

TESIS
ANALISIS PENGARUH AUGMENTASI DATA UNTUK
KLASIFIKASI PENYAKIT PADA DAUN TOMAT
MENGGUNAKAN MODEL YOLO11



disusun oleh
Muhammad Arif Kholis Majid
24.51.1639
Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2026

TESIS
ANALISIS PENGARUH AUGMENTASI DATA UNTUK
KLASIFIKASI PENYAKIT PADA DAUN TOMAT
MENGGUNAKAN MODEL YOLO11

ANALYSIS OF THE EFFECT OF DATA AUGMENTATION FOR
DISEASE CLASSIFICATION ON TOMATO LEAVES USING THE
YOLO11 MODEL

Diajukan untuk memenuhi salah satu syarat mencapai derajat Pascasarjana
Program Studi S2 Informatika



disuse oleh

Muhammad Arif Kholls Majid

24.51.1639

Konsentrasi : Business Intelligence

FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2026

HALAMAN PERSETUJUAN

HALAMAN PERSETUJUAN

ANALISIS PENGARUH AUGMENTASI DATA UNTUK KLASIFIKASI PENYAKIT
PADA DAUN TOMAT MENGGUNAKAN MODEL YOLO11

ANALYSIS OF THE EFFECT OF DATA AUGMENTATION FOR DISEASE
CLASSIFICATION ON TOMATO LEAVES USING THE YOLO11 MODEL

yang disusun dan diajukan oleh

Muhammad Arif Kholis Majid

24.51.1639

telah disetujui oleh Dosen Pembimbing Tesi
pada tanggal 2 Februari 2026

Dosen Pembimbing,



Dhani Ariantanto, M.Kom., Ph.D.
NIK. 190302197

HALAMAN PENGESAHAN

HALAMAN PENGESAHAN

ANALISIS PENGARUH AUGMENTASI DATA UNTUK KLASIFIKASI PENYAKIT
PADA DAUN TOMAT MENGGUNAKAN MODEL YOLO11

ANALYSIS OF THE EFFECT OF DATA AUGMENTATION FOR DISEASE
CLASSIFICATION ON TOMATO LEAVES USING THE YOLO11 MODEL

yang disusun dan diajukan oleh

Muhammad Arif Kholis Majid

24.51.1639

Telah dipertahankan di depan Dewan Penguji
pada tanggal 2 Februari 2026

Susunan Dewan Penguji

Nama Penguji

Tanda Tangan

Ema Yulia Irbil, S.T., M.Kom., Ph.D.
NIK. 190302125

Dr. Anli Suryoto, S.Kom., M.Kom.
NIK. 190302052

Dhani Ariatmanto, S.Kom., M.Kom., Ph.D.
NIK. 190302197

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer
Tanggal 2 Februari 2026

DEKAN FAKULTAS ILMU KOMPUTER



Prof. Dr. Kusriani, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Muhammad Arif Khofis Majid
NIM : 24.51.1639

Menyatakan bahwa Tesis dengan judul berikut

Analisis Pengaruh Augmentasi Data Untuk Klasifikasi Penyakit Pada Daun Tamar Menggunakan Model YOLO11

Dosen Pembimbing : Dhani Ariatmanto, M.Kom., Ph.D.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 2 Februari 2026

Yang Menyatakan,



Muhammad Arif Khofis Majid

HALAMAN PERSEMBAHAN

Alhamdulillah, segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, karunia, dan hidayah-Nya, sehingga penulis dapat menyelesaikan tesis ini dengan baik dan tepat pada waktunya. Tesis ini penulis persembahkan kepada:

1. Allah SWT yang senantiasa memberikan rahmat, nikmat, serta petunjuk-Nya sehingga penulis mampu menyelesaikan tesis ini dengan baik.
2. Kedua orang tua tercinta yang selalu memberikan doa, dukungan, semangat, serta kasih sayang yang tiada henti kepada penulis dalam menyelesaikan tesis ini.
3. Bapak Dhani Ariatmanto, M.Kom., Ph.D. selaku dosen pembimbing yang telah meluangkan waktu, memberikan arahan, masukan, saran, serta bimbingan selama proses penyusunan tesis ini.
4. Teman-teman Magister Informatika Angkatan 32 yang telah menjadi rekan seperjuangan selama masa perkuliahan dan memberikan banyak kenangan berharga.
5. Seluruh pihak yang telah membantu dan mendukung penulis dalam proses penyusunan tesis ini, yang tidak dapat disebutkan satu per satu, penulis ucapkan terima kasih yang sebesar-besarnya.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, hidayah, dan karunia-Nya, sehingga penulis dapat menyelesaikan tesis ini dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada junjungan Nabi Muhammad SAW, beserta keluarga, sahabat, dan seluruh pengikut beliau hingga akhir zaman.

Tesis ini disusun sebagai salah satu syarat untuk menyelesaikan studi pada Program Magister di Universitas Amikom Yogyakarta serta untuk memperoleh gelar Magister Komputer.

Penyusunan tesis ini tidak terlepas dari bantuan, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis menyampaikan terima kasih kepada:

1. Bapak Prof. Dr. M. Suyanto, M.M. selaku Rektor Universitas Amikom Yogyakarta.
2. Bapak Robert Marco, M.T., Ph.D. selaku Ketua Program Studi S2 Informatika Universitas Amikom Yogyakarta.
3. Bapak Theopilus Bayu Sasongko, S.Kom., M.Eng. selaku dosen pembimbing yang telah memberikan arahan, masukan, saran, serta bimbingan selama proses penyusunan tesis ini.
4. Bapak dan Ibu dosen Universitas Amikom Yogyakarta yang telah memberikan ilmu, wawasan, dan pengalaman selama masa perkuliahan.
5. Seluruh pihak yang telah memberikan bantuan, baik berupa dukungan moral, pemikiran, maupun tenaga dalam penyelesaian tesis ini.

Penulis juga menyampaikan terima kasih kepada semua pihak yang tidak dapat disebutkan satu per satu. Semoga Allah SWT membalas segala kebaikan dan melimpahkan rahmat serta hidayah-Nya kepada kita semua.

Penulis telah berusaha semaksimal mungkin dalam menyusun tesis ini. Namun, penulis menyadari bahwa tesis ini masih memiliki kekurangan, baik dari

segi isi maupun tata bahasa. Oleh sebab itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan dan penyempurnaan karya ini.

Akhir kata, dengan segala kerendahan hati, penulis berharap semoga tesis ini dapat memberikan manfaat bagi para pembaca dan pihak-pihak yang membutuhkan. Aamiin.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 2 Februari 2026

Penulis



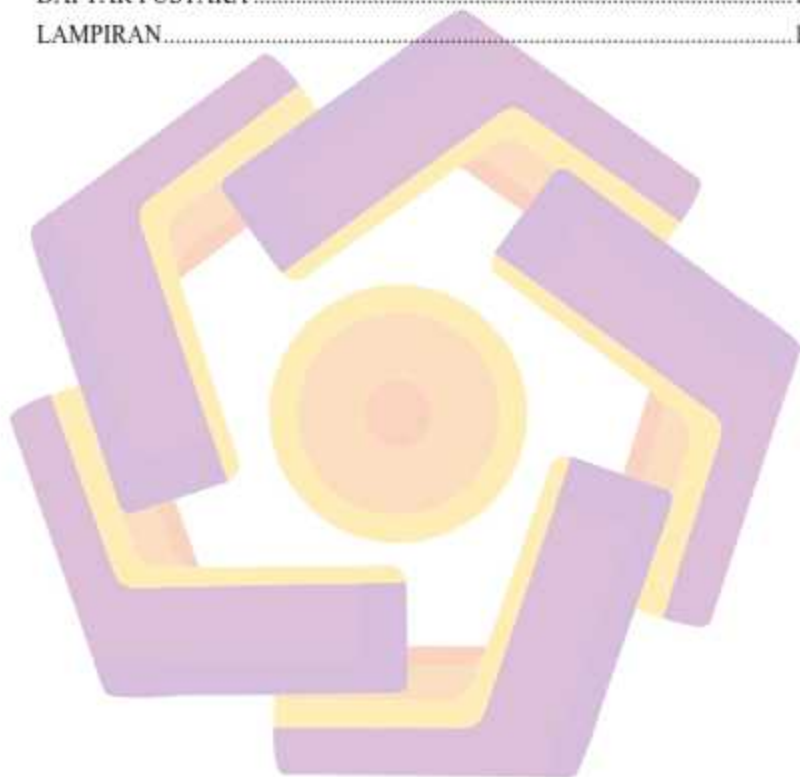
DAFTAR ISI

HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN KEASLIAN TESIS	iv
HALAMAN PERSEMBAHAN	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR LAMPIRAN.....	xiv
DAFTAR LAMBANG DAN SINGKATAN	xv
DAFTAR ISTILAH.....	xvi
INTISARI	xvii
ABSTRACT	xviii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	5
BAB 2 TINJAUAN PUSTAKA	6
2.1 Tinjauan Pustaka.....	6
2.2 Keaslian Penelitian	10
2.3 Landasan Teori	15
2.3.1 Augmentasi Data	15
2.3.1.1 Advanced Mixing Augmentation.....	17
2.3.1.2 Photometric Augmentation	19
2.3.1.3 Spatial Transformations	21
2.3.2 Tomat	23
2.3.2.1 Penyakit Daun Tomat	24
2.3.3 Computer Vision.....	29

2.3.4 CNN (Convolutional Neural Networks)	30
2.3.5 Klasifikasi dengan YOLO11	31
2.3.5.1 Paramater tuning YOLO	35
2.3.6 Evaluasi Kinerja Model	38
2.3.6.1 Akurasi	38
2.3.6.2 Presisi	38
2.3.6.3 Recall	39
2.3.6.4 F1-Score	39
BAB 3 METODE PENELITIAN	40
3.1 Jenis, Sifat, dan Pendekatan Penelitian	40
3.2 Metode Pengumpulan Data	40
3.3 Metode Analisis Data	41
3.4 Alur Penelitian	43
BAB 4 HASIL PENELITIAN DAN PEMBAHASAN	46
4.1 Dataset	46
4.2 Split Data	48
4.3 Augmentasi Data	49
4.3.1 Non Augmentation	49
4.3.2 Advanced Mixing Augmentation	51
4.3.3 Photometric Augmentation	55
4.3.4 Spatial Transformations	60
4.3.5 All Augmentation	65
4.4 Train Model YOLO11	68
4.4.1 Non Augmentation	68
4.4.2 Advanced Mixing Augmentation	70
4.4.3 Photometric Augmentation	73
4.4.4 Spatial Transformations	75
4.4.5 All Augmentation	77
4.5 Evaluasi	79
4.5.1 Non Augmentation	79

4.5.1.1 Hasil Evaluasi	80
4.5.1.2 Hasil Precision, Recall, F1-Score	83
4.5.1.3 Confusion Matrix	85
4.5.1.4 Analisis Confidence Score	87
4.5.2 Advanced Mixing Augmentation	89
4.5.2.1 Hasil Evaluasi	90
4.5.2.2 Hasil Precision, Recall, F1-Score	92
4.5.2.3 Confusion Matrix	94
4.5.2.4 Analisis Confidence Score	97
4.5.3 Photometric Augmentation	100
4.5.3.1 Hasil Evaluasi	101
4.5.3.2 Hasil Precision, Recall, F1-Score	103
4.5.3.3 Hasil Confusion Matrix	105
4.5.3.4 Analisis Confidence Score	108
4.5.4 Spatial Transformations	112
4.5.4.1 Hasil Evaluasi	112
4.5.4.2 Hasil Precision, Recall, F1-Score	115
4.5.4.3 Hasil Confusion Matrix	117
4.5.4.4 Analisis Confidence Score	120
4.5.5 All Augmentation	123
4.5.5.1 Hasil Evaluasi	124
4.5.5.2 Hasil Precision, Recall, F1-Score	127
4.5.5.3 Hasil Confusion Matrix	128
4.5.5.4 Analisis Confidence Score	131
4.6 Perbandingan Hasil	134

4.6.1 Perbandingan Hasil dari 5 Skenario Augmentasi	134
4.6.2 Perbandingan Hasil dengan Penelitian Terdahulu	138
BAB 5 PENUTUP	143
5.1 Kesimpulan	143
5.2 Saran	144
DAFTAR PUSTAKA	145
LAMPIRAN.....	154



DAFTAR TABEL

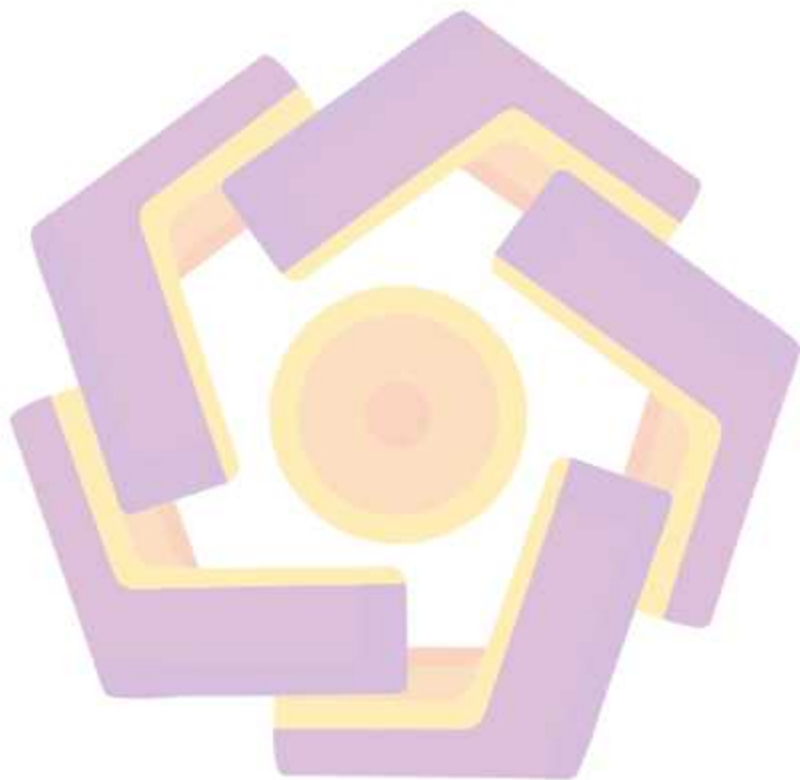
Tabel 2.1 Matriks literatur review dan posisi penelitian.....	10
Tabel 2.2 Tabel Confusion Matrix.....	38
Tabel 4.1 Informasi Dataset.....	46
Tabel 4.2 Tabel Hasil Split Data.....	49
Tabel 4.3 Tabel Hasil Dataset Non Augmentation.....	50
Tabel 4.4 Hasil Gambar Setelah Advanced Mixing Augmentation.....	54
Tabel 4.5 Hasil Gambar Setelah Photometric Augmentation.....	59
Tabel 4.6 Hasil Gambar Setelah Augmentasi Spatial Transformations.....	64
Tabel 4.7 Gambar Hasil All Augmentation.....	67
Tabel 4.8 Tabel Parameter Skenario Non Augmentation.....	70
Tabel 4.9 Tabel Parameter Skenario Advanced Mixing Augmentation.....	72
Tabel 4.10 Tabel Parameter Skenario Photometric Augmentation.....	74
Tabel 4.11 Tabel Parameter Skenario Spatial Transformations.....	77
Tabel 4.12 Tabel Parameter Skenario All Augmentation.....	79
Tabel 4.13 Tabel Hasil 10 Trial Skenario Non Augmentation.....	80
Tabel 4.14 Tabel Hasil Evaluasi Non Augmentation.....	81
Tabel 4.15 Tabel Metriks Precision, Recall, F1-Score per kelas.....	83
Tabel 4.16 Semua Trial Skenario Advanced Mixing Augmentation.....	90
Tabel 4.17 Tabel Hasil Evaluasi Advanced Mixing Augmentation.....	90
Tabel 4.18 Tabel Hasil Precision, Recall, F1-Score.....	92
Tabel 4.19 Hasil Skenario Photometric Augmentation.....	101
Tabel 4.20 Hasil Skenario photometric Augmentation Trail ke 4.....	101
Tabel 4.21 Hasil Precision, Recall, F1-Score Photometric Augmentation.....	103
Tabel 4.22 Hasil 10 Trial Skenario Spatial Transformations.....	113
Tabel 4.23 Tabel Hasil Evaluasi Skenario Spatial Transformations Trial ke 6.....	113
Tabel 4.24 Hasil Precision, Recall, F1-Score Spatial Transformations.....	115
Tabel 4.25 Hasil Semua Trial Skenario All Augmentation.....	124
Tabel 4.26 Hasil Evaluasi Skenario All Augmentation Trial ke 8.....	124
Tabel 4.27 Hasil Precision, Recall, F1-Score All Augmentation.....	127
Tabel 4.28 Tabel Perbandingan Hasil Penelitian.....	135
Tabel 4.29 Tabel Perbandingan Hasil dengan Penelitian Terdahulu.....	138

DAFTAR GAMBAR

Gambar 2. 1 Penyakit Bacterial Spot.....	25
Gambar 2. 2 Early Blight.....	25
Gambar 2. 3 Penyakit Late Blight pada Daun Tomat.....	25
Gambar 2. 4 Leaf Mold.....	26
Gambar 2. 5 Two-Spotted Spider Mite.....	26
Gambar 2. 6 Tomato Yellow Leaf Curl Virus.....	27
Gambar 2. 7 Tomato Mosaic Virus.....	27
Gambar 2. 8 Target Spot.....	28
Gambar 2. 9 Septoria Leaf Spot.....	28
Gambar 2. 10 Daun Tomat Sehat.....	29
Gambar 2. 11 Arsitektur CNN.....	30
Gambar 2. 12 Arsitektur YOLO11.....	34
Gambar 3. 1 Diagram Alur Penelitian.....	45
Gambar 4. 1 Jenis Penyakit pada Daun Tomat.....	48
Gambar 4. 2 Source Code Advanced Mixing Augmentation.....	51
Gambar 4. 3 Hasil dari Advanced Mixing Augmentation.....	55
Gambar 4. 4 Source Code Photometric Augmentation.....	56
Gambar 4. 5 Hasil Photometric Augmentation.....	60
Gambar 4. 6 Source Code Spatial Transformations.....	61
Gambar 4. 7 Hasil Augmentasi Spatial Transformations.....	64
Gambar 4. 8 Source Code Skenario Non Augmentation.....	69
Gambar 4. 9 Source Code Klasifikasi Advanced Mixing Augmentation.....	71
Gambar 4. 10 Source Code Skenario Photometric Augmentation.....	73
Gambar 4. 11 Source Code Skenario Spatial Transformations.....	76
Gambar 4. 12 Source Code Skenario All Augmentation.....	78
Gambar 4. 13 Grafik Train dan Val Loss trial ke 9 Non Augmentation.....	83
Gambar 4. 14 Confusion Matriks Non Augmentation.....	85
Gambar 4. 15 Hasil Confidence Gambar.....	87
Gambar 4. 16 Gambar Visualisasi Label Prediksi Non Augmentation.....	88
Gambar 4. 17 Train/loss dan Val/loss skenario Advanced Mixing Augmentation.....	92
Gambar 4. 18 Confusion Matrix Advanced Mixing Augmentation.....	95
Gambar 4. 19 Hasil Confidence Score Setiap Gambar.....	97
Gambar 4. 20 Visualisasi Lable Prediksi.....	98
Gambar 4. 21 Train Val Loss Photometric Augmentation.....	103
Gambar 4. 22 Gambar Confusion Matrix Skenario Photometric Augmentation.....	106
Gambar 4. 23 Hasil Confidence Score Setiap Gambar.....	109
Gambar 4. 24 Visualisasi Label Prediksi Skenario Photometric Augmentation.....	109
Gambar 4. 25 Grafik Train Val Loss Spatial Transformations.....	115
Gambar 4. 26 Gambar Confusion Matrix Skenario Spatial Transformations.....	118
Gambar 4. 27 Hasil Confidence Score Skenario Spatial Transformations.....	120
Gambar 4. 28 Visualisasi Label Prediksi Skenario Spatial Transformations.....	121
Gambar 4. 29 Train/Loss dan Val/Loss All Augmentation.....	126
Gambar 4. 30 Gambar Confusion Matirx Skenario All Augmentation.....	129
Gambar 4. 31 Hasil Confidence Score Scenario All Augmentation.....	131
Gambar 4. 32 Visualisasi Label Prediksi Skenario All Augmentation.....	132

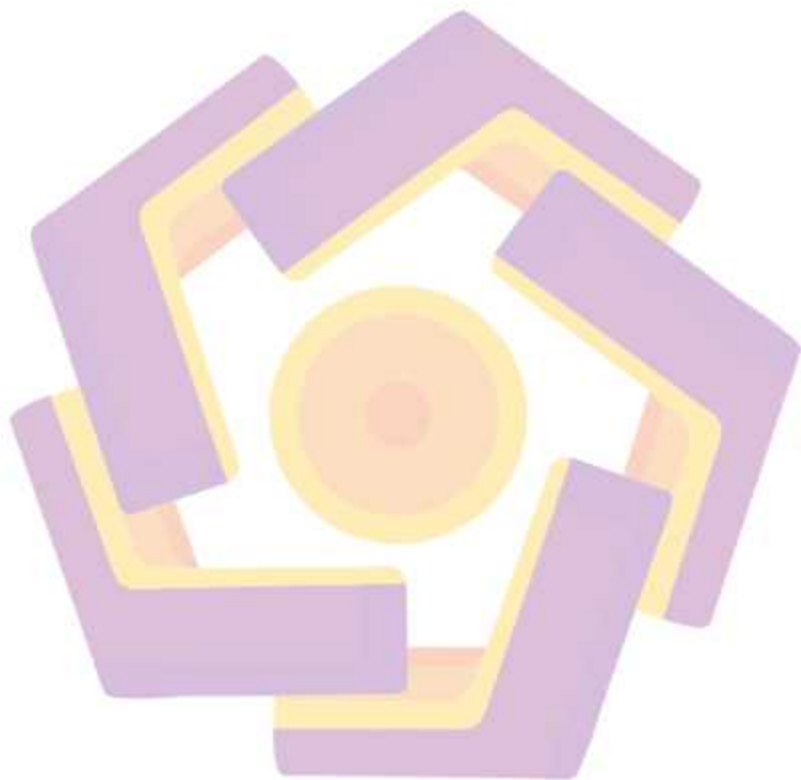
DAFTAR LAMPIRAN

Lampiran 1 Dataset.....	154
-------------------------	-----



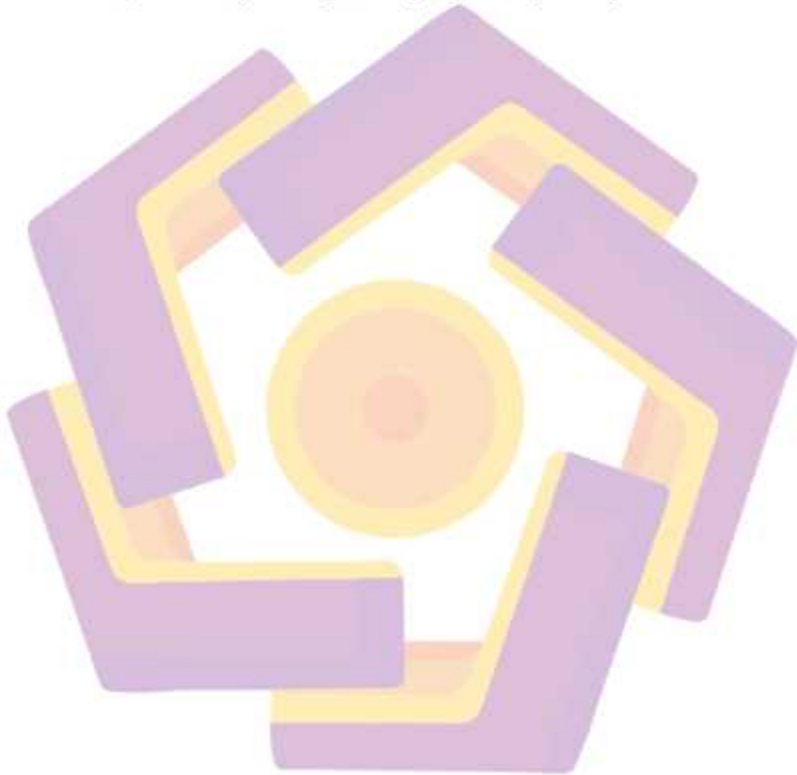
DAFTAR LAMBANG DAN SINGKATAN

YOLO You Only Look Once



DAFTAR ISTILAH

1. YOLO11 : Model deteksi dan klasifikasi objek berbasis konvolusi yang digunakan secara real-time
2. Augmentasi Data : Teknik yang digunakan untuk memperbesar ukuran dataset dengan menciptakan variasi gambar melalui transformasi gambar seperti rotasi, pemotongan, dan perubahan pencahayaan.



INTISARI

Tomat (*Solanum lycopersicum* L.) merupakan komoditas hortikultura penting di Indonesia, dengan budidaya yang luas dan berperan besar dalam pendapatan petani. Serangan penyakit pada daun tomat dapat berdampak langsung terhadap produktivitas dan kualitas tanaman, yang berpotensi menyebabkan kerugian ekonomi serta fluktuasi harga. Penelitian ini menggunakan model YOLO11n-cls untuk klasifikasi penyakit daun tomat berbasis citra, dengan membandingkan lima skenario augmentasi data yaitu non augmentation, advanced mixing augmentation, photometric augmentation, spatial transformations, dan all augmentation. Hasil evaluasi menunjukkan bahwa penerapan teknik augmentasi dapat meningkatkan kinerja model, dengan spatial transformations memberikan hasil terbaik. Akurasi mencapai 0.987250, dengan precision dan recall masing-masing 0.994, berkat kemampuan teknik ini dalam menangani variasi geometris pada gambar seperti rotasi dan pemotongan, yang relevan dalam pengklasifikasian penyakit yang muncul dalam berbagai orientasi daun. Sebaliknya, photometric augmentation menunjukkan hasil terendah dengan akurasi 0.95425, karena perubahan pencahayaan dan kontras mengaburkan perbedaan visual yang halus antara penyakit yang mirip, seperti Bacterial Spot yang salah diklasifikasikan sebagai Yellow Leaf Curl Virus. Advanced mixing augmentation memberikan akurasi 0.98015, dengan precision dan recall yang tinggi (0.991), namun beberapa kesalahan klasifikasi masih terjadi, seperti kebingungan antara Bacterial Spot dan Yellow Leaf Curl Virus. Gabungan teknik augmentasi dalam all augmentation menghasilkan akurasi 0.981, dengan precision dan recall 0.987, meskipun sedikit lebih rendah dibandingkan dengan spatial transformations. Temuan ini menegaskan bahwa augmentasi berbasis transformasi geometris adalah yang paling efektif untuk meningkatkan kinerja model YOLO11, sementara photometric augmentation perlu digunakan dengan hati-hati agar tidak mengganggu fitur diagnostik pada daun tomat.

Kata kunci: Augmentasi data, YOLO11, klasifikasi penyakit, daun tomat.

ABSTRACT

Tomato (*Solanum lycopersicum* L.) is an important horticultural commodity in Indonesia, with significant cultivation and a key role in farmers' income. Disease attacks on tomato leaves can directly impact productivity and quality, leading to economic losses and price fluctuations. This study applies the YOLO11n-cs model for tomato leaf disease classification based on image data, comparing five data augmentation scenarios: non-augmentation, advanced mixing augmentation, photometric augmentation, spatial transformations, and all augmentation. The results show that data augmentation improves model performance, with spatial transformations yielding the best results. The accuracy reached 0.987250, with precision and recall of 0.994, due to the technique's ability to handle geometric variations like rotation and cropping. In contrast, photometric augmentation resulted in the lowest performance with an accuracy of 0.95425, as changes in lighting and contrast blurred subtle visual differences between similar diseases, such as Bacterial Spot misclassified as Yellow Leaf Curl Virus. Advanced mixing augmentation improved accuracy to 0.98015 with precision and recall of 0.991, but still showed misclassifications like confusion between Bacterial Spot and Yellow Leaf Curl Virus. All augmentation reached 0.981 accuracy, with precision and recall of 0.987, slightly lower than spatial transformations. These findings confirm that augmentation based on geometric transformations is most effective in enhancing YOLO11 model performance, while photometric augmentation should be carefully controlled to avoid interfering with diagnostic features on tomato leaves.

Keyword: Data augmentation, YOLO11, disease classification, tomato leaves.

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Tomat (*Solanum lycopersicum* L.) adalah komoditas hortikultura penting dari famili Solanaceae yang berasal dari Amerika Selatan dan kini dibudidayakan secara luas di seluruh dunia. Tanaman ini memiliki batang berbulu halus, daun majemuk, dan menghasilkan buah merah atau kuning yang kaya vitamin C dan likopen, bermanfaat untuk pencegahan penyakit kardiovaskular dan kanker [1]. Tomat tumbuh optimal di iklim hangat dengan sinar matahari cukup serta dipengaruhi oleh suhu, kelembapan, dan kualitas tanah. Budidaya bisa di lahan terbuka maupun rumah kaca dengan teknik hidroponik untuk kontrol lingkungan yang lebih baik [2].

Tanaman tomat rentan terhadap penyakit yang disebabkan bakteri, jamur, dan virus sehingga menurunkan hasil dan kualitas buah. Tanpa klasifikasi dini, penyakit dapat menyebar luas dan memicu kerugian ekonomi besar [3]. Karena itu, klasifikasi penyakit secara akurat sangat penting untuk pengelolaan yang efektif serta pencegahan kerugian lanjutan [4].

Berdasarkan Badan Pusat Statistik (2024) [5], luas lahan budidaya tomat Indonesia mencapai 58.782,74 hektar dengan total produksi 11.466.079,40 kuintal. Potensi ini menopang ketahanan pangan dan kesejahteraan petani. Namun pada minggu kedua tanggal 06 Januari 2025, harga tomat di Pasar Wonosari, Bondowoso, naik 71,43% dari Rp 7.000 menjadi Rp 12.000 per kilogram [6].

Kenaikan dipengaruhi permintaan, pasokan terbatas, biaya produksi meningkat, dan tanaman terjangkit penyakit [6].

Kemajuan computer vision dan machine learning membuka peluang baru untuk klasifikasi penyakit tanaman, dengan pendekatan deep learning yang meningkatkan akurasi dan efisiensi [7]. Salah satu model unggulan adalah YOLO11 (You Only Look Once, versi 11), yang cepat dan akurat dalam klasifikasi objek [8]. YOLO11 dapat mengidentifikasi dan mengklasifikasikan kondisi penyakit pada daun tomat secara real-time, memungkinkan tindakan preventif segera [9]. Model ini efisien dalam mengelola data gambar besar dan variatif, serta mengurangi kesalahan klasifikasi yang sering terjadi pada pemeriksaan manual, bahkan dalam kondisi lapangan dan pencahayaan yang berbeda [10].

Augmentasi data berpengaruh signifikan pada performa klasifikasi penyakit daun tomat menggunakan YOLO11 [11]. Teknik ini membuat model lebih robust terhadap variasi sudut pengambilan, pencahayaan, dan posisi daun [12]. Dengan memperbesar dataset secara virtual, augmentasi menurunkan risiko overfitting sehingga model belajar pola penting, bukan sekadar menghafal [13].

Meskipun YOLO dirancang untuk multi-object detection, penggunaannya dalam klasifikasi penyakit daun tomat tetap efektif meskipun hanya ada single object. YOLO mampu mengklasifikasikan objek tunggal dengan cepat dan akurat, ideal untuk klasifikasi real-time dengan menangani variasi gambar secara efisien [8][9][10]. Teknologi klasifikasi penyakit daun tomat berbasis YOLO11 memanfaatkan dataset gambar daun sehat dan terinfeksi yang diambil dengan kamera atau drone. Gambar digunakan untuk melatih model agar efektif

mengklasifikasikan penyakit di lingkungan pertanian yang kompleks [14]. YOLO11 dengan akurasi tinggi membantu mengurangi kesalahan dalam pengklasifikasian yang sering terjadi pada metode konvensional atau pemeriksaan visual manusia yang rentan terhadap kondisi cuaca [15].

Penelitian sebelumnya, seperti yang dilakukan oleh Abdullah (2024) [16] menggunakan YOLOV8s untuk klasifikasi penyakit pada daun tomat, berhasil mencapai presisi 93,2 % dan recall 90,3 %, namun menggunakan model yang berbeda dari YOLO11, yang memiliki keunggulan dalam klasifikasi. Sementara itu, Lestari et al. (2024) [17] mengembangkan DenseNet-PSO yang menunjukkan akurasi tinggi (97.39%), namun masih menggunakan teknik augmentasi standar yang terbatas. Hal ini menunjukkan bahwa teknik augmentasi yang lebih kompleks, seperti Advanced Mixing, Photometric, dan Spatial Transformations dapat memberikan robustness lebih tinggi dalam menghadapi variasi pencahayaan dan sudut pandang di dunia nyata.

Selain itu, penelitian Andi Nurdin et al. (2024) [18] menunjukkan bahwa penggunaan Inception V3 dengan Transfer Learning dapat mempercepat pelatihan, namun tidak mendalami Parameter tuning yang dapat lebih meningkatkan performa model. Penelitian yang diajukan berfokus pada penyetalan hyperparameter, seperti learning rate dan ukuran batch, serta probabilitas dan kekuatan augmentasi, yang penting untuk menghindari overfitting atau underfitting, dan memastikan generalization yang optimal. Dengan menggunakan YOLO11 dan augmentasi data lanjutan, penelitian ini bertujuan untuk mengatasi kekurangan di penelitian sebelumnya, serta meningkatkan performa model dalam mengklasifikasikan

penyakit pada daun tomat dengan robustness terhadap kondisi lapangan yang bervariasi.

1.2 Rumusan Masalah

Rumusan masalah yang akan dikaji pada penelitian ini yaitu :

1. Sejauh mana penerapan teknik augmentasi data mempengaruhi akurasi model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat?
2. Teknik augmentasi data mana yang terbaik untuk meningkatkan kinerja model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat?
3. Bagaimana strategi augmentasi data dapat mempengaruhi daya tahan (robustness) model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat dalam berbagai kondisi pencahayaan dan sudut pandang?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini memuat tentang :

1. Dataset yang tersedia tidak mencakup variasi kondisi nyata, seperti perubahan pencahayaan, sudut pandang kamera, atau kerusakan daun yang berbeda-beda.
2. Penelitian ini dilakukan dengan hardware dan waktu pelatihan yang terbatas, sehingga parameter model dan teknik augmentasi disesuaikan dengan kapasitas yang ada.

1.4 Tujuan Penelitian

Bagian ini memuat penjelasan mengenai tujuan penelitian :

1. Menganalisis pengaruh teknik augmentasi data terhadap performa model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat.

2. Menilai teknik augmentasi data yang terbaik dalam meningkatkan kinerja model YOLO11 dalam klasifikasi penyakit pada daun tomat.
3. Mengukur pengaruh strategi augmentasi data terhadap ketahanan (robustness) model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat di berbagai kondisi pencahayaan dan sudut pandang.

1.5 Manfaat Penelitian

Bagian ini memuat penjelasan tentang manfaat penelitian :

1. Penelitian ini dapat membantu petani dalam mengklasifikasi penyakit pada daun tomat dengan lebih cepat dan akurat, yang dapat meningkatkan hasil pertanian dan mengurangi kerugian akibat penyakit.
2. Penelitian ini menambah wawasan dalam bidang teknologi pengolahan citra untuk klasifikasi penyakit pada tanaman, serta memberikan dasar untuk penelitian lebih lanjut di bidang tersebut.
3. Hasil penelitian ini bisa dijadikan referensi untuk mengembangkan aplikasi berbasis AI dalam memantau kesehatan tanaman, yang bisa meningkatkan efisiensi dan mengurangi penggunaan pestisida.
4. Penelitian ini memberikan pemahaman tentang penerapan teknik augmentasi data pada model YOLO, yang dapat diterapkan di berbagai bidang seperti agrikultur, kesehatan, keamanan, dan industri lainnya.

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian Abdullah (2024) [16] mengembangkan model YOLOV8s untuk klasifikasi penyakit pada daun tomat. Model ini berhasil mencapai presisi 93,2 % dan recall 90,3 %, lebih unggul dibandingkan dengan YOLOV5 yang mencapai presisi 90,1 % dan recall 87,1 % dan Faster R-CNN yang hanya presisi 76,8 % dan recall 81,7 %. Keunggulan dari YOLOV8s adalah akurasi yang tinggi dan kecepatan inferensi 141,5 FPS, namun model ini lebih kecil dibandingkan dengan YOLOV5 dan Faster R-CNN, yang mengorbankan ukuran model yang lebih besar pada YOLOV5. Penelitian yang diajukan dapat meningkatkan performa ini dengan menambahkan YOLO11 yang dilengkapi dengan augmentasi data canggih untuk menghasilkan robustness yang lebih baik pada variasi kondisi lapangan seperti pencahayaan dan sudut pandang.

