

Identifikasi Warna Bola dan SILO Menggunakan Metode You Only Look Once Pada Robot Pengambil Bola

Revanza Akmal Pradipta¹, Ryan Yudha Adhitya², Muhammad Khoirul Hasin³, Ii' Munadhif⁴, Mohammad Abu Jami'in⁵, Zindhu Maulana Ahmad Putra⁶

e-mail: revanzaakmal@student.ppns.ac.id, ryanyudhaadhitya@ppns.ac.id, khoirul.hasin@ppns.ac.id, iimunadhif@ppns.ac.id, jammy@ppns.ac.id, zindhu@ppns.ac.id.

^{1,2,3,4,5,6}Jurusan Teknik Otomasi, Politeknik Perkapalan Negeri Surabaya, Jalan Teknik Kimia Kampus ITS, Surabaya, Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 30 April 2024

Direvisi 26 Mei 2024

Diterbitkan 31 Mei 2024

Kata kunci:

YOLOv5

Robocon

Deteksi Objek

Keywords:

YOLOv5

Robocon

Deteksi Objek

ABSTRAK

Robots are a modern technology that is experiencing very rapid growth. Robots also have several types, one type of robot is Robot Vision. Robot Vision uses image processing and computer vision in carrying out its tasks, including recognizing objects and controlling robot movements. Taking the concept from the ABU (Asia Pacific Broadcasting Union) robot contest in 2024 where the robot is required to be able to pick up balls with red and blue colors which are then inserted into a place called SILO. The detection method used in this study uses the YOLO (You Only Look Once) method with the YOLOv5 version. The reason for choosing the YOLOv5 version is because of its ability to be more accurate than previous versions and lighter to run than newer versions. From the results of this study, it was found that the confidence value for each object was quite high, which was 0.90. In addition, the precision value against recall for all classes gets a value of 0.991 or 99.1%. For the ball and SILO classes, the precision to recall value is 0.993 or 99.3%. So it can be concluded that this method has good value in terms of detection and accuracy

ABSTRACT

Robot adalah teknologi modern yang mengalami pertumbuhan yang sangat cepat. Robot juga memiliki beberapa jenis, salah satu jenis robot adalah Robot Vision. Robot Vision menggunakan pemrosesan gambar dan visi komputer dalam melaksanakan tugasnya, termasuk dalam mengenali objek dan mengontrol gerakan robot. Mengambil konsep dari kontes robot ABU (Asia Pacific Broadcasting Union) pada tahun 2024 dimana robot diharuskan dapat mengambil bola dengan warna merah dan biru yang kemudian di masukkan ke dalam sebuah tempat yang bernama SILO. Metode deteksi yang digunakan dalam penelitian ini menggunakan metode YOLO (You Only Look Once) dengan versi YOLOv5. Alasan pemilihan versi YOLOv5 adalah karena kemampuannya yang lebih akurat dibandingkan versi sebelumnya dan lebih ringan dijalankan dibandingkan versi yang lebih baru. Dari hasil penelitian tersebut diadapat nilai confidence pada setiap objek cukup tinggi, yaitu diangka 0.90. Selain itu nilai precision terhadap recall untuk seluruh kelas mendapatkan nilai 0.991 atau 99.1%. Untuk kelas bola dan SILO mendapatkan nilai precision terhadap recall sebesar 0.993 atau 99.3%. Sehingga dapat disimpulkan bahwa metode ini memiliki nilai yang baik dari segi deteksi maupun keakuratan.



