Analisis Lightweight Yolo pada Deteksi Mangga Segar dan Busuk Berbasis Deep Learning

Deep Learning-Based Analysis of Lightweight YOLO for Fresh and Rotten Mango Detection

Mi'raj Fuadi[1*], Yuhendra AP[1], Hary Kurniawan[1], Agriananta Fahmi Hidayat[1]
[1]Program Studi Teknik Pertanian, Fakultas Teknologi Pangan dan Agroindustri, Universitas Mataram
*Corresponding author: Mi'raj Fuadi, mirajfuadi@unram.ac.id

ABSTRAK

Buah mangga merupakan komoditas hortikultura tropis bernilai ekonomi tinggi, namun kualitasnya dapat menurun akibat tercampurnya buah segar dan buah busuk selama proses distribusi dan pemasaran. Proses sortasi yang masih dilakukan secara manual berpotensi menimbulkan kesalahan karena dipengaruhi subjektivitas, kelelahan, dan keterbatasan manusia. Penelitian ini bertujuan untuk menganalisis performa beberapa model lightweight YOLO dalam mendeteksi mangga segar dan mangga busuk berbasis deep learning. Model yang digunakan meliputi YOLOv5n, YOLOv8n, dan YOLOv11n. Dataset diperoleh dari platform Kaggle yang terdiri atas dua kelas objek, yaitu mangga segar sebanyak 1.813 gambar dan mangga busuk sebanyak 2.247 gambar. Penelitian dilakukan menggunakan metode eksperimen komparatif melalui tahapan preprocessing, augmentasi data, pembagian dataset menggunakan stratified split dengan rasio 80% data training, 10% validation, dan 10% testing, serta pelatihan model selama 100 epoch menggunakan GPU NVIDIA A100 pada Google Colab. Evaluasi performa dilakukan menggunakan precision, recall, F1-Score, mAP@50, dan mAP@50:95. Hasil penelitian menunjukkan bahwa ketiga model menghasilkan performa deteksi yang sangat baik dengan nilai evaluasi di atas 0,95. YOLOv5n memperoleh precision tertinggi sebesar 0,955, recall tertinggi sebesar 0,966, dan F1-Score tertinggi sebesar 0,960. Sementara itu, YOLOv11n memperoleh nilai mAP@50 tertinggi sebesar 0,987 dan mAP@50:95 tertinggi sebesar 0,959, serta memiliki nilai GFLOPs paling rendah sebesar 6,3. Berdasarkan hasil tersebut, YOLOv11n direkomendasikan sebagai model terbaik karena memiliki kemampuan lokalisasi objek yang lebih baik dan efisiensi komputasi yang lebih tinggi untuk penerapan real-time. Penelitian ini menunjukkan bahwa model lightweight YOLO berpotensi diterapkan pada sistem sortasi otomatis buah mangga berbasis computer vision.

Keywords: Computer Vision; Deep Learning; Lightweight YOLO; Mangga; Object Detection

ABSTRACT

Mango is a tropical horticultural commodity with high economic value; however, its quality may decline when fresh and rotten fruits are mixed during distribution and marketing. Manual sorting is still commonly used, but it is prone to errors caused by subjectivity, fatigue, and human limitations. This study aims to analyze the performance of lightweight YOLO models for detecting fresh and rotten mangoes based on deep learning. The models evaluated were YOLOv5n, YOLOv8n, and YOLOv11n. The dataset was obtained from Kaggle and consisted of two classes: 1,813 fresh mango images and 2,247 rotten mango images. The research used a comparative experimental method involving preprocessing, data augmentation, stratified dataset splitting with an 80:10:10 ratio for training, validation, and testing, and model training for 100 epochs using an NVIDIA A100 GPU on Google Colab. Model performance was evaluated using precision, recall, F1-score, mAP@50, and mAP@50:95. The results showed that all models achieved high detection performance, with evaluation scores above 0.95. YOLOv5n obtained the highest precision, recall, and F1-score, with values of 0.955, 0.966, and 0.960, respectively. Meanwhile, YOLOv11n achieved the highest mAP@50 of 0.987 and mAP@50:95 of 0.959, with the lowest computational cost of 6.3 GFLOPs. Therefore, YOLOv11n is recommended as the best model for real-time mango quality detection due to its superior localization performance and computational efficiency.

Keywords: *computer vision; deep learning; lightweight YOLO; mango; object detection*

ARTICLE IDENTIFIERS

ORIGINAL ARTICLE
Volume 9 No. 10, 2026
Pages 4842-4851
DOI: 10.56338/jks.v9i10.12159

ARTICLE HISTORY

Received: 29 August 2026
Revised: 30 September 2026
Accepted: 16 October 2026

CC BY-SA 4.0

PENDAHULUAN

Buah mangga merupakan salah satu komoditas hortikultura tropis yang memiliki nilai ekonomi tinggi di berbagai negara, termasuk Indonesia. Produksi mangga terus meningkat seiring dengan tingginya permintaan pasar domestik maupun ekspor. Namun, kualitas buah mangga sangat menentukan nilai jual dan kepuasan konsumen. Salah satu permasalahan utama dalam distribusi dan pemasaran mangga adalah adanya buah busuk yang tercampur dengan buah segar sehingga menurunkan kualitas produk secara keseluruhan. Proses sortasi mangga di lapangan maupun gudang penyimpanan umumnya masih dilakukan secara manual berdasarkan pengamatan visual manusia, sehingga berpotensi menimbulkan kesalahan akibat faktor kelelahan, subjektivitas, dan keterbatasan kecepatan kerja (Zhong et al., 2024).

