

Sistem Deteksi Penyakit Ikan Koi Menggunakan Metode YOLOv5

Koi Fish Disease Detection System Using the YOLOv5 Method

Citra Zaskia Pratiwi^{1*}, Shara Jayanti², Raedy Anwar Subiantoro³, Bobby Wisely Ziliwu⁴, Eddy Mustono⁵

¹Mekanisasi Perikanan, Politeknik Kelautan dan Perikanan Sidoarjo
Jl. Raya Buncitan Kotak Pos 1, Sedati, Sidoarjo

² Teknik Penanganan Patologi Ikan, Politeknik KP Sidoarjo
Jl. Raya Buncitan Kotak Pos 1, Sedati, Sidoarjo

^{3,5} Permesinan Perikanan, Politeknik AUP Jakarta
Jl. AUP Barat, Pasar Minggu, Jakarta Selatan

⁴ Mekanisasi Perikanan, Politeknik Kelautan dan Perikanan Sorong
Jl. Kapitan Pattimura Tanjung Kasuari Suprau, Sorong, Papua Barat Daya
citra.pratiwi@kpk.go.id^{1*}, sharajayanti@gmail.com², raedysubiantoro@gmail.com³,
bobywisel@gmail.com⁴, mustonoeddy@gmail.com⁵

Abstrak – Penyebab utama peningkatan kematian di budidaya perikanan adalah penyakit ikan. Salah satu ikan yang mudah terserang hama dan penyakit adalah ikan koi. Identifikasi penyakit ikan koi secara otomatis pada tahap awal merupakan langkah penting untuk mencegah penyebaran penyakit. Deteksi penyakit ikan koi dapat dilakukan melalui berbagai cara yaitu pemeriksaan visual, penggunaan sensor fisik, analisis genetik, teknologi citra dan pengolahan citra, biosensor dan biochips, teknologi saringan molekuler, jaringan saraf tiruan (artificial neural networks) dan pembelajaran mesin. Sistem deteksi penyakit ikan koi pada penelitian ini menggunakan YOLOv5. Hal ini dikarenakan YOLOv5 memiliki beberapa kelebihan antara lain memiliki tingkat akurasi yang tinggi, dapat mendeteksi secara real time, model ringan, sederhana dalam training, dan open source. Penelitian ini melalui beberapa tahap, yaitu pengumpulan dan penyiapan data, pelatihan model dengan algoritma YOLOv5, serta proses evaluasi terhadap performa model. Pada tahap ini, model dievaluasi berdasarkan nilai accuracy, recall, precision, dan mean Average Precision (mAP). Nilai accuracy sebesar 90% didapatkan sebagai hasil evaluasi model, nilai precision untuk ikan sehat (healthy-fish) sebesar 83,33% sedangkan untuk ikan sakit (sick-fish) sebesar 80%, recall sebesar 100%, dan mean Average Precision (mAP) sebesar 81,67%. Hal ini menunjukkan bahwa model mampu mengklasifikasikan secara akurat ikan sehat dan ikan sakit.

Kata Kunci: penyakit ikan, ikan koi, pembelajaran mesin, YOLOv5, evaluasi model

Abstract – The main reason for higher mortality in aquaculture is fish-related diseases. One type of fish that is highly susceptible to pests and diseases is the koi fish. Early-stage automatic identification of koi fish diseases is an essential step in preventing the spread of infection. Koi fish disease detection can be conducted through various methods, including visual inspection, physical sensors, genetic analysis, image technology and image processing, biosensors and biochips, molecular screening technology, artificial neural networks, and machine learning. The disease detection system in this study uses YOLOv5, due to its several advantages: high accuracy, real-time detection capability, lightweight model, simplicity in training, and being open-source. This research comprises a series of steps, starting from data preparation and model training using YOLOv5, to the evaluation process which measures accuracy, precision, recall, and mean Average Precision (mAP). A 90% accuracy was achieved through the evaluation of the model, precision scores were 83.33% for healthy fish and 80% for sick fish. The model achieved a recall score was 100%,

with mAP score was 81.67%. This model evaluation confirms the accurate detection of both healthy and sick fish.

Keywords: fish diseases, koi fish, machine learning, YOLOv5, model evaluation

1. Pendahuluan

Penyakit ikan adalah masalah yang perlu diatasi dalam dunia perikanan karena biasanya penyakit ini timbul pada saat proses pembibitan dan pembesaran [1]. Masalah pada penyakit ikan merupakan masalah yang serius dikarenakan mudah menyebar di dalam air sehingga dibutuhkan diagnosis yang cepat dan akurat untuk mengontrol penyakit ikan. Penyakit ikan dapat dideteksi dengan berbagai cara yaitu pemeriksaan visual, penggunaan sensor fisik, analisis genetik, teknologi citra dan pengolahan citra, biosensor dan biochips, teknologi saringan molekuler, jaringan saraf tiruan (*artificial neural networks*) dan pembelajaran mesin, serta sistem pemantauan jarak jauh (*remote monitoring systems*).

Beberapa penelitian yang berhubungan dengan deteksi penyakit ikan menggunakan pengolahan citra telah dilakukan sebelumnya. Metode *Principal Component Analysis* (PCA) digunakan untuk identifikasi penyakit ikan yaitu *Epizootic Ulcerative Syndrome*/EUS [2]. Tingkat akurasi hasil deteksi yang diperoleh adalah sebesar $> 90\%$. Pendeteksian penyakit EUS juga diteliti dengan menggunakan beberapa teknik kombinasi, salah satunya kombinasi *Features from Accelerated Segment Test* (FAST), *Principles Component Analysis* (PCA), dan *Neural Network* (NN) yang menghasilkan tingkat akurasi yang lebih baik dibandingkan lainnya yaitu sebesar 86% [3]. Penyakit ikan adalah penyebab utama peningkatan kematian di budidaya perikanan. Identifikasi penyakit ikan secara otomatis pada tahap awal merupakan langkah penting untuk mencegah penyebaran penyakit. Metode yang diajukan untuk deteksi penyakit adalah *Convolutional Neural Network* (CNN) yang digunakan untuk ekstraksi fitur dan klasifikasi sedangkan proses segmentasi menggunakan Distribusi Gaussian [4]. Ada tiga tipe penyakit yang dapat diidentifikasi secara otomatis yaitu *Epizootic Ulcerative Syndrome* (EUS), *Ichthyophthirius* (Ich) dan *Columnaris*. Berdasarkan metode yang diajukan, tingkat akurasi untuk mendeteksi penyakit ikan adalah $> 95\%$. Deteksi penyakit ikan juga dapat dilakukan dengan metode logika fuzzy K-means dan C-means [5]. Tingkat akurasi yang diperoleh sebesar $96,48\%$ untuk K-means clustering dan C-means sebesar $97,9\%$.