Penulis Korespondensi:

Revanza Akmal Pradipta,
Jurusan Teknik Otomasi,
Politeknik Perkapalan Negeri Surabaya,
Jalan Teknik Kimia Kampus ITS, Surabaya, Indonesia
Email: revanzaakmal@student.ppns.ac.id
Nomor HP/WA aktif: +6285853898190

1. PENDAHULUAN

Robot adalah teknologi modern yang mengalami pertumbuhan yang sangat cepat. Negara-negara maju sedang bersaing untuk menciptakan berbagai jenis robot canggih dengan tujuan tertentu. Pada saat ini, robot sudah banyak menggunakan sistem otomatis atau sistem otonom. Sistem otonom memiliki kemampuan untuk membuat keputusan untuk mencapai tujuan yang telah ditetapkan tanpa panduan atau kontrol manusia atau lainnya [1]. Dengan kemajuan teknologi, berbagai robot dibuat dengan keahlian khusus. Ini sangat penting untuk industri modern [2]. Robot juga memiliki beberapa jenis, salah satu jenis robot adalah *Robot Vision*. Teknologi *Robot Vision* menggunakan pemrosesan gambar dan visi komputer dalam melaksanakan tugasnya, termasuk dalam mengenali objek dan mengontrol gerakan robot[3]. Pendeteksian objek dari *Robot Vision* menjadi tantangan tersendiri dalam *Object Detection* agar dapat mendeteksi objek dengan tepat.

Penelitian ini mengambil inspirasi dari konsep kontes robot ABU (Asia Pacific Broadcasting Union) pada tahun 2024. Robot diharuskan dapat mengambil bola dengan warna merah dan biru yang kemudian di masukkan ke dalam sebuah tempat yang bernama *SILLO* [4]. Dari permasalahan tersebut dibutuhkan sebuah metode yang dapat mendeteksi bola dan *SILLO* dengan akurat. Terdapat beberapa metode yang dapat digunakan dalam pengembangan aplikasi pendeteksian objek, dan salah satunya melibatkan penerapan teknik *Convolutional Neural Network* (CNN) dan pendekatan *You Only Look Once* (YOLO) [5]. Pendeteksian objek pada gambar atau citra melalui ekstraksi fitur dan klasifikasi dilakukan dengan menggunakan *Convolutional Neural Network* (CNN). Namun efektivitas CNN dinilai kurang optimal. Untuk mengatasi hal ini, pada tahun 2016 muncul metode baru bernama *You Only Look Once* (YOLO) oleh Redmon. YOLO merupakan perkembangan lebih lanjut dari CNN yang memungkinkan deteksi objek secara lebih efisien[6]. Perkembangan YOLO sangat cepat, dengan versi pertama diluncurkan pada tahun 2016 dan mencapai versi YOLOv7 pada bulan Juni 2022 [7]. You Only Look Once (YOLO) adalah metode deteksi objek yang cukup populer dalam pengolahan citra dan visi komputer. Algoritma YOLO (You Only Look Once) adalah salah satu teknik dalam Deep Learning dan merupakan bagian dari Machine Learning yang saat ini sangat populer untuk mendeteksi wajah atau objek [8]. YOLO bekerja seperti otak manusia, yaitu dengan menggunakan *Jaringan saraf tiruan* (JST)[9]. Karena menggunakan konsep tersebut, sehingga dibutuhkan sebuah dataset yang kemudian akan dilakukan training pada dataset tersebut[10].

Penelitian sebelumnya yang berjudul "Perancangan Sistem Deteksi Objek Bola Dan Gawang Pada Robot Sepakbola Menggunakan Metode Darknet YOLO " Dari penelitian tersebut, hasil pengujian YOLO V3 yang sudah di training untuk dapat mendeteksi bola menunjukkan kinerja yang baik dalam mendeteksi objek bola dan gawang pada robot sepakbola humanoid ROBOTIS OP3, dengan nilai akurasi sebesar 0.97, presisi sebesar 0.99, recall sebesar 0.96, dan F1-score sebesar 0.97 [8].

Kemudian pada penelitian yang berjudul "Implementasi Sistem Object Tracking Untuk Mendeteksi Dua Objek Berbasis Deep Learning" dapat disimpulkan bahwa hasil training YOLOv5 yang sudah ditraining dengan dataset bolamenghasilkan pembacaan yang bagus dengan *Mean Average Precision* (mAP) mencapai 0.985, presisi sebesar 0.971, dan nilai recall mencapai 0.981, menunjukkan kemampuan yang baik dalam mendeteksi objek bola dan robot kawan pada sistem tracking yang dirancang[11].

