

# Deteksi dan Klasifikasi Penyakit Pada Daun Kopi Menggunakan YOLOv7

Ardiansyah<sup>[1]\*</sup>, Nur Fitriani Hasan<sup>[2]</sup>

Program Studi Teknologi Informasi, Universitas Muhammadiyah Klaten, Klaten, Jawa Tengah<sup>[1]</sup>

Program Studi Ilmu Komputer, Universitas Muhammadiyah Papua, Kota Jayapura, Papua<sup>[2]</sup>

\*e-mail: ardiansyah@umkla.ac.id<sup>[1]</sup>, hi.fitri@umpapua.ac.id<sup>[2]</sup>

**Abstract**— In improving the economy of developing countries, the highest export commodity is the coffee plant. Indonesia produces 639 thousand tons of coffee every year. Therefore, establishing Indonesia as 4th in the world. However, decreased productivity due to diseases of the coffee plant leaves can reduce the productivity of coffee production. Leaf diseases include Miner, Rust, Phoma, and Cercospora. Based on extant issues in agriculture, utilization such as artificial intelligence, Computer Vision, and Bigdata can decrease the costs incurred to trade with plant diseases. With significant advances in artificial intelligence in Machine Learning comes the Deep Learning method. YOLO is Deep Learning seeded as an object detection compared to other approaches. YOLOv7 is the latest version of the YOLO architecture that can detect speed, high Precision, easy-to-train data, and implementation. The main contribution of this research was to develop using model-based YOLOv7, use coffee leaf costume datasets, data augmentation, and preprocessing datasets. This research utilizes Google Colab and GPU Tesla T4 to get a result F1-score of 0.93, Precision of 0.926, Recall of 0.932, mAP@IoU .5 of 0.956, mAP@IoU .5:.95 of 0.927 for the entire trained data class. However, the best result is the binary class to get a result F1-score of 0.99, Precision of 0.991, Recall of 1, mAP@IoU .5 of 0.998, mAP@IoU .5:.95 of 0.994.

**Keywords**— YOLOv7, Object Detection, Leaf diseases, Computer Vision

**Abstrak**— Dalam peningkatan ekonomi negara berkembang komoditas ekspor tertinggi adalah tanaman kopi. Seperti halnya Indonesia yang mampu memproduksi kopi 639ribu ton pertahun sehingga menempatkan Indonesia urutan 4 di dunia. Namun, penurunan produktifitas dikarenakan penyakit pada daun tanaman kopi menjadi hal yang dapat menurunkan produktifitas dari produksi kopi. Penyakit daun antara lain Miner, Rust, Phoma, Cercospora. Berdasarkan permasalahan yang ada pada bidang agrikultur pemanfaatan seperti kecerdasan buatan, Computer Vision, serta Bigdata mampu mengurangi biaya yang dikeluarkan untuk menangani penyakit tanaman. Dengan kemajuan signifikan pada kecerdasan buatan pada Machine Learning muncul metode Deep Learning. YOLO menjadi Deep Learning yang diunggulkan sebagai objek deteksi dibandingkan dengan metode lain. YOLOv7 merupakan versi terbaru dari arsitektur YOLO yang memiliki karakter pada kecepatan deteksi, presisi tinggi, dan mudah dalam latihan data serta di implementasikan. Pada penelitian yang dilakukan mengusulkan penggunaan dataset kostum daun kopi, penggunaan model-based YOLOv7, data augmentation, serta preprosesing dataset menjadi kontribusi pada penelitian ini. Pada penelitian ini menggunakan Google Colab dengan GPU Tesla T4

mendapatkan hasil F1-score 0.93, Precision 0.926, Recall 0.932, mAP@IoU .5 0.956, mAP@IoU .5:.95 0.927 untuk keseluruhan kelas data yang dilatih. Sedangkan hasil terbaik untuk binary class F1-score 0.99, Precision 0.991, Recall 1, mAP@IoU .5 0.998, mAP@IoU .5:.95 0.994.

**Kata Kunci**— YOLOv7, Object Detection, Leaf diseases, Computer Vision

## I. PENDAHULUAN

Dalam meningkatkan ekonomi sebuah negara berkembang, tanaman kopi menjadi salah satu komoditas yang tinggi untuk di ekspor [1]. Seperti halnya Indonesia yang mampu memproduksi kopi 639 ribu ton per tahun. Dengan jumlah tersebut menempatkan Indonesia pada urutan ke empat di dunia [2]. Meskipun tanaman kopi merupakan komoditas yang tinggi, terdapat beberapa kendala yang mengakibatkan berkurangnya produktivitas seperti usia tanaman, penyakit yang terdapat pada tanaman kopi, berasal dari hama, serta keadaan lingkungan [3][4][5].

