

# Deteksi Objek Menggunakan Metode Yolo dan Implementasinya pada Robot Bawah Air

Husnan, Chastine Fatichah, Rudy Dikairono

Departemen Teknik Informatika, Institut Teknologi Sepuluh Nopember (ITS)

*e-mail:* chastine@if.its.ac.id

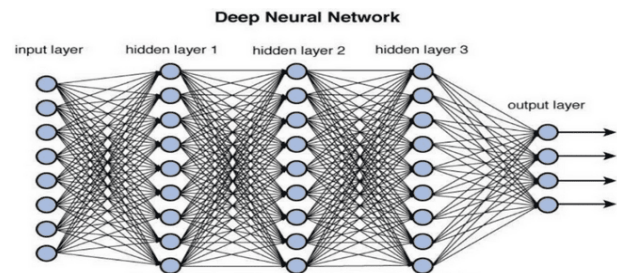
**Abstrak**—Penelitian ini membahas penggunaan berbagai arsitektur model *deep learning* dalam mendeteksi objek bawah air seperti gerbang, tiang, bola, dan baskom untuk meningkatkan performa robot dalam eksplorasi bawah air dalam konteks kompetisi SAUVC (*Singapore AUV Challenge*). Metode yang digunakan adalah YOLO (You Only Look Once) dan menggunakan berbagai jenis YOLOv5, seperti YOLOv5s, YOLOv5m, YOLOv5l, dan YOLOv5x. Hasil pengujian menunjukkan bahwa YOLOv5x memiliki rata-rata jarak deteksi terjauh sebesar 6,12 meter dan  $mAP@[0.5:0.95]$  paling tinggi yaitu 0,881, namun ukurannya yang besar memerlukan daya komputasi yang tinggi. Di sisi lain, YOLOv5s memiliki ukuran model yang lebih kecil yaitu 14,5 MB, namun tetap memberikan performa yang baik dengan  $mAP@[0.5:0.95]$  sebesar 0,872. Berdasarkan temuan ini, YOLOv5s lebih sesuai untuk digunakan dalam mendeteksi objek bawah air pada kompetisi SAUVC karena selain ukurannya yang lebih kecil, YOLOv5s juga memberikan performa yang memadai. Penggunaan model ini diharapkan dapat meningkatkan kinerja robot dalam eksplorasi bawah air dan membantu dalam menyelesaikan misi yang ditugaskan dalam waktu yang ditentukan.

**Kata Kunci**—SAUVC, Robot Bawah Air, *Deep Learning*, *You Only Look Once (YOLO)*.

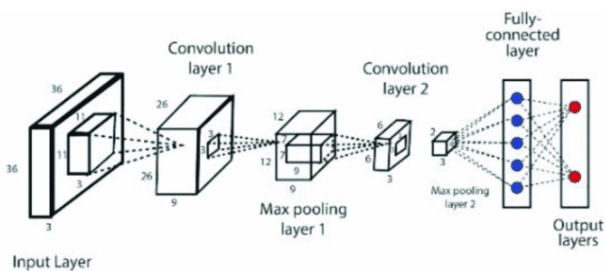
## I. PENDAHULUAN

ROBOT bawah air merupakan inovasi teknologi yang berpotensi untuk menggantikan atau mendampingi manusia dalam menyelesaikan misi-misi di bawah air yang tidak diketahui dan kompleks dengan efisien. Dengan kemampuan adaptasi dan fungsionalitas yang terus berkembang, robot bawah air mampu melakukan tugas-tugas yang berisiko tinggi dan memerlukan ketelitian tinggi tanpa mengorbankan keselamatan manusia. Keberadaan robot bawah air membuka peluang baru dalam eksplorasi laut, survei bawah air, penelitian lingkungan laut, dan sektor-sektor industri seperti pertambangan dan perkapalan. Dalam beberapa tahun terakhir, teknologi robotika bawah air telah mengalami kemajuan pesat, membuktikan bahwa peran robot bawah air bukan hanya sebagai alat bantu, tetapi sebagai mitra setia manusia dalam menghadapi tantangan-tantangan bawah laut [1].

SAUVC (*Singapore AUV Challenge*) adalah kompetisi robot bawah air yang diikuti oleh sejumlah negara di dunia, sekaligus menjadi tempat pertemuan ilmiah terkait inovasi robot bawah air. Tujuan utamanya adalah menguji kemampuan robot bawah air dalam melakukan eksplorasi bawah air dan menyelesaikan misi yang diberikan dalam waktu yang ditentukan. Dalam kompetisi ini, data visual, terutama yang berkaitan dengan lingkungan bawah air, menjadi bagian penting yang harus diperoleh oleh tim robot bawah air. Robot bawah air harus mampu mengenali berbagai objek yang ada di dalam kolam seperti gerbang, tiang, bola,



Gambar 1. Arsitektur *Deep Neural Network*.

