

Deteksi Real-Time Hama dan Penyakit Jamur pada Daun Tanaman Berbasis Web Menggunakan YOLO26

Reni Yunita^{1*}, Egi Dio Bagus Sudewo², Bela Astuti³, Muhammad Alfian Rauf Azmal⁴

¹²³ Ilmu Komputer, Sistem Informasi, Universitas Royal, Kisaran, Indonesia

² Ekonomi dan Hukum, Manajemen, Universitas Royal, Kisaran, Indonesia

Email: ¹reniyunita@royal.ac.id, ²egidiobagussudewo@royal.ac.id, ³belaastuti@royal.ac.id, ⁴malfianraufazmal@gmail.com

Email Penulis Korespondensi: reniyunita@royal.ac.id

Abstrak—Serangan hama pertanian dapat menurunkan produksi tanaman secara signifikan dan menghambat kegiatan pertanian. Kesulitan mendeteksi hama secara manual sejak tahap awal membuat petani sering menggunakan pestisida secara berlebihan, yang dapat menyebabkan pencemaran lingkungan dan risiko terhadap kesehatan. Untuk mengatasi permasalahan tersebut, berbagai sistem telah dikembangkan untuk membantu mendeteksi hama sejak dini sehingga petani dapat mengetahui lokasi hama secara lebih tepat. Namun, proses identifikasi di lapangan yang masih dilakukan secara manual sering memerlukan waktu lama, tenaga yang besar, serta rentan terhadap kesalahan identifikasi. Selain itu, beberapa sistem yang telah dikembangkan masih memiliki keterbatasan, seperti belum mampu melakukan deteksi secara *real-time* dan belum dilengkapi dengan sistem pemantauan berbasis web. Penelitian ini mengusulkan sistem deteksi hama dan penyakit jamur pada daun tanaman secara *real-time* menggunakan pendekatan *deep learning* berbasis arsitektur YOLO26. Model YOLO26 dipilih karena memiliki kemampuan deteksi objek yang cepat dan efisien sehingga cocok untuk aplikasi pemantauan pertanian secara *real-time*. Model yang dikembangkan juga diimplementasikan dalam aplikasi berbasis web sehingga memungkinkan proses deteksi hama dan penyakit jamur dilakukan secara *real-time* untuk mendukung pemantauan kondisi tanaman secara lebih praktis. *Dataset* yang digunakan terdiri dari citra daun tanaman yang telah dianotasi ke dalam dua kelas objek, yaitu *pest* dan *fungus*. Hasil pengujian menunjukkan bahwa model yang diusulkan mampu mencapai *precision* sebesar 82%, *recall* sebesar 71%, dan *mAP@0.5* sebesar 78%, dengan nilai *mAP@0.5–0.95* sebesar 37,4%. Selain itu, model memiliki waktu inferensi sekitar 12,1 ms/citra, sehingga mampu melakukan deteksi secara *real-time*. Secara keseluruhan, sistem yang dikembangkan berpotensi membantu petani dalam melakukan pemantauan tanaman secara otomatis dan mendeteksi serangan hama serta penyakit jamur sejak dini, sehingga dapat mengurangi kerusakan tanaman, menekan penggunaan pestisida secara berlebihan.

Kata Kunci: Deep Learning, Deteksi Objek Real-Time, Deteksi Hama, Penyakit Jamur Tanaman, Pengolahan Citra

Abstract—Agricultural pest infestations can significantly reduce crop yields and hinder farming activities. The difficulty of manually detecting pests in their early stages often leads farmers to overuse pesticides, which can cause environmental pollution and pose health risks. To address these issues, various systems have been developed to help detect pests early on, enabling farmers to pinpoint their locations more accurately. However, the identification process in the field, which is still conducted manually, often requires a long time, significant effort, and is prone to misidentification. Additionally, some of the systems that have been developed still have limitations, such as an inability to perform real-time detection and a lack of a web-based monitoring system. This study proposes a real-time detection system for plant pests and fungal diseases on leaves using a deep learning approach based on the YOLO26 architecture. The YOLO26 model was selected because it offers fast and efficient object detection capabilities, making it suitable for real-time agricultural monitoring applications. The dataset consists of annotated plant leaf images categorized into two object classes: pests and fungi. Test results show that the proposed model achieves a precision of 82%, a recall of 71%, and a *mAP@0.5* of 78%, with a *mAP@0.5–0.95* value of 37.4%. In addition, the model has an inference time of approximately 12.1 ms per image, enabling real-time detection. Overall, the developed system has the potential to help farmers automatically monitor their crops and detect pest infestations and fungal diseases at an early stage, thereby reducing crop damage and curbing excessive pesticide use.

