

Depth Realsense Camera-Based Human Detection Using Yolov5

Wati Putri Suside Simanjuntak¹, Anugerah Wibisana²

Department of Electrical Engineering, Politeknik Negeri Batam, Batam, Kepulauan Riau, Indonesia

Article Info

Article history:

Received month dd, yyyy

Revised month dd, yyyy

Accepted month dd, yyyy

Keywords:

Deep learning

YOLOv5

Mean average precision (MAP)

Deteksi Objek

Intel RealSense D455

ABSTRACT

Penelitian ini mengembangkan sistem deteksi manusia berbasis kamera depth Intel RealSense D455 menggunakan algoritma YOLOv5n. Sistem ini dirancang untuk mengatasi tantangan deteksi objek dalam berbagai kondisi lingkungan dan pencahayaan, serta untuk aplikasi real-time dengan keterbatasan perangkat keras. Hasil pengujian menunjukkan bahwa sistem ini memiliki akurasi tinggi dalam mendeteksi jarak dan sudut pada siang hari, dengan tingkat kesalahan gabungan sekitar 2.439%. Namun, kinerja sistem menurun pada malam hari dengan tingkat kesalahan gabungan mencapai sekitar 10.042%, menunjukkan kerentanan terhadap perubahan pencahayaan rendah. Evaluasi menggunakan metrik mean Average Precision (mAP) menunjukkan model mencapai nilai mAP sebesar 0.99 pada threshold IoU 0.5 dan nilai rata-rata mAP sebesar 0.9 pada berbagai threshold dari 0.5 hingga 0.95, menandakan akurasi tinggi dalam deteksi dan klasifikasi objek. Integrasi informasi kedalaman dari kamera RealSense dan kemampuan deteksi real-time dari YOLOv5n menunjukkan efektivitas yang baik dalam deteksi manusia. Namun, diperlukan peningkatan lebih lanjut untuk meningkatkan kinerja dalam kondisi pencahayaan rendah. Secara keseluruhan, sistem yang dikembangkan menunjukkan potensi signifikan dalam aplikasi deteksi manusia terutama dalam kondisi pencahayaan terang namun memerlukan penyesuaian lebih lanjut untuk konsistensi deteksi dalam berbagai kondisi pencahayaan.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Anugerah Wibisana

Department of Electrical Engineering, Politeknik Negeri Batam

Jl. Ahmad Yani, Kel. Teluk Tering, Kec. Batam Kota, Kota Batam, Kepulauan Riau 29461, Indonesia

Email: wibisana@gmail.com

1. INTRODUCTION

Pengenalan citra merupakan bagian penting dalam bidang komputer *vision*. Pengenalan citra digital bertujuan untuk memahami informasi citra sehingga komputer dapat mengenali objek pada citra selayaknya manusia[1]. Banyak teknologi memanfaatkan komputer *vision* untuk berbagai aplikasi seperti sistem keamanan, analisis perilaku, hingga interaksi manusia-komputer. Dalam konteks ini, deteksi manusia menjadi salah satu tantangan signifikan, terutama dalam berbagai kondisi lingkungan dan pencahayaan.

Untuk mengatasi tantangan ini, teknologi deteksi objek berbasis *deep learning*, seperti *You Only Look Once* (Yolo), telah berkembang pesat dan menjadi solusi populer. Yolo merupakan metode deteksi objek yang terkenal karena kemampuannya yang cepat dan akurat dalam mengenali berbagai objek dalam satu kali pemrosesan.

Versi terbaru dari yolo, seperti yoloV7 dan yoloV8, menawarkan peningkatan performa yang signifikan dibandingkan versi-versi sebelumnya, baik dalam hal kecepatan maupun *Accuracy*. Namun, dalam penelitian ini, kami memilih untuk menggunakan varian yoloV5n (*nano*) karena memiliki ukuran model yang lebih kecil dan kecepatan inferensi yang lebih tinggi tanpa mengorbankan terlalu banyak *Accuracy*. YoloV5n dirancang untuk perangkat dengan keterbatasan sumber daya komputasi, menjadikannya pilihan ideal untuk aplikasi *real-time* dan lingkungan yang memiliki keterbatasan perangkat keras[2]. Menurut penelitian oleh Jocher et al[3], yoloV5n menunjukkan efisiensi tinggi dengan latensi rendah, yang sangat berguna dalam aplikasi deteksi *real-time* pada perangkat dengan kapasitas komputasi terbatas.