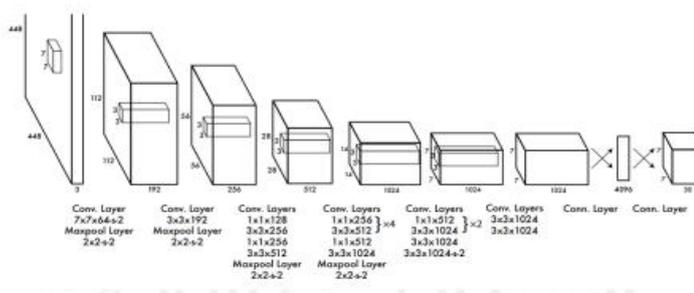
Dari berbagai referensi yang telah dikaji, penelitian ini memfokuskan pada pendeteksian objek yang berbeda dengan penelitian sebelumnya, yaitu bola dan *SILLO* sesuai dengan aturan yang ditetapkan. Kemudian untuk



penggunaan metode deteksi yang digunakan dalam penelitian ini menggunakan metode You Only Look Once (YOLO) dengan versi YOLOv5. Penggunaan versi YOLOv5 adalah karena kemampuannya yang lebih akurat dibandingkan versi sebelumnya dan lebih ringan dijalankan dibandingkan versi yang lebih baru. Selain itu kelebihan lain dari YOLOv5 adalah karena proses yang cukup sederhana dalam penggunaannya dan dikembangkan secara opensource oleh Ultralytics[12].

2. METODE PENELITIAN

Pada penelitian ini digunakan metode YOLOv5 sebagai *Object Detection* agar dapat memprediksi bola dan *SILO*. Tugas utama dari *Object Detection* adalah mencari sebuah objek pada sebuah gambar yang kemudian mengklasifikasikan object tersebut sesuai dengan hasil training [13]. YOLOv5 akan otomatis menganotasi mana benda yang terdeteksi dan memberi Bounding Box pada object yang memiliki nilai tinggi. YOLOv5 berbeda dengan versi sebelumnya. Versi ini adalah model dari keluarga YOLO yang ditulis menggunakan arsitektur *PyTorch*. Kelebihan penggunaan *PyTorch* adalah arsitekturnya yang lebih kecil dan mudah digunakan, yang membuat implementasi algoritma YOLOv5 lebih cepat dan lebih ringan dari pada versi sebelumnya. Cara kerja dari algoritma YOLO adalah dengan membagi objek menjadi beberapa bagian kecil atau membagi menjadi 9 grid[14].



Gambar 1: Cara kerja deteksi metode YOLO

Setelah membagi gambar menjadi grid 9, YOLOv5 melakukan penugasan objek ke setiap grid berdasarkan posisi tengah objek. Setiap grid kemudian membuat prediksi terkait objek yang ada di dalamnya, termasuk bounding box, kelas objek, dan skor kepercayaan atau confidence. Langkah selanjutnya adalah melakukan penekanan *non-maksimum* (NMS) untuk mengurangi tumpang tindih antara bounding box yang diprediksi. YOLOv5 menggabungkan hasil deteksi dari semua grid menjadi satu set hasil deteksi akhir, termasuk informasi tentang kelas objek, lokasi bounding box, dan skor kepercayaan. Metode ini memungkinkan YOLOv5 untuk secara efisien mendeteksi objek dalam gambar dengan memanfaatkan paralelisme dari grid-grid yang terbagi.

Dalam penelitian ini, berfokus pada deteksi bola berwarna merah dan biru yang memiliki bentuk yang sama. Selain bola, dalam penelitian ini juga robot diharuskan untuk dapat mendeteksi *SILO*. Aplikasi deteksi object YOLO V5 ini mencakup beberapa proses, diantaranya pengumpulan dataset, pembagian dataset, anotasi dataset, training dataset, dan pengujian hasil training.