Keywords: Deep Learning, Real-Time Object Detection, Pest Detection, Plant Fungal Diseases, Image Processing

1. PENDAHULUAN

Pertanian merupakan sektor yang sangat penting karena menyediakan berbagai kebutuhan dasar manusia seperti pangan, obat-obatan, bahan pakaian, serta sumber penghidupan bagi banyak masyarakat. Namun, perkembangan sektor pertanian sering menghadapi berbagai kendala, salah satunya adalah seringnya terjadi serangan hama tanaman yang dapat menurunkan hasil produksi secara signifikan. Jenis hama yang beragam, tingkat penyebaran yang luas, serta dampaknya yang besar sering menyebabkan kerugian ekonomi yang cukup besar dalam sektor pertanian. Oleh karena itu, pengendalian hama menjadi salah satu perhatian utama dalam menjaga keberlanjutan produksi pertanian dan ketahanan pangan.

Dalam praktik pertanian tradisional, informasi mengenai keberadaan dan perkembangan populasi hama di lahan pertanian biasanya diperoleh melalui pengamatan langsung di lapangan. Cara ini sudah lama digunakan, tetapi memiliki beberapa kelemahan yang cukup jelas. Prosesnya membutuhkan banyak tenaga kerja, memerlukan waktu yang lama, serta sering bergantung pada pengalaman petani dalam mengenali jenis hama yang menyerang tanaman. Selain itu, ukuran hama yang kecil, jumlahnya yang padat, serta kondisi lingkungan yang kompleks sering membuat proses identifikasi menjadi tidak akurat. Akibatnya, petani sering menggunakan pestisida secara berlebihan untuk mencegah kerusakan tanaman, yang pada akhirnya dapat menimbulkan dampak negatif terhadap lingkungan dan kesehatan manusia.

Berbagai penelitian telah dilakukan untuk membantu proses deteksi hama secara otomatis menggunakan teknologi pengolahan citra dan *computer vision*. Pada tahap awal, beberapa penelitian menggunakan pendekatan *machine learning* yang mengekstraksi fitur tertentu dari gambar hama, seperti warna, bentuk, dan tekstur, kemudian menggunakan algoritma klasifikasi untuk menentukan jenis hama. Penerapan berbagai algoritma *machine learning* untuk mendeteksi

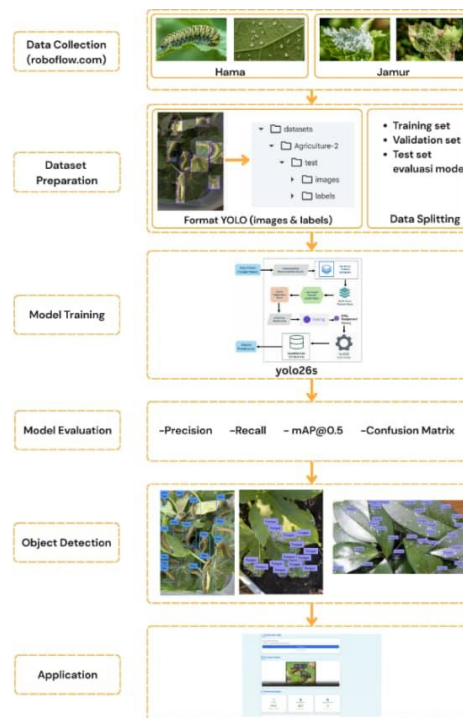
penyakit tanaman kedelai dengan memanfaatkan data citra dan data sensor. Model yang dikembangkan digunakan untuk proses deteksi, prediksi, dan analisis kondisi kesehatan tanaman [1]. Dengan memanfaatkan data citra, kondisi cuaca, dan data tanah, model *machine learning* mampu mengidentifikasi potensi serangan hama dan penyakit secara lebih cepat dan akurat [2][3][4][5][6]. Identifikasi hama tanaman menggunakan pendekatan *machine learning* dengan *dataset* besar yang berisi lebih dari 334.000 gambar dari berbagai jenis tanaman dan hama [7]. Prediksi hama tanaman menggunakan algoritma *machine learning* seperti *Random Forest*, *AdaBoost*, *KNN*, dan *Logistic Regression*, mampu memprediksi keberadaan hama dengan tingkat akurasi yang cukup tinggi [8]. *Decision Tree*, *Random Forest*, *SVM*, *XGBoost*, dan *KNN* untuk mengklasifikasikan jenis serangga berdasarkan fitur suara [9].

Perkembangan teknologi *deep learning* kemudian membawa perubahan besar dalam penelitian deteksi objek, termasuk dalam bidang pertanian. Sistem deteksi hama menggunakan model YOLO yang terintegrasi dengan platform pemantauan berbasis web [10][11][12][13]. Model SRNet-YOLO untuk mendeteksi hama berukuran sangat kecil pada tanaman kapas [14]. Model PestLite berbasis YOLOv5 untuk mendeteksi hama tanaman dengan model yang lebih ringan [15][16]. Model AgriPest-YOLO untuk mendeteksi berbagai jenis hama dari gambar perangkat cahaya [17]. Model Pest-YOLO untuk mendeteksi berbagai jenis hama berukuran kecil yang muncul secara padat pada tanaman [18].