Selain itu, penelitian sebelumnya yang dilakukan oleh Jing Wang dkk. menunjukkan bahwa yoloV5n telah mencapai versi yang relatif stabil dalam melakukan pendeteksian objek di berbagai kondisi pencahayaan. Penelitian ini menunjukkan bahwa yoloV5 memberikan performa yang lebih baik dibandingkan dengan algoritma lainnya dalam hal deteksi objek di lingkungan dengan pencahayaan rendah, mencapai mAP_{0,5} sebesar 71,9% pada dataset Mine_Extdark, yang 4,4% lebih tinggi[4].

Namun, meskipun yoloV5 telah menunjukkan kinerja yang luar biasa dalam berbagai aplikasi, tantangan tetap ada ketika diterapkan dalam kondisi pencahayaan yang bervariasi, terutama di lingkungan luar ruangan. Selain itu, penggunaan kamera *depth* seperti *Realsense* dapat memberikan informasi tambahan yang berguna dalam meningkatkan *Accuracy* deteksi manusia dengan memanfaatkan data kedalaman. Kamera *depth* memungkinkan untuk mendapatkan informasi bentuk tubuh manusia secara lebih akurat dengan memanfaatkan data kedalaman, yang dapat membantu dalam kondisi pencahayaan yang buruk atau lingkungan yang kompleks. Penelitian oleh Zhang et al Menunjukkan bahwa penggunaan data kedalaman dapat meningkatkan *Accuracy* deteksi objek dalam kondisi pencahayaan yang buruk[5]. Selain itu, penelitian oleh Wang et al. Mengkaji integrasi data RGB dan *depth* untuk meningkatkan kinerja sistem deteksi manusia[6].

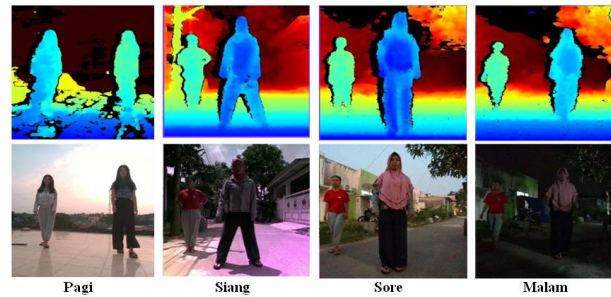
Penelitian ini akan menggabungkan YOLOv5n dan kamera *depth Realsense* untuk mengatasi tantangan deteksi manusia dalam berbagai kondisi pencahayaan. Metodologi yang diusulkan akan dievaluasi melalui serangkaian eksperimen yang dirancang untuk menguji kinerja sistem dalam berbagai skenario praktis yang diharapkan dapat meningkatkan *Accuracy* dan keandalan deteksi dalam kondisi pencahayaan yang bervariasi.

2. METHOD

Penelitian dimulai dengan menghubungkan kamera *depth* ke komputer dan menyiapkan lingkungan pengujian yang mencakup berbagai kondisi pencahayaan. Data dikumpulkan melalui pengambilan video yang di *convert* ke gambar, lalu diproses *anotasi*, kemudian model deteksi YOLOv5 diimplementasikan dengan menyesuaikan arsitektur dan melatihnya menggunakan dataset yang telah diproses. Evaluasi model dilakukan untuk mengukur kinerja dalam mendeteksi manusia, termasuk *Accuracy* jarak, dan sudut deteksi.

2.1. Dataset Collection

Pada tahap pengambilan data gambar dilakukan pada empat periode utama dalam sehari, yaitu pagi, siang, sore, dan malam hari, pengambilan *dataset* bentuk tubuh manusia ini dilakukan dengan menggunakan *depth camera* yang merepresentasikan bentuk tubuh manusia seperti pada gambar 1, yang mampu menghasilkan resolusi *output* kedalaman hingga 1280×720 [7].



