OPTIMASI MODEL KLASIFIKASI TINGKAT KEMATANGAN BUAH MANGGA DENGAN METODE YOLO11

Ferdi Gusnanto, Nining Rahaningsih, Raditya Danar Dana, Mulyawan
Teknik Informatika, STMIK IKMI Cirebon
Jl. Perjuangan No.10B, Karyamulya, Kec. Kesambi, Kota Cirebon, Jawa Barat 45135
fergoesnant72@gmail.com

ABSTRAK

Klasifikasi tingkat kematangan buah mangga secara akurat penting untuk mendukung efisiensi sektor pertanian, terutama dalam menentukan waktu panen optimal. Namun, variasi pencahayaan dan latar belakang pada citra sering memengaruhi akurasi model. Penelitian ini bertujuan meningkatkan akurasi klasifikasi kematangan mangga dengan menggunakan YOLO11, algoritma deteksi objek terbaru, untuk membedakan tingkat kematangan mangga dalam empat kategori: *unripe*, *early ripe*, *partially ripe*, dan *ripe*. Dataset diperoleh dari Kaggle, melibatkan proses *preprocessing* seperti *resizing* dan pembagian data, serta pelatihan model dengan penyesuaian parameter seperti *learning rate* dan *batch size*. Model YOLO11 berhasil mencapai akurasi 97,3%, *precision* 97,5%, *recall* 97,3%, dan *F1-score* 97,3%, menunjukkan performa tinggi dalam mengklasifikasikan kematangan mangga. Namun, pengujian dengan *webcam* mengungkapkan tantangan dalam kondisi pencahayaan bervariasi, terutama untuk kelas kematangan yang kompleks. Hal ini mengindikasikan kebutuhan optimasi tambahan seperti augmentasi data. Penelitian ini membuktikan efektivitas YOLO11 untuk klasifikasi kematangan mangga, sekaligus menunjukkan potensi pengembangan lebih lanjut. Temuan ini diharapkan dapat berkontribusi pada teknologi pertanian cerdas dan membuka peluang penelitian dalam penerapan deteksi *real-time* berbasis citra di bidang agrikultur.

Kata kunci : *YOLO11, klasifikasi citra, kematangan mangga, deep learning*

1. PENDAHULUAN

Perkembangan pesat di bidang informatika telah memberikan kontribusi yang signifikan terhadap berbagai aspek kehidupan, termasuk teknologi, bisnis, dan pendidikan. Salah satu terobosan di bidang ini adalah penerapan kecerdasan buatan (*Artificial Intelligence*/AI) dan visi komputer (*Computer Vision*), yang memungkinkan sistem untuk melakukan tugas-tugas analisis secara lebih cepat dan akurat. Salah satu penerapannya yang relevan adalah dalam identifikasi objek melalui citra digital menggunakan algoritma deep learning seperti YOLO (*You Only Look Once*). Dalam beberapa tahun terakhir, penelitian terkait penggunaan YOLO untuk berbagai aplikasi, seperti deteksi objek, klasifikasi, dan analisis citra, telah menunjukkan potensi yang signifikan dalam mengatasi berbagai permasalahan di dunia nyata [1]. Dalam konteks pertanian, klasifikasi tingkat kematangan buah mangga menggunakan teknologi ini dapat membantu petani dalam menentukan waktu panen yang optimal, sehingga meningkatkan efisiensi dan kualitas hasil panen.

Meskipun telah banyak penelitian yang menggunakan algoritma YOLO untuk deteksi objek, tantangan tetap ada dalam meningkatkan akurasi klasifikasi tingkat kematangan buah mangga, terutama dalam kondisi pencahayaan dan latar belakang yang bervariasi [2]. Penelitian sebelumnya menunjukkan bahwa kualitas dataset serta kompleksitas fitur objek, seperti warna dan tekstur pada mangga, berpengaruh besar terhadap tingkat akurasi algoritma YOLO [3]. Kualitas dataset juga menjadi perhatian pada penelitian menggunakan YOLO untuk objek lain, seperti plat nomor kendaraan dan daun tanaman, yang menghasilkan variasi akurasi karena faktor pencahayaan dan kondisi objek [1], [4]. Kendala ini semakin signifikan dalam konteks buah tropis seperti mangga yang memiliki karakteristik kematangan yang kompleks dan bervariasi [5]. Dalam penelitian ini, klasifikasi kematangan mangga berdasarkan warna menjadi tantangan karena perubahan warna dapat dipengaruhi oleh kondisi lingkungan yang dinamis, seperti pencahayaan yang tidak merata [6]. Hal ini menunjukkan bahwa meski YOLO merupakan metode yang efektif untuk deteksi objek, kemampuan klasifikasi akurasinya masih memerlukan peningkatan ketika diaplikasikan pada buah-buahan dengan variasi yang besar dalam warna dan tekstur [7]. Selain itu, penelitian terkait deteksi objek makanan khas menunjukkan bahwa algoritma YOLO rentan terhadap perubahan fitur visual objek dalam skenario yang tidak terkontrol [8]. Dengan demikian, diperlukan penelitian lebih lanjut yang mengatasi tantangan-tantangan ini, khususnya melalui optimasi parameter atau peningkatan kualitas dataset, guna menghasilkan model klasifikasi yang lebih akurat. Langkah-langkah perbaikan ini diharapkan dapat meningkatkan ketahanan model terhadap kondisi pencahayaan dan latar belakang yang bervariasi, sehingga hasil klasifikasi menjadi lebih stabil [9]. Maka, penelitian ini bertujuan untuk mengisi kesenjangan tersebut dengan fokus pada penerapan YOLO untuk klasifikasi kematangan buah mangga dengan meningkatkan ketahanan terhadap variasi visual objek mangga [10].

