

Sistem Deteksi Otomatis Buah Berbasis Computer Vision Menggunakan YOLO

Alfin Afriza Maulana¹

¹Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri

E-mail: [*alfinsiesica@gmail.com](mailto:alfinsiesica@gmail.com)

Abstrak – Proses identifikasi dan penyortiran buah secara manual dinilai kurang efisien dan subjektif, sehingga diperlukan otomatisasi menggunakan teknologi Computer Vision. Penelitian ini bertujuan mengembangkan aplikasi deteksi buah real-time pada perangkat Android menggunakan algoritma YOLOv8 yang dioptimalkan untuk efisiensi komputasi mobile. Metode penelitian menerapkan pendekatan eksperimental yang meliputi pengumpulan 8.481 citra dataset multi-kelas, teknik augmentasi data, serta pelatihan model menggunakan framework PyTorch selama 50 epoch. Hasil pengujian menunjukkan bahwa model memiliki karakteristik konservatif dengan rata-rata Precision sebesar 66% dan Recall 45%. Variasi performa sangat dipengaruhi oleh kompleksitas visual objek, di mana buah dengan fitur distingtif seperti semangka memiliki akurasi lebih tinggi dibandingkan objek kecil yang bergerombol seperti anggur. Secara fungsional, sistem berhasil diintegrasikan ke dalam lingkungan Android dan mampu melakukan inferensi secara lancar tanpa hambatan yang berarti. Penelitian ini menyimpulkan bahwa arsitektur YOLOv8 terbukti efektif sebagai solusi portabel yang ringan untuk identifikasi buah di lapangan, meskipun pengayaan variasi dataset masih diperlukan untuk meningkatkan sensitivitas model terhadap kondisi latar belakang yang kompleks.

Kata Kunci — Android, Computer Vision, Deteksi Buah, Real-time, YOLOv8.

1. PENDAHULUAN

Letak geografi Indonesia yang strategis, tepat di garis khatulistiwa, memberikan keunggulan alam yang signifikan dalam bidang hortikultura, terutama dalam hal produksi buah-buahan [1]. Namun, besarnya potensi produksi ini sering kali menghadapi kendala pada tahap penanganan pascapanen. Di lapangan, proses identifikasi dan penyortiran kualitas buah umumnya masih dilakukan secara manual melalui pengamatan visual yang bersifat subjektif dan rentan terhadap kesalahan, terutama ketika menangani volume panen yang besar atau pada kondisi lingkungan yang kurang ideal [2]. Ketergantungan pada metode manual ini menjadi tantangan serius bagi para petani karena memerlukan waktu dan tenaga yang cukup besar untuk memastikan kualitas buah sesuai dengan standar pasar. Selain inefisiensi waktu, penyortiran secara manual juga sering menyebabkan informasi klasifikasi mutu yang dihasilkan berakhir subjektif dan tidak konsisten, sehingga sulit menjamin standarisasi produk [3].

Meskipun teknologi telah mengalami perkembangan pesat, proses klasifikasi kualitas buah di lapangan sering kali masih bergantung pada penilaian subjektif manusia yang rentan terhadap ketidakkonsistenan, memakan waktu lama, serta dipengaruhi oleh faktor kelelahan dan perbedaan persepsi antar pengamat [1]. Untuk mengatasi kendala fisik dan keterbatasan penglihatan manusia tersebut, penerapan metode kecerdasan buatan (Artificial Intelligence) menjadi solusi krusial guna meningkatkan efisiensi dan akurasi dalam sistem deteksi mutu buah [3]. Kemajuan ini membuka potensi besar dalam bidang Computer Vision, yang memungkinkan komputer untuk menganalisis dan menginterpretasi gambar dengan cara yang menyerupai persepsi manusia [1]. Sebagai implementasi dari teknologi tersebut, algoritma deteksi objek YOLOv8 kini diterapkan untuk menggantikan metode konvensional. Efektivitas model ini telah dibuktikan dalam berbagai konteks, termasuk menunjukkan performa sangat baik pada varian YOLOv8 dan membuka jalan bagi pengembangan aplikasi interaktif berbasis AI [4]. Namun, untuk kebutuhan implementasi pada perangkat dengan sumber daya terbatas, varian YOLOv8n (Nano) menjadi pilihan yang lebih ideal karena memiliki jumlah parameter yang kecil dan kebutuhan komputasi yang rendah, sehingga memungkinkan pengenalan objek yang cepat dan presisi secara otomatis [5].

