

# **Laporan Tugas Besar**

## **Deteksi Cacat Biji Kopi**

Mata Kuliah: Visi Komputer Fundamental



# **Universitas Telkom**

Muhammad Abiya Makruf (203012420034)

S2 Informatika

Fakultas Informatika

2026

## Contents

I.	PENDAHULUAN .....	3
1.1.	Latar Belakang .....	3
1.2.	Deskripsi Proyek .....	3
1.3.	Tujuan .....	3
1.4.	Permasalahan .....	4
1.5.	Batasan Penelitian.....	4
II.	DATASET .....	4
2.1.	Deskripsi Dataset Biji Kopi Arabika.....	4
2.2.	Preprocessing .....	5
2.2.1.	Resize.....	5
2.2.2.	Augmentasi Data.....	5
2.3.	Pembagian Data .....	6
III.	METODE.....	6
3.1.	Formulasi Masalah sebagai Object Detection.....	6
3.2.	Alasan Pemilihan YOLO .....	6
3.3.	Skema Eksperimen.....	7
3.4.	Metrik Evaluasi.....	7
IV.	IMPLEMENTASI .....	7
4.1.	Lingkungan dan Tools.....	7
4.2.	Konfigurasi Training.....	7
4.3.	Pembuatan Video Conveyor.....	8
V.	EVALUASI .....	8
5.1.	Evaluasi Pemilihan Versi YOLO .....	8
5.2.	Evaluasi Pengaruh Ukuran Image.....	9
5.3.	Evaluasi Penggunaan Augmentasi Data.....	10
5.4.	Perbandingan Inference Time .....	11
5.5.	Visualisasi Bagaimana Model Melihat .....	12
5.6.	Evaluasi False Positive dan False Negatif .....	13
5.7.	Implementasi Studi Kasus.....	14
VI.	KESIMPULAN.....	15

# I. PENDAHULUAN

## 1.1. Latar Belakang

Deteksi cacat pada produk pertanian merupakan salah satu permasalahan penting dalam bidang *computer vision*, khususnya untuk mendukung proses *quality control* secara otomatis. Pada industri kopi, kualitas biji kopi sangat menentukan nilai jual dan mutu produk akhir. Proses inspeksi kualitas biji kopi secara manual memiliki keterbatasan, antara lain bersifat subjektif, membutuhkan waktu yang lama, serta sulit diterapkan secara konsisten dalam skala besar.

Perkembangan *deep learning*, khususnya pada tugas *object detection*, membuka peluang untuk membangun sistem deteksi cacat biji kopi yang lebih akurat dan efisien. Model berbasis convolutional neural network (CNN) mampu mengekstraksi fitur visual kompleks yang sulit ditangkap oleh metode tradisional. Salah satu pendekatan yang banyak digunakan adalah You Only Look Once (YOLO), yang dikenal memiliki kecepatan inferensi tinggi dan performa deteksi yang baik.

Oleh karena itu, penelitian ini berfokus pada penerapan model YOLO untuk mendeteksi cacat pada biji kopi Arabika sebagai bentuk implementasi *anomaly detection* dalam *computer vision*.

## 1.2. Deskripsi Proyek

Proyek ini bertujuan untuk membangun sistem object detection yang mampu mendeteksi dan mengklasifikasikan cacat pada biji kopi Arabika. Cacat biji kopi didefinisikan ke dalam beberapa kelas tertentu yang merepresentasikan kondisi abnormal pada biji kopi.

Model yang digunakan adalah YOLO, dengan proses pemilihan versi model dilakukan melalui serangkaian eksperimen. Eksperimen meliputi pemilihan versi YOLO berdasarkan performa validasi, pengujian pengaruh ukuran model (nano, small, dan medium), variasi ukuran citra input, serta penggunaan augmentasi data. Selain itu, model terbaik yang diperoleh diimplementasikan pada skenario inferensi video, yaitu simulasi biji kopi yang bergerak di atas conveyor.

## 1.3. Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

- a. Mengimplementasikan model object detection berbasis YOLO untuk mendeteksi cacat pada biji kopi Arabika.
- b. Menganalisis pengaruh variasi ukuran model YOLO terhadap performa deteksi.
- c. Menganalisis pengaruh ukuran citra input terhadap nilai mAP.
- d. Mengevaluasi dampak penggunaan augmentasi data terhadap performa model.
- e. Mengimplementasikan model terbaik pada skenario inferensi video untuk studi kasus deteksi dan penghitungan objek biji kopi.

#### 1.4. Permasalahan

Berdasarkan latar belakang yang telah dijelaskan, permasalahan yang diangkat dalam penelitian ini adalah:

- a. Bagaimana performa model YOLO dalam mendeteksi cacat pada biji kopi Arabika?
- b. Versi YOLO manakah yang memberikan performa terbaik berdasarkan nilai mAP pada data validasi?
- c. Bagaimana pengaruh ukuran citra input terhadap akurasi dan kecepatan inferensi model?
- d. Apakah penggunaan augmentasi data dapat meningkatkan performa deteksi cacat biji kopi?
- e. Bagaimana performa model ketika diterapkan pada data video biji kopi yang bergerak di atas conveyor?

#### 1.5. Batasan Penelitian

Agar penelitian lebih terfokus, maka batasan penelitian yang diterapkan adalah sebagai berikut:

- a. Dataset yang digunakan terbatas pada citra biji kopi Arabika dengan kelas cacat yang telah ditentukan.
- b. Metode yang digunakan dibatasi pada pendekatan object detection menggunakan model YOLO.
- c. Eksperimen hanya dilakukan pada variasi ukuran model YOLO nano, small, dan medium.
- d. Ukuran citra input yang diuji dibatasi pada resolusi tertentu.
- e. Implementasi inferensi video dilakukan pada skenario simulasi conveyor, bukan pada sistem industri nyata.