Perkembangan teknologi kecerdasan buatan (Artificial Intelligence) dan *Deep Learning* memberikan solusi baru dalam bidang identifikasi kualitas buah secara otomatis. Salah satu cabang *Deep Learning* yang banyak digunakan pada pengolahan citra adalah object detection. Teknologi ini mampu mendeteksi sekaligus mengenali objek secara real-time dengan tingkat akurasi yang tinggi. Dalam bidang pertanian modern, implementasi *computer vision* dan *Deep Learning* telah banyak diterapkan untuk klasifikasi buah, deteksi penyakit tanaman, identifikasi kematangan buah, hingga sistem panen otomatis berbasis robotik (Badgujar et al., 2024).

Salah satu algoritma object detection yang populer adalah YOLO (You Only Look Once). Algoritma YOLO dikenal memiliki keunggulan pada kecepatan deteksi dan performa yang baik dibandingkan metode deteksi objek dua tahap (two-stage detector). Seiring perkembangannya, YOLO hadir dalam berbagai varian ringan (lightweight) seperti YOLOv5n, YOLOv8n, dan YOLOX-Nano yang dirancang agar dapat dijalankan pada perangkat dengan sumber daya terbatas tanpa mengorbankan performa secara signifikan. Model lightweight menjadi penting karena implementasi sistem deteksi buah di lapangan membutuhkan komputasi cepat dan efisien, terutama pada perangkat *edge computing* atau perangkat bergerak (Ge et al., 2021).

Penelitian mengenai deteksi buah menggunakan YOLO telah banyak dilakukan. Zhong et al. (Zhong et al., 2024) mengembangkan model Light-YOLO untuk deteksi mangga di lingkungan alami dengan parameter yang lebih ringan namun tetap memiliki performa tinggi. Hasil penelitian menunjukkan bahwa model lightweight mampu meningkatkan efisiensi komputasi sekaligus mempertahankan akurasi deteksi. Selain itu, penelitian oleh Nuha dan Alexandro (Nuha & Alexandro H., 2022) mengenai pengenalan kesegaran buah mangga menggunakan YOLO menunjukkan bahwa algoritma ini mampu membedakan kondisi mangga segar dan busuk secara otomatis berdasarkan citra digital.

Meskipun berbagai penelitian terkait deteksi buah telah berkembang pesat, studi mengenai analisis performa model lightweight YOLO khusus untuk mendeteksi mangga segar dan busuk masih relatif terbatas. Sebagian besar penelitian berfokus pada deteksi objek buah secara umum atau identifikasi tingkat kematangan, sedangkan analisis terhadap efisiensi model ringan untuk klasifikasi kondisi kesegaran buah belum banyak dibahas secara mendalam. Oleh karena itu, penelitian ini dilakukan untuk menganalisis performa beberapa varian lightweight YOLO dalam mendeteksi mangga segar dan busuk berbasis *Deep Learning*. Analisis dilakukan dengan mengevaluasi parameter performa seperti precision, recall, F1-score, dan mean Average Precision (mAP), sehingga dapat diketahui model yang paling efektif untuk diterapkan pada sistem deteksi kualitas buah secara *real-time*.

METODE

Penelitian ini menggunakan metode eksperimen komparatif (*comparative experimental method*) dengan pendekatan kuantitatif. Penelitian dilakukan untuk menganalisis dan membandingkan performa beberapa varian lightweight YOLO dalam mendeteksi mangga segar dan mangga busuk berbasis *Deep Learning*. Model yang digunakan pada penelitian ini terdiri atas YOLOv5n, YOLOv8n, dan YOLOv11n yang termasuk kategori lightweight karena memiliki ukuran parameter lebih kecil dan kebutuhan komputasi yang lebih ringan dibandingkan varian YOLO lainnya.

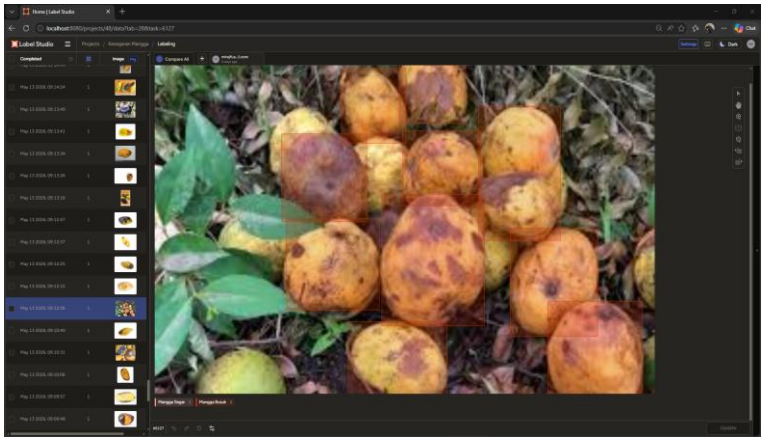
Analisis performa dilakukan menggunakan parameter evaluasi *object detection* seperti *precision*, *recall*, *F1-score*, dan *mean Average Precision* (mAP). Penelitian ini bertujuan untuk mengetahui model lightweight YOLO yang memiliki performa terbaik dalam mendeteksi kualitas buah mangga secara *real-time*.

