

**KLASIFIKASI PENYAKIT PADA TANAMAN JAGUNG BERDASARKAN
CITRA DAUN MENGGUNAKAN VGG16 DENGAN
*HYPERPARAMETER TUNING***

SKRIPSI

Oleh :
UMI FARZAH NADLIROH
NIM. 220605110186



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

**KLASIFIKASI PENYAKIT PADA TANAMAN JAGUNG BERDASARKAN
CITRA DAUN MENGGUNAKAN VGG16 DENGAN
*HYPERPARAMETER TUNING***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
UMI FARZAH NADLIROH
NIM. 220605110186

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

HALAMAN PERSETUJUAN

KLASIFIKASI PENYAKIT PADA TANAMAN JAGUNG BERDASARKAN
CITRA DAUN MENGGUNAKAN VGG16 DENGAN
HYPERPARAMETER TUNING

SKRIPSI

Oleh :
UMI FARZAH NADLIROH
NIM. 220605110186

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 12 Mei 2026

Pembimbing I,



Prof. Dr. Suhartono, M.Kom
NIP. 19680519 200312 1 001

Pembimbing II,



Prof. Dr. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Supriyono, M. Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

KLASIFIKASI PENYAKIT PADA TANAMAN JAGUNG BERDASARKAN CITRA DAUN MENGGUNAKAN VGG16 DENGAN HYPERPARAMETER TUNING

SKRIPSI

Oleh :

UMI FARZAH NADLIROH
NIM. 220605110186

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 19 Juni 2026

Susunan Dewan Penguji

Ketua Penguji	: <u>Dr. Irwan Budi Santoso, M.Kom</u> NIP. 19770103 201101 1 004	()
Anggota Penguji I	: <u>A'la Syaqui, M.Kom</u> NIP. 19771201 200801 1 007	()
Anggota Penguji II	: <u>Prof. Dr. Suhartono, M.Kom</u> NIP. 19680519 200312 1 001	()
Anggota Penguji III	: <u>Prof. Dr. Muhammad Faisal, M.T</u> NIP. 19740510 200501 1 007	()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Sumarto, M. Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Umi Farzah Nadliroh
NIM : 220605110186
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Klasifikasi Penyakit pada Tanaman Jagung
Berdasarkan Citra Daun Menggunakan VGG16
dengan Hyperparameter Tuning

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 22 Juni 2026

Yang membuat pernyataan,



Umi Farzah Nadliroh
NIM.220605110186

MOTTO

“عِشْ مَا شِئْتَ فَإِنَّكَ مَيِّتٌ وَأَحْبِبْ مَنْ أَحَبَبْتَ فَإِنَّكَ مُفَارِقُهُ وَأَعْمَلْ مَا شِئْتَ فَإِنَّكَ لَأَقْبِيهِ”

HALAMAN PERSEMBAHAN

Alhamdulillah rabbil ‘alamin puji syukur kepada Allah *Subhānahu wa Ta ‘ālā* karena berkat rahmat dan petunjuk-Nya penulis dapat menyelesaikan skripsi ini. Shalawat serta salam selalu tercurahkan kepada Rasulullah *Ṣallallāhu ‘Alaihi Wasallam* yang telah membawa kita dari zaman jahiliyah menuju *addinul Islam*. Skripsi ini penulis persembahkan kepada seluruh pihak yang telah berperan dan berjasa dalam perjalanan penelitian ini.

Pertama, kepada diri sendiri. Terima kasih telah bertahan melewati setiap bagian dari perjalanan ini termasuk bagian yang tidak pernah diceritakan kepada siapapun. Tidak semua perjuangan terlihat, namun semua itu nyata dan layak untuk dihargai. Karya ini adalah bukti bahwa penulis mampu menyelesaikan apa yang telah dimulai meskipun jalannya tidak selalu mudah.

Kedua, kepada satu-satunya kakak laki-laki penulis. Tidak ada untaian kata yang kiranya mampu membalas seluruh pengorbanan yang telah diberikan. Merupakan bentuk cinta yang paling tulus yang pernah penulis rasakan, ia memilih mengubur mimpinya agar mimpi adiknya bisa berdiri dan tumbuh. Tanpa kerja keras dan keikhlasannya, nama penulis tidak akan pernah ada di bangku perkuliahan ini.

Ketiga, kepada mereka yang hadir dalam perjalanan ini dengan caranya masing-masing termasuk mereka yang tanpa disadari mengajarkan ketangguhan melalui luka yang ditinggalkan. Dan kepada sahabat-sahabat seperjuangan serta

seluruh teman angkatan 2022, terima kasih telah menjadi tempat bernaung yang hangat di tengah perjalanan panjang yang tidak selalu berjalan sesuai rencana.

Semoga segala kebaikan dan bantuan yang telah diberikan mendapatkan balasan terbaik dari Allah *Subhānahu wa Ta'ālā. Aamiin.*

KATA PENGANTAR

Segala puji hanya milik Allah *Subhānahu wa Ta‘ālā* atas segala nikmat dan kasih sayang-Nya yang telah memudahkan penulis untuk menyelesaikan skripsi yang berjudul “KLASIFIKASI PENYAKIT PADA TANAMAN JAGUNG BERDASARKAN CITRA DAUN MENGGUNAKAN VGG16 DENGAN HYPERPARAMETER TUNING”. Semoga shalawat dan salam senantiasa terlimpah kepada Nabi Muhammad *Ṣallallāhu ‘Alaihi Wasallam*. Dan semoga kita semua mendapat syafaatnya di hari kiamat nanti, *Aamiin*. Penulis mengucapkan rasa syukur dan terima kasih yang tak terhingga kepada semua pihak-pihak yang selalu memberikan bantuan dan motivasi kepada penulis untuk dapat menyelesaikan skripsi ini. Ucapan ini penulis sampaikan kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP, selaku Rektor Universitas Islam Negeri Malang.
2. Dr. Agus Mulyono, M.Kes, selaku Dekan Fakultas Saintek Universitas Islam Negeri Malang.
3. Supriyono, M.Kom., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Prof. Dr. Suhartono, S.Si, M.Kom. selaku Dosen Pembimbing I, dan Prof. Dr. Muhammad Faisal, M.T. selaku Dosen Pembimbing II, atas segala bimbingan, arahan, ilmu, dan waktu yang telah diberikan dengan penuh kesabaran selama proses penyusunan skripsi ini.

5. Dr. Irwan Budi Santoso, M.Kom. selaku Ketua Penguji, dan A'la Syauqi, M.Kom. selaku Anggota Penguji, atas segala masukan, koreksi, dan saran yang membangun demi penyempurnaan skripsi ini.
6. Segenap Dosen, Admin, Laboran dan Mahasiswa Program Studi Teknik Informatika yang telah memberikan banyak dukungan dan bimbingan selama proses kuliah dan pengerjaan skripsi ini.
7. Orang tua penulis, atas doa dan dukungan yang senantiasa mengiringi perjalanan penulis.
8. Kakak laki-laki penulis, atas segala pengorbanan dan kerja keras yang menjadi fondasi terkuat bagi penulis untuk terus melangkah hingga titik ini.
9. Diri sendiri, atas segala ketekunan dan keberanian untuk terus bertahan dan menyelesaikan apa yang telah dimulai.

Malang, 9 Maret 2026

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TULISAN	Error! Bookmark not defined.
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR.....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
البحث مستخلص.....	xvii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	6
1.5 Manfaat Penelitian.....	6
BAB II STUDI PUSTAKA	7
2.1 Penelitian Terdahulu	7
2.2 Penyakit Daun Tanaman Jagung	11
2.2.1 Hawar Daun (<i>Northern Leaf Blight</i>).....	12
2.2.2 Bercak Daun (<i>Gray Leaf Spot</i>)	13
2.2.3 Karat Daun (<i>Common Rust</i>).....	13
2.2.4 Tanaman Sehat (<i>Healthy</i>).....	14
2.3 <i>Convolutional Neural Network</i> (CNN) dengan Arsitektur VGG16.....	14
2.4 Hyperparameter Tuning (Grid Search)	19
BAB III DESAIN DAN IMPLEMENTASI SISTEM	19
3.1 Desain Penelitian	19
3.2 Pengumpulan Data	21
3.3 Desain Sistem	24
3.4 <i>Preprocessing</i>	25
3.5 Experimental Setup	27
3.6 Evaluasi	31
BAB IV HASIL DAN PEMBAHASAN.....	35
4.1 Hasil <i>Preprocessing</i>	35
4.1.1 Hasil Resize dan Normalisasi Citra	35
4.1.2 Hasil Augmentasi Data Latih	36
4.2 Hasil Eksperimen	38
4.2.1 Eksperimen Rasio 50:50	39

4.2.2 Eksperimen Rasio 60:40	41
4.2.3 Eksperimen Rasio 70:30	44
4.2.4 Eksperimen Rasio 80:20	46
4.2.5 Eksperimen Rasio 90:10	48
4.3 Perbandingan Hasil Model	50
4.4 Pembahasan	56
4.5 Integrasi Sains dan Islam.....	62
4.5.1 Tujuan dan Manfaat Integrasi	64
BAB V KESIMPULAN DAN SARAN	69
5.1 Kesimpulan.....	69
5.2 Saran.....	71
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 2.1 Hawar Daun (Northern Leaf Blight)	12
Gambar 2.2 Bercak Daun (Gray Leaf Spot).....	13
Gambar 2.3 Karat Daun (Common Rust)	13
Gambar 2.4 Tanaman Sehat (Healthy)	14
Gambar 2.5 Arsitektur CNN VGG16.....	15
Gambar 2.6 Arsitektur Model CNN VGG16 yang Digunakan dalam Penelitian (Convolutional Base Frozen + Custom Classifier)	19
Gambar 2.7 Flowchart Proses Grid Search	21
Gambar 3.1 Desain Penelitian.....	21
Gambar 3.2 Desain Sistem Penelitian.....	25
Gambar 4.1 Perbandingan citra sebelum dan sesudah resize serta normalisasi....	36
Gambar 4.2 Contoh hasil augmentasi data latih (rotasi, zoom, dan horizontal flip).....	37
Gambar 4.3 Grafik training history rasio 50:50	39
Gambar 4.4 Confusion matrix rasio 50:50 - baseline & GS (kombinasi terbaik). 41	
Gambar 4.5 Grafik training history baseline rasio 60:40	42
Gambar 4.6 Confusion matrix rasio 60:40 - baseline & GS (kombinasi terbaik). 44	
Gambar 4.7 Grafik training history baseline rasio 70:30	44
Gambar 4.8 Confusion matrix rasio 70:30 - baseline & GS (kombinasi terbaik). 46	
Gambar 4.9 Grafik training history rasio baseline rasio 80:20	46
Gambar 4.10 Confusion matrix rasio 80:20 - baseline & GS (kombinasi terbaik)48	
Gambar 4.11 Grafik training history baseline rasio 90:10	49
Gambar 4.12 Confusion matrix rasio 90:10 - baseline & GS (kombinasi terbaik)50	
Gambar 4.13 Grafik perbandingan F1-Score Baseline vs Grid Search Terbaik pada seluruh rasio.....	54