Dalam penelitian ini, algoritma YOLOv8 dipilih sebagai metode utama karena reputasinya sebagai kerangka kerja canggih yang terbukti memiliki kinerja terbaik dalam hal akurasi dan efisiensi jika dibandingkan dengan versi YOLO lainnya [6]. Popularitas model ini dalam tugas deteksi waktu nyata (real-time) menjadi aspek krusial, terutama untuk aplikasi yang membutuhkan respons cepat dan efisiensi operasional di lapangan. Selain keunggulan teknis tersebut, YOLOv8 merupakan metode yang telah mapan dengan dukungan komunitas pengguna yang luas, yang menjamin ketersediaan sumber daya implementasi yang mudah diakses. Fleksibilitas algoritma ini, yang dapat diterapkan dalam berbagai ukuran versi model, menjadikannya solusi yang adaptif untuk pengembangan sistem dengan kebutuhan komputasi yang bervariasi [7].

Pemilihan metode ini didukung oleh berbagai literatur yang membuktikan efektivitas algoritma YOLO dalam skenario serupa. Sebagai contoh, penelitian terdahulu telah berhasil mengembangkan model deteksi kematangan

buah manggis menggunakan algoritma ini, yang dipilih karena kemampuannya memberikan hasil deteksi cepat dan akurat melalui metode bounding boxes, menjadikannya sangat cocok untuk aplikasi yang memerlukan respons waktu nyata [1]. Lebih jauh lagi, implementasi varian terbaru YOLOv8 diharapkan mampu menyelesaikan tantangan deteksi dan penghitungan objek yang lebih kompleks dengan efisiensi tinggi, serta memberikan kontribusi nyata terhadap perkembangan teknologi deteksi objek dalam skala besar [8]. Berdasarkan landasan tersebut, penelitian ini mengadopsi arsitektur YOLOv8 untuk diterapkan pada deteksi multi-kelas (apel, pisang, semangka) di platform mobile, guna menghadirkan solusi yang tidak hanya akurat tetapi juga praktis bagi pengguna.

2. METODE PENELITIAN

Penelitian ini menerapkan metode eksperimental dengan pendekatan pengembangan sistem Deep Learning untuk deteksi objek pada perangkat mobile. Tahapan penelitian meliputi pengumpulan data, pra-pemrosesan data, pelatihan model menggunakan algoritma YOLOv8, dan implementasi pada platform Android..

2.1 Inisialisasi Penelitian

Tahapan awal penelitian dimulai dengan inisialisasi yang mencakup perencanaan sistematis dan pemahaman mendalam terhadap tujuan studi. Di sini, peneliti menentukan fokus utama penelitian, yaitu pengembangan sistem deteksi objek multi-kelas untuk komoditas buah (apel, pisang, anggur, jeruk, jambu, dan semangka) dengan mengimplementasikan algoritma YOLOv8. Dasar pemilihan ini merujuk pada literatur yang menyatakan bahwa YOLO merupakan algoritma Deep Learning berbasis CNN yang menerapkan fitur deteksi dalam bentuk bounding box dan klasifikasi, serta dinilai sangat cocok untuk implementasi kasus real-time dengan estimasi kemampuan memproses gambar hingga 45 FPS (Frame per Second) [9]. Berlandaskan kapabilitas teknis tersebut, tujuan spesifik dari tahapan ini adalah merancang solusi otomatisasi yang mampu menggantikan proses identifikasi manual, dengan menargetkan integrasi model ke dalam perangkat bergerak (mobile) agar dapat digunakan secara praktis dan responsif di lapangan.

2.2 Pengumpulan dan Pra-pemrosesan Data

Penelitian ini menggunakan dataset publik yang diperoleh dari repositori Kaggle. Platform ini dipilih karena menyediakan ruang komprehensif bagi ilmuwan data untuk berbagi ide dan mengeksplorasi aplikasi nyata, dengan ketersediaan dataset yang bersifat autentik dan relevan dalam berbagai format untuk keperluan referensi pelatihan model [10]. Dataset yang dikumpulkan terdiri dari citra digital buah-buahan dengan variasi objek multi-kelas (Apel, Pisang, Anggur, Jeruk, Jambu, dan Semangka), sebagaimana ditampilkan pada Gambar 1. Total dataset berjumlah 8.481 citra, yang kemudian dipartisi ke dalam tiga sub-himpunan untuk memastikan evaluasi model yang objektif. Proses pembagian data (data splitting) dilakukan dengan mengalokasikan proporsi terbesar untuk data latih (training set) sebanyak 7.110 gambar (sekitar 84%) guna memaksimalkan proses pembelajaran fitur. Sementara itu, sebanyak 914 gambar (11%) dialokasikan sebagai data validasi (validation set) untuk memantau kinerja model selama pelatihan, dan sisanya sebanyak 457 gambar (5%) digunakan sebagai data uji (testing set) untuk mengukur performa akhir model pada data yang belum pernah dilihat sebelumnya.



