

Penggunaan YOLOv8 untuk Deteksi Penyakit Daun Kopi

Marcelino Bitra¹, Christine Dewi^{2*}

^{1,2}Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Salatiga, Jawa Tengah, Indonesia

E-mail: marcelinobitra@gmail.com¹, christine.dewi@uksw.edu²

Abstract

One of the products of plantation with a significant role in economic activities in Indonesia is coffee. But, coffee production in Indonesia is experienced a decline, where one of the causes is pest and disease attacks. Artificial intelligence can be a solution to help farmers detect diseases in coffee plants using object detection algorithm. This research uses the YOLOv8 object detection algorithm to carry out detection of the state and diseases of coffee plant leaves which are divided into four classifications, namely miner, rust, phoma and healthy. The research was conducted in three experimental scenarios which were differentiated based on a comparison of data distribution in the test set, validation set, and test set, where in sequence of train, validation, and test, the first scenario had a comparison of 80:10:10, the second scenario 70: 15:15, and third scenario 70:20:10. The research process using the YOLOv8s model got a model with the best performance results in data comparison of 70% train set, 20% validation set, and 10% test set. The best performing model has a mAP value of 97.8%, precision 95.2%, recall 96.6%, and f1-score 96%.

Keywords: YOLOv8, Object Detection, Coffee Leaf Disease, Artificial Intelligence

Abstrak

Salah satu hasil perkebunan dengan peranan cukup penting dalam kegiatan perekonomian di Indonesia adalah kopi. Tapi, produksi kopi di Indonesia mengalami penurunan, dimana salah satu penyebabnya adalah serangan hama dan penyakit. Kecerdasan buatan dapat menjadi solusi untuk membantu petani mendeteksi penyakit pada tanaman kopi dengan menggunakan algoritma deteksi objek. Penelitian ini menggunakan algoritma deteksi objek yolov8 untuk melakukan deteksi pada keadaan dan penyakit daun tanaman kopi yang dibagi ke dalam empat klasifikasi, yaitu miner, rust, phoma dan healthy. Penelitian dilakukan dalam tiga skenario percobaan yang dibedakan dengan berdasarkan pada perbandingan pembagian data pada train set, validation set, dan test set, dimana secara berurutan dari train, validation, dan test, skenario pertama memiliki perbandingan 80:10:10, skenario kedua 70:15:15, dan skenario ketiga 70:20:10. Proses penelitian menggunakan model YOLOv8s mendapatkan model dengan hasil performa terbaik pada perbandingan data 70% train set, 20% validation set, dan 10% test set. Model dengan performa terbaik memiliki nilai mAP sebesar 97,8%, precision 95,2%, recall 96,6%, dan f1-score 96%.

Kata kunci: YOLOv8, Deteksi Objek, Penyakit Daun Kopi, Kecerdasan Buatan

1. Pendahuluan

Dalam kegiatan perekonomian Indonesia, sektor pertanian memiliki peranan yang cukup penting. Pada tahun 2022, sektor pertanian berkontribusi terhadap Produk Domestik Bruto (PDB) sekitar 12,40 persen [1]. Dalam sektor pertanian, subsektor perkebunan menjadi salah satu subsektor dengan potensi yang cukup besar, yang berkontribusi sebesar 3,76 persen terhadap total PDB dan 30,32 persen terhadap sektor Pertanian, Kehutanan, dan Perikanan pada tahun 2022. Selain itu, subsektor

perkebunan merupakan penyedia bahan baku untuk sektor industri, penyerap tenaga kerja, dan penghasil devisa.