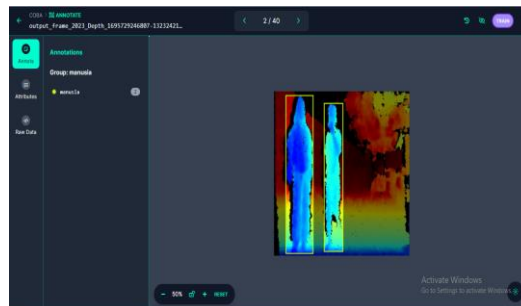
Gambar 1. Dataset Collection

Dataset ini terdokumentasi secara rinci dalam Table 1 Yang mencatat total dataset sebanyak 4000 gambar yang dikumpulkan, dengan setiap periode menyumbang 1000 gambar Selanjutnya, *dataset* gambar tersebut dipilah menjadi 2800 gambar untuk pelatihan data, 500 gambar untuk validasi data, dan 400 gambar untuk pengujian, memastikan adanya distribusi yang seimbang dan representatif untuk proses analisis dan evaluasi selanjutnya[8].

Tabel 1. Summary of Dataset Collection and Processing

Tahap Pengambilan Data	Periode Waktu	Jumlah Gambar
<i>Dataset Collection</i>	Pagi	1000
	Siang	1000
	Sore	1000
	Malam	1000
Total		4000
Pengolahan Dataset	<i>Training Set</i>	2800
	<i>Validation Set</i>	500
	<i>Testing Set</i>	400

Tahap selanjutnya adalah melakukan *anotasi* pada gambar-gambar tersebut. *Anotasi* dilakukan untuk menandai objek-objek yang akan dideteksi, di mana setiap objek diberikan kotak persegi panjang sebagai kotak pembatas. Selain itu, setiap kotak pembatas diberikan identifikasi kelas untuk menunjukkan jenis objek yang akan dideteksi, dalam hal ini, objek yang ditandai adalah manusia. Dapat dilihat pada gambar 2, yang dimana proses *anotasi* ini dilakukan menggunakan perangkat lunak *Roboflow*[9][10]. Proses *anotasi* ini penting untuk mempersiapkan data latih yang tepat untuk pelatihan model deteksi objek.

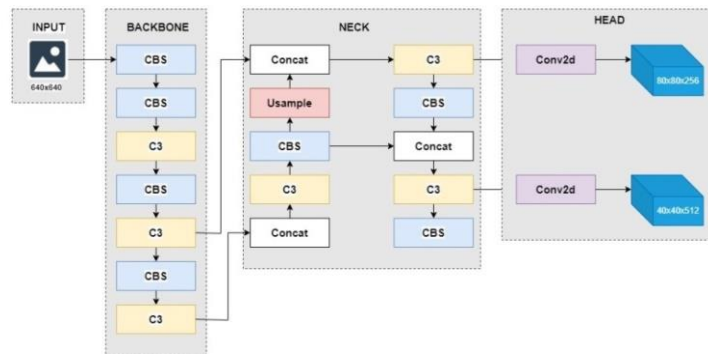


Gambar 2. Proses anotasi gambar di roboflow

2.2. Arsitektur Deep Learning

Paper's should be the fewest possible that accurately describe ... (First Author)

Pada penelitian ini, kami melakukan pendeteksian objek pada gambar menggunakan detektor objek YoloV5n. YoloV5n terkenal mampu melakukan deteksi objek dengan cepat dan akurat, bahkan dalam variasi ukuran, posisi, yang tumpang tindih yang berbeda dan memiliki ukuran model yang lebih kecil dengan kecepatan inferensi yang lebih tinggi yang tidak terlalu banyak *Accuracy*. YoloV5n juga dirancang untuk perangkat dengan keterbatasan sumber daya komputasi, yang menjadikannya pilihan ideal untuk aplikasi *real-time* dan lingkungan yang memiliki keterbatasan perangkat keras. Proses dimulai dengan menginisialisasi detektor YoloV5n untuk melatih model menggunakan arsitektur YoloV5n pada gambar 3 yang dimana arsitektur YoloV5n terdiri dari empat komponen utama: *Input*, *Backbone*, *Neck*, dan *Head*.