CC BY-SA 4.0

Dataset Penelitian

Dataset penelitian diperoleh dari platform Kaggle (https://www.kaggle.com/datasets/muhammad0subhan) yang menyediakan dataset terbuka untuk kebutuhan penelitian *machine learning* dan *computer vision*. Dataset terdiri atas dua kelas objek, 1) Mangga segar (1813 gambar), 2) Mangga busuk (2247 gambar).

Citra dataset berupa gambar buah mangga dengan variasi kondisi pencahayaan, posisi objek, dan latar belakang. Dataset kemudian digunakan untuk proses pelatihan (*training*), validasi (*validation*), dan pengujian (*testing*) model YOLO. Pemberian label menggunakan software Label Studio. Pemberian format anotasi dataset menggunakan format YOLO (*.txt*) yang berisi informasi kelas objek dan koordinat *bounding box*. Setiap citra memiliki pasangan file anotasi dengan format: class x_center y_center width height.



Gambar 1. Anotasi mangga

```
0 0.3353971962616824 0.45981308411214955 0.17313084112149543 0.2355140186915888
0 0.7251168224299066 0.533177570093458 0.15630841121495337 0.180373831775701
0 0.5558411214953268 0.6696261682242985 0.20887850467289704 0.3373831775700928
0 0.8814252336448597 0.88411214953271 0.18855140186915909 0.23177570093457908
0 0.48609813084112163 0.9074766355140185 0.19415887850467292 0.18504672897196264
0 0.17207943925233643 0.9434579439252334 0.17172897196261677 0.11308411214953268
0 0.3567757009345795 0.6873831775700939 0.18785046728971966 0.2514018691588793
0 0.43668224299065417 0.2719626168224299 0.1317757009345794 0.2
0 0.2992990654205606 0.37943925233644854 0.14018691588785046 0.19439252336448593
0 0.5558411214953272 0.23598130841121495 0.14018691588785046 0.18971962616822435
0 0.5747663551401869 0.4401869158878499 0.15981308411214962 0.19439252336448581
0 0.7282710280373831 0.37757009345794396 0.16261682242990674 0.1925233644859813
0 0.7980140186915881 0.719158878504673 0.1745327102803732 0.2719626168224297
```

Gambar 2. Hasil anotasi dalam bentuk .txt

Preprocessing Data

Tahap *preprocessing* dilakukan untuk meningkatkan kualitas data sebelum proses pelatihan model. Seluruh citra diubah ke ukuran 640 × 640 piksel agar sesuai dengan kebutuhan model YOLO. Proses normalisasi dilakukan untuk menyetarakan rentang nilai piksel sehingga proses pelatihan model menjadi lebih stabil dan efisien.

Augmentasi Data

Augmentasi data dilakukan untuk meningkatkan variasi dataset dan mengurangi risiko *overfitting* akibat jumlah data yang terbatas. Teknik augmentasi diterapkan secara otomatis selama proses pelatihan model menggunakan framework Ultralytics YOLO. Beberapa teknik augmentasi yang digunakan seperti *horizontal flip, rotation, brightness adjustment, scaling dan mosaic augmentation*. Penggunaan augmentasi data bertujuan agar model mampu mengenali objek pada berbagai kondisi pencahayaan, posisi, dan sudut pengambilan gambar sehingga performa deteksi menjadi lebih baik pada kondisi nyata (*real-time environment*).

CC BY-SA 4.0

Pembagian Dataset

Dataset dibagi menjadi tiga bagian yaitu data *training*, *validation*, dan *testing*. Pembagian data dilakukan dengan rasio:
80% data training: 3248 gambar
10% data validation: 406 gambar
10% data testing: 406 gambar
Data *training* digunakan untuk melatih model YOLO, data *validation* digunakan untuk memantau performa model selama pelatihan, sedangkan data *testing* digunakan untuk evaluasi akhir performa model. Pembagian dataset dilakukan menggunakan metode *stratified split* agar distribusi kelas tetap seimbang pada setiap subset data.

Proses Pelatihan Model

Proses pelatihan model dilakukan menggunakan platform Google Colab dengan layanan Colab Pro dan GPU NVIDIA A100. Penggunaan GPU A100 bertujuan mempercepat proses pelatihan model *Deep Learning* terutama pada proses komputasi matriks dan pelatihan dataset berukuran besar. Framework yang digunakan dalam penelitian adalah Ultralytics YOLO berbasis Python. Setiap model dilatih menggunakan parameter yang sama agar hasil evaluasi performa dapat dibandingkan secara objektif. Parameter pelatihan model ditunjukkan pada Tabel 1.