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait Mengenai Klasifikasi CNN	10
Tabel 2.2 Deskripsi Layer Arsitektur VGG16.....	16
Tabel 2.3 Kandidat Kombinasi Hyperparameter dalam Grid Search.....	22
Tabel 2.4 Experimental Setup Model.....	24
Tabel 3.1 Contoh Sampel Gambar Dari Dataset Yang Digunakan	22
Tabel 3.2 Experimental Setup Model.....	31
Tabel 4.1 Hasil evaluasi model pada rasio 50:50	40
Tabel 4.2 Hasil evaluasi model pada rasio 60:40	42
Tabel 4.3 Hasil evaluasi model pada rasio 70:30	45
Tabel 4.4 Hasil evaluasi model pada rasio 80:20	47
Tabel 4.5 Hasil evaluasi model pada rasio 90:10	49
Tabel 4.6 Perbandingan Baseline & Grid Search pada seluruh rasio.....	51
Tabel 4.7 Rekap perbandingan Baseline & Grid Search Terbaik pada seluruh rasio.....	53
Tabel 4.8 Perbandingan hasil penelitian ini dengan penelitian terdahulu	60

ABSTRAK

Nadliroh, Umi Farzah. 2026. **Klasifikasi penyakit pada Tanaman Jagung Berdasarkan Citra Daun Menggunakan VGG16 dengan Hyperparameter Tuning.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Prof. Dr. Suhartono, S.Si, M.Kom. (II) Prof. Dr. Muhammad Faisal, M.T.

Kata Kunci: VGG16, Transfer Learning, Grid Search, Hyperparameter Tuning, Penyakit Daun Jagung.

Penelitian ini bertujuan menganalisis performa model klasifikasi berbasis arsitektur VGG16 dengan pendekatan transfer learning dalam mengklasifikasikan penyakit daun tanaman jagung ke dalam empat kelas, yaitu Gray Leaf Spot, Common Rust, Northern Leaf Blight, dan Healthy, serta menganalisis pengaruh hyperparameter tuning terhadap performa model pada lima variasi rasio pembagian data latih dan data uji. Metode penelitian mencakup prapemrosesan citra, pelatihan model baseline, serta hyperparameter tuning menggunakan Grid Search yang mengeksplorasi kombinasi tiga parameter, yaitu learning rate, batch size, dan dropout rate, menghasilkan delapan kombinasi pada setiap rasio sehingga total terdapat 45 skenario eksperimen. Hasil penelitian menunjukkan bahwa Grid Search berhasil meningkatkan performa model pada empat dari lima rasio yang diuji, dengan akurasi dan F1-score terbaik secara keseluruhan dicapai pada rasio 60:40 menggunakan kombinasi learning rate 0,0005, batch size 32, dan dropout rate 0,3, yaitu sebesar 93,67% dan 93,74%. Dropout rate 0,3 secara konsisten menjadi konfigurasi optimal pada seluruh rasio, sementara kelas Gray Leaf Spot secara konsisten menjadi kelas dengan performa terendah karena kemiripan visualnya dengan Northern Leaf Blight. Hyperparameter tuning secara sistematis terbukti lebih efektif dibandingkan penggunaan parameter default dalam meningkatkan performa klasifikasi penyakit daun jagung berbasis citra.

ABSTRACT

Nadliroh, Umi Farzah. 2026. **Classification of Diseases in Corn Plants Based on Leaf Images Using VGG16 with Hyperparameter Tuning**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University Malang. Advisors: (I) Prof. Dr. Suhartono, S.Si, M.Kom. (II) Prof. Dr. Muhammad Faisal, M.T.

Keywords: VGG16, Transfer Learning, Grid Search, Hyperparameter Tuning, Corn Leaf Disease.

This research aims to analyze the performance of a classification model based on the VGG16 architecture using a transfer learning approach to classify corn leaf diseases into four classes, namely Gray Leaf Spot, Common Rust, Northern Leaf Blight, and Healthy, as well as to analyze the effect of hyperparameter tuning on model performance across five variations of training-testing data split ratios. The research method includes image preprocessing, baseline model training, and hyperparameter tuning using Grid Search that explores combinations of three parameters, namely learning rate, batch size, and dropout rate, producing eight combinations for each ratio, resulting in a total of 45 experimental scenarios. The results show that Grid Search successfully improved model performance in four out of the five tested ratios, with the best overall accuracy and F1-score achieved at the 60:40 ratio using a combination of learning rate 0.0005, batch size 32, and dropout rate 0.3, reaching 93.67% and 93.74% respectively. A dropout rate of 0.3 consistently proved to be the optimal configuration across all ratios, while the Gray Leaf Spot class consistently showed the lowest performance due to its visual similarity to Northern Leaf Blight. Systematic hyperparameter tuning proved more effective than using default parameters in improving the performance of image-based corn leaf disease classification.

البحث مستخلص

نضيرة، أمي فرزه. ٢٠٢٦. تصنيف الأمراض في نبات الذرة بناءً على صور الأوراق باستخدام VGG16 مع ضبط المعاملات الفائقة (Hyperparameter Tuning). أطروحة. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية، مالانج. المشرفون (I) البروفيسور الدكتور سوهارتونو، (II) M.Kom البروفيسور الدكتور محمد فيصل، M.T.

الكلمات المفتاحية: VGG16، التعلم بالنقل، البحث الشبكي، ضبط المعاملات الفائقة، أمراض أوراق الذرة

تهدف هذه الدراسة إلى تحليل أداء نموذج تصنيف يعتمد على بنية VGG16 باستخدام أسلوب التعلم بالنقل (Transfer Learning) لتصنيف أمراض أوراق نبات الذرة إلى أربع فئات، وهي تبقع الأوراق الرمادي (Gray Leaf Spot)، والصدأ الشائع (Common Rust)، وحرق الأوراق الشمالي (Northern Leaf Blight)، والأوراق الصحية (Healthy)، كما تهدف إلى تحليل تأثير ضبط المعاملات الفائقة على أداء النموذج عبر خمس نسب مختلفة لتقسيم بيانات التدريب والاختبار. تشمل منهجية البحث المعالجة المسبقة للصور، وتدريب النموذج الأساسي (Baseline)، وضبط المعاملات الفائقة باستخدام طريقة البحث الشبكي (Grid Search) التي تستكشف توليفات ثلاثة معاملات، وهي معدل التعلم (Learning Rate)، وحجم الدفعة (Batch Size)، ومعدل الإسقاط (Dropout Rate)، مما ينتج عنه ثماني توليفات لكل نسبة، وبذلك يصل إجمالي السيناريوهات التجريبية إلى ٤٥ سيناريو. أظهرت النتائج أن طريقة البحث الشبكي نجحت في تحسين أداء النموذج في أربع من النسب الخمس التي تم اختبارها، حيث تم تحقيق أفضل دقة ودرجة F1 الإجمالية عند النسبة ٦٠:٤٠ باستخدام توليفة معدل التعلم ٠,٠٠٠٠٥، وحجم الدفعة ٣٢، ومعدل الإسقاط ٠,٣، بنسبة دقة بلغت ٩٣,٦٧٪ ودرجة F1 بلغت ٩٣,٧٤٪. وقد ثبت أن معدل الإسقاط ٠,٣ هو التكوين الأمثل بشكل ثابت في جميع النسب، في حين كانت فئة تبقع الأوراق الرمادي (Gray Leaf Spot) باستمرار الفئة ذات الأداء الأضعف نظرًا لتشابهها البصري مع حرق الأوراق الشمالي (Northern Leaf Blight). وقد تبين أن ضبط المعاملات الفائقة بشكل منهجي أكثر فعالية من استخدام المعاملات الافتراضية في تحسين أداء تصنيف أمراض أوراق الذرة باستخدام الصور.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Jagung (*Zea mays* L.) merupakan salah satu komoditas pangan strategis di Indonesia selain padi dan gandum (Rusnadi Padjung et al., 2024). Tanaman ini berperan penting sebagai sumber pangan, bahan baku industri, pakan ternak, serta penopang ketahanan pangan nasional (Luo et al., 2023). Namun, berdasarkan data Badan Pusat Statistik (BPS), produksi jagung pipilan kering dengan kadar air 14% pada tahun 2023 diperkirakan mencapai 14,46 juta ton, mengalami penurunan sebesar 12,50% dibandingkan tahun 2022 yang mencapai 16,53 juta ton (Badan Pusat Statistik, 2023). Kondisi tersebut menunjukkan bahwa produktivitas tanaman jagung masih menghadapi berbagai tantangan yang perlu mendapat perhatian. Salah satu faktor yang mempengaruhi produktivitas tanaman jagung adalah serangan penyakit (Azizah, 2023). Tanaman jagung rentan terhadap berbagai jenis penyakit sepanjang siklus pertumbuhannya, baik pada fase vegetatif maupun generatif, yang dapat menghambat pertumbuhan dan menurunkan hasil panen (Khoiri et al., 2021). Penyakit tersebut umumnya menimbulkan gejala visual pada daun, seperti perubahan warna, bercak, dan nekrosis. Beberapa penyakit daun jagung yang umum dijumpai antara lain hawar daun (*Northern Leaf Blight*), bercak daun (*Gray Leaf Spot*), dan karat daun (*Common Rust*) (Iswantoro & Handayani Un, 2022).

