



## *Detection of Papaya Fruit Ripeness Using the YOLOv11s Algorithm Based on Digital Images*

### **Deteksi Tingkat Kematangan Buah Pepaya Menggunakan Algoritma YOLOv11s Berbasis Citra Digital**

**Fikri Maulana<sup>1\*</sup>**

<sup>1</sup>Department of Information Technology, Faculty of Information Technology,  
Nusa Mandiri University, Indonesia

E-Mail: <sup>1</sup>fikrimaulana85@gmail.com

Received Nov 27th 2025; Revised Dec 27th 2025; Accepted Jan 28th 2026; Available Online Jan 31th 2026

Corresponding Author: Fikri Maulana

Copyright ©2026 by Authors, Published by Institut Riset dan Publikasi Indonesia (IRPI)

#### **Abstract**

*Conventional methods for detecting papaya fruit ripeness are often subjective and inconsistent, negatively impacting operational efficiency and standardizing product quality. This study aims to develop an objective automatic detection system using the YOLOv11s Deep Learning algorithm based on digital images. The main contribution of this research lies in the application of the YOLOv11s architecture to address the challenges of visual variability and complex color gradations in papaya fruit. The research methodology involved training the model using a dataset of 3,172 images stratified into three classes: Unripe, Semi-ripe, and Ripe, with a training, validation, and test data split of 70:20:10. The evaluation results show that the model achieved optimal performance with a global Mean Average Precision (mAP@50) value of 0.857. Specifically, the Raw class had the highest detection precision (AP 0.888), followed by the Ripe class (AP 0.875), while the Semi-ripe class achieved AP 0.808. Visual validation on test data confirmed the model's robustness in localizing objects with high precision across various lighting conditions. This study concludes that the implementation of YOLOv11s is effective as a reliable automatic sorting system solution to support Smart Farming.*

**Keywords:** Deep Learning, Object Detection, Papaya Ripeness, Smart Farming, YOLOv11s.

#### **Abstrak**

Metode konvensional untuk deteksi tingkat kematangan buah pepaya seringkali bersifat subjektif dan tidak konsisten, yang berdampak negatif pada efisiensi operasional serta standarisasi kualitas produk. Penelitian ini bertujuan mengembangkan sistem deteksi otomatis yang objektif menggunakan algoritma Deep Learning YOLOv11s berbasis citra digital. Kontribusi utama penelitian ini terletak pada penerapan arsitektur YOLOv11s untuk menangani tantangan variabilitas visual dan gradasi warna kompleks pada buah pepaya. Metodologi penelitian melibatkan pelatihan model menggunakan 3.172 dataset citra yang terstratifikasi menjadi tiga kelas: Mentah, Mengkal, dan Matang, dengan pembagian data latih, validasi, dan uji menggunakan rasio 70:20:10. Hasil evaluasi menunjukkan model mencapai performa optimal dengan nilai Mean Average Precision (mAP@50) global sebesar 0,857. Secara spesifik, kelas Mentah memiliki presisi deteksi tertinggi (AP 0,888), diikuti kelas Matang (AP 0,875), sedangkan kelas Mengkal mencapai AP 0,808. Validasi visual pada data uji mengonfirmasi ketangguhan model dalam melokalisasi objek secara presisi pada berbagai kondisi pencahayaan. Penelitian ini menyimpulkan bahwa implementasi YOLOv11s efektif sebagai solusi sistem sortasi otomatis yang andal guna mendukung Smart Farming.

**Kata Kunci:** Deep Learning, Deteksi Objek, Kematangan Pepaya, Smart Farming, YOLOv11s.

#### **1. PENDAHULUAN**

Buah pepaya merupakan komoditas hortikultura strategis yang memiliki nilai ekonomi tinggi dan permintaan pasar yang stabil [1], [2]. Namun, dalam rantai pasok industri, penentuan tingkat kematangan buah menjadi parameter kritis yang menentukan standar kualitas akhir. Hingga saat ini, proses penyortiran di lapangan masih didominasi oleh metode manual yang mengandalkan pengamatan visual manusia terhadap perubahan fisik kulit buah [3]. Pendekatan ini memiliki kelemahan fundamental, yaitu bersifat subjektif,

tidak konsisten, dan rentan terhadap human error, yang berdampak pada penurunan efisiensi operasional dan kerugian nilai ekonomi produk [4], [5].

Penerapan teknologi computer vision telah menjadi solusi menjanjikan untuk mengatasi subjektivitas tersebut [6]. Namun demikian, tantangan utama dalam deteksi kematangan pepaya terletak pada kompleksitas gradasi warna kulit dan tekstur yang seringkali sulit dibedakan secara cepat oleh model konvensional. *Research gap* yang diidentifikasi dalam penelitian ini adalah belum adanya eksplorasi mendalam mengenai penerapan arsitektur terbaru YOLOv11s pada komoditas pepaya, di mana studi sebelumnya lebih banyak berfokus pada varian lama atau metode CNN standar [7]. YOLOv11s menawarkan pembaruan pada blok ekstraksi fitur yang dirancang untuk meningkatkan efisiensi parameter tanpa mengorbankan akurasi, sebuah aspek yang belum dimanfaatkan optimal dalam studi-studi terdahulu [8].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan sistem deteksi otomatis tingkat kematangan buah pepaya yang objektif dan akurat menggunakan algoritma Deep Learning YOLOv11s. Penelitian ini diharapkan memberikan kontribusi teknis berupa model deteksi yang tangguh terhadap variasi pencahayaan dan latar belakang kompleks, guna mendukung modernisasi sistem sortasi pada Smart Farming.

## 2. TINJAUAN LITERATUR

Penelitian oleh Mahdiyyah et al. (2025) [9] menanggapi kebutuhan efisiensi penyortiran di industri buah melalui pengembangan sistem klasifikasi kualitas lemon otomatis berbasis *Convolutional Neural Network* (CNN). Metodologi penelitian melibatkan pengolahan 2.533 citra sekunder yang dipartisi menjadi 80% data latih dan 20% validasi, serta penerapan teknik augmentasi data untuk memperkaya variabilitas fitur pembelajaran. Hasilnya menunjukkan performa model yang impresif pada fase pelatihan dengan akurasi mendekati 100%, serta kemampuan prediksi yang baik pada pengujian 60 sampel independen. Meskipun demikian, metrik performa menyoroti adanya fluktuasi akurasi validasi yang signifikan pada rentang 88-98%, yang mengindikasikan potensi ketidakstabilan atau overfitting pada model. Selain itu, karakteristik dataset yang didominasi oleh pengambilan gambar di atas permukaan beton seragam berpotensi membatasi robustitas dan generalisasi sistem ketika dihadapkan pada kompleksitas visual latar belakang di lingkungan operasional yang lebih heterogen.