Algoritma YOLO adalah metode pendeteksian objek yang menyederhanakan tugas pengenalan objek lebih cepat karena jaringan konvolusional hanya dilewati satu kali dalam mendeteksi objek [6]. Deteksi objek dengan YOLO dilakukan dengan memprediksi probabilitas kelas yang berada di kotak pembatas [7]. YOLO (*You Only Look Once*) merupakan model yang unggul dalam efisiensi dan akurasi untuk deteksi cepat dan pengenalan target [8]. Bounding box merupakan kotak ukuran $S \times S$ yang berasal dari gambar masukan yang menjadi area pendeteksian sebuah objek berdasarkan probabilitas kelas [9].

Berdasarkan penelitian-penelitian sebelumnya maka diajukan sistem deteksi penyakit ikan koi dengan menggunakan YOLOv5. Ikan koi dipilih sebagai objek penelitian dikarenakan ikan koi merupakan ikan hias unggulan di Indonesia sehingga budidaya ikan koi semakin berkembang, baik sebagai hobi maupun sumber penghasilan. Namun, ikan ini termasuk jenis yang rentan terhadap hama dan penyakit sehingga banyak pembudidaya yang masih mengalami kesulitan dalam menangani hal tersebut [10]. YOLOv5 merupakan salah versi dari *You Only Look Once* (YOLO) yang digunakan untuk meningkatkan kecepatan dan akurasi. YOLOv5 adalah algoritma pembelajaran mendalam terancang yang telah menunjukkan kinerja luar biasa dalam tugas deteksi objek [11]. YOLOv5 memiliki beberapa kelebihan antara lain akurasi tinggi, deteksi secara *real time*, model ringan, sederhana dalam training, dan *open source* [12]. YOLOv5 memiliki karakteristik yaitu deteksi secara cepat, efisiensi dan fleksibilitas tinggi dalam pengenalan target [13]. Metode YOLOv5 dapat melakukan pendekatan deteksi objek yang akurat dan cepat [14]. Selain itu, YOLOv5 dapat mengenali objek pada skala yang jauh lebih kecil [15]. Memori yang dibutuhkan YOLOv5 tidak terlalu besar sehingga mengintegrasikan dengan perangkat *Internet of Things* (IoT) menjadi lebih mudah [16]. YOLOv5 dituliskan dengan bahasa

pemrograman Python dimana versi sebelumnya menggunakan bahasa C sehingga mudah untuk diinstal [17]. Penggunaan YOLOv5 dalam penelitian ini menjadi kontribusi penting karena mampu meningkatkan kecepatan dan ketepatan dalam mendeteksi objek [18]. Versi lainnya yaitu algoritma YOLOv5m (DFYOLO) dapat meningkatkan presisi rata-rata dari 94,52% menjadi 99,38% [19].

2. Metode Penelitian

Penelitian deteksi penyakit ikan menggunakan algoritma YOLOv5 yang memiliki beberapa kelebihan antara lain dapat mendeteksi secara cepat, efisiensi dan fleksibilitas tinggi, akurasi tinggi, dapat mendeteksi secara *real time*, model ringan, tidak memerlukan memori yang besar, sederhana dalam training, dan *open source*.

2.1. Komponen Penelitian

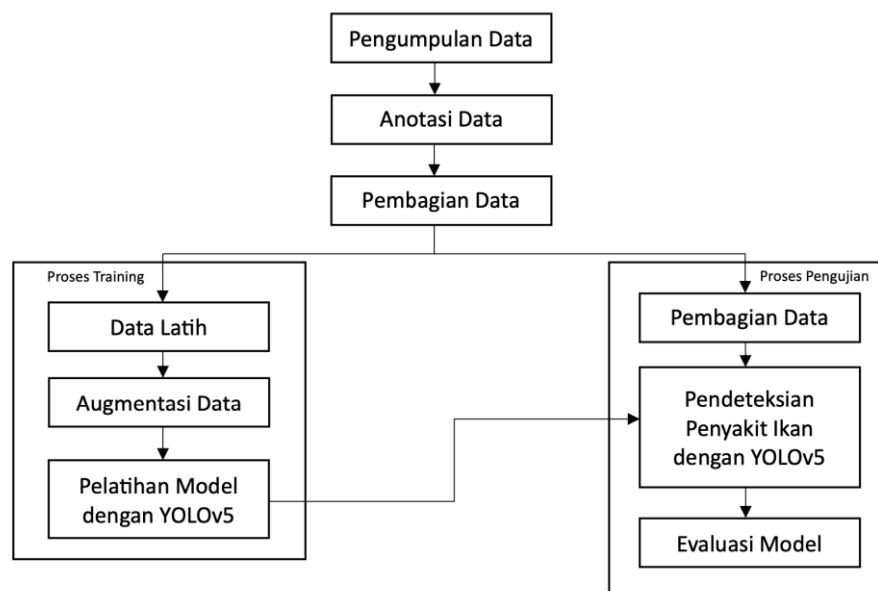
Komponen penelitian terbagi menjadi dua yaitu *software* (perangkat lunak) yang terdiri dari Google Collaboratory, Python, PyTorch, dan repositori YOLOv5 dari GitHub dan *hardware* (perangkat keras) dengan detail spesifikasi tercantum dalam Tabel 1.