Penyakit daun pada tanaman jagung tidak hanya berdampak pada kerusakan visual, tetapi juga menurunkan luas daun efektif yang berperan dalam proses fotosintesis. Penurunan aktivitas fotosintesis berimplikasi pada terganggunya proses pembentukan dan distribusi karbohidrat sebagai hasil asimilasi, yang berpengaruh langsung terhadap pengisian biji dan pembentukan tongkol. Akibatnya, bobot tongkol dan produktivitas hasil panen dapat menurun secara signifikan (Khoiriyah et al., 2024). Dengan demikian, deteksi dan penanganan penyakit daun sejak dini menjadi langkah penting dalam menjaga stabilitas produksi jagung. Identifikasi penyakit tanaman jagung di lapangan hingga saat ini masih banyak dilakukan secara manual melalui pengamatan visual oleh petani atau tenaga penyuluh (Azizah, 2023). Metode ini memiliki keterbatasan, seperti bersifat subjektif, memerlukan keahlian khusus, serta kurang efisien jika diterapkan pada lahan yang luas (Tuncel & Gunturk, 2024). Keterlambatan dalam mendeteksi penyakit juga dapat menyebabkan penyebaran yang lebih luas dan berdampak pada penurunan hasil panen secara signifikan (Baksh et al., 2024). Oleh karena itu, diperlukan pendekatan berbasis teknologi yang mampu mendukung proses identifikasi penyakit secara cepat dan objektif. Perkembangan teknologi pengolahan citra digital dan kecerdasan buatan membuka peluang dalam mendukung sektor pertanian, khususnya dalam deteksi dan klasifikasi penyakit tanaman berbasis citra daun (Vankudoth, 2025). Salah satu metode yang efektif adalah *Convolutional Neural Network* (CNN), yang mampu mengekstraksi fitur citra secara otomatis tanpa perancangan fitur manual (Simran et al., 2021), Mawarni et al., 2023). Dalam penelitian ini digunakan arsitektur VGG16 *pretrained* melalui

pendekatan *transfer learning* karena memiliki struktur jaringan yang dalam dan performa yang baik pada berbagai tugas klasifikasi citra.

Kinerja CNN sangat dipengaruhi oleh pengaturan *hyperparameter* selama proses pelatihan, seperti *learning rate*, *batch size*, dan jumlah *epoch*. Pengaturan *hyperparameter* yang tidak optimal dapat menyebabkan *overfitting* atau *underfitting* sehingga menurunkan akurasi model (Cardoza et al., 2022). Namun, masih banyak penelitian yang belum menerapkan optimasi *hyperparameter* secara sistematis, sehingga terdapat peluang untuk meningkatkan performa model melalui *hyperparameter tuning*. Dalam konteks integrasi keilmuan dan keislaman, pemanfaatan teknologi untuk menjaga kualitas dan keberlanjutan pangan sejalan dengan ajaran Islam yang mendorong manusia untuk berpikir dan mengamati fenomena alam. Hal ini sebagaimana disebutkan dalam QS. An-Nahl ayat 11

يُنْبِثُ لَكُمْ بِهِ الرَّزْعَ وَالزَّيْتُونَ وَالنَّخِيلَ وَالْأَعْنَابَ وَمِنْ كُلِّ الثَّمَرَاتِ إِنَّ فِي ذَلِكَ لَآيَةً لِّقَوْمٍ يَتَفَكَّرُونَ

“Dengan (air hujan) itu Dia menumbuhkan untukmu tumbuh-tumbuhan, zaitun, kurma, anggur, dan segala macam buah-buahan. Sesungguhnya pada yang demikian itu benar-benar terdapat tanda (kebesaran Allah) bagi orang yang berpikir.” (Q.S An-Nahl:11)

Ayat tersebut menjadi landasan filosofis bahwa pengkajian ilmiah terhadap tanaman, termasuk upaya mendeteksi penyakitnya, merupakan bagian dari ikhtiar manusia dalam memanfaatkan akal sebagai amanah dari Allah *Subhanahu Wa Ta’ala* untuk menjaga dan mengelola ciptaan-Nya.

Berdasarkan uraian tersebut, penelitian ini mengusulkan penerapan *Convolutional Neural Network* (CNN) dengan arsitektur VGG16 yang dikombinasikan dengan *hyperparameter tuning* untuk mengklasifikasikan penyakit daun tanaman jagung berdasarkan citra digital. Hyperparameter tuning dilakukan

melalui pendekatan Grid Search yang mengeksplorasi kombinasi tiga parameter, yaitu *learning rate*, *batch size*, dan *dropout rate*, dengan tujuan memperoleh konfigurasi model yang menghasilkan performa klasifikasi terbaik. Penelitian difokuskan pada klasifikasi empat kelas daun jagung yang terdiri atas tiga jenis penyakit dan satu kelas daun sehat menggunakan dataset citra publik. Diharapkan penelitian ini dapat menghasilkan model CNN yang optimal serta berkontribusi dalam pengembangan sistem otomatis untuk identifikasi penyakit tanaman jagung.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah bagaimana performa klasifikasi berbasis arsitektur VGG16 dengan pendekatan *transfer learning* dalam mengklasifikasikan penyakit daun tanaman jagung ke dalam empat kelas, yaitu tiga jenis penyakit dan satu kelas daun sehat, berdasarkan citra digital daun?

1.3 Batasan Masalah

Agar penelitian ini lebih terarah dan sesuai dengan tujuan yang telah ditetapkan, maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Penelitian ini menggunakan dataset citra daun jagung repositori Github *Maize-Diseases-Detection* oleh R4j4n pada 08 September 2020 yang telah tersedia, tanpa melakukan proses akuisisi data secara langsung di lapangan.
2. Klasifikasi dibatasi pada empat kelas, yaitu Hawar Daun (*Northern Leaf Blight*), Bercak Daun (*Gray Leaf Spot*), Karat Daun (*Common Rust*), dan daun sehat (*Healthy*).

3. Citra yang digunakan merupakan citra daun tunggal dengan ukuran yang disesuaikan menjadi 224×224 piksel sesuai dengan kebutuhan input arsitektur VGG16.
4. Model yang digunakan adalah arsitektur VGG16 *pretrained* ImageNet dengan pendekatan *transfer learning*. Bagian *convolutional base* dibekukan (*freeze*), dan tidak dilakukan proses *fine-tuning* pada layer konvolusi.
5. *Hyperparameter tuning* dilakukan menggunakan metode *Grid Search* yang mengeksplorasi kombinasi tiga parameter, yaitu *learning rate* (0,001 dan 0,0005), *batch size* (16 dan 32), serta *dropout rate* (0,3 dan 0,5), menghasilkan total delapan kombinasi yang diuji pada tiap rasio pembagian data, tanpa melakukan optimasi pada parameter lain seperti jumlah layer atau arsitektur jaringan.
6. Proses pelatihan tiap skenario dibatasi maksimum 50 epoch, dengan mekanisme *Early Stopping* berdasarkan *validation loss* (*patience* 10 epoch) untuk menghentikan pelatihan lebih awal apabila tidak terdapat peningkatan performa.
7. Evaluasi performa model dilakukan berdasarkan metrik *accuracy*, *precision*, *recall*, *F1-score*, serta *confusion matrix* pada data uji, tanpa implementasi sistem secara *real-time* atau pada perangkat keras tertentu.
8. Penelitian ini berfokus pada analisis performa model klasifikasi dan tidak membahas aspek integrasi sistem, aplikasi *mobile*, maupun implementasi pada skala industri.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menganalisis performa model klasifikasi berbasis arsitektur VGG16 dengan pendekatan *transfer learning* dalam mengklasifikasikan penyakit daun tanaman jagung ke dalam empat kelas berdasarkan citra digital.

1.5 Manfaat Penelitian

Dengan dilakukannya penelitian ini diharapkan dapat memberi beberapa manfaat yaitu:

1. Bagi Petani:

Penelitian ini berpotensi membantu petani dalam mengidentifikasi penyakit daun jagung secara lebih cepat dan objektif melalui pendekatan berbasis citra digital. Deteksi yang lebih dini dapat mendukung pengambilan keputusan pengendalian penyakit secara tepat, sehingga risiko penurunan hasil panen dapat diminimalkan.

2. Bagi Dinas Pertanian

Model klasifikasi berbasis CNN yang dikembangkan dalam penelitian ini dapat menjadi dasar pengembangan sistem deteksi penyakit tanaman jagung secara otomatis. Pemanfaatan model ini berpotensi mendukung proses pemantauan kondisi tanaman secara lebih efisien serta mempercepat distribusi informasi terkait serangan penyakit kepada petani.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terdahulu

Klasifikasi merupakan salah satu tugas utama dalam *supervised learning* yang bertujuan memetakan data masukan ke dalam kategori tertentu berdasarkan pola yang dipelajari dari data berlabel. Dalam konteks penyakit tanaman, klasifikasi digunakan untuk mengelompokkan citra daun sesuai dengan jenis penyakit yang menyerang. Pada tanaman jagung, daun menjadi objek utama karena umumnya menampilkan gejala awal penyakit, seperti perubahan warna, kemunculan bercak, dan variasi tekstur permukaan daun (Setiawan et al., 2022). Pendekatan klasifikasi penyakit daun jagung secara konvensional banyak diterapkan menggunakan algoritma *machine learning* klasik, seperti *Support Vector Machine* (SVM) dan *K-Nearest Neighbor* (KNN). Metode-metode tersebut umumnya memerlukan proses ekstraksi fitur secara manual berdasarkan karakteristik visual daun, seperti warna, bentuk, dan tekstur. Ketergantungan pada perancangan fitur manual menyebabkan performa model sangat bergantung pada kualitas fitur dan keahlian perancang, sehingga sering kali kurang mampu merepresentasikan kompleksitas pola visual penyakit secara optimal (Anand, 2025).

Perkembangan *deep learning*, khususnya *Convolutional Neural Network* (CNN), menghadirkan pendekatan klasifikasi citra yang lebih adaptif. CNN mampu mengekstraksi fitur secara otomatis langsung dari citra mentah melalui lapisan konvolusi, sehingga dapat mengenali pola visual yang kompleks tanpa memerlukan

perancangan fitur eksplisit. Sejumlah penelitian menunjukkan bahwa CNN efektif dalam klasifikasi penyakit daun jagung, baik menggunakan arsitektur CNN konvensional maupun melalui pendekatan *transfer learning* (Yu et al., 2021) (Fraiwani et al., 2022).

Klasifikasi penyakit daun jagung menjadi penting karena setiap jenis penyakit memiliki karakteristik penyebab, tingkat penyebaran, serta metode penanganan yang berbeda (Murphy et al., 2024). Tanpa proses klasifikasi yang tepat, tindakan pengendalian di lapangan berisiko tidak sesuai sasaran, misalnya penggunaan fungisida yang tidak tepat jenis atau dosisnya. Selain itu, klasifikasi memungkinkan sistem otomatis memberikan rekomendasi penanganan yang lebih spesifik berdasarkan jenis penyakit yang teridentifikasi. Dari sisi penelitian, proses klasifikasi juga memungkinkan evaluasi performa model secara terukur melalui metrik seperti akurasi, presisi, recall, dan *confusion matrix*, sehingga efektivitas pendekatan yang digunakan dapat dibandingkan secara objektif. Dengan demikian, klasifikasi bukan sekadar proses pengelompokan citra, melainkan langkah krusial dalam mendukung pengambilan keputusan yang lebih cepat, tepat, dan berbasis data dalam pengelolaan penyakit tanaman jagung.