Selanjutnya penelitian Lestari et al. (2024) [17] menggunakan DenseNet-PSO untuk fine-tuning arsitektur DenseNet dalam mendeteksi penyakit pada daun tomat. Hasil penelitian menunjukkan bahwa DenseNet-PSO memberikan akurasi yang lebih tinggi (97,39%) dibandingkan dengan DenseNet-121 (90,31%), ResNet-101 (87,95%), dan InceptionV3 (91,69%). Keunggulannya adalah model yang lebih efisien dengan parameter lebih sedikit, konvergensi lebih cepat, dan penggunaan memori yang lebih sedikit. Namun, kelemahannya adalah teknik augmentasi yang digunakan masih standar, yang mungkin tidak cukup untuk menangani variasi

kondisi di dunia nyata. Penelitian yang diajukan bertujuan untuk mengatasi hal ini dengan menggunakan augmentasi lanjutan yang lebih robust terhadap perubahan pencahayaan dan posisi dalam deteksi penyakit.

Penelitian ke tiga oleh Kumar et al. (2024) [19] mengembangkan ensemble model dengan Exponential Moving Average (EMA) dan Enhanced Weighted Gradient Optimization (EWGO) untuk klasifikasi penyakit pada daun tomat, menggunakan model VGG-16 dan NASNet. Hasilnya menunjukkan akurasi 98.7% dengan precision 97.9%, recall 98.6%, dan F1-score 98.7%. Keunggulannya adalah akurasinya yang sangat tinggi dan penggunaan model ensemble untuk menangani variasi penyakit dengan efisien. Namun, kelemahannya adalah penggunaan ensemble model yang kompleks, yang memerlukan lebih banyak sumber daya komputasi. Penelitian ini tidak menganalisis secara mendalam pengaruh augmentasi data terhadap robustness model terhadap perubahan cahaya dan sudut pandang. Penelitian yang diajukan bertujuan untuk menambahkan augmentasi lanjutan dan analisis terhadap robustness model dalam menghadapi kondisi dunia nyata yang lebih dinamis.

Penelitian oleh Marchel Adias Pradana et al. (2023) [20] menggunakan CNN dengan arsitektur VGG16 untuk klasifikasi penyakit pada daun tomat. Hasil penelitian menunjukkan akurasi pengujian 91% dan akurasi validasi 92%. Keunggulannya adalah VGG16 yang efisien untuk ekstraksi fitur, namun performanya lebih rendah ketika diterapkan pada dataset yang lebih beragam. Penelitian ini menggunakan augmentasi data standar (seperti normalisasi, rotasi, dan flipping) yang terbatas. Penelitian yang diajukan dapat meningkatkan hasil ini

dengan menggunakan teknik augmentasi lanjutan yang lebih kompleks, seperti Advanced Mixing dan Photometric Augmentation, untuk memperbaiki robustness model terhadap variasi di dunia nyata.

Penelitian Andi Nurdin et al. (2024) [18] mengaplikasikan CNN dengan Inception V3 dan melakukan optimasi dengan tiga algoritma optimizer (Adam, SGD, RMSProp) untuk deteksi penyakit pada daun tomat. Hasil terbaik diperoleh menggunakan Adam optimizer dengan akurasi 93.8%, precision 93.9%, recall 93.8%, dan F1-score 93.7%. Keunggulannya adalah penggunaan Inception V3 dan Transfer Learning yang mempercepat pelatihan model. Namun, penelitian ini lebih berfokus pada pemilihan model dan optimizer, sementara eksplorasi tentang tuning hyperparameter kurang mendalam. Penelitian yang diajukan dapat meningkatkan pendekatan ini dengan analisis lebih mendalam tentang Parameter tuning dan menggunakan YOLO11 yang lebih efisien dalam deteksi real-time.

Penelitian oleh Ibnu Topan Adib Amrulloh et al. (2024) [21] menggunakan YOLOv8 dengan teknik data augmentasi seperti horizontal flipping, jittering scale, dan random brightness untuk mendeteksi penyakit daun tebu. Hasilnya menunjukkan peningkatan nilai mAP dari 40.3% menjadi 50.5% setelah augmentasi data. Keunggulannya adalah augmentasi data yang memperkaya dataset dan mengatasi overfitting, serta meningkatkan generalisasi model. Namun, teknik augmentasi yang digunakan terbatas pada flipping, scale jittering, dan brightness, yang masih bisa diperluas. Penelitian yang diajukan berencana untuk menggunakan YOLO11 dengan augmentasi data lanjutan, seperti Advanced Mixing dan

Photometric Augmentation, untuk meningkatkan robustness model terhadap variasi pencahayaan dan kondisi dunia nyata.

Penelitian Theopilus Bayu Sasongko et al. (2023) [22] menggunakan CNN dengan Transfer Learning menggunakan MobileNet, EfficientNet, dan NASNetMobile untuk dataset kecil. Hasilnya menunjukkan peningkatan akurasi sebesar 0.03% hingga 0.1% dengan penerapan random erase dan zoom range augmentasi. Keunggulannya adalah penggunaan model yang efisien dengan parameter lebih sedikit, cocok untuk dataset kecil. Namun, kelemahannya adalah dataset yang sangat kecil, yang membatasi kemampuan model untuk belajar fitur lebih kompleks. Penelitian yang diajukan akan mengatasi hal ini dengan menggunakan augmentasi lebih kompleks, YOLO11, dan analisis pengaruh augmentasi terhadap kinerja model dalam kondisi yang lebih bervariasi.

Penelitian terakhir dari Sri Silpa Padmanabhuni et al. (2022) menggunakan GAN dengan Meta-Learning untuk sintesis data daun tomat, yang menghasilkan gambar sintetis dengan akurasi 99.29% untuk identifikasi penyakit. Keunggulannya adalah Meta-Learning GAN yang menghasilkan data sintetis dengan lebih sedikit noise dibandingkan dengan GAN tradisional, serta menghasilkan gambar dengan kualitas lebih tinggi menggunakan kernel filtering dan transformasi geometrik. Kelemahannya adalah proses training dan pembuatan gambar sintetis yang membutuhkan banyak waktu dan sumber daya. Penelitian yang diajukan menggunakan YOLO11 dengan augmentasi data lanjutan untuk meningkatkan robustness model terhadap variasi pencahayaan dan sudut pandang, serta lebih efisien dalam deteksi real-time.

2.2 Keaslian Penelitian

Tabel 2.1 Matriks literatur review dan posisi penelitian
 ANALISIS PENGARUH AUGMENTASI DATA UNTUK KLASIFIKASI PENYAKIT PADA DAUN TOMAT
 MENGGUNAKAN MODEL YOLO11

No	Judul Penelitian	Nama Peneliti, Tahun, Index	Metode Penelitian	Hasil	Keunggulan dan Kelemahan	Perbandingan
1	A Deep-Learning-Based Model for the Detection of Diseased Tomato Leaves	Abdullah (2024) Q1	Model YOLOV8s	YOLOV8s mencapai mAP 92.5%, mengalahkan YOLOV5 (89.6%) dan Faster R-CNN (77.3%)	Keunggulan: Presisi 93,2 % dan recall 90,3 % tinggi 141.5 FPS Kelemahan: Ukuran model lebih kecil dibandingkan YOLOV5 dan Faster R-CNN, tetapi mengorbankan ukuran model yang lebih besar pada YOLOV5	Penelitian ini menambahkan model YOLO11 dengan augmentasi data menghasilkan performa yang lebih baik
2	Exploring DenseNet architectures with PSO: efficient tomato leaf disease detection	Lestari et al. (2024) Q2	DenseNet-PSO dengan optimasi partikel swarm untuk fine-tuning	DenseNet-PSO mencapai akurasi lebih tinggi (97.39%) dibandingkan dengan model	Keunggulan: Model lebih efisien dengan lebih sedikit parameter, lebih cepat dalam konvergensi, dan	Menggunakan teknik augmentasi yang lebih tahan terhadap perubahan kondisi pencahayaan dan posisi, yang

			arsitektur DenseNet	lain seperti DenseNet-121 (90.31%), ResNet-101 (87.95%), dan InceptionV3 (91.69%).	penggunaan lebih sedikit memori Kelemahan: Teknik augmentasi masih bersifat standar dan tidak sepenuhnya cukup untuk menangani kerumitan variasi kondisi di dunia nyata.	meningkatkan robustness model terhadap perubahan lingkungan yang lebih luas.
3	Improved Tomato Leaf Disease Classification through Adaptive Ensemble Models with Exponential Moving Average Fusion and Enhanced Weighted Gradient Optimization	Kumar V., Senthil Kumar A. M., Praveen J. I. R., et al., (2024) Q1	Ensemble Model dengan Exponential Moving Average (EMA) dan Enhanced Weighted Gradient Optimization (EWGO), menggunakan model VGG-16 dan NASNet	Akurasi 98.7%, precision 97.9%, recall 98.6%, F1-score 98.7%	Keunggulan: Akurasi sangat tinggi (98.7%), penggunaan ensemble model untuk meningkatkan akurasi klasifikasi, sangat efisien Kelemahan: Penggunaan ensemble model yang lebih kompleks, membutuhkan lebih banyak sumber daya komputasi variasi kondisi di dunia nyata.	Tidak secara spesifik menganalisis pengaruh jenis augmentasi terhadap robustness model terhadap perubahan cahaya, sudut pandang, dan kondisi lapangan

4	Klasifikasi Penyakit Tanaman Tomat Menggunakan Metode Convolutional Neural Network (CNN) VGG16	Marchel Adias Pradana et al. (2023) Sinta 5	Convolutional Neural Network (CNN) dengan architecture VGG16	Akurasi pengujian 91%, akurasi validasi 92%	Kelebihan: Arsitektur CNN VGG16 yang dalam dan efisien untuk ekstraksi fitur Kelemahan: Performa lebih rendah pada dataset yang lebih beragam atau di lapangan	Penelitian ini hanya menggunakan augmented data standar (normalisasi dan beberapa teknik dasar seperti rotasi, flipping).
5	Klasifikasi Penyakit Daun Tomat Dengan Metode Convolutional Neural Network Menggunakan Arsitektur Inception-V3	Andi Nurdin et al.(2024) Sinta 5	Convolutional Neural Network (CNN) dengan Inception V3, optimasi dengan 3 jenis algoritma optimizer (Adam, SGD, RMSProp)	Akurasi terbaik menggunakan Adam optimizer: 93.8%, Precision: 93.9%, Recall: 93.8%, F1-Score: 93.7%	Kelebihan: Penggunaan Inception V3 dan Transfer Learning mempercepat pelatihan Kekurangan: Penelitian ini lebih berfokus pada pemilihan model dan optimizer terbaik daripada eksplorasi lebih mendalam tentang tuning hyperparameter	Kurang mendalam tentang tuning hyperparameter
6	Evaluasi Augmentasi Data	Ibnu Topan Adib	YOLOv8 dengan data	Peningkatan nilai mAP dari 40.3%	Kelebihan:	Menggunakan YOLO11 dengan

	pada Deteksi Penyakit Daun Tebu dengan YOLOv8	Amrulloh, Betha Nurina Sari, Tesa Nur Padilah (2024) Sinta 4	augmentasi (horizontal flipping, jittering scale, random brightness)	menjadi 50.5% setelah penggunaan augmentasi data	Data augmentasi memperkaya dataset dan membantu model mengatasi overfitting dan meningkatkan generalisasi Kekurangan: Augmentasi data terbatas pada teknik flipping, scale jittering, dan brightness saja, mungkin perlu eksplorasi teknik lain	augmentasi data lanjutan yang lebih kompleks seperti Advanced Mixing, Photometric Augmentation, dan Spatial Transformations, yang memberikan robustness lebih tinggi terhadap variasi pencahayaan, sudut pandang, dan kondisi dunia nyata.
7	Analisis Efek Augmentasi Dataset dan Fine Tune pada Algoritma Pre-Trained Convolutional Neural Network (CNN)	Theopilus Bayu Sasongko, Haryoko, Agit Amrullah (2023) Sinta 2	CNN dengan Transfer Learning menggunakan MobileNet, EfficientNet, dan NASNetMobile untuk dataset kecil, Fine-tuning dan augmentasi dataset digunakan	Peningkatan akurasi antara 0.03% hingga 0.1% setelah penerapan random erase dan zoom range augmentasi.	Kelebihan: Penggunaan Transfer Learning dengan model MobileNet, EfficientNet, dan NASNetMobile yang memiliki parameter sedikit dan lebih efisien untuk dataset kecil Kekurangan: Dataset yang digunakan sangat kecil, yang mungkin	Menambahkan augmentasi yang lebih kompleks yang lebih tahan terhadap variasi dunia nyata, menggunakan YOLO11 dan menambahkan analisis terhadap pengaruh augmentasi data terhadap kinerja model dalam kondisi yang lebih dinamis dan bervariasi.

			untuk meningkatkan performa model		membatasi kemampuan model untuk belajar fitur yang lebih kompleks	
8	Synthetic Data Augmentation of Tomato Plant Leaf using Meta Intelligent Generative Adversarial Network: Milgan	Sri Silpa Padmanabhuni, Pradeepini Gera. (2022) Q4	Generative Adversarial Network (GAN) dengan Meta-Learning untuk sintesis data daun tomat	Meta-Learning GAN menghasilkan gambar sintetis dengan akurasi 99.29% untuk identifikasi penyakit, melampaui model-model tradisional seperti CNN atau GAN standar	Kelebihan: Meta-Learning GAN menghasilkan data sintetis dengan lebih sedikit noise dibandingkan GAN tradisional. Menghasilkan gambar dengan kualitas lebih tinggi menggunakan kernel filtering dan teknik transformasi geometrik Kekurangan: Proses training dan pembuatan gambar sintetis masih membutuhkan waktu dan sumber daya yang besar	GAN dengan Meta-Learning untuk menghasilkan data sintetis dan meningkatkan kualitas gambar dengan lebih sedikit noise, tetapi menggunakan model lebih kompleks dan waktu pelatihan lebih lama. Penelitian yang diajukan menggunakan YOLO11 dengan augmentasi data lanjutan yang lebih robust terhadap variasi kondisi lapangan seperti pencahayaan dan sudut pandang.

Source: Author (2007). Gunakan style Citation for Table AMIKOM

2.3 Landasan Teori

Landasan teori adalah kerangka konseptual yang menghimpun konsep, teori, dan temuan terdahulu untuk menjelaskan variabel serta hubungan di antaranya, sehingga menjadi pijakan dalam perumusan hipotesis, pemilihan metode, dan penafsiran hasil.

2.3.1 Augmentasi Data

Augmentasi data adalah teknik penting dalam machine learning, terutama dalam aplikasi yang menggunakan data gambar, seperti klasifikasi, deteksi objek, dan segmentasi [23]. Teknik ini melibatkan transformasi dataset asli untuk menghasilkan variasi data baru tanpa mengubah informasi penting yang terkandung di dalamnya. Augmentasi data bertujuan untuk meningkatkan performa model dengan memperkaya keragaman dataset, mengurangi overfitting, dan membantu model untuk generalisasi lebih baik terhadap data yang belum pernah dilihat sebelumnya [24].

Dataset besar dengan label yang akurat adalah salah satu tantangan utama. Pengumpulan data sering kali membutuhkan waktu, biaya, dan sumber daya yang signifikan. Selain itu, dataset kecil cenderung menyebabkan model mengalami overfitting, yaitu situasi di mana model terlalu terfokus pada pola spesifik dalam data pelatihan dan gagal bekerja dengan baik pada data uji atau data dunia nyata. Augmentasi data menjadi solusi penting untuk mengatasi masalah ini. Dengan menghasilkan variasi baru dari dataset yang ada, augmentasi data memungkinkan model untuk melihat lebih banyak skenario tanpa memerlukan data tambahan [24].

Augmentasi data bertujuan memperluas distribusi pelatihan tanpa pengambilan data baru, mengurangi varians model dan meningkatkan invarian terhadap perubahan lingkungan. Pada klasifikasi penyakit daun tomat, augmentasi membantu model membedakan pola lesi dari perubahan non-esensial seperti pencahayaan, orientasi, atau skala daun, sehingga akurasi dan generalisasi meningkat pada kondisi lapangan yang beragam. Terdapat berbagai teknik augmentasi data yang dapat diterapkan tergantung pada jenis data dan tujuan aplikasi.

Augmentasi data menawarkan sejumlah keuntungan signifikan dalam pengembangan model. Pertama, teknik ini membantu mengurangi overfitting dengan mencegah model menghafal pola spesifik dalam dataset pelatihan, karena variasi baru yang diperkenalkan membuat model lebih tangguh terhadap data dunia nyata. Selain itu, augmentasi data meningkatkan generalisasi model, memungkinkan model untuk mengenali pola yang lebih umum, sehingga performanya pada data uji atau data yang tidak terlihat menjadi lebih baik [25].

Pemanfaatan augmentasi data, penggunaan dataset kecil menjadi lebih efisien, di mana dataset kecil yang telah di-augmentasi dapat mencapai performa yang sebanding dengan dataset yang lebih besar. Terakhir, banyak studi menunjukkan bahwa augmentasi data secara konsisten meningkatkan metrik performa model, seperti akurasi, presisi, dan recall, menjadikannya strategi yang sangat berharga dalam pengembangan model [26]. Berikut adalah beberapa teknik augmentasi data yang digunakan untuk penelitian ini:

2.3.1.1 Advanced Mixing Augmentation

Advanced mixing augmentation adalah sekumpulan teknik augmentasi data tingkat lanjut yang bekerja dengan menggabungkan dua atau lebih gambar serta mengombinasikan labelnya, sehingga menghasilkan sampel pelatihan yang lebih kompleks dan lebih bervariasi dibandingkan augmentasi konvensional seperti rotasi, flipping, atau perubahan pencahayaan [27]. Teknik ini didesain untuk meningkatkan kemampuan generalisasi model deep learning dengan memberikan contoh-contoh yang lebih menantang dan tidak mudah ditebak oleh model [27].

Secara umum, advanced mixing augmentation bertujuan untuk membuat model tidak hanya mempelajari pola dari satu gambar tunggal, tetapi juga memahami bagaimana objek atau bagian dari objek muncul dalam kondisi yang sangat bervariasi dan terkadang tidak terduga. Hal ini menjadikan model lebih kuat terhadap overfitting dan meningkatkan ketahanan terhadap noise, variasi pose, kondisi lingkungan, serta deformasi bentuk [28].

Salah satu metode yang paling dikenal adalah MixUp, yaitu teknik yang mencampurkan dua gambar secara linear. Pada metode ini, gambar baru dibentuk dengan menggabungkan dua gambar dengan proporsi tertentu, biasanya ditentukan oleh sebuah nilai α yang diambil dari distribusi Beta [29]. Selain gambar, label kedua gambar tersebut juga dicampurkan, sehingga menghasilkan label campuran. Teknik ini mendorong model agar tidak terlalu yakin terhadap satu kelas tertentu, sehingga mengurangi terjadinya overfitting dan membuat decision boundary lebih halus [28].

Metode lain adalah CutMix, yang dianggap lebih efektif karena mempertahankan struktur spasial objek [28]. Dalam CutMix, sebuah patch atau potongan gambar dari gambar kedua ditempelkan ke dalam gambar pertama. Label untuk gambar hasil potongan ini dihitung berdasarkan proporsi luas patch yang ditempel. Teknik ini tidak hanya menambah variasi visual, tetapi juga mengajarkan model untuk melakukan fokus lokal pada bagian-bagian objek yang berbeda, sehingga performa model meningkat terutama pada tugas-tugas yang membutuhkan pemahaman visual detail [28].

Selain itu, terdapat metode Mosaic, yang sangat populer dalam model-model YOLO modern. Mosaic menggabungkan empat gambar sekaligus ke dalam satu kanvas berukuran besar yang dibagi menjadi empat kuadran. Teknik ini sangat efektif untuk meningkatkan kemampuan klasifikasi objek kecil karena objek-objek tersebut dapat muncul dalam berbagai ukuran dan posisi. Mosaic juga secara signifikan menambah variasi posisi, jarak, dan tata letak objek tanpa harus memperbesar dataset secara manual [30].

Metode lainnya, yaitu copy-paste augmentation, menyalin objek tertentu dari sebuah gambar lalu menempelkannya ke gambar lain. Teknik ini sangat berguna terutama untuk dataset dengan distribusi objek yang tidak merata. Dengan menambahkan objek secara artifisial, model dapat dilatih untuk mengenali objek dalam berbagai konteks dan latar belakang [30].

Secara keseluruhan, advanced mixing augmentation memberikan peningkatan signifikan terhadap kualitas data pelatihan dengan menciptakan contoh-contoh baru yang tidak hanya bervariasi secara tekstur dan warna, tetapi

juga bervariasi secara struktural dan kontekstual. Kombinasi berbagai pendekatan ini terbukti mampu meningkatkan performa model, mengurangi overfitting, serta memperbaiki robustness terhadap kondisi dunia nyata [29].

2.3.1.2 Photometric Augmentation

Photometric augmentation adalah teknik augmentasi data yang berfokus pada perubahan sifat-sifat pencahayaan dan karakteristik warna pada gambar tanpa mengubah struktur atau bentuk objek di dalamnya. Augmentasi ini hanya memodifikasi aspek-aspek visual seperti intensitas cahaya, saturasi, kontras, atau distribusi warna, sehingga gambar tetap mempertahankan bentuk objek aslinya tetapi terlihat berbeda dari segi kondisi pencahayaan [31]. Pendekatan ini digunakan untuk membuat model deep learning lebih tahan terhadap variasi kondisi pencahayaan di dunia nyata [31].

Secara umum, photometric augmentation bertujuan untuk melatih model agar tidak terlalu sensitif terhadap perubahan cahaya, bayangan, atau kualitas kamera. Dalam konteks computer vision, perubahan pencahayaan merupakan salah satu dari sumber variasi terbesar yang dapat menyebabkan model gagal mengenali objek, terutama jika dataset asli tidak mencakup kondisi pencahayaan yang beragam. Dengan menambahkan variasi warna dan pencahayaan secara artifisial, model akan lebih banyak belajar mengenali objek meskipun tampilan visualnya sangat berubah [32].

Beberapa teknik yang termasuk dalam photometric augmentation antara lain brightness adjustment, yaitu penyesuaian tingkat kecerahan gambar dengan cara menambah atau mengurangi intensitas cahaya. Teknik ini menirukan kondisi

pengambilan gambar pada lingkungan yang terang atau gelap [33]. Selain itu, terdapat contrast adjustment, yang mengubah tingkat perbedaan antara area terang dan gelap pada gambar. Kontras yang lebih tinggi memberikan tampilan gambar yang lebih tajam, sedangkan kontras yang lebih rendah menghasilkan gambar yang lebih datar.

Teknik lainnya adalah saturation adjustment, yang mengubah tingkat kejenuhan warna pada gambar. Ketika saturasi ditingkatkan, warna terlihat lebih cerah dan kuat, sedangkan penurunan saturasi membuat gambar terlihat pudar atau bahkan mendekati grayscale [34]. Hue adjustment juga termasuk dalam kategori ini, yaitu menggeser rona warna, sehingga gambar dapat tampak memiliki palet warna yang berbeda dari aslinya.

Selain manipulasi warna dasar, photometric augmentation juga mencakup teknik seperti color jitter, yang merupakan kombinasi acak dari perubahan brightness, contrast, saturation, dan hue secara bersamaan [34]. Teknik ini sangat efektif untuk menciptakan variasi visual secara acak dan membuat model lebih robust terhadap perbedaan warna kamera, kualitas sensor, kondisi cuaca, atau setelan pencahayaan [34].

Beberapa implementasi photometric augmentation yang lebih ekstrem termasuk adding noise seperti Gaussian noise, salt-and-pepper noise, atau speckle noise. Penambahan noise mensimulasikan kondisi kamera yang buruk, gangguan sensor, atau kualitas gambar yang rendah. Teknik ini mengajarkan model untuk tetap mengenali objek meskipun gambar mengalami kerusakan visual [31].

Secara keseluruhan, photometric augmentation memberikan peningkatan signifikan terhadap kemampuan generalisasi model deep learning dengan menciptakan variasi warna dan pencahayaan yang luas. Teknik ini memastikan bahwa model dapat bekerja dengan baik meskipun gambar yang dihadapi memiliki kondisi pencahayaan berbeda dari dataset asli. Dengan demikian, photometric augmentation menjadi salah satu tahapan penting dalam pelatihan model yang akan digunakan di lingkungan nyata yang penuh variasi [32].

2.3.1.3 Spatial Transformations

Spatial transformations adalah kelompok teknik augmentasi data yang memodifikasi struktur spasial atau susunan geometris dari sebuah gambar tanpa mengubah informasi warna atau pencahayaan [35]. Teknik ini bekerja dengan mengubah posisi, bentuk, skala, atau orientasi objek dalam gambar, sehingga menghasilkan variasi data yang membantu model deep learning belajar mengenali objek dari berbagai sudut pandang, jarak, dan kondisi perspektif [34].

Tujuan utama dari spatial transformations adalah membuat model lebih robust terhadap perubahan posisi dan orientasi objek di dunia nyata. Dalam banyak kasus, objek dapat muncul dalam berbagai bentuk tampilan: miring, terbalik, lebih dekat, lebih jauh, atau terpotong sebagian. Jika dataset pelatihan tidak memiliki keragaman seperti ini, model sering kesulitan mengenali objek dalam kondisi yang berbeda dari gambar latihannya. Oleh karena itu, spatial transformations menjadi sangat penting terutama dalam tugas computer vision seperti deteksi objek, klasifikasi, segmentasi, dan tracking [35].

Salah satu teknik yang paling umum adalah rotation, yaitu memutar gambar atau objek dalam sudut tertentu. Dengan memutar gambar ke kanan, kiri, atau bahkan secara acak, model dilatih untuk tetap mengenali objek meskipun orientasinya berubah. Teknik ini sangat bermanfaat untuk objek yang bentuknya tidak simetris atau ketika posisi kamera tidak stabil.

Selain rotasi, terdapat teknik flipping, yang mencerminkan gambar secara horizontal atau vertikal. Flipping horizontal adalah yang paling sering digunakan karena dapat mensimulasikan kondisi di mana objek terlihat dari arah berlawanan. Sementara itu, flipping vertikal digunakan lebih selektif, terutama pada dataset yang memang memungkinkan objek terlihat terbalik, seperti dalam fotografi drone atau citra medis tertentu [36].

Teknik penting lainnya adalah scaling, yaitu memperbesar atau memperkecil ukuran objek dalam gambar. Scaling memungkinkan model memahami objek pada berbagai jarak dan ukuran. Ketika objek terlihat lebih kecil atau lebih besar, model tetap dapat mengenalinya karena sudah dilatih pada variasi skala [36].

Spatial transformations juga mencakup translation, yaitu menggeser posisi objek ke kanan, kiri, atas, atau bawah. Translation mengajarkan model untuk tidak hanya mengandalkan posisi absolut saat mengenali objek. Ini sangat penting dalam klasifikasi karena objek di dunia nyata jarang berada di posisi yang sama setiap kali [35].

Selanjutnya, terdapat teknik cropping, yaitu memotong sebagian area gambar. Dengan memotong area tertentu, model dilatih untuk tetap mengenali

objek meskipun sebagian bagian objek tertutup atau keluar dari bidang pandang kamera. Teknik ini sangat berguna dalam situasi nyata di mana objek sering kali terhalang atau hanya terlihat sebagian [37].

Beberapa implementasi spasial transformations yang lebih kompleks mencakup affine transformations dan perspective transformations. Affine transformations melibatkan kombinasi scaling, translation, rotation, dan shear secara bersamaan, sehingga menghasilkan perubahan bentuk yang lebih beragam. Perspective transformations meniru perubahan sudut pandang kamera sehingga objek tampak seperti dilihat dari posisi yang lebih condong atau dari sudut tertentu [28].

Secara keseluruhan, spatial transformations memberikan kontribusi penting dalam memperluas variasi dataset dengan menciptakan perubahan posisi, orientasi, ukuran, dan bentuk objek secara sintetis. Teknik ini membantu model deep learning beradaptasi dengan kondisi visual dunia nyata yang sangat dinamis dan tidak terstruktur. Dengan augmentasi spasial yang tepat, model dapat lebih tangguh, fleksibel, dan mampu mengenali objek dalam berbagai kondisi geometris yang berbeda dari data aslinya [38].

2.3.2 Tomat

Tomat (*Solanum lycopersicum* L.) merupakan salah satu komoditas hortikultura penting yang termasuk dalam famili Solanaceae, berasal dari Amerika Selatan dan telah dibudidayakan secara luas di berbagai belahan dunia sebagai tanaman sayuran bernilai ekonomi tinggi [39]. Tanaman ini merupakan tumbuhan perennial yang dapat tumbuh hingga ketinggian 1-3 meter, memiliki batang berbulu

halus, daun majemuk, dan menghasilkan buah berwarna merah atau kuning yang kaya akan vitamin C, likopen, serta seperti likopen yang bermanfaat bagi kesehatan terutama dalam pencegahan penyakit kardiovaskular dan kanker [1].

Secara agronomi, tomat adalah tanaman semusim yang tumbuh optimal di iklim hangat dengan kebutuhan sinar matahari yang cukup. Pertumbuhannya dipengaruhi oleh faktor-faktor lingkungan seperti suhu, kelembapan, dan tanah yang kaya akan unsur hara. Di dalam budidayanya, tomat dapat ditanam baik di lahan terbuka maupun dalam rumah kaca dengan teknik hidroponik. Penggunaan varietas unggul dan pemeliharaan yang baik dapat meningkatkan hasil panen dan kualitas buah tomat [2].

2.3.2.1 Penyakit Daun Tomat

Tanaman tomat rentan terhadap berbagai penyakit yang dapat mengurangi hasil panen dan kualitas buah. Daun tomat dapat terinfeksi oleh berbagai jenis patogen, termasuk bakteri, jamur, dan virus. Penyakit pada daun tomat, seperti tomato mosaic virus, target spot, bacterial spot, tomato yellow leaf curl virus, late blight, leaf mold, early blight, spider mites two-spotted spider mite, tomato healthy, dan septoria leaf spot dapat menyebar dengan cepat dan menyebabkan kerugian besar bagi petani.

Penyakit yang sering ditemukan pada daun tomat meliputi beberapa jenis dengan penyebab, gejala, dampak, dan cara penularan yang berbeda. Bacterial spot, disebabkan oleh bakteri *Xanthomonas campestris* pv. *vesicatoria*, menyebabkan bercak kecil berair dan gelap, yang dapat mengakibatkan nekrosis dan penurunan kualitas buah [1]. Gambar 2. 1 merupakan contoh penyakit bacterial spot.



Gambar 2. 1 Penyakit Bacterial Spot

Early blight, akibat jamur *Alternaria solani*, ditandai dengan bercak coklat kehitaman pada daun tua, yang dapat menyebabkan defoliasi [7]. Gambar 2. 2 merupakan contoh penyakit early blight.



Gambar 2. 2 Early Blight

Late blight, disebabkan oleh jamur *Phytophthora infestans*, menunjukkan lesi coklat gelap hingga hitam dan dapat menyebar cepat dalam kondisi lembab, menyebabkan kehilangan hasil yang signifikan [7]. Gambar 2. 3 merupakan contoh penyakit late blight.



Gambar 2. 3 Penyakit Late Blight pada Daun Tomat

Leaf Mold, penyakit ini disebabkan oleh jamur *Cladosporium fulvum* dan menyebabkan bercak abu-abu atau hijau pada daun tomat. Penyakit ini mengurangi

efisiensi fotosintesis tanaman dan sering terlihat pada kondisi lembap. Meskipun tidak membunuh tanaman dengan cepat, ia dapat mengurangi hasil panen secara signifikan [7]. Gambar 2. 4 merupakan contoh penyakit leaf mold.



Gambar 2. 4 Leaf Mold

Two-Spotted Spider Mite merupakan penyakit ini disebabkan oleh hama tungau, terutama *Tetranychus urticae*, yang menghisap cairan dari daun tomat. Gejalanya meliputi daun yang menguning dan mengering, serta penurunan pertumbuhan tanaman. Tungau ini juga menyebabkan jaring laba-laba kecil yang menutupi tanaman. Hama ini sering kali menjadi masalah di kondisi panas dan kering [7]. Gambar 2, 5 merupakan contoh penyakit two-spotted spider mite.



Gambar 2. 5 Two-Spotted Spider Mite

Tomato Yellow Leaf Curl Virus, yaitu penyakit yang menyebabkan daun tomat menjadi kunir dan melengkung ke bawah. Gejalanya termasuk pertumbuhan yang terhambat, daun menguning, dan daun muda menggulung. Penyakit ini terutama disebarkan oleh vektor kutu daun (whiteflies) dan dapat menyebabkan

kerugian besar pada tanaman [37]. Gambar 2. 6 merupakan contoh penyakit tomat yellow leaf curl virus.



Gambar 2. 6 Tomato Yellow Leaf Curl Virus

Tomato Mosaic Virus, penyakit ini disebabkan oleh virus yang menyerang tanaman tomat. Gejalanya berupa bercak-bercak hijau terang atau kuning yang tidak teratur di daun, yang menyebabkan deformasi daun, pertumbuhan terhambat, dan penurunan hasil. Virus ini ditularkan melalui kontak langsung atau oleh serangga pemakan daun. Gambar 2. 7 merupakan contoh penyakit tomat mosaic virus.



Gambar 2. 7 Tomato Mosaic Virus

Target Spot, penyakit ini disebabkan oleh jamur *Corynespora cassiicola*. Gejalanya berupa bercak coklat atau hitam dengan tepi yang jelas dan area pusat yang terang. Penyakit ini dapat menyebabkan defoliasi (jatuhnya daun) yang

mengurangi kemampuan fotosintesis tanaman [40]. Gambar 2. 8 merupakan contoh penyakit target spot.



Gambar 2. 8 Target Spot

Septoria Leaf Spot adalah penyakit yang disebabkan oleh jamur *Septoria lycopersici*, yang menyerang daun tomat. Penyakit ini ditandai dengan munculnya bercak-bercak kecil berwarna cokelat atau hitam pada daun, dengan tepi yang seringkali dikelilingi oleh area kuning. Seiring perkembangan penyakit, bercak-bercak ini dapat membesar dan menyebabkan daun menjadi kering dan gugur [41]. Gambar 2. 9 merupakan contoh penyakit septoria leaf spot.



Gambar 2. 9 Septoria Leaf Spot

Daun tomat sehat merujuk pada kondisi tanaman tomat yang bebas dari penyakit atau gangguan. Tanaman yang sehat memiliki daun hijau, pertumbuhan yang baik, dan hasil yang optimal. Gambar 2. 10 merupakan contoh daun tomat sehat.



Gambar 2. 10 Daun Tomat Sehat

Tanpa pengklasifikasian penyakit yang lebih dini, penyakit-penyakit ini dapat menyebar ke seluruh tanaman dan bahkan ke ladang lain, menyebabkan kerugian ekonomi yang signifikan [3].

Oleh karena itu, pengklasifikasian penyakit dan pengelolaan yang tepat menjadi kunci untuk menjaga kesehatan produktivitas tanaman tomat dan pencegahan kerugian lebih lanjut [42].

2.3.3 Computer Vision

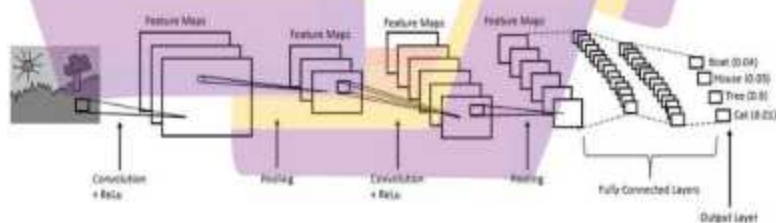
Computer vision adalah cabang ilmu dalam kecerdasan buatan (AI) yang memungkinkan komputer untuk memahami, menganalisis, dan menafsirkan data visual dari dunia sekitar [43]. Teknologi ini memungkinkan mesin untuk melihat melalui kamera, video, atau sensor lain, kemudian memproses dan mengambil keputusan berdasarkan informasi yang diterima. Proses ini melibatkan berbagai teknik, termasuk klasifikasi, segmentasi gambar, deteksi objek, dan pelacakan gerakan. Computer vision banyak digunakan dalam berbagai aplikasi seperti pengenalan wajah, deteksi objek, kendaraan otonom, diagnosis medis berbasis gambar, dan pemantauan keamanan. Dengan memanfaatkan algoritma machine learning dan deep learning, computer vision mampu mengolah data visual dalam jumlah besar dengan akurasi tinggi. Teknologi ini terus berkembang, membuka

peluang baru dalam berbagai industri dengan mengotomatisasi tugas-tugas yang sebelumnya hanya bisa dilakukan oleh manusia [44].

2.3.4 CNN (Convolutional Neural Networks)

Convolutional Neural Networks (CNN) adalah jenis arsitektur jaringan saraf yang dirancang khusus untuk memproses data berbentuk grid, seperti gambar [45]. CNN menggunakan lapisan konvolusi untuk mengekstrak fitur dari input dengan menerapkan filter yang bergerak di atas data, diikuti oleh lapisan pooling yang mengurangi dimensi data sambil mempertahankan informasi penting. Proses ini memungkinkan CNN untuk mengenali pola dan objek dalam gambar secara efisien. CNN sering digunakan dalam berbagai aplikasi, termasuk pengenalan wajah, klasifikasi gambar, dan deteksi objek, berkat kemampuannya dalam belajar representasi hierarkis dari data visual tanpa memerlukan ekstraksi fitur manual [46].

Arsitektur Convolutional Neural Networks (CNN) terdiri dari beberapa lapisan utama yang bekerja secara sinergis untuk memproses data. Gambar 2. 11 adalah contoh arsitektur CNN.