Tabel 1. Parameter Training Model

Parameter	Nilai
Epoch	100
Batch Size	Auto
Input Size	640 × 640
Optimizer	AdamW
GPU	NVIDIA A100 40GB
Augmentasi	Blur, CLAHE, ToGray

Evaluasi Performa Model

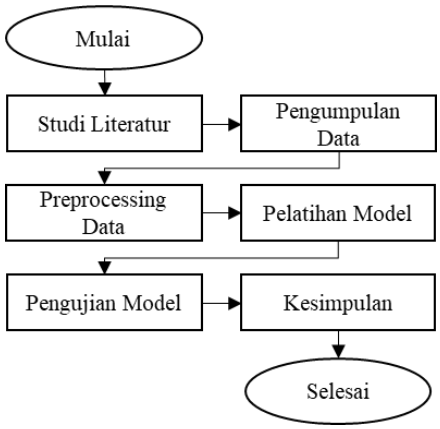
Evaluasi performa dilakukan untuk mengetahui kemampuan model dalam mendeteksi mangga segar dan mangga busuk. Parameter evaluasi yang digunakan meliputi *precision*, *recall*, *F1-score*, dan *mean Average Precision* (mAP).

Deployment

Deployment merupakan tahap penerapan model deteksi yang telah dilatih ke dalam sistem agar dapat digunakan untuk mendeteksi kondisi buah mangga secara langsung. Pada penelitian ini, tahap deployment dilakukan setelah model YOLOv5n, YOLOv8n, dan YOLOv11n selesai dilatih dan dievaluasi menggunakan data uji. Model terbaik yang diperoleh dari proses evaluasi kemudian disimpan dalam bentuk bobot model, seperti file .pt, untuk digunakan pada proses inferensi.

CC BY-SA 4.0

Diagram Alur Penelitian



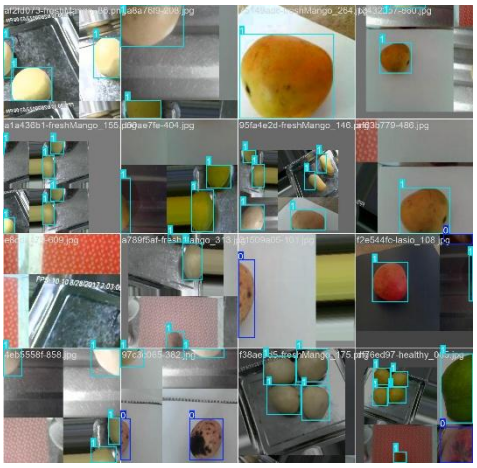
Gambar 3. Diagram alur penelitian

Alur penelitian digunakan untuk memastikan setiap tahapan dilakukan secara sistematis sehingga hasil analisis performa model dapat diperoleh secara optimal.

HASIL DAN PEMBAHASAN

Hasil Pelatihan Model Lightweight YOLO

Proses pelatihan model dilakukan menggunakan platform Google Colab dengan dukungan GPU NVIDIA A100 40GB. Pelatihan dilakukan selama 100 epoch menggunakan tiga varian lightweight YOLO, yaitu YOLOv5n, YOLOv8n, dan YOLOv11n. Seluruh model dilatih menggunakan parameter yang sama agar proses evaluasi dan perbandingan performa dapat dilakukan secara objektif.



Gambar 4. Training dataset mangga



Gambar 5. Validasi dataset mangga

Tabel 2. Hasil Pelatihan Model YOLO

Model	YOLOv5n	YOLOv8n	YOLOv11n
Layer	85	73	101
Parameter	2.503.334	3.006.038	2.582.542
GFLOPs	7,1	8,1	6,3
Ukuran Model	5,3 MB	6,3 MB	5,5 MB
Waktu Training	0,359 jam	0,337 jam	0,423 jam

CC BY-SA 4.0

Berdasarkan Tabel 2, setiap model memiliki karakteristik arsitektur yang berbeda. Jumlah *layer* menunjukkan kedalaman jaringan (*network depth*) pada model. Semakin banyak *layer*, maka kemampuan model dalam mengekstraksi fitur objek menjadi lebih kompleks dan detail karena jaringan mampu mempelajari pola visual pada berbagai tingkat abstraksi (Heaton, 2017). YOLOv11n memiliki jumlah *layer* terbanyak yaitu 101 *layer*, sehingga model memiliki kemampuan ekstraksi fitur yang lebih baik dalam mengenali warna, tekstur, dan bentuk objek mangga. Namun, penambahan *layer* juga dapat meningkatkan kompleksitas komputasi dan waktu pelatihan model.