Berbagai studi telah dilakukan untuk mengklasifikasikan penyakit daun tanaman jagung berbasis citra dengan memanfaatkan pendekatan *Convolutional Neural Network* (CNN). Rachmad et al., (2023) mengembangkan model CNN dari awal (*from scratch*) menggunakan dataset publik *PlantVillage* dan berhasil memperoleh tingkat akurasi yang cukup tinggi pada data pengujian. Namun, penelitian tersebut menggunakan arsitektur CNN yang bersifat umum tanpa

penjelasan spesifik mengenai jenis arsitektur yang diterapkan, serta belum menganalisis pengaruh pengaturan *hyperparameter* terhadap performa model. Penelitian yang dilakukan oleh Mutia Fadhilla et al., (2023) mengklasifikasikan empat kelas daun jagung tiga jenis penyakit dan satu kelas daun sehat menggunakan CNN dasar dengan beberapa lapisan konvolusi. Hasilnya menunjukkan performa yang baik pada data pelatihan, tetapi terjadi penurunan kinerja pada data validasi yang mengindikasikan potensi *overfitting*. Selain itu, penelitian tersebut belum menerapkan proses optimasi *hyperparameter* secara terstruktur, sehingga konfigurasi model yang digunakan belum dapat dipastikan sebagai konfigurasi yang optimal.

Selanjutnya, Pratama & Pristyanto, (2023) melakukan perbandingan beberapa arsitektur CNN klasik, seperti *LeNet*, *AlexNet*, dan *VGG16*, dalam tugas klasifikasi penyakit daun jagung. Penelitian ini menunjukkan bahwa arsitektur yang lebih kompleks, seperti *VGG16*, mampu menghasilkan akurasi yang lebih tinggi dibandingkan CNN klasik. Akan tetapi, fokus penelitian tersebut lebih diarahkan pada perbandingan nilai akurasi tanpa mempertimbangkan efisiensi komputasi, jumlah parameter, serta potensi penerapan pada perangkat dengan keterbatasan sumber daya. Selain itu, aspek optimasi *hyperparameter* juga belum menjadi perhatian utama. Tinjauan sistematis yang dilakukan oleh Setiawan et al., (2022) terhadap penelitian-penelitian terkait klasifikasi penyakit jagung berbasis *machine learning* dan *deep learning* mengungkapkan sejumlah permasalahan umum, seperti keterbatasan ukuran dataset, rendahnya variasi kondisi pengambilan citra, serta minimnya pengembangan model CNN ringan dengan konfigurasi *hyperparameter*

yang dioptimalkan. Temuan ini menunjukkan bahwa peningkatan kinerja model tidak selalu harus bergantung pada arsitektur yang sangat kompleks, melainkan dapat dicapai melalui pemilihan arsitektur yang sesuai dan pengaturan parameter yang tepat. Di sisi lain, penelitian oleh (Tri Wahyuningrum et al., 2021) menerapkan arsitektur CNN yang lebih kompleks, yaitu *DenseNet-169*, dan berhasil mencapai tingkat akurasi yang sangat tinggi dalam klasifikasi penyakit daun jagung. Meskipun demikian, kompleksitas arsitektur serta kebutuhan komputasi yang besar menjadikan model tersebut kurang efisien untuk diterapkan pada sistem deteksi penyakit tanaman berbasis perangkat lapangan atau aplikasi bergerak yang memiliki keterbatasan sumber daya. Berdasarkan hasil-hasil penelitian tersebut, masih terdapat celah penelitian dalam pengembangan model CNN dengan arsitektur yang efektif serta dioptimalkan melalui pengaturan *hyperparameter* secara sistematis. Oleh karena itu, penelitian ini memfokuskan pada penerapan *Convolutional Neural Network* (CNN) dengan arsitektur VGG16 yang dikombinasikan dengan *hyperparameter tuning* untuk mengklasifikasikan penyakit daun tanaman jagung ke dalam empat kelas, yaitu *Northern Leaf Blight*, *Gray Leaf Spot*, *Common Rust*, dan daun sehat, menggunakan dataset citra publik. Berikut ringkasan penelitian terkait dapat dilihat pada tabel 2.1:

Tabel 2.1 Penelitian Terkait Mengenai Klasifikasi CNN

Kriteria	Penelitian 1	Penelitian 2	Penelitian 3	Penelitian 4	Penelitian 5	Penelitian ini
Peneliti (tahun)	Rachmad (2023)	Fadhilla, Suryani, & Labellapansa (2023)	Pratama & Pristyanto (2023)	Setiawan, Rochman, & Satoto (2022)	Tri Wahyuningrum & Kusumaningsih (2021)	Umi Farzah Nadliroh
Metode	CNN dengan arsitektur sederhana	CNN dasar dengan beberapa	Perbandingan arsitektur CNN	Studi tinjauan penerapan <i>machine</i>	CNN arsitektur kompleks	CNN arsitektur VGG16 dengan

	untuk klasifikasi penyakit daun jagung	lapisan konvolusi untuk klasifikasi empat kelas daun jagung	klasik dan modern (<i>LeNet</i> , <i>AlexNet</i> , <i>VGG16</i>)	<i>learning</i> dan <i>deep learning</i> pada penyakit jagung	(<i>DenseNet-169</i>)	<i>hyperparameter tuning</i>
Dataset	Dataset publik <i>PlantVillage</i> citra daun jagung	Dataset citra daun jagung hasil akuisisi lapangan	Dataset <i>PlantVillage</i> dan dataset lokal	Berbagai dataset publik dan dataset lapangan dari studi sebelumnya	Dataset <i>PlantVillage</i> citra daun jagung	Dataset publik <i>Maize Diseases Detection</i>
Jumlah kelas	4 kelas	4 kelas	4 kelas	4 kelas	Bervariasi	4 kelas
Hasil /performa	Akurasi mencapai 97,2% pada data uji terkontrol	Akurasi mencapai 94,5% dengan precision dan recall di atas 90%	VGG16 menghasilkan akurasi tertinggi sebesar 98,1%	CNN secara umum lebih unggul dibanding metode klasik (SVM, KNN)	Akurasi mencapai 99,1% dengan model berarsitektur dalam	-
Keterbatasan	Belum diuji pada variasi pencahayaan dan latar belakang alami	Dataset relatif kecil dan berpotensi overfitting	Model kompleks memerlukan sumber daya komputasi tinggi	Mayoritas studi tidak mengevaluasi efisiensi dan generalisasi model	Arsitektur sangat kompleks dan kurang efisien untuk sistem ringan	Terbatas pada optimasi hyperparameter VGG16 pada klasifikasi citra tanpa analisis efisiensi sistem secara mendalam.

2.2 Penyakit Daun Tanaman Jagung

Penyakit daun jagung merupakan salah satu faktor utama yang menyebabkan penurunan produktivitas tanaman karena kerusakan jaringan daun dapat mengganggu proses fotosintesis (Baireddy et al., 2022). Daun jagung menjadi objek yang paling mudah diamati dalam proses deteksi penyakit karena menampilkan perubahan visual yang jelas, seperti bercak, perubahan warna, dan

tekstur permukaan daun (Setiawan et al., 2022). Karakteristik visual tersebut menjadikan daun sebagai indikator utama dalam klasifikasi penyakit jagung berbasis citra digital (Rachmad et al., 2023). Pada penelitian ini, pembahasan difokuskan pada tiga jenis penyakit daun jagung yang umum digunakan dalam dataset citra publik, yaitu *Northern Leaf Blight*, *Gray Leaf Spot*, dan *Common Rust*, serta satu kelas daun sehat sebagai pembanding. Beberapa penyakit daun jagung antara lain:

2.2.1 Hawar Daun (Northern Leaf Blight)



Gambar 2.1 Hawar Daun (*Northern Leaf Blight*)

Hawar daun disebabkan oleh jamur patogen *Exserohilum turcicum* dan ditandai dengan munculnya bercak memanjang berbentuk lonjong pada permukaan daun (Zhang et al., 2021). Bercak awal berwarna abu-abu kehijauan dan berkembang menjadi coklat keabu-abuan dengan tepi yang jelas, sering kali disertai zona klorosis di sekitarnya (Peña et al., 2023). Ciri visual utama hawar daun meliputi bercak lonjong berukuran besar, kontras warna yang jelas antara lesi dan jaringan sehat, serta nekrosis luas pada tingkat infeksi yang berat. Karakteristik tersebut menjadikan penyakit ini relatif mudah dikenali melalui pendekatan klasifikasi citra digital.

2.2.2 Bercak Daun (*Gray Leaf Spot*)



Gambar 2.2 Bercak Daun (*Gray Leaf Spot*)

Gray Leaf Spot disebabkan oleh jamur *Cercospora zae-maydis* dan ditandai dengan bercak berbentuk persegi panjang atau elips yang sejajar dengan tulang daun (Baireddy et al., 2022). Gejala visual yang muncul meliputi bercak memanjang sejajar tulang daun, warna abu-abu kecoklatan dengan tepi lebih gelap, serta penyatuan bercak yang membentuk area nekrotik luas. Pola bercak yang khas ini menjadi fitur penting dalam proses klasifikasi berbasis CNN.

2.2.3 Karat Daun (*Common Rust*)



Gambar 2.3 Karat Daun (*Common Rust*)

Karat daun atau *Common Rust* disebabkan oleh jamur *Puccinia sorghi* dan ditandai dengan munculnya pustula kecil menonjol berwarna oranye kecoklatan pada permukaan daun (Anikster et al., n.d.). Seiring perkembangan penyakit, warna pustula dapat berubah menjadi lebih gelap. Ciri visual utama penyakit ini meliputi pustula kecil berwarna oranye kecoklatan, permukaan daun yang tidak rata akibat

tonjolan pustula, serta perubahan warna daun pada fase lanjut. Kontras warna pustula terhadap jaringan daun menjadikannya indikator visual penting dalam klasifikasi citra.