Gambar 2. 11 Arsitektur CNN

Pertama, Convolutional Layer melakukan operasi konvolusi untuk mengekstraksi fitur penting dari data, seperti tepi, tekstur, atau pola tertentu. Selanjutnya, Pooling Layer berfungsi untuk mengurangi dimensi data sambil

mempertahankan fitur penting, sehingga mengurangi kompleksitas komputasi dan meningkatkan efisiensi. Setelah itu, Fully Connected Layer menghubungkan fitur yang diekstraksi dari lapisan sebelumnya untuk membuat keputusan akhir, seperti prediksi label. Terakhir, Activation Function, seperti ReLU, digunakan untuk menangkap hubungan kompleks dalam data dengan menambahkan elemen non-linearitas pada model [47].

2.3.5 Klasifikasi dengan YOLO11

Meskipun YOLO (You Only Look Once) dirancang untuk multi-object detection, penggunaan YOLO untuk klasifikasi penyakit daun tomat dapat tetap dianggap sebagai pilihan terbaik karena beberapa alasan penting. Pertama, YOLO sangat cepat dalam melakukan deteksi dan klasifikasi objek dalam gambar, bahkan ketika hanya ada satu objek per gambar [8]. Kecepatan ini sangat berguna dalam aplikasi dunia nyata, terutama ketika jumlah gambar yang harus diproses sangat besar. Kecepatan YOLO memungkinkan aplikasi real-time yang efisien, yang penting dalam pengawasan pertanian, di mana deteksi penyakit secara cepat dapat membantu dalam pengambilan keputusan yang lebih tepat [8].

Kedua, meskipun YOLO dirancang untuk mendeteksi banyak objek, model ini juga cukup efektif dalam mengidentifikasi objek tunggal [9]. Dalam kasus seperti klasifikasi penyakit daun tomat, meskipun dataset hanya berisi satu objek per gambar, kemampuan YOLO untuk melakukan deteksi dan klasifikasi secara simultan tetap dapat memberikan hasil yang sangat baik [9]. YOLO mampu mengoptimalkan penggunaan informasi spasial dalam gambar dengan cara yang

sangat efisien, memastikan bahwa penyakit yang muncul pada daun dapat dideteksi dengan cepat dan akurat.

Ketiga, YOLO memiliki kemampuan untuk menangani variasi dalam posisi dan orientasi objek secara lebih baik [10]. Meskipun dalam tugas klasifikasi sederhana mungkin tidak dibutuhkan deteksi objek multi, kemampuan YOLO dalam menangani perubahan sudut pandang, rotasi, dan variasi lainnya memberi keuntungan tambahan dalam situasi dunia nyata di mana kondisi gambar dapat berubah-ubah. Ini membuat YOLO lebih fleksibel dalam menghadapi dataset yang lebih dinamis atau gambar yang mungkin terdistorsi karena faktor pencahayaan atau sudut pengambilan gambar [10].

Selain itu, YOLO memiliki arsitektur yang dapat ditingkatkan dan diterapkan dalam berbagai domain, memberikan potensi untuk pengembangan lebih lanjut jika diperlukan. Menggunakan YOLO untuk tugas klasifikasi juga memudahkan pengembang untuk memanfaatkan model yang sudah terlatih dengan baik pada dataset yang lebih besar dan lebih kompleks [36]. Hal ini memungkinkan transisi yang lebih mudah ke aplikasi yang lebih rumit dengan berbagai jenis objek.

Secara keseluruhan, meskipun YOLO dirancang untuk deteksi objek multi, kemampuannya dalam hal kecepatan, fleksibilitas, dan efisiensi membuatnya menjadi pilihan terbaik untuk klasifikasi penyakit daun tomat, terutama jika diinginkan aplikasi yang lebih cepat dan mampu menangani variasi dalam kondisi gambar dunia nyata.

Dalam konteks penggunaan model YOLO11 (You Only Look Once, versi 11), perbedaan antara klasifikasi dan deteksi objek sangat penting untuk dipahami.

Deteksi objek merujuk pada kemampuan model untuk menemukan dan menandai lokasi objek dalam gambar, serta mengklasifikasikan jenis objek tersebut dalam satu langkah [48]. Dalam hal ini, YOLOv1 dapat mendeteksi berbagai objek dalam satu gambar dan memberikan kotak pembatas (bounding box) pada setiap objek yang terdeteksi, sekaligus memberi label kategori objek tersebut [49]. Sebaliknya, klasifikasi objek berfokus pada identifikasi dan penentuan kategori objek, tanpa mengutamakan penandaan lokasi objek. Dalam klasifikasi, YOLO11 lebih diarahkan untuk menganalisis dan mengklasifikasikan objek yang ada dalam gambar secara keseluruhan, dengan hasil yang lebih terperinci pada jenis objek atau penyakit yang muncul pada daun tomat [48]. Perbedaan ini membuat YOLO11 sangat fleksibel, mampu tidak hanya mendeteksi penyakit pada tanaman tetapi juga mengklasifikasikannya dengan akurasi tinggi, memberikan gambaran lebih jelas bagi petani dalam melakukan penanganan yang tepat [50].

YOLO11 bekerja dengan membagi gambar input menjadi beberapa grid, di mana setiap sel grid bertugas untuk memprediksi keberadaan objek di area tersebut. YOLO11 akan memprediksi bounding box (Bbox) dan probabilitas kelas (Cls) melalui satu jaringan [51]. Selain itu, ia unggul dalam akurasi tinggi meskipun menggunakan ukuran model yang lebih kecil dan dapat dilatih pada satu GPU [52].

Arsitektur YOLO11 terdiri dari tiga komponen utama, yaitu backbone, neck, dan head. Gambar 2. 12 menunjukkan arsitektur YOLO11.

Secara keseluruhan, arsitektur ini dirancang untuk meningkatkan akurasi dan efisiensi dalam mengklasifikasikan objek dalam gambar, dengan memperhatikan berbagai skala dan fitur yang diperoleh dari backbone dan neck [54].

2.3.5.1 Paramater tuning YOLO

Paramater tuning merupakan proses penting dalam optimasi model deep learning, termasuk YOLO (You Only Look Once) yang digunakan dalam klasifikasi objek. Hyperparameter adalah parameter yang tidak dipelajari selama pelatihan, tetapi mempengaruhi bagaimana model belajar dan kinerjanya. Dalam konteks YOLO, yang merupakan model klasifikasi objek berbasis Convolutional Neural Network (CNN), terdapat beberapa hyperparameter kunci yang perlu disesuaikan untuk meningkatkan performa model [55]. Di antaranya adalah learning rate, ukuran batch, jumlah epoch, serta probabilitas dan kekuatan augmentasi data.

Learning Rate adalah salah satu hyperparameter yang paling kritikal dalam pelatihan YOLO. Learning rate mengontrol seberapa besar langkah yang diambil oleh algoritma selama proses optimasi untuk memperbarui bobot model pada setiap iterasi. Jika nilai learning rate terlalu tinggi, model bisa melewati global minima atau local minima, yang menyebabkan pelatihan tidak konvergen atau menghasilkan model yang buruk [56]. Sebaliknya, jika terlalu rendah, pelatihan akan sangat lambat dan memerlukan lebih banyak iterasi untuk mencapai hasil yang optimal. Oleh karena itu, memilih learning rate yang tepat sangat penting untuk memastikan konvergensi yang efisien dan hasil model yang baik [57].

Ukuran batch adalah parameter lain yang sangat penting. Ukuran batch mengacu pada jumlah sampel yang diproses sebelum model melakukan pembaruan bobot. Dalam YOLO, ukuran batch yang lebih besar membantu stabilitas dalam proses gradient descent, karena lebih banyak data yang diproses dalam satu langkah optimasi. Namun, ukuran batch yang lebih besar membutuhkan lebih banyak memori komputasi, yang bisa menjadi kendala pada perangkat keras dengan kapasitas terbatas [58]. Di sisi lain, ukuran batch yang lebih kecil cenderung mempercepat pelatihan, namun dapat menyebabkan fluktuasi yang lebih besar dalam pembaruan bobot dan kesulitan dalam menemukan global minima [57].

Selanjutnya, jumlah epoch juga perlu dipertimbangkan. Epoch mengacu pada jumlah siklus pelatihan yang dilakukan model dengan seluruh dataset. Terlalu sedikit epoch mungkin menyebabkan underfitting, di mana model tidak cukup terlatih untuk menangkap pola dalam data [56]. Sebaliknya, terlalu banyak epoch bisa menyebabkan overfitting, di mana model terlalu menghafal data pelatihan dan kehilangan kemampuannya untuk generalisasi pada data baru [55]. Oleh karena itu, penyesuaian jumlah epoch perlu dilakukan secara hati-hati untuk menemukan keseimbangan yang tepat antara pelatihan yang cukup dan menghindari overfitting [59].

Selain itu, augmentasi data merupakan teknik yang sangat penting untuk meningkatkan robustness model YOLO terhadap variasi data yang tidak terduga [55]. Teknik augmentasi seperti rotasi, flipping, scaling, cropping, brightness, dan contrast adjustment dapat membantu model generalize dengan lebih baik terhadap perubahan sudut pandang, pencahayaan, dan posisi objek [57]. Dalam YOLO,

augmentasi data juga membantu meningkatkan ketahanan model terhadap overfitting, karena memberikan variasi pada data yang digunakan dalam pelatihan. Selain teknik augmentasi standar, augmentasi lanjutan seperti Advanced Mixing Augmentation, Photometric Augmentation, dan Spatial Transformations menawarkan fleksibilitas lebih besar dalam menghadapi variabilitas lingkungan, seperti perbedaan pencahayaan atau distorsi perspektif [60].

Regularization techniques, seperti dropout atau batch normalization, juga penting untuk tuning model agar tidak terjadi overfitting. Dropout secara acak menghilangkan beberapa unit pada setiap iterasi untuk mencegah model menjadi terlalu bergantung pada fitur tertentu. Batch normalization, di sisi lain, membantu dalam normalisasi data input pada setiap layer untuk mempercepat konvergensi dan mencegah masalah gradient vanishing atau exploding [60].

Penting untuk dicatat bahwa penyetelan hyperparameter dalam YOLO juga harus dilakukan dengan menggunakan cross-validation untuk memastikan bahwa model tidak hanya bekerja dengan baik pada dataset pelatihan, tetapi juga mampu melakukan generalization pada data yang belum terlihat sebelumnya. Teknik-teknik seperti grid search, random search, atau algoritma Bayesian optimization dapat digunakan untuk menemukan kombinasi terbaik dari hyperparameter, termasuk learning rate, ukuran batch, epoch, dan lainnya [55].

Dengan demikian, Paramater tuning yang efektif sangat krusial dalam optimasi model YOLO untuk mencapai akurasi yang maksimal serta menghindari overfitting atau underfitting. Dengan melakukan penyetelan yang tepat pada hyperparameter ini, model YOLO tidak hanya dapat memberikan hasil

yang akurat dalam klasifikasi penyakit pada daun tomat, tetapi juga dapat beradaptasi dengan baik terhadap beragam kondisi di dunia nyata, serta melakukan klasifikasi [57].

2.3.6 Evaluasi Kinerja Model

Dalam proses evaluasi, metode yang digunakan adalah confusion matrix, dimana dari matriks tersebut dapat dihitung nilai akurasi, presisi, recall, dan F1-score dari setiap model yang dihasilkan [61]. Confusion matrix digunakan untuk mengevaluasi kinerja dari model yang dihasilkan. Tabel 2. 2 adalah tabel confusion matriks.

Tabel 2. 2 Tabel Confusion Matrix

Actual	Predicted	
	True	False
True	TP	FN
False	FP	TN

2.3.6.1 Akurasi

Akurasi, adalah salah satu metrik yang digunakan untuk mengukur sejauh mana prediksi model sesuai dengan data yang sebenarnya [48].

$$\text{Akurasi} = \frac{TP+FP}{TP+FP+FN+TN} \quad (2.1)$$

2.3.6.2 Presisi

Presisi, adalah metrik yang mengukur seberapa banyak prediksi positif yang benar-benar positif [48]. Perhitungan presisi dapat dilakukan dengan menggunakan metode persamaan di bawah ini.

$$\text{Presisi} = \frac{TP}{TP+FP} \quad (2.2)$$

2.3.6.3 Recall

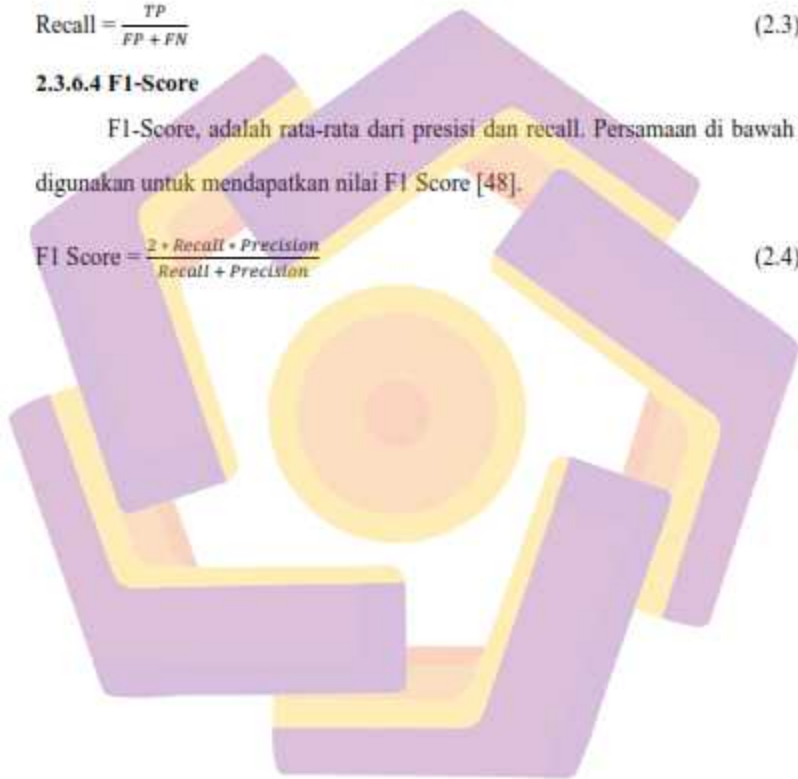
Recall, adalah metrik yang mengukur seberapa banyak objek positif berhasil ditemukan model [48]. Perhitungan Recall menggunakan persamaan di bawah ini.

$$\text{Recall} = \frac{TP}{FP + FN} \quad (2.3)$$

2.3.6.4 F1-Score

F1-Score, adalah rata-rata dari presisi dan recall. Persamaan di bawah ini digunakan untuk mendapatkan nilai F1 Score [48].

$$\text{F1 Score} = \frac{2 \cdot \text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} \quad (2.4)$$



BAB 3

METODE PENELITIAN

3.1 Jenis, Sifat, dan Pendekatan Penelitian

Penelitian ini merupakan penelitian eksperimen yang bertujuan untuk menguji pengaruh berbagai teknik augmentasi data lanjutan terhadap kinerja model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat. Sifat penelitian ini adalah kuantitatif karena menggunakan data numerik untuk dianalisis secara statistik, serta bersifat deskriptif karena memberikan gambaran dan penjelasan mengenai efek masing-masing teknik augmentasi terhadap performa model. Pendekatan yang digunakan adalah pendekatan eksperimental dan komputasional, di mana eksperimen dilakukan dengan menerapkan teknik-teknik augmentasi seperti Advanced Mixing, Photometric, dan Spatial Transformations pada dataset, kemudian mengevaluasi hasilnya menggunakan model deep learning YOLO11 sebagai bagian dari proses komputasional. Penelitian ini secara keseluruhan dirancang untuk menilai secara sistematis efektivitas augmentasi lanjutan dalam meningkatkan akurasi klasifikasi penyakit daun tomat.

3.2 Metode Pengumpulan Data

Metode pengumpulan data dalam penelitian ini dilakukan melalui studi literatur dan pengumpulan data sekunder. Studi literatur dilakukan dengan menelusuri jurnal ilmiah, buku, artikel, dan prosiding konferensi dari sumber yaitu Google Scholar, IEEE Xplore, dan SpringerLink untuk memperoleh informasi mengenai teknik augmentasi data serta penggunaan model YOLO dalam klasifikasi

penyakit tanaman. Selain itu, penelitian juga memanfaatkan data sekunder berupa dataset gambar daun tomat yang terinfeksi penyakit yang diperoleh dari repositori publik yaitu Kaggle. Dokumen teknis terkait model YOLO11 turut digunakan untuk memahami pengaturan dan penerapan model secara optimal. Seluruh data yang diperoleh ini menjadi dasar teoritis dan bahan pendukung dalam pelaksanaan eksperimen serta analisis hasil penelitian.

3.3 Metode Analisis Data

Dalam penelitian ini, analisis data dilakukan untuk mengevaluasi pengaruh teknik augmentasi data terhadap kinerja model YOLO11 dalam klasifikasi penyakit pada daun tomat. Proses analisis melibatkan beberapa tahap, mulai dari pra-pemrosesan data, pelatihan model, evaluasi model, hingga perbandingan hasil eksperimen. Berikut adalah penjelasan lengkap mengenai metode dan alat yang digunakan untuk analisis data.

Proses pertama adalah pra-pemrosesan data yang mencakup beberapa langkah penting. Dataset gambar daun tomat yang terinfeksi penyakit akan diproses untuk memastikan bahwa gambar-gambar tersebut dalam kondisi yang baik, dengan ukuran dan resolusi yang konsisten. Setelah itu, augmentasi data akan diterapkan menggunakan teknik Advanced Mixing Augmentation, Photometric Augmentation, dan Spatial Transformations. Teknik-teknik ini bertujuan untuk memperkenalkan variasi dalam dataset pelatihan sehingga model dapat mengenali penyakit dengan lebih baik.

Selain itu, langkah penting dalam pra-pemrosesan adalah pembagian data (split data). Dataset akan dibagi menjadi tiga bagian utama: data pelatihan (train),

data validasi (validation), dan data pengujian (test). Data pelatihan digunakan untuk melatih model, data validasi digunakan untuk mengevaluasi performa model selama proses pelatihan dan untuk mengatur hyperparameter model, sementara data pengujian digunakan untuk mengukur kinerja akhir model setelah pelatihan selesai. Pembagian data ini penting untuk menghindari overfitting dan memastikan model dapat diuji dengan data yang belum pernah dilihat sebelumnya.

Setelah data dipersiapkan dan dibagi, langkah selanjutnya adalah pelatihan model YOLO11. Model YOLO11 dilatih menggunakan data pelatihan yang telah di-augmentasi dan data pelatihan tanpa augmentasi untuk membandingkan hasilnya. Selama proses pelatihan, beberapa hyperparameter, seperti learning rate, batch size, dan jumlah epoch, akan disesuaikan untuk mengoptimalkan performa model.

Setelah pelatihan selesai, evaluasi model dilakukan dengan menguji model pada data pengujian. Metrik evaluasi yang digunakan meliputi akurasi, yang mengukur tingkat keberhasilan model dalam mengklasifikasikan penyakit pada daun tomat, serta precision, recall, dan F1-score, yang memberikan gambaran lebih rinci tentang kinerja model. Confusion matrix juga digunakan untuk menunjukkan kesalahan klasifikasi yang dilakukan oleh model.

Setelah evaluasi dilakukan, tahap selanjutnya adalah perbandingan hasil antara model yang dilatih menggunakan augmentasi data dan yang tidak. Perbandingan ini dilakukan dengan melihat hasil metrik evaluasi pada kedua grup data (dengan dan tanpa augmentasi) untuk mengetahui seberapa besar pengaruh

augmentasi terhadap akurasi dan kinerja model YOLO11 dalam klasifikasi penyakit pada daun tomat.

Untuk melaksanakan eksperimen ini, perangkat keras yang digunakan adalah laptop Acer Swift 3 SF314-41 yang dilengkapi dengan prosesor AMD Ryzen 5 3500U, RAM 12GB, dan penyimpanan 512GB. Laptop ini mendukung proses analisis meskipun tidak dilengkapi dengan GPU dedicated. VSCode digunakan sebagai editor kode utama, dan Python 3.13 dipilih sebagai bahasa pemrograman. Untuk pelatihan model deep learning, digunakan library Ultralytics. Sementara pustaka seperti NumPy, Matplotlib, dan OpenCV digunakan untuk pengolahan citra dan analisis data.

Dengan metode analisis ini, diharapkan penelitian dapat memberikan pemahaman yang lebih mendalam tentang pengaruh teknik augmentasi data terhadap kinerja model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat, serta meningkatkan robustness model di berbagai kondisi pencahayaan dan sudut pandang.

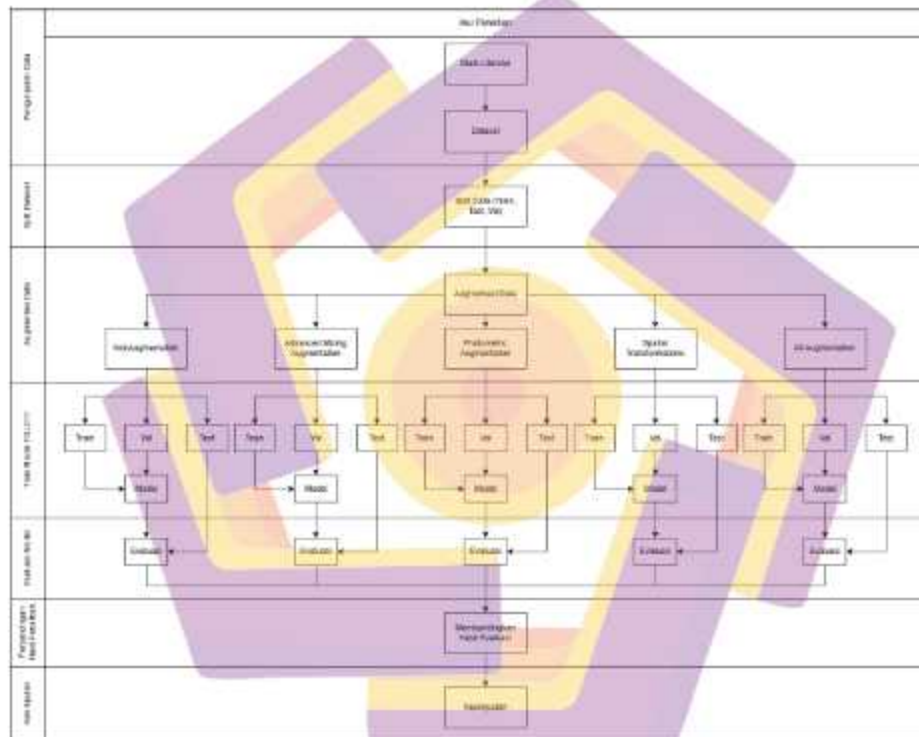
3.4 Alur Penelitian

Alur penelitian menggambarkan langkah-langkah yang diambil dari awal hingga akhir penelitian untuk menjawab masalah penelitian atau mencapai tujuan yang ditetapkan. Penelitian ini bertujuan untuk menganalisis pengaruh teknik augmentasi data terhadap kinerja model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat. Sebelum memulai eksperimen dan analisis, penting untuk memahami tahapan-tahapan yang diambil dalam penelitian ini, dimulai dari

pengumpulan dataset hingga perbandingan hasil evaluasi yang diperoleh dari berbagai teknik augmentasi data.

Alur penelitian ini diawali dengan penentuan alur penelitian secara konseptual melalui studi literatur dan penetapan dataset yang digunakan sebagai objek uji. Setelah dataset terkumpul, data dibagi menjadi tiga bagian, yaitu data latih (train), data validasi (validation), dan data uji (test) untuk menjaga objektivitas penilaian performa model dan meminimalkan risiko overfitting. Selanjutnya, pada tahap augmentasi data, dilakukan beberapa skenario pengolahan citra, yaitu tanpa augmentasi (non-augmentation), Advanced Mixing Augmentation, Photometric Augmentation, Spatial Transformations, serta kombinasi seluruh teknik (All Augmentation). Setiap skenario augmentasi menghasilkan versi dataset yang berbeda dan diproses secara terpisah.

Untuk masing-masing skenario augmentasi, model YOLO11 dilatih menggunakan data latih dan divalidasi dengan data validasi hingga diperoleh model yang stabil. Model yang sudah terlatih kemudian dievaluasi menggunakan data uji dan akan diperoleh nilai akurasi, presisi, recall, f1-score sehingga diperoleh nilai kinerja untuk tiap teknik augmentasi. Tahap berikutnya adalah membandingkan hasil evaluasi antar model guna mengidentifikasi teknik augmentasi yang paling efektif dalam meningkatkan performa klasifikasi objek. Pada tahap akhir, disusun kesimpulan yang merangkum temuan utama, menegaskan kontribusi penggunaan berbagai teknik augmentasi terhadap peningkatan kinerja YOLO11, serta memberikan rekomendasi untuk penelitian lanjutan.



Gambar 3.1 Diagram Alur Penelitian

BAB 4 HASIL PENELITIAN DAN PEMBAHASAN

4.1 Dataset

Dalam penelitian ini, dataset diperoleh dari Kaggle (<https://www.kaggle.com/datasets/kaustubhb999/tomatoleaf>). Tomato Leaf Diseases Detection. Semua data gambar yang diperoleh dalam penelitian ini disimpan dalam format JPG dengan resolusi 224x224. Dataset ini memiliki 11000 data citra penyakit daun tomat yang terdiri dari 10 kelas.

Setiap kelas memiliki 1.100 citra yang kemudian dibagi menjadi dua subset, yaitu data latih (train) dan data validasi (validation). Untuk masing-masing kelas dialokasikan 1.000 citra sebagai data latih dan 100 citra sebagai data validasi. Dengan demikian, jumlah total data latih adalah 10.000 citra dan jumlah total data validasi adalah 1.000 citra. Proporsi data latih terhadap keseluruhan data pada setiap kelas adalah 90,91%, sedangkan proporsi data validasi adalah 9,09%. Berikut penjelasan isi dataset pada Tabel 4. 1.

Tabel 4. 1 Informasi Dataset

No.	Penyakit Daun Tomat	Train	Val	Jumlah
1.	Mosaic Virus	1000	100	1100
2.	Target Spot	1000	100	1100
3.	Bacterial Spot	1000	100	1100
4.	Yellow Leaf Curl Virus	1000	100	1100
5.	Late Blight	1000	100	1100
6.	Leaf Mold	1000	100	1100
7.	Early Blight	1000	100	1100
8.	Spider Mites	1000	100	1100
9.	Tomato Healthy	1000	100	1100
10.	Septoria Leaf Spot	1000	100	1100

Adapun terdapat 10 kelas penyakit diantaranya tomato mosaic virus, target spot, bacterial spot, tomato yellow leaf curl virus, late blight, leaf mold, early blight, spider mites two-spotted spider mite, tomato healthy, dan septoria leaf spot yang ditunjukkan pada Gambar 4. 1 Jenis Penyakit pada Daun Tomat.



(a) Bacterial Spot

(b) Early Blight

(c) Healthy



(d) Late Blight

(e) Leaf Mold

(f) Septoria Leaf Spot



(g) Spider Mites

(h) Target Spot

(i) Mosaic Virus



(j) Yellow Leaf Curl
Virus

Gambar 4. 1 Jenis Penyakit pada Daun Tomat.

4.2 Split Data

Pada tahap split data, dataset yang awalnya terdiri dari 10.000 gambar untuk data train dan 1.000 gambar untuk validation (val), dengan total 11.000 gambar dari 10 kelas, akan dibagi menjadi tiga bagian, yaitu train, test, dan validation. Tujuan dari pembagian ini adalah untuk menambahkan data test, yang sebelumnya tidak ada dalam dataset. Setelah pembagian, data train akan berjumlah 8.000 gambar, validation tetap 1.000 gambar, dan data test yang baru ditambahkan akan berjumlah 2.000 gambar.

Proses ini dilakukan dengan cara mengurangi 2.000 gambar dari data train dan memindahkannya ke data test. Pemindahan ini dilakukan secara acak untuk memastikan distribusi kelas tetap seimbang antara train dan test. Dengan pembagian ini, model akan dilatih menggunakan 8.000 gambar dari train, divalidasi dengan 1.000 gambar dari validation, dan akhirnya diuji pada 2.000 gambar test yang sebelumnya tidak digunakan dalam proses pelatihan atau validasi.

Pembagian ini penting untuk memastikan model dapat dievaluasi dengan data yang tidak pernah dilihat sebelumnya, memberikan gambaran yang lebih akurat mengenai performa model pada data dunia nyata. Dengan demikian, dataset yang sebelumnya hanya terdiri dari data train dan validation sekarang terbagi menjadi tiga bagian, yaitu 8.000 gambar untuk train, 2.000 gambar untuk test, dan 1.000 gambar untuk validation. Pembagian ini memungkinkan evaluasi yang lebih objektif dan membantu model untuk generalisasi dengan lebih baik pada data yang

belum pernah dilihat sebelumnya. Jumlah dataset setelah split data dapat dilihat pada Tabel 4. 2 Tabel Hasil Split Data Tabel 4. 2 berikut.

Tabel 4. 2 Tabel Hasil Split Data

No.	Penyakit Daun Tomat	Train	Test	Val
1.	Mosaic Virus	800	200	100
2.	Target Spot	800	200	100
3.	Bacterial Spot	800	200	100
4.	Yellow Leaf Curl Virus	800	200	100
5.	Late Blight	800	200	100
6.	Leaf Mold	800	200	100
7.	Early Blight	800	200	100
8.	Spider Mites	800	200	100
9.	Tomato Healthy	800	200	100
10.	Septoria Leaf Spot	800	200	100
Jumlah		8000	2000	1000

4.3 Augmentasi Data

Pada bagian ini, dilakukan 5 skenario augmentasi data yang nantinya akan diterapkan pada model YOLO11 dalam proses klasifikasi penyakit pada daun tomat. Teknik augmentasi yang digunakan adalah Advanced Mixing Augmentation, Photometric Augmentation, dan Spatial Transformations.

4.3.1 Non Augmentattion

Eksperimen dilakukan untuk membandingkan hasil klasifikasi penyakit pada daun tomat menggunakan model YOLO11 tanpa penerapan teknik augmentasi data. Tujuan utama dari eksperimen ini adalah untuk mengukur kinerja baseline dari model, yaitu bagaimana model dapat mengenali penyakit pada daun tomat dengan dataset yang asli tanpa ada perubahan atau peningkatan data. Hal ini sangat penting karena memberikan gambaran mengenai kemampuan model dalam

memprediksi dengan data yang terbatas, serta sejauh mana model dapat bekerja dengan data yang tidak mengalami variasi atau perbaikan.

Selain itu, eksperimen Non-Augmentation juga bertujuan untuk memberikan gambaran awal tentang kekuatan dan kelemahan model sebelum penerapan teknik augmentasi. Dengan menggunakan data yang tidak diubah, bisa memahami bagaimana model berfungsi dalam kondisi standar dan mengidentifikasi area yang mungkin membutuhkan perbaikan. Hasil yang diperoleh dari eksperimen ini nantinya akan digunakan sebagai acuan untuk membandingkan efektivitas teknik-teknik augmentasi yang diterapkan pada eksperimen selanjutnya. Dengan kata lain, eksperimen Non-Augmentation memberikan dasar yang jelas mengenai performa awal model yang kemudian dapat dievaluasi dan dibandingkan dengan hasil dari model-model yang dilatih menggunakan augmentasi data, untuk melihat apakah teknik augmentasi memberikan kontribusi signifikan terhadap peningkatan kinerja model. Adapun hasil dataset tanpa augmentasi dapat dilihat pada.

Tabel 4. 3 Tabel Hasil Dataset Non Augmentation

No.	Penyakit Daun Tomat	Sebelum Augmentasi			Setelah Augmentasi		
		Train	Test	Val	Train	Test	Val
1.	Mosaic Virus	800	200	100	800	200	100
2.	Target Spot	800	200	100	800	200	100
3.	Bacterial Spot	800	200	100	800	200	100
4.	Yellow Leaf Curl Virus	800	200	100	800	200	100
5.	Late Blight	800	200	100	800	200	100
6.	Leaf Mold	800	200	100	800	200	100
7.	Early Blight	800	200	100	800	200	100
8.	Spider Mites	800	200	100	800	200	100
9.	Tomato Healthy	800	200	100	800	200	100
10.	Septoria Leaf Spot	800	200	100	800	200	100
Jumlah		8000	2000	1000	8000	2000	1000

4.3.2 Advanced Mixing Augmentation

Teknik augmentasi advanced mixing augmentation ini dilakukan untuk meningkatkan keberagaman dan kualitas dataset yang digunakan untuk melatih model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat. Proses ini bertujuan untuk memperkenalkan variasi yang lebih besar dalam gambar, yang memungkinkan model untuk lebih tahan terhadap perubahan dan gangguan yang mungkin muncul pada gambar dunia nyata. Gambar 4. 2 Source Code Advanced Mixing Augmentation.

```

INPUT_DIR = r"D:\52\ Eksperimen\Thesis\Advanced Mixing Augmentation\train"
OUTPUT_DIR = r"D:\52\ Eksperimen\Thesis\Advanced Mixing Augmentation\Augmentation"

IMG_EXTS = [".jpg", ".jpeg", ".png", ".bmp", ".webp", ".tiff", ".tiff"]
MAX_PER_CLASS = 400
random.seed(42)
np.random.seed(42)

# ----- Augmentation Pipeline for Advanced Mixing -----
T = A.Compose([
    # Color and Contrast Adjustments
    A.OneOf([
        A.RandomBrightnessContrast(brightness_limit=0.25, contrast_limit=0.25, p=1.0),
        A.ColorJitter(brightness=0.25, contrast=0.25, saturation=0.25, hue=0.05, p=1.0),
        A.RandomGamma(gamma_limit=(80, 120), p=1.0),
    ], p=1.0),

    # Adding Noise
    A.GaussNoise(var_limit=(10.0, 40.0), p=0.3), # Gaussian Noise

    # Mixup / CutMix transformations
    A.OneOf([
        A.RandomRotate90(p=0.5),
        A.HorizontalFlip(p=0.5),
        A.VerticalFlip(p=0.5),
    ], p=1.0),

    # Additional Transformations
    A.CLAHE(clip_limit=2.0, tile_grid_size=(8, 8), p=0.5),
    A.FancyPCA(alpha=0.1, p=0.5),
])

```

Gambar 4. 2 Source Code Advanced Mixing Augmentation

Berikut adalah penjelasan rinci mengenai langkah-langkah yang dilakukan sesuai dengan source code yang ada. Pertama, skrip mengatur direktori input

(INPUT_DIR) dan output (OUTPUT_DIR), di mana gambar yang akan diproses disimpan di direktori input dan gambar hasil augmentasi akan disimpan di direktori output. Skrip memeriksa ekstensi file gambar yang valid (seperti .jpg, .jpeg, .png, .bmp, dll.) dan memastikan hanya gambar dengan ekstensi tersebut yang diproses. Selain itu, jumlah gambar yang diproses untuk setiap kelas dibatasi hingga 400 gambar untuk menjaga keseimbangan dataset dan menghindari pemrosesan yang berlebihan pada kelas yang sangat besar.

Selanjutnya, skrip mendefinisikan pipeline augmentasi menggunakan pustaka Albumentations yang melibatkan serangkaian transformasi untuk meningkatkan variasi gambar

Penyesuaian Kecerahan dan Kontras, beberapa teknik digunakan untuk menyesuaikan kecerahan dan kontras gambar secara acak, termasuk RandomBrightnessContrast, ColorJitter, dan RandomGamma. Random Brightness Contrast secara acak mengubah kecerahan dan kontras gambar dalam rentang yang telah ditentukan ($\text{brightness_limit}=0.25$ dan $\text{contrast_limit}=0.25$), yang memungkinkan model mengenali objek dengan berbagai kondisi pencahayaan. ColorJitter mengubah kecerahan, kontras, saturasi, dan hue gambar untuk menciptakan variasi warna. RandomGamma digunakan untuk menyesuaikan gamma gambar, yang dapat mempengaruhi pencahayaan secara keseluruhan.

Penambahan noise, teknik GaussNoise digunakan untuk menambahkan noise acak pada gambar. Noise Gaussian ditambahkan pada gambar dengan varians yang ditentukan ($\text{var_limit}=(10.0, 40.0)$), yang membantu model belajar mengenali objek meskipun ada gangguan visual atau noise pada gambar.

Selanjutnya skrip memilih dua gambar secara acak dan menggabungkannya menggunakan teknik Mixup, di mana kedua gambar dicampur dengan bobot yang dihitung menggunakan distribusi Beta ($\lambda = np.random.beta(\alpha, \alpha)$). Teknik ini menghasilkan gambar campuran yang menggabungkan dua gambar asli secara proporsional, yang dapat meningkatkan kemampuan model dalam mengenali pola yang lebih kompleks. Jika skrip memilih teknik CutMix, maka gambar akan dipotong secara acak menggunakan distribusi Beta untuk menentukan ukuran bagian yang dipotong. Bagian gambar yang terpotong akan dipasteikan pada gambar lainnya. Teknik ini sangat berguna untuk mengurangi overfitting dan membantu model memahami objek yang lebih bervariasi secara visual. Skrip menggunakan parameter seperti `cut_rat` untuk menentukan ukuran potongan berdasarkan distribusi Beta dan kemudian menempelkan potongan tersebut pada gambar lain.