Berdasarkan kajian terhadap penelitian-penelitian sebelumnya, masih terdapat keterbatasan dalam mengembangkan sistem yang mampu mendeteksi hama dan penyakit jamur secara *real-time* dengan tetap mempertahankan kecepatan dan akurasi pada kondisi lapangan yang kompleks. Selain itu, sebagian besar penelitian lebih berfokus pada pengembangan model deteksi, sementara implementasi model ke dalam aplikasi yang dapat digunakan secara langsung untuk pemantauan tanaman masih terbatas. Selain itu, gambar hama pada tanaman sering memiliki latar belakang yang kompleks, ukuran objek yang sangat kecil, serta jumlah objek yang padat dalam satu gambar sehingga proses deteksi menjadi lebih sulit. Kondisi ini menunjukkan bahwa sistem deteksi hama yang ada masih perlu dikembangkan agar dapat bekerja secara lebih cepat, akurat, dan efisien dalam kondisi nyata di bidang pertanian. Oleh karena itu, penelitian ini mengembangkan model YOLO26 yang diimplementasikan pada aplikasi berbasis web untuk mendukung deteksi hama dan penyakit jamur secara *real-time* sehingga dapat dimanfaatkan sebagai media pemantauan tanaman yang lebih praktis.

2. METODOLOGI PENELITIAN

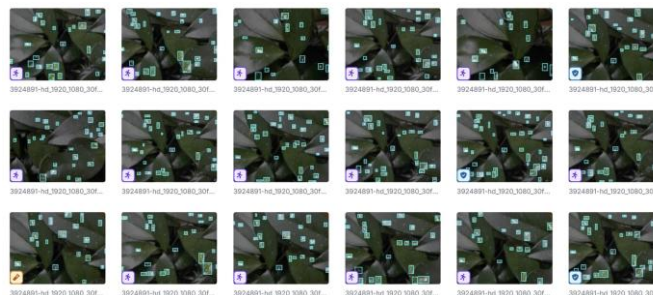
Penelitian ini meliputi beberapa tahapan yaitu pengumpulan *dataset*, persiapan data, pelatihan model, evaluasi model, dan proses deteksi objek. Alur keseluruhan metode yang digunakan dalam penelitian ini ditunjukkan pada Gambar 1.



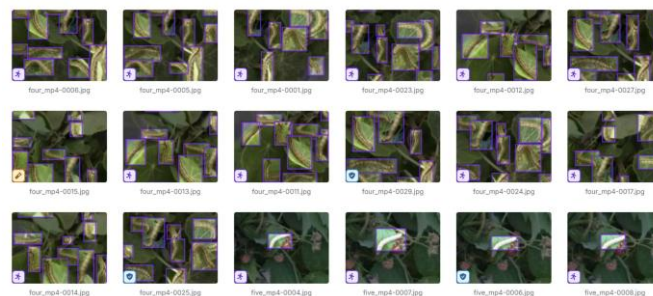
Gambar 1. Alur metodologi sistem deteksi hama dan penyakit jamur pada daun tanaman

2.1 Data Collection

Pada tahap ini dilakukan pengumpulan *dataset* berupa citra daun tanaman yang mengandung hama dan penyakit jamur. *Dataset* diperoleh dari platform Roboflow yang menyediakan data citra beserta anotasi objek yang akan dideteksi.



Gambar 2. Contoh *dataset* penyakit jamur



Gambar 3. Contoh *dataset* hama

2.2 Dataset Prepration

Pada tahap ini *dataset* yang telah diperoleh dari Roboflow disusun terlebih dahulu agar dapat digunakan oleh model YOLO. Seperti yang terlihat pada Gambar 1. bagian (*Dataset Preparation*), *dataset* disimpan dalam format YOLO yang terdiri dari dua bagian utama yaitu *images* yang berisi file gambar dan *labels* yang berisi *file* anotasi objek. Setiap gambar memiliki satu *file* label yang berisi informasi posisi *bounding box* dan kelas objek yang dideteksi. Selain itu, *dataset* juga diorganisasi dalam struktur folder tertentu agar mudah digunakan saat proses pelatihan model. Struktur folder terdiri dari beberapa bagian seperti *train*, *valid*, dan *test* yang masing-masing berisi folder *images* dan *labels*.