Penelitian oleh Hawibowo dan Muhimmah (2024) [6] merespons tantangan subjektivitas dan inefisiensi waktu pada penilaian visual manual kematangan buah pepaya melalui pengembangan sistem deteksi otomatis berbasis *Convolutional Neural Network* (CNN) yang terintegrasi pada platform Android. Metodologi penelitian memanfaatkan 315 citra primer yang diakuisisi langsung dari perkebunan, yang kemudian melalui tahap pra-pemrosesan *resizing* menjadi 32x32 piksel dengan pembagian data 70% latih, 10% validasi, dan 20% uji. Hasilnya menunjukkan performa model yang signifikan, dengan akurasi pengujian mencapai 96,97% dan keberhasilan klasifikasi 100% pada pengujian sampel acak. Namun, evaluasi reduksi dimensi citra input yang ekstrem ke resolusi 32x32 piksel berpotensi menghilangkan fitur tekstur halus yang krusial untuk deteksi. Selain itu, adanya celah antara akurasi pelatihan (98,63%) dan validasi (90,00%) serta keterbatasan jumlah dataset mengindikasikan perlunya augmentasi data yang lebih masif untuk memitigasi risiko overfitting dan menjamin generalisasi model yang lebih baik di lingkungan nyata.

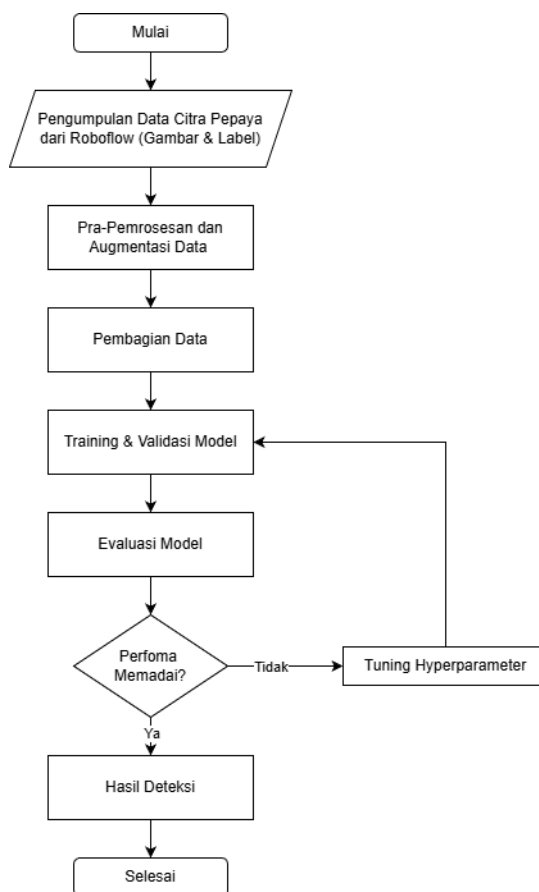
Penelitian oleh Sutrisna et al. (2024) [10] bertujuan mengatasi keterbatasan akurasi dan efisiensi pada identifikasi manual kematangan buah pepaya melalui penerapan algoritma *Convolutional Neural Network* (CNN). Metodologi penelitian memanfaatkan dataset sekunder "*Papaya Classification*" yang terdiri dari 300 citra, yang dipartisi dengan rasio 80:20 untuk data latih dan uji, serta menerapkan augmentasi data pada arsitektur model dengan variasi 1 hingga 3 lapisan konvolusi. Hasil eksperimen menunjukkan bahwa model dengan 3 lapisan konvolusi dan optimizer Nadam mencapai performa optimal dengan akurasi pengujian sebesar 96,63%. Namun, kelemahan fundamental terkait kuantitas dataset yang sangat terbatas untuk standar Deep Learning, yang memicu fenomena overfitting signifikan di mana akurasi validasi cenderung menurun saat akurasi pelatihan meningkat. Kesenjangan ini mengindikasikan bahwa meskipun metrik akurasi tinggi, model memiliki keterbatasan generalisasi yang serius, sehingga ekspansi volume dataset menjadi imperatif untuk menjamin stabilitas prediksi pada implementasi nyata.

Penelitian oleh Saputra et al. (2022) [11] mengkaji permasalahan subjektivitas penentuan kematangan buah melon yang berdampak krusial pada kualitas panen dan umur simpan melalui pendekatan analisis tekstur citra digital. Metodologi penelitian mengkomparasi ekstraksi fitur *Gray-Level Co-occurrence Matrix* (GLCM) pada variasi 4 dan 8 sudut orientasi yang diklasifikasikan menggunakan algoritma *Support Vector Machine* (SVM) dengan empat fungsi kernel berbeda (Linear, RBF, Polynomial, Sigmoid) pada 650 dataset citra. Hasil eksperimen menunjukkan bahwa kombinasi kernel Linear dengan 8 sudut orientasi GLCM menghasilkan performa terbaik dengan akurasi 80%, presisi 81%, dan recall 80%. Namun capaian akurasi 80% masih tergolong moderat untuk standar sistem visi komputer modern, serta adanya disparitas performa yang ekstrem antar fungsi kernel di mana kernel Sigmoid hanya mencapai akurasi 34% mengindikasikan sensitivitas tinggi model terhadap pemilihan parameter. Selain itu, ketergantungan pada pra-pemrosesan

cropping manual membatasi efisiensi dan kemampuan otomatisasi penuh sistem saat diimplementasikan pada kondisi latar belakang perkebunan yang kompleks.

### 3. METODE

Metodologi penelitian ini dilaksanakan melalui serangkaian tahapan terstruktur yang bertujuan untuk menyelesaikan permasalahan penelitian secara komprehensif. Diagram alir yang merepresentasikan langkah-langkah strategis dalam studi ini dapat dilihat pada Gambar 1. Tahapan penelitian dimulai dari akuisisi data sebagai basis informasi, diikuti dengan pra-pemrosesan citra guna optimalisasi kualitas data. Langkah berikutnya mencakup penentuan dan pengaturan parameter model, yang menjadi landasan bagi proses pelatihan algoritma. Penelitian kemudian dilanjutkan ke tahap pengujian dan evaluasi performa model, hingga akhirnya ditarik kesimpulan berdasarkan interpretasi data hasil pengujian tersebut.



**Gambar 1.** Alur Penelitian

#### 3 1. Akuisisi Dataset

Data yang digunakan dalam penelitian ini bersumber dari repositori dataset publik Roboflow. Total dataset yang dikumpulkan berjumlah 3.172 citra digital buah pepaya (*Carica papaya L.*) yang diambil dalam berbagai kondisi lingkungan dan pencahayaan untuk merepresentasikan variabilitas data yang tinggi. Setiap citra dalam dataset telah melalui proses anotasi manual dan diberi label berdasarkan tiga kategori tingkat kematangan, yaitu: Mentah (*Unripe*), Mengkal (*Half-ripe*), dan Matang (*Ripe*) (lihat Tabel 1).
















