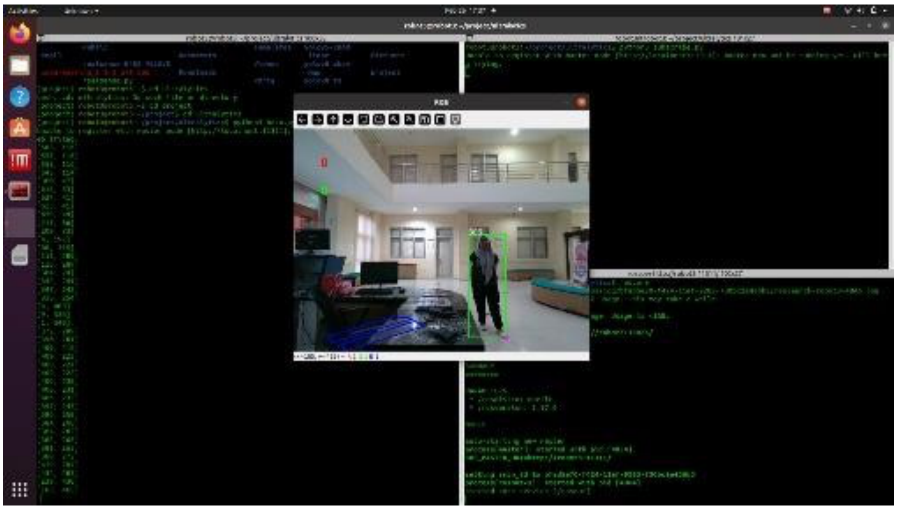

Figure 2. Single Person Object Detection

Object detection for a single person in an image or video generally has a higher accuracy rate due to the absence of interference from other objects around. However, when the system is faced with the detection of three people in one frame, the YOLOv5 algorithm can still recognize each individual well, although there is a possibility of a slight decrease in accuracy due to object overlap and position variation. In addition, the processing time tends to increase as the number of objects to be detected in one frame increases.

ROS (Robot Operating System) has now become the most widely used robot software development platform among academics. This popularity is supported by its completeness of devices, distributed design, and excellent scalability[18]. In Figure 3, an Ultralytics and ROS-based system is implemented in a Linux (Ubuntu) environment to support the object detection process using YOLOv5.

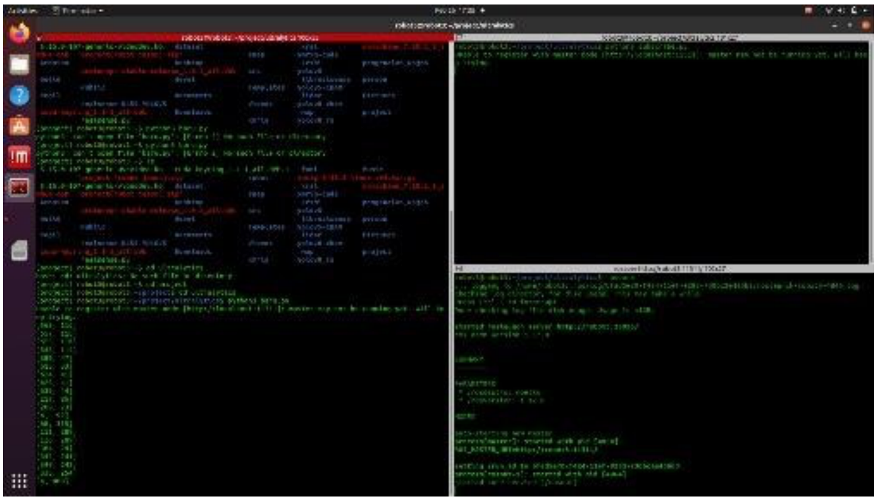

Figure 3. Object Detection Program View

On the left terminal, various commands are executed to run the Python script (baru.py and subscribe.py). However, the system experiences an error message indicating that the master node has not been able to connect to the address http://localhost:11311. This message indicates that ROS Master has not been run before the execution of the object detection script begins. ROS Master acts as the core of communication in the ROS system, facilitating data exchange between nodes, including image processing, object detection, and decision-making nodes. To overcome this problem, the user runs the roscore command on the lower terminal. After ROS Master is active, the object detection script can run properly and display the coordinates of the detection results in the form of X and Y value pairs. These values most likely represent the bounding box of the detected object in a frame.

In this context, YOLOv5 acts as a Convolutional Neural Network (CNN)-based object detection model that is able to identify and provide object position information in images in real-time. The main advantage of YOLOv5 lies in its speed and accuracy in detecting objects, thus supporting the efficiency of ROS-based systems. This is very important in applications that require fast response, such as facial recognition, autonomous robot navigation, or automatic environmental monitoring. The integration between YOLOv5 and ROS in this study shows how the ROS distributed framework can be utilized to optimize the object detection pipeline, from data acquisition, processing, to decision making. However, it is important to ensure that all system components, including the ROS Master, have been properly initialized before the execution of other nodes to avoid inter-node communication failures.

4. CONCLUSION

Based on the research results, it can be concluded that the YOLOv5 method is able to detect human objects with a high level of accuracy and fast inference time. This model shows optimal performance in bright lighting conditions with a precision of 95% and a recall of 92%, and an inference speed of 30 FPS. Although there is a decrease in accuracy of about 4% in low lighting conditions, the model remains consistent with a precision of 88% and a recall of 85%.