Jumlah *parameter* menunjukkan banyaknya bobot (*weights*) yang dipelajari model selama proses pelatihan. Semakin besar jumlah parameter, maka kapasitas model dalam mempelajari pola data menjadi lebih tinggi (Zou et al., 2023). YOLOv8n memiliki jumlah parameter terbesar yaitu 3.006.038 parameter sehingga model memiliki kemampuan representasi fitur yang lebih baik. Akan tetapi, jumlah parameter yang besar dapat meningkatkan penggunaan memori dan risiko *overfitting* apabila dataset yang digunakan terbatas (Shorten & Khoshgoftaar, 2019).

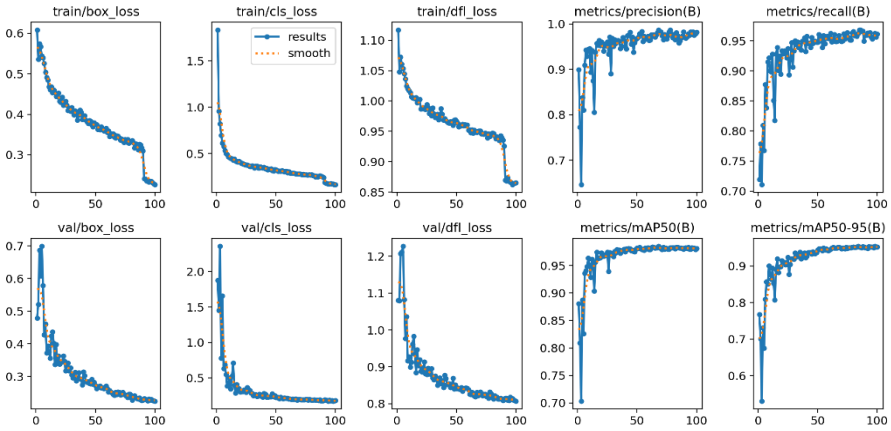
GFLOPs (*Giga Floating Point Operations*) menunjukkan jumlah operasi komputasi yang dibutuhkan model saat proses inferensi. Semakin kecil nilai GFLOPs, maka model semakin ringan dan efisien untuk dijalankan pada perangkat dengan keterbatasan sumber daya komputasi (Tan & Le, 2019). YOLOv11n memiliki nilai GFLOPs paling rendah yaitu 6,3 GFLOPs sehingga lebih efisien dibandingkan model lainnya. Nilai GFLOPs yang rendah sangat penting pada implementasi *real-time* dan perangkat *edge computing* karena dapat mempercepat proses inferensi dan mengurangi konsumsi daya komputasi.

Ukuran model (*model size*) menunjukkan besar file bobot hasil pelatihan yang digunakan saat proses inferensi. Model dengan ukuran kecil lebih mudah diterapkan pada perangkat *embedded system*, perangkat bergerak (*mobile devices*), maupun sistem *edge computing* (Howard et al., 2019). YOLOv5n memiliki ukuran model paling kecil yaitu 5,3 MB sehingga lebih ringan dalam proses penyimpanan dan distribusi model. Model lightweight dengan ukuran kecil umumnya lebih sesuai untuk aplikasi *real-time* karena membutuhkan memori yang lebih rendah tanpa mengurangi performa deteksi secara signifikan.

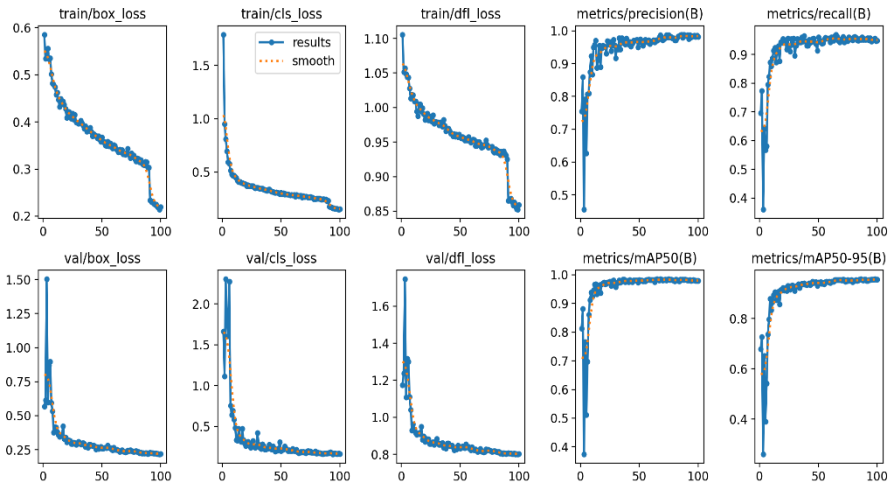
Dari hasil pelatihan tersebut, YOLOv8n menunjukkan waktu pelatihan tercepat yaitu 0,337 jam, sedangkan YOLOv11n membutuhkan waktu pelatihan paling lama sebesar 0,423 jam. Meskipun demikian, YOLOv11n menunjukkan efisiensi komputasi yang lebih baik karena memiliki nilai GFLOPs paling rendah dengan performa deteksi yang tetap tinggi. Hal ini menunjukkan bahwa arsitektur YOLOv11n lebih optimal dalam menyeimbangkan kompleksitas model, efisiensi komputasi, dan akurasi deteksi objek berbasis *Deep Learning*.