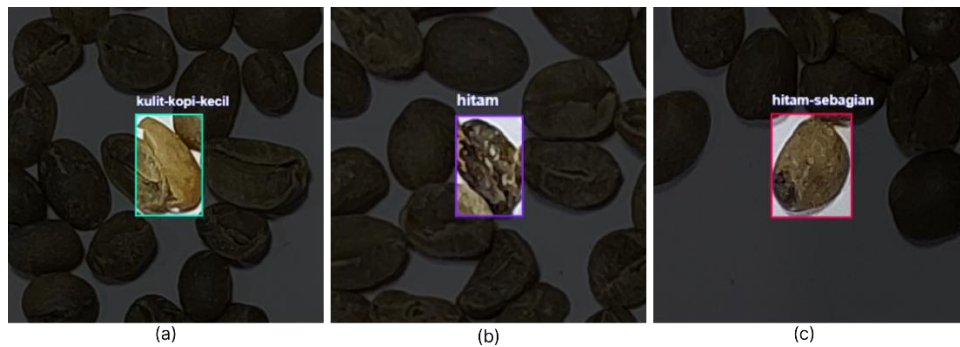
## II. DATASET

#### 2.1. Deskripsi Dataset Biji Kopi Arabika

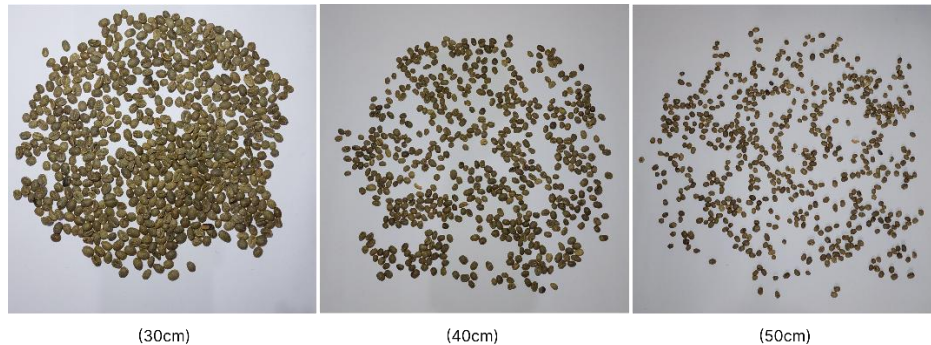
Dataset yang digunakan dalam penelitian ini merupakan dataset citra biji kopi Arabika yang difokuskan pada pendeteksian cacat visual. Setiap citra berisi 75gram biji kopi yang ditangkap dalam kondisi pencahayaan ruangan dan ketinggian yang berbeda-beda, ketinggian dimulai dari 30cm, 40cm, hingga 50cm. Kelas cacat yang digunakan dalam penelitian ini terdiri dari:

- a. Hitam, yaitu biji kopi yang seluruh permukaannya berwarna gelap atau hitam.
- b. Hitam-sebagian, yaitu biji kopi yang hanya sebagian permukaannya berwarna hitam.
- c. Kulit-kopi-kecil, yaitu sisa kulit kopi berukuran kecil yang masih menempel atau terpisah dari biji utama.

Dataset diperoleh dari <https://universe.roboflow.com/lulus-vpibo/green-coffee-bean-defects>, dan telah dilengkapi dengan anotasi bounding box untuk setiap objek yang termasuk dalam kelas cacat. Dataset ini dipilih karena relevan dengan skenario anomaly detection pada proses quality control biji kopi. Visualisasi dari masing-masing kelas dapat dilihat pada Gambar 1.



Gambar 1. Visualisasi masing-masing kelas. (a) kulit kopi kecil, (b) hitam, (c) hitam sebagian



Gambar 2. Variasi ketinggian yang terdapat pada dataset.

## 2.2. Preprocessing

Sebelum digunakan pada proses pelatihan model, dataset melalui beberapa tahap preprocessing untuk memastikan kesesuaian dengan arsitektur model YOLO serta meningkatkan kualitas data latih. Tahapan preprocessing yang dilakukan meliputi:

### 2.2.1. Resize

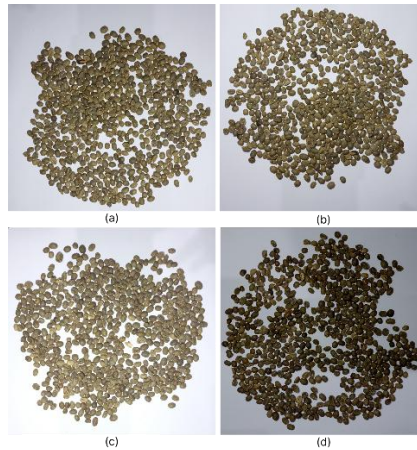
Seluruh citra pada dataset diubah ukurannya dari resolusi awal  $3456 \times 3456$  piksel ke resolusi  $640 \times 640$  dan  $1024 \times 1024$  agar sesuai dengan kebutuhan input model YOLO. Proses resize ini bertujuan untuk menyamakan dimensi citra input sehingga pelatihan dapat dilakukan secara konsisten.

### 2.2.2. Augmentasi Data

Untuk meningkatkan variasi data latih dan mengurangi risiko overfitting, dilakukan augmentasi data dengan beberapa teknik sebagai berikut:

- Flip, untuk mensimulasikan variasi orientasi objek.
- Rotate, untuk menambah variasi sudut pandang objek biji kopi.
- Brightness adjustment, dengan perubahan tingkat kecerahan sebesar  $\pm 15\%$  untuk mensimulasikan kondisi pencahayaan yang berbeda.

Augmentasi ini diterapkan pada data latih dan disesuaikan dengan format anotasi bounding box yang digunakan oleh model YOLO. Visualisasi data setelah proses augmentasi dapat dilihat pada Gambar 3.