2.2.4 Tanaman Sehat (*Healthy*)



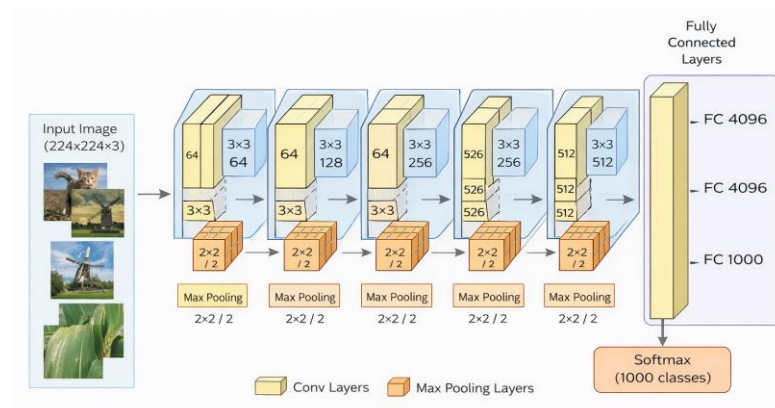
Gambar 2.4 *Tanaman Sehat (Healthy)*

Daun jagung sehat menunjukkan kondisi fisiologis yang optimal tanpa adanya gejala infeksi patogen. Citra daun sehat digunakan sebagai kelas pembanding dalam sistem klasifikasi penyakit daun jagung berbasis citra (Tri Wahyuningrum et al., 2021). Karakteristik visual daun sehat meliputi warna hijau segar yang merata, permukaan daun yang halus tanpa bercak atau lesi, serta struktur daun yang utuh tanpa nekrosis. Keberadaan kelas daun sehat membantu model CNN membedakan kondisi normal dan daun yang terinfeksi penyakit.

2.3 *Convolutional Neural Network (CNN)* dengan Arsitektur VGG16

Convolutional Neural Network (CNN) merupakan salah satu metode *deep learning* yang dirancang khusus untuk mengolah data citra digital dengan mempertahankan informasi spasial antar piksel. CNN bekerja melalui lapisan konvolusi untuk mengekstraksi fitur visual secara otomatis dan hierarkis, mulai dari fitur sederhana seperti tepi dan tekstur hingga pola visual yang lebih kompleks.

Proses pembelajaran dilakukan secara *end-to-end* sehingga tidak memerlukan perancangan fitur manual sebagaimana pada metode *machine learning* konvensional (Ihtada et al., 2025). Pemilihan arsitektur CNN menjadi faktor penting dalam menentukan keberhasilan proses klasifikasi. Arsitektur yang terlalu sederhana cenderung kurang mampu merepresentasikan pola visual yang kompleks, sedangkan arsitektur yang terlalu kompleks membutuhkan sumber daya komputasi besar dan berpotensi mengalami *overfitting* pada dataset yang terbatas. Oleh karena itu, diperlukan arsitektur CNN yang memiliki keseimbangan antara kemampuan representasi fitur dan efisiensi komputasi. Dalam konteks klasifikasi penyakit daun jagung, VGG16 dipilih karena memiliki struktur yang relatif sederhana dibandingkan arsitektur modern seperti DenseNet atau ResNet, namun tetap mampu memberikan performa yang tinggi. Selain itu, struktur lapisan yang teratur memudahkan proses modifikasi dan optimasi parameter pelatihan sesuai dengan karakteristik dataset (Duklan et al., 2024).



Gambar 2.5 Arsitektur CNN VGG16

Secara keseluruhan, arsitektur VGG16 terdiri dari dua bagian utama yaitu *convolutional base* dan *fully connected layers*. *Convolutional base* berfungsi

sebagai *feature extractor* yang mengekstraksi representasi visual dari citra input, sedangkan *fully connected layers* berfungsi sebagai *classifier* yang memetakan fitur ke kelas output. Berikut penjelasan detail setiap kelompok layer dalam arsitektur VGG16:

Tabel 2.2 Deskripsi Layer Arsitektur VGG16

Blok	Layer	Filter / Ukuran	Output Shape	Keterangan
Input	Input Layer	-	$224 \times 224 \times 3$	Citra RGB ukuran 224×224 piksel
Blok 1	Conv2D ($\times 2$)	64 filter, 3×3	$224 \times 224 \times 64$	Dua lapisan konvolusi filter 3×3 , stride 1, padding same
	MaxPooling2D	2×2 , stride 2	$112 \times 112 \times 64$	Reduksi dimensi spasial menjadi setengahnya
Blok 2	Conv2D ($\times 2$)	128 filter, 3×3	$112 \times 112 \times 128$	Dua lapisan konvolusi, jumlah filter digandakan
	MaxPooling2D	2×2 , stride 2	$56 \times 56 \times 128$	Reduksi dimensi spasial
Blok 3	Conv2D ($\times 3$)	256 filter, 3×3	$56 \times 56 \times 256$	Tiga lapisan konvolusi, fitur semakin kompleks
	MaxPooling2D	2×2 , stride 2	$28 \times 28 \times 256$	Reduksi dimensi spasial
Blok 4	Conv2D ($\times 3$)	512 filter, 3×3	$28 \times 28 \times 512$	Tiga lapisan konvolusi, fitur abstrak tingkat tinggi
	MaxPooling2D	2×2 , stride 2	$14 \times 14 \times 512$	Reduksi dimensi spasial
Blok 5	Conv2D ($\times 3$)	512 filter, 3×3	$14 \times 14 \times 512$	Tiga lapisan konvolusi terakhir pada feature extractor
	MaxPooling2D	2×2 , stride 2	$7 \times 7 \times 512$	Output akhir convolutional base
Custom	Flatten	-	25.088	Mengubah feature map $7 \times 7 \times 512$ menjadi vektor 1 dimensi
	Dense (ReLU)	128 neuron	128	Lapisan fully connected dengan aktivasi ReLU
	Dropout	0,5	128	Mematikan 50% neuron secara acak saat training — cegah overfitting
	Dense (Softmax)	4 neuron	4	Output probabilitas 4 kelas penyakit daun jagung

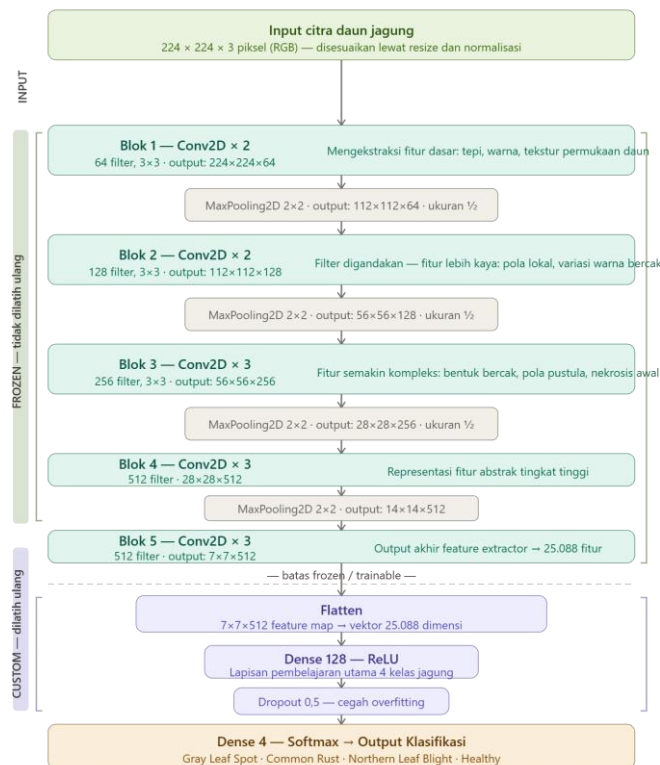
Setiap blok konvolusi pada VGG16 terdiri dari beberapa lapisan Conv2D yang diikuti oleh satu lapisan MaxPooling2D. Lapisan Conv2D menggunakan filter berukuran 3×3 secara konsisten di seluruh jaringan. Ini merupakan salah satu ciri khas VGG16 yang membedakannya dari arsitektur sebelumnya yang menggunakan filter lebih besar. Penggunaan filter kecil 3×3 secara bertumpuk terbukti lebih efektif dalam menangkap fitur visual kompleks dibanding satu filter besar, sekaligus mengurangi jumlah parameter yang harus dipelajari.

Lapisan MaxPooling2D dengan ukuran 2×2 dan stride 2 ditempatkan setelah setiap blok konvolusi. Fungsinya adalah mengurangi dimensi spasial output sebesar setengahnya, sehingga komputasi menjadi lebih efisien sekaligus memberikan sifat translasi-invarian pada model yang artinya model tetap mampu mengenali pola penyakit meskipun posisinya berbeda dalam gambar (Septiandi et al., 2021).

Dalam penelitian ini, bagian convolutional base VGG16 dibekukan (*freeze*) sehingga bobotnya tidak diperbarui selama proses pelatihan. Pendekatan ini disebut *transfer learning* memanfaatkan pengetahuan yang sudah dipelajari model dari dataset ImageNet yang berisi jutaan gambar. Pembekuan dilakukan karena jumlah data latih dalam penelitian ini relatif terbatas (1.500 citra), sehingga melatih ulang seluruh 14 juta parameter konvolusi berisiko menyebabkan overfitting. Hanya lapisan custom di bagian akhir Flatten, Dense(128), Dropout(0,5), dan Dense(4,

Softmax) yang dilatih dari awal untuk menyesuaikan model dengan empat kelas penyakit daun jagung.

Lapisan Flatten berfungsi mengubah output *feature map* berukuran $7 \times 7 \times 512$ dari *convolutional base* menjadi vektor satu dimensi berisi 25.088 elemen. Vektor ini kemudian diproses oleh Dense layer dengan 128 neuron dan fungsi aktivasi ReLU sebagai lapisan pembelajaran utama. Lapisan Dropout dengan nilai 0,5 ditambahkan setelahnya untuk mencegah overfitting dengan cara mematikan 50% neuron secara acak pada setiap iterasi pelatihan. Pada lapisan output, digunakan Dense layer dengan 4 neuron dan fungsi aktivasi Softmax yang menghasilkan distribusi probabilitas untuk keempat kelas, yaitu Gray Leaf Spot, Common Rust, Northern Leaf Blight, dan Healthy. Total parameter model adalah 17.926.596, dengan 3.211.908 parameter yang dapat dilatih dan 14.714.688 parameter yang dibekukan.