#### 3 2. Pra-Pemrosesan Data

Selanjutnya adalah tahap pra-pemrosesan, tahap ini dilakukan untuk menstandarisasi format data input agar sesuai dengan arsitektur model YOLOv11 [12]. Langkah pertama adalah penyeragaman dimensi citra (*resizing*) menjadi ukuran 640×640 piksel. Proses ini menerapkan teknik *letterboxing*, yaitu mempertahankan rasio aspek asli citra dengan menambahkan area kosong (*padding*) pada sisi gambar agar tidak terjadi distorsi bentuk buah yang dapat mempengaruhi akurasi deteksi. Selain itu, nilai intensitas piksel pada citra dinormalisasi dari rentang 0 - 255 menjadi 0 - 1 untuk mempercepat proses konvergensi gradien selama pelatihan [13].

Untuk meningkatkan kemampuan generalisasi model dan mencegah terjadinya *overfitting* pada data latih yang terbatas, teknik augmentasi data diterapkan secara otomatis (*on-the-fly*) selama proses pelatihan [14]. Berdasarkan konfigurasi eksperimen, augmentasi diaktifkan (*augment=True*) menggunakan pustaka

bawaan Ultralytics. Teknik utama yang diterapkan meliputi Mosaic Data Augmentation, yaitu penggabungan empat citra latih yang berbeda ke dalam satu bingkai input secara acak. Metode ini memaksa model untuk belajar mendeteksi objek dalam skala yang lebih kecil dan konteks latar belakang yang kompleks [15]. Selain itu, transformasi geometri dan kolorimetri juga diaplikasikan, meliputi pembalikan horizontal (*random horizontal flip*) dengan probabilitas 50%, penyesuaian skala (*scaling*), serta perturbasi warna pada ruang warna HSV (*Hue, Saturation, Value*) untuk mensimulasikan variasi kondisi pencahayaan di lingkungan nyata.

**Table 1.** Contoh Dataset

Tingkat Kematangan		
Mentah	Mengkal	Matang
		
		
		
		
		

### 3.3. Arsitektur Model YOLOv11

Arsitektur yang digunakan dalam penelitian ini adalah varian *Small* dari YOLOv11 (YOLOv11s), sebuah detektor satu tahap (*one-stage detector*) terbaru yang dikembangkan oleh Ultralytics. Dibandingkan dengan versi sebelumnya, YOLOv11s memperkenalkan pembaruan pada blok ekstraksi fitur yang bertujuan untuk meningkatkan efisiensi parameter tanpa mengurangi akurasi model [8]. Secara umum, arsitektur ini terdiri dari tiga komponen utama, yaitu: *Backbone*, *Neck*, dan *Head*. Komponen *Backbone* berfungsi untuk mengekstraksi fitur hirarkis dari citra input dengan menggunakan modul konvolusi yang dimodifikasi (C3k2) serta *Spatial Pyramid Pooling - Fast* (SPPF) untuk menangkap konteks spasial global. Pada bagian *Neck*, digunakan struktur *Path Aggregation Network* (PANet) yang berperan dalam menggabungkan fitur multi-skala, sehingga mampu mendeteksi objek dengan berbagai ukuran, baik kecil maupun besar. Bagian *Head* beroperasi menggunakan mekanisme anchor-free yang memisahkan proses klasifikasi objek dan regresi kotak pembatas (*decoupled head*), memungkinkan pelatihan model berjalan dengan konvergensi yang lebih cepat [16].

Selama proses pelatihan, optimasi bobot model dilakukan dengan meminimalkan *Total Loss Function*. *Loss Function* pada YOLOv11s merupakan gabungan dari tiga komponen utama, yaitu: *Classification Loss*, *Box Regression Loss*, dan *Distribution Focal Loss*. *Total Loss Function* dapat diformulasikan pada persamaan 1.

$$\mathcal{L}_{total} = \lambda_{box} \mathcal{L}_{box} + \lambda_{cls} \mathcal{L}_{cls} + \lambda_{dfl} \mathcal{L}_{dfl} \quad (1)$$

Dimana  $\lambda$  adalah koefisien penyeimbang untuk setiap komponen kerugian. Untuk komponen klasifikasi ( $\mathcal{L}_{cls}$ ), digunakan *Binary Cross Entropy* (BCE) *Loss* untuk mengukur perbedaan distribusi probabilitas prediksi dengan label kematangan pepaya yang sebenarnya, diformulasikan pada persamaan 2.

$$\mathcal{L}_{cls} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (2)$$

Sedangkan untuk regresi kotak pembatas ( $\mathcal{L}_{box}$ ), model menggunakan *Complete Intersection over Union* (CIoU) *Loss*. CIoU dipilih karena kemampuannya mengatasi kelemahan IoU standar dengan memperhitungkan jarak pusat bounding box dan rasio aspek, yang dinyatakan dalam persamaan 3.

$$\mathcal{L}_{IoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \quad (3)$$

Dimana  $\rho^2$  adalah jarak Euclidean antara titik pusat kotak prediksi  $b$  dan *ground truth* ( $b^{gt}$ ),  $c$  adalah panjang diagonal dari kotak penutup terkecil yang melingkupi kedua kotak,  $\alpha$  adalah parameter keseimbangan, dan  $v$  mengukur konsistensi rasio aspek. Penggunaan *Loss Function* ini memastikan bahwa model tidak hanya memprediksi kelas kematangan yang benar, tetapi juga menghasilkan posisi kotak deteksi yang presisi.

### 3.4. Evaluasi Kinerja Model

Untuk mengukur efektivitas model YOLOv11s dalam mendeteksi dan mengklasifikasikan tingkat kematangan pepaya, penelitian ini menggunakan serangkaian metrik evaluasi kuantitatif standar yang diadopsi dari protokol evaluasi Microsoft COCO. Indikator utama yang digunakan meliputi *Precision* (P), *Recall* (R), *F1-Score*, dan *Mean Average Precision* (mAP). Perhitungan metrik ini didasarkan pada elemen *Confusion Matrix* yaitu: *True Positive* (TP) untuk deteksi benar, *False Positive* (FP) untuk deteksi salah, dan *False Negative* (FN) untuk objek yang gagal terdeteksi [17].

#### 3.4.1. Presisi (*Precision*)

Presisi digunakan untuk mengukur tingkat ketepatan model dalam memprediksi kelas positif. Nilai ini merepresentasikan rasio antara jumlah deteksi yang benar (*True Positive*) terhadap total seluruh objek yang diprediksi sebagai positif oleh model. Nilai presisi yang tinggi mengindikasikan bahwa model memiliki tingkat kesalahan deteksi palsu yang rendah. Persamaan presisi ditunjukkan pada persamaan 4.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

Dimana:  $TP$  adalah jumlah objek yang dideteksi dengan benar sesuai label kelasnya dan  $FP$  adalah jumlah objek yang salah dideteksi sebagai kelas tersebut.