Penurunan produktivitas yang sering terjadi dikarenakan adanya penyakit pada bagian batang, akar, buah, ataupun daun tanaman kopi [4]. Sedangkan, Penyakit pada daun tanaman kopi seperti Miner, Rust, Phoma dan Cercospora [1]. Untuk ke efektifan dalam pengontrolan penyakit perlu adanya pengenalan tingkat persentasi keparahan dari penyakit tanaman kopi [6][7]. Pengukuran dapat dilakukan dengan melihat serta mengamati daun tanaman kopi robusta, arabika, ataupun liberika untuk meningkatkan produktivitas dari kopi [4][8].

Pada era sekarang bidang argikultur tidak hanya memanfaatkan kecerdasan buatan saja [9] yang telah mengalami kemajuan untuk peningkatan efisiensi produksi dengan mengurangi efek negatif lingkungan [10]. Pemanfaatan teknologi dapat mengurangi biaya yang dikeluarkan untuk penyemprotan pestisida pada tanaman sebanyak 90% [11]. Kombinasi kecerdasan buatan dengan Computer Vision dalam waktu belakang telah menjadi sesuatu yang sangat baik dalam penerapan di bidang argikultur [12] bahkan terkadang dalam prakteknya ditambahkan dengan teknik Bigdata [13] untuk menyelesaikan seperti pendeteksian penyakit tanaman [14]. Kecerdasan buatan yang mengalami kemajuan secara signifikan terkhusus pada Machine Learning sehingga muncul Deep Learning (DL) [6].

Penggunaan metode DL seperti Convolutional Neural

Networks (CNN) sering kali menjadi solusi dalam memecahkan sebuah masalah di berbagai bidang seperti bidang argikultur [15]. CNN pada dasarnya dibagi menjadi tahapan yang mempertimbangan sekumpulan objek kecil dari lokasi objek yang di distribusikan dan tahapan klasifikasi pada jaringan yang terpadu [16]. Sedangkan metode deteksi objek dengan pendekatan DL menempatkan *You Only Look Once* (YOLO) menjadi metode yang lebih diunggulkan dari metode yang lain [17] dikarenakan YOLO dalam proses deteksi objek mengeluarkan fitur dari gambar dan mengkalkulasikan *Bounding Box* dengan probabilitas dari kelas yang kemudian memberikan hasil dari kelas objek dan informasi lokasi dari gambar [18].

Pada penelitian yang dilakukan oleh Esgario, dkk terkait identifikasi penyakit pada daun tanaman kopi menggunakan DL dengan menambahkan penggunaan semantik segmentasi untuk meningkatkan *Intersection Over Union* (IoU) pada UNet. Sehingga saat klasifikasi menggunakan ResNet50 mampu menghasilkan akurasi 97.07% [5]. Penelitian yang menggunakan ResNet50 lain dilakukan oleh Suprihanto, dkk mendapatkan akurasi dan f1-score untuk binary kelas 92.68% dan 92.88%. Namun, penurunan akurasi terjadi pada multi kelas mendapatkan akurasi 88.98% dan f1-score 88.44% [2]. Penelitian klasifikasi penyakit tanaman kopi yang dilakukan Sabrina dan Al-Maki menggunakan CNN dengan arsitektur EfficientNet-B0, data augmentasi, serta evaluasi performa seperti Adam, RMSprop. Dari penelitian tersebut mendapatkan hasil akurasi 91% dan f1-score 91% secara keseluruhan [4].

Penelitian yang lain dilakukan Marcos A, dkk menggunakan model CNN dengan spesifikasi perangkat keras NVIDIA GeForce GTX 1050 Ti, Core™ i5-4440 CPU 3.10 GHz, 8 Gb RAM. Sedangkan saat latih data penelitian ini menggunakan 500 epochs dan mendapatkan akurasi 95% [19]. Sedangkan penelitian yang dilakukan oleh Önlér E menggunakan beberapa model dari YOLOv5 mendapatkan hasil beragam. YOLOv5x hasil dari presisi, recall, mAP @IoU :0.5, mAP @IoU :0.5:0.95 yaitu: 0.743, 0.697, 0.588, 0.464. YOLOv5l hasil dari presisi, recall, mAP @IoU :0.5, mAP @IoU :0.5:0.95 yaitu: 0.747, 0.690, 0.589, 0.463. YOLOv5m hasil dari presisi, recall, mAP @IoU :0.5, mAP @IoU :0.5:0.95 yaitu: 0.746, 0.686, 0.591, 0.464. YOLOv5s hasil dari presisi, recall, mAP @IoU :0.5, mAP @IoU :0.5:0.95 yaitu: 0.740, 0.697, 0.575, 0.451 [11].

Berdasarkan penelitian yang telah dilakukan penelitian ini bertujuan untuk mendeteksi penyakit daun tanaman kopi serta mengukur performa dari YOLOv7 yang merupakan edisi terbaru dari YOLO series tanpa menaikan *Inference Cost* dari *Object Detection* [20]. Penggunaan dataset kostum daun kopi. Penggunaan model-based YOLOv7, data augmentation, serta preprosesing dataset menjadi kontribusi pada penelitian ini.