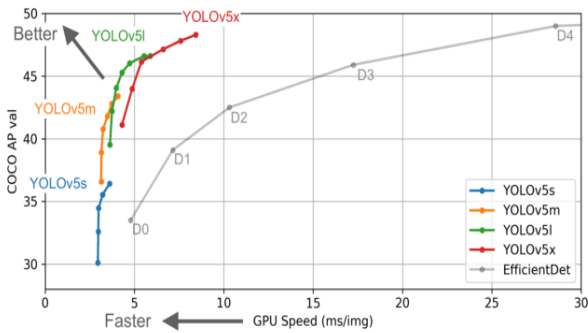


Gambar 2. Arsitektur *Convolutional Neural Network*.

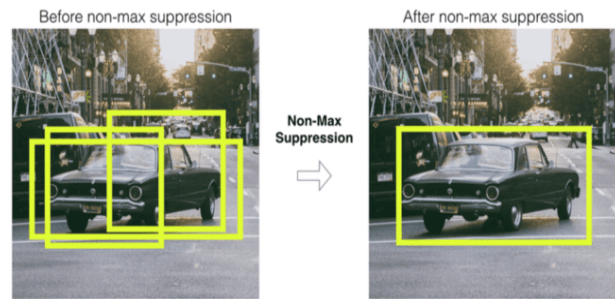
dan baskom. Namun, ada aturan ketat yang mengatur dalam SAUVC, yaitu larangan untuk menghindari tiang berwarna merah yang berfungsi sebagai rintangan di bawah air. Oleh karena itu, deteksi tiang berwarna merah menjadi salah satu hal yang sangat penting dalam persiapan dan pelaksanaan kompetisi ini. Dengan mampu mengenali dan menghindari tiang berwarna merah, tim robot bawah air dapat menunjukkan keunggulan teknologi yang mereka miliki dalam menghadapi tantangan di lingkungan bawah air [2].

Dalam mengenali objek yang ada di bawah air, dapat digunakan pendekatan *deep learning* menggunakan berbagai arsitektur. Metode yang digunakan harus bisa mendeteksi objek di bawah air. Pada penelitian sebelumnya terdapat dataset yang terdiri dari 5 jenis objek bawah air sekitar 800 gambar yang dilatih menggunakan model *deep learning* dengan arsitektur YOLOv3-tiny, dataset tersebut kemudian dibagi menjadi 80% data pelatihan dan 20% data pengujian, hasil dari penelitian tersebut mendapatkan loss sebesar 0,6579 dan  $mAP@0.5$  sebesar 88,05% [2]. Pada penelitian sebelumnya belum menerapkan model *deep learning* dengan arsitektur yang lebih baru. Inovasi terkait deteksi objek khususnya di bidang bawah air sangat diperlukan agar performa robot bawah air lebih baik kedepannya. Pada penelitian sebelumnya terkait performa model deteksi objek seperti YOLOv5 dan Mask R-CNN yang mana hasil penelitiannya menunjukkan bahwa YOLOv5 lebih unggul dibandingkan Mask R-CNN dengan  $mAP@[0.5:0.95]$  sebesar 0,987 beserta waktu pelatihan yang lebih cepat [3].

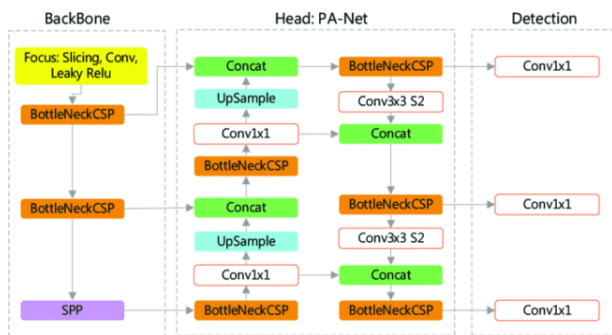
Dari permasalahan yang telah disebutkan sebelumnya, sistem deteksi objek dibuat menggunakan model *deep learning*. Model melakukan deteksi terhadap gerbang, tiang, bola dan baskom yang terdapat di bawah air. Sistem tersebut



Gambar 3. Performa Algoritma YOLOv5.