Penggabungan teknik augmentasi lainnya yaitu CLAHE (Contrast Limited Adaptive Histogram Equalization) dan FancyPCA. Teknik CLAHE ini diterapkan dengan probabilitas 50% menggunakan parameter `clip_limit=2.0` dan `tile_grid_size=(8, 8)`. CLAHE meningkatkan kontras lokal pada gambar, terutama pada area yang memiliki pencahayaan yang kurang, sehingga memungkinkan model untuk mengenali detail yang lebih halus pada daun tomat yang terpengaruh oleh penyakit. Sedangkan Teknik FancyPCA digunakan untuk melakukan transformasi berbasis Principal Component Analysis (PCA) pada warna gambar dengan parameter `alpha=0.1`. Teknik ini berfungsi untuk mengubah warna gambar,

menghasilkan variasi yang lebih besar dalam dataset, dan memungkinkan model untuk belajar mengenali pola penyakit meskipun ada perubahan warna pada daun.

Setelah transformasi augmentasi diterapkan, gambar akan disimpan dengan nama file unik yang mengandung UUID (Universal Unique Identifier) untuk memastikan bahwa setiap gambar hasil augmentasi memiliki nama yang berbeda dan tidak tumpang tindih dengan gambar asli. Nama file gambar hasil augmentasi akan memiliki format {nama_file_asli}_aug_{uid}.jpg, di mana uid adalah ID acak yang dihasilkan menggunakan `uuid.uuid4().hex[:8]` untuk memberikan keunikan pada setiap gambar.

Skrip kemudian melanjutkan proses augmentasi ini untuk setiap kelas dalam dataset, di mana setiap gambar diproses secara individual, diterapkan teknik Mixup atau CutMix secara acak, dan seluruh pipeline augmentasi lainnya diterapkan. Hasil akhirnya disimpan dalam direktori output yang sesuai dengan nama kelas gambar asalnya, memastikan struktur folder tetap terorganisir.

Setelah augmentasi, jumlah gambar untuk setiap jenis penyakit daun tomat meningkat menjadi 400 gambar, dengan total 4.000 gambar dari 10 kelas yang berbeda. Gambar kemudian dimasukkan ke data train sesuai dengan kelas penyakitnya. Adapun gambar hasil augmentasi advanced mixing augmentation dapat dilihat pada Tabel 4. 4 berikut.

Tabel 4. 4 Hasil Gambar Setelah Advanced Mixing Augmentation.

No.	Penyakit Daun Tomat	Sebelum Augmentasi			Setelah Augmentasi		
		Train	Test	Val	Train	Test	Val
1.	Mosaic Virus	800	200	100	1200	200	100
2.	Target Spot	800	200	100	1200	200	100
3.	Bacterial Spot	800	200	100	1200	200	100
4.	Yellow Leaf Curl Virus	800	200	100	1200	200	100

5.	Late Blight	800	200	100	1200	200	100
6.	Leaf Mold	800	200	100	1200	200	100
7.	Early Blight	800	200	100	1200	200	100
8.	Spider Mites	800	200	100	1200	200	100
9.	Tomato Healthy	800	200	100	1200	200	100
10.	Septoria Leaf Spot	800	200	100	1200	200	100
Jumlah		8000	2000	1000	12000	2000	2000

Dengan menerapkan teknik-teknik augmentasi ini, dataset yang digunakan untuk melatih model menjadi lebih beragam dan kaya, dengan berbagai variasi pencahayaan, noise, warna, dan komposisi gambar yang lebih kompleks. Hal ini memungkinkan model untuk lebih baik mengenali penyakit pada daun tomat meskipun gambar yang dihadapi dalam situasi dunia nyata memiliki berbagai kondisi visual yang berbeda. Gambar 4. 3 adalah hasil dari augmentasi data advanced mixing augmentation



Gambar 4. 3 Hasil dari Advanced Mixing Augmentation

4.3.3 Photometric Augmentation

Photometric Augmentation, yang melibatkan modifikasi warna dan pencahayaan citra, menunjukkan hasil yang signifikan dalam meningkatkan ketahanan model terhadap variasi pencahayaan dan kondisi cuaca yang berbeda. Model yang dilatih dengan augmentasi ini mampu mengenali penyakit pada daun tomat dengan akurasi yang lebih tinggi pada citra dengan kondisi pencahayaan yang kurang ideal.

Teknik augmentasi photometric augmentation ini, dilakukan serangkaian transformasi warna untuk memperkaya dataset gambar dengan variasi yang lebih beragam. Proses augmentasi bertujuan untuk meningkatkan keberagaman gambar dalam hal pencahayaan, warna, dan kontras, yang memungkinkan model untuk lebih robust dalam mengenali penyakit pada daun tomat, meskipun gambar yang diberikan memiliki perbedaan kondisi pencahayaan atau warna. Gambar 4. 4 Source Code Photometric Augmentation.

```
# ----- Photometric-only pipeline -----
T = A.Compose([
  A.OneOf([
    A.RandBrightnessContrast(brightness_limit=0.25, contrast_limit=0.25, p=1.0),
    A.ColorJitter(brightness=0.25, contrast=0.25, saturation=0.25, hue=0.05, p=1.0),
    A.RandomGamma(gamma_limit=(80, 120), p=1.0),
  ], p=1.0),
  A.OneOf([
    A.RandomSaturationValue(hue_shift_limit=10, sat_shift_limit=20, val_shift_limit=10, p=1.0),
    A.RGBShift(r_shift_limit=15, g_shift_limit=15, b_shift_limit=15, p=1.0),
  ], p=0.8),
  A.OneOf([
    A.CLAHE(clip_limit=2.0, tile_grid_size=(8, 8), p=1.0),
    A.FancyPCA(alpha=0.1, p=1.0),
  ], p=0.5),
  A.GaussianNoise(var_limit=(10.0, 30.0), mean=0, p=0.2),
  A.ChannelShuffle(p=0.05),
  A.ToGray(p=0.05),
], p=1.0)
```

Gambar 4. 4 Source Code Photometric Augmentation

Berikut adalah penjelasan rinci mengenai langkah-langkah yang dilakukan dalam skrip ini, sesuai dengan source code yang ada. Skrip dimulai dengan menentukan direktori input (INPUT_DIR) yang berisi gambar-gambar asli yang akan diproses, dan direktori output (OUTPUT_DIR) untuk menyimpan gambar hasil augmentasi. Skrip memverifikasi bahwa file gambar yang diproses memiliki ekstensi yang valid seperti .jpg, .jpeg, .png, .bmp, dll., untuk memastikan hanya gambar yang sesuai yang diproses. Selain itu, jumlah gambar yang diproses per

kelas dibatasi hingga 400 gambar untuk menjaga keseimbangan data, dan setiap gambar akan diaugmentasi 1 kali.

Setelah itu terdapat perintah pipeline augmentasi untuk mengubah warna dengan menggunakan pustaka Albumentations untuk mendefinisikan pipeline augmentasi yang berfokus pada transformasi warna. Beberapa teknik augmentasi warna yang diterapkan antara lain. Random Brightness Contrast, ColorJitter, dan RandomGamma. Teknik ini mengubah kecerahan, kontras, dan gamma gambar secara acak. Teknik ini membantu model mengenali objek di bawah berbagai kondisi pencahayaan yang berbeda. Setiap teknik diterapkan dengan probabilitas yang telah ditentukan ($p=1.0$), memastikan bahwa salah satu dari transformasi ini selalu diterapkan pada gambar.

Teknik HueSaturationValue dan RGBShift diterapkan untuk mengubah hue, saturasi, dan nilai warna gambar. Dengan probabilitas 0.8, teknik-teknik ini memungkinkan model untuk mengenali objek meskipun ada variasi warna dan saturasi, yang sering terjadi pada citra yang diambil dalam kondisi pencahayaan yang berbeda.

CLAHE dan FancyPCA: Teknik CLAHE (Contrast Limited Adaptive Histogram Equalization) digunakan untuk meningkatkan kontras lokal pada gambar, sementara FancyPCA mengubah warna gambar menggunakan transformasi berbasis PCA. Kedua teknik ini diterapkan dengan probabilitas 0.5, bertujuan untuk meningkatkan detail visual pada gambar dan membantu model mengenali objek dalam kondisi kontras rendah atau perubahan warna yang besar.

Selain transformasi warna, skrip ini juga menambahkan beberapa teknik augmentasi tambahan untuk meningkatkan variasi gambar. Teknik GaussNoise menambahkan noise Gaussian pada gambar dengan varians yang acak ($\text{var_limit}=(10.0, 40.0)$) dan rata-rata nol ($\text{mean}=0$). Ini bertujuan untuk membuat model lebih toleran terhadap gangguan yang mungkin ada pada gambar asli, seperti noise yang dihasilkan oleh sensor kamera.

Selain itu, dilakukan transformasi ChannelShuffle yang akan mengacak urutan saluran warna pada gambar (merah, hijau, biru) dengan probabilitas 0.05, yang membantu model untuk lebih fleksibel dalam mengenali objek meskipun ada perubahan urutan saluran warna.

ToGray: Dengan probabilitas 0.05, gambar dapat diubah menjadi citra hitam-putih (grayscale). Ini membantu model untuk mengenali objek berdasarkan tekstur dan bentuk, bukan hanya warna.

Setelah pipeline augmentasi ditentukan, dilanjutkan dengan membaca gambar dari direktori input. Gambar yang dibaca kemudian diubah dari format BGR (Blue-Green-Red) ke format RGB (Red-Green-Blue) menggunakan OpenCV untuk diproses dengan transformasi yang telah didefinisikan. Skrip ini menerapkan augmentasi pada setiap gambar dengan menggunakan fungsi `T(image=rgb) ["image"]`, yang secara otomatis memilih dan menerapkan salah satu dari transformasi yang telah didefinisikan dalam pipeline. Proses ini dilakukan dengan probabilitas yang telah ditentukan di setiap langkah augmentasi.

Setelah gambar selesai diaugmentasi, gambar hasil augmentasi disimpan dalam format JPG menggunakan fungsi `imwrite_jpg`. Nama file gambar hasil

augmentasi diberikan UUID (Universal Unique Identifier) untuk memastikan bahwa setiap gambar hasil augmentasi memiliki nama yang unik dan tidak terjadi duplikasi dengan gambar asli. Nama file gambar hasil augmentasi akan memiliki format {nama_file_asli}_phaug_{uid}.jpg, yang memungkinkan identifikasi yang mudah. Gambar hasil augmentasi kemudian disimpan di dalam direktori output yang sesuai dengan nama kelas gambar asalnya.

Proses augmentasi ini diterapkan secara terpisah untuk setiap kelas dalam dataset. Skrip ini akan memeriksa setiap folder kelas dalam direktori input, memproses gambar-gambar yang ada di dalamnya, dan menyimpan gambar yang telah diaugmentasi dalam folder yang sesuai di direktori output. Setelah proses augmentasi selesai, skrip mencetak jumlah gambar yang telah diproses untuk setiap kelas dan menyimpan hasilnya di direktori yang sesuai.

Setelah augmentasi, jumlah gambar untuk setiap jenis penyakit daun tomat meningkat menjadi 400 gambar, dengan total 4.000 gambar dari 10 kelas yang berbeda. Gambar kemudian dimasukkan ke data train sesuai dengan kelas penyakitnya. Adapun hasil augmentasi dapat dilihat di Tabel 4. 5.

Tabel 4. 5 Hasil Gambar Setelah Photometric Augmentation

No.	Penyakit Daun Tomat	Sebelum Augmentasi			Setelah Augmentasi		
		Train	Test	Val	Train	Test	Val
1.	Mosaic Virus	800	200	100	1200	200	100
2.	Target Spot	800	200	100	1200	200	100
3.	Bacterial Spot	800	200	100	1200	200	100
4.	Yellow Leaf Curl Virus	800	200	100	1200	200	100
5.	Late Blight	800	200	100	1200	200	100
6.	Leaf Mold	800	200	100	1200	200	100
7.	Early Blight	800	200	100	1200	200	100
8.	Spider Mites	800	200	100	1200	200	100
9.	Tomato Healthy	800	200	100	1200	200	100
10.	Septoria Leaf Spot	800	200	100	1200	200	100

Jumlah	8000	2000	1000	12000	2000	1000
--------	------	------	------	-------	------	------

Dengan menggunakan teknik-teknik augmentasi ini, dataset yang digunakan untuk melatih model menjadi lebih kaya dan bervariasi. Variasi dalam pencahayaan, warna, kontras, dan penambahan noise pada gambar memungkinkan model untuk lebih robust dalam mengenali penyakit pada daun tomat dalam berbagai kondisi dunia nyata. Augmentasi ini juga membantu model dalam mengurangi overfitting dengan memperkenalkan variasi yang lebih banyak dan memungkinkan model untuk generalisasi lebih baik pada data yang belum pernah dilihat sebelumnya. Adapun hasil augmentasi dapat dilihat di Gambar 4. 5.



Gambar 4. 5 Hasil Photometric Augmentation

4.3.4 Spatial Transformations

Spatial Transformations Augmentation, pada augmentasi ini dilakukan serangkaian transformasi spasial untuk memperkaya dataset gambar dengan variasi posisi dan orientasi objek, yang memungkinkan model untuk lebih robust dalam mengenali penyakit pada daun tomat di berbagai kondisi spasial yang berbeda.

Augmentasi ini bertujuan untuk membuat gambar lebih bervariasi dalam hal rotasi, perubahan posisi, dan distorsi, sehingga model dapat belajar untuk mengenali objek meskipun ada perubahan dalam orientasi atau posisi gambar. Berikut adalah penjelasan rinci mengenai langkah-langkah yang dilakukan dalam

skrip ini, sesuai dengan source code yang ada. Gambar 4. 6 Source Code Spatial Transformations.

```

INPUT_DIR = e"0152/Experiment/thesis/Spatial_Transformations/train"
OUTPUT_DIR = e"0152/Experiment/thesis/Spatial_Transformations/augmentation"
IMG_EXTS = ["*.jpg", "*.jpeg", "*.png", "*.bmp", "*.webp", "*.tiff", "*.tif"]
MAX_PER_CLASS, MAX_PER_IMAGE = 400, 1
random.seed(42); np.random.seed(42)

try:
    import albumentations as A
except ImportError:
    raise SystemExit("install dulu pip install albumentations (source python code)")

T = A.Compose([
    A.OneOf([
        A.Rotate(limit=30, border_mode=cv2.BORDER_REFLECT_101, interpolation=cv2.INTER_LINEAR, p=1),
        A.ShiftScaleRotate(shift_x=0.1, shift_y=0.1, border_mode=cv2.BORDER_REFLECT_101, interpolation=cv2.INTER_LINEAR, p=1),
        A.Perspective(scale=(0.0, 0.0), keep_size=True, pad_mode=cv2.BORDER_REFLECT_101, p=1),
    ], p=1),
    A.OneOf([A.HorizontalFlip(p=1), A.VerticalFlip(p=1), A.RandomRotate90(p=1), p=1),
    A.OneOf([A.PlasticTransform(alpha=10, sigma=1, alpha_affine=10, border_mode=cv2.BORDER_REFLECT_101, p=1),
        A.GridDistortion(num_steps=5, distort_limit=0.2, border_mode=cv2.BORDER_REFLECT_101, p=1),
        A.GridDistortionDistort(limit=0.2, shift_limit=0.05, border_mode=cv2.BORDER_REFLECT_101, p=1)], p=0.5),
])

```

Gambar 4. 6 Source Code Spatial Transformations

Proses dimulai dengan mendefinisikan direktori input (INPUT_DIR) yang berisi gambar asli yang akan diproses, dan direktori output (OUTPUT_DIR) tempat gambar hasil augmentasi akan disimpan. Skrip memverifikasi bahwa file gambar yang diproses memiliki ekstensi yang valid, seperti .jpg, .jpeg, .png, dan lainnya. Jumlah gambar yang diproses untuk setiap kelas dibatasi hingga 400 gambar, dan setiap gambar akan diaugmentasi 1 kali. Hal ini untuk menjaga keseimbangan dataset dan mencegah pengolahan data yang berlebihan.

Skrip ini menggunakan pustaka Albumentations untuk mendefinisikan pipeline augmentasi spasial yang melibatkan serangkaian teknik untuk mengubah posisi, orientasi, dan struktur gambar. Beberapa teknik yang diterapkan yaitu teknik rotate dan shiftscalerotate,

Teknik Rotate dan ShiftScaleRotate digunakan untuk memutar gambar dalam rentang sudut yang telah ditentukan (misalnya, rotasi hingga 30 derajat). Transformasi ini juga mencakup perubahan skala dan pergeseran posisi gambar.

Proses ini memungkinkan model untuk mengenali objek meskipun posisinya bergeser atau objek dalam gambar diputar.

Selanjutnya penggunaan transformasi perspektif untuk memberi efek distorsi perspektif pada gambar, di mana objek dalam gambar dipindahkan untuk memberikan perspektif yang berbeda. Ini membantu model untuk belajar mengenali objek dalam posisi yang tidak selalu statis atau lurus.

Teknik `HorizontalFlip`, `VerticalFlip`, dan `RandomRotate90` diterapkan dengan probabilitas 0.7. Teknik ini membuat model lebih kuat dalam mengenali objek dari berbagai sudut pandang dan posisi. Misalnya, gambar yang dibalik secara horizontal atau vertikal memungkinkan model untuk mengenali objek tanpa bergantung pada posisi objek yang spesifik dalam gambar.

Adapun teknik `ElasticTransform` digunakan untuk menerapkan distorsi elastis pada gambar. Distorsi ini mengubah gambar dengan cara yang menyerupai perubahan elastis, menciptakan tampilan gambar yang lebih alami dan dinamis. Teknik ini diterapkan dengan probabilitas 0.5 dan memungkinkan model untuk belajar dari gambar yang mengalami perubahan tekstur atau deformasi.

Teknik `GridDistortion` dan `OpticalDistortion` juga diterapkan dengan probabilitas 0.5. Teknik-teknik ini memperkenalkan distorsi berbasis grid atau optik pada gambar, yang menghasilkan perubahan bentuk yang lebih kompleks. Distorsi ini membantu model mengenali objek meskipun ada perubahan besar pada bentuk atau struktur gambar, yang sering terjadi dalam kondisi dunia nyata.

Setelah pipeline augmentasi spasial diterapkan, skrip melanjutkan dengan membaca setiap gambar dari direktori input. Setiap gambar kemudian diubah dari

format BGR (Blue-Green-Red) ke RGB (Red-Green-Blue) untuk diproses oleh fungsi augmentasi. Skrip ini memilih secara acak dan menerapkan salah satu dari teknik augmentasi spasial yang telah didefinisikan dalam pipeline menggunakan fungsi `T(image=rgb) ["image"]`. Setelah augmentasi diterapkan, gambar dikonversi kembali ke format BGR untuk disimpan.

Setelah gambar diaugmentasi, hasil gambar disimpan dalam format JPG dengan nama file yang unik menggunakan UUID (Universal Unique Identifier). Nama file gambar hasil augmentasi mengikuti format `{nama_file_asli}_aug_{uid}.jpg`, di mana uid adalah ID acak yang dihasilkan menggunakan `uuid.uuid4().hex[:8]` untuk memastikan nama file unik dan mudah diidentifikasi. Gambar yang telah diaugmentasi kemudian disimpan di dalam folder yang sesuai dengan kelas gambar asalnya di direktori output.

Proses augmentasi ini diterapkan untuk setiap kelas dalam dataset, di mana setiap kelas gambar diproses secara terpisah. Skrip ini memeriksa setiap folder kelas dalam direktori input, memproses gambar-gambar yang ada di dalamnya, dan menyimpan gambar yang telah diaugmentasi dalam folder yang sesuai di direktori output. Setelah augmentasi selesai, skrip mencetak jumlah gambar yang telah diproses untuk setiap kelas, memastikan bahwa setiap langkah augmentasi berhasil diterapkan.

Setelah augmentasi, jumlah gambar untuk setiap jenis penyakit daun tomat meningkat menjadi 400 gambar, dengan total 4.000 gambar dari 10 kelas yang berbeda. Gambar kemudian dimasukkan ke data train sesuai dengan kelas penyakitnya. Adapun hasil augmentasi dapat dilihat di Tabel 4. 6 berikut.

Tabel 4. 6 Hasil Gambar Setelah Augmentasi Spatial Transformations

No.	Penyakit Daun Tomat	Sebelum Augmentasi			Setelah Augmentasi		
		Train	Test	Val	Train	Test	Val
1.	Mosaic Virus	800	200	100	1200	200	100
2.	Target Spot	800	200	100	1200	200	100
3.	Bacterial Spot	800	200	100	1200	200	100
4.	Yellow Leaf Curl Virus	800	200	100	1200	200	100
5.	Late Blight	800	200	100	1200	200	100
6.	Leaf Mold	800	200	100	1200	200	100
7.	Early Blight	800	200	100	1200	200	100
8.	Spider Mites	800	200	100	1200	200	100
9.	Tomato Healthy	800	200	100	1200	200	100
10.	Septoria Leaf Spot	800	200	100	1200	200	100
Jumlah		8000	2000	1000	12000	2000	2000

Dengan menerapkan teknik-teknik augmentasi spasial ini, dataset menjadi lebih bervariasi dalam hal posisi, orientasi, dan struktur gambar. Hal ini meningkatkan kemampuan model untuk mengenali penyakit pada daun tomat, meskipun gambar yang diberikan memiliki perubahan posisi atau distorsi. Augmentasi ini membantu model menjadi lebih fleksibel dan mampu mengenali objek dalam berbagai kondisi spasial, yang memungkinkan model untuk mengklasifikasikan penyakit dengan lebih akurat pada gambar dunia nyata yang lebih beragam. Gambar 4. 7 adalah hasil augmentasi spatial transformations.



Gambar 4. 7 Hasil Augmentasi Spatial Transformations

4.3.5 All Augmentation

All Augmentation dilakukan dengan menggabungkan tiga teknik augmentasi data, yaitu Advanced Mixing, Photometric, dan Spatial Transformations untuk meningkatkan keberagaman dan kualitas dataset yang digunakan untuk melatih model YOLO11 dalam mengklasifikasikan penyakit pada daun tomat. Penggabungan ini bertujuan untuk memperkenalkan berbagai variasi pada gambar, mulai dari perubahan warna, distorsi spasial, hingga perubahan komposisi gambar dengan teknik seperti Mixup dan CutMix. Proses ini memungkinkan model untuk lebih tangguh dalam mengenali penyakit pada daun tomat meskipun gambar yang diberikan memiliki variasi besar dalam kondisi pencahayaan, posisi, dan komposisi gambar.

Pertama, pipeline augmentasi dimulai dengan teknik Advanced Mixing yang menggabungkan beberapa gambar menggunakan teknik Mixup dan CutMix. Teknik Mixup menggabungkan dua gambar secara proporsional, menggunakan distribusi Beta untuk menghasilkan gambar campuran, sementara CutMix memotong bagian dari gambar dan menyisipkannya ke gambar lain secara acak. Kedua teknik ini memberikan model data yang lebih variatif, di mana model harus belajar mengenali pola meskipun ada kombinasi objek yang berbeda atau bagian-bagian gambar yang digabungkan. Teknik Advanced Mixing ini meningkatkan kemampuan model dalam mengklasifikasikan penyakit pada daun tomat meskipun terdapat perubahan besar pada komposisi visual gambar.

Kedua, Photometric Augmentation diterapkan untuk mengubah elemen warna dan kontras gambar. Transformasi seperti RandomBrightnessContrast, ColorJitter, dan RandomGamma diterapkan untuk

mengubah kecerahan, kontras, dan saturasi gambar secara acak, membuat model lebih fleksibel dalam mengenali objek dalam berbagai kondisi pencahayaan dan warna. Selain itu, teknik seperti HueSaturationValue dan RGBShift memperkenalkan variasi pada hue dan saturasi gambar, sementara CLAHE dan FancyPCA meningkatkan kontras lokal dan variasi warna. Dengan probabilitas yang telah ditentukan, teknik-teknik ini membantu model mengenali daun tomat yang terkena penyakit meskipun ada perubahan besar pada warna dan pencahayaan gambar.

Ketiga, teknik Spatial Transformations diterapkan untuk mengubah posisi dan orientasi gambar. Transformasi ini mencakup rotasi gambar menggunakan Rotate dan ShiftScaleRotate, yang mengubah posisi objek dalam gambar. Selain itu, teknik Perspective diterapkan untuk memberi distorsi perspektif pada gambar, sementara teknik Flip baik secara horizontal, vertikal, maupun rotasi acak memberikan variasi orientasi gambar. Untuk distorsi lebih lanjut, ElasticTransform, GridDistortion, dan OpticalDistortion diterapkan untuk memberikan efek distorsi elastis dan optik, yang memungkinkan model mengenali objek meskipun ada perubahan bentuk atau struktur visual.

Dengan menggabungkan ketiga teknik augmentasi ini, dataset yang digunakan untuk melatih model menjadi jauh lebih bervariasi dan kaya. Gambar yang dihasilkan dari augmentasi ini mencakup variasi pencahayaan, posisi objek, kontras, dan noise yang sangat berbeda dari gambar asli. Proses augmentasi yang menyeluruh ini membuat model lebih mampu mengenali penyakit pada daun tomat dalam kondisi yang lebih kompleks dan variatif. Selain itu, dengan penerapan

Mixup dan CutMix, model dapat belajar dari kombinasi objek dan potongan gambar yang sangat berbeda, yang pada gilirannya mengurangi kemungkinan overfitting dan membantu model untuk menggeneralisasi dengan lebih baik pada data yang belum pernah dilihat sebelumnya.

Setelah melakukan serangkaian augmentasi dengan menggabungkan Advanced Mixing, Photometric, dan Spatial Transformations, dataset yang digunakan untuk melatih model YOLO11 menjadi jauh lebih kaya dan beragam. Setelah augmentasi, jumlah gambar untuk setiap jenis penyakit daun tomat meningkat menjadi 1200 gambar, dengan total 20.000 gambar dari 10 kelas yang berbeda. Gambar kemudian dimasukkan ke data train sesuai dengan kelas penyakitnya. Tabel 4. 7 adalah hasil setelah augmentasi.

Tabel 4. 7 Gambar Hasil All Augmentation

No.	Penyakit Daun Tomat	Sebelum Augmentasi			Setelah Augmentasi		
		Train	Test	Val	Train	Test	Val
1.	Mosaic Virus	800	200	100	2000	200	100
2.	Target Spot	800	200	100	2000	200	100
3.	Bacterial Spot	800	200	100	2000	200	100
4.	Yellow Leaf Curl Virus	800	200	100	2000	200	100
5.	Late Blight	800	200	100	2000	200	100
6.	Leaf Mold	800	200	100	2000	200	100
7.	Early Blight	800	200	100	2000	200	100
8.	Spider Mites	800	200	100	2000	200	100
9.	Tomato Healthy	800	200	100	2000	200	100
10.	Septoria Leaf Spot	800	200	100	2000	200	100
Jumlah		8000	2000	1000	20000	2000	1000

Secara keseluruhan, penggabungan Advanced Mixing, Photometric, dan Spatial Transformations dalam satu pipeline augmentasi memberikan keberagaman data yang memungkinkan model untuk menjadi lebih robust dan akurat dalam mengenali penyakit pada daun tomat. Teknik-teknik ini bekerja bersama-sama

untuk memperkenalkan berbagai variasi visual yang membantu model belajar dari pola yang lebih kompleks dan variatif, yang sangat penting dalam aplikasi dunia nyata di mana gambar yang dihadapi sering kali memiliki kondisi pencahayaan, posisi, dan bentuk yang sangat berbeda.

4.4 Train Model YOLO11

Sebelum membahas hasil eksperimen, penting untuk memahami proses pelatihan model YOLO11. Adapun 5 skenario penelitian ini yaitu.

4.4.1 Non Augmentation

Pada eksperimen ini, beberapa konfigurasi penting diterapkan untuk melatih model YOLO11 dalam tugas klasifikasi penyakit pada daun tomat. Gambar 4. 8 adalah source code perintah untuk klasifikasi skenario pertama non augmentation YOLO11.

```
from ultralytics import YOLO
import os
import pandas as pd

# =====
# CONFIG
# =====
ROOT_DIR = r"D:\S2\Eksperimen\Thesis\NonAugmentation"
PROJECT_DIR = os.path.join(ROOT_DIR, "Hasil_Eksperimen")
MODEL_NAME = "yolo11n-cls.pt"
RUN_PREFIX = "NonAug_trial"

EPOCHS = 20
PATIENCE = 4
IMGSZ = 224
BATCH = 8
DEVICE = "cpu" # ganti "0" / "cuda:0" kalau GPU

N_TRIALS = 10
```

Gambar 4. 8 Source Code Skenario Non Augmentation

Pertama, `root_dir` menentukan lokasi dataset yang berisi folder `train`, `val`, dan `test`, dengan `train_dir`, `val_dir`, dan `test_dir` mengarah ke masing-masing folder dataset untuk pelatihan, validasi, dan pengujian. Pelatihan dilakukan dengan 20 epoch maksimal, di mana pelatihan akan dihentikan lebih awal jika tidak ada peningkatan dalam 4 epoch berturut-turut berkat pengaturan `early stopping` yang diatur pada `patience=4`. Ukuran gambar input diatur menjadi 224x224 piksel untuk memastikan gambar memiliki dimensi yang sesuai untuk model, dan `batch size` diatur ke 8 untuk setiap iterasi pelatihan.

Model yang digunakan adalah `YOLO11n-cls.pt`, yaitu model YOLO untuk klasifikasi yang sudah dipra-latih. Pelatihan model dilakukan menggunakan perintah `model.train()`, di mana berbagai parameter seperti jumlah epochs, `patience`, `imgsz`, dan `batch size` ditentukan. Proses pelatihan dijalankan pada CPU, namun dapat beralih ke GPU jika diperlukan. Selama pelatihan, data diproses menggunakan 4 `workers` untuk mempercepat pemuatan gambar, dan hasil pelatihan disimpan dalam folder yang ditentukan di `project`.

Dalam hal konfigurasi optimasi, YOLO secara otomatis memilih optimizer terbaik, yaitu AdamW, dengan `learning rate` yang disesuaikan otomatis menjadi 0.000714 dan momentum sebesar 0.9. Parameter lainnya, seperti `weight decay` diatur menjadi 0.0005 untuk membantu mencegah `overfitting`. Dengan pengaturan ini, model diharapkan dapat beradaptasi dengan baik selama pelatihan dan menghasilkan kinerja yang optimal. Selain itu, `early stopping` diaktifkan untuk menghentikan pelatihan jika tidak ada peningkatan dalam 4 epoch berturut-turut, memastikan efisiensi pelatihan. `N_TRIALS` adalah jumlah run/percobaan pelatihan

ulang yang dilakukan; N_TRIALS = 10 berarti model dilatih dan dievaluasi 10 kali untuk melihat konsistensi hasil. Tabel 4. 8 adalah parameter yang digunakan pada skenario ini.

Tabel 4. 8 Tabel Parameter Skenario Non Augmentation

No.	Parameter	Nilai
1.	Epoch	20
2.	Patience	4
3.	imgsz	224 × 224
4.	Batch Size	8
5.	Model	YOLO11n-cls.pt
6.	Device	CPU (opsional GPU)
7.	Workers	4
8.	Optimizer	AdamW
9.	Learning Rate	0.000714
10.	Momentum	0,9
11.	Weight Decay	0.0005
12.	N Trials	10

Setelah pelatihan selesai, model terbaik yang memiliki performa terbaik selama pelatihan (disimpan sebagai best.pt) dievaluasi pada test set. Hasil evaluasi ini mencakup metrik seperti Top-1 Accuracy dan Top-5 Accuracy, yang memberikan gambaran tentang kinerja model dalam mengklasifikasikan penyakit pada daun tomat. Dengan menggunakan pengaturan ini, model YOLO11 dilatih, dievaluasi, dan diuji dengan tujuan menghasilkan model yang akurat dalam klasifikasi penyakit daun tomat, dengan pemanfaatan teknik-teknik optimasi dan evaluasi yang efisien.

4.4.2 Advanced Mixing Augmentation

Pada eksperimen ini, berbagai konfigurasi penting diterapkan untuk melatih model YOLO11 menggunakan teknik Advanced Mixing Augmentation. Gambar 4. 9 adalah source code perintah untuk klasifikasi skenario kedua advanced mixing augmentation YOLO11.

```

from ultralytics import YOLO
import os
import pandas as pd

# -----
# CONFIG
# -----
ROOT_DIR = r"D:\S2\Eksperimen\Thesis\Advanced Mixing Augmentation"
PROJECT_DIR = os.path.join(ROOT_DIR, "Hasil_Eksperimen")
MODEL_NAME = "yolo11n-cls.pt"
RUN_PREFIX = "AMA_trial"

EPOCHS = 20
PATIENCE = 4
IMGSZ = 224
BATCH = 8
DEVICE = "cpu" # ganti "0" / "cuda:0" kalau GPU
N_TRIALS = 10

```

Gambar 4. 9 Source Code Klasifikasi Advanced Mixing Augmentation

Pertama, `root_dir` menentukan lokasi dataset yang berisi folder `train`, `val`, dan `test`. `train_dir`, `val_dir`, dan `test_dir` mengarah ke masing-masing folder dataset untuk pelatihan, validasi, dan pengujian. Pelatihan dilakukan selama 20 epoch maksimal, dengan `patience` diatur pada 4 epoch untuk `early stopping`, yang berarti pelatihan akan berhenti jika tidak ada peningkatan kinerja dalam 4 epoch berturut-turut. Ukuran gambar input diatur ke 224x224 piksel dan `batch size` ditetapkan pada 8 untuk setiap iterasi pelatihan, sementara proses pelatihan dilakukan pada CPU.

Model yang digunakan adalah `YOLO11n-cls.pt`, model YOLO untuk klasifikasi yang telah dipra-latih. Proses pelatihan dimulai dengan memuat model menggunakan perintah `model.train()`, di mana berbagai parameter seperti `epochs`, `patience`, `imgsz`, dan `batch size` telah ditentukan. Proses ini dijalankan pada CPU, namun jika menggunakan GPU, parameter tersebut dapat disesuaikan menjadi `"cuda"`. Pelatihan dilakukan dengan menggunakan 4 `workers` untuk mempercepat pemuatan data, dan hasil pelatihan disimpan pada folder yang ditentukan di `project`.

Terkait dengan konfigurasi optimasi, YOLO secara otomatis memilih optimizer yang paling sesuai, yaitu AdamW, dengan learning rate yang disesuaikan menjadi 0.000714 dan momentum sebesar 0.9. Parameter weight decay diatur pada 0.0005 untuk mencegah overfitting. Selain itu, early stopping diaktifkan dengan patience=4, yang memastikan pelatihan akan dihentikan secara otomatis jika tidak ada peningkatan dalam kinerja model setelah 4 epoch. N_TRIALS adalah jumlah run/percobaan pelatihan ulang yang dilakukan. N_TRIALS = 10 berarti model dilatih dan dievaluasi 10 kali untuk melihat konsistensi hasil. Tabel 4. 9 adalah parameter yang digunakan pada skenario penelitian ini.

Tabel 4. 9 Tabel Parameter Skenario Advanced Mixing Augmentation

No.	Parameter	Nilai
1.	Epoch	20
2.	Patience	4
3.	imgsz	224 × 224
4.	Batch Size	8
5.	Model	YOLO11n-cls.pt
6.	Device	CPU (opsional GPU)
7.	Workers	4
8.	Optimizer	AdamW
9.	Learning Rate	0.000714
10.	Momentum	0.9
11.	Weight Decay	0.0005
12.	N_Trials	10

Setelah pelatihan selesai, model terbaik (dengan best.pt) dievaluasi menggunakan data test. Evaluasi akhir dilakukan untuk mengukur Top-1 Accuracy dan Top-5 Accuracy, yang memberikan gambaran tentang bagaimana model dapat mengklasifikasikan gambar dengan tepat pada test set yang tidak digunakan selama pelatihan. Dengan pengaturan ini, model YOLO11 yang telah dipra-latih dilatih

kembali dengan augmentasi untuk meningkatkan kinerjanya dalam mengklasifikasikan penyakit pada daun tomat secara akurat dan efisien.

4.4.3 Photometric Augmentation

Pada eksperimen ini, berbagai konfigurasi penting diterapkan untuk melatih model YOLO11 menggunakan teknik Photometric Augmentation. Gambar 4. 10 adalah source code perintah untuk klasifikasi skenario ketiga photometric augmentation YOLO11.

```

from ultralytics import YOLO
import os
import pandas as pd

# -----
# CONFIG
# -----
ROOT_DIR = r"D:\S2\Eksperimen\Thesis\Photometric Augmentation"
PROJECT_DIR = os.path.join(ROOT_DIR, "Hasil Eksperimen")
MODEL_NAME = "yolov11-cls.pt"
RUN_PREFIX = "Photometric_trial"

EPOCHS = 20
PATIENCE = 6
IMGSZ = 224
BATCH = 8
DEVICE = "cpu" # ganti "0" / "cuda:0" kalau GPU
N_TRIALS = 10

```

Gambar 4. 10 Source Code Skenario Photometric Augmentation

Pertama, `root_dir` menentukan lokasi dataset yang berisi folder `train`, `val`, dan `test`. `train_dir`, `val_dir`, dan `test_dir` mengarah ke masing-masing folder dataset untuk pelatihan, validasi, dan pengujian. Proses pelatihan dilakukan selama 20 epoch maksimal, dengan `patience` diatur pada 6 epoch untuk `early stopping`, yang berarti pelatihan akan berhenti jika tidak ada peningkatan kinerja dalam 4 epoch berturut-turut. Ukuran gambar input diatur menjadi 224x224 piksel dan `batch size` diatur pada 8 untuk setiap iterasi pelatihan, sementara proses pelatihan dilakukan pada CPU.

