

IMPLEMENTASI YOLOV8 PADA SISTEM DETEKSI PENYAKIT IKAN MAS KOKI MENGGUNAKAN RASPBERRY PI5

Galih Alif Artanto^{*1}, Zikri Hanafi², Rizwaldi Muhammad Iman³, Arnisa Stefanie⁴

^{1,2,3,4}Teknik Elektro, Universitas Singaperbangsa Karawang. Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat, Indonesia

Keywords: .

Ikan Mas Koki; YOLOv8;
Raspberry Pi 5; Deteksi
Penyakit; Pendeteksian
Objek.

Correspondent Email:

2110631160046@student.unsika.ac.id

Abstrak. Penelitian ini bertujuan untuk mengembangkan sistem pendeteksian penyakit pada ikan mas koki (*Carassius auratus*) menggunakan model YOLOv8 yang diimplementasikan pada perangkat Raspberry Pi 5. Alasan utama penelitian ini adalah pentingnya mengatasi kesulitan pembudidaya dalam mengenali gejala penyakit yang sering menyebabkan kerugian ekonomi. Dataset yang digunakan mencakup 1950 gambar, yang dibagi ke dalam folder train, valid, dan test. Metode penelitian melibatkan pelatihan model dengan 150 epoch menggunakan pustaka Ultralytics. Evaluasi kinerja dilakukan berdasarkan metrik akurasi, presisi, recall, dan F1-score. Hasil menunjukkan bahwa YOLOv8m memiliki akurasi tertinggi sebesar 79.6%, sementara YOLOv8n menunjukkan waktu deteksi tercepat dengan rata-rata 261.04 ms pada video (FPS: 3.58). Sistem ini efektif dalam mendeteksi empat jenis penyakit ikan mas koki dengan tingkat akurasi yang baik, meskipun terdapat kekurangan pada kecepatan model YOLOv8m dan performa pada kelas tertentu seperti goldfish_septicemia. Temuan ini menunjukkan potensi besar untuk pengembangan sistem pendeteksian penyakit portabel yang hemat biaya, dengan peluang untuk meningkatkan akurasi dan efisiensi sistem melalui optimasi model dan perluasan dataset. Hasil penelitian ini diharapkan dapat membantu pembudidaya ikan hias dalam mengurangi risiko kerugian akibat penyakit.



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract. This study aims to develop a disease detection system for goldfish (*Carassius auratus*) using the YOLOv8 model implemented on Raspberry Pi 5. The primary motivation for this research is addressing the challenges faced by breeders in recognizing disease symptoms, which often lead to economic losses. The dataset consisted of 1950 images, divided into train, valid, and test folders. The research method involved training the model over 150 epochs using the Ultralytics library. Performance evaluation was based on accuracy, precision, recall, and F1-score metrics. Results showed that YOLOv8m achieved the highest accuracy of 79.6%, while YOLOv8n demonstrated the fastest detection time with an average of 261.04 ms on video (FPS: 3.58). The system effectively detected four types of goldfish diseases with good accuracy, although it had limitations in the detection speed of YOLOv8m and performance on certain classes like goldfish septicemia. These findings highlight the potential of developing a portable and cost-effective disease detection system, with opportunities to enhance accuracy and efficiency through model optimization and dataset expansion. The research outcomes are expected to assist ornamental fish breeders in reducing losses due to diseases.

1. PENDAHULUAN

Ikan mas koki (*Carassius auratus*) merupakan salah satu jenis ikan hias yang populer di kalangan penggemar dan pecinta ikan hias. Daya tariknya terletak pada keunikan bentuk tubuh, variasi warna yang indah, serta harganya yang relatif terjangkau, sehingga diminati oleh berbagai kalangan, mulai dari kelas atas, menengah, hingga rumah tangga[1]. Ragam jenis dan karakteristik yang dimiliki oleh ikan mas koki membuatnya sangat diminati di pasaran. Berdasarkan data statistik dari Kementerian Kelautan dan Perikanan (KKP), ikan mas koki adalah salah satu komoditas ikan hias yang paling banyak diminati, dengan total produksi mencapai 66.823.000 ekor pada tahun 2010 dan meningkat menjadi 72.997.000 ekor pada tahun 2014[2].