Gambar 5. Non-Maximum Suppression.



Gambar 4. Arsitektur Algoritma YOLOv5.

diujikan menggunakan berbagai jenis model YOLOv5. Dari hasil pengujian tersebut didapatkan nilai presisi, recall dan mAP pada beberapa jenis algoritma yang kemudian digunakan oleh tim Banyubramanta ITS dalam perlombaan SAUVC.

## II. TINJAUAN PUSTAKA

### A. Jaringan Saraf Tiruan

Jaringan saraf tiruan terinspirasi dari sistem saraf manusia yang mana terdiri dari kumpulan sel saraf yang terhubung satu sama lain, biasanya struktur jaringan saraf tiruan ini terdiri dari *input layer*, *output layer* dan *hidden layer*. Pada *input layer* sebuah nilai *input* akan diproses menjadi nilai *output*, secara tidak langsung *hidden layer* terkoneksi dengan lapisan *input* dan *output* untuk memperluas kemampuan jaringan saraf tiruan, sedangkan pada *output layer* menjadi solusi dari nilai *input* [4].

### B. Deep Learning

*Deep learning* telah menjadi kunci pada pengembangan aplikasi kecerdasan buatan salah satunya di bidang visi komputer. *Deep learning* didasarkan pada *Deep Neural Network* (DNN) yang mana tidak terlepas dari jaringan saraf tiruan. Pada DNN memiliki banyak *hidden layer* yang membuatnya berbeda dari jaringan saraf tiruan lainnya, dengan banyaknya *hidden layer* pada DNN membutuhkan upaya komputasi yang kompleks dan penyimpanan yang besar untuk mencapai kinerja tinggi. Ilustrasi arsitektur *Deep Neural Network* (DNN) ditunjukkan pada Gambar 1 [5].

### C. Convolutional Neural Network

*Convolutional Neural Network* (CNN) merupakan salah satu metode *deep learning* yang umum digunakan untuk mengolah data gambar atau video. CNN terdiri dari beberapa *layer* yang terdiri dari *input layer*, *output layer* dan *layer* yang tersembunyi terdiri dari beberapa *convolution layer*, *pooling*

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Gambar 6. Intersection over Union.

*layer* dan *fully connected layer*. Pada *convolution layer* berfungsi untuk mengekstraksi fitur yang mana didalamnya perlu mengatur ukuran *kernel*, pergeseran setiap *kernel* dan menambahkan *padding* pada *input* agar ukuran *output* sama dengan ukuran *input*. Adapun pada *pooling layer* berfungsi untuk mengurangi dimensi sehingga *layer* berikutnya tidak memerlukan komputasi yang besar. Sedangkan *fully connected layer* berfungsi untuk melakukan klasifikasi berdasarkan fitur yang diekstrak. Contoh arsitektur *Convolutional Neural Network* (CNN) dapat dilihat pada Gambar 2 [6].

### D. You Only Look Once

YOLO (*You Only Look Once*) merupakan algoritma *deep learning* untuk melakukan deteksi objek secara *real time* dengan memanfaatkan algoritma *Convolutional Neural Network* (CNN) untuk memproses data visual. YOLO mendeteksi objek dengan menerima sebuah *input* gambar kemudian dibagi menjadi kotak dengan ukuran  $S \times S$  untuk membuat *bounding box* dan sistem memprediksi probabilitas kelas berdasarkan tingkat keyakinan dalam mendeteksi sebuah objek [7].

Salah satu pengembangan dari YOLO kini telah sampai di versi kelima yaitu YOLOv5 yang memiliki performa yang lebih baik dibandingkan dengan YOLO versi sebelumnya. YOLOv5 memiliki beberapa jenis model diantaranya adalah YOLOv5s, YOLOv5m, YOLOv5l dan YOLOv5x. Huruf kecil yang dituliskan setelah jenis model memiliki arti *small*, *medium*, *large* dan *extra-large*. Semakin besar jenis model YOLOv5 yang digunakan dapat menyebabkan proses pelatihan nya membutuhkan spesifikasi komputer yang tinggi serta ukuran berkas program menjadi semakin besar, akan tetapi model yang dihasilkan dari hasil pelatihan menjadi semakin cepat dan akurat. Performa algoritma YOLOv5 dapat dilihat pada Gambar 3 [8].

Arsitektur algoritma YOLOv5 terdiri dari tiga komponen utama yaitu *Backbone*, *Head* dan *Detection*. *BackBone* adalah CNN yang mengumpulkan serta membentuk fitur gambar

Tabel 1.  
Jumlah Dataset Tiap Kelas.