Model yang digunakan adalah YOLO11n-cls.pt, sebuah model YOLO untuk klasifikasi yang sudah dipra-latih. Pelatihan dimulai dengan memuat model menggunakan perintah `model.train()`, dengan parameter seperti epochs, patience, imgsz, dan batch size telah ditentukan. Selama pelatihan, proses dijalankan pada CPU, meskipun bisa disesuaikan dengan GPU jika diperlukan. Penggunaan 4 workers ditetapkan untuk mempercepat pemuatan data, dan hasil pelatihan disimpan dalam folder yang telah ditentukan dalam project.

Untuk optimasi, YOLO secara otomatis memilih optimizer yang terbaik, yaitu AdamW, dengan learning rate yang disesuaikan menjadi 0.000714 dan momentum sebesar 0.9. Parameter weight decay diatur pada 0.0005 untuk mencegah overfitting. Selain itu, early stopping diaktifkan dengan patience=4, memastikan bahwa pelatihan berhenti otomatis jika tidak ada peningkatan selama 4 epoch berturut-turut. N_TRIALS adalah jumlah run/percobaan pelatihan ulang yang dilakukan. N_TRIALS = 10 berarti model dilatih dan dievaluasi 10 kali untuk melihat konsistensi hasil. Tabel 4. 10 adalah parameter yang digunakan pada skenario penelitian ini.

Tabel 4. 10 Tabel Parameter Skenario Photometric Augmentation

No.	Parameter	Nilai
1.	Epoch	20
2.	Patience	4
3.	imgsz	224 × 224
4.	Batch Size	8
5.	Model	YOLO11n-cls.pt
6.	Device	CPU (opsional GPU)
7.	Workers	4
8.	Optimizer	AdamW
9.	Learning Rate	0.000714
10.	Momentum	0.9
11.	Weight Decay	0.0005

12.	N_Trials	10
-----	----------	----

Setelah pelatihan selesai, model terbaik yang disimpan sebagai best.pt digunakan untuk evaluasi pada test set. Evaluasi ini mengukur Top-1 Accuracy dan Top-5 Accuracy, yang menunjukkan kemampuan model dalam mengklasifikasikan gambar dengan benar pada test set yang tidak digunakan selama pelatihan. Dengan pengaturan ini, model YOLO11 yang telah dipra-latih dilatih dengan augmentasi Photometric untuk meningkatkan akurasi dalam mengklasifikasikan penyakit pada daun tomat dengan cara yang lebih robust dan efisien.

4.4.4 Spatial Transformations

Pada eksperimen ini, konfigurasi dasar untuk pelatihan model YOLO11 menggunakan teknik Spatial Transformations telah disiapkan dengan cermat. Gambar 4. 11 adalah source code perintah untuk klasifikasi skenario keempat spatial transformations YOLO11.

```

from ultralytics import YOLO
import os
import pandas as pd

# -----
# CONFIG
# -----
ROOT_DIR = r"D:\S2\Eksperimen\Thesis\Spatial Transformations"
PROJECT_DIR = os.path.join(ROOT_DIR, "Hasil_Eksperimen")
MODEL_NAME = "yolo11n-cls.pt"
RUN_PREFIX = "Spatial_trial"

EPOCHS = 20
PATIENCE = 4
IMGSZ = 224
BATCH = 8
DEVICE = "cpu" # ganti "0" / "cuda:0" kalau GPU
N_TRIALS = 10

```

Gambar 4. 11 Source Code Skenario Spatial Transformations

Dataset terletak pada `root_dir`, yang berisi subfolder `train`, `val`, dan `test`, masing-masing untuk data pelatihan, validasi, dan pengujian. `train_dir`, `val_dir`, dan `test_dir` digunakan untuk mengarahkan ke lokasi masing-masing dataset. Eksperimen dilakukan dengan 20 epoch maksimal, di mana pelatihan akan dihentikan secara otomatis jika tidak ada peningkatan dalam 4 epoch berturut-turut (dengan pengaturan `patience=4`). Ukuran gambar input ditetapkan menjadi 224x224 piksel, dan batch size diatur pada 8 untuk setiap iterasi pelatihan, dengan CPU sebagai perangkat keras yang digunakan (meskipun bisa diganti dengan GPU jika diperlukan).

Model `YOLO11n-cls.pt`, yang merupakan model YOLO ringan untuk klasifikasi dan telah dipra-latih, digunakan untuk eksperimen ini. Model dimuat menggunakan perintah `model.train()`, yang memungkinkan pelatihan dilakukan pada dataset yang telah disiapkan. Beberapa parameter penting seperti `epochs`, `patience`, `imgsz`, dan `batch size` telah ditentukan sebelumnya. Penggunaan 4 `workers` juga diatur untuk mempercepat pemuatan data selama pelatihan.

Proses optimasi model menggunakan AdamW optimizer, yang secara otomatis dipilih oleh YOLO, dengan `learning rate` yang disesuaikan menjadi 0.000714 dan `momentum` sebesar 0.9. Selain itu, `weight decay` diatur pada 0.0005 untuk mencegah overfitting pada model. Early stopping diterapkan dengan `patience=4` agar pelatihan dihentikan jika tidak ada peningkatan dalam 4 epoch berturut-turut. `N_TRIALS` adalah jumlah run/percobaan pelatihan ulang yang dilakukan. `N_TRIALS = 10` berarti model dilatih dan dievaluasi 10 kali untuk

melihat konsistensi hasil. Tabel 4. 11 adalah parameter yang digunakan pada skenario penelitian ini.

Tabel 4. 11 Tabel Parameter Skenario Spatial Transformations

No.	Parameter	Nilai
1.	Epoch	20
2.	Patience	4
3.	imgsz	224 × 224
4.	Batch Size	8
5.	Model	YOLO11n-cls.pt
6.	Device	CPU (opsional GPU)
7.	Workers	4
8.	Optimizer	AdamW
9.	Learning Rate	0.000714
10.	Momentum	0.9
11.	Weight Decay	0.0005
12.	N_Trials	10

Setelah pelatihan selesai, model terbaik yang disimpan sebagai best.pt digunakan untuk evaluasi pada test set. Evaluasi ini mengukur Top-1 Accuracy dan Top-5 Accuracy, yang menggambarkan sejauh mana model mampu mengklasifikasikan gambar dengan benar pada test set yang tidak terlibat dalam proses pelatihan. Dengan konfigurasi yang tepat dan penggunaan teknik Spatial Transformations, eksperimen ini berhasil meningkatkan akurasi model dalam mengklasifikasikan penyakit pada daun tomat.

4.4.5 All Augmentation

Eksperimen ini melibatkan penggunaan model YOLO11n untuk klasifikasi gambar dengan penerapan All Augmentation. Gambar 4. 12 Gambar 4. 12 adalah source code perintah untuk klasifikasi skenario kelima all augmentation YOLO11.

```

from ultralytics import YOLO
import os
import pandas as pd

# -----
# CONFIG
# -----
ROOT_DIR = r"D:\SS\Experiment\thesis\All-Augmentation"
PROJECT_DIR = os.path.join(ROOT_DIR, "max11_Experiments")
MODEL_NAME = "yolo11n-cls.pt"
RUN_PREFIX = "All_trial"

EPOCHS = 20
PATIENCE = 4
IMGSZ = 224
BATCH = 8
DEVICE = "cpu" # ganti "0" / "cuda:0" kalau GPU
MEMBERS = 4
N_TRIALS = 10

```

Gambar 4. 12 Source Code Skenario All Augmentation

Proses dimulai dengan konfigurasi dasar untuk pelatihan model, yang mencakup penentuan root directory untuk dataset yang berisi train, val, dan test dataset. Model dilatih selama 20 epoch dengan early stopping yang diterapkan pada 4 epoch berturut-turut tanpa peningkatan. Ukuran gambar input diatur pada 224x224 piksel, dan batch size ditetapkan pada 8 untuk setiap iterasi pelatihan, dengan perangkat keras CPU yang digunakan (meskipun bisa diubah ke GPU jika diperlukan).

Model YOLO11n-cls.pt yang telah dipra-latih digunakan untuk klasifikasi, dan proses pelatihan dimulai menggunakan perintah `model.train()`. Beberapa parameter penting seperti `epochs`, `patience`, `imgsz`, dan `batch size` sudah diatur sebelumnya. Untuk optimasi model, digunakan AdamW optimizer yang secara otomatis dipilih oleh YOLO, dengan learning rate yang disesuaikan menjadi 0.000714 dan momentum sebesar 0.9. `N_TRIALS` adalah jumlah run/percobaan pelatihan ulang yang dilakukan. `N_TRIALS = 10` berarti model dilatih dan dievaluasi 10 kali untuk melihat konsistensi hasil. Tabel 4. 12 adalah parameter yang digunakan pada skenario penelitian ini.

Tabel 4. 12 Tabel Parameter Skenario All Augmentation

No.	Parameter	Nilai
1.	Epoch	30
2.	Patience	6
3.	imgsz	224 × 224
4.	Batch Size	8
5.	Model	YOLO11n-cls.pt
6.	Device	CPU (opsional GPU)
7.	Workers	4
8.	Optimizer	AdamW
9.	Learning Rate	0.000714
10.	Momentum	0.9
11.	Weight Decay	0.0005
12.	N Trials	10

Setelah proses pelatihan selesai, bobot model terbaik (best.pt) diambil untuk evaluasi pada test set. Hasil evaluasi menunjukkan Top-1 Accuracy dan Top-5 Accuracy yang sangat baik pada test set yang tidak terlibat dalam pelatihan. Selama pelatihan, model menunjukkan peningkatan yang signifikan, dengan Top-1 Accuracy mencapai 0.995 dan Top-5 Accuracy mencapai 1.000 pada akhir pelatihan, menunjukkan bahwa penerapan All Augmentation telah berhasil meningkatkan kinerja model dalam mengklasifikasikan gambar.

4.5 Evaluasi

Setelah model dilatih dengan data Train dan divalidasi menggunakan data Val, data Test digunakan untuk mengevaluasi seberapa baik model bekerja pada data. Berikut evaluasi dari 5 skenario penelitian ini.

4.5.1 Non Augmentation

Sebelum membahas lebih lanjut mengenai hasil evaluasi model pada eksperimen non-augmentation, mari tinjau terlebih dahulu beberapa aspek dari pelatihan yang telah dilakukan. Pada eksperimen ini, model YOLO11 dilatih tanpa

menggunakan teknik augmentasi data, yang memberikan tantangan tersendiri dalam mengoptimalkan performa model.

4.5.1.1 Hasil Evaluasi

Berdasarkan data yang diperoleh dari 10 trial dalam skenario non-augmentation, hasil yang cukup menarik dapat ditemukan. Rata-rata akurasi yang dicapai adalah 0.977, yang menunjukkan bahwa model yang diuji menunjukkan performa yang sangat baik dalam klasifikasi. Dari 10 trial yang dilakukan, akurasi tertinggi tercatat pada trial ke-3 dengan nilai 0.983, sementara nilai terendah tercatat pada trial ke-10 dengan akurasi 0.969. Meskipun ada sedikit variasi antara trial, semua nilai akurasi tetap berada di atas 0.96, yang menunjukkan bahwa model konsisten dalam mengklasifikasikan data dengan baik. Pencapaian ini menggambarkan stabilitas model yang baik dalam skenario tanpa augmentasi data, meskipun ada kemungkinan bahwa penambahan augmentasi data dapat memberikan sedikit peningkatan dalam akurasi lebih lanjut. Tabel 4. 13 Tabel Hasil 10 Trial Skenario Non Augmentation.

Tabel 4. 13 Tabel Hasil 10 Trial Skenario Non Augmentation

Trial	Train Loss	Val Loss	Accuracy
1	0.07020	0.11802	0.971
2	0.02294	0.08110	0.978
3	0.10003	0.05451	0.983
4	0.00788	0.07001	0.981
5	0.16908	0.12778	0.971
6	0.04183	0.09728	0.974
7	0.00892	0.04787	0.983
8	0.02591	0.06969	0.977
9	0.00472	0.05804	0.983
10	0.08112	0.09552	0.969
Rata - Rata			0.977

Trial terbaik berdasarkan akurasi adalah trial ke-3, trial ke-7, dan trial ke-9, yang masing-masing mencapai akurasi 0.983. Namun, trial ke-9 memiliki train loss yang lebih rendah (0.00472) dibandingkan dengan trial ke-3 (0.10003) dan trial ke-7 (0.00892), yang menunjukkan bahwa model pada trial ke-9 lebih stabil dalam hal pembelajaran.

Hasil evaluasi berikut ini memberikan gambaran tentang bagaimana model tersebut berperilaku selama pelatihan pada setiap epoch, termasuk metrik seperti akurasi, loss, serta waktu yang dibutuhkan untuk setiap epoch. Berikut adalah tabel hasil evaluasi di trial ke 9.

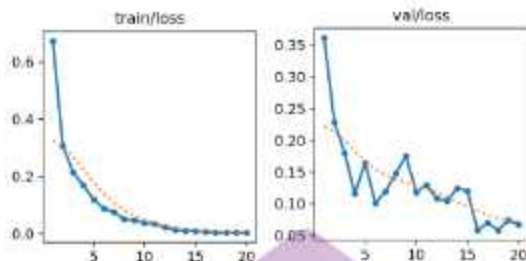
Tabel 4. 14 Tabel Hasil Evaluasi Non Augmentation

Epoch	time	train/loss	accuracy top1	accuracy top5	val/loss
1	509.788	0.67407	0.889	0.996	0.36147
2	1001.79	0.30504	0.929	0.998	0.22837
3	1490.53	0.21444	0.938	0.997	0.17978
4	1988.2	0.16972	0.961	0.998	0.11603
5	2476.97	0.12002	0.944	0.999	0.16482
6	2968.54	0.08851	0.963	0.998	0.1009
7	3476.28	0.07613	0.966	1	0.1195
8	3969.33	0.05118	0.956	0.998	0.14805
9	4470.01	0.04711	0.949	0.999	0.17527
10	4967.07	0.03811	0.969	0.999	0.11818
11	5473.88	0.03431	0.966	0.997	0.12948
12	5968.95	0.02305	0.973	0.999	0.10892
13	6463.15	0.01308	0.981	0.999	0.10438
14	6968.24	0.01092	0.969	0.999	0.12493
15	7478.55	0.00915	0.97	0.998	0.12037
16	7983.36	0.00732	0.982	0.999	0.05831
17	8486.65	0.00447	0.979	0.998	0.07032
18	8998.14	0.00472	0.983	1	0.05804
19	9502.16	0.00405	0.977	0.999	0.07422
20	10004.3	0.0037	0.982	1	0.06701

Pada trial ini, accuracy top-1 menunjukkan peningkatan yang konsisten seiring berjalannya waktu, dimulai dari 0.889 pada epoch pertama dan mencapai 0.982 pada epoch ke-20. Hal ini menunjukkan bahwa model semakin dapat mengklasifikasikan data dengan akurasi yang lebih tinggi pada top-1. Accuracy top-5 juga tetap sangat baik, hampir mendekati 1 di setiap epoch, menunjukkan bahwa meskipun ada variasi kecil, model masih dapat memberikan prediksi yang tepat dalam 5 teratas.

Dari sisi train loss, terdapat penurunan signifikan sejak epoch pertama yang bernilai 0.67407, menuju nilai yang sangat rendah yaitu 0.00370 pada epoch ke-20. Hal ini mengindikasikan bahwa model semakin baik dalam meminimalkan kesalahan prediksi selama pelatihan. Namun, meskipun terjadi penurunan yang signifikan, terdapat fluktuasi kecil pada beberapa epoch yang disebabkan oleh penurunan laju pembelajaran dan adaptasi model terhadap data.

Val loss menunjukkan penurunan yang stabil dari epoch pertama, di mana nilainya 0.36147, menuju nilai terendah yaitu 0.05804 pada epoch ke-18, meskipun sedikit meningkat pada epoch ke-19 dan 20. Penurunan val loss yang konsisten ini menunjukkan bahwa model tidak hanya belajar dengan baik dari data pelatihan, tetapi juga mampu menggeneralisasi data validasi dengan baik. Meskipun ada sedikit kenaikan pada val loss pada epoch akhir, nilai tersebut tetap cukup rendah, yang menunjukkan bahwa model masih dalam kondisi yang baik untuk diterapkan pada data baru. Gambar 4. 13 Grafik Train dan Val Loss trial ke 9 Non Augmentation



Gambar 4. 13 Grafik Train dan Val Loss trial ke 9 Non Augmentation

Secara keseluruhan, model ini menunjukkan performa yang sangat baik dengan tingkat akurasi yang terus meningkat, serta penurunan train loss dan val loss yang stabil, yang menandakan proses pelatihan yang sukses dan kemampuan generalisasi yang baik pada data yang tidak terlihat sebelumnya.

4.5.1.2 Hasil Precision, Recall, F1-Score

Berikut adalah hasil evaluasi model pada skenario non-augmentation dengan menampilkan Precision, Recall, dan F1-Score untuk setiap kelas yang terklasifikasi oleh model.

Tabel 4. 15 Tabel Metriks Precision, Recall, F1-Score per kelas

Class	Precision	Recall	F1-score	Support
Bacterial spot	1.000	0.995	0.997	200
Early blight	0.990	0.960	0.975	200
Healthy	1.000	1.000	1.000	200
Late blight	0.961	0.985	0.973	200
Leaf Mold	0.995	0.980	0.987	200
Mosaic virus	0.990	0.995	0.993	200
Septoria leaf spot	0.990	0.985	0.987	200
Spider mites	0.980	0.975	0.977	200
Target Spot	0.957	0.990	0.973	200
Yellow Leaf Curl Virus	1.000	0.995	0.997	200
Rata - Rata	0.986	0.986	0.986	200

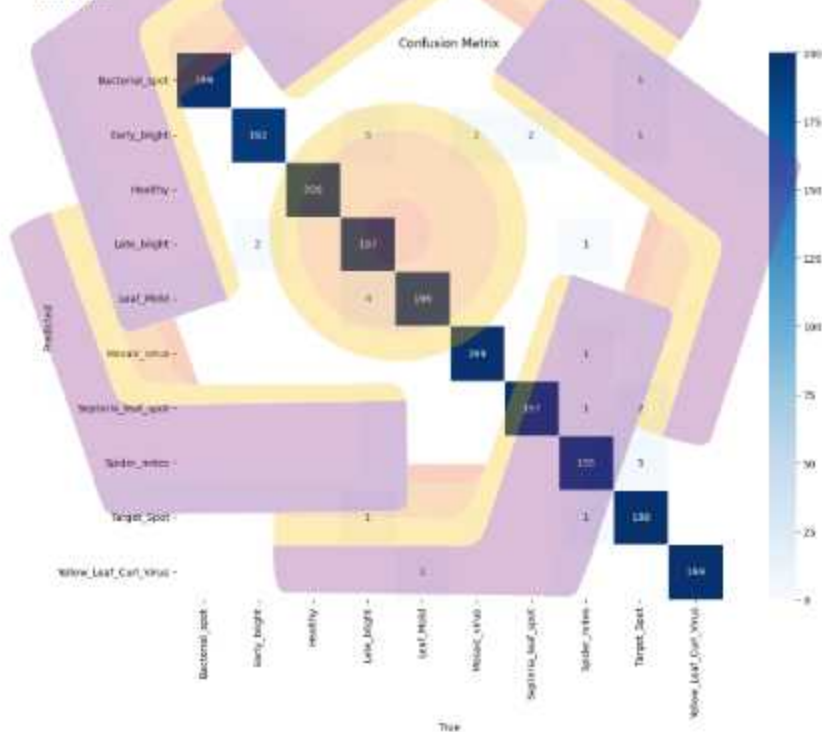
Tabel di atas menunjukkan hasil evaluasi model klasifikasi dengan metrik precision, recall, dan F1-score untuk berbagai kelas penyakit tanaman, dengan support yang sama yaitu 200 untuk setiap kelas. Healthy mencapai hasil sempurna dengan nilai precision dan recall masing-masing 1.000, yang menunjukkan bahwa model dapat dengan sangat akurat mengklasifikasikan kelas ini. Kelas Bacterial_spot dan Yellow_Leaf_Curl_Virus juga memiliki performa yang sangat baik, dengan nilai precision, recall, dan F1-score yang hampir sempurna (1.000 untuk precision dan sangat mendekati 1.000 untuk recall dan F1-score).

Di sisi lain, kelas Early_blight memiliki precision yang sedikit lebih rendah (0.990) dan recall yang lebih rendah lagi (0.960), yang menunjukkan bahwa meskipun model dapat mengenali sebagian besar contoh dengan akurat, ada sedikit kesalahan dalam klasifikasi kelas ini. Target_Spot mencatatkan precision yang lebih rendah (0.957), namun recall yang lebih tinggi (0.990), menandakan bahwa meskipun banyak data yang terklasifikasi, model lebih sering menghasilkan kesalahan dalam identifikasi kelas ini.

Secara keseluruhan, nilai rata-rata untuk precision, recall, dan F1-score masing-masing adalah 0.986, yang menunjukkan bahwa model memiliki performa yang sangat baik dalam klasifikasi seluruh kelas, dengan keseimbangan yang baik antara mengidentifikasi contoh positif dengan benar dan meminimalkan kesalahan. Rata-rata yang konsisten ini menunjukkan bahwa model bekerja dengan efisien pada berbagai kelas tanpa kecenderungan bias terhadap kelas tertentu.

4.5.1.3 Confusion Matrix

Confusion matrix yang ditampilkan pada gambar ini menunjukkan kinerja model dalam mengklasifikasikan penyakit daun dengan menggunakan data pengujian. Setiap baris pada matriks ini mewakili kelas sebenarnya (true class), sedangkan setiap kolom mewakili kelas prediksi model (predicted class). Nilai yang ada dalam setiap sel matriks ini menggambarkan jumlah prediksi benar (diagonal) dan kesalahan prediksi (sel di luar diagonal). Gambar 4. 14 adalah confusion matriks.



Gambar 4. 14 Confusion Matriks Non Augmentation

Confusion matrix menunjukkan gambaran yang jelas mengenai kinerja model dalam mengklasifikasikan berbagai jenis penyakit tanaman. Secara keseluruhan, model menunjukkan kinerja yang sangat baik dalam mengidentifikasi kelas-kelas yang ada. *Bacterial_spot* berhasil diklasifikasikan dengan sangat baik, dengan hanya satu kasus yang salah dikategorikan sebagai *Healthy*. *Early_blight* juga memiliki hasil yang baik dengan 192 prediksi yang benar, meskipun ada sedikit kebingungannya dengan kelas *Bacterial_spot* dan *Late_blight*. Kelas *Healthy* diprediksi dengan sempurna, tanpa ada kesalahan klasifikasi, yang menunjukkan model mampu mengenali tanaman yang sehat dengan sangat akurat.

Kelas *Late_blight* memiliki 197 prediksi yang benar, meskipun ada beberapa kesalahan minor, seperti satu kasus yang salah dikategorikan sebagai *Bacterial_spot* dan *Yellow_Leaf_Curl_Virus*. *Leaf_Mold* juga menunjukkan performa yang baik dengan 196 prediksi benar, namun ada sedikit kebingungannya dengan *Bacterial_spot*. *Mosaic_virus* hanya memiliki satu kesalahan klasifikasi yang sangat kecil, yaitu satu kasus yang salah diprediksi sebagai *Target_Spot*.

Kelas *Septoria_leaf_spot* berhasil diklasifikasikan dengan baik, namun terdapat beberapa kesalahan dengan *Mosaic_virus* dan *Spider_mites*. Untuk *Spider_mites*, meskipun ada beberapa prediksi yang salah sebagai *Target_Spot*, model ini tetap memiliki akurasi yang cukup tinggi dengan 195 prediksi benar. *Target_Spot* juga menunjukkan performa yang baik dengan hanya satu kesalahan klasifikasi. Terakhir, *Yellow_Leaf_Curl_Virus* berhasil diklasifikasikan dengan sangat baik, dengan hanya satu kesalahan yang salah dikategorikan sebagai *Bacterial_spot*.

Secara keseluruhan, confusion matrix ini menunjukkan bahwa model bekerja dengan sangat baik, meskipun ada sedikit kesalahan yang terjadi antar kelas yang serupa. Meskipun demikian, model ini cukup handal dan memiliki akurasi yang sangat tinggi dalam mengidentifikasi penyakit tanaman yang berbeda.

4.5.1.4 Analisis Confidence Score

Berdasarkan nilai confidence (yaitu probabilitas maksimum yang dipilih model pada setiap sampel), terlihat pemisahan yang jelas antara prediksi yang tepat dan prediksi yang keliru. Gambar 4. 15 adalah hasil confidence score tiap gambar skenario non augmentation.

Sample	Actual Class	Predicted Class	Confidence Score
101_Healthy_plant_01.jpg	Healthy_plant	Healthy_plant	0.996
102_Healthy_plant_02.jpg	Healthy_plant	Healthy_plant	0.996
103_Healthy_plant_03.jpg	Healthy_plant	Healthy_plant	0.996
104_Healthy_plant_04.jpg	Healthy_plant	Healthy_plant	0.996
105_Healthy_plant_05.jpg	Healthy_plant	Healthy_plant	0.996
106_Healthy_plant_06.jpg	Healthy_plant	Healthy_plant	0.996
107_Healthy_plant_07.jpg	Healthy_plant	Healthy_plant	0.996
108_Healthy_plant_08.jpg	Healthy_plant	Healthy_plant	0.996
109_Healthy_plant_09.jpg	Healthy_plant	Healthy_plant	0.996
110_Healthy_plant_10.jpg	Healthy_plant	Healthy_plant	0.996
111_Healthy_plant_11.jpg	Healthy_plant	Healthy_plant	0.996
112_Healthy_plant_12.jpg	Healthy_plant	Healthy_plant	0.996
113_Healthy_plant_13.jpg	Healthy_plant	Healthy_plant	0.996
114_Healthy_plant_14.jpg	Healthy_plant	Healthy_plant	0.996
115_Healthy_plant_15.jpg	Healthy_plant	Healthy_plant	0.996
116_Healthy_plant_16.jpg	Healthy_plant	Healthy_plant	0.996
117_Healthy_plant_17.jpg	Healthy_plant	Healthy_plant	0.996
118_Healthy_plant_18.jpg	Healthy_plant	Healthy_plant	0.996
119_Healthy_plant_19.jpg	Healthy_plant	Healthy_plant	0.996
120_Healthy_plant_20.jpg	Healthy_plant	Healthy_plant	0.996
121_Healthy_plant_21.jpg	Healthy_plant	Healthy_plant	0.996
122_Healthy_plant_22.jpg	Healthy_plant	Healthy_plant	0.996
123_Healthy_plant_23.jpg	Healthy_plant	Healthy_plant	0.996
124_Healthy_plant_24.jpg	Healthy_plant	Healthy_plant	0.996
125_Healthy_plant_25.jpg	Healthy_plant	Healthy_plant	0.996
126_Healthy_plant_26.jpg	Healthy_plant	Healthy_plant	0.996
127_Healthy_plant_27.jpg	Healthy_plant	Healthy_plant	0.996
128_Healthy_plant_28.jpg	Healthy_plant	Healthy_plant	0.996
129_Healthy_plant_29.jpg	Healthy_plant	Healthy_plant	0.996
130_Healthy_plant_30.jpg	Healthy_plant	Healthy_plant	0.996
131_Healthy_plant_31.jpg	Healthy_plant	Healthy_plant	0.996
132_Healthy_plant_32.jpg	Healthy_plant	Healthy_plant	0.996
133_Healthy_plant_33.jpg	Healthy_plant	Healthy_plant	0.996
134_Healthy_plant_34.jpg	Healthy_plant	Healthy_plant	0.996
135_Healthy_plant_35.jpg	Healthy_plant	Healthy_plant	0.996
136_Healthy_plant_36.jpg	Healthy_plant	Healthy_plant	0.996
137_Healthy_plant_37.jpg	Healthy_plant	Healthy_plant	0.996
138_Healthy_plant_38.jpg	Healthy_plant	Healthy_plant	0.996
139_Healthy_plant_39.jpg	Healthy_plant	Healthy_plant	0.996
140_Healthy_plant_40.jpg	Healthy_plant	Healthy_plant	0.996
141_Healthy_plant_41.jpg	Healthy_plant	Healthy_plant	0.996
142_Healthy_plant_42.jpg	Healthy_plant	Healthy_plant	0.996
143_Healthy_plant_43.jpg	Healthy_plant	Healthy_plant	0.996
144_Healthy_plant_44.jpg	Healthy_plant	Healthy_plant	0.996
145_Healthy_plant_45.jpg	Healthy_plant	Healthy_plant	0.996
146_Healthy_plant_46.jpg	Healthy_plant	Healthy_plant	0.996
147_Healthy_plant_47.jpg	Healthy_plant	Healthy_plant	0.996
148_Healthy_plant_48.jpg	Healthy_plant	Healthy_plant	0.996
149_Healthy_plant_49.jpg	Healthy_plant	Healthy_plant	0.996
150_Healthy_plant_50.jpg	Healthy_plant	Healthy_plant	0.996
151_Healthy_plant_51.jpg	Healthy_plant	Healthy_plant	0.996
152_Healthy_plant_52.jpg	Healthy_plant	Healthy_plant	0.996
153_Healthy_plant_53.jpg	Healthy_plant	Healthy_plant	0.996
154_Healthy_plant_54.jpg	Healthy_plant	Healthy_plant	0.996
155_Healthy_plant_55.jpg	Healthy_plant	Healthy_plant	0.996
156_Healthy_plant_56.jpg	Healthy_plant	Healthy_plant	0.996
157_Healthy_plant_57.jpg	Healthy_plant	Healthy_plant	0.996
158_Healthy_plant_58.jpg	Healthy_plant	Healthy_plant	0.996
159_Healthy_plant_59.jpg	Healthy_plant	Healthy_plant	0.996
160_Healthy_plant_60.jpg	Healthy_plant	Healthy_plant	0.996
161_Healthy_plant_61.jpg	Healthy_plant	Healthy_plant	0.996
162_Healthy_plant_62.jpg	Healthy_plant	Healthy_plant	0.996
163_Healthy_plant_63.jpg	Healthy_plant	Healthy_plant	0.996
164_Healthy_plant_64.jpg	Healthy_plant	Healthy_plant	0.996
165_Healthy_plant_65.jpg	Healthy_plant	Healthy_plant	0.996
166_Healthy_plant_66.jpg	Healthy_plant	Healthy_plant	0.996
167_Healthy_plant_67.jpg	Healthy_plant	Healthy_plant	0.996
168_Healthy_plant_68.jpg	Healthy_plant	Healthy_plant	0.996
169_Healthy_plant_69.jpg	Healthy_plant	Healthy_plant	0.996
170_Healthy_plant_70.jpg	Healthy_plant	Healthy_plant	0.996
171_Healthy_plant_71.jpg	Healthy_plant	Healthy_plant	0.996
172_Healthy_plant_72.jpg	Healthy_plant	Healthy_plant	0.996
173_Healthy_plant_73.jpg	Healthy_plant	Healthy_plant	0.996
174_Healthy_plant_74.jpg	Healthy_plant	Healthy_plant	0.996
175_Healthy_plant_75.jpg	Healthy_plant	Healthy_plant	0.996
176_Healthy_plant_76.jpg	Healthy_plant	Healthy_plant	0.996
177_Healthy_plant_77.jpg	Healthy_plant	Healthy_plant	0.996
178_Healthy_plant_78.jpg	Healthy_plant	Healthy_plant	0.996
179_Healthy_plant_79.jpg	Healthy_plant	Healthy_plant	0.996
180_Healthy_plant_80.jpg	Healthy_plant	Healthy_plant	0.996
181_Healthy_plant_81.jpg	Healthy_plant	Healthy_plant	0.996
182_Healthy_plant_82.jpg	Healthy_plant	Healthy_plant	0.996
183_Healthy_plant_83.jpg	Healthy_plant	Healthy_plant	0.996
184_Healthy_plant_84.jpg	Healthy_plant	Healthy_plant	0.996
185_Healthy_plant_85.jpg	Healthy_plant	Healthy_plant	0.996
186_Healthy_plant_86.jpg	Healthy_plant	Healthy_plant	0.996
187_Healthy_plant_87.jpg	Healthy_plant	Healthy_plant	0.996
188_Healthy_plant_88.jpg	Healthy_plant	Healthy_plant	0.996
189_Healthy_plant_89.jpg	Healthy_plant	Healthy_plant	0.996
190_Healthy_plant_90.jpg	Healthy_plant	Healthy_plant	0.996
191_Healthy_plant_91.jpg	Healthy_plant	Healthy_plant	0.996
192_Healthy_plant_92.jpg	Healthy_plant	Healthy_plant	0.996
193_Healthy_plant_93.jpg	Healthy_plant	Healthy_plant	0.996
194_Healthy_plant_94.jpg	Healthy_plant	Healthy_plant	0.996
195_Healthy_plant_95.jpg	Healthy_plant	Healthy_plant	0.996
196_Healthy_plant_96.jpg	Healthy_plant	Healthy_plant	0.996
197_Healthy_plant_97.jpg	Healthy_plant	Healthy_plant	0.996
198_Healthy_plant_98.jpg	Healthy_plant	Healthy_plant	0.996
199_Healthy_plant_99.jpg	Healthy_plant	Healthy_plant	0.996
200_Healthy_plant_100.jpg	Healthy_plant	Healthy_plant	0.996

Gambar 4. 15 Hasil Confidence Gambar

Untuk prediksi yang tepat, confidence cenderung sangat tinggi dan stabil, dengan rata-rata sekitar 0,996 serta median 1,000, sehingga model umumnya memberikan keputusan dengan keyakinan kuat ketika hasilnya sesuai label. Bahkan pada bagian confidence terendah sekalipun untuk prediksi tepat, nilainya masih berada di kisaran $\approx 0,999$ (sekitar persentil 10), yang menunjukkan konsistensi tingkat keyakinan pada keputusan yang benar.

Sebaliknya, pada prediksi yang keliru, confidence secara umum lebih rendah dan lebih bervariasi, dengan rata-rata sekitar 0,734 dan median sekitar

0,762. Pola ini mengindikasikan bahwa ketika model melakukan kesalahan, model sering kali berada pada kondisi “ragu” (confidence tidak setinggi prediksi tepat). Namun demikian, terdapat sebagian kecil kasus keliru dengan confidence yang sangat tinggi (hingga sekitar $\approx 0,981$ pada persentil 90, bahkan beberapa contoh mendekati 0,998), yang menandakan adanya kondisi tertentu di mana model dapat salah namun tetap sangat yakin, biasanya berkaitan dengan kemiripan pola visual antarkelas atau adanya fitur dominan yang membuat model “terkunci” pada satu kelas. Temuan ini penting untuk praktik penggunaan sistem: confidence dapat dimanfaatkan sebagai indikator kebutuhan verifikasi, misalnya dengan menetapkan ambang confidence untuk menandai prediksi yang perlu pemeriksaan ulang, serta mempertimbangkan kalibrasi confidence agar nilai probabilitas lebih merefleksikan tingkat ketidakpastian model secara proporsional.

Dalam visualisasi hasil prediksi dengan model YOLO11 pada skenario non-augmentation, kita dapat melihat contoh gambar yang diprediksi dengan benar dan salah. Gambar 4. 16 Gambar Visualisasi Label Prediksi Non Augmentation.



Gambar 4. 16 Gambar Visualisasi Label Prediksi Non Augmentation

Model YOLO11 melakukan kesalahan dalam mengklasifikasikan penyakit daun tomat pada sembilan gambar yang salah. Misalnya, model salah mengklasifikasikan Bacterial Spot sebagai Target Spot karena perbedaan antara bercak basah dan bercak kering yang tidak cukup dikenali. Selain itu, Early Blight

sering tercampur dengan Late Blight, meskipun keduanya memiliki perbedaan pada ukuran dan batas bercak yang jelas. Leaf Mold dan Late Blight juga sulit dibedakan, meskipun Leaf Mold memiliki bercak yang lebih halus akibat infeksi jamur, sementara Late Blight menunjukkan bercak lebih gelap dengan tepi yang tajam. Pada kasus Septoria Leaf Spot, model keliru mengklasifikasikan sebagai Spider Mites, padahal keduanya memiliki pola kerusakan yang berbeda: bercak kecil pada Septoria dan kerusakan akibat penghisapan cairan oleh Spider Mites. Begitu juga dengan Yellow Leaf Curl Virus, yang keliru diklasifikasikan sebagai Leaf Mold atau Septoria Leaf Spot, meskipun gejala utamanya adalah daun yang melengkung dengan bercak kuning, berbeda dengan pola bercak jamur atau bercak bulat pada Septoria.

Kesalahan prediksi ini menunjukkan bahwa model kesulitan membedakan penyakit dengan gejala yang mirip, terutama ketika bercak pada daun terlihat serupa. Untuk meningkatkan akurasi, model perlu dilatih dengan dataset yang lebih variatif dan beragam, serta meningkatkan sensitivitas terhadap perbedaan halus dalam pola bercak atau perubahan warna pada daun.