2.1 Pengumpulan Data

Pengambilan data berupa foto terhadap objek bola dan *SILO* pada penelitian ini diambil menggunakan webcam yang kemudian dilakukan pengambilan gambar dari setiap sudut. Selain pengambilan gambar sendiri, terdapat juga pengambilan dari sumber internet yang memiliki bentuk yang sama. Dengan banyaknya pengambilan data maka akan semakin akurat model yang dihasilkan.

2.2 Pembagian Dataset

Pada tahap ini data foto yang sudah dikumpulkan kemudian dilakukan proses pembagian data menjadi 3 bagian utama, yaitu data *Train*, data *Validation*, dan data *Test*. Pembagian jumlah foto tersebut berdasarkan urgensi



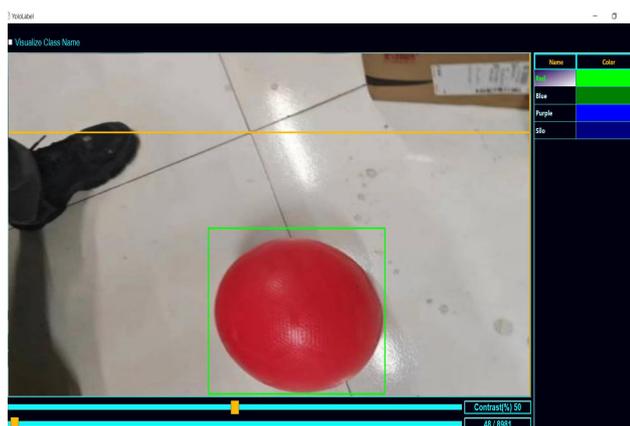
dari kelas yang akan dideteksi nantinya. Pada penelitian ini terdapat 15 kelas atau jenis objek yang akan dideteksi dan akan memfokuskan pada 3 kelas saja. Jumlah foto yang digunakan pada setiap bagian *Train*, *Valid*, dan *test* berdasarkan urgensi dari kelas yang akan dideteksi. Pada TABEL I Merupakan pembagian dari jumlah foto pada setiap bagian. Bola merah ditandai dengan kelas *RED*, bola biru ditandai dengan kelas *BLUE*, dan *SILO* ditandai dengan kelas *SILO*.

TABEL I : JUMLAH DATASET PENGUJIAN

Kelas	Pembagian Dataset		
	Train	Validation	Test
B	204	25	10
BB	39	14	15
BBB	51	25	12
BBR	30	19	12
BR	50	20	9
BRB	44	20	7
R	66	19	13
RBB	43	17	16
RBR	64	22	11
RR	65	23	9
RRR	67	43	12
SILO	1782	112	20
BLUE	4982	477	243
PURPLE	4335	375	187
RED	5688	571	287

2.3 Anotasi dataset

Anotasi atau labelling pada dataset melibatkan pemberian bounding box pada objek yang ada di gambar dan memberi nama kelas pada objek tersebut. Pada penelitian ini, pelabelan dataset menggunakan bantuan aplikasi openosource yololabel. Kemudian dilakukan proses pemberian bounding box pada setiap object yang berada pada foto tersebut dan pemberian kelas. Pada Gambar 2 merupakan proses pemberian bounding box pada object dan pemberian nama kelas pada objek.