2.3 Model Training

Penelitian ini menggunakan model YOLO (*You Only Look Once*) merupakan algoritma deteksi objek berbasis *deep learning* yang melakukan proses klasifikasi dan lokalisasi objek dalam satu tahap (*single-stage detector*), sehingga memiliki kecepatan inferensi yang tinggi dibandingkan metode *two-stage detector*. YOLO26 merupakan pengembangan terbaru dari keluarga YOLO yang dirancang untuk meningkatkan efisiensi komputasi, kemampuan mendeteksi objek berukuran kecil [19]. *Dataset* yang telah dianotasi dibagi menjadi tiga bagian, yaitu data pelatihan (*training set*), data validasi (*validation set*), dan data pengujian (*testing set*). Proses pelatihan model dilakukan menggunakan arsitektur YOLO26s dengan tugas deteksi objek (*object detection*). *Dataset* yang telah diberi anotasi digunakan sebagai data masukan melalui berkas *data.yaml*. Pelatihan model dilakukan selama 60 epoch dengan ukuran citra (*image size*) sebesar 640×640 piksel. Selama proses pelatihan, sistem secara otomatis menghasilkan grafik performa model untuk memantau perkembangan proses pembelajaran. Implementasi pelatihan dilakukan menggunakan GPU NVIDIA Tesla T4 dengan kapasitas memori 15 GB dan dukungan CUDA 13.0 untuk mempercepat proses komputasi dan optimasi model. Untuk meningkatkan kemampuan generalisasi model dan mengurangi risiko *overfitting*, proses pelatihan memanfaatkan beberapa teknik augmentasi data yang diterapkan secara otomatis oleh *framework Ultralytics YOLO*. Teknik augmentasi yang digunakan beserta parameter dan nilai konfigurasinya ditunjukkan pada Tabel 1.

Tabel 1. Parameter Augmentasi yang Digunakan pada Pelatihan Model YOLO26s

No	Teknik Augmentasi	Parameter	Nilai
1	Mosaic	mosaic	1.0
2	Horizontal Flip	flip_lr	0.5
3	Hue Adjustment	hsv_h	0.015
4	Saturation Adjustment	hsv_s	0.7

5	Brightness Adjustment	hsv_v	0.4
6	Translation	translate	0.1
7	Scaling	scale	0.5
8	Random Erasing	erasing	0.4

Berdasarkan konfigurasi pada Tabel 1. augmentasi yang diterapkan mencakup variasi posisi, ukuran, warna, dan sebagian area objek pada citra. Penerapan augmentasi tersebut bertujuan untuk meningkatkan keragaman data pelatihan sehingga model mampu mengenali objek pada berbagai kondisi dan menghasilkan performa deteksi yang lebih baik pada data yang belum pernah dilihat sebelumnya.

2.4 Model Evaluation

Setelah proses pelatihan selesai, dilakukan evaluasi terhadap model untuk mengetahui tingkat kinerjanya. Evaluasi dilakukan menggunakan beberapa metrik seperti *precision*, *recall*, *mAP@0.5*, serta *confusion matrix*.

2.4.1 Precision

Precision digunakan untuk mengukur tingkat ketepatan model dalam mendeteksi objek. Nilai *precision* menunjukkan persentase objek yang diprediksi sebagai objek target dan benar-benar merupakan objek target. Semakin tinggi nilai *precision*, semakin sedikit kesalahan deteksi (*false positive*) yang dihasilkan model [20].

$$\text{Precision} = \frac{TP}{TP+FP} \quad (1)$$

2.4.2 Recall

Recall digunakan untuk mengukur kemampuan model dalam menemukan seluruh objek target yang terdapat pada data. Nilai *recall* yang tinggi menunjukkan bahwa sebagian besar objek berhasil dideteksi dan hanya sedikit objek yang terlewat (*false negative*).

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2)$$

2.4.3 Mean Average Precision (mAP)

Mean Average Precision (mAP) merupakan metrik utama yang digunakan untuk mengevaluasi performa model deteksi objek. Nilai mAP diperoleh dari rata-rata nilai *Average Precision* (AP) pada seluruh kelas objek. AP dihitung berdasarkan luas area di bawah kurva *Precision-Recall* (*Precision-Recall Curve*). Semakin tinggi nilai mAP, semakin baik kemampuan model dalam melakukan klasifikasi dan lokalisasi objek [20].

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3)$$

2.4.4 Confusion Matrix

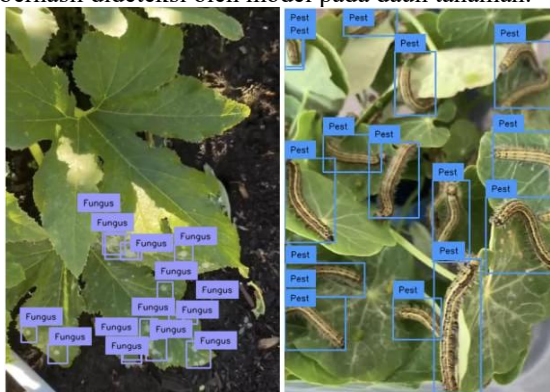
Confusion Matrix merupakan metode evaluasi yang digunakan untuk membandingkan hasil prediksi model dengan data aktual. Matriks ini terdiri atas empat komponen utama, yaitu *True Positive* (TP), *False Positive* (FP), *False Negative* (FN), dan *True Negative* (TN). Melalui nilai-nilai tersebut dapat dihitung berbagai metrik evaluasi seperti *precision*, *recall*, *accuracy*, dan F1-score untuk menilai kinerja model deteksi objek. Pada penelitian ini, *confusion matrix* digunakan untuk menganalisis kemampuan model dalam mengidentifikasi objek hama dan penyakit jamur secara benar maupun kesalahan deteksi yang terjadi selama proses pengujian [21].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