Gambar 1. Dataset buah

2.3 Analisis Sistem

Sistem identifikasi dan penyortiran buah konvensional umumnya masih mengandalkan pengamatan visual secara manual oleh manusia. Pendekatan ini memiliki keterbatasan karena sangat bergantung pada subjektivitas pengamat, yang sering kali menyebabkan ketidakkonsistenan penilaian mutu akibat faktor kelelahan atau perbedaan persepsi visual. Selain itu, proses manual mengharuskan pengguna untuk memeriksa setiap buah satu per satu secara fisik, yang merupakan proses yang memakan waktu dan kurang efisien untuk menangani volume panen yang besar.

Sistem yang diusulkan bertujuan untuk mengotomatisasi proses tersebut dengan mengembangkan aplikasi berbasis Android yang memanfaatkan teknologi Computer Vision. Berbeda dengan sistem manual, sistem ini menggunakan algoritma YOLOv8 untuk mendeteksi dan mengklasifikasikan jenis buah (apel, pisang, semangka, anggur, jeruk, jambu) secara real-time melalui kamera perangkat mobile. Dengan pendekatan ini, sistem diharapkan mampu memberikan penilaian yang lebih objektif, cepat, dan akurat, serta menyajikan informasi visual langsung kepada pengguna melalui antarmuka aplikasi tanpa memerlukan koneksi internet yang berat (pemrosesan on-device).

2.4 Proses Pelabelan Data (Labeling)

Setelah seluruh data citra terkumpul, penelitian berlanjut ke tahap pelabelan data (data labeling). Proses ini melibatkan pembuatan anotasi berupa bounding box (kotak pembatas) yang secara presisi melingkupi area objek target yaitu buah apel, pisang, anggur, jambu dan semangka pada setiap gambar dalam dataset. Penandaan ini bertujuan vital untuk melatih model agar mampu mengenali karakteristik fitur visual serta lokasi spasial dari masing-masing kelas objek yang terdapat di dalam citra, sehingga model dapat membedakan antar jenis buah dengan akurat saat proses pelatihan berlangsung.

2.5 Arsitektur Model YOLOv8

Penelitian ini menerapkan arsitektur YOLOv8 (You Only Look Once version 8) sebagai inti dari sistem pemrosesan citra. Arsitektur ini berfungsi sebagai kerangka komputasi yang memungkinkan sistem untuk mempelajari pola visual kompleks dari data latih dan melakukan prediksi pada data baru. Kegunaan utama arsitektur ini terletak pada kemampuannya menyeimbangkan dua aspek krusial: kecepatan inferensi dan akurasi deteksi.

Jika arsitektur ini tidak digunakan atau digantikan dengan metode konvensional, sistem akan menghadapi kendala signifikan dalam memproses variasi morfologi buah yang dinamis. Tanpa mekanisme ekstraksi fitur otomatis yang dimiliki YOLOv8 (melalui struktur Backbone dan Head-nya), komputer tidak akan mampu membedakan objek target dari latar belakang secara efektif, terutama pada kondisi pencahayaan yang berubah-ubah. Selain itu, pemilihan spesifik YOLOv8 didasarkan pada desainnya yang lightweight (ringan), yang memungkinkan model dijalankan secara efisien pada perangkat mobile dengan sumber daya terbatas tanpa menyebabkan latensi (lag) yang mengganggu pengalaman pengguna.