## II. METODOLOGI

### A. Dataset

Penggunaan dataset pada penelitian yang dilakukan menggunakan dataset sekunder dari *opensource* Kaggle. Dari dataset tersebut di dapatkan 4 jenis penyakit daun tanaman kopi atau kelas yaitu: *Miner leaf*, *Phoma leaf*, *Rust leaf*, *Good leaf*. Gambar yang terdapat pada dataset tersebut memiliki ukuran

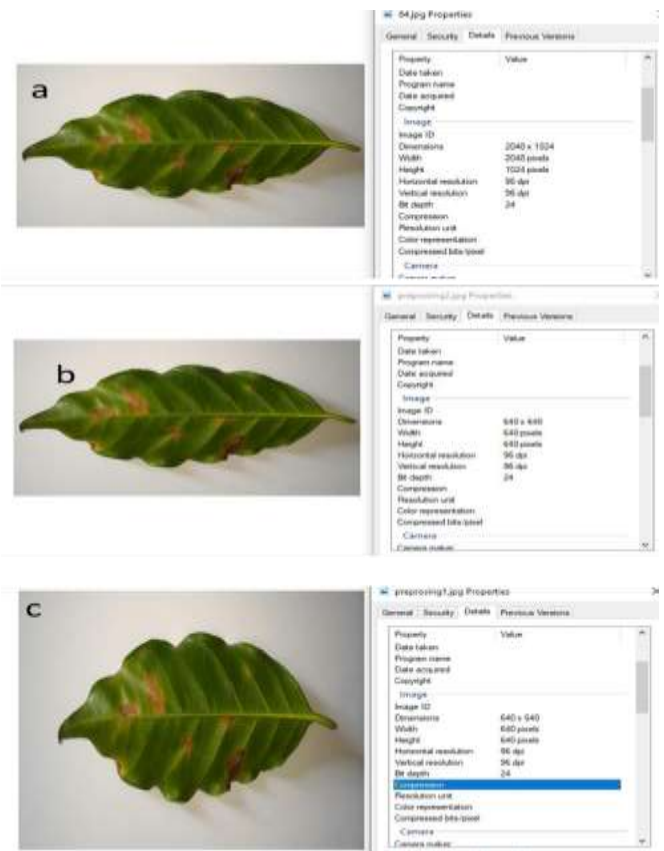
dimensi 2048 x 1024pixel dengan jenis RGB dan sebanyak 1427 gambar. Dari dataset yang digunakan akan di distribusikan ke tiap kelas nya masing masing sebelum dan sesudah proses *preprocessing* seperti yang terlihat pada table 1.

### B. Preprosesing

Preprosesing merupakan representasi peningkatan fitur pembersihan gambar atau memperbaiki dari *noise* atau nilai yang tidak lengkap pada gambar untuk proses tambahan [21]. Penggunaan preprosesing seperti pemotongan gambar, *resize* [22], dan menghapus dataset ganda untuk mendapatkan bentuk daun yang lebih baik untuk digunakan sebagai dataset [4]. Pada penelitian semua gambar dilakukan *resize* 640 x 640pixel tanpa merusak struktur gambar asli seperti yang terlihat pada gambar 1 yang merepresentasikan perubahan ukuran gambar secara visual meskipun gambar dengan ukuran sama dapat berakibatkan struktur gambar menjadi berbeda. Selain itu, penghapusan gambar ganda pada dataset hasil dari preprosesing dataset terlihat pada tabel 1.

TABEL I. PREPROSESING DATASET DAUN TANAMAN KOPI

Dataset	Preprocessing Dataset				
	Good	Miner	Phoma	Rust	Total
1664	97	115	121	94	421



Gambar. 1. Gambar dataset asli 2048x1024 (a) Gambar resize 640x640 (b) Gambar resize 640x640 namun tidak digunakan untuk dataset (c)

Preprosesing yang dilakukan bertujuan untuk tetap menjaga

dataset *training* dalam bentuk yang sama dengan dataset asli meskipun ukurannya berubah. Serta menyesuaikan proses latih data YOLOv7 yang menggunakan ukuran 640x640.

C. Data Augmentasi

DL membutuhkan dataset yang sangat banyak [23] sehingga membutuhkan teknik data augmentasi yang merupakan lanjutan dari pre-prosesing gambar [21] Augmentasi sangat disarankan kepada peneliti [24] bertujuan untuk meningkatkan jumlah dari dataset tanpa harus menambahkan dataset yang dimiliki peneliti [25]. Pada awalnya augmentasi diperkenalkan berguna untuk menangani sebuah jumlah dataset yang *Imbalanced* [25].