2.5 Object Detection

Model yang telah dilatih kemudian digunakan untuk mendeteksi objek pada citra daun tanaman. Pada tahap ini sistem akan menampilkan hasil deteksi berupa *boundingbox* dan label pada objek yang berhasil dikenali. Gambar 4. menunjukkan contoh hasil deteksi objek pada daun tanaman menggunakan model yang telah dilatih. Pada gambar tersebut terlihat bahwa sistem mampu mengenali dua jenis objek yaitu *pest* (hama) dan *fungus* (jamur) yang terdapat pada

permukaan daun. Setiap objek yang terdeteksi ditandai dengan *boundingbox* serta label kelas yang menunjukkan jenis objek yang dikenali oleh model. Pada bagian kiri gambar terlihat beberapa area daun yang terdeteksi sebagai *fungus*, yang ditandai dengan kotak pembatas dan label pada setiap objek yang teridentifikasi. Sementara itu pada bagian kanan gambar terlihat beberapa objek *pest* yang berhasil dideteksi oleh model pada daun tanaman.



Gambar 4. Hasil deteksi objek

2.6 Application

Tahap terakhir adalah penerapan model deteksi ke dalam sebuah aplikasi berbasis *website*, sehingga pengguna dapat melakukan proses deteksi hama dan penyakit jamur pada daun tanaman secara lebih mudah dan praktis.

3. HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan model YOLO26 yang digunakan untuk mendeteksi hama dan penyakit jamur pada daun tanaman. Kinerja model dievaluasi menggunakan metrik *precision*, *recall*, dan mAP. Selain itu, ditampilkan juga beberapa contoh hasil deteksi objek pada citra daun tanaman.

3.1 Hasil Evaluasi Model

Hasil penelitian menunjukkan bahwa model YOLO26 dapat diimplementasikan pada aplikasi berbasis web untuk melakukan deteksi hama dan penyakit jamur secara *real-time*. Temuan ini berbeda dengan sebagian besar penelitian terdahulu seperti YOLO-PEST [11], Improved YOLOv5m [12], SRNet-YOLO [14], PestLite [15], AgriPest-YOLO [17], dan Pest-YOLO [18] yang lebih berfokus pada pengembangan arsitektur model untuk meningkatkan kemampuan deteksi objek. Meskipun penelitian-penelitian tersebut berhasil meningkatkan performa deteksi, implementasi model ke dalam sistem pemantauan yang dapat digunakan secara langsung oleh pengguna masih terbatas. Selain itu, penelitian terdahulu umumnya difokuskan pada deteksi hama tanaman, sedangkan penelitian ini mengintegrasikan deteksi hama dan penyakit jamur dalam satu sistem berbasis web yang dapat digunakan untuk proses pemantauan secara *real-time*. Dengan adanya integrasi antara model YOLO26 dan aplikasi web, pengguna dapat melakukan proses deteksi secara langsung tanpa memerlukan konfigurasi model secara terpisah. Hal ini menunjukkan bahwa penelitian ini tidak hanya berfokus pada aspek pengembangan model deteksi, tetapi juga pada penerapan sistem yang lebih praktis untuk mendukung kegiatan pemantauan tanaman di lapangan.

Tabel 2. Hasil Evaluasi Model YOLO26

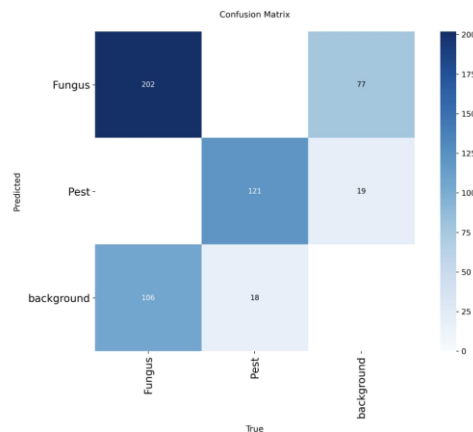
Class	Images	Instances	Precision (P)	Recall (R)	mAP@0.5	mAP@0.5:0.95
All	59	447	0.820	0.710	0.780	0.374
Fungus	13	308	0.753	0.571	0.659	0.211
Pest	46	139	0.887	0.849	0.900	0.536