Kelas	Jumlah Gambar
Baskom biru	2.266
Gerbang	2.007
Bola pink	2.080
Tiang merah	1.962
Tiang kuning	1.615

Tabel 2.  
Jumlah Dataset Pelatihan Tiap Kelas.

Kelas	Jumlah Gambar
Baskom biru	1.621
Gerbang	1.402
Bola pink	1.701
Tiang merah	1.378
Tiang kuning	1.413

Tabel 3.  
Jumlah Dataset Validasi Tiap Kelas.

Kelas	Jumlah Gambar
Baskom biru	313
Gerbang	317
Bola pink	187
Tiang merah	298
Tiang kuning	101

Tabel 4.  
Jumlah Dataset Pengujian Tiap Kelas.

Kelas	Jumlah Gambar
Baskom biru	332
Gerbang	288
Bola pink	192
Tiang merah	286
Tiang kuning	101

pada perincian yang berbeda. Adapun *Head* adalah serangkaian lapisan untuk menggabungkan fitur gambar untuk proses prediksi. Sedangkan *Detection* adalah proses yang memanfaatkan fitur dari *Head* dan mengambil langkah prediksi kotak dan kelas. Arsitektur dari algoritma YOLOv5 dapat dilihat pada Gambar 4 [9].

#### E. Non-Maximum Suppression

*Non-Maximum Suppression* (NMS) adalah sebuah teknik yang sering digunakan dalam deteksi objek untuk mengatasi masalah penggandaan saat deteksi objek yang tumpang tindih. NMS bertujuan untuk menyaring hasil deteksi dan menghilangkan deteksi yang berlebihan dari objek yang sama. Hasil dari proses NMS adalah kumpulan deteksi yang saling berdekatan atau tumpang tindih yang telah dibersihkan sehingga hanya ada satu deteksi untuk setiap objek yang terdeteksi. Dengan menghilangkan deteksi yang berlebihan, deteksi objek menjadi lebih akurat dan memberikan hasil yang lebih bersih dan terorganisir. Ilustrasi dari NMS dapat dilihat pada Gambar 5.

#### F. Evaluasi Performa

IoU (*Intersection over Union*) digunakan untuk mengukur sejauh mana prediksi model mendekati label sebenarnya dari objek dalam suatu gambar. Konsep dasar dari IoU adalah menghitung rasio area dari bagian bersama antara prediksi dan label sebenarnya terhadap area dari gabungan keduanya. Nilai ambang batas IoU ditentukan tergantung pada kasus penggunaan dan tingkat toleransi kesalahan yang dapat diterima. Jika IoU melampaui ambang batas tertentu, maka prediksi dianggap benar; sebaliknya, jika IoU di bawah ambang batas tersebut, prediksi dianggap salah. Ilustrasi dari proses perhitungan IoU dapat dilihat pada Gambar 6.

mAP (*mean Average Precision*) merupakan nilai rata-rata dari AP (*Average Precision*) pada suatu model. Untuk mengukur performa deteksi objek pada model, mAP dapat dihitung dengan membatasi IoU dari hasil deteksi. *Average Precision* didapatkan dengan menghitung nilai rata-rata presisi dibandingkan dengan nilai *recall* dalam rentang 0 hingga 1. Untuk menghitungnya, dapat menggunakan metode AUC (*Area Under Curve*) presisi-*recall* [10].

Presisi adalah tingkat ketepatan antara informasi yang diminta dengan hasil prediksi yang diberikan oleh model. Nilai presisi didapatkan dari TP (*True Positive*) dibagi dengan total TP (*True Positive*) dan FP (*False Positive*). Adapun *recall* adalah keberhasilan sebuah model dalam menemukan kembali sebuah informasi. Nilai *recall* didapatkan dari TP (*True Positive*) dibagi dengan total TP (*True Positive*) dan FN (*False Negative*) [11].

### III. PERANCANGAN

#### A. Perangkat yang Digunakan

Narudaka merupakan robot bawah air yang didesain khusus oleh tim Banyubramanta ITS untuk perlombaan *Singapore AUV Challenge* (SAUVC). Kemampuan robot bawah air ini diharapkan dapat melakukan pendeteksian objek di bawah air, bergerak dengan stabil ke segala arah untuk mendekati objek dan menghindari rintangan. Robot bawah air ini memiliki dimensi dengan panjang 59 cm, lebar 45 cm, tinggi 45 cm dan memiliki berat 18 kg.