Tabel 1. Spesifikasi *hardware*

No.	Nama Variabel	Spesifikasi
1.	CPU	8-core
2.	GPU	7-core
3.	RAM	8 GB
4.	<i>Storage</i>	256 GB
5.	<i>Operating System</i>	macOS Sonoma 14.0

2.2. Tahapan Penelitian

Tahapan dalam penelitian deteksi penyakit ikan mencakup pengambilan dataset, anotasi data, pemisahan data uji, pelatihan model, dan pengujian, yang keseluruhannya ditampilkan dalam Gambar 1.



Gambar 1. Diagram blok alur penelitian

2.2.1. Pengambilan Dataset

Sumber dataset berasal dari ikan koi yang dipelihara secara langsung dan selama proses pemeliharaan tersebut terdapat ikan koi sehat dan terjangkit penyakit. Kemudian dataset dikumpulkan dengan cara mengcapture ikan koi secara langsung dari berbagai *angle* menggunakan *handphone*. Selain itu, juga digunakan gambar ikan koi yang berasal dari internet untuk menambahkan jumlah dataset. Spesifikasi dan konfigurasi device pengambilan gambar ikan ditunjukkan pada Tabel 2.

Tabel 2. Spesifikasi dan konfigurasi *device* pengambilan gambar ikan

No.	Nama Variabel	Spesifikasi
1.	Device	iPhone 15
2.	Zoom	No Zooms
3.	Flash Mode	Off
4.	Image Type	HEIC

2.2.2. Pra-Pemrosesan Data (Anotasi Data)

Frame citra gambar akan dianotasi atau diberi label dengan menggunakan *tool* Roboflow, dimana luaran anotasi menghasilkan file dengan format .txt. Hasil luaran anotasi ini merupakan *bounding box* yang memiliki ukuran tinggi dan lebar dengan koordinat x dan y sebagai pusat yang menginformasikan daftar kelas objek. File dengan format .txt merupakan file anotasi yang menunjukkan parameter label dan *bounding box* dan disimpan bersamaan dengan file berformat .jpg yang mewakili frame citra. Data hasil anotasi menjadi data latih dan data validasi yang berisi objek yang berada di dalam *bounding box* dan telah diberi label.

2.2.3. Pembagian Data

Dataset sebanyak 70% digunakan untuk pelatihan, 20% untuk validasi, dan sisanya 10% digunakan sebagai data pengujian. Data tersebut digunakan dalam proses pendeteksian objek.

2.2.4. Augmentasi Data

Proses augmentasi data bertujuan untuk meningkatkan jumlah citra melalui berbagai teknik modifikasi atau manipulasi terhadap gambar.

2.2.5. Pelatihan Model dengan YOLOv5

Model yang digunakan dalam penelitian adalah YOLOv5 yang diimplementasikan dalam PyTorch untuk proses *training* objek yang akan dideteksi yaitu ikan koi yang telah terinfeksi penyakit.

2.2.6. Pengujian Model dengan YOLOv5

Pada proses pelatihan dihasilkan bobot terbaik yang digunakan untuk proses pengujian model. Selain itu, data dalam pengujian model berbeda dengan data pada proses pelatihan sehingga dapat mengetahui kinerja sistem yang sebenarnya. Bahasa pemrograman Python pada Google Collaboratory digunakan pada proses pelatihan dan pengujian model.

2.2.7. Evaluasi Model

Evaluasi pada tahap ini mencakup penghitungan beberapa nilai yaitu *precision* yang merepresentasikan tingkat ketepatan model dalam menghasilkan prediksi yang benar pada kelas positif, *recall* berfungsi untuk menilai sejauh mana model dapat mengidentifikasi secara lengkap kasus yang sebenarnya positif, *F1-score* adalah metrik yang mengkombinasikan *precision* dan *recall*, serta mAP untuk mengukur sejauh mana ketepatan model dalam mendeteksi dan mengklasifikasikan objek secara benar. Nilai *precision* maupun *recall* dihitung berdasarkan nilai *True Positive* yang menyatakan keberhasilan objek dalam mendeteksi secara akurat sesuai lokasi sebenarnya, *False Positive* merupakan hasil deteksi yang tidak mengandung objek target, dan *False Negative* yang menunjukan objek di dalam gambar tidak dikenali. Nilai *Intersection over Union* (IoU) dihitung dengan membandingkan luas area yang tumpang tindih (*overlap*) dengan

luas gabungan total area (*union*) antara prediksi dan *ground truth* berdasarkan Persamaan (1) – (4) berikut ini:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (1)$$

Pada penelitian ini, *threshold* untuk IoU sebesar 0,5 dengan ketentuan sebagai berikut:

- 1) Objek dikategorikan sebagai *True Positive* (TP) apabila nilai IoU-nya mencapai atau melebihi 0,5
- 2) Objek dikategorikan sebagai *False Positive* (FP) ketika hasil deteksi memiliki nilai IoU tidak melebihi 0,5
- 3) Objek diklasifikasikan sebagai *True Negative* (TN) jika citra yang tidak terdeteksi pada setiap bagian
- 4) Objek dinyatakan sebagai *False Negative* (FN) apabila objek terdapat pada *ground truth*, namun tidak berhasil terdeteksi oleh model

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

Penggambaran kurva menunjukkan nilai *precision* dan *recall*. *Average Precision* (AP) dihitung berdasarkan luas area di bawah kurva untuk masing-masing kelas yang berhasil dideteksi, dan nilai rata-ratanya disebut sebagai *mean Average Precision* (mAP).

3. Hasil dan Pembahasan

Tahapan dalam penelitian deteksi penyakit ini antara lain penyiapan dataset, pelatihan data (training) menggunakan model YOLOv5, dan evaluasi model.

3.1. Penyiapan Dataset

Penyiapan dataset merupakan aspek yang crucial karena menentukan hasil training model.