#### 3.4.2. Sensitivitas (*Recall*)

Sensitivitas, atau sering disebut *recall*, mengukur kemampuan model dalam menemukan kembali seluruh objek yang relevan yang ada di dalam dataset. Nilai ini dihitung berdasarkan rasio antara deteksi yang benar (*True Positive*) terhadap jumlah total objek asli yang seharusnya terdeteksi (*True Positive* ditambah *False Negative*). Nilai *recall* yang tinggi menunjukkan bahwa model mampu meminimalisir objek yang terlewat. Persamaan *recall* ditunjukkan pada persamaan 5.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

Dimana:  $TP$  adalah jumlah deteksi yang benar dan  $FN$  adalah jumlah objek asli yang gagal dideteksi oleh model.

#### 3.4.3. F1-Score

Mengingat sering terjadinya *trade-off* antara *Precision* dan *Recall*, digunakan metrik F1-Score sebagai rata-rata harmonik dari keduanya. Nilai ini memberikan gambaran keseimbangan performa model, terutama ketika distribusi data antar kelas tidak seimbang. Persamaan F1-Score ditunjukkan pada persamaan 6.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

Dimana: *Precision* adalah nilai ketepatan prediksi model, *Recall* adalah nilai sensitivitas model dan Nilai  $F1$  mendekati 1 mengindikasikan keseimbangan optimal antara presisi dan sensitivitas.

#### 3.4.4. Mean Average Precision (mAP)

Metrik utama yang menjadi acuan performa deteksi objek adalah *Mean Average Precision* (mAP). Nilai ini merepresentasikan rata-rata luas area di bawah kurva *Precision-Recall* (PR Curve) untuk seluruh kelas kategori ( $N$ ). Penelitian ini mengevaluasi dua varian mAP: mAP@50 (mAP pada ambang batas *Intersection over Union* sebesar 0.5) dan mAP@50-95 (rata-rata mAP pada rentang IoU 0.5 hingga 0.95 dengan interval 0.05). Persamaan mAP didefinisikan sebagai persamaan 7.

$$mAP = \frac{1}{N} \sum_{i=1}^N \int_0^1 P(R) dR \quad (7)$$

Dimana:  $N$  adalah jumlah total kelas kategori yang diuji (dalam penelitian ini  $N = 3$ ),  $P(R)$  adalah fungsi Presisi sebagai fungsi dari  $Recall$  dan  $\int_0^1 P(R) dR$  adalah integral yang menghitung luas area di bawah kurva *Precision-Recall* (*Average Precision*).

### 3.5. Validasi

Proses validasi dalam penelitian ini dilakukan menggunakan metode *Hold-out Validation*, di mana performa model dievaluasi secara iteratif terhadap dataset validasi pada setiap akhir epoch pelatihan. Pendekatan ini bertujuan untuk memantau kemampuan generalisasi model dan mendeteksi indikasi *overfitting* atau *underfitting* secara dini [18]. Selama proses validasi, bobot model tidak diperbarui, namun *validation loss* dan akurasi (mAP) dihitung untuk menentukan apakah model mengalami peningkatan konvergensi.

Untuk menjamin efisiensi pelatihan dan mencegah *overtraining*, mekanisme *Early Stopping* diterapkan. Berdasarkan konfigurasi eksperimen, parameter *patience* ditetapkan sebesar 50 epoch. Artinya, proses pelatihan akan dihentikan secara otomatis jika tidak terjadi peningkatan kinerja pada metrik validasi utama selama 50 epoch berturut-turut [19]. Selain itu, sistem akan secara otomatis menyimpan bobot model terbaik (*best weights*) berdasarkan *fitness score* tertinggi, bukan sekadar mengambil bobot dari epoch terakhir. Fungsi *fitness score* ini didefinisikan sebagai kombinasi terbobot dari mAP pada ambang batas IoU yang berbeda untuk memprioritaskan presisi deteksi yang ketat, ditunjukkan pada persamaan 8.

$$Fitness = w_1 \cdot mAP_{0.5} + w_2 \cdot mAP_{0.5:0.95} \quad (8)$$

Dimana  $w_1$  adalah bobot koefisien untuk mAP pada IoU 0.5 yang bernilai 0.1, sedangkan  $w_2$  adalah bobot untuk rata-rata mAP pada rentang IoU 0.5 hingga 0.95 yang bernilai 0.9. Mekanisme pembobotan ini memprioritaskan model yang memiliki ketepatan lokalisasi tinggi, memastikan bahwa sistem tidak hanya mampu mendeteksi keberadaan buah pepaya tetapi juga menentukan posisi kotak pembatas (*bounding box*) dengan presisi yang maksimal.

## 4. HASIL DAN PEMBAHASAN

Seluruh hasil eksperimen yang diperoleh dari implementasi algoritma YOLOv11s dalam mendeteksi tingkat kematangan buah pepaya akan disajikan pada bab ini. Hasil penelitian ini dipaparkan melalui dua pendekatan analisis utama, yaitu analisis kuantitatif dan analisis kualitatif. Analisis kuantitatif difokuskan pada evaluasi metrik kinerja model selama proses pelatihan dan validasi, yang mencakup pergerakan kurva *Loss Function*, serta metrik evaluasi seperti *Precision*, *Recall*, dan nilai *Mean Average Precision* (mAP) [20]. Sementara itu, analisis kualitatif bertujuan untuk memverifikasi kemampuan model dalam melakukan lokalisasi dan klasifikasi objek pada citra uji yang belum pernah dilihat sebelumnya (*unseen data*). Selain itu, dilakukan pembahasan mendalam mengenai interpretasi fenomena yang muncul pada *Confusion Matrix* untuk memahami pola kesalahan deteksi antar kelas, yang selanjutnya mengonfirmasi efektivitas model yang diusulkan.

### 4.1. Akuisisi Data

Tahap awal eksperimen ini melibatkan konsolidasi data citra buah pepaya yang siap digunakan dalam pelatihan model. Proses pengumpulan data dilakukan melalui platform manajemen dataset Roboflow, yang menghasilkan total dataset final sebanyak 3.172 citra. Dataset ini telah melalui tahap validasi kualitas untuk memastikan setiap citra dilengkapi dengan anotasi (*bounding box*) yang akurat, sesuai dengan tiga kategori kelas target: Mentah (*Unripe*), Mengkal (*Half-ripe*), dan Matang (*Ripe*).