#### B. Perancangan Dataset

Pada penelitian kali ini, diperlukan dataset berupa gambar yang sudah dilabeli untuk dilakukan proses pelatihan model *deep learning*. Dataset berupa gambar tersebut diambil secara manual di kolam renang yang sudah diletakkan beberapa objek sesuai dengan ketentuan kompetisi *Singapore AUV Challenge* (SAUVC).

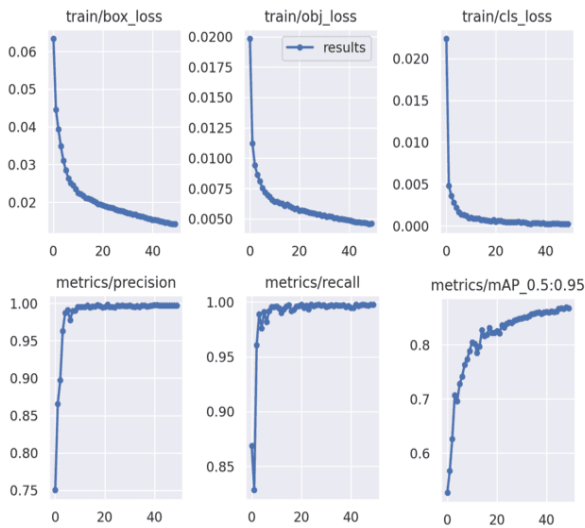
Adapun dataset berisi 9.930 gambar. Dataset berupa gambar tersebut dilabeli menggunakan aplikasi berbasis web bernama Roboflow. Beberapa kelas yang digunakan antara lain baskom biru, gerbang, bola pink, tiang kuning dan tiang merah. Setelah dilakukan pelabelan, didapatkan jumlah dataset pada setiap kelas. Rinciannya dapat dilihat pada Tabel 1.

#### C. Pembagian Dataset

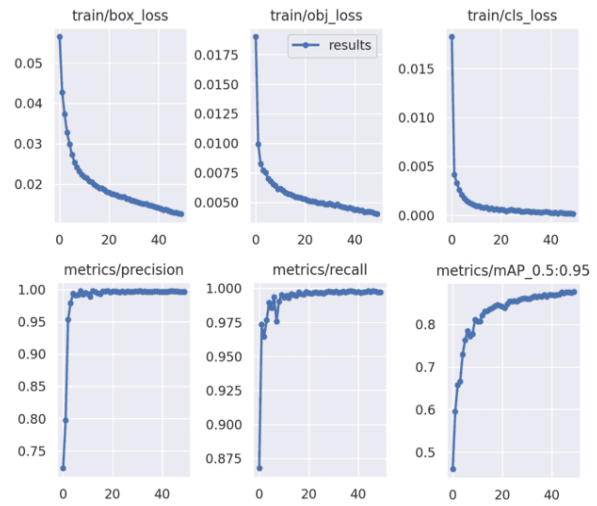
Setelah dilakukan proses pelabelan dataset kemudian digunakan untuk pelatihan model. Dataset dibagi menjadi 76% data pelatihan, 12% data validasi dan 12% data pengujian. Dari pembagian tersebut didapatkan data pelatihan sebanyak 7.515 gambar, data validasi sebanyak 1.216 gambar dan data pengujian sebanyak 1.199 gambar. Setelah dilakukan pembagian dataset didapatkan jumlah data pelatihan, data validasi dan data pengujian pada setiap kelas. Bisa dilihat pada Tabel 2, Tabel 3 dan Tabel 4.

#### D. Pelatihan Model Deteksi Objek

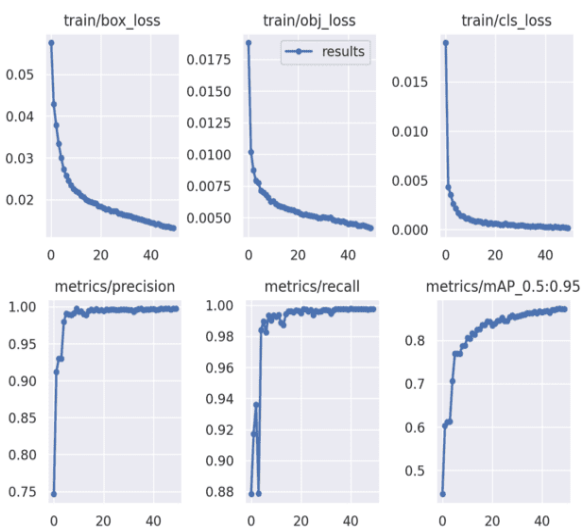
Pada penelitian ini menggunakan beberapa jenis YOLOv5 seperti YOLOv5s, YOLOv5m, YOLOv5l dan YOLOv5x untuk dilakukan pelatihan model agar mengetahui kinerja setiap jenis model dalam mendeteksi objek bawah air sehingga dapat memilih jenis model YOLOv5 yang tepat untuk meningkatkan performa robot bawah air. Pelatihan