3.1.1. Pengambilan Dataset

Ikan koi yang telah dirawat sebelumnya di masing-masing bak fiber digunakan sebagai objek dalam penelitian ini. Selama proses pemeliharaan dan pengamatan ikan koi, didapatkan ikan koi dalam keadaan sehat dan sakit. Ikan koi dalam keadaan sakit disebabkan oleh bakteri dan virus seperti *aeromonas* sp, koi herpes virus (khv), dan *argulus* sp. Pada penelitian ini data ikan koi sesuai pada Gambar 2 diklasifikasikan menjadi 2 kelompok yaitu ikan sehat dan ikan sakit. Gambar ikan koi sehat dan ikan sakit dicapture dengan menggunakan handphone dimana hasilnya dalam format HEIC, kemudian dikonversi menjadi format JPG.



(a) Ikan koi sehat

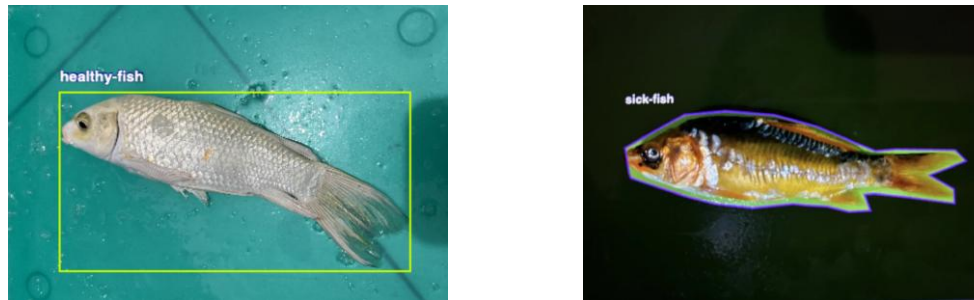


(b) Ikan koi sakit

Gambar 2. Data ikan koi

3.1.2. Proses Anotasi Data

Langkah selanjutnya ada proses anotasi data atau pemberian label dengan menggunakan *tool* Roboflow. Anotasi data atau *labeling* dilakukan dengan cara menandai objek dengan membuat kotak dan koordinat *ground-truth* [20]. Gambar ikan koi diberi *bounding box* dan label untuk menunjukkan kelas objek yang terdiri dari 2 kelas yaitu *healthy-fish* dan *sick-fish* seperti yang ditunjukkan pada Gambar 3. Hasil luaran anotasi menghasilkan file dengan format *.txt*.



Gambar 3. Proses anotasi data

3.1.3. Pembagian Dataset

Gambar ikan koi yang telah dianotasi dengan bounding box dan label dibagi ke dalam tiga bagian dataset: latih (70%), validasi (20%), dan uji (10%). Kemudian dilakukan *preprocessing* dengan 2 tahapan yaitu *Auto-Orient* dan *Resize*. *Auto-Orient* berfungsi menyesuaikan rotasi atau orientasi sebuah objek atau gambar secara otomatis [21]. *Resize* digunakan untuk menyeragamkan ukuran semua data, dimana dalam penelitian ini menggunakan ukuran 640x640.

3.2. Pelatihan Data (Training) Menggunakan Model YOLOv5

Pelatihan data (training) menggunakan Google Colab dengan model YOLOv5. Pada proses ini 70% dataset digunakan untuk data latih sehingga program dapat mendeteksi objek yaitu ikan koi sehat dan sakit. Proporsi 70/30 atau 80/20 merupakan praktik standar dalam pengembangan sistem berbasis pembelajaran mesin, termasuk dalam tugas-tugas seperti deteksi objek [22]. Pemilihan proporsi 70% untuk data latih dalam deteksi objek didasarkan pada praktik umum yang bertujuan untuk memberikan data bagi model untuk belajar dan menyisahkan data untuk evaluasi kinerja. Selain itu, nilai total loss dipengaruhi oleh jumlah data latih. Proses training optimal menghasilkan nilai total loss yang rendah [23]. Pada proses training ini terbagi menjadi 2 kelas yaitu 0 dan 1, dimana 0 menunjukkan kelas *healthy-fish* dan 1 menunjukkan kelas *sick-fish* dan didapatkan hasil seperti Gambar 4.



Gambar 4. Hasil training data

3.3. Evaluasi Model

Evaluasi model yang dilakukan terdiri dari nilai *precision*, *recall*, *accuracy*, *mAP*, dan *confusion matrix*.

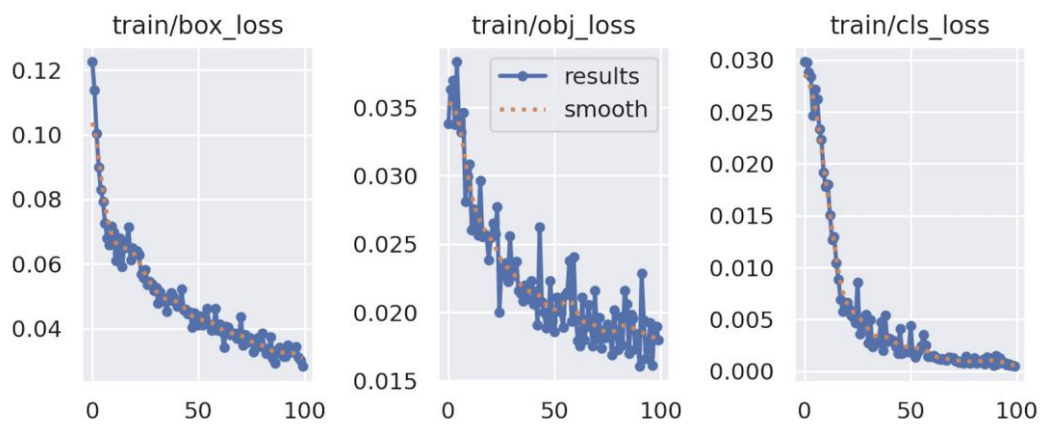
3.3.1. Analisis Kinerja Berdasarkan Epoch

Keberhasilan proses pelatihan ditentukan oleh salah satu faktor penting yaitu Epoch yang berfungsi mengatur sebanyak apakah data akan dilatih. Jumlah epoch mempengaruhi hasil pelatihan dimana terlalu sedikit epoch dapat menyebabkan *underfitting* yang menyebabkan model terlalu fokus pada data pelatihan dan data baru tidak dapat digeneralisasi dengan baik, jika terlalu banyak epoch dapat menyebabkan *overfitting*. Evaluasi model dilakukan dengan menggunakan beberapa jumlah epoch yaitu 50, 100, 150, 200, 250, 300, 350, 400, dan 450 yang nantinya digunakan untuk mengetahui jumlah epoch terbaik berdasarkan kinerja model yang ditentukan berdasarkan nilai *Precision* (P), *Recall* (R), *mAP*₅₀ dan *mAP*₅₀₋₉₅. Perbandingan berbagai epoch pada evaluasi model yang disajikan dalam Tabel 3.