Gambar 3. Visualisasi hasil dari augmentasi. (a) flip, (b) rotate, (c) brighter, (d) darker

### 2.3. Pembagian Data

Dataset yang digunakan dalam penelitian ini terdiri dari total 1002 citra biji kopi Arabika. Dataset tersebut dibagi ke dalam tiga subset utama, yaitu data latih, data validasi, dan data uji, dengan rincian sebagai berikut:

- a. 722 citra digunakan sebagai data latih (training set)
- b. 80 citra digunakan sebagai data validasi (validation set)
- c. 200 citra digunakan sebagai data uji (test set)

Selain skema pembagian data awal tersebut, dilakukan pula eksperimen tambahan dengan menerapkan augmentasi data secara khusus pada data latih. Pada skema ini, hanya data latih yang dikenai augmentasi berupa:

- a. penyesuaian brightness,
- b. rotasi, dan
- c. flip.

Melalui proses augmentasi tersebut, jumlah data latih bertambah dari 722 citra menjadi 1444 citra, sementara jumlah data validasi dan data uji tetap sama, yaitu masing-masing 80 citra dan 200 citra. Skema ini digunakan untuk menganalisis pengaruh augmentasi data terhadap performa model deteksi cacat biji kopi.

## III. METODE

### 3.1. Formulasi Masalah sebagai Object Detection

Permasalahan deteksi cacat pada biji kopi Arabika dalam penelitian ini diformulasikan sebagai masalah object detection. Setiap biji kopi atau bagian biji kopi yang mengalami cacat diperlakukan sebagai sebuah objek yang harus dideteksi dan diklasifikasikan ke dalam kelas cacat tertentu.

Model diharapkan tidak hanya mampu mengidentifikasi keberadaan cacat pada citra, tetapi juga menentukan lokasi objek cacat tersebut dalam bentuk bounding box serta memberikan label kelas yang sesuai.

### 3.2. Alasan Pemilihan YOLO

Model YOLO dipilih sebagai metode utama dalam penelitian ini karena memiliki keunggulan pada kecepatan inferensi dibandingkan dengan metode object detection dua tahap seperti R-CNN dan turunannya. YOLO menggunakan pendekatan satu tahap (single-stage detector) yang memungkinkan proses deteksi dilakukan secara end-to-end dalam satu kali forward pass.

Karakteristik ini menjadikan YOLO sangat sesuai untuk aplikasi yang membutuhkan pemrosesan cepat, termasuk implementasi inferensi pada video dan skenario yang mendekati kondisi real-time, seperti simulasi conveyor biji kopi.

### **3.3. Skema Eksperimen**

Untuk memperoleh model dengan performa terbaik, dilakukan serangkaian eksperimen terstruktur. Skema eksperimen yang diterapkan meliputi:

1. Pemilihan versi YOLO berdasarkan performa mAP pada data validasi.
2. Pengujian pengaruh ukuran model (nano, small, dan medium).
3. Pengujian pengaruh ukuran citra input.
4. Perbandingan pelatihan dengan dan tanpa augmentasi data.
5. Implementasi model terbaik pada skenario inferensi video.

Setiap eksperimen dilakukan secara terpisah untuk memastikan bahwa pengaruh setiap variabel dapat dianalisis secara jelas.

### **3.4. Metrik Evaluasi**

Evaluasi performa model dilakukan menggunakan metrik yang umum digunakan pada tugas object detection. Metrik utama yang digunakan dalam penelitian ini adalah mean Average Precision (mAP).

Selain itu, metrik tambahan seperti precision dan recall juga digunakan untuk menganalisis kemampuan model dalam mendeteksi objek cacat secara benar serta menghindari kesalahan deteksi. Metrik-metrik ini digunakan sebagai dasar dalam pemilihan model terbaik dan analisis hasil eksperimen.

## **IV. IMPLEMENTASI**

### **4.1. Lingkungan dan Tools**

Implementasi sistem deteksi cacat biji kopi Arabika pada penelitian ini dilakukan menggunakan lingkungan pengembangan berbasis deep learning dengan dukungan GPU serta framework deep learning yang digunakan adalah YOLO (Pytorch). Proses pelatihan dan inferensi model dijalankan pada perangkat keras dengan spesifikasi sebagai berikut:

- a. Prosesor: AMD Ryzen 7 8700F
- b. RAM: 32 GB
- c. GPU: NVIDIA RTX 5070 Ti dengan 16 GB VRAM

### **4.2. Konfigurasi Training**

Pelatihan model YOLO dilakukan dengan konfigurasi parameter yang ditetapkan secara konsisten pada seluruh eksperimen, kecuali pada parameter yang divariasikan

sesuai skema eksperimen. Konfigurasi pelatihan yang digunakan adalah sebagai berikut:

*Tabel 1. Konfigurasi Training*

No	Nama	Value
1	Jumlah epoch	50
2	Batch size	16
3	Ukuran gambar	640 dan 1024
4	Patience	100
5	Jumlah worker	8
6	Pre-trained	True
7	Freeze	False