Gambar 3. Arsitektur Deep learning

Sumber: [J.Wang et al., 2023]

Komponen *Input* bertugas menerima dan menyiapkan gambar input berukuran 640x640 piksel untuk diproses, sering kali melibatkan langkah-langkah seperti penskalaan adaptif gambar dan peningkatan data mosaik. *Backbone* berfungsi sebagai ekstraktor fitur utama jaringan, menggunakan serangkaian lapisan *konvolusi* yang terdiri dari blok CBS (*Convolutional Block with SiLU*) dan modul C3 yang didasarkan pada CSPNet untuk meningkatkan representasi fitur. Komponen *Neck* menggabungkan fitur dari *Backbone* dan meningkatkan kekuatan representasionalnya melalui struktur *Feature Pyramid Network* (FPN) dan *Pixel Aggregation Network* (PAN), yang mencakup operasi seperti *concatenation* dan *upsampling*. Terakhir, komponen *Head* memprediksi kotak pembatas (*bounding boxes*) dan *probabilitas* kelas untuk setiap objek dalam gambar *input* menggunakan lapisan Conv2d. Fungsi kerugian yang digunakan adalah CIOU Loss, dan *Non-Maximum Suppression* (NMS) diterapkan untuk menyaring kotak multi-objek, menghasilkan *output* prediksi yang akurat.

2.3. YOLOv5 Architecture Evaluation Method

Penelitian ini dilakukan menggunakan metode evaluasi *Mean Average Precision* (mAP). mAP merupakan parameter yang digunakan untuk mengevaluasi pendeteksian objek[10]. *Precision* (P) adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan sistem. *Recall* (R) adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. *Accuracy* (A) adalah perhitungan akurat antara prediksi dan hasil sesungguhnya. *Average Precision* dihitung untuk setiap kelas dan dirata-ratakan untuk mendapatkan mAP. perlu diketahui nilai *recall*(1), *precision*(2), dan *accuracy*(3). Berikut merupakan presentasi prediksi positif yang benar dan hasil prediksi, yang dapat dihitung dengan menggunakan persamaan berikut.

$$P = \frac{TP}{TP + FP} \quad (1)$$

$$R = \frac{TP}{TP + FN} \quad (2)$$

$$A = \frac{TP + FN}{TP + FP + FN + TN} \quad (3)[11]$$

Rumus perhitungan (3) Kondisi dikatakan *True Positive (TP)* ketika objek mendeteksi manusia sedangkan aktualnya manusia, selanjutnya kondisi *False Positive (FP)* ketika objek mendeteksi manusia sedangkan aktualnya bukan manusia, kemudian *True Negative (TN)* ketika objek mendeteksi bukan manusia sedangkan aktualnya bukan manusia, dan *False Negative (FN)* ketika objek mendeteksi bukan manusia sedangkan aktualnya manusia[12].

Selanjutnya untuk mendapatkan nilai mAP, harus menemukan nilai IoU terlebih dahulu. *IoU (Intersection Over Union)* merupakan metrik evaluasi untuk mengukur keakuratan detektor objek pada dataset tertentu (4). IoU bertujuan Untuk menentukan kotak pembatas antara kotak pembatas hasil prediksi atau kotak pembatas pada saat anotasi. Dapat dilihat dibawah ini merupakan persamaan untuk mendapatkan nilai IoU.

$$IoU = \frac{Area\ Overlap}{Area\ Union} \quad (4)[13]$$

Selanjutnya, dalam evaluasi deteksi objek, *confidence score* digunakan untuk menilai seberapa yakin sistem dalam memprediksi keberadaan objek dalam gambar. *Confidence score* biasanya diperoleh dari model deteksi objek dan menunjukkan tingkat keyakinan model terhadap keberadaan objek serta akurasi dari prediksi yang dibuat (5).

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

Setiap AP yang sudah dihitung akan dirata-ratakan agar mendapatkan nilai akhir atau nilai mAP selain membagi nilai AP dengan jumlah *class*, AP juga dihitung berdasarkan nilai *threshold*. Oleh karena itu mAP juga dihitung dengan rata-rata dari nilai *threshold* IoU yang berbeda. Dibawah ini merupakan persamaan untuk mendapatkan nilai mAP (6).