Tabel 3. Evaluasi model pada setiap epoch

Nilai Epoch	P	R	mAP ₅₀	mAP ₅₀₋₉₅
50	0,874	0,925	0,995	0,692
100	0,962	1	0,995	0,788
150	0,949	1	0,995	0,829
200	0,957	1	0,995	0,809
250	0,941	0,994	0,995	0,817
300	0,941	0,994	0,995	0,826

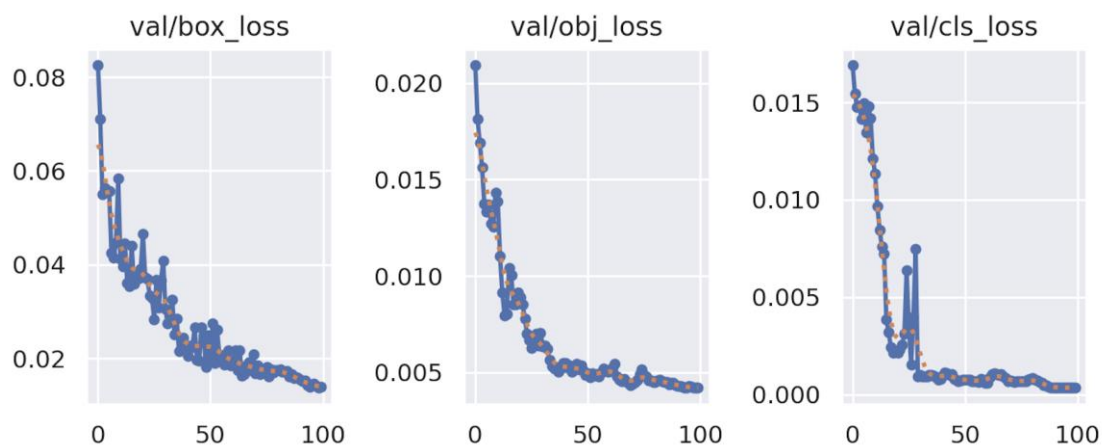
Berdasarkan Tabel 3. epoch 100 menunjukkan kinerja paling baik pada evaluasi model karena memiliki nilai *Precision* (P) sebesar 0,962; *Recall* (R) sebesar 1, dan *mAP*₅₀ sebesar 0,995 yang merupakan nilai tertinggi diantara epoch lainnya. Pada Gambar 5 ditampilkan hasil training untuk epoch ke-100, di mana *box loss* (*box_loss*) adalah 0,0301; *objectness loss* (*obj_loss*) 0,0189; *classification loss* (*cls_loss*) 0,0005; dan total dari ketiga *loss* 0,0495.



Gambar 5. Nilai loss pada hasil traning untuk epoch 100

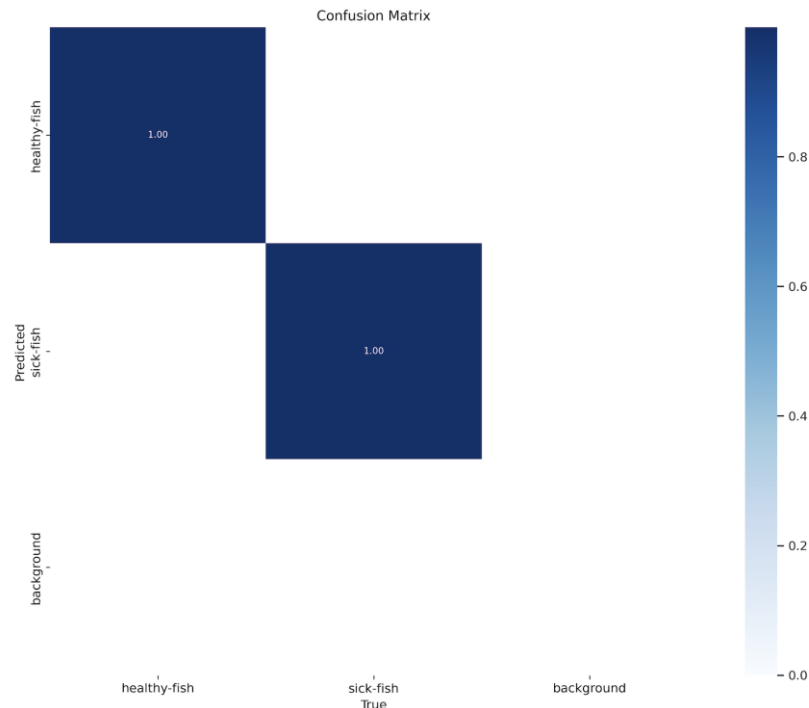
Gambar 5 menunjukkan hubungan antara loss dan epoch dimana epoch berada pada sumbu x sedangkan nilai loss berada pada sumbu y dan grafik biru menunjukkan perubahan loss. Ketika loss semakin berkurang, kemampuan model untuk mendeteksi ikan koi semakin meningkat. Selain itu, berdasarkan grafik hubungan antara loss dan epoch berbanding terbaik dimana semakin besar epoch maka semakin rendah nilai lossnya. Pada Gambar 5, pembentukan model dengan menggunakan dataset menunjukkan hasil yang baik ditandai dengan nilai kedua epoch cenderung turun menuju nol [24].