2.6 Pelatihan Model (Training)

Proses pelatihan model dieksekusi menggunakan kerangka kerja (framework) PyTorch. Konfigurasi hyperparameters diatur secara spesifik untuk mencapai performa optimal, di mana jumlah epoch ditetapkan sebanyak 50. Pemilihan durasi pelatihan ini bertujuan untuk memberikan kesempatan yang cukup bagi model dalam mempelajari fitur visual secara mendalam hingga mencapai titik konvergensi, tanpa memicu terjadinya overfitting. Sejalan dengan upaya tersebut, strategi augmentasi data diterapkan untuk meningkatkan generalisasi model terhadap berbagai kondisi dunia nyata, yang meliputi teknik pembalikan horizontal, rotasi, penyesuaian kecerahan, dan penskalaan [11]. Selama proses iterasi ini, fungsi loss dihitung secara berkelanjutan untuk meminimalkan margin kesalahan prediksi, baik pada penentuan bounding box maupun klasifikasi objek. Selain itu, strategi transfer learning dengan memanfaatkan bobot pra-latih (pre-trained weights) dari dataset turut diterapkan guna mempercepat proses adaptasi model terhadap dataset buah lokal yang digunakan.

2.7 Evaluasi

Kinerja model dievaluasi menggunakan metrik standar untuk deteksi objek. Metrik utama yang digunakan meliputi Precision, Recall, dan Mean Average Precision (mAP). Perhitungan metrik ini didasarkan pada komponen Confusion Matrix, yaitu True Positive (TP), False Positive (FP), dan False Negative (FN) .

Secara matematis, Precision (P) didefinisikan sebagai rasio prediksi benar positif terhadap keseluruhan hasil prediksi positif, Persamaan untuk menghitung Precision dapat dilihat pada Persamaan (1):

$$P = \frac{TP}{TP+FP} \dots \dots \dots (1)$$

Sedangkan Recall (R) mengukur kemampuan model dalam menemukan kembali seluruh objek yang relevan (ground truth) dalam citra, sebagaimana ditunjukkan pada Persamaan (2) :

$$R = \frac{TP}{TP+FN} \dots \dots \dots (2)$$

Untuk mengukur akurasi keseluruhan yang menyeimbangkan nilai Precision dan Recall, digunakan metrik Mean Average Precision (mAP). Nilai mAP diperoleh dengan menghitung rata-rata dari Average Precision (AP) untuk seluruh kelas (N) yang dideteksi, seperti terlihat pada Persamaan (3):

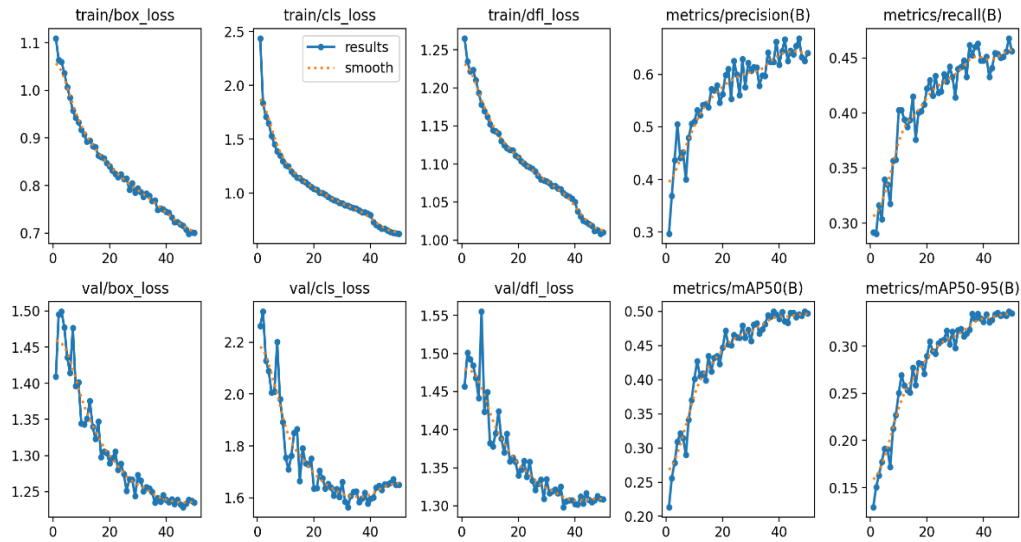
$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \dots \dots \dots (3)$$

Kombinasi metrik ini digunakan sebagai indikator validitas model dalam mengenali jenis buah dan memposisikan bounding box dengan tepat.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pelatihan Model (*Training Results*)

Proses pelatihan model YOLOv8 dilakukan menggunakan dataset yang telah divalidasi. Indikator keberhasilan pelatihan diamati melalui grafik penurunan nilai Loss dan peningkatan nilai mAP (mean Average Precision). Penurunan loss menunjukkan bahwa model semakin mampu meminimalisir kesalahan prediksi seiring bertambahnya epoch.