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

Kemudian untuk menghitung kesalahan atau error pada sistem pendeteksian menggunakan persamaan berikut (7).

$$Error = \frac{Prediksi - Aktual}{Aktual} \times 100\% \quad (7)$$

2.4. Distance Object Detection

Jarak dihitung untuk mengestimasi jarak antara objek yang dideteksi dan kamera. Proses pengukuran jarak ini penting untuk mengetahui seberapa jauh objek tersebut dari kamera, yang berguna dalam berbagai aplikasi seperti navigasi kendaraan otonom, perhitungan ukuran objek, atau pemantauan keamanan[14].

Untuk mendapatkan data jarak dari kamera depth RealSense, beberapa langkah utama perlu dilakukan. Pertama, instal Intel *RealSense* SDK dan *library pyrealsense2* untuk memungkinkan akses dan interaksi dengan data dari kamera depth menggunakan *Python*. Setelah itu, inisialisasi kamera *depth RealSense* dan mulai streaming data kedalaman secara *real-time* dengan mengkonfigurasi *pipeline* yang tepat. Kemudian mendeteksi objek manusia menggunakan YOLOv5n, kami *mengekstrak bounding box* yang mengelilingi objek tersebut. Posisi tengah objek dalam gambar (TengahX, TengahY) dihitung dari *bounding box*. Selanjutnya, jarak dari kamera ke objek dihitung menggunakan informasi kedalaman yang disediakan oleh kamera *RealSense*. Kami mengambil nilai kedalaman dari titik tengah objek pada gambar kedalaman dengan menggunakan fungsi *depth_frame.get_distance* (TengahX, TengahY). Jarak sebenarnya dihitung dalam meter dengan menggunakan informasi kedalaman yang diperoleh. untuk menghitung jarak menggunakan rumus perhitungan (8)

$$\text{Jarak} = \text{depth_frame.get_distance}(\text{Tengahy}, \text{Tengahx}) \quad (8)$$

2.5. Angle Object Detection

Dalam implementasi deteksi objek menggunakan YoloV5n dan penggunaan *RealSense* untuk mengukur jarak dan orientasi objek, kami menggunakan metode perhitungan sudut untuk menentukan posisi relatif objek terhadap titik tengah citra (*center point*) dalam bidang gambar. Metode ini memberikan informasi tambahan yang berguna untuk analisis orientasi objek secara visual.

Sudut yang dihitung adalah sudut antara garis lurus yang menghubungkan titik tengah citra dengan posisi *horizontal* objek terdeteksi. Sudut ini dihitung menggunakan fungsi trigonometri *arctangent* (*atan*) untuk menentukan orientasi relatif objek terhadap pusat pandang kamera[15].

Rumus (9) untuk menghitung sudut (θ) berdasarkan posisi objek terhadap titik tengah citra. (*TengahX* dan *TengahY*) koordinat pusat objek terdeteksi, (*image_center_x*) koordinat *horizontal* titik tengah citra, (480) adalah tinggi citra dalam piksel, (*atan2*) fungsi *atan* yang menerima dua argumen untuk menghasilkan sudut dalam semua kuadran, (*Angel*) sudut pandang *horizontal* yang diukur dalam derajat, *lebar_tampak* adalah lebar tampak objek yang diukur dalam piksel, dan *panjang_fokus* adalah panjang fokus kamera.

$$\theta = \text{atan2} \times \left(\frac{\text{TengahX} - \text{image_center_x}}{480 - \text{Tengahy}} \right) \times \frac{180}{\pi} \quad (9)$$

3. RESULTS AND DISCUSSION

Pengujian dilakukan menggunakan *hardware* dengan spesifikasi pada Tabel 2

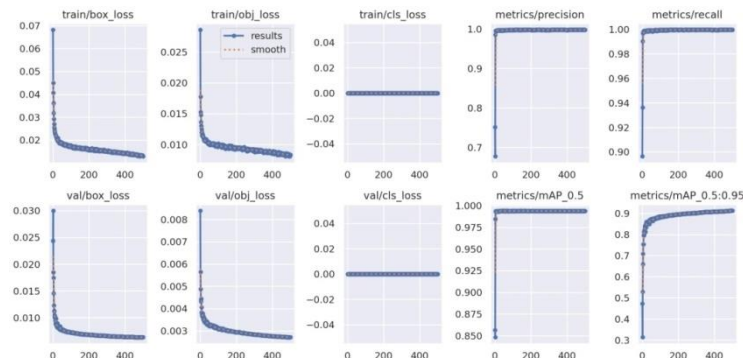
Tabel 2. Spesifikasi Hardware yang Digunakan

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

Pengujian yang dilakukan adalah hasil pelatihan kumpulan data, pendeteksian objek, penghitungan jarak objek dan sudut objek manusia yang dilakukan di luar ruangan dalam 4 periode dalam sehari (pagi, siang, sore, malam) dengan penggunaan GPU 74%..