Indonesia memiliki potensi besar untuk mengembangkan komoditas ikan hias, baik untuk pasar domestik maupun ekspor. Dengan keanekaragaman spesies yang dimiliki, Indonesia memiliki keunggulan komparatif dalam produksi ikan hias, termasuk ikan mas koki. Hingga saat ini, perkembangan sektor ikan hias di Indonesia terus menunjukkan peningkatan. Ikan mas koki tidak hanya populer, tetapi juga relatif mudah untuk dibudidayakan serta memiliki nilai ekonomi yang cukup tinggi. Budidaya ikan mas koki dapat dilakukan tanpa memerlukan lahan yang luas, dan siklus reproduksinya yang singkat menambah potensi keuntungan bagi para pembudidaya.

Namun, meskipun mudah dibudidayakan, ikan mas koki tergolong rentan terhadap berbagai penyakit. Banyak orang menganggap ikan ini mudah mati, tetapi penyebab utama kematiannya bukan hanya karena kekuatan fisiknya yang lemah, melainkan juga dipengaruhi oleh kesulitan dalam mengidentifikasi gejala penyakit dengan akurat. Pembudidaya ikan mas koki sering menghadapi kesulitan dalam mengenali penyakit ikan ini, terutama karena banyaknya kemiripan gejala antar-penyakit yang menyerang ikan mas koki[3].

Pengamatan terhadap gejala penyakit pada ikan mas koki sebaiknya dilakukan setiap hari, dengan memperhatikan perubahan perilaku, seperti ikan yang kurang aktif berenang, lebih sering berada di dasar kolam atau permukaan,

dan penurunan nafsu makan[3]. Selain itu, kondisi fisik ikan perlu dipantau, misalnya adanya butiran kecil menyerupai garam pada tubuh atau ekor, produksi lendir berlebih, sirip yang menutup, kerusakan pada sirip atau ekor, munculnya warna putih di tepi ekor, serta luka atau bercak kemerahan pada tubuh, sirip, atau ekor. Kemiripan gejala pada berbagai penyakit ini menyebabkan pembudidaya kesulitan dalam mengenali jenis penyakit secara akurat, yang pada akhirnya dapat meningkatkan risiko kematian ikan[3].

Metode pendeteksian penyakit pada ikan mas koki saat ini masih menggunakan cara-cara tradisional, seperti observasi langsung melalui pemeriksaan fisik dan pengamatan visual untuk mengidentifikasi gejala penyakit. Namun, seiring perkembangan zaman, kemajuan dalam bidang kecerdasan buatan dan pengolahan citra digital telah membuka peluang baru untuk meningkatkan efisiensi dan akurasi dalam mendeteksi penyakit pada ikan mas koki. Dalam penelitian ini, penulis bertujuan untuk membangun sistem pendeteksian penyakit ikan mas koki menggunakan metode You Only Look Once versi 8 (YOLOv8), sebuah model deteksi objek berbasis deep learning.

YOLO adalah algoritma yang menggunakan Convolutional Neural Network (CNN) sebagai inti dari metode deteksi objeknya, yang mampu melakukan identifikasi dan klasifikasi objek dalam gambar secara cepat dan akurat[4]. Dalam penelitian ini, kami akan mengevaluasi kinerja model YOLOv8 dalam mendeteksi gejala-gejala penyakit pada ikan mas koki dengan mengukur akurasi prediksi.

Selain itu, sistem ini akan diimplementasikan menggunakan perangkat Raspberry Pi 5, sebuah komputer single-board generasi terbaru yang menawarkan kinerja tinggi dengan ukuran yang ringkas. Raspberry Pi 5 memiliki prosesor yang cepat, dukungan RAM yang besar, serta kemampuan pengolahan grafis yang baik, sehingga sangat cocok untuk mendukung sistem portabel dan hemat biaya[5]. Dengan menggunakan Raspberry Pi 5, sistem pendeteksian ini diharapkan dapat diterapkan secara portabel dan hemat biaya, sehingga lebih mudah diakses oleh pembudidaya ikan mas koki. Diharapkan penelitian ini dapat memberikan kontribusi dalam meningkatkan deteksi penyakit pada ikan mas koki dan menjadi dasar pengembangan solusi yang lebih

canggih dan efektif dalam penanganan penyakit pada ikan hias.

2. TINJAUAN PUSTAKA

2.1. Raspberry Pi5

Raspberry Pi 5 digunakan sebagai platform deployment untuk model YOLOv8 dalam mendeteksi penyakit ikan mas koki. Raspberry Pi adalah komputer single-board yang terjangkau, kompak, dan mampu menjalankan algoritma pembelajaran mesin atau pembelajaran mendalam tanpa memerlukan koneksi internet atau komputer tambahan. Platform ini memungkinkan komputasi tepi (edge computing) yang sangat efisien untuk perangkat portabel [6].