### 4.3. Pembuatan Video Conveyor

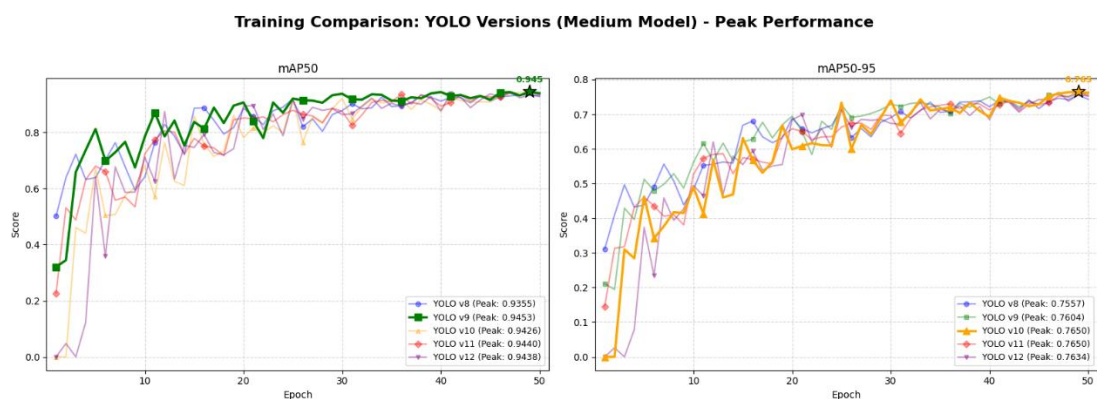
Sebagai bagian dari implementasi studi kasus, dibuat sebuah video yang mensimulasikan pergerakan biji kopi di atas conveyor. Video ini digunakan sebagai input pada tahap inferensi untuk menguji kemampuan model dalam mendeteksi objek secara berurutan pada data video. Video dibuat dengan cara menggabungkan gambar dari dataset dan diedit sehingga bergerak dari kanan ke kiri seperti berasa diatas conveyor.

## V. EVALUASI

### 5.1. Evaluasi Pemilihan Versi YOLO

Model YOLO memiliki berbagai varian versi yang terus berkembang, mulai dari YOLOv8 hingga YOLOv12. Untuk menentukan versi YOLO yang paling sesuai pada penelitian ini, dilakukan pengujian terhadap seluruh varian tersebut. Pada tahap ini, seluruh model dilatih menggunakan ukuran model medium, karena ukuran ini dianggap mewakili keseimbangan antara kompleksitas model dan performa, sehingga sesuai digunakan sebagai basis perbandingan antar versi.

Berdasarkan hasil pelatihan, seluruh versi YOLO yang diuji berhasil mencapai nilai mAP50 di atas 0.94 pada data pelatihan. YOLOv9 menunjukkan nilai puncak mAP50 tertinggi dengan nilai 0.9453, sementara performa terbaik pada metrik mAP50–95 dicapai oleh YOLOv10 dengan nilai puncak sebesar 0.7650. Visualisasi grafik pelatihan untuk seluruh versi model ditunjukkan pada Gambar 4.

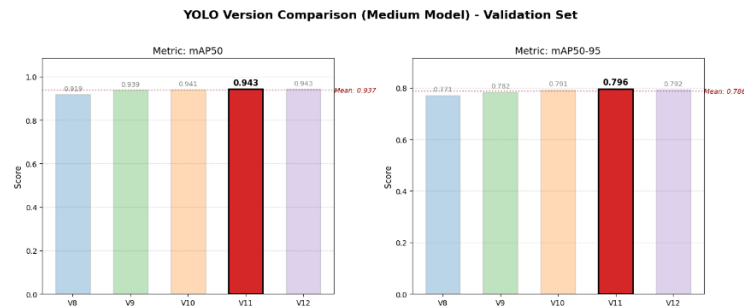


*Gambar 4. Perbandingan grafik pelatihan (mAP50 dan mAP50–95) untuk berbagai versi YOLO..*



Meskipun demikian, grafik pelatihan tidak dijadikan sebagai satu-satunya tolok ukur dalam pemilihan model terbaik. Hal ini disebabkan karena nilai mAP pada fase pelatihan dan validasi belum tentu merepresentasikan performa model secara optimal pada data yang benar-benar belum pernah dilihat sebelumnya. Oleh karena itu, pemilihan model terbaik ditentukan berdasarkan performa evaluasi pada data testing.

Berdasarkan hasil evaluasi pada data testing yang ditunjukkan pada Gambar 5, model dengan performa tertinggi adalah YOLOv11, dengan nilai mAP50 sebesar 0.943 dan mAP50–95 sebesar 0.796.



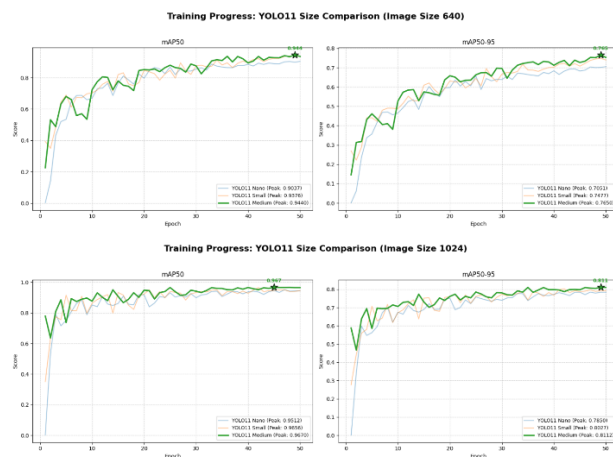
Gambar 5. Perbandingan performa mAP50 dan mAP50–95 pada data testing untuk berbagai versi YOLO.

Berdasarkan hasil tersebut, YOLOv11 dipilih sebagai model utama dan digunakan pada tahap eksperimen selanjutnya dalam penelitian ini.

## 5.2. Evaluasi Pengaruh Ukuran Image

Pada tahap ini, model yang digunakan adalah YOLOv11 sebagai model terpilih dari hasil evaluasi sebelumnya. Pengujian dilakukan dengan menggunakan dua ukuran citra input, yaitu  $640 \times 640$  dan  $1024 \times 1024$ . Selain variasi ukuran citra, pada tahap ini juga dilakukan pengujian terhadap beberapa ukuran model, yaitu nano, small, dan medium.

Pengujian ini menjadi penting karena dalam satu citra terdapat sekitar 75 gram biji kopi yang terdiri dari puluhan hingga ratusan biji kopi, sehingga ukuran objek relatif kecil dan padat dalam satu gambar. Oleh karena itu, analisis pengaruh ukuran citra input terhadap kemampuan model dalam mendeteksi objek kecil menjadi krusial.