Gambar 2. Grafik Loss/MAP dari hasil training

Berdasarkan Gambar 2, terlihat bahwa model mulai mencapai konvergensi (stabil) pada epoch ke-40 hingga ke-45. Hal ini menandakan bahwa model telah mempelajari fitur visual dari buah apel, pisang, jeruk, anggur, mangga, dan semangka dengan cukup baik tanpa mengalami overfitting yang signifikan. Hasil ini sejalan dengan karakteristik arsitektur YOLO yang dirancang untuk melakukan ekstraksi fitur secara cepat berkat struktur Backbone dan Head jaringannya yang efisien.

3.2 Evaluasi Kinerja Kuantitatif

Setelah proses pelatihan selesai, dilakukan pengujian menggunakan data uji (test set) untuk mengukur performa model secara objektif. Hasil evaluasi berdasarkan metrik Precision, Recall, dan F1-Score disajikan dalam Tabel 3.1.

Tabel 1. Hasil Evaluasi Kinerja Model YOLOv8

No	Kelas (Class)	True Positive (TP)	False Positive (FP)	False Negative (FN)	Precision (%)	Recall (%)	F1-Score (%)
1	Apple	235	110	322	68%	42%	52%
2	Banana	184	95	206	66%	47%	55%
3	Orange	294	184	516	61%	36%	45%
4	Grape	355	178	745	67%	32%	43%
5	Mango	81	31	73	72%	53%	61%
6	Watermelon	135	76	84	64%	62%	63%
	Rata-rata	-	-	-	66%	45%	53%

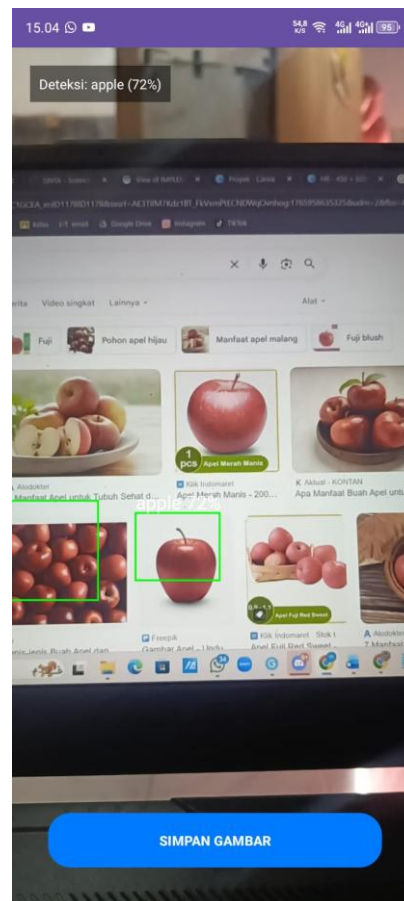
Berdasarkan Tabel 1, sistem berhasil mencapai rata-rata F1-Score sebesar 53% dengan rata-rata Precision 66% dan Recall 45%. Analisa mendalam terhadap hasil ini menunjukkan beberapa temuan penting:

- 3.2.1
- Dominasi presisi terhadap recall menunjukkan nilai rata-rata Precision (66%) yang lebih tinggi dibandingkan Recall (45%) mengindikasikan bahwa model cenderung "konservatif". Artinya, ketika model mendeteksi buah, kemungkinannya benar cukup tinggi, namun model masih kesulitan mengenali seluruh buah yang ada di dalam gambar (banyak objek terlewat atau False Negative).

- 3.2.2 Analisis lebih lanjut pada tingkat kelas menunjukkan adanya variasi performa yang signifikan. Kelas Mango dan Watermelon mencatatkan kinerja paling seimbang dengan F1-Score masing-masing mencapai 61% dan 63%, yang kemungkinan besar disebabkan oleh karakteristik visual bentuk serta tekstur kedua buah tersebut yang lebih distingtif dan mudah dibedakan dari latar belakang. Berbeda halnya dengan kelas Grape (Anggur) yang menunjukkan performa terendah dengan nilai Recall hanya 32% dan tingginya angka False Negative mencapai 745 kasus. Rendahnya sensitivitas pada kelas ini dapat dimaklumi mengingat kompleksitas visual anggur yang sering kali bergerombol (occlusion) serta ukurannya yang relatif kecil dibandingkan buah lain seperti semangka, sehingga menyulitkan model untuk mengidentifikasi objek tersebut secara utuh.
- 3.2.3 Tingginya false negative pada kelas Apple dan Orange, jumlah False Negative (objek tidak terdeteksi) lebih tinggi daripada True Positive. Faktor pencahayaan atau kemiripan warna buah dengan latar belakang dataset kemungkinan menjadi penyebab utama rendahnya sensitivitas model pada kelas-kelas ini.