Pada penelitian yang dilakukan menggunakan augmentasi sebagai berikut: rotasi = 20, *Zoom Range* = 0.15, *Width Shift Range* = 0.2, *Height Shift Range* = 0.2, *Shear Range* = 0.15, *Horizontal Flip*, *Fill Mode* = *Nearest*. Hasil gambar data augmentasi yang digunakan untuk penelitian ini terlihat pada tabel 2 berikut:

TABEL II. HASIL DATA AUGMENTASI

Hasi Data Augmentasi				
Good	Miner	Phoma	Rust	Total
1301	1496	1214	1090	5101

D. Annotasi

Anotasi merupakan proses sebelum melakukan latih data [26] yang bertujuan merepresentasikan *Ground Truth* [27] dan label kelas dari objek pada citra yang akan digunakan [28] seperti yang terlihat pada gambar 2 anotasi menggunakan LabelImg serta tabel 3 menjelaskan tentang kelas pada objek. YOLO format menjadi anotasi yang digunakan pada penelitian akan simpan ke dalam file berekstensi TXT yang berisikan informasi Id Kelas, *x*, *y*, *width* (*w*), *height* (*h*) [29]. Proses untuk mendapatkan nilai dari *x*, *y*, *w*, *h* menggunakan persamaan matematika seperti berikut [26]:

$$x = \frac{(x_{min} + x_{max})}{2 * W} \tag{1}$$

$$y = \frac{(y_{min} + y_{max})}{2 * H} \tag{2}$$

$$w = \frac{(x_{max} - x_{min})}{W} \tag{3}$$

$$h = \frac{(y_{max} - y_{min})}{H} \tag{4}$$



Gambar. 2. Anotasi dataset menggunakan labelImg

TABEL III. LABEL KELAS PADA DATASET

Id kelas	Nama kelas
0	good_leaf
1	miner_leaf
2	phoma_leaf
3	rust_leaf

E. YOLO

Algoritma DL menjadi salah satu algoritma yang telah banyak digunakan serta membuktikan untuk mendeteksi objek lebih akurat dengan parameter akurasi dan kecepatan [30]. YOLO dengan metode *one-stage* mampu mendeteksi serta mengklasifikasi objek dengan sangat baik [31]. YOLOv7 merupakan versi terbaru dari arsitektur YOLO yang memiliki karakteristik pada kecepatan deteksi, presisi tinggi, dan mudah dalam latih data serta di implementasikan [20]. Objek deteksi YOLOv7 menggunakan metode *bag-of-freebies* untuk mendapatkan hasil akurasi yang tinggi serta kecepatan inferensi yang lebih baik, sesuai dengan *State-Of-The-Art* [32].

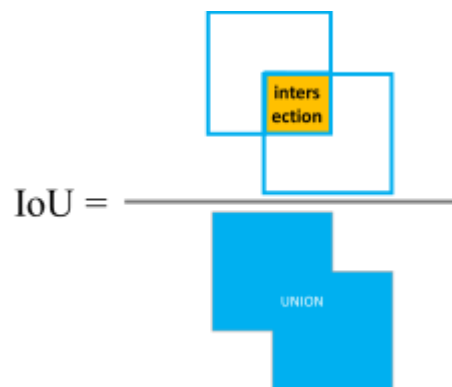
Selain itu, YOLOv7 memiliki kelebihan dari YOLOv4 karena dapat menurunkan 75% parameter, 35% komputasi, dan mendapatkan 1.5% AP yang lebih tinggi meskipun menggunakan pengaturan yang sama [32] seperti yang terlihat pada gambar 3.

Model	#Param	FLOPs	AP					
			val	val 50	val 75	val S	val M	val L
YOLOv4	64.4M	142.8G	49.70%	68.20%	54.30%	32.90%	54.80%	63.70%
YOLOv4-u5 (r6.1)	46.5M	109.1G	50.20%	68.70%	54.60%	33.20%	55.50%	63.70%
YOLOv4-CSP	52.9M	120.4G	50.30%	68.60%	54.90%	34.20%	55.60%	65.10%
YOLOv4-CSP	52.9M	120.4G	50.80%	69.50%	55.30%	33.70%	56.00%	65.40%
YOLOv7	39.9M	104.7G	51.20%	69.70%	55.50%	35.20%	56.00%	66.70%
improvement	-43%	-15%	+0.4	+0.2	+0.2	+1.5	=	+1.3

Gambar. 3. Perbandingan objek detektor size 640

F. Matrix Evaluasi Performa

Evaluasi performa dibedakan menjadi 2 jenis yaitu, hasil deteksi dan performa klasifikasi [33]. Matrix evaluasi performa dalam objek deteksi untuk menganalisa akurasi dari arsitektur YOLO menggunakan Intersection over Union (IoU) [34] berguna mendapatkan nilai kesalahan dari kotak prediksi dan kotak kebenaran seperti yang terlihat pada gambar 4 dibawah ini.