2.2. Deteksi Objek

Deteksi dan pengenalan objek adalah tugas visi komputer penting yang digunakan untuk mendeteksi contoh objek visual dari kelas tertentu, misalnya manusia, hewan, mobil, atau bangunan dalam gambar digital seperti foto atau bingkai video. Tujuan dari deteksi objek adalah untuk mengembangkan model komputasi yang menyediakan informasi paling mendasar yang dibutuhkan oleh aplikasi visi komputer. Teknik untuk deteksi objek termasuk ke dalam pendekatan berbasis machine learning dan pendekatan berbasis *deep learning*. Metodologi deep learning mampu melakukan deteksi objek dengan algoritma yang dirancang untuk melakukan deteksi objek berdasarkan dua pendekatan, yaitu deteksi objek satu tahap dan deteksi objek dua tahap. Detektor satu tahap memiliki kecepatan inferensi yang tinggi, sementara detektor dua tahap memiliki lokalisasi dan akurasi pengenalan yang tinggi [7,8].

2.3. You Look Only Once (YOLO)

YOLO (You Only Look Once) adalah metode deteksi objek berbasis visi komputer dan pembelajaran mendalam yang dirancang untuk mendeteksi dan menentukan lokasi objek dalam gambar atau video dengan efisiensi tinggi [9]. Algoritma ini mendukung deteksi real-time dengan kecepatan dan akurasi yang optimal, sehingga sangat cocok untuk aplikasi yang memerlukan respons cepat.

2.4. Confusion Matriks

Confusion matrix adalah representasi matriks dari hasil prediksi model klasifikasi terhadap data uji. Matriks ini memuat empat elemen utama, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Elemen-elemen ini memungkinkan evaluasi performa model secara rinci, tidak hanya berdasarkan akurasi, tetapi juga dengan mempertimbangkan aspek lain seperti sensitivitas dan spesifisitas [10].

Dari elemen-elemen tersebut, beberapa metrik evaluasi utama dapat dihitung:

2.4.1 Akurasi

Nilai akurasi dibutuhkan untuk mengukur proporsi prediksi yang benar terhadap keseluruhan data, dirumuskan sebagai:

$$\text{Akurasi} = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

2.4.2 Presisi (Precision)

Nilai presisi dibutuhkan untuk mengukur proporsi prediksi positif yang benar terhadap total prediksi positif, dirumuskan sebagai:

$$\text{Presisi} = \frac{TP}{TP+FP} \quad (2)$$

2.4.3 Recall (Sensitivitas)

Nilai recall dibutuhkan untuk mengukur proporsi data positif yang berhasil diklasifikasikan dengan benar, dirumuskan sebagai:

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

2.4.4 F1-Score

Nilai F1-score merupakan penggabungan nilai *presisi* dan *recall* dalam satu metrik harmonis, dirumuskan sebagai:

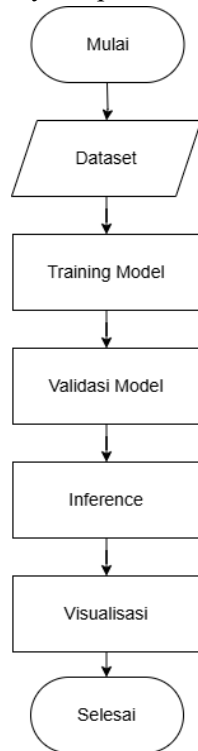
$$F1 - score = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \quad (4)$$

Konsep-konsep ini penting untuk mengevaluasi performa model klasifikasi dalam situasi yang berbeda, seperti ketidakseimbangan kelas atau kebutuhan untuk meminimalkan kesalahan tertentu (contohnya, meminimalkan FN pada aplikasi medis). Evaluasi berbasis *confusion matrix* memberikan wawasan yang lebih menyeluruh dibandingkan hanya menggunakan akurasi.

3. METODE PENELITIAN

3.1. Alur Penelitian

Secara umum, alur metode yang akan dibuat dijelaskan dengan flowchat pada gambar 1. Dalam metode penelitian ini dijabarkan tahapan – tahapan yang akan dilakukan dalam proses pendeteksian penyakit pada ikan mas koki.