Tabel 2. menunjukkan hasil evaluasi model YOLO26 pada data pengujian. Secara keseluruhan, model menghasilkan nilai *precision* sebesar 0,82, *recall* sebesar 0,71, dan mAP@0.5 sebesar 0,78. Hasil tersebut menunjukkan bahwa model mampu mendeteksi objek hama dan penyakit jamur dengan tingkat ketepatan yang cukup baik. Kelas *Pest* memperoleh nilai *precision* sebesar 88,7%, *recall* 84,9%, dan mAP@0.5 sebesar 90%, sedangkan kelas *Fungus* hanya memperoleh *precision* 75,3%, *recall* 57,1%, dan mAP@0.5 sebesar 65,9%. Perbedaan performa ini menunjukkan bahwa model lebih mudah mengenali objek hama dibandingkan gejala penyakit jamur pada daun tanaman. Kesalahan deteksi yang terjadi pada kelas *Fungus* diduga disebabkan oleh karakteristik visual gejala penyakit jamur yang beragam, seperti perubahan warna daun, bercak, dan area infeksi yang memiliki kemiripan dengan kondisi daun sehat. Selain itu, beberapa citra memiliki latar belakang yang kompleks serta variasi pencahayaan yang menyebabkan batas antara area terinfeksi dan area sehat menjadi kurang jelas. Kondisi tersebut dapat menyebabkan objek penyakit jamur tidak terdeteksi atau terdeteksi

pada lokasi yang kurang tepat. Selain faktor visual, jumlah data dan distribusi objek pada setiap kelas juga dapat memengaruhi performa model. Objek hama umumnya memiliki bentuk yang lebih jelas dan kontras terhadap latar belakang sehingga lebih mudah dipelajari oleh model YOLO26. Sebaliknya, gejala penyakit jamur sering muncul dalam bentuk pola yang tidak seragam sehingga meningkatkan tingkat kesulitan proses klasifikasi dan lokalisasi objek.

3.2 Confusion Matrix

Confusion matrix pada penelitian ini digunakan untuk mengklasifikasikan objek hama (*pest*) dan penyakit jamur (*fungus*) pada daun tanaman. *Confusion matrix* menunjukkan jumlah sampel yang diklasifikasikan dengan benar maupun salah pada setiap kelas serta untuk menghitung metrik evaluasi [22].

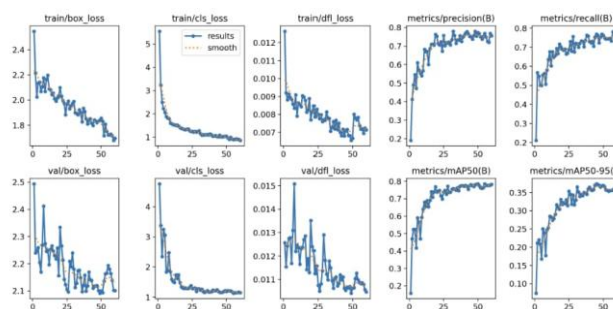


Gambar 5. Confusion Matrix

Pada gambar 5 model mampu mengenali kelas *fungus* dengan cukup baik, yang ditunjukkan oleh 202 prediksi yang sesuai dengan kelas sebenarnya. Namun demikian, masih terdapat beberapa kesalahan prediksi, di mana 77 objek *fungus* terdeteksi sebagai *background*. Hal ini menunjukkan bahwa pada beberapa kondisi model belum mampu mengenali objek jamur secara tepat. Pada kelas *pest*, model juga menunjukkan kemampuan deteksi yang cukup baik dengan 121 prediksi yang benar. Meskipun demikian, masih terdapat 19 objek *pest* yang terdeteksi sebagai *background*, yang menunjukkan bahwa beberapa objek hama tidak terdeteksi dengan baik oleh model. Selain itu, terdapat juga beberapa objek yang sebenarnya merupakan *fungus* maupun *pest*, tetapi tidak terdeteksi oleh model dan dianggap sebagai *background*, yaitu masing-masing sebanyak 106 data pada kelas *fungus* dan 18 data pada kelas *pest*. Kondisi ini biasanya terjadi ketika objek berukuran kecil, tertutup oleh bagian daun lain, atau kualitas citra kurang jelas.

3.3 Evaluasi Model YOLO26

Gambar 6. menunjukkan grafik perkembangan proses pelatihan dan evaluasi model YOLO26 selama beberapa *epoch*. Grafik pada baris pertama menampilkan nilai *train loss*, yang terdiri dari *train box loss*, *train cls loss*, dan *train dfl loss*.