Gambar 2: Proses Labelling Objek



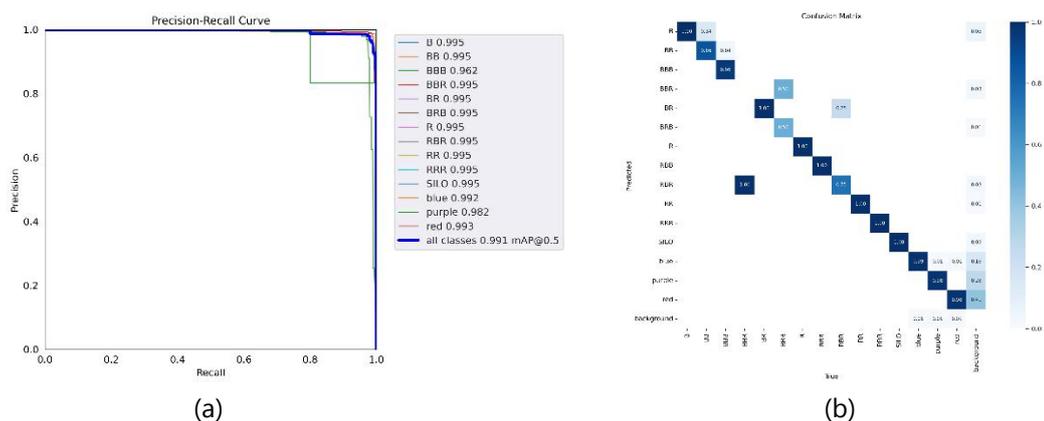
2.4 Training Dataset

Setelah melakukan proses labelling pada setiap data *Train*, *Validation*, dan *Test* kemudian dilakukan proses training pada dataset tersebut. Dalam penelitian ini, proses training dataset menggunakan platform *Google Colab*. Peneliti menggunakan *Google Colab* karena platform ini memiliki spesifikasi GPU untuk training yang cukup tinggi, sehingga dapat melakukan proses training secara cepat serta efisiensi. Proses training pada model YOLOv5 ini menggunakan parameter ukuran gambar sebesar 640, jumlah Iterasi atau epoch sebesar 300 dan Batch training sebesar 64. Proses training ini memakan waktu hingga 22 Jam. Tujuan dari training dataset tersebut adalah untuk menghasilkan paramater atau bobot yang nantinya akan digunakan dalam proses deteksi objek. Setelah proses training selesai, kemudian didapatkan file *weight* atau file bobot yang nantinya akan digunakan untuk proses deteksi objek. langkah berikutnya adalah melakukan pengujian deteksi objek sesuai dengan jarak dan sudut tertentu. Setelah berhasil mendapatkan file *weight* atau file bobot, langkah selanjutnya adalah memverifikasi dan membandingkan hasil. Jika objek yang terdeteksi sesuai dengan nama kelasnya, proses pelatihan dapat dikatakan berhasil. Karena deteksi objek membutuhkan tingkat akurasi yang tinggi, evaluasi akurasi harus dilakukan untuk membuat deteksi objek lebih stabil pada gambar dan video. Dalam penelitian ini, proses pengujian dilakukan secara realtime menggunakan *webcam*.

3. HASIL DAN PEMBAHASAN

Sistem deteksi bola dan *SILO* pada penelitian dibangun menggunakan metode YOLOv5 sebagai deteksinya. Agar dapat mendeteksi bola dan *SILO*, algoritma YOLOv5 harus dilakukan proses training terlebih dahulu. Sehingga dibutuhkan dataset yang digunakan sebagai bahan untuk training. Dataset sendiri dibagi menjadi 3 jenis, yaitu *Train*, *Validation*, dan *Test*. Setelah dataset tersebut dikelompokan berdasarkan kriteria tersebut, kemudin dilakukan proses labelling pada dataset. Proses labelling pada dataset melibatkan pemberian *bounding box* pada object yang terdapat pada dataset tersebut dan pemberian nama kelas. Setelah dilakukan pelabelan pada dataset, kemudian dilakukan proses training agar YOLOv5 dapat memprediksi dan mendeteksi objek dengan baik.