Penelitian-penelitian terdahulu telah berkontribusi dalam pengembangan model deteksi dan klasifikasi menggunakan YOLO. Misalnya, penelitian oleh Mulyana menunjukkan penggunaan YOLOv5

dalam klasifikasi kendaraan secara real-time, yang berhasil mencapai akurasi tinggi dengan optimasi pada parameter model [10]. Penelitian lain oleh Syafi'I mengaplikasikan metode transformasi ruang warna HSI untuk mengklasifikasikan kematangan buah mangga Garifta Merah, meskipun pendekatan tersebut kurang efektif pada kondisi pencahayaan rendah [7]. Selain itu, penelitian oleh Ayu menggunakan model *Convolutional Neural Network* (CNN) untuk mendiagnosa daun mangga, yang menunjukkan hasil yang menjanjikan namun terbatas pada jenis daun tertentu [5]. Berdasarkan kajian literatur ini, masih terdapat peluang untuk mengembangkan model klasifikasi kematangan buah mangga yang lebih akurat dengan memanfaatkan YOLO versi terbaru, yaitu YOLO11. Model ini diharapkan mampu menangkap perbedaan variasi warna pada buah mangga dengan lebih efektif, bahkan dalam kondisi pencahayaan yang tidak ideal. Kontribusi utama dari penelitian ini adalah penerapan YOLO11 yang dioptimalkan untuk deteksi tingkat kematangan buah mangga, yang sebelumnya belum banyak dibahas dalam konteks klasifikasi buah. Model ini diharapkan dapat mengatasi tantangan yang ada, memberikan hasil deteksi yang konsisten dan akurat, serta menjadi landasan untuk implementasi di industri pertanian dalam rangka meningkatkan kualitas dan efisiensi dalam pemilihan buah mangga berdasarkan tingkat kematangannya.

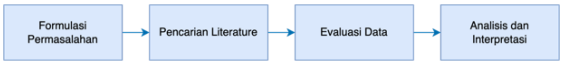
Penelitian ini bertujuan untuk meningkatkan akurasi klasifikasi tingkat kematangan buah mangga dengan menggunakan metode YOLO11. Fokus utama penelitian ini adalah untuk mengembangkan model yang mampu mengatasi variasi pencahayaan dan latar belakang dalam citra mangga sehingga menghasilkan prediksi yang lebih akurat. Kontribusi dari penelitian ini adalah memperkenalkan pendekatan yang lebih efektif dalam klasifikasi kematangan buah mangga, yang dapat menjadi referensi bagi penelitian selanjutnya di bidang *Computer Vision.* Secara praktis, penelitian ini diharapkan dapat membantu petani dalam menentukan waktu panen yang tepat, meningkatkan efisiensi produksi, dan mengurangi potensi kerugian akibat panen yang tidak optimal.

Pendekatan yang digunakan dalam penelitian ini melibatkan pengumpulan dataset citra buah mangga dengan berbagai tingkat kematangan, yang kemudian akan dilabeli sebagai *unripe, early ripe, partially ripe,* dan *ripe*. Model YOLO11 akan dilatih menggunakan dataset tersebut dengan optimasi pada parameter-parameter penting seperti *learning rate* dan *batch size*. Metode ini dipilih karena kemampuan YOLO yang unggul dalam deteksi objek secara real-time [2]. Proses validasi model akan dilakukan secara langsung dengan kondisi pencahayaan dan latar belakang tertentu.

Jika tujuan penelitian ini tercapai, maka hasilnya diharapkan dapat memberikan kontribusi signifikan dalam pengembangan teknologi klasifikasi citra menggunakan YOLO. Penelitian ini juga dapat menjadi acuan bagi pengembangan aplikasi otomatisasi dalam industri pertanian, seperti sistem monitoring kematangan buah yang terintegrasi dengan perangkat IoT. Selain itu, model yang dihasilkan dapat digunakan oleh peneliti lain dalam pengembangan sistem klasifikasi serupa untuk buah-buahan lain, sehingga memperluas cakupan penerapan metode YOLO di bidang lain. Hasil penelitian ini berpotensi memberikan dampak positif terhadap efisiensi dan kualitas hasil panen di sektor pertanian.

2. TINJAUAN PUSTAKA

Metode *literature review* yang digunakan dalam penelitian ini adalah dengan cara me-*review* pada artikel-artikel yang terkait dengan Peningkatan Model Klasifikasi Tingkat Kematangan Buah Mangga Menggunakan Metode YOLO11. Langkah-langkah dari *literature review* meliputi 4 tahapan , yaitu (1) formulasi permasalahan, (2) pencarian literature, (3) evaluasi data, (4) analisis dan interpretasi [11].