Gambar 2.6 Arsitektur Model CNN VGG16 yang Digunakan dalam Penelitian (Convolutional Base Frozen + Custom Classifier)

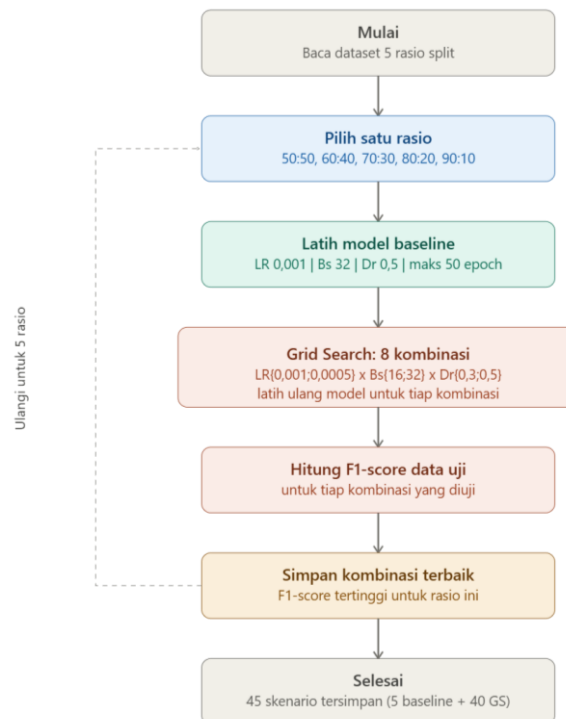
2.4 Hyperparameter Tuning (Grid Search)

Hyperparameter tuning merupakan proses optimasi parameter pelatihan yang ditentukan sebelum proses pembelajaran model dimulai. Berbeda dengan parameter model (seperti bobot dan bias) yang dipelajari secara otomatis selama *training*, *hyperparameter* ditetapkan secara manual dan sangat mempengaruhi performa model. Contoh *hyperparameter* pada model *Convolutional Neural Network* (CNN) antara lain *learning rate*, jumlah *epoch*, ukuran batch (*batch size*), serta konfigurasi *optimizer* (Morales-Hernández et al., 2023). Pemilihan nilai *hyperparameter* yang kurang tepat dapat menyebabkan model mengalami *underfitting* atau *overfitting*. *Learning rate* yang terlalu besar dapat membuat proses pembelajaran tidak stabil, sedangkan *learning rate* yang terlalu kecil dapat

memperlambat konvergensi dan meningkatkan waktu pelatihan. *Batch size* mempengaruhi stabilitas estimasi gradien pada setiap iterasi pelatihan, sedangkan *dropout rate* mempengaruhi tingkat regularisasi yang diterapkan untuk mencegah *overfitting*. Oleh karena itu, diperlukan strategi pencarian kombinasi hyperparameter yang optimal, yang tidak hanya berfokus pada satu parameter saja.

Salah satu metode yang umum digunakan untuk pencarian hyperparameter secara sistematis adalah *Grid Search*. *Grid Search* bekerja dengan cara mendefinisikan terlebih dahulu sekumpulan nilai kandidat untuk setiap *hyperparameter* yang akan dioptimalkan. Selanjutnya, model dilatih dan dievaluasi untuk setiap kombinasi nilai yang ada. Kombinasi dengan performa terbaik kemudian dipilih sebagai konfigurasi optimal. Pendekatan ini menjamin bahwa seluruh kombinasi kandidat dieksplorasi secara menyeluruh dan transparan (Sholeh et al., 2025).

Dalam penelitian ini *Grid Search* diimplementasikan secara manual menggunakan iterasi Python untuk menguji kombinasi tiga parameter hyperparameter, yaitu *learning rate*, *batch size*, dan *dropout rate*, pada setiap rasio pembagian data. Ketiga parameter tersebut dipilih karena secara langsung mempengaruhi proses optimasi bobot (*learning rate*), stabilitas estimasi gradien pada setiap iterasi (*batch size*), dan tingkat regularisasi model (*dropout rate*), sehingga kombinasi ketiganya memberikan eksplorasi yang lebih komprehensif dibandingkan optimasi satu parameter saja. Alur proses *Grid Search* yang diterapkan dalam penelitian ini ditunjukkan pada Gambar 2.7 berikut.



Gambar 2.7 Flowchart Proses Grid Search

Berdasarkan flowchart pada Gambar 2.7, proses dimulai dengan membaca seluruh folder dataset dari Google Drive yang mencakup lima variasi rasio pembagian data, yaitu 50:50, 60:40, 70:30, 80:20, dan 90:10. Setiap folder rasio berisi subfolder train dan test. Untuk setiap rasio, proses berjalan dalam dua tahap secara berurutan. Tahap pertama adalah pelatihan konfigurasi *baseline*, di mana model VGG16 dilatih dengan *learning rate* 0,001, *batch size* 32, *dropout rate* 0,5, maksimum 50 epoch, dan mekanisme *Early Stopping* berdasarkan *validation loss* dengan *patience* 10 epoch. Hasil evaluasi berupa nilai *accuracy*, *precision*, *recall*, dan *F1-score* kemudian disimpan sebagai titik acuan untuk rasio tersebut.

Tahap kedua adalah proses Grid Search itu sendiri. Pada tahap ini, didefinisikan kandidat nilai untuk setiap parameter, yaitu *learning rate* (0,001;

0,0005), *batch size* (16; 32), dan *dropout rate* (0,3; 0,5). Kombinasi seluruh nilai kandidat tersebut menghasilkan $2 \times 2 \times 2 = 8$ kombinasi *hyperparameter* yang diuji secara bergantian untuk setiap rasio pembagian data. Pada setiap kombinasi, model VGG16 dilatih ulang dari awal dengan parameter yang sedang diuji, maksimum 50 epoch, dan mekanisme *Early Stopping* yang sama dengan tahap *baseline*. Setelah pelatihan selesai, nilai *F1-score* pada data uji dihitung dan dibandingkan dengan nilai terbaik (*best_f1*) yang tersimpan. Apabila *F1-score* kombinasi saat ini lebih tinggi, maka kombinasi tersebut ditetapkan sebagai kombinasi terbaik sementara. Proses ini berlanjut hingga seluruh 8 kombinasi selesai diuji.

Setelah seluruh kombinasi dievaluasi untuk suatu rasio, kombinasi dengan *F1-score* tertinggi ditetapkan sebagai konfigurasi optimal dan hasilnya disimpan sebagai representasi performa Grid Search untuk rasio tersebut. Seluruh proses kemudian diulang untuk rasio pembagian data berikutnya hingga kelima rasio selesai diproses. Tabel 2.3 berikut menyajikan seluruh kombinasi hyperparameter yang dieksplorasi dalam proses Grid Search pada penelitian ini untuk rasio 50:50 sebagai representasi pola yang sama diterapkan pada keempat rasio lainnya.

Tabel 2.3 Kandidat Kombinasi Hyperparameter dalam Grid Search

No	Rasio Split	Learning Rate	Batch Size	Dropout Rate	Max Epoch
1	50:50	0,001	16	0,3	50
2	50:50	0,001	16	0,5	50
3	50:50	0,001	32	0,3	50
4	50:50	0,001	32	0,5	50
5	50:50	0,0005	16	0,3	50
6	50:50	0,0005	16	0,5	50

7	50:50	0,0005	32	0,3	50
8	50:50	0,0005	32	0,5	50

Tabel 2.3 menunjukkan bahwa parameter yang dieksplorasi dalam *Grid Search* pada penelitian ini mencakup tiga parameter, yaitu *learning rate* (0,001 dan 0,0005), *batch size* (16 dan 32), serta *dropout rate* (0,3 dan 0,5). Nilai *learning rate* 0,001 dipilih sebagai nilai *default* optimizer Adam yang umum digunakan sebagai titik awal dalam penelitian CNN (Cardoza et al., 2022), sedangkan 0,0005 dipilih sebagai setengah dari nilai *default* untuk mengeksplorasi potensi konvergensi yang lebih stabil. Nilai *batch size* 32 dipilih sebagai konfigurasi umum yang menyeimbangkan kecepatan pelatihan dan stabilitas estimasi gradien, sedangkan *batch size* 16 dieksplorasi untuk melihat pengaruh ukuran *batch* yang lebih kecil terhadap akurasi model pada dataset yang relatif terbatas. Nilai *dropout rate* 0,5 dipilih sebagai nilai umum untuk regularisasi yang kuat, sedangkan 0,3 dieksplorasi untuk melihat pengaruh regularisasi yang lebih ringan terhadap kemampuan generalisasi model.

Parameter lain seperti *batch size*, jumlah neuron pada dense layer, nilai dropout, dan arsitektur jaringan tidak dioptimalkan dalam penelitian ini dan ditetapkan secara konstan di seluruh eksperimen. Hal ini dilakukan untuk memastikan bahwa perbedaan performa yang diamati benar-benar disebabkan oleh perbedaan nilai *learning rate*, bukan oleh faktor lain yang tidak dikontrol. Selain itu, pembatasan ruang pencarian ini juga mempertimbangkan keterbatasan sumber

daya komputasi yang tersedia, mengingat setiap kombinasi membutuhkan satu siklus pelatihan penuh.

Pemilihan LR terbaik dilakukan berdasarkan nilai akurasi pada data uji (*test set*). Dari dua kandidat yang diuji untuk setiap rasio, kandidat dengan akurasi tertinggi ditetapkan sebagai learning rate terpilih dan hasilnya digunakan sebagai representasi performa *Grid Search* untuk rasio tersebut.

Tabel 2.4 Experimental Setup Model

No	Rasio <i>Train:Test</i>	Pendekatan	Jumlah Skenario	Max Epoch
1	50:50	Baseline	1	50
2	60:40	Baseline	8	50
3	70:30	Baseline	1	50
4	80:20	Baseline	8	50
5	90:10	Baseline	1	50
6	50:50	Grid Search	8	50
7	60:40	Grid Search	1	50
8	70:30	Grid Search	8	50
9	80:20	Grid Search	1	50
10	90:10	Grid Search	8	50

Tabel 2.4 menunjukkan keseluruhan 45 skenario eksperimen yang dijalankan dalam penelitian ini, terdiri dari 5 skenario *baseline* (satu untuk setiap rasio pembagian data) dan 40 skenario *Grid Search* (8 kombinasi hyperparameter $\times$ 5 rasio pembagian data). Setiap skenario dilatih hingga maksimum 50 epoch dengan mekanisme *Early Stopping* berdasarkan *validation loss* (*patience* 10 epoch), sehingga jumlah epoch aktual pada setiap skenario dapat berhenti lebih awal apabila tidak terdapat peningkatan performa pada data validasi.