3.3 Implementasi dan Pengujian Real-time

Model yang telah dilatih kemudian diekspor dan diintegrasikan ke dalam aplikasi Android. Pengujian dilakukan secara real-time untuk memvalidasi kemampuan deteksi dalam skenario penggunaan nyata. Hasil deteksi antarmuka aplikasi ditampilkan pada Gambar 3.2.



Gambar 3. Hasil deteksi buah secara real-time pada aplikasi Android

Berdasarkan gambar 3 secara fungsional, sistem telah mampu menggambar bounding box dan menampilkan label kelas beserta skor kepercayaan (confidence score) secara real-time pada perangkat mobile. Kelancaran ini didukung oleh arsitektur YOLOv8 yang ringan (lightweight), sejalan dengan teori optimasi model untuk perangkat dengan sumber daya terbatas (edge devices).

Namun, terdapat fenomena menarik saat pengujian lapangan yang sedikit berbeda dengan hasil data uji:

- 3.3.1 Isu False Positive (Overconfidence): Meskipun data tabel menunjukkan masalah utama adalah False Negative (objek terlewat), pada kondisi real-time tertentu ditemukan adanya False Positive. Model terkadang mendeteksi objek latar belakang atau benda asing sebagai buah. Hal ini mengindikasikan

adanya background confusion, di mana model kesulitan membedakan fitur visual target dengan noise di lingkungan yang kompleks.

3.4 Analisis dan Pembahasan Hasil

Berdasarkan serangkaian pengujian yang dilakukan mulai dari pelatihan model hingga implementasi pada perangkat Android, evaluasi komprehensif menunjukkan bahwa model YOLOv8 yang dikembangkan memiliki karakteristik "konservatif". Hal ini terlihat dari rata-rata nilai Precision (66%) yang secara signifikan lebih tinggi dibandingkan rata-rata Recall (45%). Disparitas ini mengindikasikan bahwa sistem memprioritaskan ketepatan prediksi—memastikan objek yang terdeteksi adalah benar-benar buah—dibandingkan sensitivitas untuk menemukan seluruh objek. Dalam konteks penggunaan aplikasi oleh pengguna awam, karakteristik ini memberikan keuntungan berupa minimnya informasi yang menyesatkan (misleading), meskipun pengguna mungkin perlu melakukan penyesuaian sudut pengambilan gambar untuk mengatasi keterbatasan model dalam mendeteksi objek secara menyeluruh.

Analisis lebih mendalam pada tingkat kelas memperlihatkan adanya korelasi kuat antara kompleksitas visual objek dengan tingkat keberhasilan deteksi. Buah dengan fitur morfologi yang solid dan tekstur kulit khas, seperti Semangka dan Mangga, terbukti lebih mudah dikenali oleh model. Sebaliknya, objek dengan fitur repetitif, berukuran kecil, dan rentan mengalami oklusi (saling menumpuk) seperti Anggur menjadi tantangan tersendiri bagi arsitektur model. Fenomena ini membuktikan bahwa pada kondisi dataset dengan jumlah sampel terbatas, variasi bentuk dan tekstur objek sangat mempengaruhi kemampuan generalisasi model dalam membedakan fitur antar kelas.

Selain evaluasi metrik, terdapat temuan menarik terkait kesenjangan antara hasil data uji dengan kondisi lapangan. Secara statistik, kelemahan utama model adalah tingginya False Negative (objek terlewat). Namun, pengujian di lingkungan nyata terkadang memunculkan fenomena False Positive akibat background noise. Disparitas ini menunjukkan bahwa variasi latar belakang pada data latih (training set) belum sepenuhnya merepresentasikan kompleksitas lingkungan operasional yang dinamis, sehingga memicu kebingungan (confusion) model saat dihadapkan pada objek asing yang memiliki kemiripan spektrum warna dengan target buah.