Evaluasi Performa Model

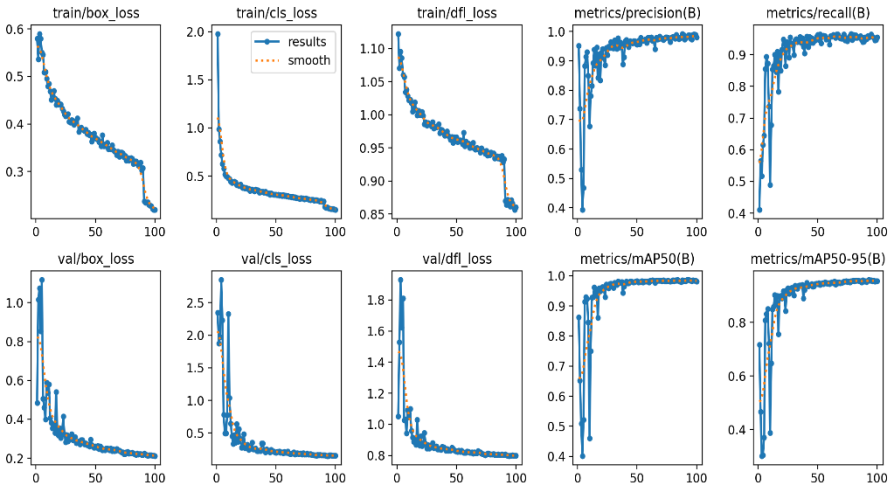
Evaluasi performa dilakukan untuk mengetahui kemampuan masing-masing model dalam mendeteksi objek mangga segar dan mangga busuk secara akurat. Parameter evaluasi yang digunakan pada penelitian ini meliputi *precision*, *recall*, *F1-score*, mAP@50, dan mAP@50:95. Parameter tersebut umum digunakan dalam penelitian *object detection* untuk mengukur tingkat ketepatan, sensitivitas, dan kemampuan localization model (Padilla et al., 2021).



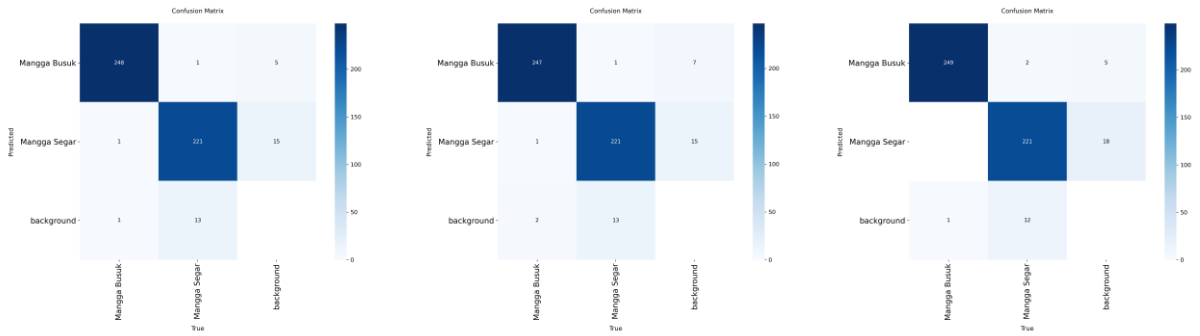
Gambar 6. Loss map YOLOv5n



Gambar 7. Loss map YOLOv8n



Gambar 8. Loss map YOLOv11n



(a) (b) (c)

Gambar 9. Confussion matrix YOLOv5n (a) YOLOv8n (b) YOLOv11n (c)

CC BY-SA 4.0

Tabel 3. Hasil Evaluasi Performa Model

Model	YOLOv5n	YOLOv8n	YOLOv11n
Precision	0,955	0,951	0,949
Recall	0,966	0,964	0,964
F1-Score	0,96	0,957	0,956
mAP@50	0,982	0,983	0,987
mAP@50:95	0,954	0,957	0,959

Berdasarkan Tabel 3, ketiga model lightweight YOLO menunjukkan performa yang sangat baik dalam mendeteksi mangga segar dan mangga busuk. Nilai precision berada pada rentang 0,949–0,955, recall pada rentang 0,964–0,966, F1-Score pada rentang 0,956–0,960, serta mAP@50:95 pada rentang 0,954–0,959. Hasil ini menunjukkan bahwa seluruh model memiliki kemampuan deteksi yang tinggi, baik dalam mengenali kelas objek maupun menentukan lokasi objek melalui bounding box. Dalam evaluasi object detection, metrik mAP menjadi indikator penting karena tidak hanya menilai ketepatan klasifikasi, tetapi juga kualitas lokalisasi objek berdasarkan nilai Intersection over Union atau IoU (Padilla et al., 2021).