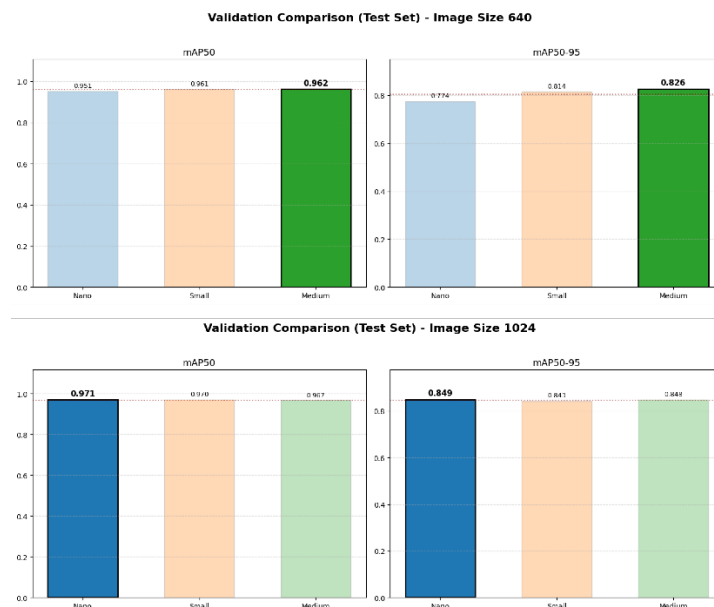


Gambar 6. Perbandingan grafik pelatihan untuk YOLOv11 dengan variasi ukuran citra dan ukuran model

Berdasarkan grafik pelatihan yang ditunjukkan pada Gambar 6, ukuran citra  $1024 \times 1024$  menghasilkan nilai mAP50 yang lebih tinggi dibandingkan dengan ukuran  $640 \times 640$ , masing-masing sebesar 0.967 dan 0.944. Meskipun perbedaan nilai mAP50 relatif tidak terlalu besar, peningkatan yang lebih signifikan terlihat pada metrik mAP50–95, di mana ukuran 1024 mencapai nilai 0.811, sedangkan ukuran 640 hanya mencapai 0.765.

Pada kedua ukuran citra tersebut, model dengan ukuran medium secara konsisten memberikan nilai mAP tertinggi, diikuti oleh model small, dan kemudian nano. Namun demikian, sebagaimana pada tahap evaluasi sebelumnya, pemilihan model tidak didasarkan pada grafik pelatihan, melainkan pada performa model pada data testing.

Hasil evaluasi pada data seperti yang terlihat pada Gambar 7 testing menunjukkan pola yang menarik. Ukuran citra  $1024 \times 1024$  tetap memberikan performa mAP yang lebih tinggi dibandingkan dengan  $640 \times 640$ , namun konfigurasi ukuran model terbaik berbeda untuk masing-masing ukuran citra. Pada ukuran citra 640, model dengan performa terbaik adalah YOLOv11 medium dengan nilai mAP50 sebesar 0.962 dan mAP50–95 sebesar 0.826. Sebaliknya, pada ukuran citra 1024, model dengan performa terbaik adalah YOLOv11 nano dengan nilai mAP50 sebesar 0.971 dan mAP50–95 sebesar 0.849.



Gambar 7. Perbandingan performa pada data testing untuk YOLOv11 dengan variasi ukuran citra dan ukuran model

Perbandingan performa tersebut menunjukkan bahwa ukuran citra input memiliki pengaruh yang signifikan terhadap performa deteksi, baik dari sisi akurasi maupun pemilihan ukuran model yang optimal.

### 5.3. Evaluasi Penggunaan Augmentasi Data

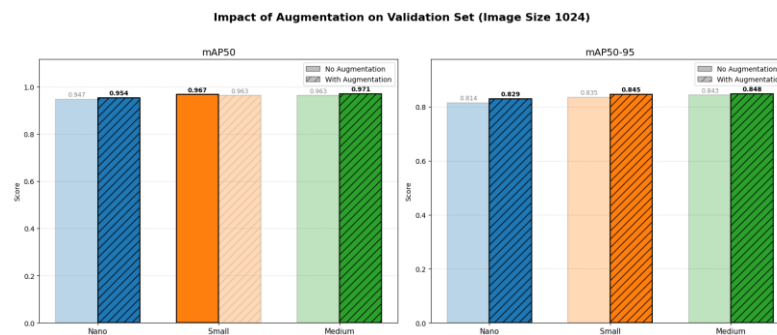
Berdasarkan hasil evaluasi pada tahap sebelumnya, diketahui bahwa ukuran citra input memiliki pengaruh signifikan terhadap nilai mAP. Oleh karena itu, pada tahap ini digunakan ukuran citra terbaik, yaitu  $1024 \times 1024$ , untuk melakukan pelatihan ulang dengan menerapkan augmentasi data.

Berdasarkan hasil evaluasi pada data testing yang ditunjukkan pada Gambar 8, terlihat bahwa hampir seluruh ukuran model mengalami peningkatan nilai mAP ketika dilatih menggunakan data augmentasi. Satu-satunya pengecualian terjadi pada metrik mAP50 untuk model small, di mana nilai tanpa augmentasi sebesar 0.967 sedikit lebih tinggi dibandingkan dengan nilai dengan augmentasi sebesar 0.963.

Untuk metrik mAP50, model nano mengalami peningkatan dari 0.947 menjadi 0.954, yang setara dengan peningkatan sekitar 0.74%. Model medium juga menunjukkan peningkatan dari 0.963 menjadi 0.971, atau sekitar 0.83%. Sementara itu, model small mengalami penurunan sekitar 0.41% pada metrik mAP50.