Gambar 1. Alur Penelitian

3.1.1. Training Model

Pelatihan model adalah tahap awal dalam pengembangan sistem pembelajaran mesin, di mana model dilatih menggunakan data latih untuk mempelajari pola dan hubungan dalam data tersebut. Proses ini melibatkan penyesuaian parameter model untuk meminimalkan kesalahan prediksi. Misalnya, dalam konteks YOLOv8, pelatihan dilakukan dengan menetapkan jumlah epoch, ukuran batch, dan ukuran gambar tertentu untuk mengoptimalkan kinerja deteksi objek [11].

3.1.2. Validasi Model

Validasi model adalah proses evaluasi kinerja model yang telah dilatih menggunakan data yang tidak digunakan selama pelatihan, biasanya disebut sebagai data validasi. Tujuan dari validasi adalah untuk mengukur

kemampuan generalisasi model terhadap data baru dan untuk mendeteksi masalah seperti overfitting atau underfitting [11].

3.1.3. Inference

Inferensi adalah tahap di mana model yang telah dilatih digunakan untuk membuat prediksi atau klasifikasi pada data baru. Dalam konteks deteksi penyakit pada ikan mas koki menggunakan YOLOv8, inferensi melibatkan penerapan model pada gambar atau video untuk mengidentifikasi dan mengklasifikasikan penyakit yang muncul. Proses ini harus efisien dan cepat, terutama ketika diterapkan pada perangkat dengan sumber daya terbatas seperti Raspberry Pi 5 [11].

3.1.4. Visualisasi

Visualisasi adalah proses penyajian hasil model dalam bentuk yang dapat dipahami oleh manusia, seperti grafik, diagram, atau tampilan interaktif. Dalam proyek deteksi penyakit ikan, visualisasi dapat berupa penandaan area yang terdeteksi sebagai penyakit pada gambar atau video, yang membantu pengguna dalam memahami hasil deteksi model [11].

3.2. Dataset

Dataset adalah kumpulan dari objek dan atributnya, di mana atribut merepresentasikan sifat atau karakteristik dari suatu objek. Dataset gambar ikan mas koki yang digunakan dalam penelitian ini diperoleh dari berbagai sumber yang berfokus pada penyakit ikan hias, khususnya ikan mas koki. Dalam penelitian ini, dataset yang digunakan memiliki total 1950 gambar ikan mas koki. Dataset ini telah diorganisasi ke dalam folder yang terbagi menjadi dua bagian, yaitu *train* sebanyak 1800 gambar, *valid* sebanyak 100 gambar dan *testing* sebanyak 50 gambar [12]. Setiap gambar memiliki label yang menunjukkan pola untuk proses deteksi ke dalam beberapa kelas yang mewakili berbagai kondisi penyakit pada ikan mas koki. Jumlah dataset pada setiap kelas adalah sebagai berikut:

Tabel 1. Label pada folder *Train*

Kelas	Jumlah label
<i>Goldfish</i>	642
<i>Goldfish_dropsy</i>	462
<i>Goldfish_ich</i>	456

<i>Goldfish_septicemia</i>	468
----------------------------	-----

Tabel 2. Label pada folder *Valid*

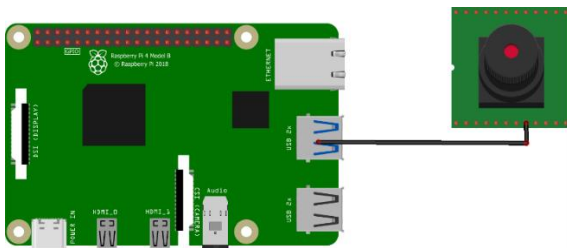
Kelas	Jumlah label
<i>Goldfish</i>	27
<i>Goldfish_dropsy</i>	29
<i>Goldfish_ich</i>	28
<i>Goldfish_septicemia</i>	30

Tabel 3. Label pada folder *Test*

Kelas	Jumlah label
<i>Goldfish</i>	15
<i>Goldfish_dropsy</i>	9
<i>Goldfish_ich</i>	21
<i>Goldfish_septicemia</i>	17

3.3. Rancangan Alat

Pada rancangan alat ini, Raspberry Pi 5 dihubungkan dengan modul kamera sebagai input utama untuk menangkap gambar ikan mas koki secara langsung. Rangkaian ini dirancang sedemikian rupa untuk memaksimalkan efisiensi perangkat keras. Proses pendeteksian dimulai dengan modul kamera menangkap gambar, yang kemudian diproses oleh model YOLOv8 yang sudah di-deploy di Raspberry Pi 5. Dengan kapasitas komputasi Raspberry Pi, sistem dapat mendeteksi empat jenis penyakit ikan secara real-time dengan fleksibilitas antara kecepatan deteksi dan akurasi.