3.1. Training Data

Pengujian pendeteksian objek dilakukan secara realtime dengan menggunakan kamera *depth ralsense D455*. Hasil pengujian pelatihan data ditunjukkan pada Gambar 4 yang merupakan grafik pada saat melakukan pelatihan data dengan 500 *epoch*. Grafik pelatihan didapatkan ketika model mempelajari kumpulan data gambar setelah dilakukan label bisa dilihat pada grafik berikut.



Gambar 4. Grafik Pelatihan Data

Gambar di atas menunjukkan hasil evaluasi pelatihan model YoloV5n, Grafik-grafik tersebut mencerminkan beberapa metrik kunci yang digunakan untuk mengevaluasi kinerja model, khususnya *precision*, *recall*, dan *mean Average Precision* (mAP). *Precision* dan *recall* adalah metrik utama yang dihasilkan dari *confusion matrix*. *Precision* yang tinggi menunjukkan bahwa model jarang memberikan deteksi palsu (False Positives), sedangkan *recall* yang tinggi menunjukkan bahwa model mampu mendeteksi sebagian besar objek yang ada (*True Positives*). Grafik *metrics/precision* dan *metrics/recall* menunjukkan bahwa kedua metrik mencapai nilai mendekati 1, yang menandakan kinerja deteksi yang sangat baik.

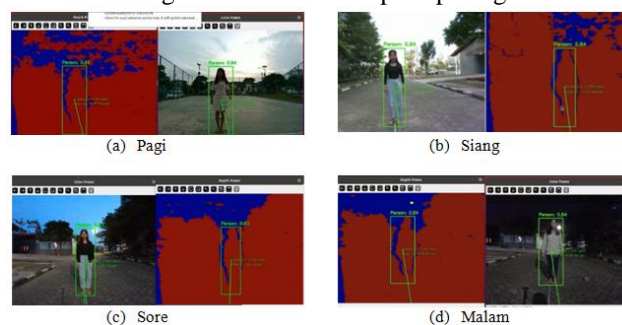
Mean Average Precision (mAP) adalah metrik yang menggabungkan *precision* dan *recall* di berbagai *threshold* IoU (*Intersection over Union*). Grafik *metrics/mAP_0.5* menunjukkan mAP pada *threshold* IoU 0,5 mencapai sekitar 0,99, sementara grafik *metrics/mAP_0.5:0.95* menunjukkan mAP rata-rata pada berbagai *threshold* dari 0,5 hingga 0,95 mencapai sekitar 0,9. Nilai mAP yang tinggi ini menunjukkan bahwa model sangat akurat dalam mendeteksi dan mengklasifikasikan objek.

3.2.1. Results of Evaluation Method

Setelah proses pelatihan selesai, langkah selanjutnya adalah melakukan pengujian untuk mengevaluasi kinerja model. Pada bagian ini, model diuji pada subjek manusia untuk menentukan jarak dan sudut mereka. Deteksi manusia dilakukan pada empat periode utama sepanjang hari: pagi, siang, sore, dan malam hari. Pengujian ini bertujuan untuk mengevaluasi keakuratan model dalam mendeteksi objek manusia yang terlihat dalam bingkai warna yang diambil oleh kamera. Pada tahap ini, sistem secara otomatis mendeteksi dan mengidentifikasi manusia dalam setiap bingkai yang diterima dari kamera. Setiap deteksi objek manusia dilengkapi dengan informasi koordinat persegi panjang, Confidence Score, dan data kedalaman yang diambil dari sensor *RealSense*. Selanjutnya, sistem menghitung jarak relatif dari objek manusia terhadap kamera dengan menggunakan informasi kedalaman yang tersedia, serta menghitung sudut orientasi relatif objek terhadap pusat gambar kamera. Hasil perhitungan ini kemudian divisualisasikan dalam bingkai warna dan kedalaman, menampilkan persegi panjang sebagai penanda objek manusia, informasi jarak dalam meter, sudut orientasi dalam derajat, serta elemen visual seperti garis dan titik untuk memvisualisasikan posisi relatif objek secara jelas.