Salah satu hasil perkebunan dengan peranan cukup penting dalam kegiatan perekonomian di Indonesia adalah kopi. Pada tahun 2022, Indonesia menjadi produsen kopi terbesar di dunia pada peringkat ketiga [2]. Namun, berdasarkan publikasi dari BPS, produksi kopi di Indonesia mengalami penurunan dibandingkan dengan tahun sebelumnya. Produksi kopi di Indonesia pada tahun 2021 adalah sebesar 786,19 ribu ton, dan turun menjadi 774,96 ribu ton atau turun sebesar 1,43% [1]. Terhambatnya peningkatan produksi di Indonesia disebabkan karena rendahnya produktivitas dan kualitas kopi yang dihasilkan. Rendahnya produktivitas dan kualitas kopi di Indonesia dapat disebabkan oleh berbagai hal, salah satunya adalah adanya gangguan dari berbagai serangan hama dan penyakit. Penyakit daun pada kopi seperti penggerek daun, karat daun, bercak daun coklat, dan bercak daun cercospora mempengaruhi tanaman kopi sehingga menyebabkan defoliiasi dan penurunan fotosintesis sehingga menurunkan hasil dan kualitas produk akhir [3].

Proses untuk melakukan pendeteksian dan klasifikasi penyakit atau hama pada tanaman yang dilakukan secara manual memiliki banyak masalah [4]. Kecerdasan buatan merupakan salah satu teknologi yang dimanfaatkan untuk mendeteksi penyakit pada tanaman. Kecerdasan buatan dapat dimanfaatkan untuk membedakan berbagai penyakit dengan gejala serupa serta mengurangi waktu dan biaya diagnosis [5]. Untuk mendeteksi penyakit pada tanaman, maka dibutuhkan algoritma deteksi objek. Salah satu algoritma deteksi objek yang sering digunakan adalah YOLO (You Only Look Once). YOLO bekerja dengan cara menggunakan seluruh gambar sebagai masukan algoritma, dan hanya melalui jaringan saraf yang secara konstan melakukan prediksi, lokasi kotak pembatas dan kategori dari objek pada gambar dapat diperoleh [6]. YOLO memiliki beberapa keunggulan untuk digunakan dalam mendeteksi objek pada gambar. YOLO bekerja dengan efisien karena hanya perlu memproses gambar satu kali, tidak seperti algoritma deteksi objek tradisional yang memerlukan beberapa kali pemrosesan. Selain itu, YOLO dapat mendeteksi banyak objek dalam sebuah gambar [7].

Berdasarkan hal tersebut, pada penelitian ini algoritma deteksi objek YOLO digunakan untuk membangun model yang dapat mendeteksi penyakit daun pada gambar daun kopi. Pembuatan model menggunakan YOLOv8 yang merupakan versi terbaru model deteksi objek YOLO. Model dibangun untuk mendeteksi beberapa penyakit dan keadaan pada daun kopi, yaitu penyakit miner, rust, dan phoma serta healthy yang diperoleh pada situs kaggle. Penelitian juga dilakukan untuk melihat pengaruh perbandingan pembagian data terhadap performa model YOLOv8.

2. Metodologi Penelitian

2.1. Tinjauan Pustaka

YOLO telah banyak digunakan dalam melakukan penelitian. Salah satu penelitian memanfaatkan YOLOv7 yang dirancang untuk mengidentifikasi dan mendeteksi penyakit daun teh menggunakan gambar yang diambil di lingkungan alami di banyak perkebunan teh di wilayah Sylhet, Bangladesh. Penelitian tersebut menunjukkan hasil berupa akurasi deteksi sebesar 97,3%, precision 96,70%, recall 96,40%, dan F1-score 96,5% [7].

Salah satu penelitian menggunakan YOLOv8 untuk melakukan deteksi buah [8]. Penelitian tersebut menggunakan dataset yang terdiri dari 15 tipe buah. Penelitian tersebut menunjukkan hasil bahwa model memiliki performa yang baik dalam melakukan deteksi pada tiap kategori gambar buah dengan akurasi rata-rata di atas 97%.

Penelitian selanjutnya menggunakan YOLOv8 untuk melakukan deteksi penyakit pada tanaman stroberi. Dataset yang dipakai memiliki delapan klasifikasi penyakit. Hasil penelitian menunjukkan bahwa model mampu melakukan klasifikasi dengan nilai akurasi sebesar 88,5%, precision 100%, recall 94%, dan F1-score 84% [9].

Penelitian lainnya menggunakan YOLOv8 untuk melakukan pendeteksian penyakit dan hama pada tanaman citrus. Penelitian tersebut menggunakan data penyakit dan hama citrus yang dibagi ke dalam enam klasifikasi. Penelitian tersebut menunjukkan performa model dengan nilai akurasi sebesar 92,5%, precision 92,6%, recall 89,4%, dan F1-score 91% [10].