Gambar 2. Rancangan Alat

4. HASIL DAN PEMBAHASAN

Pengujian dilakukan dengan menggunakan *library Ultralytics* untuk melatih (*training*) model YOLOv8 pada dataset yang digunakan. Pustaka *Ultralytics* ini menyediakan kemudahan dalam menerapkan model deteksi objek berbasis YOLO secara efisien dan cepat. Proses pelatihan model YOLOv8 dilakukan

dengan nilai *epoch* sebanyak 150, nilai *batch* sebesar 4, serta ukuran gambar (*imgsz*) sebesar 640 piksel, sebagaimana ditunjukkan pada Gambar 6. Nilai *epoch* menunjukkan jumlah iterasi pelatihan yang harus dijalankan oleh model, sedangkan nilai *batch* merujuk pada jumlah gambar yang dilatih dalam satu waktu (*batch size*). Dengan demikian, dapat diartikan bahwa model ini dilatih menggunakan 4 gambar dalam setiap iterasi, dengan total pelatihan yang dilakukan sebanyak 150 kali.

Pengujian model akan dilakukan dengan menggunakan tiga jenis model YOLOv8, yaitu YOLOv8n (nano), YOLOv8s (small), dan YOLOv8m (medium). Ketiga model ini memiliki perbedaan dalam jumlah parameter dan ukuran model, yang memungkinkan evaluasi terhadap kinerja masing-masing model dalam mendeteksi objek pada dataset yang sama. Pengujian dilakukan pada 100 gambar yang diambil dari folder *valid*. Pengujian ini bertujuan untuk mengevaluasi performa model berdasarkan metrik evaluasi utama, yaitu akurasi, presisi, recall, dan F1-score. Metrik-metrik ini memberikan gambaran yang komprehensif mengenai kemampuan model dalam mendeteksi dan mengklasifikasikan objek secara akurat.

Tabel 4. Nilai Akurasi Model

Model	Akurasi
YOLOv8n	0.792
YOLOv8s	0.760
YOLOv8m	0.796

Tabel 5. Nilai Presisi Model

Model	Class	Presisi
YOLOv8n	<i>Goldfish</i>	0.933
	<i>Goldfish_dropsy</i>	0.800
	<i>Goldfish_ich</i>	0.826
	<i>Goldfish_septicemia</i>	0.607
YOLOv8s	<i>Goldfish</i>	0.960
	<i>Goldfish_dropsy</i>	0.760
	<i>Goldfish_ich</i>	0.882
	<i>Goldfish_septicemia</i>	0.545
YOLOv8m	<i>Goldfish</i>	0.889
	<i>Goldfish_dropsy</i>	0.870
	<i>Goldfish_ich</i>	0.941
	<i>Goldfish_septicemia</i>	0.643

Tabel 6. Nilai Recall Model

Model	Class	Recall
-------	-------	--------

YOLOv8n	<i>Goldfish</i>	0.737
	<i>Goldfish dropsy</i>	0.833
	<i>Goldfish ich</i>	0.792
	<i>Goldfish septicemia</i>	0.850
YOLOv8s	<i>Goldfish</i>	0.667
	<i>Goldfish dropsy</i>	0.864
	<i>Goldfish ich</i>	0.652
	<i>Goldfish septicemia</i>	0.947
YOLOv8m	<i>Goldfish</i>	0.774
	<i>Goldfish dropsy</i>	0.909
	<i>Goldfish ich</i>	0.667
	<i>Goldfish septicemia</i>	0.857

Tabel 6. Nilai F1-score Model

Model	Class	F1-score
YOLOv8n	<i>Goldfish</i>	0.824
	<i>Goldfish dropsy</i>	0.816
	<i>Goldfish ich</i>	0.809
	<i>Goldfish septicemia</i>	0.708
YOLOv8s	<i>Goldfish</i>	0.787
	<i>Goldfish dropsy</i>	0.809
	<i>Goldfish ich</i>	0.750
	<i>Goldfish septicemia</i>	0.692
YOLOv8m	<i>Goldfish</i>	0.828
	<i>Goldfish dropsy</i>	0.889
	<i>Goldfish ich</i>	0.780
	<i>Goldfish septicemia</i>	0.735

Berdasarkan hasil pengujian, performa masing-masing model ditunjukkan pada Tabel 4. Model YOLOv8m memiliki akurasi tertinggi sebesar 79.6%, diikuti oleh YOLOv8n dengan akurasi 79.2%, dan YOLOv8s dengan akurasi 76.0%. Dalam hal presisi, model YOLOv8n menunjukkan nilai tertinggi untuk kelas "goldfish" (93.3%) dan "goldfish_dropsy" (80.0%), sementara YOLOv8m memiliki presisi terbaik untuk kelas "goldfish_ich" (94.1%) dan "goldfish_septicemia" (64.3%).