3.2.2.1 Evaluation Method Distance

Pada penelitian ini, dilakukan pengujian pengukuran jarak deteksi pada objek pada 4 periode dalam sehari (Pagi, Siang, Sore, Malam) dari jarak 1 hingga 6 meter untuk mengevaluasi seberapa akurat deteksi jarak yang dihasilkan sistem ke jarak aktual dalam berbagai kondisi waktu seperti pada gambar 5.



Gambar 5. Pengujian *Distance*

Dari hasil pengujian yang didapatkan pada Tabel 3, menunjukkan bahwa sistem deteksi jarak yang diuji mampu mengukur jarak dengan akurasi relatif tinggi dalam berbagai periode waktu (Pagi, Siang, Sore, Malam). Meskipun demikian, terdapat kecenderungan sistem untuk menghasilkan nilai eror negatif, artinya cenderung mengukur jarak yang sedikit lebih jauh dari jarak sebenarnya, dengan rata-rata nilai eror sekitar -0.11935. Variasi nilai eror juga terlihat antara periode waktu, dengan performa yang cenderung lebih baik pada pagi dan siang hari dibandingkan sore dan malam, yang mungkin dipengaruhi oleh kondisi pencahayaan atau faktor lingkungan lainnya. Evaluasi

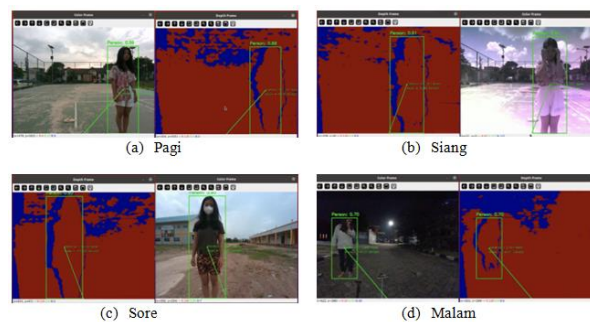
lebih lanjut terhadap kalibrasi sistem dan pengelolaan faktor-faktor lingkungan dapat membantu meningkatkan konsistensi deteksi jarak.

Tabel 3. Pengujian *Distance*

Periode Waktu	Jarak Actual (m)	Jarak Terdeteksi (m)	Nilai Error
Pagi	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
Siang	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
Sore	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
Malam	1	1.000	0
	2	2.000	0
	3	3.002	-0.002
	4	4.040	-0.04
	5	5.227	-0.227
Rata-rata			-0.11935

3.2.2.1 Evaluation Method Angle

Pada penelitian ini, dilakukan pengujian pengukuran sudut deteksi aktual dan sudut yang terdeteksi pada sistem pada objek dalam 4 periode dalam sehari (Pagi, Siang, Sore, Malam) dari 5 titik yang ditentukan (sudut -40° , -20° , 0° , 20° , 40°) untuk mengevaluasi kinerja model deteksi dan mengetahui seberapa konsisten dan akurat sistem ini seperti pada Gambar 6.



Gambar 6. Pengujian *Angle*

Dari hasil pengujian yang didapatkan pada Tabel 4 dapat di simpulkan bahwa sistem deteksi sudut menunjukkan kemampuan yang baik dalam konsistensi dan akurasi dalam mengukur sudut objek pada berbagai periode waktu (Pagi, Siang, Sore, Malam). Secara umum, nilai eror rata-rata sebesar -0.117° menunjukkan bahwa sistem cenderung mendeteksi sudut yang sedikit lebih besar dari sudut sebenarnya. Meskipun demikian, terdapat variasi nilai eror antara sudut aktual dan terdeteksi, dengan beberapa kasus nilai eror lebih besar terutama pada kondisi pencahayaan yang rendah seperti pada malam hari. Evaluasi lebih lanjut dan penyesuaian mungkin diperlukan untuk memperbaiki konsistensi dan mengurangi variasi nilai eror dalam pengukuran sudut sistem deteksi ini.