Pada metrik mAP50–95, seluruh ukuran model menunjukkan peningkatan performa. Model nano mengalami peningkatan dari 0.814 menjadi 0.829, atau sekitar 1.84%. Model small meningkat dari 0.835 menjadi 0.845, setara dengan peningkatan sekitar 1.20%, sedangkan model medium meningkat dari 0.843 menjadi 0.848, atau sekitar 0.59%.

Hasil ini menunjukkan bahwa augmentasi data secara umum memberikan dampak positif terhadap kemampuan generalisasi model, terutama pada metrik mAP50–95 yang merepresentasikan kualitas prediksi bounding box secara lebih menyeluruh.



Gambar 8. Perbandingan nilai pada data testing untuk YOLOv11 dengan ukuran citra  $1024 \times 1024$ , sebelum dan sesudah penerapan augmentasi data.

#### 5.4. Perbandingan Inference Time

Perbandingan inference time dilakukan untuk mengevaluasi efisiensi komputasi dari masing-masing ukuran model YOLOv11 pada data testing. Pengujian dilakukan menggunakan ukuran citra  $1024 \times 1024$ , dan waktu pemrosesan diukur pada setiap tahap, yaitu preprocess, inference, dan postprocess.

Berdasarkan hasil pengujian, model small menunjukkan waktu inferensi tercepat, sedangkan model medium memiliki waktu inferensi paling lambat akibat kompleksitas model yang lebih tinggi. Model nano berada di antara keduanya dengan waktu inferensi yang relatif cepat dan stabil. Hasil ini menunjukkan adanya trade-off yang jelas antara kompleksitas model dan kecepatan inferensi.

Tabel 2. Perbandingan Inference Time YOLOv11 pada Data Testing (Image Size  $1024 \times 1024$ )

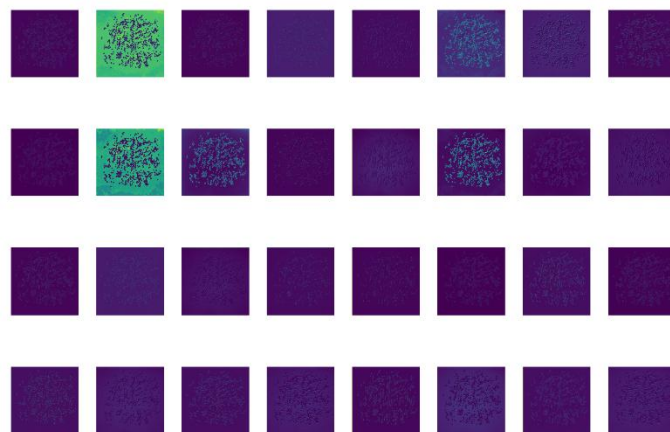
Model Size	Preprocess (ms)	Inference (ms)	Postprocess (ms)	Total Time (ms)
Nano	5.6	9.1	1.3	16.0
Small	5.1	8.2	1.1	14.4
Medium	5.1	12.8	1.2	19.1

### 5.5. Visualisasi Bagaimana Model Melihat

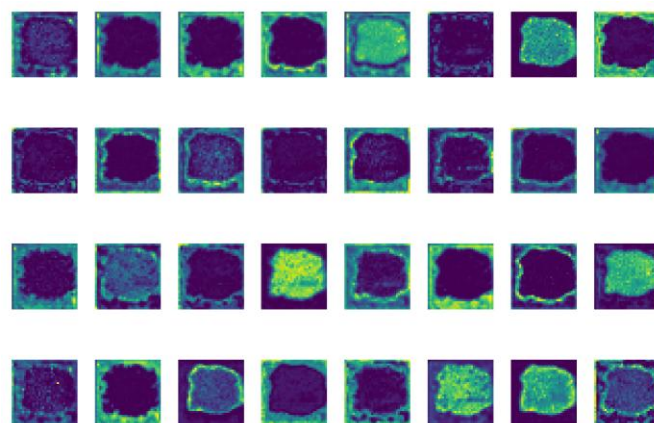
Untuk memahami bagaimana model YOLO memproses informasi visual dari citra biji kopi, dilakukan visualisasi feature map pada beberapa stage konvolusi. Pada evaluasi ini, ditampilkan feature map pada stage awal (stage 0) dan stage akhir (stage 22) untuk melihat perbedaan representasi fitur yang dihasilkan oleh model.

Pada stage 0 (Gambar 9), feature map masih mempertahankan struktur visual citra asli secara cukup jelas. Pola bentuk biji kopi, tepi objek, serta perbedaan intensitas antara biji kopi dan latar belakang masih dapat diamati. Hal ini menunjukkan bahwa pada tahap awal, layer konvolusi berfungsi sebagai low-level feature extractor yang menangkap fitur dasar seperti edge, tekstur, dan kontras.

Sebaliknya, pada stage 22 (Gambar 10) yang merupakan bagian akhir dari jaringan, feature map terlihat jauh lebih buram dan abstrak. Informasi visual tidak lagi menyerupai bentuk biji kopi secara langsung, melainkan telah direpresentasikan dalam bentuk aktivasi fitur tingkat tinggi. Tahap ini menunjukkan bahwa model telah menggabungkan informasi spasial dan semantik untuk mendukung proses klasifikasi dan prediksi bounding box, bukan lagi untuk mempertahankan detail visual asli.



Gambar 9. Visualisasi feature map pada stage 0 (awal) jaringan YOLOv11. Pada tahap ini, fitur visual dasar seperti bentuk dan tepi biji kopi masih terlihat jelas.