Terlepas dari tantangan akurasi tersebut, dari sisi komputasi, integrasi model ke dalam lingkungan Android Studio terbukti berhasil secara fungsional. Sistem mampu memproses input kamera dan melakukan inferensi secara *real-time* tanpa hambatan (*lag*) yang berarti. Capaian ini menegaskan efisiensi arsitektur YOLOv8 sebagai *single-stage detector* yang ringan, yang tidak hanya unggul dalam kecepatan ekstraksi fitur tetapi juga kompatibel untuk diterapkan pada perangkat *edge* dengan sumber daya komputasi terbatas, menjadikannya solusi yang layak untuk aplikasi berbasis *mobile*.

4. SIMPULAN

Berdasarkan hasil implementasi dan analisis pengujian sistem deteksi buah menggunakan algoritma YOLOv8 pada platform Android, dapat ditarik beberapa simpulan sebagai berikut:

1. Efisiensi Implementasi Mobile: Penelitian ini berhasil membuktikan bahwa arsitektur YOLOv8 memiliki kompatibilitas yang sangat baik untuk diterapkan pada perangkat mobile dengan sumber daya terbatas. Sistem mampu melakukan proses inferensi secara *real-time* tanpa hambatan kinerja yang signifikan, menawarkan solusi portabel yang praktis untuk kebutuhan identifikasi komoditas pertanian di lapangan.
2. Karakteristik Model Konservatif: Dari segi akurasi, model yang dihasilkan memiliki karakteristik "konservatif" dengan nilai Precision yang dominan dibandingkan Recall. Kelebihan dari karakteristik ini adalah tingginya tingkat kepercayaan pada objek yang terdeteksi (minim prediksi salah/sampah), sehingga informasi yang diterima pengguna cenderung valid. Namun, kekurangannya adalah tingkat sensitivitas yang rendah, menyebabkan sistem sering melewatkan keberadaan buah dalam frame (False Negative).
3. Pengaruh Morfologi Objek: Kemampuan deteksi model sangat dipengaruhi oleh fitur visual objek. Buah dengan bentuk solid dan tekstur distingtif (seperti Semangka dan Mangga) memiliki tingkat keberhasilan deteksi tertinggi. Sebaliknya, buah dengan karakteristik bergerombol, berukuran kecil, atau memiliki warna yang membur dengan latar belakang (seperti Anggur) menjadi titik lemah utama sistem ini.
4. Keterbatasan Lingkungan: Kendala utama yang ditemukan adalah variasi pencahayaan dan background noise pada pengujian *real-time*. Model masih mengalami kesulitan dalam membedakan objek target dengan latar belakang yang kompleks, yang mengindikasikan bahwa variasi data latih saat ini belum cukup representatif untuk mencakup seluruh dinamika kondisi lapangan.

5. SARAN

Berdasarkan hasil analisis dan evaluasi yang telah dilakukan, terdapat beberapa rekomendasi strategis untuk pengembangan penelitian selanjutnya guna menyempurnakan performa sistem deteksi buah:

1. Peningkatan Heterogenitas dan Kualitas Dataset

Merujuk pada temuan empiris mengenai rendahnya nilai sensitivitas (Recall), terutama pada deteksi objek dengan karakteristik berkerumun (clustered) dan mengalami oklusi parsial seperti anggur, penelitian selanjutnya direkomendasikan untuk memprioritaskan peningkatan kualitas representasi data. Fokus utama tidak hanya terletak pada penambahan kuantitas citra (quantity), melainkan pada eskalasi variabilitas data latih (variability). Peneliti disarankan untuk mengkurasi dataset yang mencakup skenario visual ekstrem, meliputi oklusi berat, pengambilan citra dari sudut pandang non-standar (extreme viewpoints), serta inkorporasi negative samples (citra latar belakang tanpa objek) secara proporsional. Pendekatan ini esensial guna melatih model agar lebih diskriminatif dalam membedakan fitur objek terhadap background noise, sehingga dapat memitigasi tingkat kesalahan deteksi positif (False Positive) yang diakibatkan oleh kemiripan spektrum warna lingkungan.

2. Intensifikasi Strategi Augmentasi Data

Guna meningkatkan robustas (robustness) model terhadap dinamika lingkungan yang tidak terprediksi, implementasi teknik augmentasi data yang lebih agresif sangat disarankan. Simulasi variabilitas iluminasi melalui manipulasi parameter kecerahan (brightness), kontras, dan saturasi perlu diterapkan secara intensif untuk mereplikasi kondisi pencahayaan lapangan yang fluktuatif, mulai dari intensitas cahaya tinggi (overexposed) hingga rendah (underexposed). Selain itu, injeksi distorsi visual seperti Gaussian noise dan blurring patut dipertimbangkan sebagai metode untuk meningkatkan toleransi model terhadap penurunan kualitas optik kamera atau kegagalan fokus, sehingga reliabilitas sistem tetap terjaga dalam berbagai kondisi akuisisi citra.