Gambar 1. Langkah-langkah literatur review

Formulasi permasalahan dilakukan memilih topik. Pada penelitian ini topik yang dipilih adalah mengenai kecerdasan buatan (*Artificial Intelligence/AI*) dan visi komputer (*Computer Vision*). Langkah selanjutnya adalah pencarian literatur yang relevan dengan topik penelitian. Langkah ini dapat memberikan gambaran mengenai klasifikasi tingkat kematangan buah manga menggunakan metode YOLO11. Proses pencarian dan pengumpulan artikel dengan menggunakan *tools publish or perish*, yang diambil dari database akademik yaitu *Google Scholar*, *Shinta* dan *Scopus* dengan kata kunci "klasifikasi citra", "mangga", dan "metode YOLO". Dari hasil pencarian ditemukan sebanyak 600 artikel dan jurnal. Langkah ketiga adalah evaluasi data yaitu dengan melakukan penyaringan dan pemilihan serta pemilahan artikel jurnal yang benar benar relevan dengan topik penelitian serta mempertimbangkan keterbaruannya. Keterbaruan dibatasi dengan memilih artikel 5 tahun terakhir yaitu mulai dari tahun 2024, 2023, 2022, 2021, dan 2020. Sedangkan kesesuaian dilihat dari kesesuain judul *paper*, sinta *index* , atau hal lain yang dianggap memiliki reputasi. Berdasarkan hasil evaluasi data, kemudian dipilih 15 artikel jurnal untuk di-*review*.

Setelah keempat tahapan tersebut dilakukan, proses selanjutnya adalah pelaksanaan *literature review*. Adapun teknik yang dilakukan dalam *literature review* yaitu mencai kesamaan (*compare*), mencari ketidaksamaan (*contrast*), memberikan pandangan (*criticize*), membandingkan (*synthesize*), dan meringkas (*summary*) [12].

2.1. Hasil Literatur Review

Hasil *literature review* yang telah dilakukan pada jurnal-jurnal penelitian terkait topik kecerdasan buatan

(*Artificial Intelligence/AI*) dan visi computer (*Computer Vision*) dapat dijabarkan sebagai berikut:

Paper 1 membahas terkait klasifikasi citra cuaca menggunakan algoritma *K-Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), dan *Convolutional Neural Network* (CNN). Penelitian ini menjelaskan hasil algoritma terbaik antara penggunaan algoritma *K-Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), dan *Convolutional Neural Network* (CNN) untuk klasifikasi citra cuaca. Hasil dari penelitian tersebut menunjukkan bahwa algoritma CNN memiliki performa terbaik dengan *accuracy* 0.942, *precision* 0.943, *recall* 0.942, dan *F1 Score* 0.942 [3].

Paper 2 membahas klasifikasi jenis buah mangga menggunakan metode *backpropagation*, dengan data yang terdiri dari 4 jenis mangga: mangga apel, mangga arumanis, mangga madu, dan mangga manalagi. Penelitian ini menjelaskan proses pengujian dengan variasi jumlah *neuron*, *epoch*, *goal*, dan *learning rate* untuk mendapatkan akurasi terbaik dalam klasifikasi. Hasil dari penelitian menunjukkan bahwa metode *backpropagation* dapat mengklasifikasi jenis mangga dengan akurasi tertinggi 9 sebesar 100% menggunakan *epoch* 3000, *goal* 0.0001, dan *learning rate* 0.1, serta akurasi 96.05% pada model *traingdx* [6].

Paper 3 membahas tentang sistem deteksi alat pelindung diri (APD) kepala, khususnya helm keselamatan, menggunakan algoritma YOLOv7 dan perangkat keras Jetson Nano untuk meningkatkan akurasi deteksi. Penelitian ini menjelaskan proses perancangan sistem yang dapat mendeteksi pelanggaran penggunaan helm keselamatan secara *real-time*, serta mengirimkan notifikasi melalui Telegram ketika pelanggaran terjadi. Hasil dari penelitian menunjukkan bahwa sistem deteksi dini yang dirancang memiliki performa yang baik, dengan tingkat akurasi mencapai 97,23%, *precision* sebesar 98,71%, dan *recall* sebesar 95,63% [13].

2.2. Klasifikasi Tingkat Kematangan Buah

Klasifikasi tingkat kematangan buah penting untuk meningkatkan kualitas produk dan mengurangi kerugian pascapanen. Berbagai metode telah digunakan, seperti transformasi ruang warna HSI untuk analisis warna pada buah mangga Garifta Merah [7] dan jaringan saraf tiruan *Backpropagation* untuk pengklasifikasian jenis mangga [6]. Pendekatan berbasis pembelajaran mendalam terus dikembangkan untuk meningkatkan akurasi dan efisiensi dalam klasifikasi kematangan buah.

2.3. YOLO (*You Only Look Once*)

YOLO (*You Only Look Once*) adalah algoritma deteksi objek berbasis pembelajaran mendalam yang unggul dalam kecepatan dan akurasi. Algoritma ini bekerja dengan membagi citra menjadi *grid* dan mendeteksi objek secara simultan. YOLO telah digunakan dalam berbagai aplikasi, seperti deteksi plat nomor kendaraan [1] dan deteksi manusia pada robot *mobile* [2].