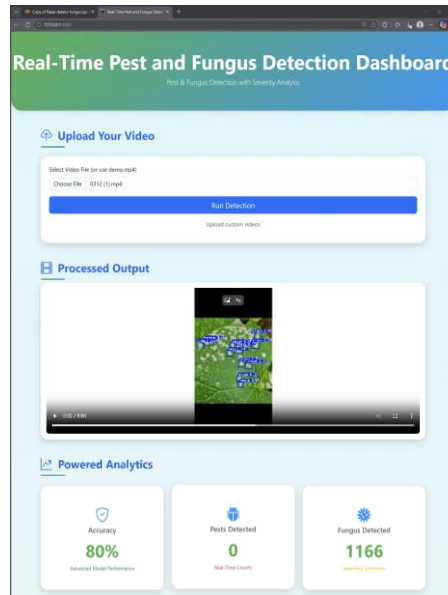
Gambar 6. Grafik performa model YOLO26

Ketiga grafik menunjukkan tren penurunan seiring bertambahnya *epoch*, yang menandakan bahwa model semakin mampu mempelajari pola pada data pelatihan. Selain itu, grafik *precision* dan *recall* juga ditampilkan pada baris pertama. Nilai *precision* dan *recall* terlihat meningkat secara bertahap hingga mencapai nilai yang relatif stabil. Hal ini menunjukkan bahwa model semakin baik dalam mendeteksi objek serta mampu mengurangi kesalahan prediksi selama proses pelatihan. Pada baris kedua ditampilkan grafik validation loss, yaitu *val box loss*, *val cls loss*, dan *val dfl loss*. Nilai loss pada data validasi juga menunjukkan kecenderungan menurun, yang mengindikasikan bahwa model tidak hanya belajar dengan baik pada data pelatihan, tetapi juga mampu melakukan generalisasi terhadap data validasi. Selain itu, grafik mAP@0.5 dan mAP@0.5:0.95 menunjukkan peningkatan performa model selama proses pelatihan. Nilai mAP yang meningkat menunjukkan bahwa kemampuan model dalam mendeteksi objek pada citra semakin baik. Secara keseluruhan, grafik-

grafik tersebut menunjukkan bahwa proses pelatihan model berjalan dengan baik dan model mampu mencapai performa yang stabil pada akhir proses *training*.

3.4 Implementasi pada Aplikasi

Gambar 7 merupakan tampilan implementasi sistem deteksi hama dan penyakit jamur pada daun tanaman dalam bentuk aplikasi berbasis web. Pada halaman utama aplikasi, pengguna dapat mengunggah video yang berisi objek daun tanaman melalui fitur *Upload Your Video*, kemudian sistem akan memproses video tersebut menggunakan model YOLO26 yang telah dilatih sebelumnya.



Gambar 7. Implementasi sistem deteksi hama dan penyakit jamur

Setelah proses deteksi selesai, hasilnya akan ditampilkan pada bagian *Processed Output*, di mana objek yang terdeteksi akan ditandai dengan *boundingbox* dan label pada video. Aplikasi juga menyediakan informasi tambahan pada bagian *Powered Analytics* yang menunjukkan ringkasan hasil analisis sistem. Informasi tersebut meliputi tingkat akurasi model, jumlah hama yang terdeteksi, serta jumlah indikasi penyakit jamur yang ditemukan.

4. KESIMPULAN

Berdasarkan hasil penelitian, model YOLO26 mampu mendeteksi hama dan penyakit jamur pada daun tanaman secara *real-time* dengan nilai precision sebesar 82%, recall sebesar 71%, dan mAP@0.5 sebesar 78%. Hasil pengujian menunjukkan bahwa model memiliki kemampuan yang cukup baik dalam mengidentifikasi objek pada berbagai kondisi citra, termasuk objek berukuran kecil dan latar belakang yang kompleks. Integrasi model ke dalam aplikasi berbasis web memungkinkan proses deteksi dilakukan secara lebih praktis dan cepat sehingga dapat mendukung kegiatan pemantauan tanaman secara langsung. Dengan demikian, sistem yang dikembangkan berpotensi menjadi solusi pendukung dalam membantu deteksi dini hama dan penyakit jamur pada tanaman.

REFERENCES

- [1] A. Deshmukh, D. Nawgaje, and K. Vyas, "Exploring the Role of Machine Learning in Advancing Crop Disease Detection," INSTICC, Sep. 2025, pp. 360–366. doi: 10.5220/0013592400004664.
- [2] O. Emi-Johnson, O. Fasanya, T. K. Amusa, and K. Nkrumah, "Predicting Crop Disease Outbreaks Using Weather and Soil Data: A Machine Learning Approach to Risk-Based Crop Protection," International Journal of Engineering Technology Research & Management, vol. 8, no. 3, pp. 217–235, 2024. doi: 10.5281/zenodo.15728788.
- [3] T. Domingues, T. Brandão, and J. C. Ferreira, "Machine Learning for Detection and Prediction of Crop Diseases and Pests: A Comprehensive Survey," Sep. 01, 2022, MDPI. doi: 10.3390/agriculture12091350.
- [4] L. V. Valleggi and F. M. Stefanini, "A mini-review on data science approaches in crop yield and disease detection," Frontiers in Agronomy, vol. 6, 2024, doi: 10.3389/fagro.2024.1352219.
- [5] I. Ahmed and P. K. Yadav, "A systematic analysis of machine learning and deep learning based approaches for identifying and diagnosing plant diseases," Sustainable Operations and Computers, vol. 4, pp. 96–104, Jan. 2023, doi: 10.1016/j.susoc.2023.03.001.
- [6] U. Ali, M. A. Ismail, R. A. Ariyuluran Habeeb, and S. R. Ali Shah, "Performance Evaluation of YOLO Models in Plant Disease Detection," Journal of Informatics and Web Engineering, vol. 3, no. 2, pp. 199–211, Jun. 2024, doi: 10.33093/jiwe.2024.3.2.15.
- [7] R. Wayama, Y. Sasaki, S. Kagiwada, N. Iwasaki, and H. Iyatomi, "Investigation to answer three key questions concerning plant pest identification and development of a practical identification framework," Jul. 2024, doi: 10.1016/j.compag.2024.109021.
- [8] S. Balasubramaniam, S. G. Nelson, M. Arishma, A. S. Rajan, and K. K. Satheesh, "Machine Learning based Disease and Pest detection in Agricultural Crops," EAI Endorsed Transactions on Internet of Things, vol. 10, 2024, doi: 10.4108/eetiot.5049.