Model YOLOv8s cenderung unggul dalam recall untuk kelas "goldfish_septicemia" (94.7%), tetapi performa recall pada kelas lain lebih rendah dibandingkan YOLOv8m. Dari segi F1-score, YOLOv8m memberikan nilai terbaik untuk kelas "goldfish_dropsy" (88.9%) dan "goldfish_ich" (78.0%), sementara YOLOv8n memberikan hasil terbaik untuk kelas "goldfish" (82.4%).

Hasil ini menunjukkan bahwa model YOLOv8m lebih seimbang dalam mendeteksi berbagai kelas penyakit, meskipun model YOLOv8n dan YOLOv8s juga memiliki

keunggulan spesifik pada beberapa metrik evaluasi tertentu.

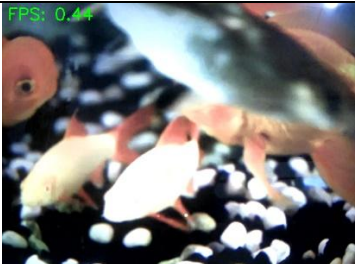
Pengujian lebih lanjut dilakukan untuk mengevaluasi performa model YOLOv8 pada perangkat Raspberry Pi 5 menggunakan input video dan webcam. Hasil pengujian ditunjukkan pada dua tabel untuk memberikan gambaran lengkap mengenai performa model YOLOv8 pada Raspberry Pi 5.

Tabel 7. Proses Deteksi Objek Model YOLOv8 pada Video

Model	Video
YOLOv8n	
YOLOv8s	
YOLOv8m	

Tabel 8. Proses Deteksi Objek Model YOLOv8 pada Webcam

Model	Webcam
YOLOv8n	

YOLOv8s	
YOLOv8m	

Perbedaan kecepatan video yang berjalan di antara model YOLOv8 disebabkan oleh variasi waktu deteksi masing-masing model. Model dengan jumlah parameter yang lebih besar, seperti YOLOv8m, cenderung membutuhkan waktu deteksi lebih lama dibandingkan model yang lebih ringan, seperti YOLOv8n. Hal ini memengaruhi *frame per second* (FPS) yang diperoleh, di mana model yang lebih ringan memiliki FPS lebih tinggi. Seperti yang ditunjukkan pada tabel berikut.

Tabel 9. Perbandingan Kecepatan Deteksi Model YOLOv8 pada Media Video

Model	Video	
	Kecepatan deteksi	FPS didapat
YOLOv8n	261,04 ms	3,58
YOLOv8s	822,48 ms	1,35
YOLOv8m	1784,12 ms	0,51

Tabel 10. Perbandingan Kecepatan Deteksi Model YOLOv8 pada Media Webcam

Model	Webcam	
	Kecepatan deteksi	FPS didapat
YOLOv8n	326,94 ms	3,13
YOLOv8s	940,06 ms	1,10
YOLOv8m	2240,68 ms	0,44

Berdasarkan hasil pengujian yang dilakukan pada Raspberry Pi 5 menggunakan beberapa model YOLOv8, dapat disimpulkan bahwa model YOLOv8n menunjukkan performa terbaik untuk deteksi baik pada input video

maupun webcam. Pada pengujian menggunakan video, YOLOv8n memiliki kecepatan deteksi tercepat, yaitu 261,04 ms, dengan frame per second (FPS) sebesar 3,58. Sementara itu, pada input dari webcam, model ini mencatat waktu deteksi sebesar 326,94 ms dengan FPS sebesar 3,13. Performa ini menunjukkan bahwa YOLOv8n adalah pilihan yang efisien untuk aplikasi real-time, khususnya pada perangkat dengan kapasitas komputasi terbatas seperti Raspberry Pi 5.

