

DETEKSI KEMATANGAN BUAH KOPI MENGGUNAKAN ALGORITMA LUCY-RICHARDSON DAN *YOU ONLY LOOK ONCE (YOLOV8)* DENGAN PENDEKATAN *PIPELINE*

Ogani Suka'^{1*}, Aryo Michael², Irene Devi Damayanti³

^{1*}Program Studi Teknik Informatika, Universitas Kristen Indonesia Toraja, Tana Toraja, Sulawesi Selatan
Email: ^{1*}oganilestari006@gmail.com, ²aryomichael@ukitoraja.ac.id, ³irenedamayanti@ukitoraja.ac.id

(Naskah masuk: 10 November 2025, direvisi: 15 November 2025, diterima: 20 Desember 2025)

Abstrak

Penelitian ini mengembangkan sistem deteksi dan klasifikasi tingkat kematangan buah kopi menggunakan model YOLOv8. Data primer dikumpulkan dari perkebunan PT. Toarco Jaya dan dianotasi dalam empat kelas: mentah, setengah matang, matang, dan terlalu matang. Setelah preprocessing, model YOLOv8n dilatih dengan dua skema: YOLO saja dan kombinasi YOLO + Lucy Richardson. Hasil menunjukkan akurasi 85,7% untuk model YOLO dan 84,5% untuk kombinasi, dengan peningkatan signifikan pada mAP untuk model kombinasi. Penelitian ini menyimpulkan bahwa YOLOv8 efektif dalam mendeteksi dan mengklasifikasi buah kopi dengan kemampuan generalisasi yang baik terhadap variasi visual.

Kata Kunci: Kematangan Buah Kopi¹, *Lucy-Richardson*², YOLOv8³, Deteksi Objek⁴, *Confusion Matrix*⁵, mAP⁶, *Pipeline*⁷, Pemrosesan Citra⁸.

DETECTION OF COFFEE FRUIT MATURITY USING THE LUCY-RICHARDSON AND YOU ONLY LOOK ONCE (YOLOV8) ALGORITHM WITH THE PIPELINE APPROACH

Abstract

This research focuses on developing a system for detecting and classifying the ripeness of coffee cherries using the YOLOv8 model. Primary data was collected from the PT. Toarco Jaya plantation and annotated into four classes: unripe, half-ripe, ripe, and overripe. After preprocessing, the YOLOv8n model was trained using two schemes: YOLO alone and a combination of YOLO + Lucy Richardson. The results showed an accuracy of 85.7% for the YOLO model and 84.5% for the combination, with a significant improvement in mean Average Precision (mAP) for the combined model. The study concludes that YOLOv8 is effective in detecting and classifying coffee cherries, demonstrating good generalization capabilities across varying visual conditions.

Keywords: Ripeness of Coffee Fruit¹, *Lucy-Richardson*², YOLOv8³, Object Detection⁴, *Confusion Matrix*⁵, mAP⁶, *Pipeline*⁷, Image Processing⁸.

I. PENDAHULUAN

Pertanian adalah salah satu sektor yang sangat penting dalam kehidupan sehari-hari. Sektor ini tidak hanya menyediakan makanan, tetapi juga berkontribusi pada perekonomian masyarakat. Salah satu buah yang banyak disukai dan memiliki nilai ekonomi tinggi adalah buah kopi. Kopi menjadi salah satu komoditas paling berharga di dunia dan produk unggulan yang diekspor dari Indonesia[1]. Sebagai negara penghasil kopi terbesar keempat, Indonesia dikenal dengan berbagai varietasnya, seperti Arabika dan Robusta. Indonesia memiliki rencana pengembangan industri kopi hingga 2026, dengan fokus pada produksi, ekspor, dan konsumsi. Pada 2022, produksi kopi mencapai 793,19 ribu ton, sementara ekspor bersihnya 413,53 ribu ton, dan konsumsi dalam negeri 379,655 ribu ton. Pada 2023, produksi sedikit turun menjadi 789,6 ribu ton, tetapi ekspor naik menjadi 417 ribu ton, sedangkan konsumsi berkurang menjadi 372,6 ribu ton. Proyeksi menunjukkan bahwa produksi akan tetap stabil sekitar 789 ribu ton hingga 2026, namun ekspor terus meningkat hingga 427 ribu ton, sedangkan konsumsi dalam negeri justru turun menjadi 361 ribu ton. Untuk meningkatkan daya saing, Indonesia perlu memperkuat keunggulan komparatif dan meningkatkan pengolahan kopi agar lebih bernilai tinggi di pasar global.[2]

Kualitas kopi sangat dipengaruhi oleh cara dan waktu pemanenan buah kopi. Jika petani memanen buahnya pada waktu yang tepat, biji kopi yang dihasilkan akan memiliki rasa dan aroma yang enak. Namun, jika buahnya dipanen terlalu awal, biji kopi bisa terasa asam dan kurang nikmat. Sebaliknya, jika dipanen terlalu terlambat, buah bisa jatuh atau mulai membusuk, yang jelas akan mengurangi kualitas biji kopi. Tingkat kematangan buah kopi dapat dilihat dari warnanya yang mana buah yang telah matang ditandai dengan warna

merah cerah[3]. Setelah pemanenan, proses penyortiran buah kopi menjadi langkah penting berikutnya. Penyortiran bertujuan untuk memisahkan buah kopi yang belum matang, matang dan terlalu matang. Proses ini biasanya dilakukan secara manual atau menggunakan mesin. Penyortiran secara manual ini sering kali dapat berpengaruh pada kesalahan klasifikasi kualitas buah dan penyortiran manual ini juga membutuhkan waktu yang lama yang berpengaruh pada efisiensi dan produktivitas dalam proses budidaya dan distribusi kopi.[4]

Teknologi pertanian telah berkembang pesat dengan adanya sistem sortir berbasis visi komputer yang digunakan untuk mengklasifikasikan objek berdasarkan bentuk dan warna. Salah satu kelemahan yang diidentifikasi adalah sensitivitas sistem terhadap kondisi pencahayaan ruangan, yang mempengaruhi akurasi deteksi warna dan bentuk. Selain itu, sistem ini masih bergantung pada kalibrasi manual untuk mendapatkan hasil optimal, yang dapat menjadi kendala dalam penerapan industri skala besar[5]. Tantangan-tantangan ini perlu diatasi agar mesin sortir dapat berfungsi lebih efektif dan menghasilkan produk berkualitas tinggi. Dalam mengatasi ini, ada beberapa penelitian telah menerapkan algoritma deteksi *Faster CNN*, *SSD*, dan *YOLO*. Berikut adalah tabel perbandingan antara *Faster CNN*, *SSD*, dan *YOLOv8* dalam mendeteksi objek :

Tabel 1. 1 Perbandingan *Faster CNN*, *SSD*, dan *YOLOv8*

Aspek	<i>Faster CNN</i>	<i>SSD (Single Shot Detector)</i>	<i>YOLOv8</i>
Arsitektur	Model dua tahap: cari dulu objeknya,	Model satu tahap dengan	Model satu tahap (langsun

	baru diklasifikasi kan	fitur dari beberapa skala gambar	g deteksi), struktur modern dan efisien
Kecepatan	Lambat (tidak real-time)	edang (cocok mobile)	Cepat (real-time)
Akurasi	Sangat akurat, cocok untuk deteksi detail tinggi	Cukup akurat, tapi kadang kesulitan dengan objek kecil	Akurat dan konsisten, cocok untuk banyak jenis objek
Kebutuhan Data	Butuh banyak data dan label yang detail untuk performa terbaik	Butuh data cukup banyak untuk hasil stabil	Bisa dilatih dengan dataset ukuran sedang
Penggunaan	Analisis mendalam: medis, forensik, sistem pengawasan yang butuh ketelitian	Aplikasi mobile, sistem robot ringan, penggunaan dengan sumber daya terbatas	Mobil otonom, kamera keamanan, drone, aplikasi cepat

Dari tabel di atas, dapat disimpulkan bahwa pemilihan antara Faster R-CNN, SSD, dan YOLOv8 tergantung pada kebutuhan spesifik aplikasi. aster R-CNN menggunakan dua tahap proses deteksi, menghasilkan akurasi sangat tinggi, tetapi lambat dan membutuhkan data serta komputasi besar cocok untuk analisis mendalam seperti forensik dan medis. SSD merupakan model satu tahap yang lebih cepat dan efisien dibanding