- [9] C. Potamitis, I. Rigakis, and K. Fysarakis, "Insect biometrics: Optoacoustic signal processing and its applications to remote monitoring of McPhail type traps," *PLoS One*, vol. 10, no. 11, 2015, doi: 10.1371/journal.pone.0140474.
- [10] W. M. On and N. F. Abubacker, "YOLO-Driven Lightweight Mobile Real-Time Pest Detection and Web-Based Monitoring for Sustainable Agriculture," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 12, pp. 658–673, 2024, doi: 10.14569/IJACSA.2024.0151267.
- [11] J. Qiang, L. Zhao, H. Wang, T. Xu, Q. Jia, and L. Sun, "YOLO-PEST: a novel rice pest detection approach based on YOLOv5s," *Plant Methods*, vol. 21, no. 1, Dec. 2025, doi: 10.1186/s13007-025-01438-w.
- [12] M. Dai, M. M. H. Dorjoy, H. Miao, and S. Zhang, "A New Pest Detection Method Based on Improved YOLOv5m," *Insects*, vol. 14, no. 1, Jan. 2023, doi: 10.3390/insects14010054.
- [13] I. Ahmad *et al.*, "Deep Learning Based Detector YOLOv5 for Identifying Insect Pests," *Applied Sciences (Switzerland)*, vol. 12, no. 19, Oct. 2022, doi: 10.3390/app121910167.
- [14] S. Yang, G. Zhou, Y. Feng, J. Zhang, and Z. Jia, "SRNet-YOLO: A model for detecting tiny and very tiny pests in cotton fields based on super-resolution reconstruction," *Front. Plant Sci.*, vol. 15, 2024, doi: 10.3389/fpls.2024.1416940.
- [15] Q. Dong, L. Sun, T. Han, M. Cai, and C. Gao, "PestLite: A Novel YOLO-Based Deep Learning Technique for Crop Pest Detection," *Agriculture (Switzerland)*, vol. 14, no. 2, Feb. 2024, doi: 10.3390/agriculture14020228.
- [16] W. Luo, "Pest-YOLO: A YOLOv5-Based Lightweight Crop Pest Detection Algorithm," *International Journal of Engineering and Technology Innovation*, vol. 15, no. 1, pp. 11–25, 2025, doi: 10.46604/ijeti.2024.13748.
- [17] W. Zhang, H. Huang, Y. Sun, and X. Wu, "AgriPest-YOLO: A rapid light-trap agricultural pest detection method based on deep learning," *Front. Plant Sci.*, vol. 13, Dec. 2022, doi: 10.3389/fpls.2022.1079384.
- [18] C. Wen *et al.*, "Pest-YOLO: A model for large-scale multi-class dense and tiny pest detection and counting," *Front. Plant Sci.*, vol. 13, Oct. 2022, doi: 10.3389/fpls.2022.973985.
- [19] R. Sapkota, R. H. Cheppally, A. Sharda, and M. Karkee, "YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection," Published in: arXiv (Computer Vision and Pattern Recognition / cs.CV), Mar. 2026, [Online]. Available: <http://arxiv.org/abs/2509.25164>
- [20] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, "A comparative analysis of object detection metrics with a companion open-source toolkit," *Electronics (Switzerland)*, vol. 10, no. 3, pp. 1–28, Feb. 2021, doi: 10.3390/electronics10030279.
- [21] M. Telçeken, D. Akgun, and S. Kacar, "An Evaluation of Image Slicing and YOLO Architectures for Object Detection in UAV Images," *Applied Sciences (Switzerland)*, vol. 14, no. 23, Dec. 2024, doi: 10.3390/app142311293.
- [22] N. Kundu *et al.*, "Iot and interpretable machine learning based framework for disease prediction in pearl millet," *Sensors*, vol. 21, no. 16, Aug. 2021, doi: 10.3390/s21165386.