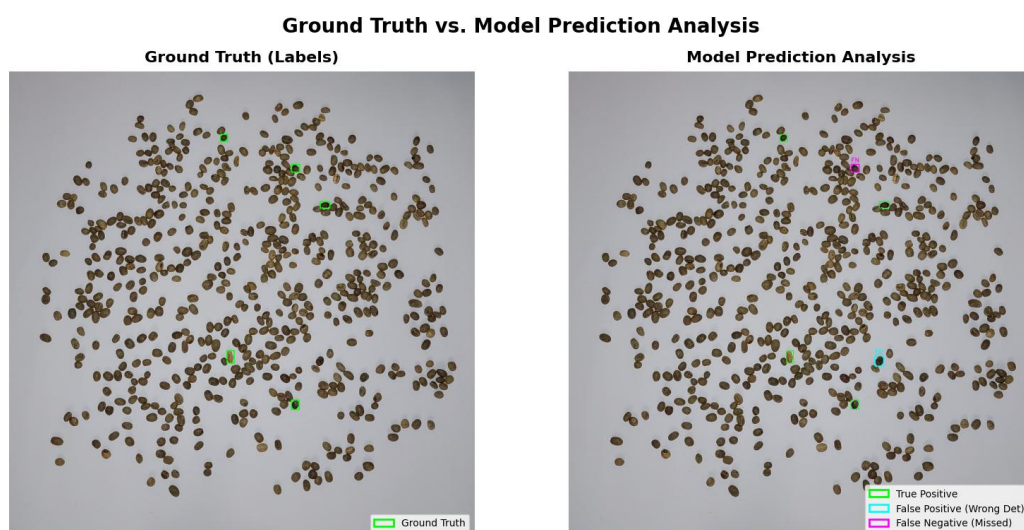


Gambar 10. Visualisasi feature map pada stage 22 (akhir) jaringan YOLOv11. Representasi fitur bersifat lebih abstrak dan buram, menunjukkan fokus model pada informasi semantik tingkat tinggi untuk proses deteksi.

## 5.6. Evaluasi False Positive dan False Negatif

Evaluasi false positive (FP) dan false negative (FN) dilakukan untuk menganalisis jenis kesalahan deteksi yang dihasilkan oleh model pada data testing. Analisis ini penting untuk memahami reliabilitas model dalam mendeteksi cacat biji kopi serta potensi dampaknya pada aplikasi quality control.

Pada Gambar 11 Gambar 12 ditampilkan visualisasi perbandingan antara ground truth (GT) dan prediksi model. Visualisasi ini membedakan hasil deteksi menjadi tiga kategori, yaitu true positive (TP) yang ditandai dengan bounding box berwarna hijau, false positive (FP) dengan warna cyan, dan false negative (FN) dengan warna magenta. Berdasarkan visualisasi tersebut, dapat diamati bahwa model secara umum mampu mendeteksi objek cacat dengan benar, yang ditunjukkan oleh dominasi bounding box TP. Namun demikian, masih terdapat beberapa kasus FP dan FN, yang menunjukkan bahwa model belum sepenuhnya bebas dari kesalahan deteksi.

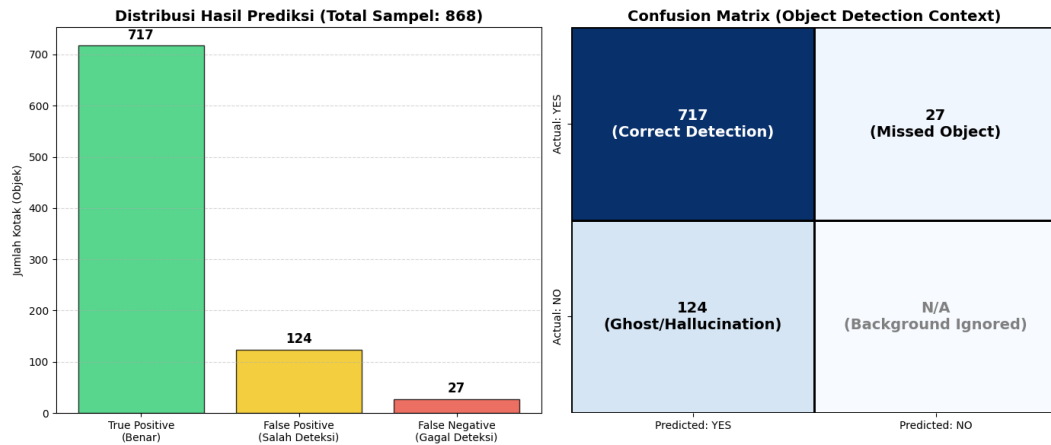


Gambar 11. Visualisasi perbandingan ground truth dan prediksi model pada data testing.

Gambar 12 selanjutnya menampilkan visualisasi kuantitatif berupa bar chart dan confusion matrix yang menggambarkan jumlah TP, FP, dan FN secara keseluruhan. Berdasarkan hasil evaluasi pada data testing, diperoleh total 717 true positive, 124 false positive, dan 27 false negative. Nilai ini menunjukkan bahwa kesalahan yang paling dominan berasal dari false positive, sementara jumlah false negative relatif lebih kecil.

Dalam konteks deteksi cacat biji kopi, false negative cenderung lebih berbahaya dibandingkan false positive. False negative berarti biji kopi yang sebenarnya cacat tidak terdeteksi oleh sistem dan berpotensi lolos ke tahap produksi selanjutnya, sehingga dapat menurunkan kualitas produk akhir. Sebaliknya, false positive hanya menyebabkan biji kopi yang sebenarnya normal diklasifikasikan sebagai cacat, yang dampaknya lebih kepada peningkatan jumlah penolakan tetapi masih dapat ditangani pada tahap sortir ulang. Oleh karena itu, meskipun model menunjukkan performa yang baik secara keseluruhan, pengurangan false negative tetap menjadi aspek penting untuk pengembangan sistem di masa mendatang.