BAB III

DESAIN DAN IMPLEMENTASI SISTEM

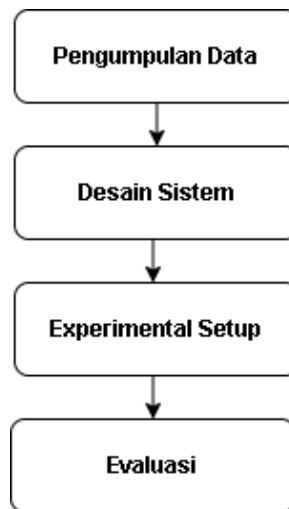
3.1 Desain Penelitian

Desain penelitian pada dasarnya menjadi arah dan pegangan selama proses penelitian berlangsung. Tanpa perencanaan yang jelas, penelitian bisa saja berjalan tetapi tidak terstruktur dan sulit untuk dianalisis secara sistematis. Karena itu, sebelum masuk ke tahap implementasi model, terlebih dahulu disusun rancangan penelitian yang menjelaskan bagaimana proses akan dilakukan dari awal sampai evaluasi akhir. Penelitian ini menggunakan pendekatan eksperimen kuantitatif. Artinya, yang ingin dilihat bukan sekadar apakah model dapat berjalan atau tidak, melainkan seberapa baik performanya ketika diuji dalam kondisi tertentu. Fokusnya memang pada metode *Convolutional Neural Network* (CNN) khususnya arsitektur VGG16, namun yang menjadi perhatian utama adalah kemampuan model dalam mengklasifikasikan citra daun jagung ke dalam empat kategori yaitu tiga jenis penyakit dan satu kelas daun sehat.

Karena yang ingin diukur adalah performa, maka perlu dibuat beberapa skenario pengujian. Tidak cukup hanya melatih model sekali lalu melihat hasil akurasi. Oleh sebab itu, penelitian ini dirancang dengan dua variasi utama. Pertama, variasi pembagian data latih dan data uji. Dataset diuji menggunakan lima rasio berbeda, yaitu 50:50, 60:40, 70:30, 80:20, dan 90:10. Rasio ini dipilih untuk melihat apakah proporsi data latih yang lebih besar benar-benar memberikan peningkatan performa yang signifikan atau tidak. Kedua, pada setiap pembagian

data tersebut dilakukan dua pendekatan pelatihan. Pendekatan pertama adalah pelatihan tanpa penyesuaian *hyperparameter*, yang disebut sebagai *baseline*. Pendekatan ini menjadi titik awal pembandingan. Pendekatan kedua menggunakan *hyperparameter tuning* dengan metode *Grid Search* untuk mencari konfigurasi yang lebih optimal, dengan mengeksplorasi kombinasi tiga parameter sekaligus, yaitu *learning rate*, *batch size*, dan *dropout rate*. Dengan cara ini, dapat dianalisis apakah tuning benar-benar memberikan perbedaan performa yang berarti dibandingkan konfigurasi awal, serta parameter mana yang memberikan kontribusi terbesar terhadap peningkatan performa.

Melalui rancangan seperti ini, hasil penelitian tidak hanya menunjukkan satu angka akurasi, tetapi memberikan gambaran yang lebih utuh mengenai bagaimana model bekerja dalam berbagai kondisi. Setiap eksperimen kemudian dievaluasi menggunakan *confusion matrix* serta metrik kuantitatif seperti *accuracy*, *precision*, *recall*, dan *F1-score*. Evaluasi tersebut diperlukan agar analisis performa tidak hanya bergantung pada satu indikator saja, melainkan dapat dilihat dari berbagai sudut pandang. Dengan demikian, desain penelitian ini disusun bukan hanya untuk membangun model klasifikasi, tetapi untuk memastikan bahwa proses pengujian dan evaluasi dilakukan secara sistematis dan dapat dipertanggungjawabkan secara ilmiah. Alur desain penelitian secara keseluruhan ditunjukkan pada Gambar 3.1.



Gambar 3.1 Desain Penelitian

3.2 Pengumpulan Data

Data yang digunakan pada penelitian ini bukan dikumpulkan sendiri dari lapangan, melainkan memanfaatkan dataset publik yang sudah tersedia. Dataset tersebut diambil dari repositori GitHub *Maize-Diseases-Detection* oleh R4j4n pada 8 September 2020. Secara keseluruhan terdapat 1500 citra daun jagung dan gambar sudah diberi label ke dalam empat kategori, yaitu *Healthy*, *Northern Leaf Blight*, *Gray Leaf Spot*, dan *Common Rust*. Keempat kelas ini mewakili kondisi daun jagung yang sehat maupun yang terinfeksi penyakit tertentu, sehingga cocok untuk skenario klasifikasi multi-kelas.

Tabel 3.1 Contoh Sampel Gambar Dari Dataset Yang Digunakan

No.	Nama Kondisi Daun Jagung	Gambar
1.	<i>Blight</i> (Hawar daun)	
2.	<i>Gray Leaf Spot</i> (Bercak daun)	
3.	<i>Common Rust</i> (Karat daun)	
4.	<i>Healthy</i> (Daun sehat)	

Pada dataset publik yang digunakan dalam penelitian ini, pembagian data awal hanya tersedia dalam satu konfigurasi, yaitu folder *train* dan *test*, yang masing-masing telah berisi empat kelas citra daun jagung. Dengan demikian, variasi rasio pembagian data 50:50 hingga 90:10 bukan merupakan pembagian bawaan dari repositori asli dataset tersebut. Untuk memenuhi kebutuhan desain eksperimen, dilakukan proses pemisahan ulang data secara manual guna menghasilkan lima variasi rasio pembagian data, yaitu 50:50, 60:40, 70:30, 80:20, dan 90:10.

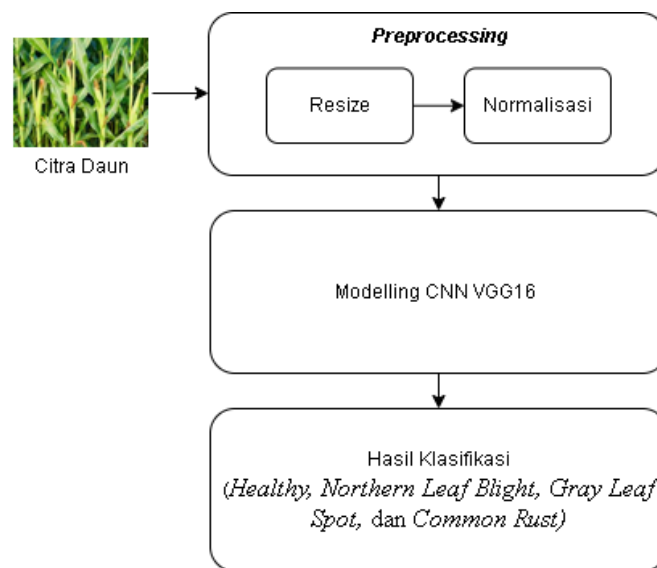
Proses pemisahan ulang ini dilakukan dengan mempertimbangkan distribusi setiap kelas agar tetap proporsional pada masing-masing rasio. Tujuannya adalah menjaga keseimbangan jumlah data antar kelas sehingga perbandingan performa model pada setiap skenario pengujian dapat dilakukan secara adil dan tidak dipengaruhi oleh ketimpangan distribusi data. Dengan pendekatan ini, hasil evaluasi yang diperoleh lebih mencerminkan pengaruh rasio pembagian data terhadap performa model, bukan akibat bias komposisi kelas. Setelah proses pemisahan selesai, setiap rasio disusun dalam folder terpisah. Di dalam masing-masing folder terdapat subfolder *train* dan *test*, yang kemudian kembali dibagi ke dalam empat subfolder sesuai dengan kelasnya. Struktur folder yang konsisten ini dirancang untuk memudahkan proses pelatihan dan pengujian model secara berulang tanpa perlu melakukan pembagian ulang data pada setiap eksperimen. Dengan demikian, variasi rasio dalam penelitian ini merupakan bagian dari desain eksperimental yang dirancang secara sistematis, bukan berasal langsung dari struktur dataset publik yang digunakan.

3.3 Desain Sistem

Sistem dalam penelitian ini dirancang untuk melakukan klasifikasi penyakit daun jagung berbasis citra menggunakan metode *Convolutional Neural Network* (CNN). Implementasi dilakukan menggunakan bahasa pemrograman *Python* dengan bantuan platform *Google Colab* sebagai lingkungan komputasi. Secara umum, sistem bekerja mulai dari pemuatan dataset hingga menghasilkan evaluasi performa model. Proses tidak langsung melibatkan model begitu saja, melainkan melalui beberapa tahapan yang saling terhubung. Tahap pertama adalah pemuatan dataset dari direktori penyimpanan. Pada tahap ini, sistem membaca struktur folder yang berisi data latih (*train*) dan data uji (*test*) beserta empat kelas penyakit daun jagung. Setelah data dimuat, sistem menampilkan distribusi jumlah citra pada masing-masing kelas. Langkah ini penting untuk memastikan komposisi data dapat diketahui sebelum proses pelatihan dilakukan.

Selanjutnya citra memasuki tahap *preprocessing*. Pada tahap ini dilakukan penyesuaian ukuran gambar agar sesuai dengan kebutuhan input model VGG16, serta normalisasi nilai piksel agar proses pelatihan lebih stabil. Data yang telah melalui tahap ini kemudian siap digunakan dalam proses pelatihan model. Model CNN kemudian dibangun dan dikompilasi. Proses pelatihan dilakukan menggunakan data *train*, sedangkan data *test* disimpan untuk tahap evaluasi. Dalam penelitian ini, pelatihan dilakukan dalam dua skenario, yaitu pengujian *baseline* tanpa penyesuaian *hyperparameter* dan pengujian dengan *hyperparameter tuning* menggunakan metode *Grid Search*. Setelah proses pelatihan selesai, model diuji

menggunakan data *test*. Hasil prediksi dibandingkan dengan label sebenarnya untuk menghasilkan *confusion matrix* serta metrik evaluasi seperti akurasi, *precision*, *recall*, dan *F1-score*. Seluruh tahapan ini dirancang agar setiap skenario eksperimen dapat dijalankan secara konsisten dan hasilnya dapat dibandingkan secara objektif. Alur desain sistem secara keseluruhan ditunjukkan pada Gambar 3.2.



Gambar 3.2 Desain Sistem Penelitian

3.4 Preprocessing

Tahap *preprocessing* dilakukan untuk menyiapkan data citra sebelum digunakan dalam proses pelatihan model *Convolutional Neural Network* (CNN). Proses ini bertujuan untuk memastikan bahwa seluruh citra memiliki format dan karakteristik yang sesuai dengan kebutuhan arsitektur VGG16. Tanpa tahap ini, perbedaan ukuran gambar, rentang nilai piksel, maupun variasi kualitas citra dapat mempengaruhi stabilitas dan kinerja model selama proses pelatihan. Oleh karena itu, *preprocessing* menjadi langkah penting agar data yang digunakan lebih terstandarisasi dan siap diproses oleh jaringan saraf. Langkah pertama dalam

preprocessing adalah melakukan penyesuaian ukuran (*resize*) citra menjadi 224×224 piksel. Ukuran ini dipilih karena sesuai dengan spesifikasi input arsitektur VGG16 yang digunakan dalam penelitian. Setelah proses *resize*, dilakukan normalisasi nilai piksel dengan mengubah rentang nilai dari 0–255 menjadi 0–1 melalui pembagian dengan 255. Normalisasi bertujuan untuk mempercepat proses konvergensi dan menjaga kestabilan pembaruan bobot selama proses pelatihan, sehingga model dapat belajar secara lebih efektif.