Dalam penelitian ini, YOLO11 sebagai versi terbaru dari YOLO akan digunakan untuk klasifikasi kematangan buah mangga. YOLO11 menawarkan perbaikan dari versi sebelumnya, seperti peningkatan akurasi dan efisiensi dalam pengolahan citra.

2.4. Metode Optimasi Model Klasifikasi

Optimasi model bertujuan meningkatkan akurasi dan efisiensi YOLO11. Teknik yang digunakan meliputi penyesuaian parameter seperti *batch size* dan *learning rate*. Pendekatan ini didasarkan pada studi sebelumnya, seperti pemanfaatan CNN untuk klasifikasi daun mangga [5] dan optimasi YOLO untuk aplikasi *real-time* [10]. Optimasi dilakukan untuk memastikan model dapat mendeteksi kematangan buah mangga secara akurat dengan memperhatikan fitur visual.

3. METODE PENELITIAN

Penelitian ini menggunakan metode kuantitatif dengan pendekatan eksperimental. Metode ini bertujuan untuk menguji model klasifikasi tingkat kematangan buah mangga menggunakan algoritma YOLO11, yang terkenal unggul dalam deteksi objek secara *real-time* menggunakan teknik *deep learning* [2]. Proses ini mencakup beberapa tahap, mulai dari pengumpulan dataset citra buah mangga dengan variasi warna yang mengindikasikan tingkat kematangan, hingga tahap implementasi dan evaluasi model klasifikasi. Pada gambar 2 ditunjukkan tahapan metode penelitian sebagai berikut:

Gambar 2. Tahapan metode penelitian

Dataset yang digunakan dalam penelitian ini adalah data sekunder yang diambil dari platform Kaggle, dengan judul "*Ripeness Detection of Mango*" yang dikembangkan oleh Muti Ur Rehman. Dataset ini terdiri dari 951 gambar buah mangga dengan variasi tingkat kematangan, yaitu *unripe*, *early ripe*, *partially ripe*, dan *ripe*. Alamat datasetini dapat diakses melalui tautan berikut: https://www.kaggle.com/datasets/mutiurrehman/ripeness-detection-of-mango. Platform Kaggle dipilih karena reputasinya dalam menyediakan dataset

berkualitas tinggi yang relevan dengan topik penelitian ini.

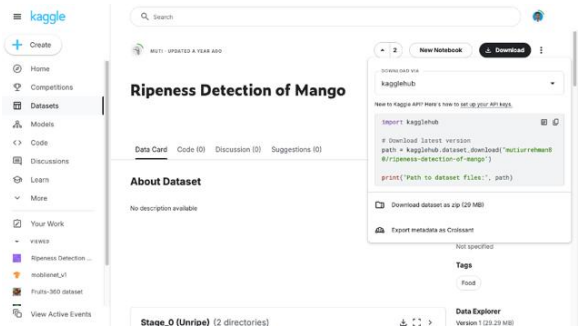
Dataset ini diunduh dalam format ZIP. Setelah diekstrak, foldernya sudah mengelompokkan gambar berdasarkan tingkat kematangan, di mana setiap kategori kematangan memiliki *folder* tersendiri. Namun, dataset juga terbagi dalam dua *subfolder*, yaitu *train* dan *test*. Untuk mempermudah proses pelatihan model dengan distribusi data yang konsisten, kedua *subfolder* tersebut digabungkan secara manual, mengelompokkan semua gambar dari setiap tingkat kematangan ke dalam satu *folder* utama. Setelah *folder train* dan *test* disatukan, data kemudian dibagi ulang menjadi data pelatihan dan pengujian (*train* dan *test*) menggunakan pustaka Python, yaitu *python_splitter*. Pembagian ini dilakukan dengan rasio 80:20 untuk *train* dan *test*, sehingga data siap untuk diproses lebih lanjut dan diimplementasikan dalam pelatihan model YOLO11.

Proses analisis data terdiri dari beberapa tahap utama, yaitu pelatihan model, evaluasi performa, dan analisis hasil. Pada tahap pertama, yaitu pelatihan model, data latih yang telah diproses akan digunakan untuk melatih model YOLO11. Selama proses pelatihan, model mempelajari fitur-fitur visual yang berkaitan dengan tingkat kematangan mangga, seperti perbedaan warna pada kulit. Tahap kedua adalah evaluasi performa model. Setelah pelatihan selesai, model diuji menggunakan data uji yang telah dipersiapkan. Pada tahap ini, performa model dievaluasi dengan menggunakan beberapa metrik: akurasi untuk menentukan seberapa sering prediksi model benar, presisi untuk mengukur ketepatan dalam mengidentifikasi tingkat kematangan tertentu, recall untuk mengukur sensitivitas model dalam mendeteksi semua tingkat kematangan, serta *F1-score* untuk memberikan gambaran seimbang antara presisi dan recall. Tahap terakhir adalah analisis hasil. Hasil dari evaluasi metrik akan dianalisis untuk mengidentifikasi kekuatan dan kelemahan model YOLO11 dalam klasifikasi kematangan mangga. Jika terdapat kategori kematangan yang kurang akurat, misalnya, analisis dilakukan untuk mengetahui apakah faktor tersebut disebabkan oleh faktor pencahayaan atau latar belakang yang bervariasi.