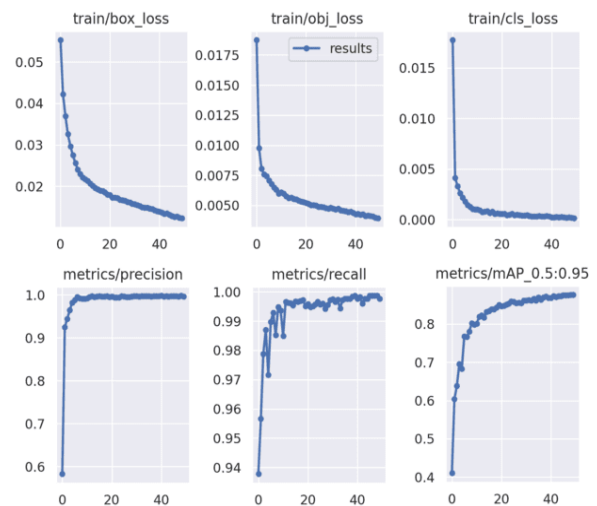
Gambar 7. Grafik Pelatihan YOLOv5s.



Gambar 9. Grafik Pelatihan YOLOv5l.



Gambar 8. Grafik Pelatihan YOLOv5m.



Gambar 10. Grafik Pelatihan YOLOv5x.

dapat dimulai dengan menambahkan beberapa parameter pada perintah pelatihan sebagai berikut:

- 1) *img* sebesar 640
- 2) *batch* sebesar 16
- 3) *epoch* sebanyak 50
- 4) Lokasi *file* "data.yaml" yang berisi informasi dataset
- 5) *weights* tergantung model yang ingin digunakan
- 6) Lokasi *file* hasil pelatihan model
- 7) *cache* untuk mempercepat proses pelatihan

#### IV. UJI COBA DAN EVALUASI

Model YOLOv5s memiliki ukuran sebesar 14,5 MB setelah melalui pelatihan, menjadikannya sebagai model dengan ukuran terkecil dibandingkan dengan model YOLOv5 lainnya. Proses pelatihan model YOLOv5s memakan waktu selama 1 jam 24 menit 26 detik. Selama pelatihan, model YOLOv5s mendapatkan *box loss* sebesar 0,01427, *obj loss* sebesar 0,004616, *cls loss* sebesar 0,0002221, presisi sebesar 0,997, *recall* sebesar 0,998 dan mAP@[0.5:0.95] sebesar 0,869 yang mana menjadikan YOLOv5s sebagai model dengan mAP@[0.5:0.95] terendah dibandingkan dengan model YOLOv5 lainnya. Grafik pelatihan dapat dilihat pada Gambar 7.

Pada pelatihan model YOLOv5m didapatkan ukuran model sebesar 42,2 MB dengan waktu pelatihan 54 menit 33 detik yang mana menjadikan model YOLOv5m sebagai model dengan waktu pelatihan tercepat dibandingkan dengan model YOLOv5 lainnya. Pada pelatihan model YOLOv5m didapatkan *box loss* sebesar 0,01319, *obj loss* sebesar 0,004173, *cls loss* sebesar 0,0001426, presisi sebesar 0,996, *recall* sebesar 0,997 dan mAP@[0.5:0.95] sebesar 0,873 yang mana grafik pelatihannya dapat dilihat pada Gambar 8.

Model YOLOv5l memiliki ukuran sebesar 92,9 MB setelah melalui pelatihan. Proses pelatihan model YOLOv5l memakan waktu selama 1 jam 8 menit. Selama pelatihan, model YOLOv5l mendapatkan *box loss* sebesar 0,0126, *obj loss* sebesar 0,004023, *cls loss* sebesar 0,0001372, presisi sebesar 0,996, *recall* sebesar 0,997 dan mAP@[0.5:0.95] sebesar 0,876 yang mana menjadikan YOLOv5l sebagai model dengan mAP@[0.5:0.95] tertinggi setara dengan YOLOv5x. Grafik pelatihan dapat dilihat pada Gambar 9.