Hasil validating pada epoch 100 didapatkan *box loss* dengan nilai 0,0137, *objectness loss* dengan nilai 0,0042, dan *classification loss* sebesar 0,0004 dengan total loss sebesar 0,0183. Nilai *total loss* mendekati 0 sehingga model dapat melakukan klasifikasi dengan baik dan memiliki kinerja yang optimal [25] seperti ditunjukkan pada Gambar 6.



Gambar 6. Nilai loss pada hasil validating untuk epoch 100

Selain itu, untuk menguji akurasi klasifikasi model secara visual dapat menggunakan *confusion matrix* dimana semakin besar akurasi di setiap kelas maka semakin dalam warna pada hasil visualisasi seperti ditunjukkan pada Gambar 7. Klasifikasi ikan sehat dan ikan sakit memiliki nilai terbaik yaitu 10. Hal ini menunjukkan bahwa model mampu memprediksi objek dengan baik [24].



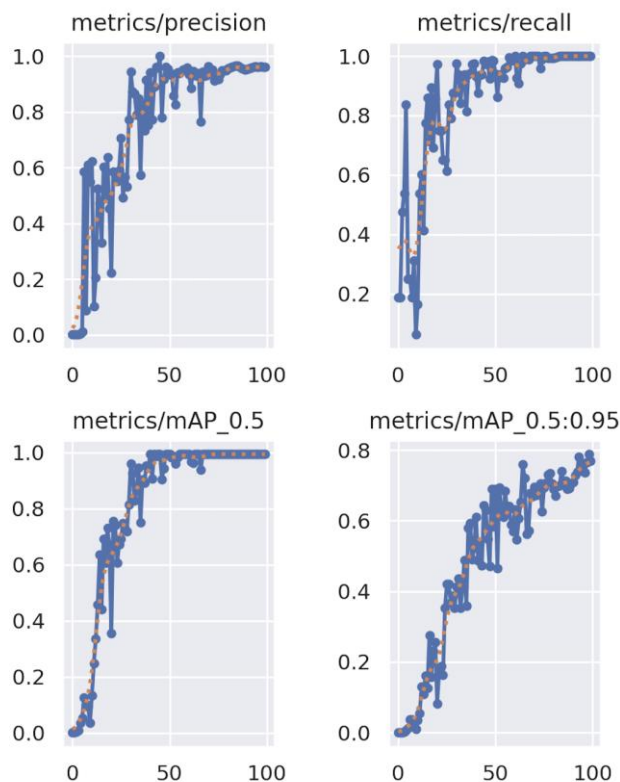
Gambar 7. Confusion matrix untuk epoch 100

Epoch 100 yang memiliki nilai *precision*, *recall*, mAP_{50} , dan mAP_{50-95} yang tertera dalam Tabel 4.

Tabel 4. Perbandingan nilai *precision*, *recall*, dan mAP

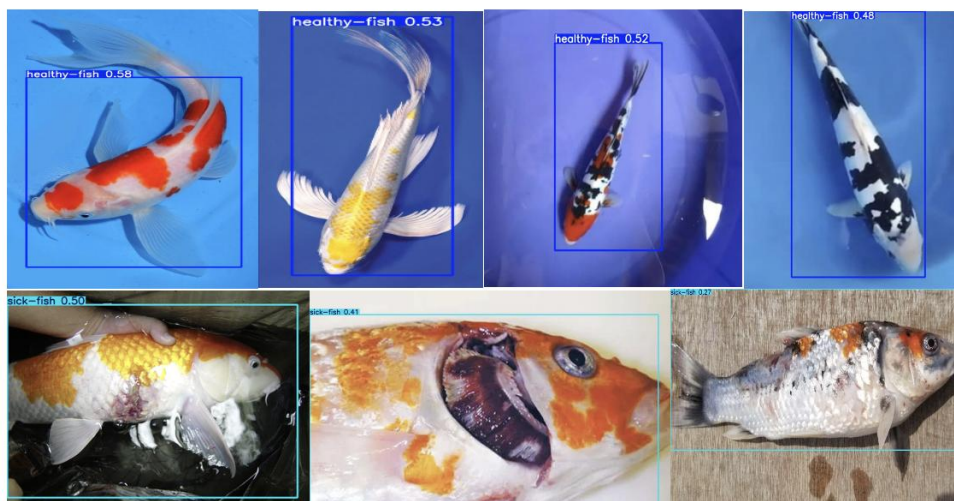
Kelas	<i>Precision</i> (P)	<i>Recall</i> (R)	mAP_{50}	mAP_{50-95}
<i>all</i>	0,962	1	0,995	0,788
<i>healthy-fish</i>	0,995	1	0,995	0,654
<i>sick-fish</i>	0,93	1	0,995	0,923

Precision mengukur seberapa banyak objek yang dapat terdeteksi secara benar. *Precision* memiliki nilai antara 0 dan 1, dimana nilai *precision* lebih tinggi, maka lebih akurat model dalam mendeteksi objek. Berdasarkan Tabel 4 didapatkan nilai *Precision* sebesar 0,962. *Recall* mengukur berapa banyak objek dapat terdeteksi model dari semua objek yang seharusnya dideteksi. *Recall* juga memiliki nilai antara 0 dan 1, dimana nilai *recall* lebih tinggi maka lebih banyak objek yang dapat dideteksi. Hasil penelitian menunjukkan nilai *recall* sebesar 1. Objek yang berada di dalam gambar atau video dapat dikenali oleh model dengan cara mempertimbangkan *precision* maupun *recall* dinyatakan dalam nilai *mean Average Precision* (mAP). Range nilai mAP dari 0 sampai 1, dengan nilai lebih tinggi menunjukkan indikasi performansi model lebih baik dimana nilai yang didapatkan selama penelitian yaitu 0,995 untuk mAP pada ambang 50 dan 0,788 pada ambang 50-95 seperti ditunjukkan pada Gambar 8.