4. HASIL DAN PEMBAHASAN

4.1. Pengumpulan Data

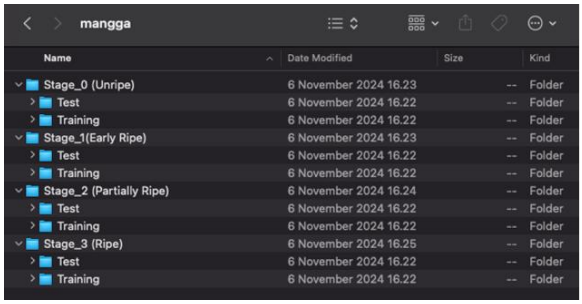
Untuk mengunduh dataset, dilakukan langkah-langkah dengan mengakses halaman dataset di Kaggle dan mengklik tombol *Download*. Data yang diunduh berupa file terkompresi dalam format ZIP, yang berisi kumpulan gambar mangga yang diklasifikasikan berdasarkan tingkat kematangan. Proses ini memastikan bahwa semua data mentah tersedia secara lengkap dan terorganisasi untuk keperluan pemrosesan dan analisis lebih lanjut.



Gambar 3. Tampilan dataset dari platform Kaggle

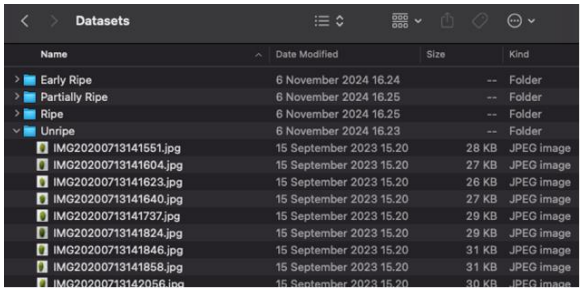
4.2. *Data Cleaning*

Setelah file ZIP dataset diekstrak, ditemukan bahwa struktur data terdiri dari empat *folder* utama yang mewakili tingkat kematangan mangga, yaitu Stage_0 (*Unripe*), Stage_1 (*Early Ripe*), Stage_2 (*Partially Ripe*), dan Stage_3 (*Ripe*). Di dalam masing-masing *folder* tersebut, terdapat dua *subfolder*, yaitu *Training* dan *Test*. Untuk mempermudah proses analisis, data dari kedua *subfolder* ini perlu digabungkan menjadi satu *folder* utama untuk setiap tingkat kematangan.



Gambar 4. Struktur dataset sebelum data cleaning

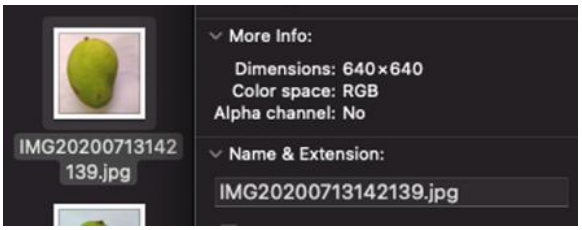
Proses pembersihan data dilakukan dengan menggabungkan isi dari *subfolder Training* dan *Test* ke dalam satu *folder* utama yang sesuai dengan nama kelas tingkat kematangannya dengan cara manual. Dengan demikian, folder Stage_0 diubah namanya menjadi *Unripe*, Stage_1 menjadi *Early Ripe*, Stage_2 menjadi *Partially Ripe*, dan Stage_3 menjadi *Ripe*. Penamaan ulang ini memberikan kejelasan yang lebih baik dan mempermudah pengolahan data pada tahap berikutnya.



Gambar 5. Struktur dataset sesudah data cleaning

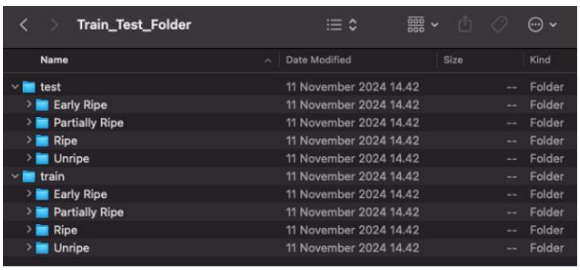
4.3. *Preprocessing*

Data gambar yang telah diorganisir sebelumnya perlu diubah ukurannya agar seragam. Untuk mencapai ini, digunakan library Python PIL (Python Imaging Library) dengan tujuan mengubah resolusi setiap gambar menjadi 640 x 640 piksel.



Gambar 6. Hasil resizing image

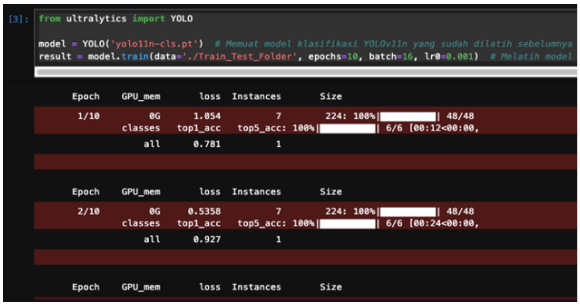
Langkah berikutnya adalah membagi *dataset* ke dalam *folder train* dan *test* dengan perbandingan 80:20, yang penting untuk melatih dan menguji model secara efektif. Proses ini dilakukan menggunakan *library python_splitter*.