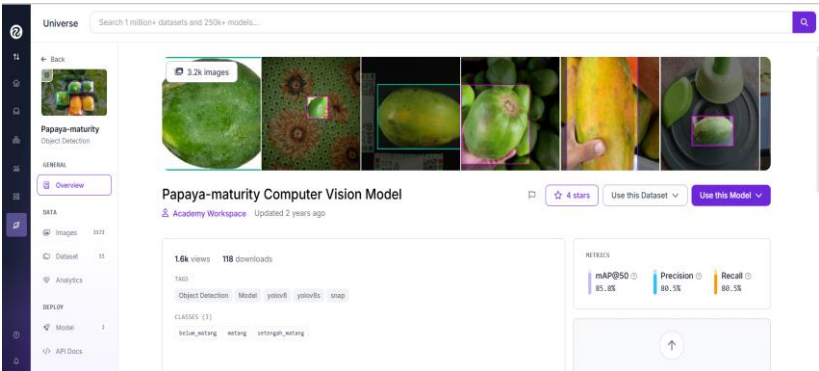
Visualisasi statistik dan manajemen dataset yang ditampilkan pada antarmuka Roboflow dapat dilihat pada Gambar 1. Platform ini memberikan transparansi terkait distribusi kelas dan kesehatan dataset. Seperti yang ditunjukkan pada *dashboard*, data telah dibagi secara otomatis menjadi tiga sub-himpunan, yaitu 2.221 citra untuk data latih (*training set*), 636 citra untuk data validasi (*validation set*), dan 315 citra untuk data uji (*testing set*). Ketersediaan jumlah data yang cukup besar ini menjadi faktor penting bagi model YOLOv11s dalam mempelajari fitur-fitur variatif buah pepaya dan mencapai tingkat konvergensi yang optimal [21].

### 4.2. Pra-Pemrosesan Data

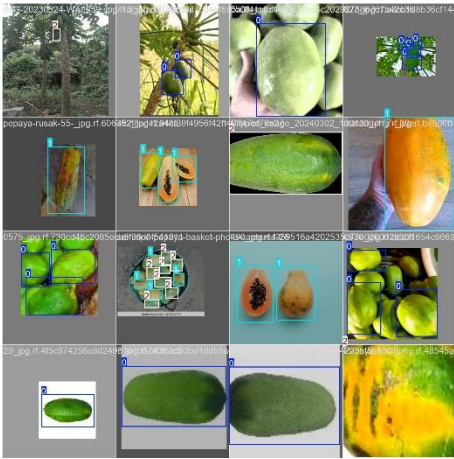
Implementasi tahapan pra-pemrosesan data yang telah dirancang pada metodologi berhasil dieksekusi oleh sistem sesaat sebelum proses pelatihan dimulai. Keberhasilan proses ini dibuktikan melalui visualisasi batch data latih pertama yang ditampilkan pada Gambar 2. Berdasarkan hasil tersebut, terlihat bahwa teknik augmentasi Mosaic bekerja secara efektif dengan menggabungkan empat potongan citra pepaya yang berbeda ke dalam satu bingkai input berukuran 640×640 piksel.

Visualisasi ini memperlihatkan adanya peningkatan kompleksitas data yang signifikan dibandingkan citra asli. Teramati variasi skala objek yang beragam, di mana sebagian buah pepaya tampak diperbesar untuk menonjolkan tekstur kulit, sementara bagian lain diperkecil untuk simulasi deteksi jarak jauh. Selain

itu, transformasi warna dan pencahayaan juga terlihat jelas, menciptakan variabilitas visual yang kaya. Kondisi data hasil pra-pemrosesan yang heterogen ini sangat krusial karena memaksa model YOLOv11s untuk tidak sekadar menghafal bentuk standar pepaya, melainkan mempelajari fitur-fitur kunci yang invarian terhadap perubahan posisi, potongan, maupun kondisi cahaya yang ekstrem.



Gambar 1. Statistik dataset dan distribusi citra pada platform Roboflow



Gambar 2. Hasil pra-pemrosesan data latih menggunakan teknik Mosaic Augmentation.

4.3. Arsitektur YOLOV11s

Berdasarkan inisialisasi pelatihan menggunakan bobot pra-latih, model berhasil dibangun dengan struktur jaringan yang efisien. Ringkasan arsitektur model yang dihasilkan selama proses pelatihan menunjukkan bahwa YOLOv11s memiliki kompleksitas komputasi yang moderat, menjadikannya sangat layak untuk diimplementasikan pada perangkat dengan sumber daya terbatas. Model ini terdiri dari ratusan lapisan yang terstruktur dalam blok *Backbone*, *Neck*, dan *Head*. Total parameter yang dapat dilatih dan operasi *floating-point* (GFLOPs) menunjukkan bahwa model memiliki kapasitas yang cukup untuk mempelajari fitur kompleks dari kulit pepaya tanpa membebani memori GPU secara berlebihan. Untuk memastikan reproduisibilitas hasil, konfigurasi *hyperparameter* final yang digunakan selama proses pelatihan dirangkum dalam Tabel 2.

Table 2. Konfigurasi Hyperparameter Pelatihan

Parameter	Nilai Konfigurasi
Epochs	10
Batch Size	16
Optimizer	Stochastic Gradient Descent (SGD)
Initial Learning Rate	0.01
Momentum	0.937
Weight Decay	0.0005
Device	GPU (CUDA)

Konfigurasi *hyperparameter* yang disajikan pada Tabel 2 dirancang secara strategis untuk mengoptimalkan keseimbangan antara kecepatan konvergensi dan kemampuan generalisasi model YOLOv11s [22]. Penetapan batas maksimum pelatihan sebanyak 100 epoch memberikan durasi yang

memadai bagi jaringan untuk mengekstraksi fitur visual pepaya secara mendalam, namun mekanisme *Early Stopping* dengan parameter *patience* sebesar 50 epoch diterapkan sebagai katup pengaman untuk mencegah *overfitting* dengan menghentikan iterasi apabila stagnasi kinerja terdeteksi pada data validasi. Proses optimasi bobot dikendalikan oleh algoritma *Stochastic Gradient Descent* (SGD) dengan momentum 0.937 dan laju pembelajaran awal 0.01, yang bertujuan untuk mempercepat pergerakan gradien menuju titik optimal global sekaligus meredam fluktuasi yang tidak diinginkan. Selain itu, stabilitas arsitektur diperkuat melalui penerapan regularisasi *weight decay* sebesar 0.0005 untuk membatasi kompleksitas bobot jaringan, didukung oleh eksekusi komputasi pada akselerator GPU dengan ukuran batch 16 yang memastikan efisiensi penggunaan memori selama proses propagasi balik.