Penelitian lainnya memanfaatkan YOLOv8 untuk mendeteksi sel darah putih menggunakan analisis transformator dan dalam gambar digital sel darah sehingga tidak subjektif yang membuatnya menjadi lebih cepat dan akurat. Penelitian tersebut menggunakan DETR untuk peningkatan akurasi dari model yang dibangun. Metode tersebut menghasilkan peningkatan signifikan sebesar 68% hasil deteksi model [11].

Penelitian selanjutnya menggunakan YOLOv7 untuk mendeteksi penyakit daun kopi. Dataset yang digunakan dibagi ke dalam empat kelas yaitu good, miner, phoma, dan rust. Dalam penelitian tersebut augmentasi digunakan untuk meningkatkan jumlah dataset. Hasil dari penelitian tersebut menunjukkan akurasi model sebesar 95,6%, precision 92,6%, recall 93,2%, dan F1-score 93% [12].

Beberapa penelitian selanjutnya menggunakan metode lain untuk mendeteksi penyakit pada gambar daun kopi. Salah satu penelitian menggunakan algoritma Support Vector Machine (SVM) dengan metode seleksi fitur Information Gain untuk melakukan klasifikasi berdasarkan fitur-fitur yang relevan. Dataset yang digunakan terbagi ke dalam lima kelas yaitu leaf miner, leaf rust, phoma, cercospora, dan healthy. Penelitian tersebut menunjukkan hasil akurasi tertinggi sebesar 68,30%, precision 55,77%, recall 57,85% [13].

Penelitian lainnya menggunakan metode Local Binary Pattern dan Random Forest. Metode Local Binary Pattern digunakan untuk ekstraksi ciri dari dataset gambar. Hasil ekstraksi ciri tersebut digunakan dalam proses klasifikasi menggunakan metode Random Forest. Penelitian tersebut menunjukkan hasil akurasi deteksi model sebesar 95,83%, precision 96%, recall 96%, dan F1-score 96% [14].

Penelitian selanjutnya menggunakan metode Convolutional Neural Network (CNN) EfficientNet untuk melakukan klasifikasi penyakit daun kopi. Dataset yang digunakan terdiri dari empat kelas, yaitu miner, phoma, rust, dan no disease. Penelitian tersebut menunjukkan hasil akurasi sebesar 97% [15].