Faster R-CNN, dengan akurasi menengah, cocok untuk sistem ringan seperti aplikasi mobile dan robot kecil. YOLOv8 adalah model satu tahap paling modern yang unggul dalam kecepatan dan akurasi, serta tidak memerlukan data sebanyak Faster R-CNN—ideal untuk aplikasi real-time seperti mobil otonom dan kamera pengawas. Singkatnya, YOLOv8 memberikan keseimbangan terbaik antara kecepatan, akurasi, dan efisiensi dalam berbagai skenario deteksi objek.[6]

Meskipun YOLOv8 menunjukkan kinerja yang sangat baik dalam mendeteksi objek, seperti kematangan buah, ada satu kekurangan signifikan: YOLO tidak memiliki kemampuan untuk mengatasi gambar yang blur. Ketika objek dalam video tidak jelas, akurasi deteksi dapat menurun drastis. Untuk mengatasi masalah ini, penulis menggunakan algoritma *Lucy-Richardson* sebagai langkah awal dalam proses deteksi. Algoritma *Lucy-Richardson* adalah metode pemrosesan citra yang efektif untuk mengurangi efek blur pada gambar. Menggunakan teknik ini, penulis dapat memperbaiki kualitas gambar yang kurang jelas sebelum menerapkan model YOLOv8. Proses ini melibatkan beberapa langkah, di mana algoritma akan menganalisis gambar blur dan mencoba mengembalikan detail yang hilang, sehingga menghasilkan citra yang lebih tajam dan jelas. Setelah gambar diperbaiki, barulah kita menerapkan model YOLOv8 untuk mendeteksi objek. Cara ini, kita dapat memanfaatkan keunggulan YOLOv8 dalam akurasi deteksi sambil meminimalkan dampak dari ketidakjelasan gambar. Pendekatan ini diharapkan akan meningkatkan hasil deteksi secara keseluruhan, terutama dalam situasi di mana kualitas video mungkin terpengaruh oleh gerakan cepat atau pencahayaan yang buruk.

II. TINJAUAN PUSTAKA

A. Penelitian Terkait

Pada tahun 2021, D. R Fauzi dkk meneliti tentang “Implementasi Noise Removal Dan Image Restoration Pada Citra Kabur (Blur) Dengan Menggunakan Metode *Lucy – Richardson Algorithm Techniques*”[7]. Dalam penelitian ini,

algoritma *Lucy Richardson* cukup efektif dalam proses deblurring gambar. Pada iterasi awal, khususnya iterasi ke-1, gambar masih tampak buram karena adanya penambahan blur sebagai bagian dari proses awal. Namun, seiring bertambahnya jumlah iterasi, kualitas gambar semakin membaik. Hasil terbaik diperoleh pada iterasi sekitar 40 kali, meskipun pada iterasi yang terlalu tinggi dapat muncul artifact pada citra. Selain itu, berdasarkan pengujian usability yang melibatkan 23 responden, sistem mendapatkan tingkat kepuasan pengguna sebesar 85,749%. Nilai ini menunjukkan bahwa sistem yang dikembangkan memiliki kualitas yang sangat baik dan diterima dengan baik oleh pengguna.

Dedi Triyanto dkk, melakukan penelitian pada tahun 2024 tentang “Pengembangan Sistem Deteksi Objek Botol Real-Time dengan YOLOv8 untuk Aplikasi Vision”[8]. Dalam penelitian ini, Penelitian ini menunjukkan bahwa YOLOv8 sangat bagus untuk mendeteksi botol plastik dan kaleng—modelnya akurat hampir sempurna. Dari 2900 gambar dan 25 kali pelatihan, hasilnya sangat tinggi: akurasi, presisi, dan recall semuanya di atas 99%. Model ini nggak bikin kesalahan dalam mengenali objek.

“Yolo-V8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah ” di lakukan oleh Yanto dkk pada tahun 2023[9]. memperlihatkan bahwa YOLO-v8 Improved Algorithm memiliki hasil akurasi yang tinggi untuk mendeteksi penggunaan masker muka, dengan akurasi kelas badmask sebesar 94%, mask sebesar 97% dan nomask sebesar 95%. Nilai F1-Confidence, Precision, dan Recall semua kelasnya juga tinggi, yaitu masing-masing sebesar 0.94, 0.96, dan 0.978. Waktu komputasi rata-rata yang diperlukan oleh algoritma ini hanya 17ms. Sehingga diharapkan mempermudah deteksi pemakaian masker wajah dan memastikan bahwa pemakaian masker dilakukan secara benar dan tepat, memastikan keselamatan dan kesehatan semua orang yang berada di lingkungan tersebut.

Pada tahun 2023, Lejar Satya dkk meneliti tentang “Sistem Pendeteksi Plat Nomor Polisi Kendaraan Dengan Arsitektur Yolov8”[10]. Dalam penelitian ini, YOLOv8 berhasil menghasilkan model yang mampu mendeteksi berbagai jenis kendaraan dan plat nomor dengan tingkat akurasi yang memadai. Skor precision sebesar 0,871 dan nilai recall sebesar 0,85. Penggunaan PaddleOCR sebagai metode pengenalan karakter pada plat nomor juga memberikan hasil yang baik dalam mengenali karakter-karakter pada plat nomor polisi, termasuk pada karakter yang terdistorsi atau memiliki kualitas gambar yang rendah. Pada uji coba menggunakan berbagai gambar, sistem mampu mengenali objek kendaraan dan plat nomor polisi dengan baik pada sebagian besar kasus. Meskipun demikian, masih terdapat beberapa kasus di mana model mengalami kesulitan dalam mengenali objek, terutama pada kelas Truk dan karakter tanggal pada plat nomor polisi. Ini menunjukkan bahwa masih ada ruang untuk peningkatan dalam pengenalan karakter yang lebih akurat.

B. Landasan Teori

1) Computer Vision

Computer Vision adalah cabang dari *Deep Learning* dan *Artificial Intelligence* yang memungkinkan mesin untuk melihat dan memahami lingkungan seperti manusia. Penerapannya ini terbagi menjadi tiga kategori utama:

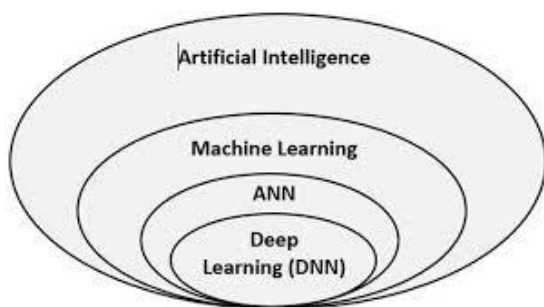
1. *Low-Level Vision* : Berfokus pada pemrosesan dasar gambar seperti penghilangan derau, pemfilteran, penajaman, dan klasifikasi tekstur. Mesin memahami gambar sebagai objek digital.
2. *Intermediate-Level Vision* : Meningkatkan kemampuan mesin untuk mendeteksi garis, tepi, dan pola. Pengenalan pola mencakup analisis bentuk seperti batas, tepi, lingkaran, dan elips, serta pencocokan pola antar objek.
3. *3D Vision* dan Gerakan : Membantu mesin memahami dunia dalam tiga dimensi dengan mengidentifikasi bentuk dan bayangan, serta

mengambil informasi dari objek yang bergerak. Teknologi ini telah diterapkan dalam kehidupan sehari-hari, misalnya pada kamera pemantau untuk deteksi plat kendaraan.