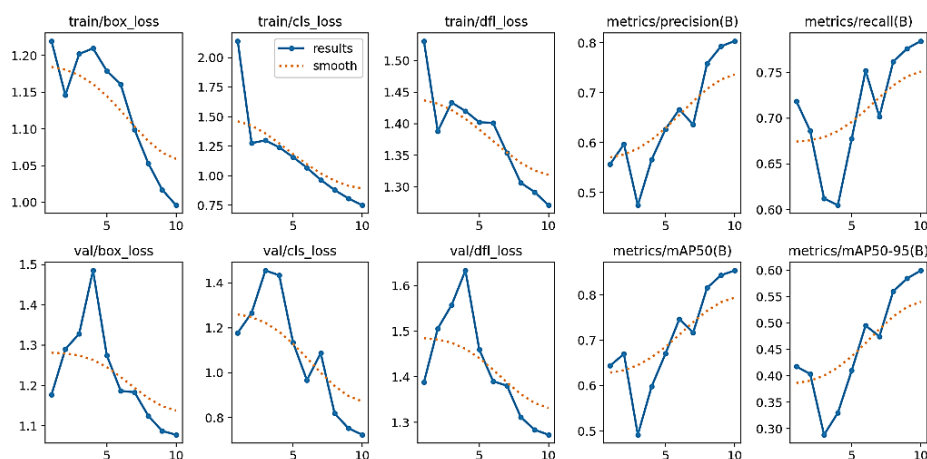
#### 4.4. Evaluasi Kinerja Model

Evaluasi kinerja model dilakukan secara menyeluruh untuk memvalidasi keandalan algoritma YOLOv11s dalam mengklasifikasikan tingkat kematangan pepaya. Analisis ini didasarkan pada metrik *Precision*, *Recall*, *F1-Score*, dan *Mean Average Precision* (mAP) yang dihasilkan setelah proses pelatihan selesai, serta visualisasi kesalahan prediksi melalui *confusion matrix* [23].

##### 4.4.1. Analisis Metrik Pelatihan (*Training Metrics*)

Pembelajaran model selama 100 epoch divisualisasikan melalui grafik *Loss Function* dan metrik performa pada Gambar 3. Berdasarkan grafik tersebut, terlihat tren penurunan yang konsisten pada ketiga komponen utama (*box\_loss*, *cls\_loss*, dan *dfl\_loss*) baik pada data latih maupun validasi. Penurunan ini mengindikasikan bahwa model berhasil meminimalkan kesalahan prediksi koordinat dan klasifikasi seiring berjalannya waktu tanpa menunjukkan gejala *overfitting* yang signifikan.

Metrik *precision* dan *recall* menunjukkan tren peningkatan yang tajam pada 50 epoch awal dan mulai stabil mendekati akhir pelatihan. Nilai mAP@50 mencapai puncaknya di kisaran 0.857, yang menegaskan bahwa model mampu menjaga keseimbangan yang baik antara ketepatan prediksi positif dan sensitivitas terhadap keberadaan objek.



**Gambar 3.** Grafik metrik pelatihan (*Loss*, *Precision*, *Recall*, dan *mAP*)

##### 4.4.2. Analisis Kurva Precision-Recall (PR Curve)

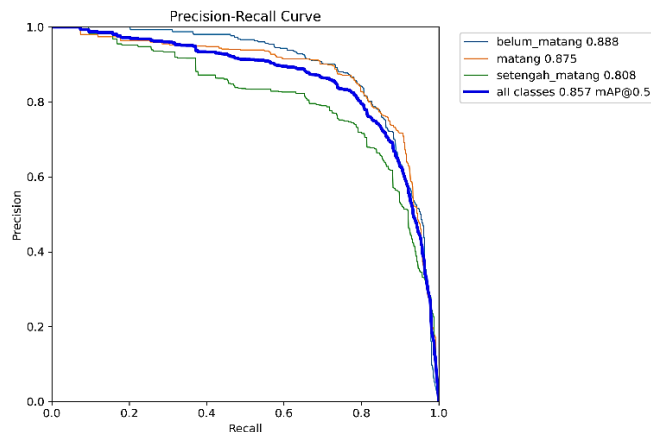
Untuk mengevaluasi kinerja deteksi pada setiap kelas secara spesifik, digunakan kurva *Precision-Recall* (PR Curve) sebagaimana ditampilkan pada Gambar 4. Kurva ini menggambarkan *trade-off* antara *precision* dan *recall* pada berbagai ambang batas kepercayaan.

Berdasarkan grafik, model mencatatkan kinerja rata-rata global (*all classes*) mAP@0.5 sebesar 0.857. Jika dirinci per kategori (lihat Tabel 3).

**Table 3.** Ringkasan Nilai Average Precision (AP) per Kelas

Kelas Kategori	Average Precision (AP) @0.5
Belum Matang	0.888
Matang	0.875
Setengah Matang	0.808
Rata-rata (mAP)	0.857

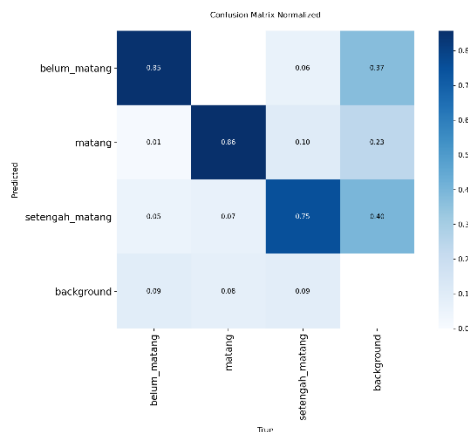




**Gambar 4.** Kurva *Precision-Recall* untuk setiap kelas tingkat kematangan pepaya.

#### 4.4.3. Analisis Confusion Matrix

Analisis mendalam mengenai kesalahan klasifikasi dilakukan menggunakan *Normalized Confusion Matrix* pada Gambar 5. Matriks ini memetakan prediksi model (sumbu Y) terhadap label sebenarnya (sumbu X).



**Gambar 5.** *Normalized Confusion Matrix* yang menunjukkan distribusi akurasi prediksi.

Secara diagonal, model menunjukkan akurasi yang tinggi: 85% untuk kelas Belum Matang, 86% untuk kelas Matang, dan 75% untuk kelas Setengah Matang. Kesalahan prediksi terbesar terjadi pada kelas "Setengah Matang", di mana terdapat *confusion* sebesar 10% data Setengah Matang yang diprediksi sebagai Matang, dan sebaliknya 7% data Matang diprediksi sebagai Setengah Matang. Fenomena ini mengonfirmasi hipotesis bahwa gradasi warna pada fase transisi merupakan fitur yang paling sulit diekstraksi. Selain itu, terdapat kesalahan deteksi latar belakang (background FN) sekitar 9%, yang berarti ada sebagian kecil objek buah yang gagal terdeteksi sama sekali, kemungkinan disebabkan oleh oklusi atau pencahayaan yang minim.

#### 4.5. Validasi Hasil Prediksi

Tahap akhir dari evaluasi model adalah validasi terpadu yang menggabungkan verifikasi statistik dan pembuktian visual. Secara statistik, model dinyatakan valid karena telah melewati mekanisme *Early Stopping* dengan *fitness score* yang optimal, serta menunjukkan konvergensi *Loss Function* yang stabil antara data latih dan data validasi [24]. Hal ini mengonfirmasi bahwa model memiliki kemampuan generalisasi yang baik dan tidak sekadar menghafal data latih. Untuk membuktikan validitas tersebut secara empiris, dilakukan pengujian inferensi pada data validasi yang merepresentasikan berbagai skenario lingkungan. Hasil prediksi visual disajikan pada Gambar 6 dan Gambar 7.