4.5.2 Advanced Mixing Augmentation

Sebelum kita membahas lebih lanjut mengenai hasil evaluasi model pada advanced mixing augmentation, mari kita tinjau terlebih dahulu beberapa aspek dari pelatihan yang telah dilakukan. Pada eksperimen ini, model YOLO11 dilatih menggunakan teknik advanced mixing augmentation.

4.5.2.1 Hasil Evaluasi

Skenario Advanced Mixing Augmentation trial terbaik ke-6 dari total 10 trial menunjukkan hasil yang sangat baik dalam hal akurasi model. Trial ini mencerminkan kemajuan yang signifikan dalam penggunaan teknik augmentasi lanjutan untuk meningkatkan kemampuan generalisasi model terhadap berbagai kondisi yang lebih kompleks. Tabel 4. 16 Semua Trial Skenario Advanced Mixing Augmentation.

Tabel 4. 16 Semua Trial Skenario Advanced Mixing Augmentation

Trial	Train Loss	Val Loss	Accuracy
1	0.06619	0.07726	0.9785
2	0.33062	0.10772	0.9635
3	0.34648	0.08848	0.9730
4	0.00746	0.04702	0.9860
5	0.00741	0.06476	0.9870
6	0.00871	0.06409	0.9910
7	0.04170	0.06231	0.9800
8	0.00911	0.04695	0.9845
9	0.01018	0.04727	0.9800
10	0.05522	0.05137	0.9780
Rata - Rata			0.980150

Hasil evaluasi berikut ini memberikan gambaran tentang bagaimana model tersebut berperforma selama pelatihan pada setiap epoch, termasuk metrik seperti akurasi, loss, serta waktu yang dibutuhkan untuk setiap epoch. Berikut adalah Tabel 4. 17 Tabel Hasil Evaluasi Advanced Mixing Augmentation yang berisi hasil evaluasi advanced mixing augmentation trial ke 6.

Tabel 4. 17 Tabel Hasil Evaluasi Advanced Mixing Augmentation

Epoch	time	train/loss	accuracy_top1	accuracy_top5	val/loss
1	722.258	0.83468	0.845	0.999	0.4669
2	1417.73	0.45856	0.946	0.999	0.15475
3	2107.49	0.35125	0.949	0.997	0.1383
4	2791.49	0.26282	0.961	1	0.12045

5	3480.05	0.18807	0.968	1	0.09713
6	4171.93	0.15096	0.954	1	0.11237
7	4866.11	0.12692	0.97	1	0.0856
8	5569.05	0.09823	0.97	0.999	0.10387
9	6266.38	0.06687	0.974	1	0.08638
10	6965.74	0.05746	0.975	1	0.06759
11	7668.22	0.04201	0.981	1	0.05736
12	8369.45	0.03038	0.981	1	0.06049
13	9071.3	0.02254	0.982	1	0.07288
14	9767.82	0.01378	0.983	1	0.06196
15	10461.6	0.01422	0.98	1	0.08395
16	11158.4	0.00871	0.984	1	0.06409
17	11859.5	0.00757	0.983	1	0.05594
18	12563.1	0.00735	0.982	1	0.06298
19	13268.3	0.00495	0.982	1	0.05915
20	13968.8	0.00311	0.984	1	0.05681

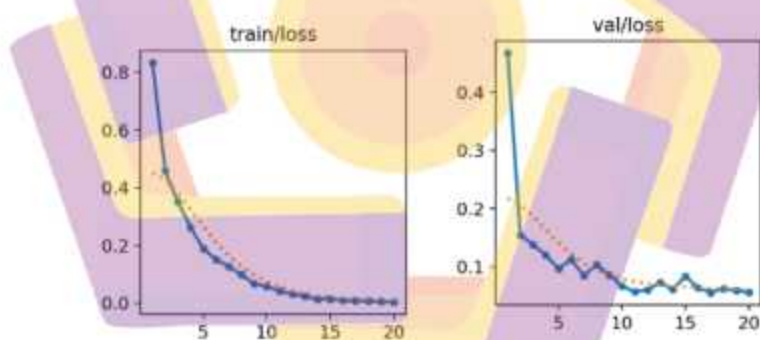
Pada epoch pertama, *train loss* sangat tinggi di angka 0.83468, sementara akurasi *top-1* masih rendah di 84.5%, menunjukkan bahwa model belum sepenuhnya belajar dari data pelatihan. Namun, seiring berjalannya epoch, terjadi penurunan yang signifikan pada *train loss* di setiap epoch berikutnya. Di epoch kedua, *train loss* turun menjadi 0.45856, dan akurasi *top-1* meningkat menjadi 94.6%. Pada epoch keempat, *train loss* mencapai 0.26282 dengan akurasi *top-1* 96.1%, menunjukkan perbaikan yang jelas dalam pelatihan model.

Selanjutnya, *train loss* terus menurun dengan laju yang stabil, mencapai 0.00311 di epoch ke-20, sementara akurasi *top-1* terus meningkat hingga mencapai 98.4%. Penurunan *train loss* yang konsisten ini menunjukkan bahwa model semakin baik dalam meminimalkan kesalahan selama pelatihan, dan kemampuannya untuk memprediksi dengan benar semakin membaik.

Pada sisi *validation loss*, meskipun *train loss* terus menurun, *validation loss* menunjukkan penurunan yang signifikan sejak awal. Di epoch pertama, *validation*

loss berada pada 0.4669, namun terus menurun stabil sepanjang pelatihan, mencapai 0.05681 di epoch terakhir. Penurunan yang konsisten ini menunjukkan bahwa model semakin baik dalam menggeneralisasi ke data yang tidak terlihat selama pelatihan.

Secara keseluruhan, data menunjukkan bahwa model mengalami peningkatan yang sangat baik. *Train loss* dan *validation loss* terus menurun secara signifikan, sementara akurasi *top-1* mengalami peningkatan stabil dari 84.5% di epoch pertama menjadi 98.4% di epoch ke-20. Hal ini menunjukkan bahwa model berhasil belajar dengan baik dari data pelatihan dan juga menunjukkan kemampuan yang baik dalam menggeneralisasi ke data yang tidak terlihat. Gambar 4. 17 adalah grafik *train/loss* dan *val/loss*.



Gambar 4. 17 Train/loss dan Val/loss skenario Advanced Mixing Augmentation

4.5.2.2 Hasil Precision, Recall, F1-Score

Berikut adalah hasil evaluasi model pada skenario advanced mixing augmentation dengan menampilkan Precision, Recall, dan F1-Score untuk setiap kelas yang terklasifikasi oleh model.

Tabel 4. 18 Tabel Hasil Precision, Recall, F1-Score

Penyakit	Precision	Recall	F1-Score	Support
Bacterial spot	0.995	0.995	0.995	200
Early blight	0.980	0.975	0.977	200
Healthy	0.995	1.000	0.998	200
Late blight	0.995	0.980	0.987	200
Leaf Mold	0.995	0.995	0.995	200
Mosaic virus	0.995	1.000	0.998	200
Septoria leaf spot	0.995	1.000	0.998	200
Spider mites	0.995	0.980	0.987	200
Target Spot	0.971	0.990	0.980	200
Yellow Leaf Curl Virus	0.995	0.995	0.995	200
Rata - rata	0.991	0.991	0.991	200

Berdasarkan tabel diatas, model menunjukkan performa yang sangat baik dalam mengklasifikasikan berbagai penyakit pada daun tomat, dengan nilai precision, recall, dan F1-score yang konsisten tinggi. Penyakit seperti Bacterial_spot, Healthy, Leaf_Mold, Mosaic_virus, Septoria_leaf_spot, dan Yellow_Leaf_Curl_Virus menunjukkan precision dan recall yang sangat tinggi, masing-masing mencapai 0.995 dan recall 1.000, dengan F1-score mendekati 0.995. Hal ini menandakan bahwa model sangat akurat dalam mengklasifikasi penyakit-penyakit ini tanpa banyak kesalahan.

Pada Early_blight dan Spider_mites, precision sedikit lebih rendah (0.980 dan 0.995), dengan recall masing-masing 0.975 dan 0.980. F1-score untuk keduanya tetap tinggi, 0.977 dan 0.987, yang menunjukkan bahwa meskipun ada penurunan kecil dalam akurasi klasifikasi, model tetap mempertahankan performa yang solid.

Penyakit Target_Spot menunjukkan precision yang lebih rendah (0.971), meskipun recall tinggi (0.990). Hal ini mengindikasikan bahwa meskipun model dapat mengklasifikasi sebagian besar kasus Target_Spot, masih ada beberapa

kesalahan klasifikasi yang menyebabkan nilai precision sedikit lebih rendah. F1-score untuk penyakit ini tercatat 0.980, yang tetap menunjukkan kinerja yang baik meskipun ada penurunan kecil.

Secara keseluruhan, model menunjukkan kemampuan yang luar biasa dalam mengklasifikasikan penyakit daun tomat dengan precision dan recall yang sangat baik pada sebagian besar penyakit, meskipun ada sedikit penurunan dalam Target_Spot yang perlu diperbaiki.

4.5.2.3 Confusion Matrix

Confusion matrix yang ditampilkan pada gambar ini menunjukkan kinerja model dalam mengklasifikasikan penyakit daun dengan menggunakan data pengujian. Berikut adalah confusion matrixnya.





Gambar 4. 18 Confusion Matrix Advanced Mixing Augmentation

Berdasarkan confusion matrix yang disediakan, model menunjukkan performa yang sangat baik dalam mengklasifikasikan penyakit daun tomat dengan sebagian besar prediksi yang benar. Untuk penyakit Bacterial Spot, hampir semua sampel terklasifikasi dengan benar (199 benar), hanya terdapat 1 kesalahan yang mengklasifikasikan sampel sebagai Early Blight dan 1 lagi sebagai Healthy, menunjukkan bahwa model sangat akurat dalam mengklasifikasi penyakit ini. Pada Early Blight, sebagian besar sampel juga terklasifikasi dengan benar (195 benar), meskipun terdapat kesalahan kecil, yakni 1 sampel yang salah klasifikasi sebagai Bacterial Spot, 1 sebagai Mosaic Virus, dan 2 sebagai Yellow Leaf Curl Virus. Hal

ini menunjukkan sedikit kebingungan antara Early Blight dan beberapa penyakit lainnya.

Model sangat baik dalam mengidentifikasi daun sehat (Healthy), dengan semua 200 sampel terklasifikasikan dengan benar, menandakan bahwa model dapat dengan mudah membedakan daun yang sehat. Untuk Late Blight, model mengklasifikasikan 196 sampel dengan benar, namun terdapat 4 kesalahan klasifikasi, di antaranya sebagai Bacterial Spot dan Target Spot, yang menunjukkan bahwa ada sedikit kebingungan antara Late Blight dan Target Spot. Penyakit Leaf Mold juga terklasifikasi dengan sangat baik, dengan hanya 1 kesalahan sebagai Mosaic Virus, sedangkan Mosaic Virus sendiri terklasifikasi dengan benar pada semua 200 sampel.

Pada Septoria Leaf Spot, model berhasil mengklasifikasikan semua sampel dengan benar, menunjukkan keakuratan tinggi pada penyakit ini. Namun, pada Spider Mites, meskipun sebagian besar sampel terklasifikasikan dengan benar (196 benar), ada 3 kesalahan, dengan 1 sampel salah klasifikasi sebagai Early Blight dan 2 sebagai Target Spot, menunjukkan sedikit kebingungan antara Spider Mites dan penyakit lain. Untuk Target Spot, meskipun 198 sampel terklasifikasi dengan benar, terdapat 1 kesalahan sebagai Spider Mites dan 1 lagi sebagai Late Blight. Terakhir, semua sampel Yellow Leaf Curl Virus terklasifikasikan dengan benar, menunjukkan bahwa model sangat baik dalam mengklasifikasi penyakit ini.

Secara keseluruhan, model menunjukkan kinerja yang sangat baik dengan kebanyakan penyakit terklasifikasi dengan benar. Meskipun ada beberapa kesalahan klasifikasi, terutama antara penyakit dengan gejala yang mirip seperti

Early Blight dengan Mosaic Virus, atau Spider Mites dengan Target Spot, kesalahan ini relatif sedikit. Model ini secara keseluruhan sangat efisien dalam mengenali berbagai penyakit daun tomat, meskipun ada sedikit ruang untuk perbaikan dalam membedakan penyakit yang gejalanya sangat mirip.

4.5.2.4 Analisis Confidence Score

Berdasarkan nilai confidence (yakni probabilitas maksimum/top-1 probability yang dipilih model pada setiap sampel) pada file probs_allclasses_trial8.csv, terlihat pemisahan yang jelas antara prediksi yang tepat dan prediksi yang keliru dari sisi tingkat keyakinan model. Gambar 4. 19 adalah hasil confidence score skenario ini.

id	prediksi	confidence	target
1	0	0.9999	0
2	0	0.9999	0
3	0	0.9999	0
4	0	0.9999	0
5	0	0.9999	0
6	0	0.9999	0
7	0	0.9999	0
8	0	0.9999	0
9	0	0.9999	0
10	0	0.9999	0
11	0	0.9999	0
12	0	0.9999	0
13	0	0.9999	0
14	0	0.9999	0
15	0	0.9999	0
16	0	0.9999	0
17	0	0.9999	0
18	0	0.9999	0
19	0	0.9999	0
20	0	0.9999	0
21	0	0.9999	0
22	0	0.9999	0
23	0	0.9999	0
24	0	0.9999	0
25	0	0.9999	0
26	0	0.9999	0
27	0	0.9999	0
28	0	0.9999	0
29	0	0.9999	0
30	0	0.9999	0
31	0	0.9999	0
32	0	0.9999	0
33	0	0.9999	0
34	0	0.9999	0
35	0	0.9999	0
36	0	0.9999	0
37	0	0.9999	0
38	0	0.9999	0
39	0	0.9999	0
40	0	0.9999	0
41	0	0.9999	0
42	0	0.9999	0
43	0	0.9999	0
44	0	0.9999	0
45	0	0.9999	0
46	0	0.9999	0
47	0	0.9999	0
48	0	0.9999	0
49	0	0.9999	0
50	0	0.9999	0
51	0	0.9999	0
52	0	0.9999	0
53	0	0.9999	0
54	0	0.9999	0
55	0	0.9999	0
56	0	0.9999	0
57	0	0.9999	0
58	0	0.9999	0
59	0	0.9999	0
60	0	0.9999	0
61	0	0.9999	0
62	0	0.9999	0
63	0	0.9999	0
64	0	0.9999	0
65	0	0.9999	0
66	0	0.9999	0
67	0	0.9999	0
68	0	0.9999	0
69	0	0.9999	0
70	0	0.9999	0
71	0	0.9999	0
72	0	0.9999	0
73	0	0.9999	0
74	0	0.9999	0
75	0	0.9999	0
76	0	0.9999	0
77	0	0.9999	0
78	0	0.9999	0
79	0	0.9999	0
80	0	0.9999	0
81	0	0.9999	0
82	0	0.9999	0
83	0	0.9999	0
84	0	0.9999	0
85	0	0.9999	0
86	0	0.9999	0
87	0	0.9999	0
88	0	0.9999	0
89	0	0.9999	0
90	0	0.9999	0
91	0	0.9999	0
92	0	0.9999	0
93	0	0.9999	0
94	0	0.9999	0
95	0	0.9999	0
96	0	0.9999	0
97	0	0.9999	0
98	0	0.9999	0
99	0	0.9999	0
100	0	0.9999	0

Gambar 4. 19 Hasil Confidence Score Setiap Gambar

Pada prediksi yang tepat, confidence cenderung sangat tinggi dan stabil, dengan rata-rata 0,9966 serta median 1,0000. Bahkan pada batas bawah sebaran yang masih umum, persentil 10 berada di 0,9990, yang menunjukkan bahwa sebagian besar keputusan benar diberikan dengan keyakinan yang nyaris maksimal; hanya sebagian kecil prediksi benar yang memiliki confidence rendah (misalnya terdapat sejumlah kecil sampel dengan confidence di bawah 0,8 dan bahkan di bawah 0,6), sehingga kasus “benar tapi ragu” relatif jarang.

Sebaliknya, pada prediksi yang keliru, confidence umumnya lebih rendah dan lebih bervariasi, ditunjukkan oleh rata-rata 0,7702 dan median 0,8205. Pola ini mengindikasikan bahwa ketika model melakukan kesalahan, model lebih sering berada pada kondisi “kurang yakin” dibandingkan saat prediksinya tepat; misalnya persentil 10 berada di 0,5393, menandakan adanya kelompok kesalahan dengan confidence rendah. Namun demikian, tetap ditemukan sebagian kecil kasus keliru dengan confidence sangat tinggi, terlihat dari persentil 90 sebesar 0,9902 dan nilai maksimum yang mencapai 1,0000. Secara praktis, nilai confidence dapat dimanfaatkan untuk penandaan kebutuhan verifikasi (misalnya meninjau ulang prediksi dengan confidence rendah) serta dipertimbangkan untuk kalibrasi confidence agar probabilitas yang dihasilkan lebih merefleksikan tingkat ketidakpastian model secara proporsional.

Dalam visualisasi hasil prediksi dengan model YOLO11 pada skenario advanced mixing augmentation, kita dapat melihat contoh gambar yang diprediksi dengan benar dan salah.



Gambar 4. 20 Visualisasi Lable Prediksi

Berdasarkan gambar, terlihat bahwa model menunjukkan beberapa kesalahan dalam klasifikasi penyakit daun tomat meskipun menggunakan teknik advanced mixing augmentation untuk memperkaya dataset pelatihan. Pada gambar

pertama, model salah mengklasifikasikan daun yang terinfeksi Bacterial Spot sebagai Yellow Leaf Curl Virus.

Gejala pada Bacterial Spot melibatkan bercak kecil dengan batas kabur, sementara Yellow Leaf Curl Virus biasanya menyebabkan daun melengkung dengan warna kuning pada tepi daun. Kesalahan ini bisa terjadi karena ada kesamaan warna kuning pada daun yang terinfeksi Bacterial Spot, yang mungkin membingungkan model dalam membedakan kedua penyakit tersebut. Di sisi lain, gambar kedua menunjukkan prediksi yang benar untuk Early Blight, namun dengan kepercayaan yang sangat tinggi (1.00), yang mengindikasikan model mungkin terlalu yakin pada prediksi ini, padahal dalam kasus yang lebih sulit atau ambigu, ini bisa menjadi masalah. Untuk gambar ketiga, model salah mengklasifikasikan Late Blight sebagai Target Spot. Meskipun kedua penyakit ini memiliki pola bercak yang serupa, Late Blight biasanya menimbulkan bercak besar dan merusak daun secara luas, sedangkan Target Spot lebih sering terlihat dengan bercak kecil yang terdefinisi jelas. Model kesulitan membedakan antara bercak besar dan kecil ini, yang menyebabkan kesalahan klasifikasi.

Gambar keempat dan kelima menunjukkan prediksi yang benar untuk Leaf Mold dan Spider Mites, masing-masing dengan kepercayaan yang sangat tinggi (1.00), yang menunjukkan bahwa model dapat dengan baik mengklasifikasi kedua penyakit tersebut. Begitu juga pada gambar keenam, Target Spot terklasifikasi dengan benar, meskipun ada sedikit penurunan kepercayaan (0.93), yang tetap menunjukkan prediksi yang tepat. Namun, pada gambar terakhir, model salah mengklasifikasikan Yellow Leaf Curl Virus sebagai Leaf Mold. Yellow Leaf Curl

Virus menyebabkan daun melengkung dengan warna kuning, sedangkan Leaf Mold lebih banyak menimbulkan bercak hijau atau kuning tanpa perubahan bentuk daun yang signifikan. Ini menunjukkan bahwa meskipun teknik advanced mixing augmentation meningkatkan keragaman data pelatihan, model masih kesulitan membedakan penyakit dengan gejala yang sangat mirip, terutama terkait perubahan warna daun atau bercak yang serupa.

Secara keseluruhan, meskipun model menunjukkan kinerja yang baik dalam banyak kasus, kesalahan ini menunjukkan adanya ruang untuk perbaikan, terutama dalam membedakan penyakit dengan gejala yang hampir identik. Salah satu alasan rasional dari kesalahan ini adalah bahwa model, meskipun terlatih dengan dataset yang lebih beragam berkat augmentasi, masih bergantung pada fitur visual utama yang tidak cukup membedakan gejala yang lebih halus. Dalam banyak kasus, perbedaan antara penyakit seperti Target Spot dan Late Blight atau antara Bacterial Spot dan Yellow Leaf Curl Virus sangat kecil dan dapat dipengaruhi oleh faktor eksternal seperti pencahayaan atau sudut pandang. Oleh karena itu, meskipun augmentasi meningkatkan keragaman data, model masih perlu disesuaikan dengan fitur yang lebih spesifik atau data tambahan yang lebih bervariasi dalam hal kondisi lingkungan dan variasi gejala untuk meningkatkan kemampuannya dalam membedakan penyakit yang sangat mirip.

4.5.3 Photometric Augmentation

Sebelum kita membahas lebih lanjut mengenai hasil evaluasi model pada photometric augmentation, mari kita tinjau terlebih dahulu beberapa aspek dari

pelatihan yang telah dilakukan. Pada eksperimen ini, model YOLO11 dilatih menggunakan teknik photometric augmentation.

4.5.3.1 Hasil Evaluasi

Hasil evaluasi berikut ini memberikan gambaran tentang bagaimana model tersebut berperforma selama pelatihan pada setiap epoch, termasuk metrik seperti akurasi, loss, serta waktu yang dibutuhkan untuk setiap epoch. Skenario ini dilakukan dengan melakukan 10 trial dengan hasil sebagai berikut.

Tabel 4. 19 Hasil Skenario Photometric Augmentation

Trial	Train Loss	Val Loss	Accuracy
1	0.07630	0.19885	0.9465
2	0.04241	0.20317	0.9559
3	0.33622	0.17414	0.9549
4	0.23671	0.16002	0.9629
5	0.04556	0.21278	0.9524
6	0.02694	0.21073	0.9534
7	0.32465	0.17063	0.9530
8	0.04247	0.20115	0.9610
9	0.01393	0.21566	0.9614
10	0.31484	0.24697	0.9405
Rata - Rata			0.9542

Berdasarkan perbandingan sepuluh percobaan, Trial 4 dipilih sebagai konfigurasi terbaik karena memberikan accuracy tertinggi (0,9629). Berikut adalah tabel hasil evaluasi trial ke 4.

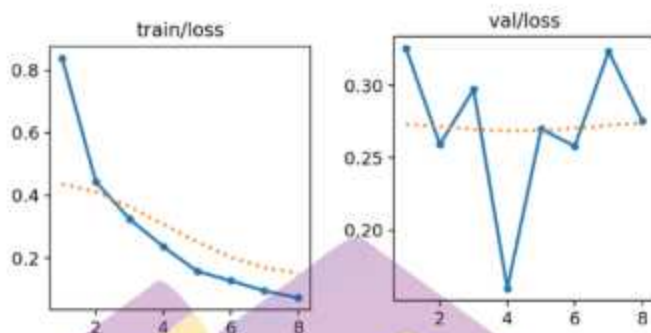
Tabel 4. 20 Hasil Skenario photometric Augmentation Trail ke 4

Epoch	time	train/loss	accuracy top1	accuracy top5	val/loss
1	701.294	0.83848	0.894	0.994	0.3254
2	1374.74	0.44376	0.913	0.999	0.25932
3	2039.81	0.32468	0.918	0.998	0.29724
4	2710.53	0.23671	0.954	0.997	0.16002
5	3377.28	0.1582	0.916	0.998	0.26988
6	4033.88	0.12803	0.929	0.997	0.25816
7	4696.12	0.09531	0.928	0.994	0.32355

8	5359.65	0.07358	0.934	0.996	0.2755
---	---------	---------	-------	-------	--------

Berdasarkan log pelatihan 8 epoch, proses optimisasi menunjukkan penurunan train/loss yang konsisten dari 0,83848 (epoch 1) menjadi 0,07358 (epoch 8), menandakan model semakin baik mempelajari pola pada data latih. Namun, performa validasi tidak membaik secara linear: val/loss awalnya turun dari 0,3254 (epoch 1) menjadi 0,25932 (epoch 2), sempat naik pada epoch 3 (0,29724), lalu mencapai nilai terbaik (terendah) pada epoch 4 sebesar 0,16002. Pada titik yang sama, accuracy_top1 juga mencapai puncaknya yaitu 0,954, sehingga epoch 4 dapat dipandang sebagai fase paling optimal karena memberikan kombinasi generalisasi terbaik di validasi dan performa prediksi tertinggi pada top-1.

Setelah epoch 4, meskipun train/loss terus menurun, val/loss justru kembali meningkat (misalnya 0,26988–0,32355) dan accuracy_top1 berfluktuasi (turun ke 0,916 di epoch 5 lalu naik perlahan hingga 0,934 di epoch 8). Pola ini mengindikasikan bahwa setelah mencapai titik optimal di epoch 4, pelatihan lanjutan lebih banyak memperbaiki kinerja pada data latih tanpa perbaikan validasi yang sebanding, sehingga berpotensi mengarah pada kecenderungan overfitting ringan. Sementara itu, accuracy_top5 sejak awal sudah sangat tinggi (0,994–0,999) dan relatif stabil, yang menunjukkan bahwa kandidat kelas benar hampir selalu berada dalam lima prediksi teratas sepanjang proses pelatihan. Gambar 4. 21 Train Val Loss Photometric Augmentation.



Gambar 4. 21 Train Val Loss Photometric Augmentation

4.5.3.2 Hasil Precision, Recall, F1-Score

Pada skenario Photometric Augmentation, model menunjukkan hasil yang cukup baik dengan metrik evaluasi seperti Precision, Recall, dan F1-Score yang terukur pada setiap kelas. Precision menggambarkan seberapa banyak prediksi yang benar dari total prediksi positif yang dilakukan oleh model, sedangkan Recall menggambarkan seberapa banyak prediksi benar dari total data positif yang sebenarnya ada dalam kelas tersebut. F1-Score adalah ukuran yang menggabungkan Precision dan Recall untuk memberikan gambaran seimbang mengenai kinerja model, terutama dalam situasi di mana terdapat ketidakseimbangan kelas. Berikut tabel hasilnya.

Tabel 4. 21 Hasil Precision, Recall, F1-Score Photometric Augmentation

Penyakit	Precision	Recall	F1-Score	Support
Bacterial spot	0.961	0.980	0.970	200
Early blight	0.932	0.960	0.946	200
Healthy	0.976	1.000	0.988	200
Late blight	0.928	0.965	0.946	200
Leaf Mold	0.989	0.915	0.951	200
Mosaic virus	0.975	0.985	0.980	200
Septoria leaf spot	0.990	0.965	0.977	200
Spider mites	0.942	0.970	0.956	200
Target Spot	0.949	0.930	0.939	200

Yellow Leaf Curl Virus	0.995	0.960	0.977	200
Rata - rata	0.964	0.963	0.963	200

Model menunjukkan kinerja yang sangat baik dengan hasil yang konsisten tinggi dalam berbagai metrik. Penyakit Healthy terklasifikasi dengan sangat baik, dengan precision 0.976 dan recall 1.000, yang berarti model hampir selalu mengidentifikasi daun sehat dengan benar tanpa menghasilkan kesalahan. F1-score yang tinggi (0.988) menegaskan bahwa model mampu mempertahankan keseimbangan yang baik antara precision dan recall untuk kelas ini.

Untuk penyakit Bacterial_spot, precision (0.961) dan recall (0.980) sedikit lebih rendah dibandingkan dengan Healthy, tetapi tetap sangat baik, dengan F1-score 0.970. Model menunjukkan kemampuan yang baik dalam mengklasifikasi Bacterial Spot dengan hanya sedikit kesalahan dalam klasifikasi.

Penyakit lainnya, seperti Mosaic_virus dan Septoria_leaf_spot, juga menunjukkan performa yang baik, dengan precision lebih dari 0.975 dan recall lebih dari 0.960, serta F1-scores tinggi di sekitar 0.977 hingga 0.980. Ini menunjukkan bahwa model dapat mengenali penyakit-penyakit ini dengan baik, dengan sedikit kesalahan.

Namun, ada beberapa penyakit yang menunjukkan sedikit penurunan dalam performa, seperti Leaf_Mold dan Target_Spot. Untuk Leaf_Mold, precision sangat tinggi (0.989), tetapi recall lebih rendah (0.915), yang menunjukkan bahwa model sering mengklasifikasikan penyakit ini dengan benar saat ada, namun lebih banyak contoh dari penyakit ini tidak terklasifikasi. Begitu juga untuk Target_Spot, yang memiliki precision dan recall yang lebih rendah (0.949 dan 0.930), menunjukkan

bahwa meskipun sebagian besar sampel terklasifikasi dengan benar, masih ada kesalahan klasifikasi yang lebih tinggi dibandingkan penyakit lainnya.

Secara keseluruhan, model memiliki accuracy yang sangat baik pada 0.963, dengan nilai macro avg dan weighted avg yang hampir identik (0.964 untuk precision dan 0.963 untuk recall dan F1-score), menunjukkan performa yang stabil dan konsisten di seluruh kelas. Kesalahan kecil di beberapa penyakit dapat diperbaiki lebih lanjut, namun model sudah mencapai tingkat akurasi yang sangat baik, dengan kemampuan kuat dalam mengidentifikasi penyakit daun tomat secara keseluruhan.

4.5.3.3 Hasil Confusion Matrix

Confusion matrix yang ditampilkan pada gambar ini menunjukkan kinerja model dalam mengklasifikasikan penyakit daun dengan menggunakan data pengujian. Berikut adalah confusion matrixnya.



Gambar 4. 22 Gambar Confusion Matrix Skenario Photometric Augmentation

Berdasarkan confusion matriks, model menunjukkan performa yang sangat baik meskipun ada beberapa kesalahan klasifikasi. Pada Bacterial Spot, model berhasil mengklasifikasikan hampir semua sampel dengan benar, dengan hanya 4 sampel salah diklasifikasikan sebagai Early Blight dan 1 sebagai Mosaic Virus. Ini menunjukkan bahwa model sangat efektif dalam mengenali penyakit ini.

Untuk Early Blight, 192 sampel terklasifikasi dengan benar, namun ada 3 sampel yang salah diklasifikasikan sebagai Bacterial Spot, 2 sebagai Healthy, dan 1 sebagai Mosaic Virus. Kesalahan ini mungkin disebabkan oleh kemiripan gejala

antara Early Blight dan penyakit lainnya, terutama dalam hal bercak pada daun yang terlihat mirip.

Pada kategori Healthy, semua 200 sampel terklasifikasi dengan benar, yang menunjukkan bahwa model sangat efektif dalam mengidentifikasi daun sehat. Namun, pada Late Blight, meskipun sebagian besar sampel terklasifikasi dengan benar (193 benar), ada 6 kesalahan yang salah diklasifikasikan sebagai Bacterial Spot dan 1 sebagai Target Spot. Hal ini menunjukkan bahwa model sedikit bingung antara Late Blight dan beberapa penyakit lainnya yang menunjukkan bercak pada daun.

Leaf Mold menunjukkan 183 klasifikasi yang benar, tetapi ada 9 sampel yang salah diklasifikasikan sebagai Early Blight, 3 sebagai Mosaic Virus, dan beberapa kesalahan minor lainnya. Pada Mosaic Virus, 197 sampel terklasifikasi dengan benar, dengan 2 kesalahan sebagai Bacterial Spot dan 1 sebagai Healthy, yang menunjukkan kemampuan model dalam mengidentifikasi penyakit ini meskipun ada beberapa kebingungan.

Pada Septoria Leaf Spot, model mengklasifikasikan 193 sampel dengan benar, namun ada beberapa kesalahan minor, termasuk 1 yang salah sebagai Bacterial Spot, 2 sebagai Early Blight, dan 1 sebagai Mosaic Virus. Pada Spider Mites, ada sedikit kebingungan, dengan 6 sampel salah diklasifikasikan sebagai Target Spot, meskipun 194 lainnya terklasifikasi dengan benar.

Target Spot menunjukkan 186 klasifikasi yang benar, namun ada beberapa kesalahan, dengan 2 sampel salah diklasifikasikan sebagai Bacterial Spot dan 5 sebagai Early Blight. Terakhir, pada Yellow Leaf Curl Virus, 192 sampel

terklasifikasikan dengan benar, namun ada 3 kesalahan yang diklasifikasikan sebagai Bacterial Spot, 5 sebagai Target Spot, dan 1 sebagai Spider Mites.

Secara keseluruhan, meskipun ada beberapa kesalahan klasifikasi pada penyakit dengan gejala yang mirip, model menunjukkan performa yang sangat baik dengan kebanyakan sampel terklasifikasi dengan benar. Kesalahan-kesalahan ini sebagian besar terjadi antara penyakit yang memiliki gejala serupa, seperti Bacterial Spot dan Early Blight, atau Mosaic Virus dan Leaf Mold. Dengan sedikit perbaikan pada pengenalan gejala yang lebih halus, model dapat lebih akurat dalam membedakan penyakit dengan gejala yang mirip.

4.5.3.4 Analisis Confidence Score

Berdasarkan analisis confidence score (yaitu probabilitas maksimum/top-1 probability yang dipilih model pada setiap sampel) pada file `probs_allclasses_trial4.csv`, terlihat perbedaan yang tegas antara prediksi yang sesuai label dan yang tidak sesuai. Pada prediksi yang sesuai label, confidence cenderung tinggi dan stabil, dengan rata-rata 0,960 serta median 0,999. Sebaran bawahnya pun masih relatif kuat, di mana persentil 10 = 0,872 dan persentil 25 = 0,982, sedangkan bagian atasnya sangat dominan pada nilai maksimal (persentil 75 dan 90 = 1,000). Pola ini menunjukkan bahwa ketika model berada pada keputusan yang benar, model biasanya memberikan keyakinan yang sangat kuat dan konsisten.

Sebaliknya, pada prediksi yang tidak sesuai label, confidence cenderung lebih rendah dan lebih menyebar, dengan rata-rata 0,622 dan median 0,6045. Distribusi kesalahan banyak terkonsentrasi pada confidence menengah-rendah,

terlihat dari persentil 25 = 0,4908 dan persentil 75 = 0,7718, sehingga mayoritas kesalahan terjadi saat model berada pada kondisi yang relatif “ragu”. Temuan ini penting dicatat karena mengindikasikan adanya kondisi tertentu di mana model dapat keliru tetapi sangat yakin, yang biasanya berkaitan dengan kemiripan pola visual antarkelas atau fitur dominan pada citra yang membuat model terkunci pada satu kelas. Gambar 4. 23 adalah hasil confidence score.

Image	Class	Confidence	...
img_bacterial_001_000	Bacterial	0.99	...
img_bacterial_002_000	Bacterial	0.99	...
img_bacterial_003_000	Bacterial	0.99	...
img_bacterial_004_000	Bacterial	0.99	...
img_bacterial_005_000	Bacterial	0.99	...
img_bacterial_006_000	Bacterial	0.99	...
img_bacterial_007_000	Bacterial	0.99	...
img_bacterial_008_000	Bacterial	0.99	...
img_bacterial_009_000	Bacterial	0.99	...
img_bacterial_010_000	Bacterial	0.99	...
img_bacterial_011_000	Bacterial	0.99	...
img_bacterial_012_000	Bacterial	0.99	...
img_bacterial_013_000	Bacterial	0.99	...
img_bacterial_014_000	Bacterial	0.99	...
img_bacterial_015_000	Bacterial	0.99	...
img_bacterial_016_000	Bacterial	0.99	...
img_bacterial_017_000	Bacterial	0.99	...
img_bacterial_018_000	Bacterial	0.99	...
img_bacterial_019_000	Bacterial	0.99	...
img_bacterial_020_000	Bacterial	0.99	...
img_bacterial_021_000	Bacterial	0.99	...
img_bacterial_022_000	Bacterial	0.99	...
img_bacterial_023_000	Bacterial	0.99	...
img_bacterial_024_000	Bacterial	0.99	...
img_bacterial_025_000	Bacterial	0.99	...
img_bacterial_026_000	Bacterial	0.99	...
img_bacterial_027_000	Bacterial	0.99	...
img_bacterial_028_000	Bacterial	0.99	...
img_bacterial_029_000	Bacterial	0.99	...
img_bacterial_030_000	Bacterial	0.99	...
img_bacterial_031_000	Bacterial	0.99	...
img_bacterial_032_000	Bacterial	0.99	...
img_bacterial_033_000	Bacterial	0.99	...
img_bacterial_034_000	Bacterial	0.99	...
img_bacterial_035_000	Bacterial	0.99	...
img_bacterial_036_000	Bacterial	0.99	...
img_bacterial_037_000	Bacterial	0.99	...
img_bacterial_038_000	Bacterial	0.99	...
img_bacterial_039_000	Bacterial	0.99	...
img_bacterial_040_000	Bacterial	0.99	...
img_bacterial_041_000	Bacterial	0.99	...
img_bacterial_042_000	Bacterial	0.99	...
img_bacterial_043_000	Bacterial	0.99	...
img_bacterial_044_000	Bacterial	0.99	...
img_bacterial_045_000	Bacterial	0.99	...
img_bacterial_046_000	Bacterial	0.99	...
img_bacterial_047_000	Bacterial	0.99	...
img_bacterial_048_000	Bacterial	0.99	...
img_bacterial_049_000	Bacterial	0.99	...
img_bacterial_050_000	Bacterial	0.99	...