2) *Machine Learning*

Machine Learning adalah teknik pemrograman komputer yang memungkinkan sistem belajar dari data dan membangun model atau inferensi tanpa instruksi eksplisit. Algoritma ini dapat mengenali pola dan membuat prediksi berdasarkan data historis, menjadikannya bagian dari kecerdasan buatan (AI). [11]

3) *Deep Learning*



Gambar 2. 1 *Deep Learning*

Deep learning adalah cabang dari *machine learning* yang menggunakan jaringan saraf tiruan dengan banyak lapisan untuk mengekstrak fitur dari data secara otomatis. Keunggulan utama *deep learning* dibandingkan metode tradisional adalah kemampuannya dalam menangani data dalam jumlah besar dan kompleks, serta kemampuannya dalam belajar dari data tanpa perlu pemrograman eksplisit. [12]

4) *Deteksi Objek*

Deteksi objek adalah proses mengenali dan menemukan berbagai objek dalam gambar atau video. Dalam dunia pertanian, deteksi objek sangat berguna, terutama untuk memantau tanaman dan buah-buahan. Misalnya, dalam budidaya buah markisa, teknologi ini membantu petani mengetahui kapan waktu yang tepat untuk memanen buah. Dengan deteksi otomatis, petani bisa melihat apakah buah markisa sudah matang dan siap dijual, sehingga kualitas dan rasa buah

dapat terjaga dengan baik. Ini menjadi solusi yang efisien dan mengurangi risiko memanen buah yang belum matang atau terlalu matang. komputer sebagai perangkat lunak untuk memproses setiap piksel dari sebuah gambar.

5) *Pengolahan Citra Digital*

Deteksi objek adalah proses mengenali berbagai jenis objek dan mengklasifikasikannya ke dalam kategori tertentu; ini merupakan salah satu tugas utama dalam *computer vision*. *Computer vision* adalah bidang yang berkembang pesat dan kemajuannya sangat penting bagi revolusi di bidang ilmu komputer. Deteksi objek melibatkan beberapa elemen utama seperti dataset, algoritma, dan teknik. Dalam deteksi objek, biasanya terdapat dataset yang berisi berbagai objek, dan tujuan dari penelitian ini adalah agar komputer dapat mendeteksi objek tersebut tanpa diberitahu secara eksplisit tentang objeknya.

6) *Convolutional Neural Network (CNN)*

Convolutional Neural Network (CNN) adalah pengembangan dari *Multilayer Perceptron (MLP)* yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis *Deep Neural Network* karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Pada kasus klasifikasi citra, MLP kurang sesuai untuk digunakan karena tidak menyimpan informasi spasial dari data citra dan menganggap setiap piksel adalah fitur yang independen sehingga menghasilkan hasil yang kurang baik. CNN pertama kali dikembangkan dengan nama *Neocognitron* oleh Kunihiko Fukushima, seorang peneliti dari NHK Broadcasting Science Research Laboratories, Kinuta, Setagaya, Tokyo, Jepang. Konsep tersebut kemudian dimatangkan oleh Yann LeChun, seorang peneliti dari AT&T Bell Laboratories di Holmdel, New Jersey, USA. Model CNN dengan nama *LeNet* berhasil diterapkan oleh LeChun pada penelitiannya mengenai pengenalan angka dan tulisan tangan. Pada tahun 2012, Alex Krizhevsky dengan penerapan CNN miliknya berhasil menjuarai kompetisi *ImageNet Large Scale Visual Recognition Challenge 2012*. Prestasi tersebut menjadi momen pembuktian bahwa metode *Deep Learning*, khususnya CNN. Metode

CNN terbukti berhasil mengungguli metode *Machine Learning* lainnya seperti SVM pada kasus klasifikasi objek pada citra.^[13]

7) Metode Lucy Richardson

Algoritma Lucy-Richardson (L-R), yang dikenal juga dengan dekonvolusi Lucy-Richardson, dikembangkan secara independen oleh Richardson (1972) dan Lucy (1974). Algoritma ini efektif jika kita mengetahui PSF tetapi hanya mengetahui sedikit mengenai derau aditif pada citra. Algoritma ini pada mulanya digunakan untuk merestorasi citra astronomi, sebelum akhirnya digunakan juga secara luas untuk merestorasi sembarang citra yang mengalami kekaburan. Algoritma ini memaksimumkan kemungkinan (*maximum likelihood*) bahwa sebuah citra bila dikonvolusi dengan PSF hasilnya adalah instansiasi dari citra kabur, dengan mengasumsikan derau tersebut dengan distribusi *Poisson*.

$$8) p(x) = e^{-a} \frac{a^x}{x!}$$

$$9) x!$$

10) Dengan x adalah peubah acak dan a adalah konstanta. Esensi dari iterasi adalah sebagai berikut: estimasi ke- $(n + 1)$ dari citra restorasi adalah estimasi ke- n citra restorasi dikali dengan citra koreksi. Persamaan iterasinya adalah:

$$11) f^{n+1} = f^n (g$$

$$12) f^n * PSF) * reflect(PSF)$$

13) yang dalam hal ini, operator $*$ menyatakan konvolusi, $\hat{f} = \hat{f}(x, y)$ menyatakan estimasi citra restorasi, $g = g(x, y)$ menyatakan citra masukan (yang mengalami degradasi), $reflect(PSF)$ menyatakan pencerminan PSF, yaitu $Reflect((PSF(x, y)) = PSF(-x, -y)$, dan $(g \hat{f}^n * PSF) * reflect(PSF)$ menyatakan citra koreksi.^[14]

8) You Only Look Once (YOLO)

YOLO adalah metode canggih untuk mendeteksi objek dalam gambar dengan cepat dan efisien. Ketika sebuah gambar dimasukkan, YOLO membaginya menjadi beberapa kotak kecil dan

setiap kotak ini memeriksa apakah ada objek di dalamnya. Jika ada, sistem akan menggambar kotak pembatas di sekitar objek tersebut dan memberikan skor kepercayaan yang menunjukkan seberapa yakin dia bahwa objek itu benar-benar ada. Untuk menghindari kebingungan ketika kotak-kotak tersebut tumpang tindih, YOLO memilih kotak yang paling akurat dan mengabaikan yang lainnya. Selain itu, untuk melatih sistem agar lebih baik, YOLO menggunakan teknik yang disebut *data augmentation*, di mana gambar diubah-ubah agar sistem bisa belajar dari berbagai situasi. Metode ini sangat berguna dalam berbagai aplikasi, seperti pengawasan keamanan, mobil otonom, dan analisis pertanian, karena mampu mendeteksi banyak objek secara bersamaan dengan akurasi tinggi.^[15]

9) You Only Look Once (YOLOv8)

YOLOv8, memiliki beberapa varian (*small, medium, large, dan extra large*) yang disesuaikan dengan kebutuhan komputasi dan akurasi. YOLOv8 (You Only Look Once versi 8) adalah model deteksi objek yang sangat populer dan diperbarui dari versi sebelumnya. Model ini dirancang untuk mendeteksi objek dalam gambar atau video dengan cepat dan akurat, sehingga sangat cocok untuk aplikasi seperti pengawasan video dan mobil otonom. Salah satu fitur utamanya adalah kemampuan untuk melakukan deteksi secara *real-time*. YOLOv8 juga memiliki arsitektur yang ditingkatkan, yang meningkatkan kecepatan dan akurasi deteksi. Selain itu, model ini mampu mendeteksi objek dengan berbagai ukuran melalui prediksi multi-skala dan mendukung transfer learning, memungkinkan pengguna untuk menyempurnakan model dengan dataset mereka sendiri tanpa memerlukan banyak sumber daya.

10) Python

Python adalah bahasa pemrograman yang bersifat *open source*. Bahasa ini dioptimalkan untuk meningkatkan produktivitas pengembang, kualitas perangkat lunak, serta portabilitas program. Python telah banyak digunakan dalam pengembangan berbagai jenis perangkat lunak, termasuk *user interface, numeric programming,*

product customization, system programming, dan scripting internet. Dalam pemrograman *Python*, library terdiri dari beberapa modul dan paket yang digabungkan untuk menyediakan sekumpulan fungsionalitas. Dengan adanya library dalam *Python*, pengembang dapat menganalisis kode yang telah digunakan dan diuji sebelumnya, sehingga memudahkan dalam pengembangan aplikasi.

11) Confusion Matrix

Confusion matrix adalah alat yang sering digunakan untuk menghitung akurasi dalam konteks data mining. Matriks ini menghitung empat keluaran utama, yaitu: *recall*, *precision*, akurasi, dan *error rate*. *Confusion matrix* adalah tabel yang menunjukkan jumlah data uji yang diklasifikasikan dengan benar dan salah. Dalam *confusion matrix*, terdapat empat istilah yang digunakan untuk merepresentasikan hasil proses klasifikasi, antara lain:

1. *True Positive (TP)*: data positif yang diprediksi sebagai positif.
2. *True Negative (TN)*: data negatif yang diprediksi sebagai negatif.
3. *False Positive (FP)*: data negatif yang salah diklasifikasikan sebagai positif.
4. *False Negative (FN)*: data positif yang salah diklasifikasikan sebagai negatif.