Di sisi lain, YOLOv8s dan YOLOv8m membutuhkan waktu deteksi yang lebih lama. YOLOv8s mencatat waktu deteksi sebesar 822,48 ms pada video dan 940,06 ms pada webcam, dengan FPS masing-masing sebesar 1,35 dan 1,10. Sementara itu, YOLOv8m memiliki waktu deteksi yang jauh lebih lama, yaitu 1784,12 ms pada video dan 2240,68 ms pada webcam, dengan FPS hanya sebesar 0,51 dan 0,44. Model YOLOv8s dan YOLOv8m mungkin lebih cocok untuk skenario di mana akurasi lebih diutamakan dibandingkan kecepatan, atau untuk penggunaan pada platform dengan kapasitas komputasi yang lebih tinggi.

Dari hasil ini, dapat disimpulkan bahwa meskipun model YOLOv8n memiliki waktu deteksi tercepat, model ini cenderung kurang akurat dibandingkan YOLOv8m. Model YOLOv8s berada di tengah-tengah dengan keseimbangan antara kecepatan deteksi dan performa akurasi. Oleh karena itu, pemilihan model tergantung pada kebutuhan spesifik, apakah prioritas utama adalah kecepatan atau akurasi.

Berdasarkan pengujian yang sudah dilakukan, dapat diambil kesimpulan bahwa model YOLOv8 menunjukkan perbedaan signifikan antara efisiensi deteksi (tabel kecepatan) dan akurasi deteksi (tabel confusion matrix) untuk masing-masing kategori. YOLOv8n, meskipun memiliki kecepatan deteksi terbaik dengan waktu deteksi tercepat (261,04 ms pada video dan 326,94 ms pada webcam) dan FPS tertinggi, memiliki tingkat akurasi keseluruhan yang lebih rendah dibandingkan dengan YOLOv8m, yaitu 0,792. Hal ini mencerminkan bahwa YOLOv8n lebih cocok digunakan untuk aplikasi real-time di mana kecepatan menjadi prioritas utama, meskipun terjadi sedikit kompromi pada akurasi, terutama pada kelas seperti

goldfish_septicemia, yang memiliki F1-Score terendah sebesar 0,708.

Di sisi lain, YOLOv8m, meskipun memiliki waktu deteksi terlama (1784,12 ms pada video dan 2240,68 ms pada webcam) dan FPS terendah, menunjukkan performa akurasi yang lebih tinggi dibandingkan YOLOv8n dan YOLOv8s. Model ini mencapai F1-Score terbaik untuk kelas goldfish_dropy (0,889) dan memiliki tingkat presisi serta recall yang cukup konsisten di seluruh kelas. Oleh karena itu, YOLOv8m lebih cocok untuk digunakan dalam aplikasi yang memprioritaskan akurasi deteksi, meskipun dengan pengorbanan kecepatan.

Sementara itu, YOLOv8s berada di antara keduanya dengan akurasi keseluruhan terendah (0,760), namun menunjukkan keseimbangan antara presisi dan recall di beberapa kelas. Model ini dapat menjadi pilihan kompromi jika diperlukan keseimbangan antara kecepatan dan akurasi.

Tabel 11. Hasil Deteksi

Video	Webcam
	
	
	

Secara keseluruhan, hasil pengujian menunjukkan bahwa pemilihan model YOLOv8 harus didasarkan pada kebutuhan spesifik aplikasi. Jika fokusnya pada pemrosesan real-time dengan perangkat keras terbatas seperti Raspberry Pi 5, YOLOv8n adalah pilihan terbaik. Namun, untuk aplikasi di mana akurasi menjadi prioritas utama, terutama dalam mendeteksi penyakit seperti

goldfish_dropy dan goldfish_ich, YOLOv8m memberikan hasil yang lebih baik meskipun memiliki kecepatan yang lebih lambat.