Gambar 4. 23 Hasil Confidence Score Setiap Gambar Dalam visualisasi hasil prediksi dengan model YOLO11 pada skenario photometric augmentation, kita dapat melihat contoh gambar yang diprediksi dengan benar dan salah.



Gambar 4. 24 Visualisasi Label Prediksi Skenario Photometric Augmentation

Berdasarkan gambar, terdapat 9 kesalahan prediksi yang dapat dijelaskan dalam konteks penggunaan teknik photometric augmentation, yang berfungsi untuk memperkenalkan variasi dalam pencahayaan, kontras, dan saturasi warna pada dataset pelatihan. Pada gambar pertama, model salah mengklasifikasikan Bacterial

Spot sebagai Yellow Leaf Curl Virus. Gejala Bacterial Spot berupa bercak kecil dengan batas kabur dapat terlihat mirip dengan Yellow Leaf Curl Virus, yang menyebabkan daun melengkung dan menguning. Pencahayaan yang tidak merata akibat augmentation ini bisa membuat perbedaan visual antara keduanya menjadi kurang jelas. Pada gambar kedua, Early Blight yang memiliki bercak melingkar dan tepi kabur salah diklasifikasikan sebagai Yellow Leaf Curl Virus, yang biasanya menyebabkan perubahan bentuk daun. Pengaruh saturasi warna dan kontras yang tinggi dapat membuat perbedaan antara bercak dan perubahan bentuk daun menjadi lebih sulit dibedakan.

Pada gambar ketiga, model salah mengidentifikasi Late Blight sebagai Leaf Mold. Late Blight menghasilkan bercak besar dan gelap, sementara Leaf Mold cenderung menimbulkan bercak jamur berwarna lebih cerah. Perbedaan pencahayaan dan kontras pada gambar dapat memperburuk perbedaan visual antara kedua penyakit ini, menyebabkan kesalahan klasifikasi. Begitu pula pada gambar keempat, di mana model salah mengklasifikasikan Leaf Mold sebagai Spider Mites. Leaf Mold memiliki bercak yang lebih besar dan terdistribusi merata, sementara Spider Mites lebih sering menyebabkan bintik kuning kecil pada daun. Variasi dalam pencahayaan yang diperkenalkan oleh augmentation bisa membuat bercak yang lebih luas pada Leaf Mold tampak lebih kecil atau terdistorsi, membingungkan model.

Pada gambar kelima, model salah mengklasifikasikan Mosaic Virus sebagai Target Spot, meskipun kedua penyakit ini memiliki bercak pada daun, namun Mosaic Virus lebih sering menyebabkan pola bercak yang lebih seragam.

Kesalahan ini bisa disebabkan oleh perbedaan pencahayaan yang membuat pola pada daun menjadi lebih terdistorsi, sehingga model kesulitan membedakan keduanya. Pada gambar keenam, model salah mengidentifikasi Spider Mites sebagai Yellow Leaf Curl Virus. Spider Mites menyebabkan bintik kuning kecil pada daun, sedangkan Yellow Leaf Curl Virus lebih sering menyebabkan perubahan bentuk dan warna daun. Pencahayaan yang lebih terang atau lebih gelap bisa mengaburkan perbedaan ini, menyebabkan model salah klasifikasi.

Di gambar ketujuh, model mengklasifikasikan Target Spot sebagai Early Blight, meskipun Target Spot memiliki bercak bulat dengan batas yang lebih jelas, sementara Early Blight memiliki bercak yang lebih kabur. Perbedaan pencahayaan yang diperkenalkan oleh augmentation bisa membuat perbedaan antara bercak bulat dan bercak kabur menjadi lebih sulit dikenali. Pada gambar kedelapan, model salah mengidentifikasi Yellow Leaf Curl Virus sebagai Spider Mites. Yellow Leaf Curl Virus menyebabkan daun melengkung dan menguning, sedangkan Spider Mites menyebabkan bintik-bintik kuning kecil. Perubahan saturasi warna yang kuat akibat augmentation bisa memperburuk perbedaan visual antara perubahan bentuk daun dan bintik kuning kecil.

Terakhir, pada gambar kesembilan, model mengklasifikasikan Bacterial Spot sebagai Target Spot. Kedua penyakit ini memiliki bercak pada daun, namun Bacterial Spot lebih sering terlihat dengan bercak yang lebih kecil dan basah, sementara Target Spot lebih terdefinisi dengan jelas. Peningkatan kontras dan kecerahan yang tidak merata dapat membuat perbedaan antara bercak melingkar

dan bercak basah menjadi kurang jelas, menyebabkan kebingungan dalam klasifikasi.

Secara keseluruhan, meskipun photometric augmentation bertujuan untuk meningkatkan keragaman data pelatihan, teknik ini juga memperkenalkan tantangan baru, terutama dalam membedakan penyakit dengan gejala yang sangat mirip. Variasi dalam pencahayaan, kontras, dan saturasi dapat membuat perbedaan visual yang halus antara penyakit sulit dikenali oleh model, yang menyebabkan kesalahan klasifikasi. Untuk meningkatkan akurasi, model mungkin perlu dilatih lebih lanjut dengan penyesuaian pada teknik augmentasi atau pengenalan fitur yang lebih spesifik untuk setiap penyakit.

4.5.4 Spatial Transformations

Spatial Transformations, model dilatih dengan menggunakan teknik augmentasi yang memfokuskan pada perubahan spasial pada gambar. Ini mencakup rotasi, pemotongan, pergeseran, serta teknik-teknik transformasi geometris lainnya yang bertujuan untuk memperkenalkan variasi dalam orientasi dan posisi objek pada gambar. Dengan cara ini, model diajari untuk mengenali objek dari berbagai sudut dan posisi, meningkatkan kemampuannya untuk menggeneralisasi ketika menghadapi gambar baru yang memiliki perubahan dalam perspektif atau posisi objek.

4.5.3.1 Hasil Evaluasi

Hasil evaluasi berikut ini memberikan gambaran tentang bagaimana model tersebut berperforma selama pelatihan pada setiap epoch, termasuk metrik seperti

akurasi, loss, serta waktu yang dibutuhkan untuk setiap epoch. Skenario ini dilakukan 10 kali trial dengan hasil berikut.

Tabel 4. 22 Hasil 10 Trial Skenario Spatial Transformations

Trial	Train Loss	Val Loss	Accuracy
1	0.00305	0.02489	0.9919
2	0.09385	0.04580	0.9860
3	0.09895	0.05059	0.9860
4	0.02873	0.03400	0.9850
5	0.22392	0.07450	0.9750
6	0.00600	0.02312	0.9940
7	0.10039	0.04295	0.9894
8	0.02502	0.04323	0.9844
9	0.00204	0.02275	0.99
10	0.06173	0.04563	0.9904
Rata - Rata			0.987250

Berdasarkan perbandingan sepuluh percobaan, Trial 6 ditetapkan sebagai trial terbaik karena memberikan kombinasi performa paling unggul, ditunjukkan oleh accuracy tertinggi (0,9940). Trial 6 merepresentasikan kondisi model yang paling optimal karena mampu mempertahankan error validasi rendah sekaligus menghasilkan performa prediksi paling tinggi dibanding trial lainnya, sehingga paling layak dijadikan hasil akhir dalam penelitian ini. Berikut adalah tabel hasil evaluasi trial ke 6.

Tabel 4. 23 Tabel Hasil Evaluasi Skenario Spatial Transformations Trial ke 6

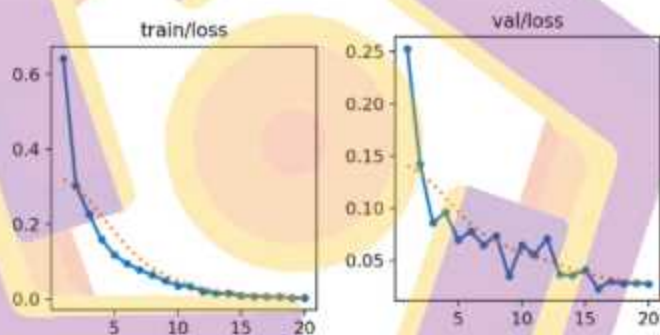
Epoch	time	train/loss	accuracy_top1	accuracy_top5	val/loss
1	661.571	0.64231	0.911	1	0.25268
2	1309.77	0.301	0.952	1	0.14225
3	1943.51	0.22699	0.973	1	0.086
4	2568.71	0.15843	0.965	1	0.09614
5	3193.4	0.11745	0.976	1	0.06961
6	3821.02	0.09364	0.974	1	0.0778
7	4442.57	0.07746	0.979	1	0.06474
8	5066.39	0.06329	0.973	1	0.07366
9	5693.96	0.0481	0.985	1	0.03483

10	6321.07	0.03519	0.973	1	0.06493
11	6957.35	0.03306	0.979	1	0.05564
12	7588.49	0.01986	0.982	1	0.07091
13	8218.2	0.01403	0.991	1	0.03643
14	8847.51	0.01466	0.991	1	0.03546
15	9517.59	0.00835	0.985	1	0.0407
16	10335.5	0.006	0.995	1	0.02312
17	10970.5	0.00593	0.993	1	0.02986
18	11602.6	0.00476	0.989	1	0.02787
19	12239	0.00217	0.994	1	0.02837
20	12877.7	0.0023	0.993	1	0.0272

Berdasarkan log pelatihan trial 6 selama 20 epoch, proses training menunjukkan tren konvergensi yang jelas. Nilai train/loss turun secara konsisten dari 0,64231 pada epoch 1 menjadi sekitar 0,00230 pada epoch 20, menandakan model semakin mampu mempelajari pola pada data latih dan error training terus mengecil. Sejalan dengan itu, accuracy_top1 meningkat cepat pada fase awal (0,911 di epoch 1 menjadi 0,973 di epoch 3), lalu bergerak fluktuatif namun tetap tinggi pada rentang 0,965–0,991 hingga menjelang akhir, dan mencapai titik puncak pada epoch 16 dengan accuracy_top1 0,995. Sepanjang pelatihan, accuracy_top5 berada di nilai 1 pada semua epoch, yang menunjukkan bahwa kelas benar hampir selalu masuk dalam lima prediksi teratas dan kesalahan yang terjadi lebih banyak terkait pemilihan peringkat teratas, bukan karena model sama sekali “melenceng” jauh dari kelas target.

Dari sisi generalisasi, val/loss memperlihatkan pola yang tidak selalu turun linier, tetapi mengalami perbaikan bertahap dengan beberapa osilasi. Val/loss awalnya turun dari 0,25268 (epoch 1) ke 0,08600 (epoch 3), sempat naik tipis pada epoch 4 (0,09614), lalu kembali membaik hingga 0,06474 (epoch 7). Perbaikan paling signifikan terjadi pada epoch 9 ketika val/loss turun ke 0,03483, kemudian

sempat naik lagi pada epoch 10–12, sebelum kembali turun dan mencapai nilai terbaik pada epoch 16 yaitu 0,02312. Setelah epoch 16, val/loss tetap berada pada kisaran rendah (sekitar 0,027–0,030), menunjukkan performa validasi yang stabil meskipun train/loss terus menurun. Pola ini mengindikasikan bahwa epoch 16 merupakan titik pelatihan paling optimal karena memberikan kombinasi terbaik antara error validasi terendah dan performa prediksi top-1 tertinggi, sementara pelatihan setelahnya cenderung menghasilkan peningkatan kecil pada training tanpa keuntungan validasi yang lebih baik. Gambar 4. 25 Grafik Train Val Loss Spatial Transformations.



Gambar 4. 25 Grafik Train Val Loss Spatial Transformations

4.5.3.2 Hasil Precision, Recall, F1-Score

Berikut adalah hasil evaluasi model pada skenario spatial transformations dengan menampilkan Precision, Recall, dan F1-Score untuk setiap kelas yang terklasifikasi oleh model.

Tabel 4. 24 Hasil Precision, Recall, F1-Score Spatial Transformations

Penyakit	Precision	Recall	F1-Score	Support
Bacterial spot	0.990	1.000	0.995	200
Early blight	0.980	0.990	0.985	200
Healthy	1.000	1.000	1.000	200

Late blight	0.990	0.980	0.985	200
Leaf Mold	1.000	0.995	0.997	200
Mosaic virus	1.000	1.000	1.000	200
Septoria leaf spot	0.990	1.000	0.995	200
Spider mites	1.000	0.990	0.995	200
Target Spot	0.990	0.985	0.987	200
Yellow Leaf Curl Virus	1.000	1.000	1.000	200
Rata -rata	0.994	0.994	0.994	200

Model menunjukkan performa yang sangat baik di seluruh kelas penyakit daun tomat, dengan nilai precision, recall, dan F1-score yang hampir sempurna. Penyakit Healthy terklasifikasi dengan akurasi sempurna (precision, recall, dan F1-score semua 1.000), yang menunjukkan bahwa model sangat efektif dalam mengenali daun sehat tanpa kesalahan klasifikasi. Begitu juga dengan Mosaic Virus dan Yellow Leaf Curl Virus, yang mendapatkan nilai 1.000 untuk precision, recall, dan F1-score, menunjukkan kemampuan model yang sangat baik dalam mengenali kedua penyakit ini.

Penyakit lain seperti Bacterial Spot, Septoria Leaf Spot, dan Spider Mites juga menunjukkan performa yang sangat baik, dengan precision di atas 0.990 dan recall mendekati 1.000. Ini menunjukkan bahwa model hampir selalu dapat mengklasifikasi penyakit-penyakit ini dengan sedikit kesalahan, meskipun ada sedikit penurunan pada Spider Mites dengan recall 0.990 dan F1-score 0.995.

Pada Early Blight, Late Blight, Leaf Mold, dan Target Spot, meskipun model tetap menunjukkan precision dan recall yang sangat baik (sekitar 0.980 hingga 1.000), ada sedikit penurunan dalam F1-score, terutama untuk Late Blight (0.985) dan Target Spot (0.987). Hal ini menunjukkan bahwa meskipun model

mengklasifikasikan sebagian besar sampel dengan benar, ada sedikit kesalahan dalam klasifikasi penyakit ini yang menyebabkan penurunan F1-score.

Secara keseluruhan, model ini menunjukkan performa yang luar biasa dengan nilai macro avg dan weighted avg yang sangat tinggi pada 0.994 untuk precision, recall, dan F1-score, menunjukkan bahwa model konsisten dalam mengenali semua kelas penyakit daun tomat dengan sangat baik. Meskipun ada sedikit penurunan pada beberapa penyakit yang lebih sulit dibedakan, seperti Early Blight dan Target Spot, model secara keseluruhan sangat efektif dalam mengklasifikasikan penyakit-penyakit ini dengan sangat akurat.

4.5.3.3 Hasil Confusion Matrix

Confusion matrix yang ditampilkan pada gambar ini menunjukkan kinerja model dalam mengklasifikasikan penyakit daun dengan menggunakan data pengujian. Berikut adalah confusion matrixnya.



Gambar 4. 26 Gambar Confusion Matrix Skenario Spatial Transformations

Berdasarkan confusion matriks, model menunjukkan kinerja yang sangat baik dengan sebagian besar prediksi yang benar. Pada Bacterial Spot, model mengklasifikasikan semua sampel dengan benar (200 benar), tanpa ada kesalahan, menunjukkan akurasi sempurna dalam mengklasifikasi penyakit ini. Untuk Early Blight, model juga menunjukkan hasil yang sangat baik dengan 198 sampel yang terklasifikasi dengan benar, meskipun ada 1 sampel yang salah diklasifikasikan sebagai Bacterial Spot dan 1 lagi sebagai Healthy. Kesalahan kecil ini mungkin disebabkan oleh gejala yang mirip antara Bacterial Spot dan Early Blight, atau adanya variasi dalam pencahayaan yang mempengaruhi model.

Pada kategori Healthy, semua 200 sampel terklasifikasi dengan benar, yang menunjukkan kemampuan model yang sangat baik dalam membedakan daun sehat dari penyakit. Untuk Late Blight, meskipun sebagian besar sampel terklasifikasi dengan benar (196 benar), ada 4 kesalahan yang salah diklasifikasikan sebagai Bacterial Spot. Hal ini menunjukkan sedikit kebingungan antara penyakit yang memiliki bercak serupa pada daun.

Model juga berhasil mengklasifikasikan Leaf Mold dengan sangat baik, dengan hanya 1 kesalahan yang diklasifikasikan sebagai Mosaic Virus, yang menunjukkan bahwa model dapat mengklasifikasi Leaf Mold hampir dengan sempurna. Begitu juga dengan Mosaic Virus, yang 200 sampel terklasifikasikan dengan benar, menunjukkan bahwa model sangat baik dalam mengenali gejala penyakit ini.

Pada Septoria Leaf Spot, model berhasil mengklasifikasikan 200 sampel dengan benar, yang menunjukkan performa yang sangat baik. Namun, pada Spider Mites, meskipun sebagian besar sampel terklasifikasi dengan benar (198 benar), ada 2 kesalahan yang salah diklasifikasikan sebagai Target Spot, menunjukkan bahwa model kesulitan membedakan kedua penyakit ini, yang memiliki gejala yang mirip.

Untuk Target Spot, model mengklasifikasikan 197 sampel dengan benar, tetapi ada 2 kesalahan yang salah diklasifikasikan sebagai Spider Mites, dan pada Yellow Leaf Curl Virus, semua 200 sampel terklasifikasikan dengan benar, menunjukkan keakuratan yang sangat baik dalam mengklasifikasi virus ini.

terdapat 91 prediksi benar yang confidence-nya masih di bawah 0,99, dan 33 prediksi benar di bawah 0,95, menandakan bahwa sebagian kecil sampel memang lebih "sulit" meskipun akhirnya tetap terklasifikasi benar. Pola ini menunjukkan bahwa ketika model keliru, pada umumnya model berada pada kondisi keyakinan yang tidak terlalu tinggi, sedangkan kesalahan dengan keyakinan sangat tinggi relatif jarang pada trial ini.

Dalam visualisasi hasil prediksi dengan model YOLO11 pada skenario spatial transformations, dapat melihat contoh gambar yang diprediksi dengan benar dan salah.



Gambar 4. 28 Visualisasi Label Prediksi Skenario Spatial Transformations

Berdasarkan gambar, terdapat lima kesalahan prediksi yang dapat dijelaskan dengan penggunaan teknik spatial transformations dalam augmentasi data untuk tugas klasifikasi. Teknik ini bertujuan untuk memperkenalkan variasi geometris dalam dataset pelatihan, seperti rotasi, pemotongan, dan perubahan ukuran gambar, yang bertujuan untuk meningkatkan kemampuan model dalam menggeneralisasi pada variasi dunia nyata. Namun, meskipun augmentasi spasial dapat meningkatkan keragaman data, kesalahan-kesalahan yang terjadi

menunjukkan bagaimana perubahan spasial ini dapat mengganggu proses klasifikasi model, terutama ketika perubahan tersebut memengaruhi cara model mengenali gejala penyakit.

Pada gambar pertama, model dengan tepat memprediksi Early Blight dengan kepercayaan tinggi (1.00), namun meskipun prediksi ini benar, rotasi atau pemotongan gambar dapat menyebabkan distorsi yang mengubah persepsi model terhadap gejala penyakit, meskipun hasil akhirnya tetap akurat. Pada gambar kedua, model salah mengklasifikasikan Late Blight sebagai Leaf Mold. Perbedaan antara Late Blight, yang memiliki bercak besar dan gelap, dan Leaf Mold, yang cenderung menghasilkan bercak lebih terang, bisa menjadi kabur akibat perubahan spasial. Rotasi atau pemotongan gambar dapat mengaburkan perbedaan ini, sehingga menyebabkan kesalahan klasifikasi.

Pada gambar ketiga, model salah mengklasifikasikan Spider Mites sebagai Target Spot. Bintik-bintik kecil kuning yang disebabkan oleh Spider Mites seharusnya lebih mudah dibedakan dari bercak bulat pada Target Spot, namun rotasi atau perubahan ukuran gambar dapat menyebabkan model kesulitan dalam mengklasifikasi pola bercak yang lebih kecil dan terdefinisi jelas. Hal yang serupa terjadi pada gambar keempat, di mana model salah mengidentifikasi Target Spot sebagai Bacterial Spot. Target Spot memiliki bercak yang lebih terdefinisi dengan jelas, sementara Bacterial Spot menunjukkan bercak yang lebih kecil dan kabur. Transformasi spasial yang memperbesar atau memotong gambar bisa mempengaruhi cara model membedakan kedua pola ini, menyebabkan kesalahan klasifikasi.

Pada gambar kelima, meskipun Target Spot terklasifikasikan dengan benar, model memberikan nilai kepercayaan yang lebih rendah (0.81). Hal ini mungkin disebabkan oleh transformasi spasial yang menyebabkan pergeseran atau distorsi dalam ukuran dan orientasi bercak pada daun, membuat model kurang yakin dalam mengidentifikasi penyakit tersebut meskipun klasifikasinya benar.

Secara keseluruhan, meskipun teknik *spatial transformations* bertujuan untuk memperkaya data dan meningkatkan kemampuan generalisasi model, teknik ini juga dapat memperkenalkan tantangan dalam klasifikasi penyakit daun tomat. Perubahan geometris yang terjadi akibat rotasi, pemotongan, dan perubahan ukuran dapat menyebabkan model kesulitan dalam membedakan penyakit dengan gejala yang mirip. Oleh karena itu, meskipun augmentasi spasial membantu meningkatkan keragaman data pelatihan, model mungkin perlu dilatih lebih lanjut untuk lebih sensitif terhadap perbedaan halus antara gejala penyakit dan mengurangi potensi distorsi yang dapat mengganggu klasifikasi yang akurat.

4.5.5 All Augmentation

Pada skenario *All Augmentation*, berbagai teknik augmentasi data digabungkan untuk melatih model, meningkatkan keberagaman data pelatihan dan memperbaiki kemampuan generalisasi model terhadap kondisi yang berbeda. Augmentasi ini mencakup pengolahan gambar yang mencakup transformasi spasial, perubahan fotometrik, serta teknik lainnya yang bertujuan untuk membuat model lebih robust dalam menghadapi data yang variatif.

4.5.3.1 Hasil Evaluasi

Hasil evaluasi berikut ini memberikan gambaran tentang bagaimana model tersebut berperilaku selama pelatihan pada setiap epoch, termasuk metrik seperti akurasi, loss, serta waktu yang dibutuhkan untuk setiap epoch. Skenario All Augmentation dilakukan dengan 10 kali trial.

Tabel 4. 25 Hasil Semua Trial Skenario All Augmentation

Trial	Train loss	Val loss	Accuracy
1	0.01031	0.05945	0.9855
2	0.07469	0.07217	0.9800
3	0.00474	0.05157	0.9870
4	0.01286	0.05739	0.9815
5	0.14749	0.08872	0.9730
6	0.02189	0.06962	0.9815
7	0.05501	0.08129	0.9790
8	0.04237	0.04759	0.9795
9	0.03465	0.05140	0.9845
10	0.11723	0.07547	0.9785
Rata - Rata			0.981

Berdasarkan perbandingan sepuluh percobaan, trial 3 ditetapkan sebagai trial terbaik karena menghasilkan nilai accuracy tertinggi, yaitu 0,987. Dengan pertimbangan tersebut, trial 3 paling layak dijadikan model final pada tahap evaluasi ini. Berikut adalah tabel hasil evaluasi trial ke 3.

Tabel 4. 26 Hasil Evaluasi Skenario All Augmentation Trial ke 3

Epoch	time	train/loss	accuracy_top1	accuracy_top5	val/loss
1	1052.35	0.83152	0.907	0.998	0.2548
2	2304.75	0.47126	0.948	0.997	0.16838
3	3393.44	0.35197	0.964	0.999	0.09337
4	4387.9	0.25789	0.961	1	0.11599
5	5386.32	0.20044	0.97	1	0.07683
6	6385.28	0.15163	0.968	1	0.08373
7	7385.08	0.11673	0.973	1	0.07153
8	8390.73	0.08622	0.97	1	0.09876
9	9394.35	0.06675	0.978	1	0.08273
10	10395.1	0.0479	0.975	0.999	0.0785

11	11407.4	0.03835	0.979	1	0.08253
12	12429.7	0.03044	0.981	1	0.06478
13	13442.1	0.01838	0.975	0.999	0.0712
14	14453.5	0.01498	0.983	1	0.06486
15	15476.1	0.01094	0.979	1	0.06644
16	16772.1	0.00819	0.983	1	0.0595
17	17852.6	0.00569	0.979	1	0.05838
18	18877.8	0.00474	0.987	1	0.05157
19	19900.5	0.00292	0.983	1	0.05266
20	20921.1	0.00178	0.984	1	0.04408

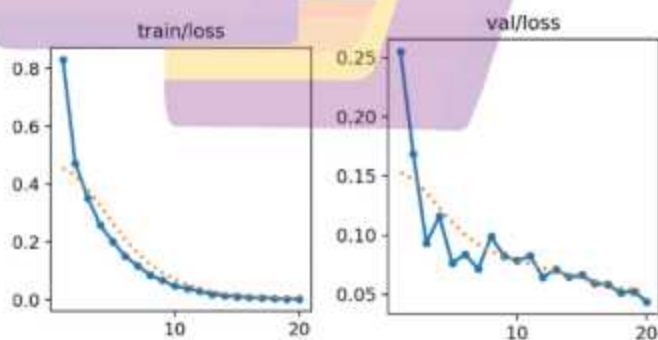
Berdasarkan data pelatihan, dapat dilihat bahwa model menunjukkan peningkatan signifikan dalam hal akurasi dan penurunan train loss sepanjang 20 epoch. Di epoch pertama, train loss cukup tinggi pada angka 0.83152, sementara akurasi top-1 masih rendah pada 90.7%. Namun, seiring berjalannya waktu, model menunjukkan perbaikan yang jelas. Di epoch kedua, train loss turun drastis menjadi 0.47126, dengan akurasi top-1 meningkat menjadi 94.8%. Pada epoch ketiga, model semakin membaik, dengan train loss menjadi 0.35197 dan akurasi top-1 mencapai 96.4%, serta `accuracy_top5` hampir mencapai 100%.

Peningkatan berlanjut pada epoch-epoch berikutnya, dengan train loss yang terus menurun, mencapai angka 0.00178 pada epoch terakhir, dan akurasi top-1 mencapai 98.4%. Pada epoch ke-20, meskipun train loss sudah sangat rendah, model masih mampu menjaga akurasi top-1 yang sangat tinggi di angka 98.4%, menunjukkan bahwa pelatihan berjalan dengan baik dan model semakin mengoptimalkan kemampuannya untuk mengklasifikasikan data pelatihan. Sementara itu, `accuracy_top5` tetap sangat tinggi sepanjang pelatihan, mencapai 100% pada epoch ke-4 dan tetap mendekati 100% hingga akhir pelatihan,

menandakan bahwa model sangat kuat dalam memberikan prediksi yang benar di posisi teratas.

Pada sisi validation loss, meskipun ada fluktuasi kecil pada beberapa epoch, secara keseluruhan validation loss terus menurun seiring berjalannya waktu, menunjukkan bahwa model tidak hanya belajar dengan baik dari data pelatihan, tetapi juga mampu menggeneralisasi dengan baik pada data yang tidak terlihat selama pelatihan. Di epoch ke-20, validation loss berada pada angka 0.04408, yang menunjukkan bahwa model berhasil mencapai keseimbangan yang baik antara mengurangi kesalahan pada data pelatihan dan meminimalkan kesalahan pada data validasi.

Secara keseluruhan, pelatihan menunjukkan progres yang sangat baik dengan penurunan signifikan pada train loss dan validation loss, serta peningkatan stabil pada akurasi. Model mencapai performa optimal dengan akurasi top-1 yang sangat tinggi, mencerminkan kemampuan model yang sangat baik dalam mengenali pola pada data pelatihan dan data validasi. Gambar 4. 29 Train/Loss dan Val/Loss All Augmentation.



Gambar 4. 29 Train/Loss dan Val/Loss All Augmentation

4.5.3.2 Hasil Precision, Recall, F1-Score

Berikut adalah hasil evaluasi model pada skenario all augmentation dengan menampilkan Precision, Recall, dan F1-Score untuk setiap kelas yang terklasifikasi oleh model. Tabel 4. 27 Hasil Precision, Recall, F1-Score All Augmentation.

Tabel 4. 27 Hasil Precision, Recall, F1-Score All Augmentation

Kelas	Precision	Recall	F1-Score	Support
Bacterial spot	0.985	1.000	0.993	200
Early blight	0.970	0.960	0.965	200
Healthy	0.995	1.000	0.998	200
Late blight	0.985	0.975	0.980	200
Leaf Mold	1.000	0.995	0.997	200
Mosaic virus	0.995	1.000	0.998	200
Septoria leaf spot	0.990	0.995	0.993	200
Spider mites	0.980	0.975	0.977	200
Target Spot	0.970	0.975	0.973	200
Yellow Leaf Curl Virus	1.000	0.995	0.997	200
Rata - rata	0.987	0.987	0.987	200

Model menunjukkan kinerja yang sangat baik dalam mengklasifikasikan penyakit daun tomat, dengan sebagian besar metrik precision, recall, dan F1-score yang sangat tinggi. Penyakit seperti Healthy dan Mosaic Virus mendapatkan precision dan recall sempurna (1.000), yang menunjukkan bahwa model dapat mengklasifikasi kondisi daun sehat atau virus dengan sangat baik tanpa kesalahan. Begitu juga dengan Yellow Leaf Curl Virus dan Leaf Mold, yang terklasifikasikan dengan precision dan recall yang sangat tinggi (1.000 dan 0.995), mencerminkan akurasi model yang hampir sempurna dalam mengklasifikasi penyakit-penyakit ini.

Penyakit lainnya, seperti Bacterial Spot, Late Blight, dan Septoria Leaf Spot, juga menunjukkan kinerja yang sangat baik dengan precision di atas 0.980 dan recall mendekati 1.000. F1-score untuk penyakit-penyakit ini juga cukup tinggi,

menunjukkan bahwa model dapat menjaga keseimbangan yang baik antara presisi dan recall dalam klasifikasi mereka.

Namun, ada sedikit penurunan dalam performa pada Early Blight, Spider Mites, dan Target Spot, di mana precision dan recall sedikit lebih rendah (sekitar 0.970 hingga 0.980), yang mengindikasikan adanya sedikit kesulitan dalam membedakan gejala antara penyakit-penyakit ini. Meskipun demikian, F1-score untuk ketiganya tetap cukup tinggi, dengan nilai sekitar 0.965 hingga 0.977, menunjukkan bahwa kesalahan klasifikasi tersebut tidak mengurangi performa model secara signifikan.

Secara keseluruhan, model menunjukkan akurasi yang sangat baik dengan nilai macro avg dan weighted avg sebesar 0.987 untuk precision, recall, dan F1-score, yang menunjukkan performa yang sangat konsisten di seluruh kelas. Meskipun ada beberapa area yang dapat diperbaiki, terutama dalam membedakan penyakit dengan gejala yang serupa, model secara keseluruhan mampu mengklasifikasikan penyakit daun tomat dengan sangat akurat.

4.5.3.3 Hasil Confusion Matrix

Confusion matrix yang ditampilkan pada gambar ini menunjukkan kinerja model dalam mengklasifikasikan penyakit daun dengan menggunakan data pengujian. Berikut adalah confusion matrixnya.



Gambar 4. 30 Gambar Confusion Matrix Skenario All Augmentation

Berdasarkan confusion matrix yang diberikan, model menunjukkan performa yang sangat baik dalam mengklasifikasikan sebagian besar penyakit daun tomat. Pada Bacterial Spot, semua 200 sampel terklasifikasikan dengan benar, yang menunjukkan bahwa model sangat efektif dalam mengenali penyakit ini. Untuk Early Blight, 192 sampel terklasifikasi dengan benar, tetapi ada 3 kesalahan yang salah diklasifikasikan sebagai Bacterial Spot, 1 sebagai Mosaic Virus, 1 sebagai Healthy, dan 2 sebagai Yellow Leaf Curl Virus. Kesalahan ini mungkin disebabkan oleh kemiripan gejala antara Early Blight dengan penyakit lain, terutama dengan Bacterial Spot dan Yellow Leaf Curl Virus.

Pada kategori Healthy, semua 200 sampel terklasifikasikan dengan benar, yang menunjukkan bahwa model sangat baik dalam mengklasifikasi daun sehat. Untuk Late Blight, meskipun sebagian besar sampel terklasifikasi dengan benar (195 benar), ada 5 kesalahan yang salah diklasifikasikan sebagai Bacterial Spot. Ini menunjukkan sedikit kebingungan antara Late Blight dan Bacterial Spot, yang keduanya menghasilkan bercak pada daun.

Pada Leaf Mold, model mengklasifikasikan 199 sampel dengan benar, namun ada 1 kesalahan yang salah diklasifikasikan sebagai Mosaic Virus, yang menunjukkan bahwa model bisa lebih baik dalam membedakan gejala antara kedua penyakit tersebut. Begitu juga dengan Mosaic Virus, yang semua 200 sampel terklasifikasi dengan benar, menunjukkan kemampuan model yang sangat baik dalam mengenali penyakit ini.

Pada Septoria Leaf Spot, 199 sampel terklasifikasikan dengan benar, dengan hanya 1 sampel salah diklasifikasikan sebagai Yellow Leaf Curl Virus. Untuk Spider Mites, meskipun sebagian besar sampel terklasifikasi dengan benar (195 benar), ada 4 kesalahan yang salah diklasifikasikan sebagai Septoria Leaf Spot. Pada Target Spot, model berhasil mengklasifikasikan 195 sampel dengan benar, dengan hanya 1 sampel salah diklasifikasikan sebagai Bacterial Spot, menunjukkan bahwa model dapat membedakan keduanya dengan baik meskipun ada sedikit kesalahan.

Terakhir, pada Yellow Leaf Curl Virus, semua 200 sampel terklasifikasikan dengan benar, menunjukkan performa yang sangat baik pada penyakit ini. Secara keseluruhan, model menunjukkan kinerja yang sangat baik dengan sebagian besar

sebagai indikator verifikasi (misalnya meninjau ulang prediksi pada rentang confidence menengah dan memeriksa kasus-kasus salah dengan confidence tinggi untuk mengetahui pasangan kelas yang paling sering tertukar).

Dalam visualisasi hasil prediksi dengan model YOLO11 pada skenario all augmentation, dapat melihat contoh gambar yang diprediksi dengan benar dan salah. Gambar 4. 32 adalah kumpulan gambar label prediksi skenario all augmentation.



Gambar 4. 32 Visualisasi Label Prediksi Skenario All Augmentation

Berdasarkan gambar yang diberikan, terdapat 7 kesalahan prediksi yang dapat dijelaskan melalui penggunaan berbagai teknik all augmentation, termasuk advanced mixing augmentation, photometric augmentation, dan spatial transformations. Meskipun teknik-teknik ini dirancang untuk meningkatkan keragaman data pelatihan dan membantu model menggeneralisasi dengan lebih baik, perubahan yang diterapkan pada gambar dapat mempengaruhi cara model mengklasifikasi gejala penyakit, terutama ketika gejala antara penyakit memiliki kesamaan visual yang sangat halus.

Pada gambar pertama, model berhasil mengklasifikasikan Early Blight dengan tepat, namun pada gambar kedua, model salah mengklasifikasikan Late Blight sebagai Early Blight. Kedua penyakit ini memiliki bercak pada daun, tetapi

Late Blight cenderung menghasilkan bercak yang lebih besar dan gelap. Penerapan spatial transformations, seperti rotasi atau pemotongan gambar, dapat menyebabkan perubahan orientasi bercak yang membuat model kesulitan membedakan keduanya. Pada gambar ketiga, model mengidentifikasi Leaf Mold dengan benar, namun pada gambar keempat, Septoria Leaf Spot salah diklasifikasikan sebagai Target Spot, yang menunjukkan bahwa perbedaan antara bercak kecil pada kedua penyakit ini bisa sangat sulit dikenali setelah rotasi atau pemotongan gambar.

Pada gambar kelima, model mengklasifikasikan Spider Mites dengan benar, namun pada gambar keenam, model salah mengklasifikasikan Target Spot sebagai Spider Mites. Meskipun keduanya menunjukkan bercak pada daun, Target Spot memiliki bercak yang lebih terdefinisi, sedangkan Spider Mites menghasilkan bintik kuning kecil. Perubahan saturasi atau kontras warna melalui photometric augmentation bisa memperburuk perbedaan ini, mengarah pada kebingungan dalam klasifikasi. Pada gambar ketujuh, model salah mengklasifikasikan Yellow Leaf Curl Virus sebagai Spider Mites. Yellow Leaf Curl Virus menyebabkan daun melengkung dan menguning, sedangkan Spider Mites lebih sering menyebabkan bintik kecil kuning. Peningkatan kontras atau saturasi warna pada gambar akibat photometric augmentation bisa membuat perubahan warna daun yang disebabkan oleh virus menjadi kabur, menyebabkan model salah dalam mengidentifikasi penyakit tersebut.