Gambar. 4. Perhitungan Intersection over Union (IoU)

Selain IoU terdapat juga evaluasi hasil posisi *Bounding Box* seperti *Precision*, *F1-Score*, *Recall*,  $mAP@IoU = 0.5$  dan,  $mAP@IoU = 0.5:0.95$  [33] terlihat pada persamaan 5 - 8. Sedangkan *Confusion Matrix* merupakan analisis secara general untuk merepresentasikan kinerja dari klasifikasi [33]. Pada proses pengklasifikasian memiliki kondisi untuk menentukan nilai objek yaitu [34]:

- *True Positive* (TP) jika IoU memiliki nilai lebih dari 0.5
- *False Positive* (FP) jika IoU memiliki nilai kurang dari 0.5
- *False Negative* (FN) jika pendeteksian yang dilakukan tidak sesuai dengan kelas yang telah di latih dan memiliki nilai IoU lebih dari 0.5

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (7)$$

$$F1 = 2 * \frac{P * R}{P + R} \quad (8)$$

### III. HASIL DAN PEMBAHASAN

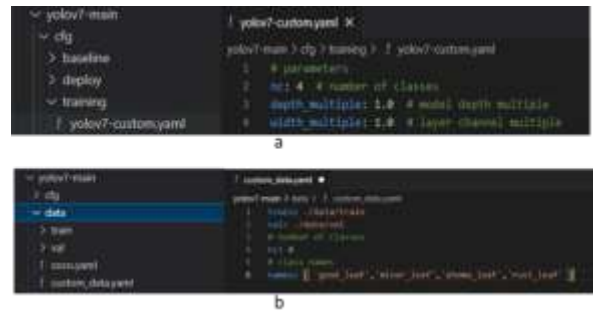
Pada bagian ini merupakan bagian yang merepresentasikan proses pelatihan dataset, evaluasi performa, serta pendeteksian objek. Penelitian ini menggunakan *Google Colab* dengan spesifikasi *Graphical Processing Unit* (GPU) NVIDIA Tesla t4, CUDA versi 11.2 dengan parameter saat latih data adalah 300 epochs, 16 batch-size, 8 worker, SGD optimizer, dan model-based YOLOv7.

#### A. Pelatihan dataset

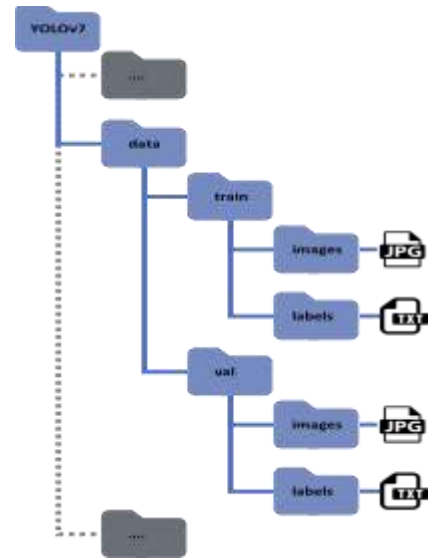
Pada penelitian yang dilakukan saat pelatihan sebelumnya dilakukan preprosesing dataset asli seperti yang dijelaskan pada bagian preprosesing lebih spesifik tergambar pada gambar 1, 2, 3. Setelah dataset melalui proses preprosesing akan dilakukan proses augmentasi yang dimana bertujuan untuk menambahkan jumlah data yang berasal dari preprosesing 421 menjadi 5101 citra latih. Selanjutnya, proses label anotasi menggunakan *LabelImg* secara manual yang menyimpan informasi tentang id kelas, nilai x, nilai y, w, dan h kedalam file label berekstensi txt sesuai dengan nama file gambar.

File label dan gambar yang telah melalui proses anotasi dibagi menjadi train dan val dengan presentasi 80:20 yang di letakan pada *directory* data seperti yang terlihat pada gambar 6. Pembagian data train dan val yaitu Good 1040 data latih dan 216 data validasi, Miner 1196 data latih dan 300 data validasi, Phoma 971 data latih dan 243 data validasi, dan Rust 872 data latih dan 218 data validasi.

Selanjutnya proses yang dilakukan adalah membuat file data *custom.yaml* dan file training *yolov7-custom.yaml* yang menjelaskan lokasi file train dan val serta menjelaskan jumlah (*number of classes*), dan nama kelas seperti yang terlihat pada gambar 5.



Gambar. 5. Bagian yolov7-custom.yaml beserta lokasi folder (a), custom\_data.yaml beserta lokasi folder (b)



Gambar. 6. Penempatan dataset kostum pada YOLOv7

Proses pelatihan data pada penelitian ini menggunakan beberapa parameter *weights*, *epochs*, *batch-size*, *img-size*, *optimizer*, *workers* sebagai berikut *yolov7*, 300, 16, 640, *Stochastic Gradient Descent* (SGD), dan 8.