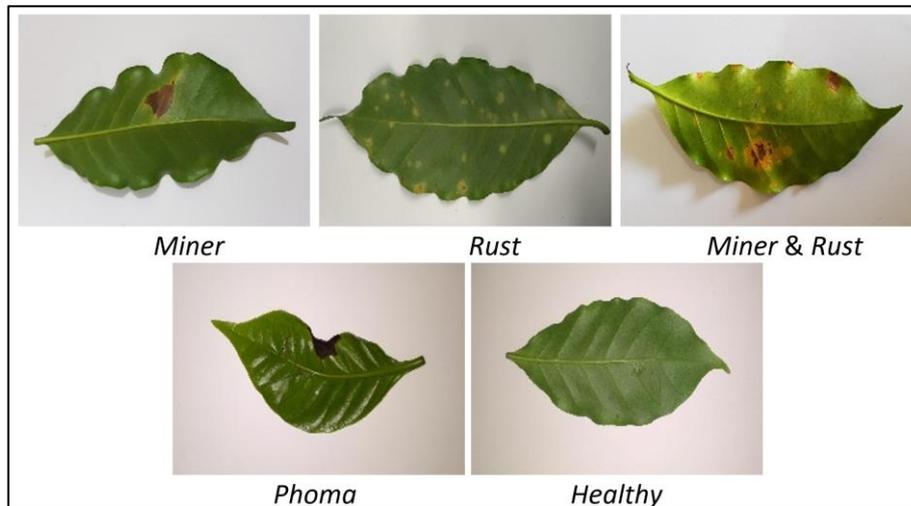
2.2. Dataset

Dataset yang digunakan untuk melakukan penelitian diambil dari situs Kaggle [1]. Data yang diambil memiliki jumlah data gambar daun kopi sebanyak 1664 yang diklasifikasikan ke dalam empat kelas. Klasifikasi data didasarkan pada keadaan dan penyakit yang terlihat pada gambar daun kopi, antara lain *miner*, *rust*, *phoma*, dan *healthy*. Dataset terdiri dari 460 gambar *miner*, 380 gambar *rust*, 484 gambar *phoma*, dan 400 gambar *healthy*. Pada dataset terdapat beberapa data memiliki dua penyakit pada gambarnya sehingga data tersebut masuk ke dalam dua klasifikasi dengan jumlah data sebanyak 60 gambar *miner* dan *rust*.

Data selanjutnya melalui proses labeling. Labeling adalah proses pemberian informasi berupa label kelas dari objek pada dataset gambar [2]. Proses labeling bertujuan untuk memberikan *bounding box* atau kotak pembatas pada gambar, sehingga koordinat dari *ground-truth bounding box* yang akan dibandingkan dengan *predicted bounding box* bisa diperoleh. Hal tersebut bertujuan untuk memperoleh nilai *Intersection over Union* (IoU) yang digunakan sebagai *metrics evaluation* [3]. Penelitian ini menggunakan Roboflow untuk melakukan proses labeling data.

Data yang sudah melewati proses labeling selanjutnya akan masuk ke tahap *preprocessing* dan augmentasi data. Pada tahap *preprocessing* data, ukuran gambar pada dataset akan diubah menjadi 640 x 640 pixel. Hal tersebut dilakukan agar ukuran file dari dataset bisa menjadi lebih kecil dan membuat proses training menjadi lebih cepat.

Dataset yang sudah melalui tahap preprocessing dibagi ke dalam tiga set untuk data *train set*, *validation set*, dan *test set*. Proses selanjutnya adalah augmentasi data pada data *train set*. Augmentasi data adalah teknik penting dalam pemrosesan gambar untuk mengoptimalkan data *training* dengan meningkatkan jumlah dan keragaman data pelatihan serta meningkatkan performa model [3], [4], [5], [6]. Beberapa augmentasi yang diterapkan pada dataset adalah *Flip*, *90° Rotate*, *Rotation*, *Shear*, dan *Brightness*.