		Predicted Class	
		Trust (Positive)	Distrust (Negative)
Actual Class	Trust (Positive)	True Positive	False Negative
	Distrust (Negative)	False Positive	True Negative

Gambar 2. 2 *Confusion Matrix*

Untuk menghitung akurasi dilakukan dengan persamaan

$$\text{Presisi prediksi} = \frac{TP}{TP+FP} \quad (2.1)$$

Presisi merupakan penggambaran seberapa tepat model dalam memprediksi kejadian positif saat kegiatan prediksi. Kemudian untuk menghitung menggunakan persamaan.

$$\text{Akurasi} = \frac{TP+TN}{P+N} \quad (2.2)$$

Akurasi berfungsi sebagai parameter yang akurat pada suatu model dalam melakukan klasifikasi. Recall dapat dihitung menggunakan persamaan.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (2.3)$$

F1 Score merupakan rata-rata dari precision dan recall yang dapat dihitung dengan

$$\text{persamaan } F1 = 2x \frac{\text{precision} \times \text{Recall}}{\text{precision} + \text{Recall}} \quad (2.4)$$

12) Mean Average Precision (mAP)

Mean Average Precision (mAP) adalah metrik evaluasi yang secara luas digunakan dalam bidang visi komputer, terutama untuk tugas pengenalan objek dan pengambilan informasi. Metrik ini mengukur efektivitas model dalam mendeteksi dan mengklasifikasikan objek dalam gambar. Berikut adalah penjelasan lengkap mengenai mAP. **Average Precision (AP)** adalah rata-rata precision pada berbagai level recall. AP memberikan gambaran tentang seberapa baik model dapat mengidentifikasi objek dengan benar pada berbagai ambang batas. **Mean Average Precision (mAP)** adalah rata-rata dari nilai AP untuk semua kelas dalam dataset. mAP memberikan gambaran keseluruhan tentang performa model di semua kategori.[16]

13) Roboflow

Roboflow adalah platform berbasis web yang menyediakan solusi lengkap untuk pengembangan model *computer vision*, mulai dari pengumpulan, pengelolaan, hingga optimasi dataset untuk pelatihan model *machine learning*. Platform ini memudahkan pengembang dalam membuat, mengajar, dan menggunakan model deteksi objek dengan berbagai fitur seperti anotasi (*labeling*) gambar menggunakan *bounding box*, *preprocessing* data, dan augmentasi data untuk meningkatkan kualitas dataset.

Roboflow dan YOLO memiliki hubungan yang erat dalam pengembangan model deteksi objek berbasis computer vision. Roboflow berperan sebagai platform yang menyediakan solusi lengkap untuk pengelolaan dataset, mulai dari anotasi, preprocessing, augmentasi, hingga ekspor data dalam format yang kompatibel dengan YOLO, termasuk YOLOv5, versi terbaru dari arsitektur.

14) Google Colab



Gambar 2. 3 Logo Google Colab

Google Colab atau *Google Colaboratory*, adalah sebuah *executable document* yang dapat digunakan untuk menyimpan, menulis, serta membagikan program yang telah ditulis melalui *Google Drive*. *Software* ini pada dasarnya serupa dengan *Jupyter Notebook* gratis berbentuk *cloud* yang dijalankan menggunakan *browser*, seperti *Mozilla Firefox* dan *Google Chrome*. Penggunaannya untuk menjalankan kode *Python* tanpa perlu melakukan proses instalasi dan *setup* lainnya. Justru, semua keperluan setting dan keperluan setting dan adjustment akan diserahkan ke *cloud*.

III. METODE PENELITIAN

3.1 Waktu dan Lokasi Penelitian

3.1.1 Waktu Penelitian

Waktu penelitian dimulai dari februari 2025 sampai Juli 2025

3.1.2 Lokasi Penelitian

Laboratorium Komputer Teknik Informatika UKI Toraja

3.2 Instrumentasi (Bahan dan Alat Penelitian)

3.2.1 Bahan Penelitian

Dalam penelitian ini bahan yang digunakan yaitu citra buah kopi mentah, matang, dan terlalu matang.

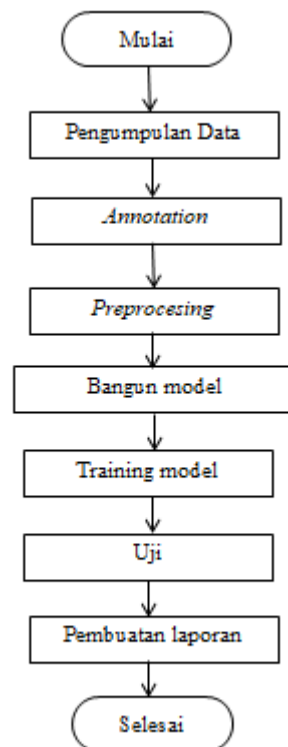
3.2.2 Alat Penelitian

Alat yang digunakan dalam penelitian ini yaitu:

1. Perangkat keras
 - a. Laptop Asus, intel corei5-7200U, up to 3.1Hz, memory 4GB
 - b. Hp Android Vivo 1807 RAM 4GB/32GB
2. Perangkat lunak
 - a. Seperangkat komputer dengan sistem operasi windows 10 64 bit
 - b. *Open Source Computer Vision Library* (*openCV*) dan *software python*.
 - c. *Roboflow*

3.3 Tahapan Penelitian

Flowchart penelitian dapat dilihat pada gambar 3.1



Gambar 3. 1 Tahapan penelitian

4.1 Hasil

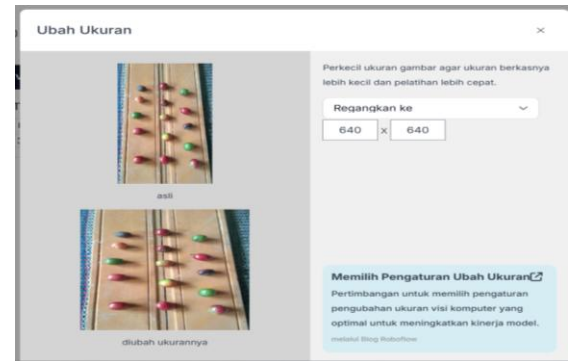
4.1.1 Pengumpulan data

Pengumpulan data yang penulis dilakukan berupa pengumpulan data primer yang dikumpulkan secara langsung oleh peneliti dari sumber asli (wawancara dan observasi). Pengumpulan data ini berupa citra kopi dengan berbagai tingkat kematangan buah kopi. Buah kopi ini di ambil secara langsung di kebun PT. Toarco Jaya yang berada di

bokin. Sebelum pengambilan gambar, buah kopi divalidasi oleh manager kebun (Pak Pakidi) untuk memisahkan buah kopi berdasarkan Tingkat kematangannya. Citra buah kopi di capture menggunakan camera hp vivo 1807 dengan rasio kamere 16:9 dan resolusi gambar 2340*4160.



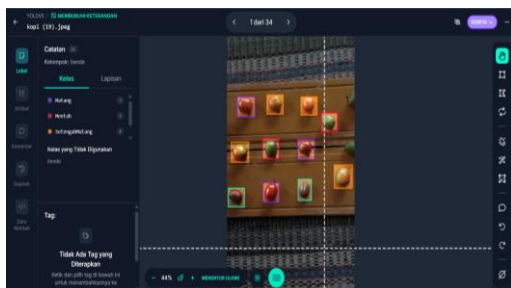
Gambar 4. 1 Sampel Citra



Gambar 4. 3 Gambar Proses *Resize* Citra

4.1.2 Annotation

Setelah penulis mengambil citra buah kopi di kebun, citra buah kopi itu di masukkan ke dalam aplikasi roboflow, sesudah itu tahap selanjutnya adalah proses pelabelan pada citra berdasarkan kelas objek buah kopi yaitu mentah, setengah mentah, matang, terlalu matang. Proses pelabelan dilakukan pada seluruh citra yang akan dijadikan dataset latih dan validasi. Peneliti menggunakan *tools* roboflow untuk melakukan pelabelan dengan teknik *bounding box* seperti pada gambar 4.2



Gambar 4. 2 Proses Pelabelan dengan *Roboflow*

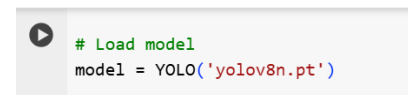
4.1.3 Preprocessing

Pada Proses *Preprocessing* ini, penulis melakukan *resize* pada setiap gambar yang sudah ada. Dan *resize* dilakukan pada aplikasi roboflow yang secara otomatis mentransformasi setiap gambar ke ukuran standar 640x640 piksel melalui algoritma interpolasi pixel. Semua citra buah kopi yang sudah di capture akan diubah ke ukuran tersebut. Hal ini bisa di lihat pada gambar 4.3.