Jika dilihat dari nilai precision, recall, dan F1-Score, model YOLOv5n memperoleh hasil tertinggi, yaitu precision 0,955, recall 0,966, dan F1-Score 0,960. Hal ini menunjukkan bahwa YOLOv5n memiliki keseimbangan paling baik dalam mengurangi kesalahan prediksi positif (*false positive*) dan objek yang tidak terdeteksi (*false negative*). Namun, selisih performa antara YOLOv5n, YOLOv8n, dan YOLOv11n sangat kecil, yaitu kurang dari 1%, sehingga ketiga model dapat dikatakan memiliki performa yang relatif setara. Perbedaan performa yang relatif kecil antar model kemungkinan dipengaruhi oleh karakteristik arsitektur jaringan, kualitas dan konsistensi anotasi dataset, serta pengaturan parameter pelatihan dan inferensi seperti *confidence threshold*, ukuran input citra (*image size*), dan *Non-Maximum Suppression* (NMS). Parameter-parameter tersebut diketahui dapat memengaruhi akurasi deteksi, presisi, recall, serta kemampuan model dalam mengurangi deteksi ganda (Sirisha et al., 2023).

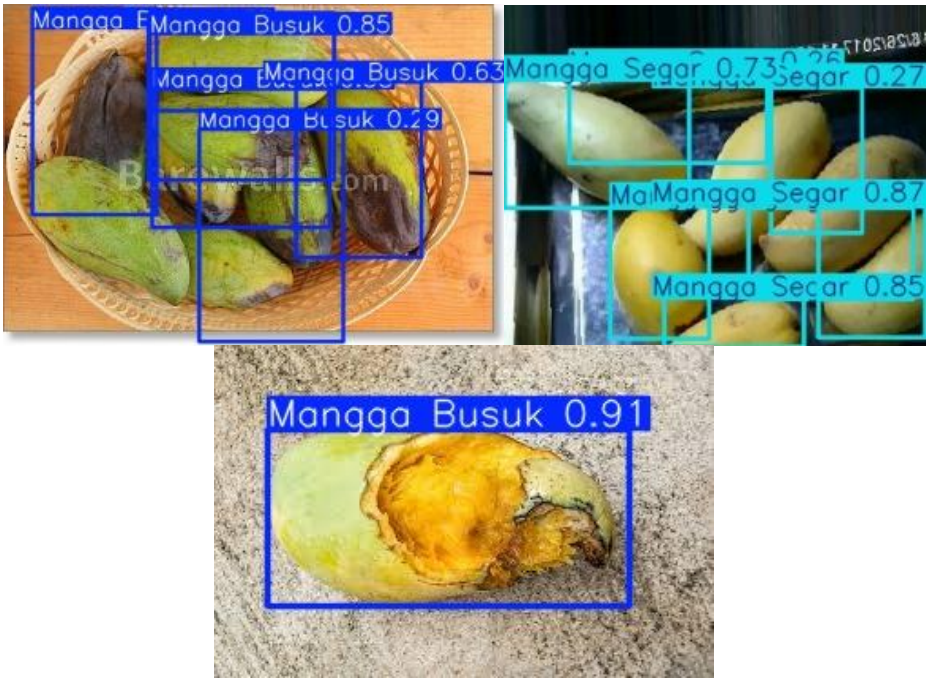
Pada metrik mAP@50 dan mAP@50:95, model YOLOv11n menunjukkan hasil terbaik dengan nilai masing-masing 0,987 dan 0,959. Nilai tersebut lebih tinggi dibandingkan YOLOv8n dengan mAP@50 0,983 dan mAP@50:95 0,957, serta YOLOv5n dengan mAP@50 0,982 dan mAP@50:95 0,954. Hal ini menunjukkan bahwa YOLOv11n memiliki kemampuan lokalisasi objek yang sedikit lebih baik, terutama pada evaluasi dengan ambang IoU yang lebih ketat. Hasil ini sejalan dengan penelitian Jegham et al. (Jegham et al., 2025), yang melaporkan bahwa YOLO11 memiliki peningkatan pada aspek akurasi, efisiensi, dan kemampuan ekstraksi fitur dibandingkan beberapa versi YOLO sebelumnya.

Jika dibandingkan dengan penelitian lain di bidang pertanian, hasil pada penelitian ini tergolong sangat tinggi. Asad et al. (Asad et al., 2026) melaporkan bahwa YOLO11n pada deteksi apel di lingkungan kebun memperoleh mAP@0.5 sebesar 0,9620 dan mAP@0.5:0.95 sebesar 0,6065, lebih rendah dibandingkan hasil YOLOv11n pada penelitian ini. Gamani et al. (Gamani et al., 2024) juga melaporkan bahwa YOLOv8n pada segmentasi tingkat kematangan stroberi di lapangan terbuka memperoleh mAP sekitar 80,9%. Perbedaan tersebut menunjukkan bahwa kompleksitas dataset sangat memengaruhi performa model, terutama pada citra lapangan yang memiliki oklusi daun, pencahayaan tidak seragam, dan latar belakang kompleks. Secara keseluruhan, YOLOv5n unggul pada keseimbangan precision, recall, dan F1-Score, sedangkan YOLOv11n lebih unggul pada kualitas deteksi dan lokalisasi berdasarkan mAP, sehingga YOLOv11n dapat dipilih sebagai model terbaik apabila fokus utama penelitian adalah ketepatan deteksi objek secara menyeluruh.