Tabel 4. Pengujian *Angle*

Periode Watu	Sudut Actual ^o	Sudut Terdeteksi ^o	Nilai Eror
Pagi	40	41.79	-1.79
	20	20.53	-0.53
	0	0.00	0
	-20	-20.98	0.98
	-40	-40.20	0.2
Siang	40	40.19	-0.19
	20	20.44	-0.44
	0	0.000	0
	-20	-20.09	0.09
	-40	-40.85	0.85
Sore	40	40.36	-0.36
	20	20.82	-0.82
	0	0.00	0
	-20	-20.56	0.56
	-40	-40.60	0.6
Malam	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
Rata-rata			-0.117

4. CONCLUSION

Penelitian ini berhasil mengembangkan sistem deteksi manusia menggunakan kamera depth Intel RealSense D455 dan algoritma YOLOv5n. Hasil pengujian menunjukkan bahwa sistem memiliki akurasi tinggi dalam mendeteksi jarak dan sudut pada siang hari dengan tingkat kesalahan gabungan sekitar 2.439%. Namun, kinerja sistem menurun pada malam hari dengan tingkat kesalahan gabungan mencapai sekitar 10.042%, menunjukkan bahwa sistem lebih rentan terhadap perubahan lingkungan seperti pencahayaan rendah. Evaluasi menggunakan metrik mAP menunjukkan model mencapai nilai mAP sebesar 0.99 pada threshold IoU 0.5 dan nilai rata-rata mAP sebesar 0.9 pada berbagai threshold dari 0.5 hingga 0.95, menandakan akurasi tinggi dalam deteksi dan klasifikasi objek. Integrasi informasi kedalaman dari kamera RealSense dan kemampuan deteksi real-time dari YOLOv5 menunjukkan efektivitas yang baik dalam deteksi manusia. Namun, diperlukan peningkatan lebih lanjut untuk meningkatkan kinerja dalam kondisi pencahayaan rendah. Secara keseluruhan, sistem yang dikembangkan menunjukkan potensi signifikan dalam aplikasi deteksi manusia, terutama dalam kondisi pencahayaan terang, namun memerlukan penyesuaian lebih lanjut untuk konsistensi deteksi dalam berbagai kondisi pencahayaan.

Paper's should be the fewest possible that accurately describe ... (First Author)

5. ACKNOWLEDGEMENTS

Puji syukur saya panjatkan kehadirat Tuhan yang Maha Esa, yang telah memberikan kesempatan dan kelancaran untuk menyelesaikan Tugas Akhir ini. Tugas akhir ini adalah salah satu syarat mahasiswa untuk memperoleh gelar sarjana pada Program Studi Teknik Robotika. Dengan selesainya Tugas akhir ini saya mengucapkan terimakasih kepada dosen pembimbing saya atas arahan dan bimbingannya dalam pembuatan jurnal dan penyelesaian tugas akhir ini, serta keluarga dan teman-teman saya atas dukungan dalam pembuatan penelitian ini

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. Fatchah 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://arxiv.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%3Ahttps%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] D. Iskandar Mulyana and M. A. Rofik, "Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5," *J. Pendidik. Tambusai*, vol. 6, no. 3, pp. 13971–13982, 2022, doi: 10.31004/jptam.v6i3.4825.
- [12] I. D. S. Tarigan, Roni Habibi, and Rd. Nuraini Siti Fatonah, "Evaluasi Algoritma Klasifikasi Machine Learning Kategori Nilai Akhir Tunjangan Kinerja Pegawai," *J. Sist. Cerdas*, vol. 6, no. 3, pp. 251–261, 2023, doi: 10.37396/jsc.v6i3.246.
- [13] 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.
- [14] E. M. Y. dan A. Z. Khairunnas, "Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk Mobile Robot".
- [15] M. D. Anggraini, K. Kusriani, and H. Al Fatta, "Social Distancing Detection Finding Optimal Angle With Yolo V3 Deep Learning Method," *J. Tek. Inform.*, vol. 3, no. 5, pp. 1449–1455, 2022, doi: 10.20884/1.jutif.2022.3.5.390.