Pada Gambar 6, pengujian difokuskan pada objek tunggal dengan jarak pandang dekat (*close-up*). Hasil visual memperlihatkan bahwa model mampu membentuk kotak pembatas (*bounding box*) yang sangat presisi melingkupi kontur buah pepaya. Klasifikasi kelas juga terlihat akurat, model berhasil membedakan fase "Belum Matang" (hijau dominan) dan "Setengah Matang" (hijau kekuningan) dengan *confidence score* rata-rata di atas 0.80.

Selanjutnya, ketangguhan model diuji pada skenario yang lebih kompleks seperti ditampilkan pada Gambar 7. Pada skenario ini, model dihadapkan pada tantangan lingkungan alami, seperti buah yang

menggantung di pohon dengan latar belakang dedaunan rimbun, kondisi pencahayaan yang tidak merata, serta objek yang saling menumpuk (*occlusion*).



**Gambar 6.** Hasil deteksi pada skenario objek tunggal dengan berbagai variasi tingkat kematangan.



**Gambar 7.** Hasil deteksi pada skenario lingkungan kompleks, meliputi latar belakang dedaunan, buah di pohon, dan buah yang telah dipotong.

Secara keseluruhan, hasil validasi visual ini selaras dengan metrik mAP 0.857, membuktikan bahwa model yang dikembangkan siap untuk diimplementasikan pada sistem sortasi pertanian cerdas.

#### 4.6. Diskusi

Penelitian ini menawarkan keunggulan signifikan dalam aspek stabilitas model dan kemampuan generalisasi dibandingkan studi terdahulu yang menggunakan arsitektur CNN standar. Berbeda dengan penelitian Sutrisna et al. [10] dan Mahdiyyah et al. [9] yang melaporkan indikasi overfitting tinggi ditandai dengan fluktuasi akurasi validasi dan kesenjangan performa akibat dataset yang terbatas model YOLOv11s dalam studi ini menunjukkan konvergensi yang konsisten antara data latih dan validasi. Keberhasilan ini didukung secara empiris oleh penggunaan volume dataset yang jauh lebih besar (3.172 citra) serta penerapan strategi *Mosaic Augmentation*, yang terbukti efektif memperkaya variabilitas fitur pembelajaran sehingga model lebih adaptif terhadap data baru [25].

Dari segi ketangguhan teknis, algoritma YOLOv11s terbukti mampu mengatasi keterbatasan metode klasifikasi konvensional maupun pendekatan CNN beresolusi rendah. Jika dibandingkan dengan penelitian Saputra et al. [11] yang masih bergantung pada pra-pemrosesan cropping manual dan sensitif terhadap pemilihan kernel SVM, sistem yang diusulkan mampu melakukan deteksi objek secara otomatis (*end-to-end*) pada latar belakang lingkungan yang kompleks tanpa intervensi manusia. Selain itu, penggunaan resolusi input standar 640x640 piksel memperbaiki kelemahan fundamental pada studi Hawibowo dan Muhimmah [6], di mana reduksi dimensi citra yang ekstrem ke 32x32 piksel menghilangkan detail tekstur halus kulit pepaya yang krusial untuk membedakan fase transisi kematangan secara akurat.

Secara keseluruhan, penelitian ini memberikan kontribusi nyata sebagai solusi sistem sortasi yang andal dan objektif guna mendukung ekosistem Smart Farming. Capaian nilai mAP sebesar 0.857 menegaskan bahwa model ini tidak hanya valid secara akademis, tetapi juga memiliki manfaat praktis yang tinggi untuk menggantikan metode penyortiran manual yang subjektif dan rentan kesalahan. Implementasi YOLOv11s yang efisien menjadikannya aset strategis dalam meningkatkan standar kualitas produk pascapanen, meminimalkan kerugian ekonomi akibat inkonsistensi sortir, serta menjamin mutu produk yang seragam yang sulit dicapai oleh pengamatan visual manusia semata.

## 5. KESIMPULAN

Penelitian ini berhasil membuktikan bahwa implementasi algoritma Deep Learning YOLOv11s efektif sebagai solusi objektif untuk deteksi tingkat kematangan buah pepaya, dengan capaian performa presisi global (mAP@50) sebesar 0,857. Validasi pada data uji memperkuat temuan statistik tersebut, di mana model menunjukkan ketangguhan yang signifikan dalam melokalisasi objek secara presisi pada berbagai kondisi pencahayaan dan latar belakang lingkungan yang kompleks, serta menunjukkan stabilitas konvergensi yang menjamin model terhindar dari overfitting. Kontribusi ilmiah utama dari penelitian ini terletak pada pembuktian kemampuan arsitektur YOLOv11s dalam menangani tantangan variabilitas visual yang sering menjadi kendala pada metode sortasi manual maupun model komputasi terdahulu.

Meskipun model menunjukkan performa superior pada kelas "Mentah" dan "Matang", keterbatasan penelitian teridentifikasi pada deteksi fase transisi "Mengkal" (*Half-ripe*). Analisis kesalahan menunjukkan adanya tingkat ambiguitas prediksi sebesar 10% terhadap kelas "Matang", serta kegagalan deteksi latar belakang (*background false negative*) sebesar 9% pada kondisi oklusi tertentu. Hal ini mengindikasikan bahwa fitur visual pada fase peralihan warna masih memerlukan ekstraksi yang lebih mendalam.

Berdasarkan temuan tersebut, arah penelitian lanjutan disarankan untuk fokus pada dua aspek strategis, yang pertama yaitu eksplorasi teknik *attention mechanism* atau penambahan dataset varietas pepaya yang lebih beragam untuk meningkatkan sensitivitas pada fase transisi, lalu yang kedua yaitu pengembangan implementasi model ke dalam perangkat *edge computing* (IoT) atau aplikasi mobile berbasis Android. Langkah ini krusial untuk mentransformasi model teoretis ini menjadi alat aplikasi praktis yang dapat digunakan secara *real-time* oleh petani di lapangan.