Setelah mengubah *resize* citra buah kopi, selanjutnya citra buah kopi tersebut akan diupload ke google drive dengan nama file dataset. Setelah di upload di gogle drive, data tersebut akan di masukkan ke google colab. Membuka Google drive dan didalam google drive bisa membuka atau menyambungkan google colab dengan goggle dirve, supaya dataset yang ada digoogle dirve akan langsung tersambung atau ada langsung di google colab. Setelah google drive disambungkan dengan google colab, Langkah berikutnya adalah membuat citra buah kopi yang ada di dalam file dataset menjadi blur atau membuat data sintektik agar citra buah kopi akan membuat blur.

4.1.4 Bangun model

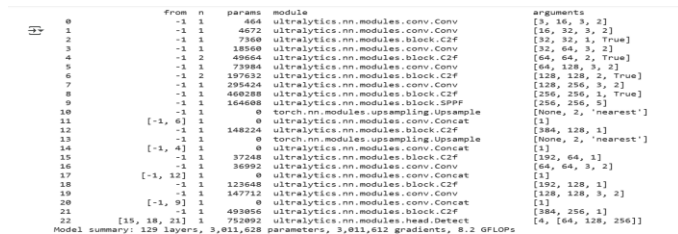
Setelah melakukan *resize* pada semua citra kopi dan sudah di upload di google dirve dan google drive sudah di sambungkan dengan google colab, penulis membangun model, atau memanggil YOLOv8n dengan seperti gambar 4.4 di bawah tersebut.



Gambar 4. 4 Gambar Perintah model YOLOv8

Perintah dalam kode tersebut digunakan untuk memuat model YOLO, yang merupakan alat untuk mendeteksi objek dalam gambar. Di bagian atas, ada komentar yang mengatakan bahwa akan memuat model. Di baris berikutnya, menggunakan perintah `model = YOLO('yolov8n.pt')`, yang artinya

membuka file bernama `yolov8n.pt` dan menyimpannya dalam variabel yang kita sebut model. Jadi, setelah menjalankan perintah ini, kita bisa menggunakan model untuk melakukan deteksi objek pada gambar-gambar yang kita inginkan. Ini seperti mengisi kotak dengan alat yang siap digunakan untuk membantu kita mengenali berbagai objek dalam foto.

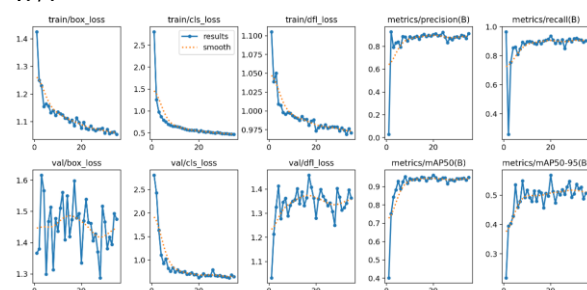


Gambar 4. 5 Gambar Arsitektur YOLOv8

Setelah memanggil model YOLOv8n, hasil dari memanggil `Y=model YOLOn8n` akan seperti pada gambar 4.5. Gambar tersebut menggambarkan arsitektur YOLOv8 dari *Ultralytics*, yang merupakan model untuk mendeteksi objek secara real-time. Secara keseluruhan, YOLOv8 dirancang untuk memberikan keseimbangan optimal antara akurasi dan efisiensi komputasi dalam mendeteksi objek.

4.1.5 Training

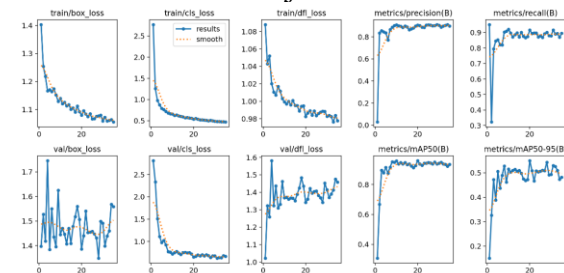
Setelah kita membangun model YOLOv8, diproses *training* ini, penulis melakukan training dua kali. Yang pertama, penulis melakukan training YOLO pada gambar 4.6 dan kedua penulis melakukan training Lucy Richardson dan YOLO.



Gambar 4. 6 Grafik Metrik YOLO

Berdasarkan hasil pelatihan dan validasi model YOLO, diketahui bahwa terjadi penurunan bertahap pada nilai `box_loss`, `cls_loss`, dan `dfl_loss` selama proses *training*, yang menunjukkan bahwa

model mampu belajar representasi fitur dengan baik. Selain itu, nilai metrik evaluasi seperti *precision*, *recall*, dan *mAP* menunjukkan tren peningkatan yang konsisten. Hal ini mengindikasikan bahwa model tidak hanya akurat dalam mendeteksi objek.



Gambar 4. 7 Gambar Grafik YOLO+Lucy Richardson

Berdasarkan grafik hasil pelatihan dan evaluasi, model YOLO menunjukkan performa yang stabil dan konsisten. Penurunan nilai loss baik pada data pelatihan maupun validasi mengindikasikan proses pembelajaran yang efektif. Peningkatan bertahap dan stabil pada metrik *precision*, *recall*, dan *mAP* menunjukkan bahwa model mampu mendeteksi objek secara akurat dan menyeluruh.

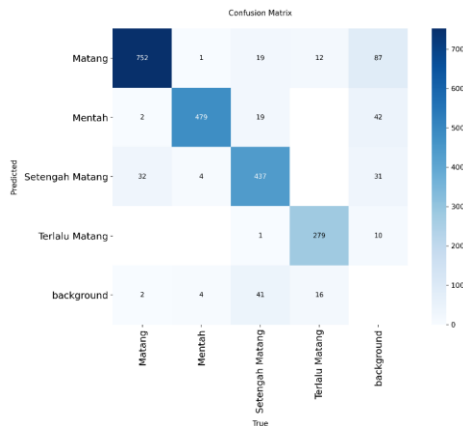
Dari kedua grafik yang ada di atas, jika di bandingkan yang mana lebih akurat dalam mendeteksi objek adalah grafik YOLO+ *Lucy Richardson*. Berdasarkan analisis terhadap grafik pelatihan dan evaluasi, model pada grafik kedua menunjukkan proses pembelajaran yang lebih efisien, ditandai dengan penurunan *nilai loss* yang lebih tajam dan stabil. Selain itu, tren peningkatan nilai *precision*, *recall*, serta *mAP* lebih konsisten dan cepat tercapai dibandingkan grafik pertama. Hal ini menunjukkan bahwa model kedua memiliki tingkat akurasi dan generalisasi yang lebih baik dalam melakukan deteksi objek pada berbagai kondisi dan ambang batas IoU. Dengan demikian, model pada grafik kedua dapat dikategorikan lebih unggul dalam performa deteksi objek secara keseluruhan.

4.1.6 Testing

Setelah *training* dilakukan, penulis akan melakukan *testing* atau pengujian. Di proses testing/pengujian penulis melakukan dua *testing*

yaitu, *testing* YOLO dan *testing Lucy richardson*+YOLO.

Pada Gambar 4.8 adalah hasil pengujian yang dapat di lihat dalam bentuk confusion matrix YOLO.



Gambar 4. 8 Confusion Matrix YOLO

Berdasarkan *confusion matrix* di atas, model klasifikasi kematangan buah kopi dievaluasi terhadap lima kelas utama. Prediksi yang benar ditunjukkan oleh nilai diagonal pada masing-masing baris, seperti kelas Matang sebanyak **752 kasus**, kelas Mentah sebanyak **479 kasus**, dan kelas Setengah Matang sebanyak **437 kasus**, yang menunjukkan bahwa model memiliki kemampuan klasifikasi yang cukup baik pada sebagian besar kategori.