Deployement

Pada tahap deployment, model YOLO yang telah dilatih disimpan dalam bentuk file bobot best.pt, kemudian digunakan untuk melakukan inferensi terhadap citra baru. Proses deployment dilakukan di Google Colab dengan memanfaatkan library Ultralytics. Citra input dimasukkan ke dalam sistem, kemudian model menghasilkan output berupa bounding box, label kelas, dan confidence score. Hasil deteksi tersebut digunakan untuk mengidentifikasi kondisi buah mangga, yaitu mangga segar atau mangga busuk.

CC BY-SA 4.0



Gambar 10. Hasil deteksi menggunakan model YOLOv11n

KESIMPULAN DAN IMPLIKASI

Berdasarkan hasil penelitian, model lightweight YOLO terbukti mampu mendeteksi kualitas buah mangga segar dan busuk dengan performa yang sangat baik melalui tahapan preprocessing, augmentasi data, stratified split, pelatihan 100 epoch, serta evaluasi menggunakan precision, recall, F1-Score, mAP@50, dan mAP@50:95. Dari ketiga model yang diuji, YOLOv5n unggul pada precision, recall, dan F1-Score, sedangkan YOLOv11n memperoleh performa terbaik pada mAP@50 sebesar 0,987 dan mAP@50:95 sebesar 0,959 serta memiliki GFLOPs paling rendah sebesar 6,3, sehingga lebih efisien untuk implementasi real-time. Dengan demikian, YOLOv11n dapat direkomendasikan sebagai model terbaik apabila fokus utama penelitian adalah kualitas deteksi dan efisiensi komputasi, sedangkan YOLOv5n tetap unggul dalam keseimbangan precision dan recall.

DAFTAR PUSTAKA

Asad, M., Gautam, A. K., Dhiman, P., & Prajapati, R. R. (2026). *A Comparative Study of Modern Object Detectors for Robust Apple Detection in Orchard Imagery*. https://arxiv.org/abs/2604.09996

Badgujar, C. M., Poulose, A., & Gan, H. (2024). Agricultural object detection with You Only Look Once (YOLO) Algorithm: A bibliometric and systematic literature review. *Computers and Electronics in Agriculture*, *223*, 109090. https://doi.org/10.1016/j.compag.2024.109090

Gamani, A.-R. A., Arhin, I., & Asamoah, A. K. (2024). *Performance Evaluation of YOLOv8 Model Configurations, for Instance Segmentation of Strawberry Fruit Development Stages in an Open Field Environment*. https://arxiv.org/abs/2408.05661

Ge, Z., Liu, S., Wang, F., Li, Z., & Sun, J. (2021). {YOLOX:} Exceeding {YOLO} Series in 2021. *CoRR*, *abs/2107.0*. https://arxiv.org/abs/2107.08430

Heaton, J. (2017). Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning. *Genetic Programming and Evolvable Machines*, *19*(1–2), 305–307. https://doi.org/10.1007/s10710-017-9314-z

Howard, A., Sandler, M., Chen, B., Wang, W., Chen, L.-C., Tan, M., Chu, G., Vasudevan, V., Zhu, Y., Pang, R., Adam, H., & Le, Q. (2019). Searching for MobileNetV3. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (pp. 1314–1324). IEEE. https://doi.org/10.1109/iccv.2019.00140

Jegham, N., Koh, C. Y., Abdelatti, M., & Hendawi, A. (2025). *YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions*. https://arxiv.org/abs/2411.00201

CC BY-SA 4.0

Nuha, M. S., & Alexandro H., R. (2022). Pemanfaatan Yolo untuk Pengenalan Kesegaran Buah Mangga. *Joutica*, *7*(1), 513. https://doi.org/10.30736/jti.v7i1.747

Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L., & da Silva, E. A. B. (2021). A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics*, *10*(3), 279. https://doi.org/10.3390/electronics10030279

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0197-0

Sirisha, U., Praveen, S. P., Srinivasu, P. N., Barsocchi, P., & Bhoi, A. K. (2023). Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection. *International Journal of Computational Intelligence Systems*, *16*(1), 126. https://doi.org/10.1007/s44196-023-00302-w

Tan, M., & Le, Q. (2019). {E}fficient{N}et: Rethinking Model Scaling for Convolutional Neural Networks. In K. Chaudhuri & R. Salakhutdinov (Eds.), *Proceedings of the 36th International Conference on Machine Learning* (Vol. 97, pp. 6105–6114). PMLR. https://proceedings.mlr.press/v97/tan19a.html

Zhong, Z., Yun, L., Cheng, F., Chen, Z., & Zhang, C. (2024). Light-YOLO: A Lightweight and Efficient YOLO-Based Deep Learning Model for Mango Detection. *Agriculture*, *14*(1), 140. https://doi.org/10.3390/agriculture14010140

Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object Detection in 20 Years: A Survey. *Proceedings of the IEEE*, *111*(3), 257–276. https://doi.org/10.1109/jproc.2023.3238524

CC BY-SA 4.0