Gambar 8. Nilai *precision*, *recall*, dan mAP untuk epoch 100

3.3.2. Hasil Deteksi Objek

Deteksi objek dilakukan dengan menggunakan weight/bobot paling baik dimana data berupa gambar dengan format .jpg. Berdasarkan hasil deteksi objek didapatkan nilai *accuracy* sebesar 90%, nilai *precision* untuk ikan sehat (*healthy-fish*) sebesar 83,33% sedangkan untuk ikan sakit (*sick-fish*) sebesar 80%, *recall* sebesar 100%, dan *mean Average Percision* (mAP) sebesar 81,67%. Hal ini menunjukkan bahwa model mampu mengklasifikasikan data ikan sehat atau sakit secara benar dan akurat seperti ditunjukkan pada Gambar 9. Beberapa penelitian juga menggunakan metode YOLO dengan versi modifikasi. Penelitian terkait deteksi penyakit ikan di lingkungan bawah air menggunakan metode DFYOLO atau versi modifikasi YOLOv5 dengan penambahan modul *Convolutional Block Attention Module* (CBAM) [26]. Penelitian ini mencapai presisi sebesar 99,75%, recall sebesar 99,31%, dan mAP50 sebesar 99,38%. Penelitian lainnya adalah metode deteksi penyakit ikan berdasarkan *multi-task learning* dengan menggunakan metode YOLO-FD dan menghasilkan akurasi mencapai mAP50 sebesar 94,2% dan mIOU sebesar 79,4% pada dataset Nocardiosis [27]. Selain itu, deteksi penyakit ikan koi menggunakan YOLOv8 dan kompensasi multi-label juga dilakukan dengan menghasilkan total tingkat deteksi adalah 95,36%, dan waktu deteksi gambar adalah 132 frame per detik [28]. Berbeda dari tiga penelitian sebelumnya yang menerapkan versi lanjutan dan modifikasi YOLO untuk menyempurnakan metode terdahulu, penelitian ini menggunakan model YOLOv5.



Gambar 9. Hasil deteksi ikan sehat (*healthy-fish*) dan ikan sakit (*sick-fish*)

4. Kesimpulan

Sistem deteksi penyakit ikan menggunakan YOLOv5 memiliki beberapa kelebihan antara lain tingkat akurasi yang tinggi, dapat mendeteksi secara *real time*, model ringan, sederhana dalam *training*, dan *open source* jika dibandingkan dengan metode Faster R-CNN kurang efisien untuk aplikasi deteksi waktu nyata [29]. Selain itu, metode lainnya seperti SSD (*Single Shot MultiBox Detector*) memiliki akurasi lebih rendah dari YOLOv5, terutama pada objek kecil [30]. Begitu pula dengan versi sebelumnya dari YOLOv5 yaitu YOLOv3 kurang optimal dari sisi kecepatan inference dan kemudahan training [31]. Tahapan dalam sistem ini terdiri dari persiapan data, proses anotasi data atau pemberian label dilanjutkan dengan proses augmentasi data untuk memperbanyak jumlah citra secara otomatis menggunakan tool Roboflow, proses pelatihan (training) data, pengujian dan evaluasi model. Dalam proses training data objek terbagi menjadi 2 kelas yaitu ikan koi sehat (*healthy-fish*) dan ikan koi sakit (*sick-fish*). Nilai *accuracy* yang didapatkan dari hasil pengujian sistem sebesar 90%, nilai *precision* untuk ikan sehat (*healthy-fish*) sebesar 83,33% sedangkan untuk ikan sakit (*sick-fish*) sebesar 80%, *recall* sebesar 100%, dan *mean Average Percision* (mAP) sebesar 81,67%, yang artinya model mampu mengklasifikasikan data ikan sehat atau sakit dengan benar.

Ucapan Terima Kasih

Para peneliti menyampaikan apresiasi atas dukungan pendanaan yang diberikan oleh Pusat Pendidikan Kelautan dan Perikanan melalui kontrak penelitian Nomor 627.21/PPK.PUSDIK/PL.430/X/2024.

Referensi

- [1] Y. J. Hardiko, N. Hidayat, and I. Cholissodin, "Diagnosis penyakit ikan koi menggunakan metode Naive Bayes Classifier," *J. Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 11, pp. 5310 – 5316, 2018.
- [2] H. Chakravorty, R. Paul, and P. Das, "Image processing technique to detect fish disease," *International Journal of Computer Science and Security (IJCSS)*, vol. 9, no. 2, pp. 121–131, 2015.
- [3] S. Malik, T. Kumar and A. K. Sahoo, "Image processing techniques for identification of fish disease," In *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*, 2017, pp. 55-59