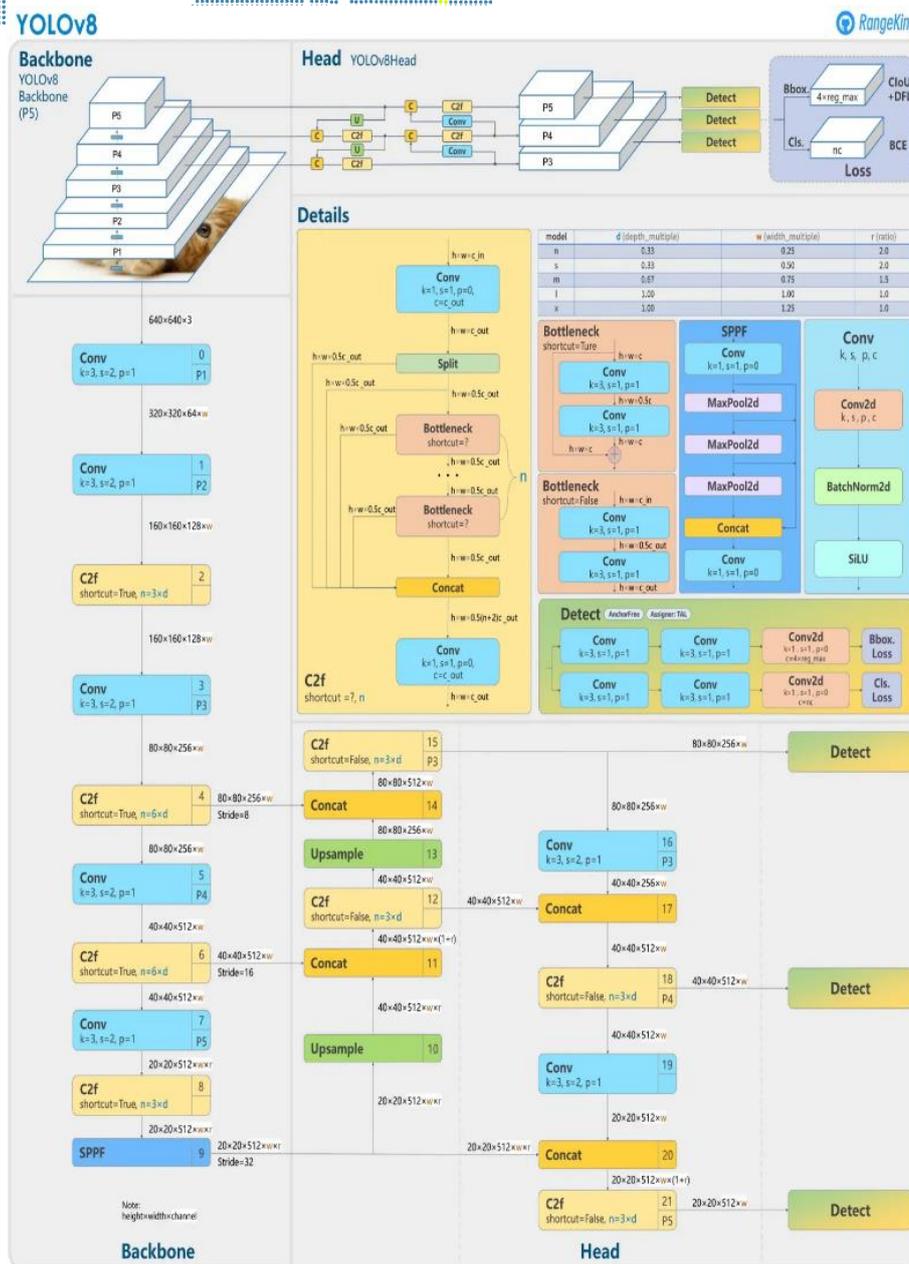
Gambar 1. Contoh Gambar Daun Kopi pada Dataset

2.3. Arsitektur YOLOv8

You Only Look Once (YOLO) merupakan algoritma atau model jaringan satu tahap yang memprediksi probabilitas dan melakukan klasifikasi dari suatu objek dan *bounding box* langsung dari gambar masukan menggunakan CNN sederhana [7], [8]. YOLOv8 merupakan versi terbaru dari model deteksi objek YOLO.

YOLOv8 mencakup banyak perubahan dan peningkatan pengalaman arsitektur dan pengembang dibandingkan dengan versi terdahulunya [9]. Arsitektur dari YOLOv8 memiliki kemiripan dengan arsitektur dari YOLOv5 dengan beberapa perubahan pada CSPLayer, sekarang disebut dengan modul C2f [10], [11]. Modul C2f (*cross-stage partial bottleneck with two convolutions*), merupakan modul yang menggabungkan fitur tingkat tinggi dengan informasi kontekstual untuk meningkatkan akurasi deteksi.

Desain dari arsitektur YOLOv8 dirancang untuk untuk memproses tugas objektivikasi, klasifikasi, dan regresi secara independen. Hal tersebut memungkinkan setiap cabang untuk fokus pada tugasnya dan meningkatkan akurasi model secara keseluruhan. Pada *output layer* dari YOLOv8, fungsi sigmoid digunakan sebagai fungsi aktivasi untuk skor objektivitas, yang mewakili probabilitas bahwa *bounding box* berisi objek. Fungsi *softmax* digunakan untuk probabilitas kelas, mewakili probabilitas objek yang termasuk dalam setiap kelas yang ada. Arsitektur YOLOv8 secara lengkap dapat dilihat pada Gambar 2 [12].



Gambar 2. Arsitektur YOLOv8

2.4. Performance Metrics

Performance metrics adalah alat penting untuk mengevaluasi keakuratan dan efisiensi dari model deteksi objek. Performance metrics menjelaskan kemampuan model untuk menemukan dan mengidentifikasi objek dalam gambar. Informasi tersebut sangat penting untuk mengevaluasi dan meningkatkan fungsionalitas dari model.