5. KESIMPULAN

Sistem deteksi penyakit ikan mas koki berbasis YOLOv8 berhasil dikembangkan menggunakan Raspberry Pi 5 dengan dataset berjumlah 1950 gambar yang dilatih menggunakan library Ultralytics, dengan parameter pelatihan 150 *epoch*, *batch size* 4, dan ukuran gambar 640 piksel. Hasil pengujian menunjukkan bahwa model YOLOv8m memiliki akurasi tertinggi (79,6%) dan performa terbaik untuk mendeteksi kelas penyakit seperti goldfish_dropy (F1-Score: 0,889) dan goldfish_ich (F1-Score: 0,780), sedangkan YOLOv8n unggul dalam kecepatan deteksi dengan waktu rata-rata 261,04 ms pada video (FPS: 3,58) dan 326,94 ms pada webcam (FPS: 3,13), menjadikannya pilihan ideal untuk aplikasi *real-time*. YOLOv8s menawarkan keseimbangan antara kecepatan dan akurasi, dengan performa cukup baik di beberapa kelas. Meskipun demikian, terdapat kekurangan berupa waktu deteksi yang lama pada YOLOv8m dan performa yang kurang optimal untuk kelas tertentu, seperti goldfish_septicemia. Sistem ini menawarkan keunggulan dalam mendeteksi empat jenis penyakit dengan akurasi baik dan mudah diimplementasikan pada perangkat portabel. Pengembangan di masa depan dapat mencakup optimasi waktu deteksi, peningkatan variasi dataset untuk generalisasi lebih baik, dan penerapan pada jenis ikan hias lain guna memperluas cakupan aplikasi, dengan pemilihan model yang disesuaikan dengan kebutuhan spesifik, baik kecepatan maupun akurasi.

DAFTAR PUSTAKA

- [1] M. T. Barkah, Abubakar, dan L. N. Azkiya, "Strategi Pengembangan Usaha Budidaya Ikan Hias Koki (*Carrasius Auratus*) di Desa Cimuning Kecamatan Mustika Jaya Kota Bekasi," *Jurnal Ilmiah Mahasiswa Agroinfo Galuh*, pp. 446–455, 2022.
- [2] M. D. Fimarsyah dan Farikhah, "Kelas Kropyokan Ikan Koki (*Carassius auratus*),"

- Jurnal Perikanan dan Kelautan*, pp. 28–37, 2024.
- [3] M. H. Wa Kafa, N. Hidayat, dan I. Kholisodin, "Diagnosis Penyakit Ikan Mas Koki Menggunakan Metode Naïve Bayes," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 472–480, 2019.
 - [4] D. N. Alfarizi, R. A. Pangestu, D. Aditya, M. A. Setiawan, dan P. Rosyani, "Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis," *Jurnal AI dan SPK: Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 1, pp. 54–63, 2023.
 - [5] J. W. W. Saputra, M. A. Naufal, dan O. A. Putra, "Implementasi YOLO V8 pada prototipe autonomous underwater robot berbasis Raspberry Pi 5 guna menanggulangi pencemaran sampah plastik di daerah perairan," *Jurnal Syntax Admiration*, vol. 5, no. 11, pp. 4483–4492, 2024.
 - [6] Md Nahiduzzaman, R. Sarmun, A. Khandakar, M. A. A. Faisal, M. S. Islam, M. K. Alam, T. Rahman, N. Al-Emadi, M. Murugappan, dan M. E. H. Chowdhury, "Deep learning-based real-time detection and classification of tomato ripeness stages using YOLOv8 on raspberry Pi," *Engineering Research Express*, vol. 7, no. 1, pp. 015219, 2025. [Online]. Available: <https://doi.org/10.1088/2631-8695/ada720>.
 - [7] A. Arnita, F. Marpaung, F. Aulia, N. S. S.Kom, dan R. C. N. S.Kom, *Computer Vision dan Pengolahan Citra Digital*. Surabaya: Pustaka Aksara, 2022.
 - [8] H. Husnan, C. Fatichah, dan R. Dikairono, "Deteksi Objek Menggunakan Metode YOLO dan Implementasinya pada Robot Bawah Air," *Jurnal Teknik ITS*, vol. 12, no. 3, pp. A221, 2023.
 - [9] M. F. Golfantara, "Penggunaan Algoritma YOLO V8 untuk Identifikasi Rempah-Rempah," *Jurnal Informatika dan Teknik Elektro Terapan (JITET)*, vol. 12, no. 351, pp. 1–9, 2024.
 - [10] R. R. Wijayanti, "Confusion Matrix," dalam *Machine Learning*, edisi ke-1, Padang: PT Global Eksekutif Teknologi, 2023, pp. 141–151.
 - [11] S. Shalev-Shwartz dan S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge, U.K.: Cambridge University Press, 2014. [Online]. Available: <http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning>.
 - [12] Universitas Singaperbangsa Karawang, "gws Dataset," *Roboflow Universe*. [Online]. Available: <https://universe.roboflow.com/universitas-singaperbangsa-karawang/gws>. [Accessed: May 27, 2025].