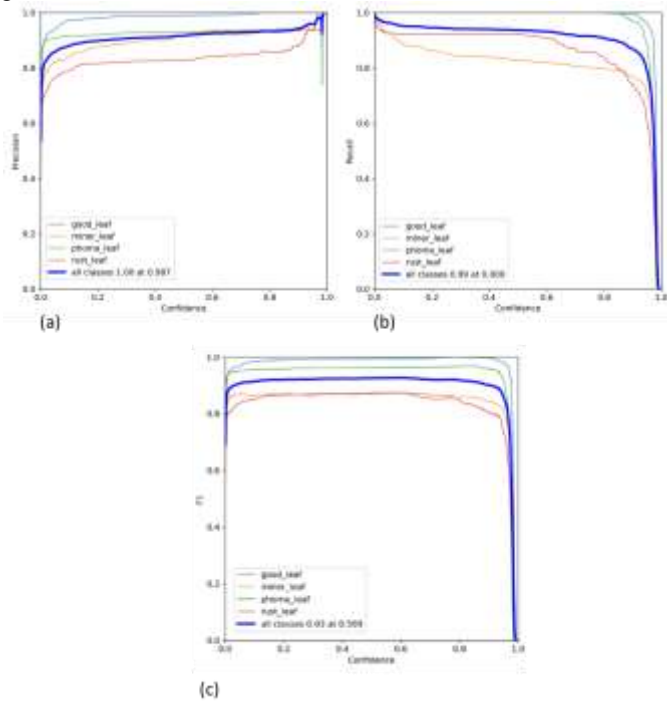
#### B. Evaluasi Performa

Evaluasi performa yang digunakan pada penelitian untuk mengukur hasil dari latih data menggunakan parameter  $mAP@IoU 0.5$ ,  $mAP@IoU 0.5:0.95$ , *Precision*, *Recall* mendapatkan nilai  $mAP@IoU 0.5$  dan  $mAP@IoU 0.5:0.95$  secara keseluruhan dari kelas yang latih 0.956, 0.927 seperti yang terlihat pada tabel 4.

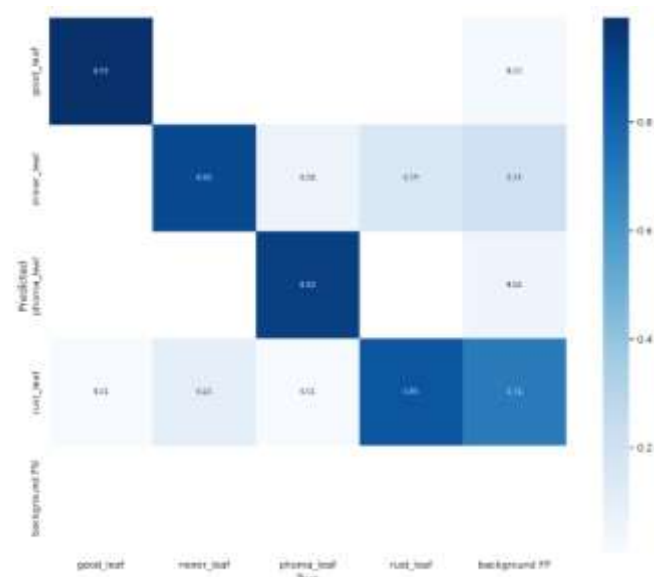
TABEL IV. Evaluasi Performa pada dataset

Class	Image / Labels	P	R	$mAP@IoU 0.5$	$mAP@IoU 0.5:0.95$
All	1022 / 1022	0.926	0.932	0.956	0.927
Good	1022 / 261	0.991	1	0.998	0.994
Miner	1022 / 300	0.937	0.82	0.921	0.844
Phoma	1022 / 242	0.932	1	0.978	0.951
Rust	1022 / 219	0.843	0.909	0.928	0.919

Grafik evaluasi performa lainnya *FI-score*, *Precision*, *Recall*, dan *Confusion matrix* terlihat pada gambar 7 dan gambar 8.



Gambar 7. Grafik *Precision* (a), Grafik *Recall* (b), Grafik *F1-Score* (c)



Gambar 8. Confusion Matrix latihan dataset

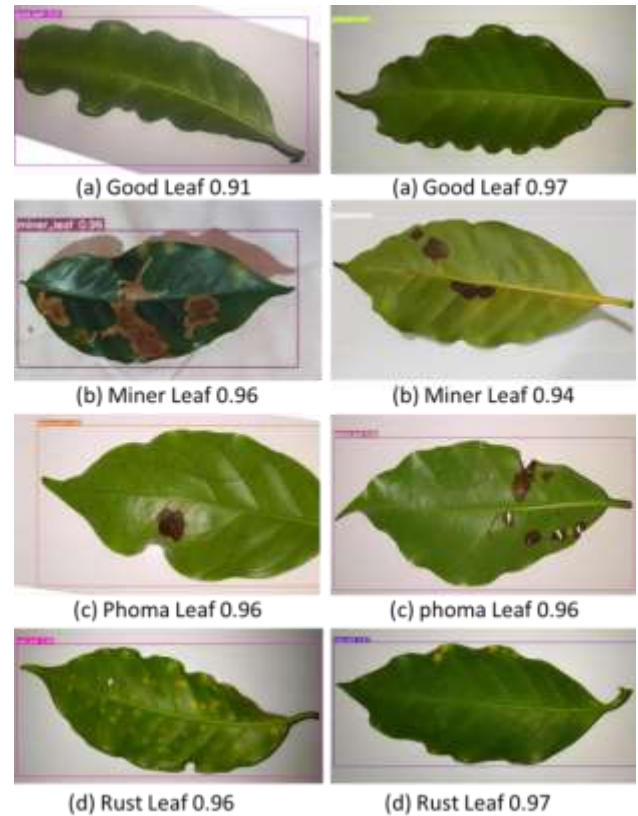
C. Pendeteksian Objek

Pendeteksian objek untuk menentukan lokasi objek dan pengklasifikasian kelas pada penelitian yang dilakukan menggunakan *weight* atau model terbaik dari hasil latihan data yang sebelumnya dilakukan. Parameter lainnya seperti terlihat pada tabel 5.