Pada pelatihan model YOLOv5x didapatkan ukuran model sebesar 173,1MB yang mana menjadikan YOLOv5x sebagai model dengan ukuran terbesar dibandingkan dengan model YOLOv5 lainnya. Proses pelatihan model YOLOv5x memakan waktu selama 1 jam 35 menit 14 detik yang mana menjadikan model YOLOv5x sebagai model dengan waktu pelatihan terlama dibandingkan dengan model YOLOv5



Tabel 5.  
Penguujian Setiap Model YOLOv5.

Model	Presisi	Recall	mAP@[0.5:0.95]
YOLOv5s	0.997	0.994	0.872
YOLOv5m	0.998	0.994	0.874
YOLOv5l	0.997	0.997	0.88
YOLOv5x	0.998	0.998	0.881

Tabel 6.  
Penguujian Jarak Maksimal Deteksi.

Model	Kelas	Jarak Maksimal Deteksi (Meter)	Rata-Rata Jarak Deteksi (Meter)
YOLOv5s	Baskom biru	5,4	3,38
	Gerbang	6	
	Bola pink	1,1	
	Tiang merah	2,4	
	Tiang kuning	2	
YOLOv5m	Baskom biru	5,6	4,92
	Gerbang	8	
	Bola pink	1,6	
	Tiang merah	7	
	Tiang kuning	2,4	
YOLOv5l	Baskom biru	6	5,56
	Gerbang	9	
	Bola pink	1,8	
	Tiang merah	8,5	
	Tiang kuning	2,5	
YOLOv5x	Baskom biru	7	6,12
	Gerbang	10	
	Bola pink	1,9	
	Tiang merah	9	
	Tiang kuning	2,7	

lainnya. Selama pelatihan, model YOLOv5x mendapatkan *box loss* sebesar 0,0123, *obj loss* sebesar 0,003935, *cls loss* sebesar 0,0001272, presisi sebesar 0,998, *recall* sebesar 0,999 dan mAP@[0.5:0.95] sebesar 0,876 yang mana menjadikan YOLOv5x sebagai model dengan mAP@[0.5:0.95] tertinggi setara dengan YOLOv5l. Grafik pelatihan dapat dilihat pada Gambar 10.

Untuk evaluasi performa digunakan *mean Average Precision* (mAP) pada data pengujian agar mengetahui kinerja model dalam mendeteksi objek bawah air sehingga dapat memilih model yang tepat untuk meningkatkan performa robot bawah air. Pengujian setiap model YOLOv5 dapat dilihat pada Tabel 5.

Pada model YOLOv5s didapatkan nilai presisi sebesar 0,997, *recall* sebesar 0,994 dan mAP@[0.5:0.95] sebesar 0,872, yang mana membuat YOLOv5s sebagai model dengan mAP@[0.5:0.95] terendah dibandingkan dengan model YOLOv5 lainnya. Model YOLOv5m memiliki presisi sebesar 0,998, *recall* sebesar 0,994, dan mAP@[0.5:0.95] sebesar 0,874. Pada model YOLOv5l didapatkan nilai presisi sebesar 0,997, *recall* sebesar 0,997 dan mAP@[0.5:0.95] sebesar 0,88. Model YOLOv5x memiliki presisi sebesar 0,998, *recall* sebesar 0,998, dan mAP@[0.5:0.95] sebesar 0,881, yang mana membuat YOLOv5x sebagai model dengan mAP@[0.5:0.95] tertinggi dibandingkan dengan model YOLOv5 lainnya.

Selanjutnya pengujian jarak maksimal deteksi yang bertujuan untuk memperoleh jarak terjauh yang dapat dideteksi oleh setiap model yang telah dilakukan pelatihan sebelumnya. Informasi mengenai hasil pengujian jarak maksimal deteksi dapat ditemukan di Tabel 6.

Melalui pengujian jarak yang telah dilakukan, terlihat

bahwa semakin besar ukuran model yang digunakan, semakin meningkat juga jarak maksimum objek yang dapat dideteksi. Dalam percobaan ini, ditemukan bahwa YOLOv5x merupakan model dengan jangkauan deteksi terjauh, mencapai rata-rata jarak deteksi 6,12 m. Sementara itu, YOLOv5s merupakan model dengan jarak deteksi terpendek, dengan rata-rata jarak deteksi sebesar 3,38 m.