Gambar 7. Hasil split data

4.4. *Training Model*

Data yang telah diproses melalui tahap *preprocessing* selanjutnya digunakan untuk melatih model menggunakan YOLO11. Model YOLO11n dipilih karena efisiensinya dalam klasifikasi gambar. Proses pelatihan dilakukan dengan data yang telah dibagi ke dalam *folder Train* dan *Test*, memastikan bahwa model dapat belajar dengan baik dan mengevaluasi performanya secara akurat.



Gambar 8. Proses training model

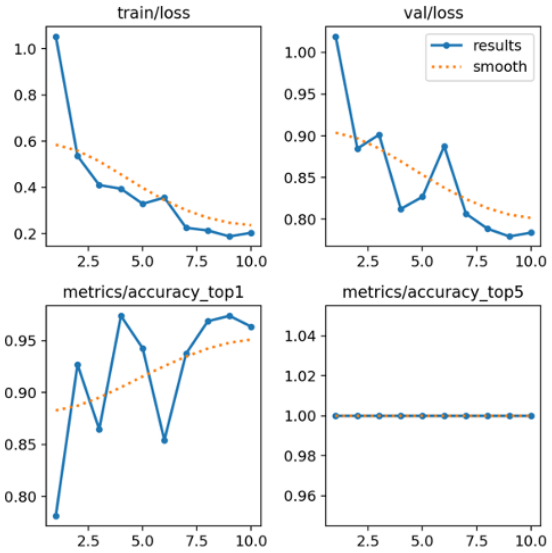
Pelatihan model dilakukan menggunakan parameter yang diatur untuk memaksimalkan efisiensi pembelajaran.

a. *data='./Train_Test_Folder'*: Menunjukkan lokasi dataset yang akan digunakan untuk melatih model. *Folder* ini berisi data yang telah diproses sebelumnya.
b. *epochs=10*: Menetapkan jumlah pengulangan penuh melalui seluruh dataset menjadi 10 *epoch*. Semakin banyak *epoch*, semakin lama model akan belajar, tetapi dengan risiko *overfitting*.
c. *batch=16*: Mengatur ukuran *batch* menjadi 16, artinya model akan memproses 16 gambar sekaligus sebelum memperbarui bobotnya.
d. *lr0=0.001*: Menetapkan nilai awal *learning rate* sebesar 0,001, yang mengontrol seberapa besar model diperbarui selama proses pelatihan.

Pelatihan dimulai dengan menjalankan model menggunakan dataset yang telah disiapkan. Model mempelajari pola dari dataset *Train* dan memvalidasinya menggunakan dataset *Test*. Setiap *epoch* mencakup tahapan berikut:

a. *Forward pass*: Data melewati model, menghasilkan prediksi awal.
b. *Backward pass*: Kesalahan dihitung dan bobot model diperbarui berdasarkan *learning rate*.
c. *Evaluation*: Performansi diukur menggunakan metrik tertentu.

Setelah menjalankan proses pelatihan, visualisasi dari hasil *training* disajikan, mencakup grafik *train/loss*, *val/loss*, *metrics/accuracy_top1*, dan *metrics/accuracy_top5*.



Gambar 9. Visualisasi hasil training model

Dari visualisasi tersebut dapat dijelaskan:

a. Grafik *train/loss* menunjukkan penurunan yang konsisten selama 10 *epoch*, yang menandakan bahwa model belajar dan memperbaiki kesalahan dalam data pelatihan.
b. Grafik *val/loss* juga menurun secara bertahap, menunjukkan bahwa model mampu menggeneralisasi data validasi dengan baik, meskipun ada sedikit fluktuasi yang mungkin mengindikasikan ketidakstabilan.
c. Grafik *metrics/accuracy_top1* menunjukkan akurasi model yang meningkat selama proses pelatihan, menunjukkan bahwa model semakin

mampu mengklasifikasikan gambar dengan benar. Meskipun ada beberapa fluktuasi, tren umumnya meningkat, menunjukkan perbaikan performa.

d. Grafik *metrics/accuracy_top5* tampak datar dan konstan, yang mengindikasikan bahwa akurasi top-5 tidak terlalu relevan untuk tugas klasifikasi ini, atau metrik ini tetap stabil sepanjang pelatihan.

4.5. Evaluasi Model

Setelah menyelesaikan proses pelatihan, performa model dievaluasi untuk mengukur akurasi, *precision*, *recall*, dan *F1-score*. Evaluasi ini dilakukan dengan menggunakan *confusion matrix* yang dihasilkan dari hasil pelatihan, yang disimpan dalam variabel *result*.