TABEL V. Deteksi Objek Parameter

Argumen	Detail Argumen	Default	Parameter yang digunakan
Weights	model.pt path(s)	yolov7.pt	Best.pt
Augment	augmented inference	True/False	False
Conf-thres	object confidence threshold	0.25	0.5
Img-size	inference size (pixels)	640	640

Percobaan pendeteksian objek serta pengklasifikasi penyakit pada daun tanaman kopi terlihat pada gambar 9 dengan intensitas ujicoba 2 kali untuk setiap kelas.



Gambar 9. Hasil deteksi dan klasifikasi penyakit tanaman kopi

IV. KESIMPULAN

Deteksi dan klasifikasi objek penyakit tanaman kopi menggunakan YOLOv7 yang berkonsentrasi pada preprocessing, data augmentasi, anotasi atau labeling objek mendapatkan hasil yang baik dibuktikan dengan evaluasi performa terlihat pada tabel 4. Serta pelatihan dataset menggunakan parameter 300 epochs, 8 worker, 16 batch-size, SGD optimizer, yolov7 model, Pada penelitian yang dilakukan menggunakan perangkat Google Colab dengan GPU Tesla T4 mendapatkan hasil *FI-score* 0.93, *Precision* 0.926, *Recall* 0.932, mAP@IoU .5 0.956, mAP@IoU .5:.95 0.927 untuk keseluruhan kelas data yang dilatih. Sedangkan hasil terbaik untuk *binary class FI-score* 0.99, *Precision* 0.991, *Recall* 1, mAP@IoU .5 0.998, mAP@IoU .5:.95 0.994.