## REFERENCES

- [1] A. M. Hasibuan, B. Sugiharto, N. F. Hayati, T. A. Dewita, and T. Bayati, 'Meningkatkan Kesejahteraan Petani: Menuju Sektor Pertanian yang Tangguh dan Berdaya Saing di Indonesia', *Journal of Law, Education and Business*, vol. 2, no. 2, pp. 1365–1371, Sep. 2024, doi: 10.57235/jleb.v2i2.3130.
- [2] E. D. Wahyuni, 'Strategi Memanfaatkan Peluang Pasar Produk Pertanian Dalam Perdagangan Internasional', *ekobistek*, pp. 57–64, Jul. 2021, doi: 10.35134/ekobistek.v10i1.84.
- [3] R. Kurniawan, 'Klasifikasi Tingkat Kematangan Buah Pepaya Berdasarkan Warna Kulit Menggunakan Sensor Warna TCS3200', *JICTEE*, vol. 4, no. 1, p. 1, Mar. 2023, doi: 10.33365/jictee.v4i1.2692.
- [4] J. Jusrawati, A. Futri, and A. B. Kaswar, 'Klasifikasi Tingkat Kematangan Buah Pisang Dalam Ruang Warna RGB Menggunakan Jaringan Syaraf Tiruan (JST)', *JESSI*, vol. 2, no. 1, p. 49, Nov. 2021, doi: 10.26858/jessi.v2i1.20327.
- [5] Departemen Teknik Mesin dan Biosistem, Fakultas Teknologi Pertanian, Institut Pertanian Bogor, Kampus IPB Dramaga, Bogor 16680, U. Ahmad, . Sabihah, and Program Studi Teknik Mesin dan Biosistem, Fakultas Teknologi Pertanian, Institut Pertanian Bogor, Kampus IPB Dramaga, Bogor 16680, 'Prediction of Ripeness Parameters of Melon Fruit Using Near Infra-red Spectroscopy', *JIPi*, vol. 23, no. 3, pp. 183–189, Dec. 2018, doi: 10.18343/jipi.23.3.183.
- [6] M. S. Hawibowo and I. Muhimmmah, 'Aplikasi Pendeteksi Tingkat Kematangan Pepaya menggunakan Metode Convolutional Neural Network (CNN) Berbasis Android', *JEPIN*, vol. 10, no. 1, p. 162, Apr. 2024, doi: 10.26418/jp.v10i1.77819.
- [7] A. Kusuma, A. R. S. Nurrohman, K. T. Anggoro, and R. Susun, 'Implementasi Algoritma Yolo Dalam Pendeteksian Tingkat Kematangan Pada Buah Pepaya', vol. 1, no. 1, 2023.
- [8] R. Khanam and M. Hussain, 'YOLOv11: An Overview of the Key Architectural Enhancements', Oct. 23, 2024, *arXiv: arXiv:2410.17725*. doi: 10.48550/arXiv.2410.17725.
- [9] S. Mahdiyyah *et al.*, 'Implementasi Object Detection untuk Deteksi Kualitas pada Buah Lemon dengan CNN', *JRAMI*, vol. 6, no. 01, pp. 215–223, Jan. 2025, doi: 10.30998/jrami.v6i01.13417.
- [10] N. P. Sutrisna *et al.*, 'Deteksi Tingkat Kematangan Buah Pepaya menggunakan Model Convolutional Neural Network', *JTIK*, vol. 11, no. 3, pp. 569–578, Jul. 2024, doi: 10.25126/jtiik.938119.
- [11] R. A. Saputra, D. Puspitasari, and T. Baidawi, 'Deteksi Kematangan Buah Melon dengan Algoritma Support Vector Machine Berbasis Ekstraksi Fitur GLCM', vol. 4, no. 2, 2022.
- [12] S. D. Vanesa, 'Sistem Deteksi Jenis Burung Menggunakan Implementasi Yolov1', vol. 9.
- [13] A. Priadana, 'Analisis Pengaruh Ukuran Citra Hasil Resizing Terhadap Jumlah Keypoint Hasil Ekstraksi Ciri Pada Metode Sift Dan Surf'.
- [14] M. A. Al-Fahrezi, 'Pengaruh Augmentasi Data Terhadap Akurasi Pelatihan Model CNN untuk Klasifikasi Jenis Ikan', *jitsi*, vol. 6, no. 2, pp. 177–185, Jun. 2025, doi: 10.62527/jitsi.6.2.471.
- [15] M. Yang, L. S. Ewe, W. K. Yew, S. Deng, and S. K. Tiong, 'A Survey of Data Augmentation Techniques for Traffic Visual Elements', *Sensors*, vol. 25, no. 21, p. 6672, Nov. 2025, doi: 10.3390/s25216672.

- 
- [16] A. I. Pradana, H. Harsanto, and W. Wijiyanto, 'Deteksi Rambu Lalu Lintas Real-Time di Indonesia dengan Penerapan YOLOv11: Solusi Untuk Keamanan Berkendara', *Jurnal Algoritma*, vol. 21, no. 2, pp. 145–155, Nov. 2024, doi: 10.33364/algoritma/v.21-2.2106.
  - [17] F. Gusnanto, N. Rahaningsih, R. Danar Dana, and M. Mulyawan, 'Optimasi Model Klasifikasi Tingkat Kematangan Buah Mangga Dengan Metode YOLO11', *jati*, vol. 9, no. 1, pp. 1773–1780, Jan. 2025, doi: 10.36040/jati.v9i1.12591.
  - [18] F. Tempola, R. Rosihan, and R. Adawiyah, 'Holdout Validation for Comparison Classification Naïve Bayes and KNN of Recipient Kartu Indonesia Pintar', *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 1125, no. 1, p. 012041, May 2021, doi: 10.1088/1757-899X/1125/1/012041.
  - [19] A. R. Muhammad, H. P. Utomo, P. Hidayatullah, and N. Syakrani, 'Early Stopping Effectiveness for YOLOv4', *J. Inf. Syst. Eng. Bus. Intell.*, vol. 8, no. 1, pp. 11–20, Apr. 2022, doi: 10.20473/jisebi.8.1.11-20.
  - [20] R. Akyas Hifdzi Rahman, A. Adi Sunarto, and A. Asriyanik, 'Penerapan You Only Look Once (Yolo) V8 Untuk Deteksi Tingkat Kematangan Buah Manggis', *jati*, vol. 8, no. 5, pp. 10566–10571, Sep. 2024, doi: 10.36040/jati.v8i5.10979.
  - [21] E. Verdiansyah, F. Nurdiansyah, and I. Istiadi, 'Papaya Type Classification Using YOLOv8', *J. Tek. Inform. (JUTIF)*, vol. 5, no. 5, pp. 1287–1297, Oct. 2024, doi: 10.52436/1.jutif.2024.5.5.2336.
  - [22] I. A. Zulkarnain, 'Yolov11 Optimization Through Hyperparameter Tuning and Data Augmentation to Improve Vehicle Detection Accuracy at Night', vol. 5, 2025.
  - [23] N. Sary, H. H. S. Pasaribu, R. J. Situmeang, and R. Darmawan, 'Implementasi Algoritma YOLOv11 Dan Roboflow Untuk Deteksi Tingkat Kematangan Anggur Berbasis Web', vol. 9, 2025.
  - [24] Z. Zulkifli and R. Fajri, 'Klasifikasi Tingkat Kematangan Buah Strawberry Menggunakan Algoritma Logistic Regression', *dsi*, vol. 4, no. 2, pp. 50–59, Dec. 2024, doi: 10.47709/dsi.v4i2.4850.
  - [25] A. Yudhana, E. P. Silmina, and Sunardi, 'Deteksi Kesegaran Daging Sapi Menggunakan Augmentasi Data Mosaic pada Model YOLOv5sM', *JRST*, pp. 63–71, Apr. 2025, doi: 10.30595/jrst.v9i1.24990.