Kesalahan klasifikasi terjadi ketika prediksi model tidak sesuai dengan label sebenarnya. Misalnya, pada kelas Setengah Matang, sebanyak **32 sampel** diklasifikasikan sebagai Matang, sementara **74 sampel** latar belakang (background) juga diklasifikasikan sebagai Setengah Matang. Hal ini menandakan bahwa terdapat kemiripan fitur visual antar kelas buah kopi yang berdekatan secara tingkat kematangan, serta tantangan dalam membedakan objek buah dari latar belakang.

Selain itu, sebanyak **87 kasus** latar belakang diklasifikasikan sebagai buah Matang, dan **42 kasus** latar diklasifikasikan sebagai Mentah, yang menunjukkan bahwa model masih cenderung mendeteksi latar sebagai objek. Namun, untuk kelas Terlalu Matang, model berhasil melakukan klasifikasi dengan cukup baik dengan **279 kasus benar** dari total **326 prediksi**, serta kesalahan klasifikasi relatif rendah.

Dengan total jumlah prediksi sebanyak **4413 sampel**, confusion matrix ini memberikan gambaran kuantitatif mengenai performa model, baik dari sisi ketepatan maupun kesalahan klasifikasi. Informasi ini selanjutnya dapat digunakan untuk menghitung metrik evaluasi seperti precision, recall, dan F1-score per kelas serta melakukan analisis kekeliruan untuk meningkatkan kualitas model.

Perhitungan Nilai Akurasi

Untuk menghitung akurasi, kita memerlukan nilai dari *confusion matrix*. Akurasi adalah rasio jumlah prediksi yang benar (yaitu, *True Positive* dan *True Negative*) terhadap total jumlah prediksi yang dibuat.

$$\text{Akurasi} = \frac{\text{Jumlah Prediksi yang benar}}{\text{Jumlah data}}$$

Berikut cara menghitung nilai akurasi *confusion matrix* YOLO

$$\begin{aligned} \text{Akurasi} &= \frac{752 + 479 + 437 + 279}{752 + 1 + 19 + 12 + 87 + 2 + 479 + 19 + 42 + 32 + 4 + 437 + 31 + 1 + 279 + 10 + 2 + 4 + 41 + 16} \end{aligned}$$

$$\text{Akurasi} = \frac{1947}{2270} = 0,8577$$

Jadi, nilai akurasi dari *confusion matrix* YOLO Adalah 85,7%.

Pada gambar 4.9 adalah hasil pengujian yang dapat di lihat dalam bentuk confusion matrix Lucy Richardson + YOLO.



Gambar 4. 9 Confusion Matrix Lucy Richardson + YOLO

Berdasarkan confusion matrix yang ditampilkan, model klasifikasi kematangan buah kopi dievaluasi terhadap lima kelas utama, yaitu Matang, Mentah, Setengah Matang, Terlalu Matang, dan Background. Nilai diagonal pada masing-masing baris menunjukkan jumlah prediksi yang benar, antara lain sebanyak **730 kasus** untuk kelas Matang, **456 kasus** untuk kelas Mentah, dan **479 kasus** untuk kelas Setengah Matang. Hasil ini mengindikasikan bahwa model memiliki kemampuan klasifikasi yang cukup baik pada sebagian besar kategori buah kopi.

Kesalahan klasifikasi terjadi ketika prediksi model tidak sesuai dengan label sebenarnya. Sebagai contoh, sebanyak **38 sampel Setengah Matang** diklasifikasikan sebagai Matang, dan **51 sampel background** juga diklasifikasikan sebagai Setengah Matang. Hal tersebut menunjukkan adanya kemiripan fitur visual antara kategori buah kopi yang berdekatan secara tingkat kematangan, serta tantangan dalam membedakan objek buah dari latar belakang.

Selain itu, sebanyak **98 kasus latar belakang** diklasifikasikan sebagai buah Matang, dan **39 kasus latar** diklasifikasikan sebagai Mentah, yang menunjukkan bahwa model masih memiliki kecenderungan untuk mendeteksi latar sebagai objek. Namun demikian, kelas Terlalu Matang menunjukkan performa deteksi yang cukup optimal, dengan **267 prediksi benar** dari total **281 prediksi**, serta kesalahan klasifikasi yang relatif rendah.

Dengan total jumlah prediksi sebanyak **2302 sampel**, confusion matrix ini memberikan gambaran kuantitatif mengenai performa model, baik dari sisi ketepatan klasifikasi maupun distribusi kesalahan antar kelas. Matriks ini menjadi dasar penting untuk menghitung metrik evaluasi seperti precision, recall, dan F1-score per kelas, serta mendukung analisis kekeliruan guna merancang strategi perbaikan model yang lebih efektif.

Perhitungan Nilai Akurasi

Untuk menghitung akurasi, kita memerlukan nilai dari *confusion matrix*. Akurasi adalah rasio jumlah prediksi yang benar (yaitu, *True Positive*

dan *True Negative*) terhadap total jumlah prediksi yang dibuat.

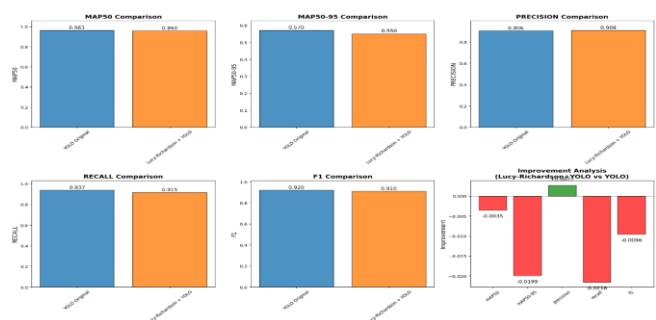
$$\text{Akurasi} = \frac{\text{Jumlah Prediksi yang benar}}{\text{Jumlah data}}$$

Berikut cara menghitung nilai akurasi *confusion matrix Lucy Richardson + YOLO*

$$\begin{aligned} \text{Akurasi} &= \frac{734 + 456 + 480 + 267}{734 + 1 + 18 + 22 + 87 + 2 + 456 + 4 + 40 \\ &\quad + 34 + 24 + 480 + 52 + 1 + 267 + 13 \\ &\quad + 18 + 6 + 15 + 18} \end{aligned}$$

$$\text{Akurasi} = \frac{1937}{2292} = 0,8451$$

Jadi, nilai akurasi dari *confusion matrix Lucy Richardson + YOLO* Adalah 84,5%.



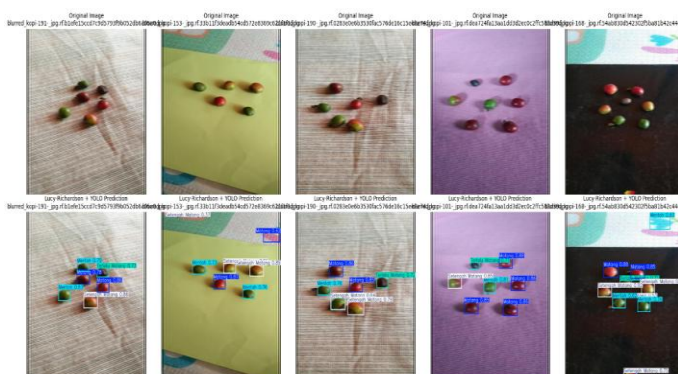
Gambar 4. 10 visualisasi perbandingan metrik evaluasi model YOLO dan Lucy-Richardson + YOLO

Berdasarkan visualisasi perbandingan metrik evaluasi, model kombinasi Lucy-Richardson dan YOLOv8 menunjukkan peningkatan performa yang signifikan pada metrik mAP@0.5 dan mAP@0.5:0.95, masing-masing sebesar +0.519 dan +0.269. Meskipun terdapat sedikit penurunan pada precision (-0.024), recall (-0.002), dan F1-score (-0.012), seluruh nilai tetap berada dalam kisaran tinggi dan tidak menurunkan stabilitas klasifikasi model secara keseluruhan. Hasil ini mengindikasikan bahwa teknik augmentasi blur berhasil memperkuat kemampuan generalisasi model dalam mengenali objek kompleks pada citra digital, menjadikannya layak diterapkan dalam sistem pertanian berbasis visi komputer.