Confusion matrix memungkinkan untuk menghitung jumlah prediksi benar dan salah untuk masing-masing kelas, yang kemudian digunakan untuk menghitung metrik performa.

a. Akurasi model tidak dihitung langsung dari *confusion matrix*, melainkan telah tersedia dalam variabel *result.top1*, yang menunjukkan persentase prediksi yang benar untuk setiap sampel dalam dataset.

b. *Precision* mengukur proporsi prediksi positif yang benar. Rumus untuk menghitung *precision* adalah sebagai berikut:

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

Di mana TP (*True Positives*) adalah jumlah prediksi positif yang benar dan FP (*False Positives*) adalah jumlah prediksi positif yang salah.

c. *Recall* mengukur kemampuan model dalam mengidentifikasi seluruh sampel positif yang sebenarnya. Rumus untuk *recall* adalah:

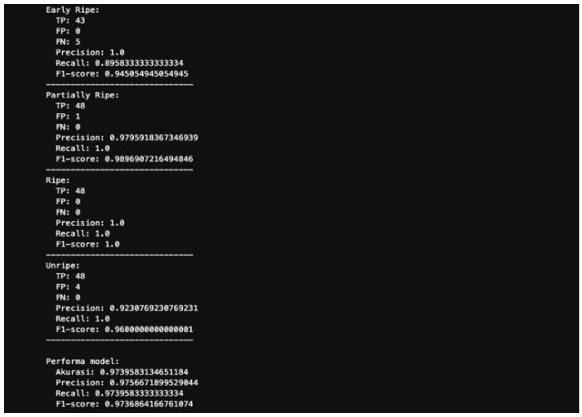
$$Recall = \frac{TP}{TP\ +\ FN} \quad (2)$$

Di mana FN (*False Negatives*) adalah jumlah sampel positif yang salah diklasifikasikan.

d. *F1-score* merupakan metrik yang memberikan keseimbangan antara *precision* dan *recall*. Rumus *F1-score* adalah:

$$F1 - Score = 2 \times \frac{Precision\ \times\ Recall}{Precision + Recall} \quad (3)$$

Rata-rata dari *precision*, *recall*, dan *F1-score* kemudian dihitung untuk memberikan gambaran menyeluruh mengenai performa model, serta untuk menilai keseimbangan antara *recall* dan *precision*. Hasilnya dapat dilihat pada gambar 10 berikut.



Gambar 10. Hasil evaluasi model

Dari hasil tersebut dapat dijelaskan:

a. *Precision* yang didapat dari masing-masing kelas: *Early Ripe* 1.0, *Partially Ripe* 0.98, *Ripe* 1.0, dan *Unripe* 0.92.

b. *Recall* yang didapat dari masing-masing kelas: *Early Ripe* 0.90, *Partially Ripe* 1.0, *Ripe* 1.0, dan *Unripe* 1.0.

c. *F1-score* yang didapat dari masing-masing kelas: *Early Ripe* 0.95, *Partially Ripe* 0.99, *Ripe* 1.0, dan *Unripe* 0.96.

d. Secara keseluruhan didapat: Akurasi 0.973, *Precision* 0.975, *Recall* 0.973, dan *F1-score* 0.973

4.6. Pengujian Model

Pengujian model dilakukan dengan menguji performa model secara langsung menggunakan *webcam*, yang memproses gambar mangga dengan tingkat kematangan berbeda-beda. Tujuan dari pengujian ini adalah untuk menjawab terkait dengan pengaruh variasi pencahayaan dan latar belakang terhadap kinerja model dalam mengklasifikasikan tingkat kematangan buah mangga.

Pada tabel 1 disajikan hasil pengujian model yang dapat dituliskan untuk masing-masing kelas.

Tabel 1. Hasil pengujian model

Percobaan	Unripe	Early Ripe	Partially Ripe	Ripe
1	66.18%	63.84%	51.92%	98.01%
2	68.92%	69.35%	54.14%	98.70%
3	76.73%	71.94%	57.13%	99.25%

Selain hasil-hasil pengujian model pada tabel 1, hasil lain yang dapat di-*capture* disajikan pada gambar 11 berikut.

Gambar 11. Hasil pengujian model

Dari hasil pengujian pada gambar 11, model dapat mengklasifikasikan tingkat kematangan buah mangga dengan akurasi *Unripe* 79.73%, *Early Ripe* 72.41%, *Partially Ripe* 58.92%, dan *Ripe* 99.85%.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, penelitian ini berhasil mengembangkan model klasifikasi tingkat kematangan buah mangga menggunakan metode YOLO11, yang dilatih dengan dataset citra mangga melalui tahapan pengumpulan data dari Kaggle, pembersihan data, serta *preprocessing* berupa *resizing* dan pemisahan data untuk pelatihan dan pengujian. Dengan konfigurasi parameter seperti jumlah *epoch*, ukuran *batch*, dan *learning rate*, model mampu mengekstraksi fitur penting dari citra untuk klasifikasi dengan performa yang sangat baik, yaitu akurasi keseluruhan sebesar 97,3%, *precision* 97,5%, *recall* 97,3%, dan *F1-score* 97,3%, serta mengenali tingkat kematangan seperti *Early Ripe*, *Partially Ripe*, *Ripe*, dan *Unripe* dengan presisi tinggi. Visualisasi hasil pelatihan, seperti grafik *train/loss* dan *val/loss*, menunjukkan konsistensi pembelajaran dan generalisasi model, meskipun variasi pencahayaan dan latar belakang citra memengaruhi performa, terutama pada intensitas cahaya rendah dan latar belakang kompleks. Pengujian dengan *webcam* mengonfirmasi akurasi tinggi untuk kelas *Ripe*, tetapi akurasi menurun pada kelas *Partially Ripe* dan *Early Ripe*, menunjukkan bahwa kondisi pencahayaan dan latar belakang yang bervariasi dapat memengaruhi kemampuan model dalam mengenali fitur yang relevan untuk klasifikasi. Beberapa saran untuk pengembangan lebih lanjut meliputi meningkatkan performa model dengan teknik augmentasi data seperti penyesuaian kecerahan, kontras, rotasi, dan penambahan *noise*, serta memanfaatkan *transfer learning* dengan model pralatih untuk akurasi yang lebih baik. Disarankan juga menggunakan dataset yang lebih besar dan beragam, mencakup variasi musim, jenis mangga, dan kondisi cuaca, serta melibatkan sumber data selain Kaggle untuk mengurangi bias. Evaluasi performa model pada perangkat keras seperti Jetson Nano atau Raspberry Pi penting dilakukan untuk mengukur efisiensi dan responsivitasnya dalam aplikasi *real-time*. Selain itu, pengembangan aplikasi berbasis *mobile* atau *web* dengan antarmuka ramah pengguna dapat mendukung petani atau pengepul mangga, serta membandingkan algoritma seperti YOLO11, CNN, dan *Random Forest* dapat membantu menentukan metode terbaik untuk klasifikasi citra mangga.