3. Integrasi Algoritma Pelacakan dan Eksplorasi Arsitektur Model

Ditinjau dari aspek pengembangan perangkat lunak, integrasi mekanisme pelacakan objek (object tracking) berbasis algoritma seperti ByteTrack atau DeepSORT direkomendasikan untuk meningkatkan stabilitas deteksi pada data video real-time. Implementasi fitur ini bertujuan untuk memitigasi redudansi perhitungan objek (double counting) serta menjamin konsistensi identitas (ID) objek selama pergerakan kamera. Selanjutnya, studi komparatif terhadap varian arsitektur yang lebih kompleks, seperti YOLOv8s (Small) atau YOLOv8m (Medium), perlu dilakukan. Analisis mendalam mengenai trade-off antara peningkatan akurasi deteksi dan latensi inferensi diperlukan untuk menentukan konfigurasi paling optimal, seiring dengan peningkatan kapabilitas komputasi perangkat mobile modern.

DAFTAR PUSTAKA

- [1] R. A. H. Rahman, A. A. Sunarto, dan Asriyanik, "Implementasi YOLOv8 untuk Deteksi dan Klasifikasi Tingkat Kematangan Buah Mangga Berdasarkan Citra Digital," *Jurnal Algoritma*, vol. 20, no. 2, 2023.
- [2] F. Satria dan E. Poerwandono, "Optimasi YOLOv8 untuk Klasifikasi Kematangan Mangga pada Aplikasi Mobile Berbasis Flutter," *Jurnal Teknoinfo*, vol. 18, no. 1, 2024.
- [3] I. A. Prasetya, F. Sukandiansyah, N. A. Fitri, dan S. Adam, "Penerapan Metode Kecerdasan Buatan dalam Sistem Klasifikasi dan Deteksi Mutu Buah Jeruk," *Jurnal Komputer Terapan*, vol. 9, no. 2, 2023.
- [4] A. A. S. Mustofa, R. Wulanningrum, dan J. Sahertian, "Deteksi dan Klasifikasi Kue Tradisional Indonesia Menggunakan YOLOv8," *Jurnal Ilmiah NERO*, vol. 10, no. 1, 2025.
- [5] M. R. Pratama dan J. Aryanto, "Deteksi dan Klasifikasi Kematangan Pisang menggunakan YOLOv8 dan Model Hibrida MobileNetV2-KNN," *Edumatic: Jurnal Pendidikan Informatika*, vol. 9, no. 3, hal. 787-796, 2025.
- [6] Anthony, Herman, dan A. Yulianto, "Pengembangan Sistem Pengenalan Plat Nomor Indonesia Menggunakan YOLOv8 dan Easy OCR," *Jurnal Ilmiah KOMPUTASI*, vol. 23, no. 4, 2024.
- [7] R. Wulanningrum, A. N. Handayani, dan A. P. Wibawa, "Penggunaan Algoritma YOLOv8 dalam Mendeteksi Pejalan Kaki," *Jurnal Simetris*, vol. 14, no. 2, 2023.

- [8] Mihuandayani, W. H. S. Pesik, S. Mamuaya, A. N. Putra, dan E. Mulyaman, "Implementasi YOLOv8 untuk Deteksi dan Hitung Objek Manusia dengan Convolutional Neural Network," *Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 4, hal. 27-35, 2024.
- [9] C. B. Valentino dan D. Hermanto, "Penerapan YOLOv8 untuk Identifikasi Makanan Khas Palembang dan Integrasi Text-to-Speech," *Jurnal Algoritme*, vol. 5, no. 2, hal. 131–139, 2025.
- [10] R. Pusparina A dan R. Rahmadewi, "Deteksi Objek Berbasis YOLOv8 untuk Mendukung Keselamatan Kerja di Lokasi Konstruksi," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 2, hal. 3188-3195, 2025.
- [11] N. Y. D. Rahmawati dan F. P. Rochim, "Chili Ripeness Level Detection with YOLOv8 and Fuzzy Logic for Harvest Decision Making," *Jurnal Teknik Pertanian Lampung*, vol. 14, no. 5, hal. 1807-1818, 2025.