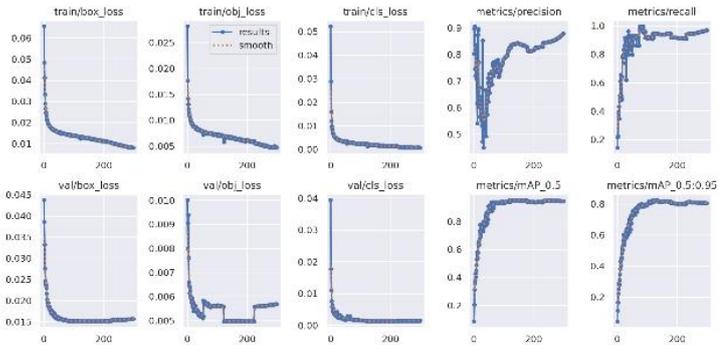
Dataset pada penelitian ini mengambil 15 kelas dengan 3 fokus kelas, yaitu bola merah, bola biru, dan *SILO*. Kelas lainnya dalam penelitian ini berfungsi sebagai referensi pada untuk membuat keputusan, sedangkan fokus deteksi hanya memfokuskan 3 kelas tersebut. Sehingga jika dilihat pada TABEL I. Jumlah dataset pada *RED*, *BLUE*, dan *SILO* memiliki data train yang banyak. Dalam pembuatan dataset ini proses pengambilan gambar dilakukan secara mandiri dan didapatkan 8.982 foto untuk data Train, 783 foto untuk data Validation, dan 410 foto untuk data Test. Jumlah anotasi kelas yang berhasil dibuat terdapat 17.510 untuk kelas Train, 1782 anotasi kelas untuk Validation, dan 863 anotasi kelas untuk Test. Dari proses training tersebut didapatkan data sebagai berikut.



Gambar 3: Kurva Precision terhadap Recall (a) dan Confusin matrix True terhadap Prediction (b)

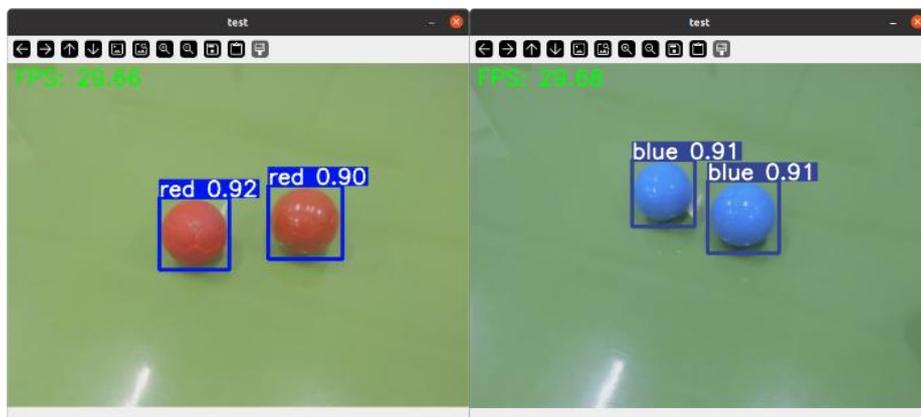


Dari Gambar 3 (a) Didapatkan hasil training nilai precision untuk semua kelas berada di nilai 0.991 terhadap nilai recal. Untuk kelas bola merah, bola biru, dan *SIL0* mendapatkan rata – rata precision sebesar 0.993 terhadap nilai recal yang berarti model training mampi memberikan prediksi yang sangat tepat, terutama pada 3 fokus kelas. Kemudian nilai pada Confusion Matrix khususnya untuk kelas bola merah, bola biru, dan *SIL0* mendapatkan nilai rata – rata 0.99 sehingga bisa dikatakan model training memberika prediksi yang sangat presisi terhadap objek sebenarnya.



Gambar 4: Grafik proses selama melakukan training pada model YOLO

Pada grafik Gambar 4 menunjukan bahwa terdapat penurunan jumlah *loss* seiring berjalannya proses iteration hingga proses iteration berhenti pada nilai *epoch*-nya yaitu 300. Pada saat *epoch* sudah menyentuh angka 300 nilai *loss* sudah mendekati dari 0.01 .Dari Hasil ini menunjukan sistem deteksi Bola dan Silo sudah bisa belajar sangat baik.