Gambar 4. 11 Deteksi dan klasifikasi objek buah kopi menggunakan model berbasis YOLOv8

Gambar di atas menampilkan hasil deteksi objek terhadap buah kopi dengan latar yang bervariasi. Pada baris pertama diperlihatkan citra asli tanpa anotasi, sedangkan baris kedua memperlihatkan hasil klasifikasi oleh model YOLO berupa *bouding box* dan label kelas pada masing-masing buah. Model mampu mengenali objek dengan ketepatan tinggi, meskipun terdapat variasi warna latar dan pencahayaan. Hasil ini menunjukkan bahwa model memiliki kemampuan generalisasi yang baik dan robust terhadap variasi kondisi visual, yang menjadi indikator bahwa sistem deteksi dapat diterapkan dalam lingkungan nyata secara efektif.



Gambar 4. 12 deteksi dan klasifikasi objek buah kopi menggunakan model Lucy Richardson dan YOLOv8

Gambar di atas memperlihatkan hasil klasifikasi objek buah kopi melalui model deteksi YOLOv8 dengan latar yang bervariasi. Baris pertama menampilkan citra asli tanpa anotasi, sedangkan baris kedua menunjukkan hasil prediksi

berupa bounding box dan label kelas. Model mampu mengidentifikasi buah kopi sebagai Healthy, Burnt, dan Rotten dengan akurasi tinggi, meskipun terdapat perbedaan latar dan pencahayaan. Visualisasi ini memperkuat analisis bahwa model memiliki kemampuan generalisasi yang baik terhadap objek buah kopi dalam berbagai kondisi tampilan citra.

4.2 PEMBAHASAN

Penelitian ini berhasil mengimplementasikan dua pendekatan dalam deteksi dan klasifikasi kematangan buah kopi, yaitu model YOLOv8 standar dan kombinasi teknik *Lucy-Richardson* dengan YOLOv8. Hasil evaluasi menunjukkan bahwa kedua model memiliki karakteristik performa yang berbeda dan memberikan kontribusi penting dalam pengembangan sistem otomatis untuk industri kopi.

Teknik *Lucy-Richardson* bekerja dengan mengurangi blur pada citra, sehingga detail visual seperti tekstur kulit buah, gradasi warna, dan transisi antar area menjadi lebih tajam. Dalam konteks klasifikasi kematangan buah kopi, peningkatan kualitas visual ini sangat krusial karena perbedaan antar kelas kematangan seringkali terletak pada detail halus seperti perubahan warna dari hijau ke kuning atau merah. Hal ini mengindikasikan bahwa teknik dekonvolusi *Lucy-Richardson* berhasil meningkatkan kualitas fitur visual pada citra buah kopi, sehingga mempermudah proses pembelajaran model dan menghasilkan representasi yang lebih baik untuk karakteristik visual yang kompleks.

Evaluasi melalui confusion matrix menunjukkan perbedaan performa yang menarik antara kedua model. Pada model YOLOv8 standar (Image 1), terlihat bahwa model berhasil mengklasifikasikan 752 kasus buah matang, 479 kasus buah mentah, 437 kasus buah setengah matang, dan 279 kasus buah terlalu matang dengan benar. Sementara itu, model kombinasi *Lucy-Richardson* + YOLOv8 (Image 2) menunjukkan performa yang berbeda dengan 734 kasus buah matang, 456 kasus buah mentah, 480 kasus buah setengah matang, dan 267 kasus buah terlalu matang yang terklasifikasi dengan benar.

Perbedaan yang paling signifikan terletak pada penanganan kelas "Setengah Matang" dimana model kombinasi menunjukkan peningkatan akurasi dari 437 menjadi 480 prediksi yang benar, menunjukkan bahwa teknik Lucy-Richardson khususnya efektif dalam menangani kelas yang memiliki karakteristik visual transisional.

Analisis akurasi keseluruhan dari kedua confusion matrix menunjukkan perbedaan performa antara model YOLOv8 standar dan model kombinasi Lucy-Richardson + YOLOv8. Model YOLOv8 standar (confusion matrix pertama) mencapai akurasi keseluruhan sebesar 85,7%, sementara model kombinasi Lucy-Richardson + YOLOv8 (confusion matrix kedua) menunjukkan akurasi keseluruhan sebesar 84,5%. Perbedaan akurasi sebesar 1,2% ini menunjukkan bahwa model YOLOv8 standar memiliki performa klasifikasi yang sedikit lebih baik secara keseluruhan dibandingkan model kombinasi.

Meskipun model YOLOv8 standar menunjukkan akurasi keseluruhan yang lebih tinggi, analisis lebih mendalam pada confusion matrix mengungkapkan bahwa model kombinasi Lucy-Richardson + YOLOv8 memiliki keunggulan pada aspek-aspek tertentu. Penurunan akurasi sebesar 1,2% pada model kombinasi dapat dikaitkan dengan trade-off yang terjadi selama proses preprocessing Lucy-Richardson, dimana peningkatan ketajaman detail visual menghasilkan sensitivitas yang lebih tinggi terhadap noise dan variasi minor dalam citra. Namun demikian, model kombinasi menunjukkan peningkatan signifikan pada kelas "Setengah Matang" dari 437 menjadi 480 prediksi benar, mengindikasikan kemampuan yang lebih baik dalam menangani kelas dengan karakteristik visual transisional yang kompleks. Hal ini menunjukkan bahwa meskipun akurasi keseluruhan sedikit menurun, model kombinasi memiliki kemampuan yang lebih spesifik dalam mengidentifikasi tingkat kematangan yang berada pada fase transisi, yang merupakan aspek kritis dalam aplikasi sorting buah kopi industri.

Hasil visualisasi deteksi menunjukkan kemampuan generalisasi yang baik dari kedua model terhadap variasi kondisi visual, dimana

model mampu menangani perbedaan pencahayaan, orientasi buah, dan kompleksitas latar belakang dengan performa yang konsisten. Kemampuan generalisasi yang baik ini sangat penting untuk implementasi dalam lingkungan industri dimana kondisi pencahayaan, sudut pandang kamera, dan variasi latar belakang tidak dapat dikontrol secara sempurna. Model yang robust terhadap variasi ini akan mengurangi kebutuhan fine-tuning yang berulang dan meningkatkan reliability sistem dalam jangka panjang, sehingga lebih praktis untuk diimplementasikan dalam skala industri.

Hasil penelitian ini memiliki implikasi praktis yang signifikan untuk industri kopi, dimana model kombinasi Lucy-Richardson + YOLOv8 dengan peningkatan mAP yang substansial menunjukkan potensi implementasi yang lebih baik dalam sistem sorting otomatis. Peningkatan akurasi deteksi dapat berkontribusi pada peningkatan kualitas produk akhir dan efisiensi proses produksi, namun untuk implementasi praktis disarankan untuk menggunakan model kombinasi dengan mempertimbangkan computational cost tambahan dari preprocessing Lucy-Richardson. Meskipun menunjukkan hasil yang promising, penelitian ini memiliki keterbatasan pada dataset yang terbatas pada satu lokasi perkebunan, sehingga penelitian lanjutan disarankan untuk melakukan validasi silang pada multiple locations dan pengembangan algoritma yang dapat menangani real-time processing dengan akurasi tinggi.

IV. KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa algoritma YOLOv8 efektif dalam mendeteksi tingkat kematangan buah kopi dengan akurasi yang baik. Pelatihan model dilakukan menggunakan citra berukuran 640x640 piksel, batch size 32, dan 50 epoch, dengan data pelatihan sebanyak 814 citra serta evaluasi melalui confusion matrix. Hasil evaluasi menunjukkan bahwa YOLO dengan dataset blur sebesar 85,7%, sedangkan kombinasi Lucy Richardson dengan

YOLO menghasilkan akurasi sebesar 84,5%. Meskipun terdapat penurunan akurasi sebesar 1,2%, model kombinasi menunjukkan peningkatan signifikan pada metrik mAP@0.5 (+0.519) dan mAP@0.5:0.95 (+0.269), mengindikasikan kemampuan deteksi dan lokalisasi objek yang lebih superior.

Model kombinasi Lucy-Richardson + YOLOv8 memiliki keunggulan dalam menangani kelas "Setengah Matang" dengan peningkatan prediksi benar dari 437 menjadi 480 kasus, yang sangat penting untuk identifikasi buah dengan tingkat kematangan transisional. Implementasi algoritma Lucy-Richardson sebagai preprocessing berhasil meningkatkan kualitas fitur visual dan konvergensi training yang lebih cepat.