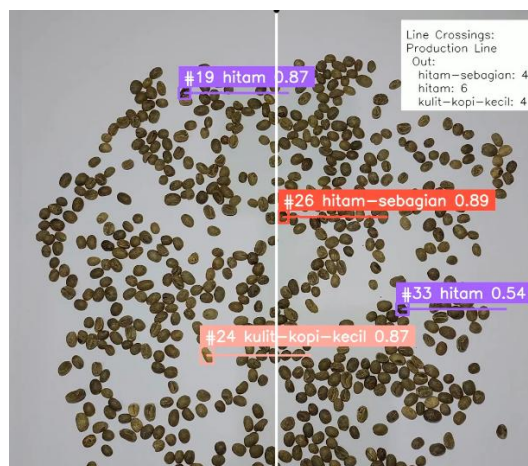
Gambar 12. Visualisasi jumlah true positive (TP), false positive (FP), dan false negative (FN) dalam bentuk bar chart dan confusion matrix pada data testing.

### 5.7. Implementasi Studi Kasus

Implementasi studi kasus dilakukan dengan menerapkan model terbaik pada data video biji kopi yang bergerak di atas conveyor. Video implementasi dapat diakses melalui tautan yang berada pada lampiran. Sebagaimana ditunjukkan pada Gambar 13, gambar tersebut merupakan *screenshot* dari hasil inferensi model pada video conveyor. Model YOLOv11 mampu mendeteksi biji kopi cacat secara akurat meskipun objek bergerak secara kontinu. Bounding box dan label kelas ditampilkan secara real-time pada setiap biji kopi yang terdeteksi.

Pada bagian tengah video terdapat sebuah garis line detector yang digunakan sebagai acuan untuk proses object counting. Setiap biji kopi yang terdeteksi dan melintasi garis tersebut akan dihitung satu kali, sehingga sistem dapat menghindari perhitungan ganda pada objek yang sama.

Selain itu, pada bagian kanan atas video ditampilkan sebuah tabel yang menunjukkan jumlah biji kopi cacat untuk setiap kelas yang telah melewati garis deteksi. Tabel ini diperbarui secara dinamis seiring dengan bertambahnya jumlah objek yang terdeteksi. Implementasi ini menunjukkan bahwa model tidak hanya mampu melakukan deteksi objek, tetapi juga dapat diintegrasikan dengan sistem penghitungan objek untuk mendukung aplikasi quality control berbasis video.



Gambar 13. Screenshot hasil implementasi model YOLOv11 pada video biji kopi yang bergerak di atas conveyor.

## VI. KESIMPULAN

Penelitian ini berhasil mengimplementasikan pendekatan object detection berbasis YOLO untuk mendeteksi cacat pada biji kopi Arabika yang terdiri dari tiga kelas, yaitu *hitam*, *hitam-sebagian*, dan *kulit-kopi-kecil*. Melalui serangkaian eksperimen terstruktur, diperoleh beberapa temuan utama yang menjadi insight dari penelitian ini.

Temuan utama (insight) dari penelitian ini menunjukkan bahwa performa model sangat dipengaruhi oleh pemilihan versi YOLO, ukuran citra input, serta penggunaan augmentasi data. YOLOv11 terpilih sebagai model terbaik karena memberikan performa paling konsisten dan unggul pada data testing dibandingkan versi lainnya. Selain itu, penggunaan image size  $1024 \times 1024$  terbukti meningkatkan nilai mAP secara signifikan, terutama pada metrik mAP50–95, yang penting untuk mendeteksi objek kecil dan padat dalam satu citra. Penggunaan augmentasi data juga secara umum meningkatkan kemampuan generalisasi model pada data testing.

Kelebihan dari pendekatan yang digunakan dalam penelitian ini adalah kemampuannya menghasilkan akurasi deteksi yang tinggi dengan waktu inferensi yang relatif cepat. Pendekatan YOLO sebagai single-stage detector memungkinkan proses deteksi dilakukan secara efisien dan dapat diterapkan pada data video secara real-time. Selain itu, model yang dikembangkan tidak hanya mampu melakukan deteksi objek pada citra statis, tetapi juga berhasil diintegrasikan dengan sistem object counting pada video conveyor, sehingga relevan untuk aplikasi quality control berbasis visual.

Keterbatasan penelitian ini terletak pada masih ditemukannya kesalahan deteksi berupa false positive dan false negative, meskipun jumlah false negative relatif kecil. Selain itu, dataset yang digunakan masih terbatas pada skenario tertentu dengan kondisi lingkungan yang relatif terkontrol. Implementasi video yang dilakukan juga masih berupa simulasi conveyor, sehingga performa model pada lingkungan industri nyata dengan variasi pencahayaan, kecepatan objek, dan kondisi visual yang lebih kompleks belum dievaluasi.

Rekomendasi untuk pengembangan selanjutnya adalah melakukan perluasan dataset dengan variasi kondisi lingkungan yang lebih beragam, serta mengeksplorasi strategi untuk menekan jumlah false negative, mengingat dampaknya yang lebih kritis pada proses quality control. Selain itu, penelitian lanjutan dapat menguji penerapan model pada sistem conveyor industri nyata serta mengintegrasikan metode tracking yang lebih lanjut untuk meningkatkan akurasi penghitungan objek pada video.

## LAMPIRAN

No	Deskripsi	Tautan
1	Code repository	<a href="https://github.com/AbiyaMakruf/TelU-Tubes-VisiKomputerFundamental-AnomaliDetection">https://github.com/AbiyaMakruf/TelU-Tubes-VisiKomputerFundamental-AnomaliDetection</a>
2	Video Conveyor	<a href="https://drive.google.com/file/d/1_IEC9ZVil0tYh-Lc91iiLzsas8Y9LQxQ/view?usp=sharing">https://drive.google.com/file/d/1_IEC9ZVil0tYh-Lc91iiLzsas8Y9LQxQ/view?usp=sharing</a>