DAFTAR PUSTAKA

[1] A. Aprilino and I. H. Al Amin, "IMPLEMENTASI ALGORITMA YOLO DAN TESSERACT OCR PADA SISTEM DETEKSI PLAT NOMOR OTOMATIS," *Jurnal TEKNOINFO*, vol. 16, no. 1, pp. 54–59, 2022.

[2] K. Khairunnas, E. M. Yuniarno, and A. Zaini, "Pembuatan Modul Deteksi Objek Manusia Menggunakan Metode YOLO untuk Mobile Robot," *JURNAL TEKNIK ITS*, vol. 10, no. 1, pp. A50–A55, 2021.

[3] M. F. Naufal, "ANALISIS PERBANDINGAN ALGORITMA SVM, KNN, DAN CNN UNTUK KLASIFIKASI CITRA CUACA," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 8, no. 2, pp. 311–318, 2021, doi: 10.25126/jtiik.202184553.

[4] D. Hidayat, "KLASIFIKASI JENIS MANGGA BERDASARKAN BENTUK DAN TEKSTUR DAUN MENGGUNAKAN METODE CONVOLUTIO NALNEURAL NETWORK(CNN)," *Journal of Information Technology and Computer Science (INTECOMS)*, vol. 5, no. 1, pp. 98–103, 2022.

[5] T. Ayu, V. Dwi, and A. E. Minarno, "PENDIAGNOSA DAUN MANGGA DENGAN MODEL CONVOLUTIONAL NEURAL NETWORK," *CESS (Journal of Computer Engineering System and Science)*, vol. 6, no. 2, pp. 230–235, 2021.

[6] J. Jamaludin, C. Rozikin, and A. S. Y. Irawan, "Klasifikasi Jenis Buah Mangga dengan Metode Backpropagation," *Techné: Jurnal Ilmiah Elektroteknika*, vol. 20, no. 1, pp. 1–12, 2021.

[7] A. M. Syafi'i, M. F. Ahadi, M. I. Rasyid, F. D. Adhinata, and A. Junaidi, "Klasifikasi Kematangan Pada Buah Mangga Garifta Merah dengan Transformasi Ruang Warna HSI," *Journal of Applied Informatics and Computing (JAIC)*, vol. 5, no. 2, pp. 117–121, 2021.

[8] L. Rahma, H. Syaputra, A. H. Mirza, and S. D. Purnamasari, "Objek Deteksi Makanan Khas Palembang Menggunakan Algoritma YOLO (You Only Look Once)," *Jurnal Nasional Ilmu Komputer*, vol. 2, no. 3, pp. 2746–1343, 2021.

[9] A. Herdiansah, R. I. Borman, D. Nurnaningsih, A. A. J. Sinlae, and R. R. Al Hakim, "Klasifikasi

Citra Daun Herbal Dengan Menggunakan Backpropagation Neural Networks Berdasarkan Ekstraksi Ciri Bentuk," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 2, pp. 388–395, 2022, doi: 10.30865/jurikom.v9i2.4066.

[10] D. I. Mulyana and M. A. Rofik, "Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan Di Indonesia Menggunakan Metode YOLOV5," *Jurnal Pendidikan Tambusai*, vol. 6, no. 3, pp. 13971–13982, 2022.

[11] T. Yuniati and M. F. Sidiq, "Literature Review: Legalisasi Dokumen Elektronik Menggunakan Tanda Tangan Digital sebagai Alternatif Pengesahan Dokumen di Masa Pandemi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 6, pp. 1058–1069, 2020, doi: 10.29207/resti.v4i6.2502.

[12] R. H. Putri and D. Mulyanti, "Literatur Riview Tentang Analisa Implementasi Sistem Informasi Manajemen Rumah Sakit (Simrs)," *Jurnal Ilmiah Kedokteran dan Kesehatan*, vol. 2, no. 2, pp. 14–28, 2023, doi: 10.55606/klinik.v2i2.1237.

[13] H. Supriyanto, S. C. Abadi, and A. Shalsabilah, "Deteksi Helm Keselamatan Menggunakan Jetson Nano dan YOLOv7," *Journal of Applied Computer Science and Technology (JACOST)*, vol. 5, no. 1, pp. 1–8, 2024, doi: 10.52158/jacost.637