## V. KESIMPULAN

Dari uji coba yang telah dilakukan dalam penelitian ini, didapatkan kesimpulan sebagai berikut: (1) Model YOLOv5s memiliki ukuran terkecil dibandingkan dengan model lainnya mampu memberikan deteksi objek yang cukup baik, penggunaan model dengan ukuran yang lebih kecil seperti YOLOv5s dapat mengurangi beban komputasi dan memungkinkan robot untuk bekerja dengan lebih baik dalam kondisi yang terbatas. Hal ini membuat YOLOv5s menjadi model deep learning yang tepat untuk mendeteksi objek bawah air pada perlombaan SAUVC; (2) Model YOLOv5 yang digunakan dalam penelitian ini memberikan hasil deteksi objek bawah air yang akurat. Pengujian menggunakan mAP menunjukkan bahwa model-model tersebut berhasil mendeteksi objek dengan tingkat keberhasilan yang tinggi. Selain itu, penelitian ini juga menguji model-model YOLOv5 untuk menentukan jarak maksimal deteksi. Hasilnya menunjukkan bahwa model YOLOv5x mampu mendeteksi objek dalam jarak terjauh dan memiliki nilai mAP tertinggi dibandingkan dengan model lain. Namun, dalam konteks perlombaan SAUVC, model YOLOv5x tidak cocok digunakan untuk mendeteksi objek bawah air karena ukurannya yang besar sehingga memerlukan daya komputasi besar. Oleh karena itu, model YOLOv5s lebih sesuai digunakan dalam mendeteksi objek bawah air pada perlombaan SAUVC karena memiliki ukuran yang lebih kecil namun tetap memberikan tingkat performa yang baik.

Berdasarkan kesimpulan uji coba dari penelitian ini, terdapat beberapa saran sebagai berikut: (1) Diperlukan eksperimen terhadap variasi YOLO lainnya. (2) Diperlukan eksperimen pada komputer yang memiliki kemampuan komputasi yang lebih besar agar dapat memanfaatkan model dengan tingkat mAP yang lebih tinggi.

## DAFTAR PUSTAKA

- [1] X. Zeng, M. Xia, Z. Luo, J. Shang, Y. Xu, and Q. Yin, "Design and control of an underwater robot based on hybrid propulsion of quadrotor and bionic undulating fin," *J. Mar. Sci. Eng.*, vol. 10, no. 9, p. 1327, 2022.
- [2] M. N. Mehta, S. Mylraj, and V. N. Bhate, "Development of AUV for Sauvc During Covid-19," in *OCEANS 2021: San Diego--Porto*, 2021, pp. 1–8.
- [3] I. W. A. S. Darma, N. Suciati, and D. Siahaan, "A Performance Comparison of Balinese Carving Motif Detection and Recognition using YOLOv5 and Mask R-CNN," in *2021 5th International Conference on Informatics and Computational Sciences (ICICoS)*, 2021, pp. 52–57.
- [4] P. Punyani, R. Gupta, and A. Kumar, "Neural networks for facial age estimation: a survey on recent advances," *Artif. Intell. Rev.*, vol. 53, pp. 3299–3347, 2020.
- [5] R. Ayachi, Y. Said, and A. Ben Abdelali, "Optimizing neural networks for efficient FPGA implementation: A survey," *Arch. Comput. Methods Eng.*, pp. 1–11, 2021.
- [6] A. Narayan and R. Muthalagu, "Image Character Recognition using Convolutional Neural Networks," in *2021 Seventh International conference on Bio Signals, Images, and Instrumentation (ICBSII)*, 2021, pp. 1–5.

- [7] T. Diwan, G. Anirudh, and J. V Tembhrne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *Multimed. Tools Appl.*, vol. 82, no. 6, pp. 9243–9275, 2023.
- [8] S. N. Bushra, G. Shobana, K. U. Maheswari, and N. Subramanian, "Smart Video Surveillance Based Weapon Identification using YOLOv5," in *2022 International Conference on Electronic Systems and Intelligent Computing (ICESIC)*, 2022, pp. 351–357.
- [9] J. Ieamsaard, S. N. Charoensook, and S. Yammen, "Deep Learning-Based Face Mask Detection using YOLOv5," in *2021 9th International Electrical Engineering Congress (iEECON)*, 2021, pp. 428–431.
- [10] B. Wang, "A Parallel Implementation of Computing Mean Average Precision," *Department Computer Science; Cornell University*; 2022.
- [11] Y. Liu, S. Yang, and others, "Application of decision tree-based classification algorithm on content marketing," *J. Math.*, vol. 2022, 2022.