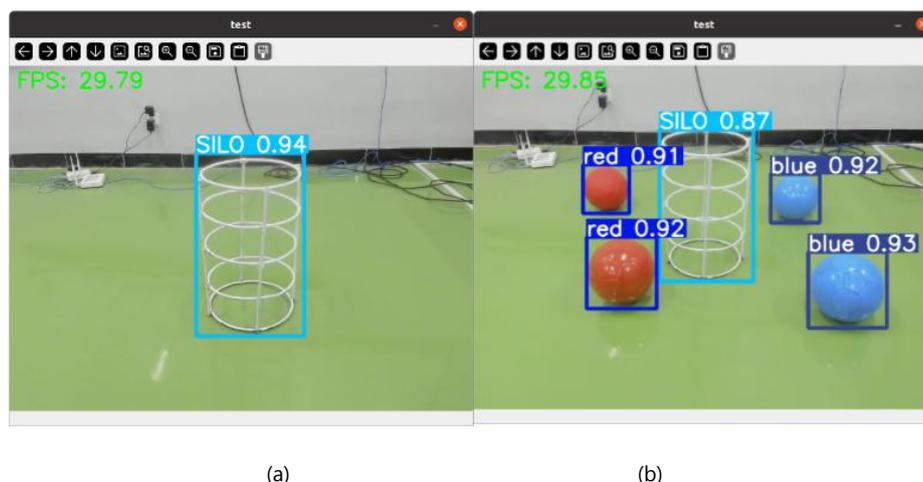


(a)

(b)

Gambar 5: Pengujian deteksi bola merah (a) dan pengujian deteksi bola biru (b)





Gambar 6: Pengujian deteksi *SILO* (a) dan pengujian deteksi bola merah, bola biru, dan *SILO* (b)

Hasil deteksi secara langsung menggunakan *webcam* ini menampilkan *bounding box* secara akurat terhadap objek yang terdeteksi. Pada Gambar 5 pengujian menggunakan bola merah dan bola biru secara langsung tanpa adanya objek lain mendapatkan hasil yang bagus, yaitu angka *confidence score* pada setiap objek berada di atas 0.90. Kemudian pada Gambar 6 ketika terdapat beberapa objek yang berbeda kelas, sistem deteksi mampu menampilkan *bounding box* secara akurat dan kelas yang benar sesuai dengan namanya.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan jika sistem deteksi bola merah, bola biru dan *SILO* menggunakan metode YOLOv5 dapat memberikan hasil yang akurat dalam pendeteksian secara langsung menggunakan *webcam*. Hasil dari uji coba tersebut mendapatkan nilai *confidence score* pada setiap objek cukup tinggi, yaitu di angka > 0.90 . Selain itu nilai *mean average precision* (MAP) untuk seluruh kelas mendapatkan nilai 0.991 atau 99.1%. Untuk kelas bola dan *SILO* mendapatkan nilai *mean average precision* (MAP) sebesar 0.993 atau 99.3%. Nilai akurasi dipengaruhi beberapa faktor, yaitu dari kualitas dataset yang diambil, sudut pengambilan, kualitas gambar, dan kualitas *webcam* atau kamera. Penggunaan dataset yang mencakup beberapa sudut pengambilan dapat meningkatkan nilai keakuratan secara signifikan. Selain itu juga, jumlah dataset yang diambil juga berperan penting dalam menentukan akurasi, oleh karena itu disarankan untuk pengambilan dataset sebisa mungkin memiliki gambar yang jelas dan mencakup banyak sudut pandang serta jumlah dataset berada di atas 1500 foto per kelas. Dengan demikian hasil pendeteksian akan mendapatkan performa yang lebih optimal.

5. UCAPAN TERIMA KASIH

Terima kasih kepada orang tua saya yang sudah membantu doa, kepada dosen yang sudah membimbing saya pada penelitian ini dan teman-teman yang saya yang terlibat dalam penelitian ini.