Intersection over Union (IOU) menjadi salah satu cara untuk menentukan apakah suatu deteksi sudah benar atau salah, yang dibutuhkan untuk mengetahui true positive (TP), false positive (FP), false negative (FN), dan true negative (TN) [13]. TP adalah deteksi benar dari ground-truth bounding box, FP adalah deteksi tidak benar dari objek yang tidak ada atau deteksi tidak benar pada objek yang ada, sementara FN adalah tidak terdeteksinya atau deteksi yang salah pada ground-truth bounding box. Pada konteks deteksi objek, yang digunakan hanya TP, FP, dan FN karena TN memiliki nilai tidak

terbatas yang menunjukkan jumlah *bounding box* yang seharusnya tidak dideteksi dalam gambar.

Terdapat beberapa *metrics* lain yang diaplikasikan pada model deteksi objek, yang pertama ada *Average Precision* (AP). AP menghitung presisi rata-rata tiap kelas yang ada. Ada juga *Mean Average Precision* (mAP), yang memperluas konsep AP dengan menghitung nilai AP rata-rata di beberapa kelas objek. Hal ini berguna dalam skenario deteksi objek yang memiliki beberapa kelas untuk memberikan evaluasi komprehensif terhadap performa model. Persamaan mAP ditampilkan pada persamaan (1).

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (1)$$

Pada persamaan (1), AP_i adalah AP dari kelas ke- i dan N adalah jumlah dari kelas data. Selanjutnya terdapat *Precision* dan *Recall*. *Precision* mengukur proporsi *true positive* di antara semua *positive predictions*, menilai kemampuan model untuk menghindari *false positive*. *Recall* menghitung proporsi *true positive* di antara semua *ground truth*, mengukur kemampuan model untuk mendeteksi semua *instance* suatu kelas. *Precision* dan *recall* ditampilkan pada persamaan (2) dan persamaan (3).

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

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

Performance metrics lainnya adalah *F1-Score*, yang merupakan rata-rata *precision* dan *recall*, yang memberikan penilaian seimbang terhadap performa model sambil mempertimbangkan *false positive* dan *false negative*. *F1-Score* ditampilkan pada persamaan (4).

$$F1 \text{ Score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

3. Hasil dan Pembahasan

Proses *training* menggunakan Google Collaborator dengan T4 GPU, 2 CPUs, 12.7 GB RAM, 26.6/78.2 GB *disk*. Proses *training* menggunakan model YOLOv8s dengan nilai epochs sebanyak 100. Model YOLOv8s dipilih karena hasil training model yang dihasilkan masih bagus dan memiliki proses training yang lebih singkat dibandingkan versi YOLOv8 lainnya.

Proses percobaan dilakukan dalam tiga skenario berdasarkan variasi dari pembagian data untuk melihat pengaruh pembagian data terhadap performa dari model yang dilatih. Skenario pertama data dibagi menjadi 80% *train set*, 10% *validation set*, dan 10% *test set*. Skenario kedua data dibagi menjadi 70% *train set*, 15% *validation set*, dan 15% *test set*. Skenario ketiga data dibagi menjadi 70% *train set*, 20% *validation set*, dan 10% *test set*.

Pelaksanaan *training* model menggunakan dataset gambar daun kopi dan model YOLOv8s akan memberikan informasi terkait dengan performa model dalam melakukan training. Data kemudian disajikan ke dalam Tabel 1.

Tabel 1. Perbandingan Hasil Training Menggunakan Model YOLOv8s

Skenario	Precision	Recall	F1-score	mAP
1	0.96	0.973	0.97	0.987
2	0.973	0.972	0.97	0.986
3	0.95	0.965	0.96	0.975

Dari hasil yang didapatkan, mAP merupakan *metrics* yang digunakan untuk mengukur performa model secara keseluruhan. Semakin tinggi nilai mAP menunjukkan performa model yang semakin baik dalam melakukan deteksi objek. Pada skenario ketiga, model memiliki nilai mAP sebesar 97,5%, dan pada skenario kedua memiliki nilai mAP sebesar 98,7%, yang menunjukkan bahwa performa model pada skenario kedua lebih baik dibandingkan pada skenario ketiga. Pada skenario pertama, didapatkan hasil terbaik dengan nilai mAP atau akurasi dari model sebesar 98,7%. Nilai tersebut menunjukkan bahwa performa model secara keseluruhan dalam mendeteksi penyakit daun kopi sangat baik. Hal tersebut menunjukkan bahwa model pada skenario pertama dalam tahap *training* merupakan model dengan performa terbaik dalam melakukan deteksi penyakit pada gambar daun kopi.