REFERENCES

- [1] D. Novtahaning, H. A. Shah, and J.-M. Kang, "Deep Learning Ensemble-Based Automated and High-Performing Recognition of Coffee Leaf Disease," *Agriculture*, vol. 12, no. 11, p. 1909, 2022.
- [2] S. Suprihanto, I. Awaludin, M. Fadhil, and M. A. Z. Zulfikar, "Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta," *Jurnal Informatika*, vol. 9, no. 2, pp. 116–122, Oct. 2022.
- [3] D. Irfansyah, M. Mustikasari, and A. Suroso, "Arsitektur Convolutional Neural Network (CNN) Alexnet Untuk Klasifikasi Hama Pada Citra Daun Tanaman Kopi," *Jurnal Informatika*, vol. 6, no. 2, 2021.
- [4] S. A. Sabrina and W. F. al Maki, "Klasifikasi Penyakit pada Tanaman Kopi Robusta Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *e-Proceeding of Engineering*, vol. 9, no. 3, p. 119, Jun. 2022.
- [5] J. G. M. Esgario, P. B. C. de Castro, L. M. Tassis, and R. A. Krohling, "An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning," *Information Processing in Agriculture*, vol. 9, no. 1, pp. 38–47, Mar. 2022, doi: 10.1016/j.inpa.2021.01.004.
- [6] J. G. M. Esgario, R. A. Krohling, and J. A. Ventura, "Deep learning for classification and severity estimation of coffee leaf biotic stress," *Comput Electron Agric*, vol. 169, Feb. 2020, doi: 10.1016/j.compag.2019.105162.
- [7] L. X. Boa Sorte, C. T. Ferraz, F. Fambrini, R. D. R. Goulart, and J. H. Saito, "Coffee leaf disease recognition based on deep learning and texture attributes," in *Procedia Computer Science*, 2019, vol. 159, pp. 135–144. doi: 10.1016/j.procs.2019.09.168.
- [8] F. J. P. Montalbo and A. A. Hernandez, "An Optimized Classification Model for Coffea Liberica Disease using Deep Convolutional Neural Networks," in *2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA)*, 2020, pp. 213–218.
- [9] R. Yang and Y. Yu, "Artificial convolutional neural network in object detection and semantic segmentation for medical imaging analysis," *Front Oncol*, vol. 11, p. 638182, 2021.
- [10] M. Dhanaraju, P. Chenniappan, K. Ramalingam, S. Pazhanivelan, and R. Kaliaperumal, "Smart Farming: Internet of Things (IoT)-Based Sustainable Agriculture," *Agriculture*, vol. 12, no. 10, p. 1745, 2022.
- [11] E. Önlü, "REAL TIME PEST DETECTION USING YOLOv5," *International Journal of Agricultural and Natural Sciences*, vol. 14, no. 3, pp. 232–246, 2021.
- [12] V. Kakani, V. H. Nguyen, B. P. Kumar, H. Kim, and V. R. Pasupuleti, "A critical review on computer vision and artificial intelligence in food industry," *J Agric Food Res*, vol. 2, p. 100033, 2020.
- [13] O. Balitskii and V. Kolesnikov, "Identification of wear products in the automotive tribotechnical system using computer vision methods, artificial intelligence and big data," in *2019 XIth International Scientific and Practical Conference on Electronics and Information Technologies (ELIT)*, 2019, pp. 24–27.
- [14] W. Zhao, W. Yamada, T. Li, M. Digman, and T. Runge, "Augmenting Crop Detection for Precision Agriculture with Deep Visual Transfer Learning—A Case Study of Bale Detection," *Remote Sens (Basel)*, vol. 13, no. 1, p. 23, 2020.
- [15] M. E. Karar, F. Alsunaydi, S. Albusaymi, and S. Alotaibi, "A new mobile application of agricultural pests recognition using deep learning in cloud computing system," *Alexandria Engineering Journal*, vol. 60, no. 5, pp. 4423–4432, 2021.
- [16] O. Wosner, G. Farjon, and A. Bar-Hillel, "Object detection in agricultural contexts: A multiple resolution benchmark and comparison to human," *Comput Electron Agric*, vol. 189, p. 106404, 2021.
- [17] W. Fang, L. Wang, and P. Ren, "Tinier-YOLO: A real-time object detection method for constrained environments," *IEEE Access*, vol. 8, pp. 1935–1944, 2019.
- [18] S. Lu, B. Wang, H. Wang, L. Chen, M. Linjian, and X. Zhang, "A real-time object detection algorithm for video," *Computers & Electrical Engineering*, vol. 77, pp. 398–408, 2019.
- [19] A. P. Marcos, N. L. S. Rodvalho, and A. R. Backes, "Coffee leaf rust detection using convolutional neural network," in *2019 XV Workshop de Visão Computacional (WVC)*, 2019, pp. 38–42.
- [20] D. Wu et al., "Detection of Camellia oleifera Fruit in Complex Scenes by Using YOLOv7 and Data Augmentation," *Applied Sciences*, vol. 12, no. 22, p. 11318, Nov. 2022, doi: 10.3390/app122211318.
- [21] J. Chaki and N. Dey, *A beginner's guide to image preprocessing techniques*. CRC Press, 2018.
- [22] G. Fenu and F. M. Mallocci, "DiaMOS plant: A dataset for diagnosis and monitoring plant disease," *Agronomy*, vol. 11, no. 11, p. 2107, 2021.
- [23] A. Morbekar, A. Parihar, and R. Jadhav, "Crop disease detection using YOLO," in *2020 International Conference for Emerging Technology (INCET)*, 2020, pp. 1–5.
- [24] P. Li, X. Liu, and X. Xie, "Learning Sample-Specific Policies for Sequential Image Augmentation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 4491–4500.
- [25] J. A. Pandian, G. Geetharamani, and B. Annette, "Data augmentation on plant leaf disease image dataset using image manipulation and deep learning techniques," in *2019 IEEE 9th international conference on advanced computing (IACC)*, 2019, pp. 199–204.
- [26] Y. Jamtsho, P. Riyamongkol, and R. Waranusast, "Real-time Bhutanese license plate localization using YOLO," *ICT Express*, vol. 6, no. 2, pp. 121–124, 2020.
- [27] C. Tang, G. Zhang, H. Hu, P. Wei, Z. Duan, and Y. Qian, "An improved YOLOv3 algorithm to detect molting in swimming crabs against a complex background," *Aquac Eng*, vol. 91, Nov. 2020, doi: 10.1016/j.aquaeng.2020.102115.
- [28] S. Tabassum, S. Ullah, N. H. Al-Nur, and S. Shatabda, "Poribohon-BD: Bangladeshi local vehicle image dataset with annotation for classification," *Data Brief*, vol. 33, 2020.
- [29] J. Dhameeshkar, S. A. Aniruthan, R. Karthika, and L. Parameswaran, "Deep Learning based Detection of potholes in Indian roads using YOLO," in *2020 International conference on inventive computation technologies (ICICT)*, 2020, pp. 381–385.
- [30] A. I. B. Parico and T. Ahamed, "Real time pear fruit detection and counting using YOLOv4 models and deep SORT," *Sensors*, vol. 21, no. 14, p. 4803, 2021.
- [31] Y. Yin, H. Li, and W. Fu, "Faster-YOLO: An accurate and faster object detection method," *Digital Signal Processing: A Review Journal*, vol. 102, Jul. 2020, doi: 10.1016/j.dsp.2020.102756.
- [32] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.
- [33] I. Ahmad et al., "Deep Learning Based Detector YOLOv5 for Identifying Insect Pests," *Applied Sciences*, vol. 12, no. 19, p. 10167, 2022.
- [34] A. F. Bayram, C. Gurkan, A. Budak, and H. KARATAŞ, "A Detection and Prediction Model Based on Deep Learning Assisted by Explainable Artificial Intelligence for Kidney Diseases," *Avrupa Bilim ve Teknoloji Dergisi*, no. 40, pp. 67–74, 2022.