Secara keseluruhan, meskipun teknik augmentasi spasial dan fotometrik bertujuan untuk memperkaya data dan meningkatkan kemampuan model dalam

mengenali variasi dunia nyata, perubahan yang diterapkan pada gambar dapat menyebabkan kebingungan dalam membedakan penyakit dengan gejala yang mirip. Model dapat lebih dioptimalkan dengan penyesuaian teknik augmentasi yang lebih hati-hati dan memperbaiki kemampuan model dalam membedakan penyakit yang sangat mirip dengan gejala halus, agar kesalahan klasifikasi dapat diminimalkan.

4.6 Perbandingan Hasil

Dalam penelitian ini, dilakukan perbandingan terhadap berbagai teknik augmentasi data yang diterapkan pada model YOLO11 untuk mengklasifikasikan penyakit pada daun tomat.

4.6.1 Perbandingan Hasil dari 5 Skenario Augmentasi

Setiap teknik augmentasi yang diuji termasuk Non-AUG, Advanced Mixing Augmentation, Photometric Augmentation, Spatial Transformations, dan All Augmentation memiliki pengaruh yang berbeda terhadap kinerja model dalam hal akurasi dan daya tahan dalam berbagai kondisi pencahayaan dan sudut pandang. Melalui analisis hasil yang diperoleh dari masing-masing skenario, dapat dilihat bahwa penerapan teknik augmentasi data secara signifikan meningkatkan Top-1 Accuracy model YOLO11, serta memperkuat kemampuan model dalam menghadapi variasi kondisi yang kompleks. Hasil perbandingan ini memberikan gambaran tentang teknik augmentasi mana yang paling efektif dalam meningkatkan performa model untuk aplikasi klasifikasi penyakit tanaman. Tabel 4. 28 berisi hasil perbandingan hasil evaluasi dari 5 skenario penelitian ini.

Tabel 4. 28 Tabel Perbandingan Hasil Penelitian

Skenario	Accuracy	Presisi	Recall
Non Augmentation	0,977	0,986	0,986
Advanced Mixing Augmentation	0,98015	0,991	0,991
Photometric Augmentation	0,95425	0,964	0,963
Spatial Transformations	0,987250	0,994	0,994
All Augmentation	0,981	0,987	0,987

Berdasarkan hasil evaluasi yang diberikan untuk masing-masing teknik augmentasi, dapat dilihat bahwa setiap metode memberikan dampak yang berbeda terhadap kinerja model dalam mengklasifikasikan penyakit daun tomat. Pada skenario Non Augmentation, model menunjukkan akurasi yang baik (0.977) dengan precision dan recall yang sangat tinggi (0.986), namun kesulitan masih ada dalam membedakan penyakit dengan gejala yang mirip, seperti Bacterial Spot dan Target Spot atau Early Blight dan Late Blight. Hal ini menunjukkan bahwa tanpa augmentasi, model cukup efektif namun tetap membutuhkan pelatihan lebih lanjut dengan dataset yang lebih variatif untuk mengenali perbedaan halus antara penyakit yang serupa.

Pada skenario Advanced Mixing Augmentation, model menunjukkan sedikit peningkatan akurasi (0.98015) dan peningkatan precision serta recall (0.991), menandakan bahwa teknik ini berhasil memperkaya dataset pelatihan, membuat model lebih robust terhadap variasi dunia nyata. Namun, beberapa kesalahan klasifikasi masih terjadi, seperti kebingungan antara Bacterial Spot dan Yellow Leaf Curl Virus, yang menunjukkan bahwa meskipun augmentasi meningkatkan kemampuan model, perbedaan visual halus seperti warna atau bentuk bercak masih dapat menyebabkan kesalahan.

Sebaliknya, Photometric Augmentation menjadi teknik yang paling buruk dalam hal klasifikasi penyakit daun tomat berdasarkan hasil yang ditunjukkan (akurasi 0.95425, precision 0.964, recall 0.963). Photometric augmentation berfokus pada variasi dalam pencahayaan, kontras, dan saturasi warna, yang meskipun dapat meningkatkan keragaman data pelatihan, justru memperburuk perbedaan visual yang halus antara penyakit yang mirip. Pencahayaan yang tidak merata dan perubahan kontras yang terlalu besar dapat membuat perbedaan antara penyakit dengan gejala yang serupa menjadi lebih kabur. Misalnya, Bacterial Spot yang memiliki bercak kecil dengan batas kabur bisa dengan mudah tercampur dengan Yellow Leaf Curl Virus yang menyebabkan daun melengkung dan menguning. Teknik ini memperkenalkan tantangan dalam membedakan perubahan halus pada warna atau bentuk daun, seperti yang terjadi pada gambar yang salah mengklasifikasikan Early Blight sebagai Yellow Leaf Curl Virus.

Meskipun photometric augmentation dapat berguna untuk meningkatkan keragaman data dalam konteks gambar yang lebih luas, untuk penyakit dengan gejala halus atau yang sangat mirip, perubahan dalam pencahayaan dan saturasi warna justru memperburuk perbedaan visual. Inilah alasan mengapa photometric augmentation cenderung menghasilkan hasil yang lebih buruk, karena model menjadi kurang mampu membedakan gejala yang lebih halus akibat perubahan warna yang diterapkan oleh augmentasi.

Spatial Transformations menjadi teknik augmentasi yang paling efektif dalam klasifikasi penyakit daun tomat karena kemampuan teknik ini untuk memperkenalkan variasi geometris dalam dataset, seperti rotasi, pemotongan, dan

perubahan ukuran gambar. Teknik ini meningkatkan kemampuan model untuk mengenali pola dan gejala penyakit dalam berbagai orientasi dan ukuran daun, yang sangat relevan ketika penyakit dapat muncul dalam berbagai sudut pandang atau kondisi visual. Dengan *spatial transformations*, model menjadi lebih robust terhadap perubahan pada posisi atau bentuk daun, yang sangat berguna dalam konteks pengklasifikasian penyakit yang memiliki pola bercak yang mungkin tergantung pada sudut atau ukuran daun yang berbeda. Hal ini tercermin dalam hasil evaluasi, di mana *spatial transformations* menunjukkan akurasi yang sangat tinggi (0.987250), dengan *precision* dan *recall* yang mencapai 0.994, menunjukkan bahwa teknik ini tidak hanya memperbaiki akurasi model, tetapi juga meningkatkan sensitivitasnya dalam mengklasifikasi perbedaan halus antara penyakit yang sangat mirip, seperti Spider Mites dan Target Spot.

Selain itu, teknik *spatial transformations* juga memiliki keunggulan dalam meningkatkan generalisasi model terhadap data dunia nyata, di mana variasi geometris seperti sudut pandang dan ukuran daun sangat sering terjadi. Itulah sebabnya banyak peneliti memilih *spatial transformations* sebagai teknik default dalam augmentasi, karena ia memberikan hasil yang stabil dan efektif dalam meningkatkan kinerja model tanpa mengorbankan akurasi dalam menghadapi variasi yang lebih kompleks.

Terakhir, pada skenario All Augmentation, yang menggabungkan berbagai teknik augmentasi, model menunjukkan akurasi yang sangat baik (0.981) dan *precision* serta *recall* yang tinggi (0.987). Gabungan teknik ini memberikan hasil yang baik dan membuat model lebih robust terhadap variasi data dunia nyata,

namun masih ada sedikit penurunan performa dibandingkan dengan spatial transformations saja. Kesalahan klasifikasi tetap terjadi, terutama pada penyakit dengan gejala yang sangat mirip, namun kombinasi augmentasi ini terbukti memperkuat model dalam mengenali gejala dengan lebih baik.

Secara keseluruhan, meskipun augmentasi membantu meningkatkan kemampuan model dalam mengklasifikasikan penyakit daun tomat, tantangan utama tetap pada kesulitan membedakan penyakit dengan gejala yang sangat mirip. Teknik spatial transformations memberikan hasil terbaik dalam hal akurasi, sementara photometric augmentation sedikit menurunkan kinerja model karena perubahan pencahayaan dan kontras. Sementara itu, advanced mixing augmentation dan all augmentation menunjukkan hasil yang baik namun masih perlu pelatihan lebih lanjut dengan data yang lebih variatif dan teknik augmentasi yang lebih tepat untuk meningkatkan kemampuan model dalam membedakan penyakit dengan gejala halus.

4.6.2 Perbandingan Hasil dengan Penelitian Terdahulu

Berikut adalah tabel perbandingan hasil dengan penelitian terdahulu yang menggunakan dataset yang sama.

Tabel 4. 29 Tabel Perbandingan Hasil dengan Penelitian Terdahulu

Penelitian	Metode	Akurasi	Presisi	Recall
Abdullah (2024)	YOLOv8s	-	0.932	0.903
	YOLOv5	-	0.901	0.871
	Faster R-CNN	-	0.768	0.817
Lestari et al. (2024)	DenseNet-PSO	0.9739	0.9747	0.9739
	DenseNet-121	0.9031	0.9262	0.9031
	InceptionV3	0.9169	0.9318	0.9169
Kumar et al. (2024)	Ensemble Model (VGG-16, NASNet, EMA, EWGO)	0.987	0.986	0.979
	CNN (VGG16)	0.91	-	-

Marchel Adias Pradana et al. (2023)	CNN (VGG16 + augmentasi standar)	0.92	-	-
Andi Nurdin et al. (2024)	CNN (Inception V3) + Optimizer Adam	0.938	0.939	0.938
Penelitian ini (Non Augmentation)	YOLO11n-cls	0,977	0.986	0.986
Penelitian ini (Advanced Mixing Augmentation)	YOLO11n-cls + Advanced Mixing Augmentation	0.98015	0.991	0.991
Penelitian ini (Photometric Augmentation)	YOLO11n-cls + Photometric Augmentation	0.95425	0.964	0.963
Penelitian ini (Spatial Transformations)	YOLO11n-cls + Spatial Transformations	0.98725	0.994	0.994
Penelitian ini (All Augmentation)	YOLO11n-cls + Advanced, Photometric, Spatial Augmentation	0.981	0.987	0.987

Penelitian Abdullah (2024) mengevaluasi beberapa metode deteksi berbasis deep learning, yaitu YOLOv8s, YOLOv5, dan Faster R-CNN, dengan pelaporan metrik presisi dan recall. YOLOv8s menunjukkan kinerja paling tinggi pada studi tersebut dengan presisi 0.932 dan recall 0.903, lebih baik dibanding YOLOv5 (presisi 0.901; recall 0.871) dan Faster R-CNN (presisi 0.768; recall 0.817). Pola ini mengindikasikan bahwa arsitektur YOLO generasi terbaru cenderung lebih efektif dalam menjaga keseimbangan ketepatan deteksi (presisi) dan kelengkapan temuan (recall) dibanding pendekatan two-stage seperti Faster R-CNN pada konteks data yang diuji oleh Abdullah.

Lestari et al. (2024) membandingkan DenseNet-PSO dengan DenseNet-121 dan InceptionV3, dan hasilnya menunjukkan bahwa DenseNet-PSO unggul signifikan dengan akurasi 0.9739, presisi 0.9747, dan recall 0.9739. Sementara itu,

DenseNet-121 dan InceptionV3 menghasilkan akurasi lebih rendah (0.9031 dan 0.9169) dengan presisi masing-masing 0.9262 dan 0.9318. Temuan ini menegaskan bahwa optimisasi berbasis PSO pada DenseNet mampu meningkatkan kualitas representasi fitur sehingga performa meningkat secara konsisten pada ketiga metrik utama.

Kumar et al. (2024) menggunakan pendekatan ensemble yang menggabungkan VGG-16, NASNet, dan strategi optimisasi/penimbangan (EMA, EWGO), serta memperoleh akurasi 0.987, presisi 0.986, dan recall 0.979. Hasil ini menunjukkan bahwa ensemble dapat meningkatkan stabilitas dan akurasi prediksi karena mengombinasikan kekuatan beberapa model, namun recall yang sedikit lebih rendah dibanding akurasi dan presisi mengindikasikan masih terdapat sebagian kasus yang terlewat (false negative) meskipun prediksi yang dihasilkan cenderung tepat.

Marchel Adias Pradana et al. (2023) melaporkan performa CNN berbasis VGG16 dengan akurasi 0.91, kemudian meningkat menjadi 0.92 setelah menerapkan augmentasi standar. Meskipun presisi dan recall tidak dilaporkan, peningkatan akurasi ini menunjukkan bahwa augmentasi dasar dapat memperkaya variasi data latih dan membantu model lebih general, walau dampaknya relatif moderat dibanding teknik optimisasi atau ensemble yang lebih kompleks.

Andi Nurdin et al. (2024) menerapkan CNN berbasis Inception V3 dengan optimizer Adam dan melaporkan akurasi 0.938, presisi 0.939, serta recall 0.938. Keseimbangan nilai presisi dan recall yang hampir identik menunjukkan model cenderung stabil dan tidak terlalu bias pada false positive maupun false negative,

namun performanya masih berada di bawah pendekatan yang menggunakan optimisasi lanjutan (misalnya DenseNet-PSO) atau strategi ensemble.

Pada skenario Non Augmentation, model menunjukkan akurasi yang baik (0.977) dengan precision dan recall yang tinggi (0.986), namun masih mengalami kesulitan dalam membedakan penyakit dengan gejala mirip, seperti Bacterial Spot dan Target Spot, atau Early Blight dan Late Blight. Hal ini menunjukkan bahwa meskipun tanpa augmentasi model cukup efektif, ia masih memerlukan pelatihan lebih lanjut dengan dataset yang lebih variatif untuk mengenali perbedaan halus antara penyakit yang serupa.

Pada skenario Advanced Mixing Augmentation, akurasi sedikit meningkat (0.98015) dengan precision dan recall yang lebih tinggi (0.991), yang menunjukkan bahwa augmentasi ini memperkaya dataset pelatihan, membuat model lebih robust terhadap variasi data. Namun, kesalahan klasifikasi masih terjadi, seperti kebingungan antara Bacterial Spot dan Yellow Leaf Curl Virus. Ini menunjukkan bahwa meskipun augmentasi meningkatkan kemampuan model, perbedaan visual halus seperti warna atau bentuk bercak tetap bisa menyebabkan kesalahan.

Pada skenario Photometric Augmentation, terjadi penurunan dalam akurasi (0.95425) serta precision dan recall (0.964 dan 0.963). Variasi dalam pencahayaan, kontras, dan saturasi warna membuat perbedaan visual antara penyakit yang mirip menjadi kabur. Contohnya, Bacterial Spot yang salah diklasifikasikan sebagai Yellow Leaf Curl Virus menunjukkan bagaimana perubahan dalam pencahayaan dan kontras dapat memperburuk kesulitan model dalam membedakan penyakit dengan gejala mirip.

Pada skenario Spatial Transformations, teknik ini memberikan kinerja terbaik dengan akurasi tinggi (0.987250) serta precision dan recall yang sangat baik (0.994). Transformasi spasial seperti rotasi, pemotongan, dan perubahan ukuran gambar memungkinkan model untuk menggeneralisasi lebih baik terhadap variasi geometris dalam data. Meskipun demikian, model masih menghadapi kesulitan dalam membedakan penyakit dengan gejala mirip, seperti Spider Mites yang salah diklasifikasikan sebagai Target Spot. Namun, secara keseluruhan, teknik ini meningkatkan kinerja model secara signifikan.

Terakhir, pada skenario All Augmentation, yang menggabungkan berbagai teknik augmentasi, model menunjukkan akurasi yang sangat baik (0.981) dengan precision dan recall yang tinggi (0.987). Kombinasi dari semua teknik ini meningkatkan kemampuan model dalam mengenali variasi data dunia nyata, meskipun ada sedikit penurunan kinerja dibandingkan dengan spatial transformations saja. Kesalahan klasifikasi tetap ada, terutama pada penyakit dengan gejala yang sangat mirip, namun penggunaan semua teknik augmentasi ini membantu model menjadi lebih robust dalam klasifikasi penyakit daun tomat.

BAB 5 PENUTUP

5.1 Kesimpulan

Kesimpulan dari penelitian ini adalah sebagai berikut.

1. Penerapan teknik augmentasi data mempengaruhi akurasi model YOLO11 secara signifikan. Spatial transformations memberikan hasil terbaik dengan akurasi 0.987250, sementara all augmentation mencapai 0.981. Advanced mixing augmentation meningkatkan akurasi menjadi 0.98015, sedangkan photometric augmentation menurunkan akurasi menjadi 0.95425 akibat perubahan pencahayaan dan kontras yang mengaburkan perbedaan penyakit. Secara keseluruhan, spatial transformations memberikan kinerja terbaik.
2. Spatial transformations adalah teknik terbaik, memberikan akurasi 0.987250, precision 0.994, dan recall 0.994. Teknik ini membantu model menggeneralisasi lebih baik terhadap variasi geometris seperti rotasi dan pemotongan gambar. Photometric augmentation malah menurunkan kinerja model karena variasi pencahayaan dan kontras yang memperburuk perbedaan gejala.
3. Spatial transformations meningkatkan daya tahan model terhadap variasi geometris, seperti perubahan sudut pandang dan ukuran daun, yang sering terjadi dalam kondisi dunia nyata. Sebaliknya, photometric augmentation memperkenalkan tantangan dengan pencahayaan dan kontras yang

mengaburkan perbedaan halus antara penyakit, sehingga tidak seefektif spatial transformations dalam meningkatkan daya tahan model terhadap kondisi dunia nyata.

5.2 Saran

Berdasarkan hasil yang diperoleh, ada beberapa aspek yang masih dapat ditingkatkan untuk memperkaya dan mengoptimalkan kinerja model dalam mengklasifikasi penyakit pada daun tomat. Dengan demikian, beberapa langkah yang dapat diambil untuk meningkatkan hasil penelitian ini adalah sebagai berikut:

- 1) Mengkombinasikan teknik augmentasi lain untuk menghadapi kondisi ekstrem seperti perubahan cuaca dan variasi lingkungan.
- 2) Perlu dilakukan pengujian dengan dataset yang lebih beragam untuk meningkatkan keandalan model dalam kondisi nyata.
- 3) Disarankan untuk mengintegrasikan YOLO11 dalam sistem deteksi berbasis waktu nyata untuk pemantauan otomatis tanaman.
- 4) Diperlukan analisis lebih dalam mengenai pengaruh Paramater tuning untuk optimasi kinerja model.
- 5) Penelitian lebih lanjut perlu dilakukan untuk memahami pengaruh variasi sudut pandang dan pencahayaan terhadap kinerja klasifikasi penyakit pada daun tomat.

DAFTAR PUSTAKA

- [1] M. Y. Ali *et al.*, "Nutritional composition and bioactive compounds in tomatoes and their impact on human health and disease: A review," *Foods*, vol. 10, no. 1, Jan. 2021, doi: 10.3390/foods10010045.
- [2] V. Cruz-López *et al.*, *Impact of Different Agricultural Management Practices on Soil Fertility in Tomato Crops Under Greenhouse Conditions*, 2024. doi: 10.2139/ssrn.4963666.
- [3] Oscar Cyrille Adantchédé Akotowanou, Euloge Sènan Adjou, Adéyèmi Berane Olubi, Sylvain Daton Kougblenou, Edwige Dahouenon Ahoussi, and Dominique CK Sohounhloué, "The tomato (*Solanum lycopersicum* L.) in community development: An overview focused on nutritional properties, agronomic constraints, reThe tomato (*Solanum lycopersicum* L.) in community development: An overview focusent achievements and future prospective," *International Journal of Frontiers in Biology and Pharmacy Research*, vol. 3, no. 2, pp. 008–016, Nov. 2022, doi: 10.53294/ijfbpr.2022.3.2.0061.
- [4] J. K. Kokate, S. Kumar, and A. G. Kulkarni, "Classification of Tomato Leaf Disease Using a Custom Convolutional Neural Network," *Current Agriculture Research Journal*, vol. 11, no. 1, pp. 316–325, May 2023, doi: 10.12944/carj.11.1.28.
- [5] Badan Pusat Statistik, "Produksi Tanaman Sayuran Menurut Provinsi dan Jenis Tanaman 2024," Feb. 06, 2024. Accessed: Apr. 28, 2025. [Online]. Available: <https://www.bps.go.id/assets/statistics-table/3/ZUhFd1JtZzJWVVpqWTJsV05XTIlhVmhRSzFoNFFUMDkjMw%3D%3D/produksi-tanaman-sayuran-dan-buah-buahan-semusim-menurut-provinsi-dan-jenis-tanaman---2024.html>
- [6] dan P. K. B. Dinas Koperasi, Perindustrian, "Informasi Harga Pasar 06 Januari 2025," Dinas Koperasi, Perindustrian, dan Perdagangan Kabupaten Bondowoso. Accessed: Apr. 28, 2025. [Online]. Available:

<https://diskoperindag.bondowosokab.go.id/berita/informasi-harga-pasar-06-januari-2025>

- [7] H. M. Zayani *et al.*, "Deep Learning for Tomato Disease Detection with YOLOv8," *Engineering, Technology and Applied Science Research*, vol. 14, no. 2, pp. 13584–13591, Apr. 2024, doi: 10.48084/etasr.7064.
- [8] F. Fistranda, F. Austin, and M. Sadrawi, "Comparison Analysis of YOLO, ResNet, and EfficientNet for contagious skin diseases," in *2023 5th International Conference on Cybernetics and Intelligent System (ICORIS)*, IEEE, Oct. 2023, pp. 1–5, doi: 10.1109/ICORIS60118.2023.10352306.
- [9] M. Magdin and Z. Balogh, "Comparison classification algorithms and the YOLO method for video analysis and object detection," *Sci. Rep.*, vol. 15, no. 1, p. 25432, Jul. 2025, doi: 10.1038/s41598-025-09814-1.
- [10] X. Liu, "Comparison of Four Convolutional Neural Network-Based Algorithms for Sports Image Classification," *International Conference on Data Scir*, 2024, pp. 178–186, doi: 10.2991/978-94-6463-370-2_20.
- [11] P. Enkvetchakul and O. Surinta, "Effective Data Augmentation and Training Techniques for Improving Deep Learning in Plant Leaf Disease Recognition," *Applied Science and Engineering Progress*, vol. 15, no. 3, Jul. 2022, doi: 10.14416/j.asep.2021.01.003.
- [12] B. Liu, C. Tan, S. Li, J. He, and H. Wang, "A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification," *IEEE Access*, vol. 8, pp. 102188–102198, 2020, doi: 10.1109/ACCESS.2020.2998839.
- [13] F. M. Firnando, D. R. I. M. Setiadi, A. R. Muslikh, and S. W. Iriananda, "Analyzing InceptionV3 and InceptionResNetV2 with Data Augmentation for Rice Leaf Disease Classification," *Journal of Future Artificial Intelligence and Technologies*, vol. 1, no. 1, pp. 1–11, May 2024, doi: 10.62411/faith.2024-4.

- [14] M. Sohan, T. Ram, and V. Ch, "A Review on YOLOv8 and Its Advancements," 2024, pp. 529–545. doi: 10.1007/978-981-99-7962-2_39.
- [15] M. Talib, A. H. Y. Al-Noori, and J. Suad, "YOLOv8-CAB: Improved YOLOv8 for Real-time Object Detection," *Karbala International Journal of Modern Science*, vol. 10, no. 1, pp. 56–68, 2024, doi: 10.33640/2405-609X.3339.
- [16] A. Abdullah, G. A. Amran, S. M. A. Tahmid, A. Alabrah, A. A. AL-Bakhrani, and A. Ali, "A Deep-Learning-Based Model for the Detection of Diseased Tomato Leaves," *Agronomy*, vol. 14, no. 7, Jul. 2024, doi: 10.3390/agronomy14071593.
- [17] C. A. D. Lestari, S. Anam, and U. Sa'adah, "Exploring DenseNet architectures with particle swarm optimization: efficient tomato leaf disease detection," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 14, no. 2, p. 1377, Apr. 2025, doi: 10.11591/ijai.v14.i2.pp1377-1385.
- [18] A. Nurdin, D. Satria, Y. Kartika, A. Rezha, and E. Najaf, "Klasifikasi Penyakit Daun Tomat Dengan Metode Convolutional Neural Network Menggunakan Arsitektur Inception-V3," 2024.
- [19] P. V *et al.*, "Improved tomato leaf disease classification through adaptive ensemble models with exponential moving average fusion and enhanced weighted gradient optimization," *Front. Plant Sci.*, vol. 15, 2024, doi: 10.3389/fpls.2024.1382416.
- [20] M. Adias Pradana *et al.*, "Klasifikasi Penyakit Tanaman Tomat Menggunakan Metode Convolutional Neural Network (CNN) VGG16," *Jurnal Riset Inovasi Bidang Informatika dan Pendidikan Informatika*, vol. 4, no. 2, 2023, doi: 10.31284/j.kernel.2023.v4i2.6829.
- [21] I. Topan Adib Amrulloh *et al.*, "EVALUASI AUGMENTASI DATA PADA DETEKSI PENYAKIT DAUN TEBU DENGAN YOLOV8," 2024.

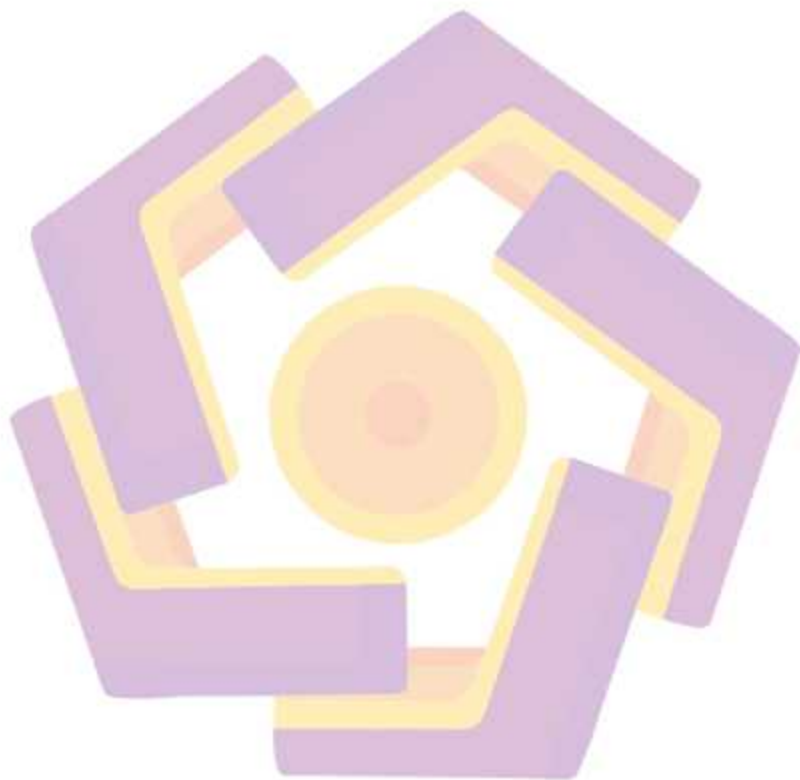
- [22] T. Bayu Sasongko and A. Amrullah, "ANALISIS EFEK AUGMENTASI DATASET DAN FINE TUNE PADA ALGORITMA PRE-TRAINED CONVOLUTIONAL NEURAL NETWORK (CNN)," *JTIK*, vol. 10, no. 4, pp. 763–768, 2023, doi: 10.25126/jtiik.2023106583.
- [23] PATMAWATI, A. Sunyoto, and E. Luthfi, "AUGMENTASI DATA MENGGUNAKAN DCGAN PADA GAMBAR TANAH," *TEKNIMEDIA: Teknologi Informasi dan Multimedia*, vol. 4, pp. 42–45, Jun. 2023, doi: 10.46764/teknimedia.v4i1.100.
- [24] I. Athiyyah Rahma and L. Hulliyyatus Suadaa, "PENERAPAN TEXT AUGMENTATION UNTUK MENGATASI DATA YANG TIDAK SEIMBANG PADA KLASIFIKASI TEKS BERBAHASA INDONESIA STUDI KASUS: DETEKSI JUDUL CLICKBAIT DAN KOMENTAR HATE SPEECH PADA BERITA ONLINE," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 2023, doi: 10.25126/jtiik.2023107325.
- [25] B. Min, T. Kim, D. Shin, and D. Shin, "Data Augmentation Method for Plant Leaf Disease Recognition," *Applied Sciences (Switzerland)*, vol. 13, no. 3, Feb. 2023, doi: 10.3390/app13031465.
- [26] J. Sanjaya and M. Ayub, "Augmentasi Data Pengenalan Citra Mobil Menggunakan Pendekatan Random Crop, Rotate, dan Mixup," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, Aug. 2020, doi: 10.28932/jutisi.v6i2.2688.
- [27] T. Kumar, A. Mileo, R. Brennan, and M. Bendeche, "RSMDA: Random Slices Mixing Data Augmentation," *Applied Sciences*, vol. 13, no. 3, 2023, doi: 10.3390/app13031711.
- [28] T. Kumar, M. Turab, K. Raj, A. Mileo, R. Brennan, and M. Bendeche, "Image Data Augmentation Approaches: A Comprehensive Survey and Future directions," *arXiv (Cornell University)*, Nov. 2023, doi: 10.48550/arxiv.2301.02830.

- [29] C. Jin, S. Qiu, N. Xiao, and H. Jia, "AdMIX: a mixed sample data augmentation method for neural machine translation," *arXiv (Cornell University)*, Nov. 2022, doi: 10.48550/arxiv.2205.04686.
- [30] D. Lewy and J. Mańdziuk, "An overview of mixing augmentation methods and augmentation strategies," *Artif. Intell. Rev.*, vol. 56, no. 3, pp. 2111–2169, 2023, doi: 10.1007/s10462-022-10227-z.
- [31] K. Rais, M. Amroune, and M. Y. Haouam, "Enhancing Medical Image Analysis through Geometric and Photometric transformations," *arXiv (Cornell University)*, Nov. 2025, doi: 10.48550/arxiv.2501.13643.
- [32] C. Gormond, "Influence of Photometric Data Augmentation," *Researchgate*, Sep. 2021.
- [33] M. Fanfani, M. Marulli, and P. Nesi, "Evaluation of Geometric and Photometric Data Augmentation for Pedestrian Detection with Thermal Cameras," 2024, pp. 353–366. doi: 10.1007/978-3-031-65318-6_24.
- [34] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," *Array*, vol. 16, p. 100258, Nov. 2022, doi: 10.1016/j.array.2022.100258.
- [35] E. K. Susanto, H. Tjandrasa, and C. Fatichah, "Data Augmentation using Spatial Transformation for Brain Tumor Segmentation Improvement," in *2024 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2024, pp. 639–644, doi: 10.1109/ISITIA63062.2024.10668391.
- [36] Z. Shaghaghian, H. Burte, D. Song, and W. Yan, "An augmented reality application and experiment for understanding and learning spatial transformation matrices," *Virtual Real.*, vol. 28, no. 1, p. 12, Mar. 2024, doi: 10.1007/s10055-023-00904-x.
- [37] Q. Wu, Y. Chen, and J. Meng, "Dcgan-based data augmentation for tomato leaf disease identification," *IEEE Access*, vol. 8, pp. 98716–98728, 2020, doi: 10.1109/ACCESS.2020.2997001.

- [38] L. Li and M. Spratling, "DATA AUGMENTATION ALONE CAN IMPROVE ADVER-SARIAL TRAINING," 2023. [Online]. Available: <https://github.com/TreeLLi/DA-Alone-Improves-AT>.
- [39] Md. Y. Ali *et al.*, "Nutritional Composition and Bioactive Compounds in Tomatoes and Their Impact on Human Health and Disease: A Review," *Foods*, vol. 10, p. 45, Dec. 2020, doi: 10.3390/foods10010045.
- [40] S. Sivagami and S. Mohanapriya, "Tomato leaf disease detection using image processing technique," *International Journal of Agricultural Technology 2021Vol*, vol. 17, no. 3, pp. 1135–1146, 2021.
- [41] M. Altalak, M. Uddin, A. Alajmi, and A. Rizg, "A Hybrid Approach for the Detection and Classification of Tomato Leaf Diseases," *Applied Sciences*, p., 2022, doi: 10.3390/app12168182.
- [42] S. Kumar, S. Pal, V. P. Singh, and P. Jaiswal, "Performance evaluation of ResNet model for classification of tomato plant disease," *Epidemiol. Methods*, vol. 12, no. 1, 2023, doi: doi:10.1515/em-2021-0044.
- [43] H. Tian, T. Wang, Y. Liu, X. Qiao, and Y. Li, "Computer vision technology in agricultural automation —A review," Mar. 01. 2020, *China Agricultural University*, doi: 10.1016/j.inpa.2019.09.006.
- [44] S. Ghazal, A. Munir, and W. S. Qureshi, "Computer vision in smart agriculture and precision farming: Techniques and applications," Sep. 01, 2024, *KeAi Communications Co.* doi: 10.1016/j.jaiia.2024.06.004.
- [45] B. Ding, "Cassava Leaf Disease Classification Based On CNN," 2022.
- [46] R. Soekarta, M. Yusuf, Muh. F. Hasa, and N. A. Basri, "IMPLEMENTASI DEEP LEARNING UNTUK DETEKSI JENIS OBAT MENGGUNAKAN ALGORITMA CNN BERBASIS WEBSITE," *JIKA (Jurnal Informatika)*, vol. 7, no. 4, p. 455, Nov. 2023, doi: 10.31000/jika.v7i4.9751.

- [47] X. Du, Y. Sun, Y. Song, H. Sun, and L. Yang, "A Comparative Study of Different CNN Models and Transfer Learning Effect for Underwater Object Classification in Side-Scan Sonar Images," *Remote Sens. (Basel)*, vol. 15, no. 3, Feb. 2023, doi: 10.3390/rs15030593.
- [48] R. Sapkota, Z. Meng, M. Churuvija, X. Du, Z. Ma, and M. Karkee, "Comprehensive Performance Evaluation of YOLOv12, YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments," *ResearchGate*, Jul. 2024, [Online]. Available: <http://arxiv.org/abs/2407.12040>
- [49] G. Rasyid and J. Sutopo, "Comparison of YOLO-v8 and YOLO-v10 in Detecting Human Facial Emotions," *Indonesian Journal of Applied Technology*, vol. 2, no. 1, Dec. 2024, doi: 10.47134/ijat.v2i1.3395.
- [50] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," *ArXiv*, Oct. 2024, [Online]. Available: <http://arxiv.org/abs/2410.17725>
- [51] A. Sugiharto and R. Kusumaningrum, "Automated Detection of Driver and Passenger Without Seat Belt using YOLOv8." [Online]. Available: www.ijacsa.thesai.org
- [52] S. B. Cho, Y. Cheng, and S. Sul, "Enhanced classification performance through GauGAN-based data augmentation for tomato leaf images," *IET Image Process.*, Dec. 2024, doi: 10.1049/ipr2.13069.
- [53] A. Sugiharto and R. Kusumaningrum, "Automated Detection of Driver and Passenger Without Seat Belt using YOLOv8," 2023. [Online]. Available: www.ijacsa.thesai.org
- [54] S. Padmanabhuni and P. Gera, "Synthetic Data Augmentation of Tomato Plant Leaf using Meta Intelligent Generative Adversarial Network: Milgan," 2022. [Online]. Available: www.ijacsa.thesai.org

- [55] M. Pascual-González, A. Jiménez-Partinen, E. J. Palomo, E. López-Rubio, and A. Ortega-Gómez, "Hyperparameter optimization of YOLO models for invasive coronary angiography lesion detection and assessment," *Comput. Biol. Med.*, vol. 196, p. 110697, 2025, doi: <https://doi.org/10.1016/j.combiomed.2025.110697>.
- [56] A. Karaman *et al.*, "Robust real-time polyp detection system design based on YOLO algorithms by optimizing activation functions and hyperparameters with artificial bee colony (ABC)," *Expert Syst. Appl.*, vol. 221, p. 119741, 2023, doi: <https://doi.org/10.1016/j.eswa.2023.119741>.
- [57] P. N. Srinivasu, G. L. A. Kumari, S. C. Narahari, S. Ahmed, and A. Alhumam, "Exploring the impact of hyperparameter and data augmentation in YOLO V10 for accurate bone fracture detection from X-ray images," *Sci. Rep.*, vol. 15, no. 1, p. 9828, 2025, doi: [10.1038/s41598-025-93505-4](https://doi.org/10.1038/s41598-025-93505-4).
- [58] D. Wahyudi, I. Soesanti, and H. A. Nugroho, "Optimizing Hyperparameters of YOLO to Improve Performance of Brain Tumor Detection in MRI Images," in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, 2023, pp. 413–418. doi: [10.1109/ICOIACT59844.2023.10455813](https://doi.org/10.1109/ICOIACT59844.2023.10455813).
- [59] X. Wang, Z. Zhang, and X. Shang, "Research on Improved YOLO11 for Detecting Small Targets in Sonar Images Based on Data Enhancement," *Applied Sciences (Switzerland)*, vol. 15, no. 12, Jun. 2025, doi: [10.3390/app15126919](https://doi.org/10.3390/app15126919).
- [60] I. S. Isa, M. S. A. Rosli, U. K. Yusof, M. I. F. Maruzuki, and S. N. Sulaiman, "Optimizing the Hyperparameter Tuning of YOLOv5 for Underwater Detection," *IEEE Access*, vol. 10, pp. 52818–52831, 2022, doi: [10.1109/ACCESS.2022.3174583](https://doi.org/10.1109/ACCESS.2022.3174583).
- [61] R. Rajamohanam and B. C. Latha, "An Optimized YOLO v5 Model for Tomato Leaf Disease Classification with Field Dataset," *Engineering*,



LAMPIRAN

Lampiran 1 Dataset