Model yang telah melauai proses *training*, selanjutnya akan melalui proses *testing*. Pada proses *testing*, model yang telah dilatih akan melakukan deteksi objek pada data *test set* untuk melihat performa model dalam mendeteksi data yang belum pernah dilihat. Hasil dari performa model dalam melakukan deteksi objek pada *test set* ditampilkan dalam Tabel 2.

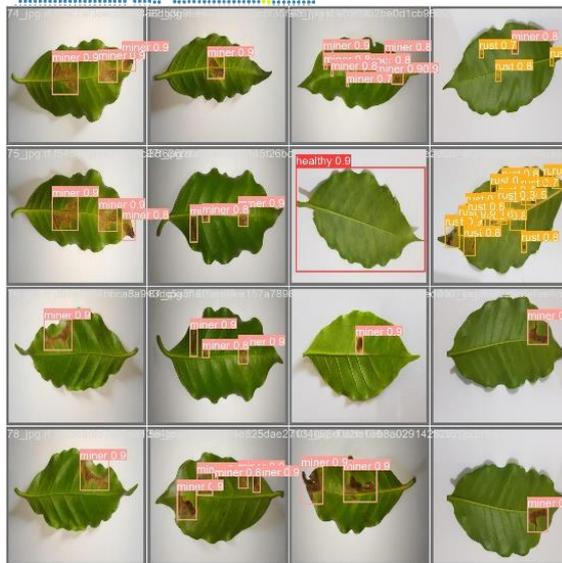
Tabel 2. Perbandingan Hasil *Testing* Menggunakan Model YOLOv8s

Skenario	Precision	Recall	F1-score	mAP
1	0.913	0.909	0.91	0.928
2	0.896	0.871	0.88	0.897
3	0.952	0.966	0.96	0.978

Dari hasil proses *testing* pada *test set*, pada skenario kedua, model memiliki nilai mAP sebesar 89,7% yang mengalami penurunan dibandingkan dengan nilai mAP pada proses *training*. Pada skenario pertama, nilai mAP yang dimiliki sebesar 92,8% yang menunjukkan penurunan dibandingkan dengan nilai mAP pada proses *training*. Pada skenario ketiga, nilai mAP atau akurasi dari model sebesar 98,7% yang mengalami peningkatan dibandingkan dengan nilai mAP pada proses *training*. Nilai tersebut menunjukkan bahwa model pada skenario ketiga merupakan model dengan performa terbaik dan secara keseluruhan model tersebut mampu mendeteksi penyakit daun kopi dengan sangat baik.



Gambar 3. Hasil Deteksi Model Pada Skenario Pertama



Gambar 4. Hasil Deteksi Model Pada Skenario Kedua



Gambar 5. Hasil Deteksi Model Pada Skenario Ketiga

4. Kesimpulan

Penelitian ini memberikan gambaran terkait penggunaan model deteksi objek YOLOv8 untuk melakukan deteksi penyakit pada gambar daun kopi. Dataset yang digunakan terdiri dari empat klasifikasi berdasarkan keadaan dan penyakit yang tersedia pada dataset, yaitu *miner*, *rust*, *phoma*, dan *healthy*. Proses penelitian menggunakan YOLOv8s dan dilakukan dalam tiga skenario berdasarkan perbandingan pembagian data ke dalam *train*, *validation*, dan *test set*. Dari proses *testing* yang telah dijalankan, didapatkan hasil performa model terbaik pada skenario ketiga dengan nilai *mAP precision* sebesar 97,5%, nilai *precision* sebesar 95%, nilai *recall* sebesar 96.5%, dan nilai *F1-score* sebesar 97%. Informasi tersebut menunjukkan bahwa model memiliki kemampuan yang efektif dalam mendeteksi objek pada data yang diberikan. Selain itu, dengan adanya skenario berbeda dalam menjalankan proses penelitian, menunjukkan bahwa pembagian data memiliki pengaruh terhadap performa model. Proses penelitian yang telah dijalankan