- [4] A. Waleed, H. Medhat, M. Esmail, K. Osama, R. Samy and T.M. Ghanim, "Automatic recognition of fish diseases in fish farms," In *14th International Conference on Computer Engineering and Systems (ICCES)*, 2019, pp. 201-206
- [5] J. Sikder, K. I. Sarek dan U. K. Das, "Fish disease detection system: a case study of freshwater fishes of Bangladesh," *Int. J. Adv. Comput. Sci. Appl.(IJACSA)*, 2021, vol. 12, no. 6, pp. 867-871.
- [6] H. M. Lathifah, L. Novamizanti and S. Rizal, "Fast and accurate fish classification from underwater video using you only look once," In *IOP Conference Series: Materials Science and Engineering*, vol. 982, no. 1, p.012003, 2020.
- [7] L. Susanti, N. K. Daulay and B. Intan, "Sistem Absensi Mahasiswa Berbasis Pengenalan Wajah Menggunakan Algoritma YOLOv5," *JURIKOM (Jurnal Riset Komputer)*, vol. 10, no.2, pp. 640-647
- [8] G. Yu, J. Zhang, A. Chen and R. Wan, "Diseased Fish Detection in the Underwater Environment Using an Improved YOLOV5 Network for Intensive Aquaculture," *Fishes*, vol. 8, no. 3, p.169, 2023.
- [9] H. Husnan, C. Fatichah and R. Dikairoo, "Deteksi Objek Menggunakan Metode YOLO dan Implementasinya pada Robot Bawah Air," *J. Teknik ITS*, vol. 12, no. 3, pp. A221-A226, 2023.
- [10] M. Z. Muttaqin, E. Santoso and B. Rahayudi, "Sistem Diagnosis Penyakit Ikan Koi Menggunakan Metode *Forward Chaining* dan *Dempster-Shafer*," *J. Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no.8, pp. 2886-2891, 2018.
- [11] A. Y. Alin, K. Kusrini and K. A. Yauan, "The Effect of Data Augmentation in Deep Learning with Drone Object Detection," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 17, no.3.
- [12] S. M. Nasution and F. M. Dirgantara, "Pedestrian Detection System using YOLOv5 for Advanced Driver Assistance System (ADAS)," *J. RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 3, pp.715-721, 2023.
- [13] W. Li, Z. Zhang, B. Jin and W. Yu, "A Real-Time Fish Target Detection Algorithm Based on Improved YOLOv5," *Journal of Marine Science and Engineering*, vol. 11, no. 3, p.572, 2023.
- [14] Q. Mohti, R. Wahyudi and H. Mustofa, "Penerapan Metode Yolo V5 Dalam Mendeteksi Penyakit Tanaman Buah Naga," In *STAINS (SEMINAR NASIONAL TEKNOLOGI & SAINS)*, vol. 3, no. 1, pp. 318-323, 2024.
- [15] F. Mattins and C. Whidden, "Evaluating multiple YOLO deep learning models for detecting fish," [Online]. Available: https://dcsi.cs.dal.ca/wp-content/uploads/2022/07/DCSI-2022_paper_2008.pdf. [Accessed Jan. 16, 2024]
- [16] N. Hidayat, S. Wahyudi and A. A. Diaz, "Pengenalan individu melalui identifikasi wajah menggunakan metode You Only Look Once (Yolov5)," *UNEJ e-Proceeding*, 2022, pp.85-98.
- [17] J. Zhong, M. Li, J. Qin, Y. Cui, K. Yang and H. Zhang, "Real-time marine animal detection using YOLO-based deep learning networks in the coral reef ecosystem," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 46, pp.301-306, 2022.
- [18] N. P. Mohod, G. H. Deshpande, A.C. Jagdale, V. S. Talokar, I. R. Daga and M. M. Deole, "Identification of Fish Species using YOLOv5," *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 10, no. 4, pp. k503-k510, 2023.
- [19] Z. Wang, H. Liu, G. Zhang, X. Yang, L. Wen and W. Zhao, "Diseased Fish Detection in the Underwater Environment Using an Improved YOLOV5 Network for Intensive Aquaculture," *Fishes*, vol. 8, no. 3, p.169, 2023.
- [20] H. Harmiansyah, R. Fil'aini, Z. Mufidah, N. W. A. Utari, J. Hendra, D. Diptaningsari, N. Wardani, Meidaliyantisya, R. Mawardi, M. A. Mustafid and M. A. Mustafid, "Sistem Smart Detection Penyakit pada Tanaman Kopi Robusta Menggunakan SSD MobileNet V2 sebagai Model Pra-Terlatih." *J. Agrikultura*, vol. 34, no. 1, pp.154-162, 2023.

- [21] M. D. M. Fauzi, T. A. Mudzakir, C. E. Sumawati and J. Indra, "Deteksi Jenis Penyakit Pada Tanaman Padi Menggunakan Yolo V5," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 5, no. 1, pp. 39-48, 2024.
- [22] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [23] S. A. Sabrina and W. F. Al. Maki, "Klasifikasi Penyakit pada Tanaman Kopi Robusta Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *eProceedings of Engineering*, vol. 9, no. 3, 2022.
- [24] H. Harmiansyah, E. T. Oviana, R. Alpaizon, D. P. Khalifah and P. Dwirotnama, "Sistem Deteksi Hama dan Penyakit Tanaman Mangga (*Mangifera indica* L.) Berbasis Deep Learning Menggunakan Model Pra Latih YOLOv5," *J. Agrikultura*, vol. 35, no. 1, pp.151-163, 2024.
- [25] G. Thiodorus, N. K. Daulay and B. Intan, "Sistem Absensi Mahasiswa Berbasis Pengenalan Wajah Menggunakan Algoritma YOLOv5," *JURIKOM (Jurnal Riset Komputer)*, vol. 10, no. 2, pp.640-647, 2023.
- [26] Z. Wang, H. Liu, G. Zhang, X. Yang, L. Wen, and W. Zhao, "Diseased Fish Detection in the Underwater Environment Using an Improved YOLOv5 Network for Intensive Aquaculture," *Fishes*, vol. 8, no. 3, p. 169, 2023. [Online]. Available: <https://www.mdpi.com/2410-3888/8/3/169>. [Accessed May 7, 2025]
- [27] X. Li, S. Zhao, C. Chen, H. Cui, D. Li, and R. Zhao, "YOLO-FD: A Fish Disease Detection Method Based on Multi-task Learning," *Expert Systems with Applications*, vol. 233, p. 121197, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0957417424019523>. [Accessed May 7, 2025]
- [28] T. Yonghua, Z. Zhipeng, L. Sen, L. Xingtong, R. Hongyang, and W. Tengchuan, "Detecting fish diseases using improved YOLOv8 and multi-label compensation," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 40, no. 23, pp. 227-234, 2024. [Online]. Available: <https://www.aeeisp.com/nygcxb/en/article/doi/10.11975/j.issn.1002-6819.202401159>. [Accessed May 7, 2025]
- [29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 2015. [Online]. Available: <https://arxiv.org/pdf/1506.01497> [Accessed May 8, 2025]
- [30] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S.Reed, C.Y.Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," *In European Conference on Computer Vision (ECCV)*, pp. 21-37, 2016. [Online]. Available: <https://link.springer.com/chapter/10.1007/978-3-319-46448-0> [Accessed May 8, 2025]
- [31] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv preprint arXiv:1804.02767*, 2018. [Online]. Available: <https://arxiv.org/abs/1804.02767> [Accessed May 8, 2025]