DAFTAR PUSTAKA

- [1] Y. Yang and T. Holvoet, "Generating safe autonomous Decision-Making in ROS," *arXiv (Cornell University)*, vol. 371, pp. 184–192, Sep. 2022, doi: 10.4204/eptcs.371.13. J. Breckling, Ed., *The Analysis of Directional Time Series: Applications to Wind Speed and Direction*, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.
- [2] N. Aisyah *et al.*, "Pengujian robot otomatis pendeteksi rintangan berbasis mikrokontroler," *Jurnal Esensi Infokom: Jurnal Esensi Sistem Informasi Dan Sistem Komputer/Jurnal Esensi Infokom*, vol. 5, no. 2, pp. 1–6, Feb. 2022, doi: 10.55886/infokom.v5i2.264.
- [3] K. D. Setyanto, I. Fibriani, and S. Sumardi, "PENGENDALIAN MOBILE ROBOT VISION MENGGUNAKAN WEBCAM PADA OBJEK ARAH PANAH BERBASIS RASPBERRY PI," May 02, 2016. <https://jurnal.unej.ac.id/index.php/E-JAEI/article/view/2533>
- [4] Asian Broadcasting Union, "Abu Robocon 2024," *Aburobocon2024*. <https://aburobocon2024.vtv.gov.vn/gamerules> (accessed Apr. 17, 2024).



- [5] D. I. Mulyana and M. A. Rofik, "Implementasi deteksi real time klasifikasi jenis kendaraan di Indonesia menggunakan metode YOLOV5," *Jurnal Pendidikan Tambusai*, vol. 6, no. 3, pp. 13971–13982, Jul. 2022, doi: 10.31004/jptam.v6i3.4825.
- [6] L. Suroiyah, Y. Rahmawati, and R. Dijaya, "FACEMASK DETECTION USING YOLO V5," *Jurnal Teknik Informatika*, vol. 4, no. 6, pp. 1277–1286, Dec. 2023, doi: 10.52436/1.jutif.2023.4.6.1043.
- [7] H.-V. Nguyen, J.-H. Bae, Y. E. Lee, H. Lee, and K.-R. Kwon, "Comparison of Pre-Trained YOLO Models on Steel Surface Defects Detector Based on Transfer Learning with GPU-Based Embedded Devices," *Sensors*, vol. 22, no. 24, p. 9926, Dec. 2022, doi: 10.3390/s22249926.
- [8] D.N. Nugroho and L. Anifah, "Perancangan Sistem Deteksi Objek Bola Dan Gawang Pada Robot Sepakbola Menggunakan Metode Darknet YOLO," *JIEET (Journal of Information Engineering and Educational Technology)*, vol. Volume 07 Nomor 01, 2023, Jun. 2023, doi: 10.26740/jieet.v7n1.p22-29.
- [9] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, "Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach," *Procedia Comput. Sci.*, vol. 132, pp. 679–688, 2018, doi: 10.1016/j.procs.2018.05.069.
- [10] A. Haq Nalband, S. Basavaraj Channi, S. S. Reddy, S. A. S, P. A. Totad, and A. N. Haq, "AI Powered YOLO based traffic management system through application advancement," 2021. [Online]. Available: <https://www.researchgate.net/publication/352645443>
- [11] W. Muldayani, "IMPLEMENTASI SISTEM OBJECT TRACKING UNTUK MENDETEKSI DUA OBJEK BERBASIS DEEP LEARNING," *Simetris: Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer/Simetris*, vol. 14, no. 1, pp. 1–14, Apr. 2023, doi: 10.24176/simet.v14i1.9236.
- [12] glenn-jocher, "Ultralytic-yolov5," 2022. <https://github.com/ultralytics/yolov5> (accessed April 20, 2024).
- [13] H. Lin, J. D. Deng, D. Albers, and F. W. Siebert, "Helmet use detection of tracked motorcycles using CNN-Based Multi-Task learning," *IEEE Access*, vol. 8, pp. 162073–162084, Jan. 2020, doi: 10.1109/access.2020.3021357.
- [14] M.H.Ashar and D.Suarna, "Implementasi algoritma YOLOV5 dalam mendeteksi penggunaan masker pada Kantor Biro Umum Gubernur Sulawesi Barat," *KLIK: Kajian Ilmiah Informatika Dan Komputer*, vol. Vol 3, No 3, Desember 2022, Dec. 2022, doi: 10.30865/klik.v3i3.559.