In addition, the YOLOv5 model was also tested in various scenarios, including changes in lighting, object distance, and the density of people in the frame. The results showed that the model experienced a slight decrease in accuracy when the object was more than 5 meters away or when there were more than 3 people in a frame, resulting in an increase in false positives.

Architecturally, YOLOv5 divides the input image into a 7x7 grid, where each grid cell predicts class probabilities and bounding boxes, enabling efficient and accurate detection. Thus, YOLOv5 is an effective solution for human object detection in a variety of environmental conditions, although there is still room for improvement in certain scenarios such as low lighting and high density.

REFERENCES

[1] Setiyadi, E. Utami, and D. Ariatmanto, “Analisa kemampuan algoritma YOLOv8 dalam deteksi objek manusia dengan metode modifikasi arsitektur,” *J-SAKTI (Jurnal Sains Komputer Dan Informatika)*, pp. 891–901, 2023.

[2] D. N. Alfarizi, R. A. Pangestu, D. Aditya, M. A. Setiawan, and P. Rosyani, “). Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis,” *J. Artif. Intel. Dan Sist. Penunjang Keputusan*, pp. 54–63, 2023.

[3] H. Lin, J. D. Deng, D. Albers, and F. W. Siebert, “Helmet use detection of tracked motorcycles using CNN-based multi-task learning. ,” 2020.

[4] I. M. D. Maleh, R. Teguh, A. S. Sahay, S. Okta, and M. P. Pratama, “Implementasi Algoritma You Only Look Once (YOLO) Untuk Object Detection Sarang Orang Utan Di Taman Nasional Sebangau,” *Jurnal Informatika*, vol. 10, no. 1, pp. 19–27, Mar. 2023, doi: 10.31294/inf.v10i1.13922.

[5] H. Herdianto, H. Hafni, D. Nasution, and S. Ramadhan, “Implementasi Metode Yolo pada Deteksi Objek Manusia,” *METHOMIKA Jurnal Manajemen Informatika dan Komputerisasi Akuntansi*, vol. 8, no. 2, pp. 234–240, Oct. 2024, doi: 10.46880/jmika.Vol8No2.pp234-240.

[6] F. Indaryanto, A. Nugroho, and A. F. Suni, “Aplikasi Penghitung Jarak dan Jumlah Orang Berbasis YOLO Sebagai Protokol Kesehatan Covid-19. Edu Komputika Journal, 8 (1), 31â€“38,” 2021.

[7] W. P. S. Simanjuntak and A. Wibisana, “Depth Camera-Based Human Detection Using Yolov5,” 2024, pp. 150–162. doi: 10.2991/978-94-6463-620-8_12.

[8] W. A. Nugraha, “Deteksi Jilbab Secara Realtime Dengan You Only Look Once (YOLO) Menggunakan Jetson Nano,” Universitas Islam Sultan Agung Semarang, 2024.

[9] K. Khairunnas, E. M. Yuniarno, and A. Zaini, “Pembuatan modul deteksi objek manusia menggunakan metode yolo untuk mobile robot,” *Jurnal Teknik ITS*, vol. 10, no. 1, pp. A50–A55, 2021.

[10] G. Jocher *et al.*, “ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation,” *Zenodo*, 2022.

[11] X. Zhou, J. Yi, G. Xie, Y. Jia, G. Xu, and M. Sun, “Human detection algorithm based on improved YOLO v4,” *Information Technology and Control*, vol. 51, no. 3, pp. 485–498, 2022.

[12] R. Mekacahyani, “Klasifikasi Penyakit Kulit Dermatitis Atopik dan Psoriasis Menggunakan Algoritma Convolutional Neural Network dengan Model Arsitektur Resnet-50,” Universitas Islam Sultan Agung Semarang, 2024.

[13] F. A. Putra, O. Opitasari, and N. W. Parwati, “Sistem Absensi dengan Metode Face Recognition Menggunakan Opencv Berbasis Web di TK Az-Zahra,” in *Seminar Nasional Riset dan Inovasi Teknologi (SEMNAS RISTEK)*, 2025, pp. 375–381.

[14] H. Adusumalli, D. Kalyani, R. K. Sri, M. Pratapteja, and P. P. Rao, “Face mask detection using opencv,” in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, IEEE, 2021, pp. 1304–1309.

[15] A. N. Sugandi and B. Hartono, “Implementasi pengolahan citra pada quadcopter untuk deteksi manusia menggunakan algoritma yolo,” in *Prosiding Industrial Research Workshop and National Seminar*, 2022, pp. 183–188.

[16] K.-S. Si *et al.*, “Accelerating Non-Maximum Suppression: A Graph Theory Perspective,” *arXiv preprint arXiv:2409.20520*, 2024.

[17] M. Y. A. Thoriq, I. A. Siradjuddin, and K. E. Permana, “Deteksi Wajah Manusia Berbasis One Stage Detector Menggunakan Metode You Only Look Once (Yolo),” *Jurnal Teknoinfo*, vol. 17, no. 1, pp. 66–73, 2023.

[18] J. Zeng and J. Fu, “Basketball robot object detection and distance measurement based on ROS and IBN-YOLOv5s algorithms,” *PLoS One*, vol. 19, no. 11, p. e0310494, 2024.