memiliki kelebihan dimana model yang berhasil dibangun memiliki performa yang baik dalam melakukan deteksi terhadap penyakit pada tanaman daun kopi. Adapun kekurangan yang dimiliki pada penelitian yang telah dijalankan adalah jumlah data yang dipakai dalam membangun model masih sedikit. Untuk pengembangan penelitian selanjutnya, disarankan untuk menggunakan dataset dengan jumlah yang lebih banyak. Selain itu, pengembangan penelitian dapat dilakukan dengan menggunakan versi lain dari model YOLOv8 untuk melihat bagaimana perbandingan dari hasil performa model YOLOv8 berdasarkan tiap versinya.

Daftar Pustaka

- [1] M. Odhiambo, "Coffee leaf diseases." Accessed: Nov. 10, 2023. [Online]. Available: <https://www.kaggle.com/datasets/badasstechie/coffee-leaf-diseases>
- [2] A. N. Sugandi and B. Hartono, "Implementasi Pengolahan Citra pada Quadcopter untuk Deteksi Manusia Menggunakan Algoritma YOLO," Jul. 2022.
- [3] H. Rezatofghi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression," Feb. 2019, Accessed: Feb. 06, 2024. [Online]. Available: <http://arxiv.org/abs/1902.09630>
- [4] H. Hassan et al., "Review and classification of AI-enabled COVID-19 CT imaging models based on computer vision tasks," Feb. 01, 2022, Elsevier Ltd. doi: 10.1016/j.compbimed.2021.105123.
- [5] A. Saleh, M. Sheaves, D. Jerry, and M. Rahimi Azghadi, "Applications of deep learning in fish habitat monitoring: A tutorial and survey," Mar. 15, 2024, Elsevier Ltd. doi: 10.1016/j.eswa.2023.121841.
- [6] C. Dewi, H. Dwi Purnomo, B. Kristanto, D. Hartomo, S. Zaiton, and M. Hashim, "Utilizing the YOLOv8 Model for Accurate Hand Gesture Recognition with Complex Background," 2017, doi: 10.1016/j.cviu.2017.00.000.
- [7] F. Sultana, A. Sufian, and P. Dutta, "A Review of Object Detection Models based on Convolutional Neural Network," May 2019, doi: 10.1007/978-981-15-4288-6_1.
- [8] C. Dewi, R. C. Chen, Hendry, and Y. T. Liu, "Similar Music Instrument Detection via Deep Convolution YOLO-Generative Adversarial Network," in 2019 IEEE 10th International Conference on Awareness Science and Technology, iCAST 2019 - Proceedings, Institute of Electrical and Electronics Engineers Inc., Oct. 2019. doi: 10.1109/ICAwST.2019.8923404.
- [9] F. Jacob Solawetz, "What is YOLOv8? The Ultimate Guide. [2024]." Accessed: Feb. 08, 2024. [Online]. Available: <https://blog.roboflow.com/whats-new-in-yolov8/>
- [10] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS," Dec. 01, 2023, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/make5040083.
- [11] C. Dewi, D. Manongga, Hendry, E. Mailoa, and K. D. Hartomo, "Deep Learning and YOLOv8 Utilized in an Accurate Face Mask Detection System," Big Data and Cognitive Computing, vol. 8, no. 1, Jan. 2024, doi: 10.3390/bdcc8010009.
- [12] RangeKing, "Brief summary of YOLOv8 model structure," Jan. 10, 2023. Accessed: May 22, 2024. [Online]. Available: <https://github.com/ultralytics/ultralytics/issues/189>.
- [3] R. Padilla, S. L. Netto, E. A. B. Da Silva, and S. L. Netto, "A Survey on Performance Metrics for Object-Detection Algorithms", doi: 10.1109/IWSSIP48289.2020.