Kedua model menunjukkan potensi baik untuk sistem klasifikasi otomatis. Model YOLOv8 standar cocok untuk aplikasi yang mengutamakan akurasi klasifikasi maksimal, sedangkan model kombinasi lebih sesuai untuk aplikasi yang memerlukan akurasi deteksi dan lokalisasi tinggi dengan kemampuan khusus menangani kasus kompleks kematangan buah kopi.

5.2 SARAN

Sebagai tindak lanjut dari hasil penelitian ini, disarankan agar proses klasifikasi kematangan buah kopi dapat terus disempurnakan melalui peningkatan pada sistem deteksi latar belakang. Meskipun model menunjukkan kinerja yang baik dalam mengenali objek utama, masih ditemukan kecenderungan untuk mengklasifikasikan latar sebagai buah kopi, sehingga perlu dilakukan pengembangan teknik pemisahan fitur, seperti penggunaan masking atau segmentasi latar untuk meminimalkan kesalahan klasifikasi. Selain itu, disarankan agar dataset ditingkatkan dengan menambahkan citra dari berbagai kondisi pencahayaan dan sudut pengambilan gambar agar model memiliki kemampuan generalisasi yang lebih kuat dalam situasi nyata. Penggunaan teknik augmentasi tambahan, seperti pengaturan kontras, rotasi acak, atau cropping, juga layak dipertimbangkan untuk meningkatkan keragaman fitur dalam data latih. Uji coba sistem dalam lingkungan *real-time*, misalnya melalui kamera portabel di kebun, sangat dianjurkan agar performa model dapat dinilai secara langsung dalam kondisi

operasional lapangan. Lebih lanjut, hasil klasifikasi model juga dapat diintegrasikan ke dalam aplikasi pendukung keputusan berbasis digital guna membantu petani atau pengelola dalam menentukan waktu panen secara lebih akurat dan efisien.

REFERENSI

- [1] Widaningsih, Ir. Roch. *Outlook Komoditas Perkebunan Kopi*. Jakarta: Pusat Data dan Sistem Informasi Pertanian Sekretariat Jenderal - Kementerian Pertanian 2022, 2022.
- [2] Sirappa, Marthen P, dan Religius Heryanto. "Standardisasi Pengolahan Biji Kopi Berkualitas," t.t.
- [3] Sidehabi, Sitti Wetenriajeng, Wahidah, Nurhayati Jabir, dan Moh. Ilyas. "IbM Sosialisasi Alat Penyortir Biji Kopi otomatis dengan menggunakan Mikrokotroller Arduino Nano pada Usaha Mikro Kecil dan Menengah (UMKM) Toraja Kawaa Roasters." *IbMAS ATIM: Jurnal Pengabdian Masyarakat* 1, no. 1 (24 Juni 2022): 37–43. <https://doi.org/10.61844/ibmasatim.v1i1.165>.
- [4] Maulani, Riska Dwi, dan Diah Wahyuningsih. "Analisis Ekspor Kopi Indonesia pada Pasar Internasional." *Pamator Journal* 14, no. 1 (23 Mei 2021): 27–33. <https://doi.org/10.21107/pamator.v14i1.8692>.
- [5] Maulani, Riska Dwi, dan Diah Wahyuningsih. "Analisis Ekspor Kopi Indonesia pada Pasar Internasional." *Pamator Journal* 14, no. 1 (23 Mei 2021): 27–33. <https://doi.org/10.21107/pamator.v14i1.8692>.
- [6] Dalmar Dakari Aboyomi dan Cleo Daniel, "A Comparative Analysis of Modern Object Detection Algorithms: YOLO vs. SSD vs. Faster R-CNN," *ITEJ (Information Technology Engineering Journals)* 8, no. 2 (28 Desember 2023): 96–106, <https://doi.org/10.24235/itej.v8i2.123>.
- [7] Fauzi, Dicky Rachmat, Anggunmeka Luhur Prasasti, dan Ashri Dinimaharawati. "IMPLEMENTASI NOISE REMOVAL DAN IMAGE RESTORATION PADA CITRA KABUR (BLUR) DENGAN MENGGUNAKAN METODE LUCY –

- RICHARDSON
TECHNIQUES,” t.t.
- [8] Triyanto, Dedi, Muhammad Zidan, Mochamad Wahyudi, Lise Pujiastuti, dan Sumanto Sumanto. “Pengembangan Sistem Deteksi Objek Botol Real-Time dengan YOLOv8 untuk Aplikasi Vision.” *Indonesian Journal Computer Science* 3, no. 1 (30 April 2024): 44–50. <https://doi.org/10.31294/ijcs.v3i1.6070>.
- [9] Yanto, Yanto, Faruq Aziz, dan Irmawati Irmawati. “Yolo-V8 Peningkatan Algoritma Untuk Deteksi Pemakaian Masker Wajah.” *Jati (Jurnal Mahasiswa Teknik Informatika)* 7, no. 3 (9 Oktober 2023): 1437–44. <https://doi.org/10.36040/jati.v7i3.7047>.
- [10] Satya, Lejar, M Ridwan Dwi Septian, Mochammad Wisuda Sarjono, Margi Cahyanti, dan Ericks Rachmat Swedia. “Sistem Pendeteksi Plat Nomor Polisi Kendaraan Dengan Arsitektur Yolov8.” *Sebatik* 27, no. 2 (15 Desember 2023): 753–61. <https://doi.org/10.46984/sebatik.v27i2.2374>.
- [11] Hayati, Nurhaliza Juliyani, Dayan Singasatia, dan Muhamad Rafi Muttaqin. “Object Tracking Menggunakan Algoritma You Only Look Once (YOLO)v8 untuk Menghitung Kendaraan.” *Komputa : Jurnal Ilmiah Komputer dan Informatika* 12, no. 2 (4 November 2023): 91–99. <https://doi.org/10.34010/komputa.v12i2.10654>.
- [12] Maulana, Imam, Nining Rahaningsih, dan Tati Suprpti. “Analisis Penggunaan Model YOLOv8 (You Only Look Once) Terhadap Deteksi Citra Senjata Berbahaya” 7, no. 6 (2023).
- [13] Fitriati, Andi, Yoan Elviralita, Nanang Roni Wibowo, Amal Mulia, dan Sri Wulandari. “Sistem Sortir Benda Berdasarkan Bentuk Dan Warna Berbasis Komputer Vision.” *Transient: Jurnal Ilmiah Teknik Elektro* 13, no. 2 (26 Juni 2024): 46–51. <https://doi.org/10.14710/transient.v13i2.46-51>.
- [14] Aulia, Fitrahuda. “Computer Vision dan Pengolahan Citra Digital,” t.t.
- [15] Arshad, Usama. “Object Detection in Last Decade - A Survey.” *Scientific Journal of Informatics* 8, no. 1 (10 Mei 2021): 60–70.
- Manalu, Sriyanti Boang. “Image Restoration Menggunakan Metode Lucy-Richardson Pada Citra Underwater” 1, no. 3 (2021).
- [17] Wang, Xueqiu, Huanbing Gao, Zemeng Jia, dan Zijian Li. “BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8.” *Sensors* 23, no. 20 (10 Oktober 2023): 8361. <https://doi.org/10.3390/s23208361>.
- [18] Eka Putra, Wayan Suartika. “Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101.” *Jurnal Teknik ITS* 5, no. 1 (18 Maret 2016). <https://doi.org/10.12962/j23373539.v5i1.15696>.
- [19] Alzubaidi, Laith, Jinglan Zhang, Amjad J. Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, J. Santamaria, Mohammed A. Fadhel, Muthana Al-Amidie, dan Laith Farhan. “Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions.” *Journal of Big Data* 8, no. 1 (31 Maret 2021): 53. <https://doi.org/10.1186/s40537-021-00444-8>.
- [20] Aboyomi, Dalmar Dakari, dan Cleo Daniel. “A Comparative Analysis of Modern Object Detection Algorithms: YOLO vs. SSD vs. Faster R-CNN.” *ITEJ (Information Technology Engineering Journals)* 8, no. 2 (28 Desember 2023): 96–106. <https://doi.org/10.24235/itej.v8i2.123>.