Selain *resize* dan normalisasi, pada tahap *preprocessing* juga diterapkan augmentasi data khusus pada data latih (*training set*). Augmentasi dilakukan dengan variasi transformasi seperti rotasi, *zoom*, dan horizontal flip untuk meningkatkan keragaman data tanpa menambah jumlah citra secara manual. Teknik ini bertujuan untuk membantu model menjadi lebih robust terhadap variasi sudut pengambilan gambar maupun kondisi lingkungan, sekaligus mengurangi risiko *overfitting*. Data uji (*test set*) tidak diberikan augmentasi agar proses evaluasi tetap mencerminkan performa model terhadap data yang belum pernah dilihat sebelumnya.

Selain itu, dalam proses pemuatan data latih digunakan parameter `validation_split=0.2` pada `ImageDataGenerator`. Parameter ini memisahkan 20% dari data latih secara otomatis sebagai data validasi internal yang digunakan selama proses pelatihan untuk memantau perkembangan nilai *validation loss* pada setiap *epoch*. Dengan demikian, data yang secara efektif digunakan untuk melatih bobot model adalah sebesar 80% dari total data latih pada masing-masing rasio, sedangkan 20% sisanya berfungsi sebagai data validasi. Data uji (*test set*) tetap

terpisah sepenuhnya dan tidak terlibat dalam proses pelatihan maupun validasi internal.

3.5 Experimental Setup

Tahap *experimental setup* pada penelitian ini dirancang untuk memastikan bahwa seluruh proses pelatihan dan pengujian model dilakukan secara sistematis serta terkontrol. Model klasifikasi yang digunakan merupakan arsitektur *Convolutional Neural Network* berbasis VGG16 dengan bobot *pretrained* dari *ImageNet*. Pemanfaatan *pretrained* model dilakukan agar proses pembelajaran tidak dimulai dari awal, melainkan memanfaatkan bobot yang telah dilatih pada jutaan citra sebelumnya. Pendekatan ini membantu proses konvergensi menjadi lebih cepat dan stabil, terutama karena jumlah data pada penelitian ini relatif terbatas. Pada tahap pemuatan model digunakan parameter *include_top=False*, sehingga lapisan *fully connected* bawaan VGG16 yang dirancang untuk klasifikasi 1000 kelas tidak disertakan. Dengan demikian, arsitektur model dapat disesuaikan dengan kebutuhan penelitian yang hanya memiliki empat kelas penyakit daun jagung.

Bagian *convolutional* dari VGG16 dipertahankan sebagai *feature extractor* dan dibekukan (*freeze*) sehingga bobot hasil *pretraining* tidak diperbarui selama proses pelatihan. Pendekatan ini dilakukan agar fitur dasar seperti pola tepi, tekstur, dan bentuk yang telah dipelajari oleh model tetap stabil, sementara proses pembelajaran difokuskan pada lapisan klasifikasi yang ditambahkan pada bagian akhir jaringan. Setelah *feature extractor*, ditambahkan beberapa lapisan baru untuk

proses klasifikasi. Lapisan pertama adalah *Flatten* yang berfungsi mengubah output *feature map* menjadi vektor satu dimensi. Selanjutnya ditambahkan *Dense layer* sebagai lapisan pembelajaran utama pada tahap klasifikasi. Untuk mengurangi risiko *overfitting*, ditambahkan lapisan *Dropout* yang secara acak menonaktifkan sebagian neuron selama proses pelatihan. Pada bagian akhir digunakan *Dense output layer* dengan empat neuron dan fungsi aktivasi *Softmax* untuk menghasilkan probabilitas pada masing-masing kelas penyakit daun jagung, yaitu *Healthy*, *Northern Leaf Blight*, *Gray Leaf Spot*, dan *Common Rust*.

Setelah arsitektur model terbentuk, model kemudian dikompilasi menggunakan optimizer Adam dengan fungsi *loss categorical_crossentropy*, yang sesuai untuk permasalahan klasifikasi multikelas. Metrik evaluasi yang dipantau selama pelatihan adalah *accuracy* untuk melihat perkembangan performa model pada setiap epoch. Selama proses pelatihan diterapkan mekanisme *Early Stopping* dengan nilai *patience* sebesar 3 untuk menghentikan pelatihan apabila nilai *validation loss* tidak mengalami perbaikan dalam tiga epoch berturut-turut. Mekanisme ini digunakan untuk mengurangi risiko *overfitting* sekaligus menghemat waktu pelatihan model. Struktur akhir model kemudian diverifikasi menggunakan fungsi *model.summary()* untuk memastikan bahwa susunan lapisan dan jumlah parameter telah sesuai dengan rancangan arsitektur yang ditetapkan.

Eksperimen dalam penelitian ini tidak dilakukan hanya pada satu kondisi pelatihan, melainkan pada beberapa kombinasi skenario untuk melihat bagaimana perubahan komposisi data dan konfigurasi pelatihan mempengaruhi performa

model. Dataset yang digunakan telah dibagi ke dalam lima variasi rasio data latih dan data uji, yaitu 50:50, 60:40, 70:30, 80:20, dan 90:10. Setiap rasio disusun dalam folder terpisah dengan struktur *train* dan *test* yang konsisten pada empat kelas. Variasi ini bertujuan untuk menganalisis pengaruh proporsi data latih terhadap kemampuan generalisasi model. Penggunaan beberapa rasio pembagian data didasarkan pada temuan penelitian sebelumnya yang menunjukkan bahwa proporsi data latih dan data uji dapat mempengaruhi stabilitas serta kemampuan generalisasi model. Variasi rasio *train-test* digunakan untuk mengamati bagaimana perubahan jumlah data latih berdampak terhadap performa klasifikasi, sekaligus menghindari bias evaluasi yang dapat muncul apabila model hanya diuji pada satu rasio pembagian data tertentu (Bichri et al., 2024).

Pada setiap rasio pembagian data dilakukan dua pendekatan pelatihan, yaitu skenario *baseline* dan skenario dengan *hyperparameter tuning*. Skenario *baseline* digunakan sebagai titik awal evaluasi performa model, di mana model CNN berbasis VGG16 dilatih menggunakan konfigurasi standar tanpa proses pencarian kombinasi hyperparameter tambahan. Pada tahap ini digunakan nilai *learning rate* sebesar 0,001 *batch size* 32, *dropout rate* 0,5, dengan jumlah epoch maksimum sebanyak 50 epoch dan mekanisme *Early Stopping* (*patience* 10 epoch). Setelah hasil *baseline* diperoleh, dilakukan pengujian lanjutan menggunakan *hyperparameter tuning* dengan metode *Grid Search*. Metode ini digunakan untuk mencari konfigurasi parameter yang lebih optimal melalui pencarian sistematis pada ruang parameter yang telah ditentukan. Parameter yang dioptimasi dalam penelitian ini mencakup tiga parameter, yaitu *learning rate* dengan dua kandidat

nilai (0,001 dan 0,0005), *batch size* dengan dua kandidat nilai (16 dan 32), serta *dropout rate* dengan dua kandidat nilai (0,3 dan 0,5).

Kombinasi ketiga parameter tersebut menghasilkan $2 \times 2 \times 2 = 8$ kombinasi hyperparameter yang diuji pada setiap rasio pembagian data. Setiap kombinasi parameter digunakan untuk melatih model hingga maksimum 50 epoch dengan mekanisme *Early Stopping* yang sama seperti pada skenario *baseline* (*patience* 10 epoch). Performa model diamati berdasarkan nilai *F1-score* pada data uji, dan kombinasi hyperparameter dengan performa terbaik dipilih sebagai konfigurasi optimal untuk rasio tersebut.

Penerapan *hyperparameter tuning* dilakukan karena kinerja model *deep learning* sangat sensitif terhadap pemilihan *hyperparameter*, tidak hanya *learning rate*, tetapi juga *batch size* dan *dropout rate*, yang masing-masing berpengaruh terhadap stabilitas konvergensi estimasi gradien, dan tingkat regularisasi model. Sejumlah penelitian menunjukkan bahwa proses optimasi *hyperparameter*, termasuk *Grid Search*, mampu meningkatkan performa model secara signifikan dibandingkan penggunaan parameter *default* (Sholeh et al., 2025). Dengan pendekatan ini, penelitian tidak hanya membandingkan performa model sebelum dan sesudah *tuning*, tetapi juga mengevaluasi sejauh mana optimasi parameter pelatihan berkontribusi dalam meningkatkan kemampuan klasifikasi citra daun jagung secara terukur dan sistematis. Tabel berikut menunjukkan bahwa setiap rasio pembagian data diuji dalam dua pendekatan pelatihan, yaitu *baseline* dan *tuning* menggunakan *Grid Search*. Total skenario yang dijalankan adalah 45, terdiri dari 5

skenario *baseline* (satu untuk setiap rasio, dengan *learning rate* 0,001, *batch size* 32, *dropout rate* 0,5, maksimum 50 epoch) dan 40 skenario Grid Search (8 kombinasi hyperparameter $\times$ 5 rasio pembagian data, masing-masing maksimum 50 epoch). Seluruh skenario menggunakan mekanisme *Early Stopping* dengan *patience* 10 epoch, artinya pelatihan dihentikan lebih awal apabila nilai *validation loss* tidak mengalami perbaikan selama 10 epoch berturut-turut.

Tabel 3.2 Experimental Setup Model

No	Rasio Train:Test	Pendekatan	Jumlah Skenario	Max Epoch
1	50:50	Baseline (LR 0,001 Bs 32 Dr 0,5)	1	50
2	50:50	Grid Search (8 kombinasi: LR x Batch Size x Dropout Rate)	8	50
3	60:40	Baseline (LR 0,001 Bs 32 Dr 0,5)	1	50
4	60:40	Grid Search (8 kombinasi: LR x Batch Size x Dropout Rate)	8	50
5	70:30	Baseline (LR 0,001 Bs 32 Dr 0,5)	1	50
6	70:30	Grid Search (8 kombinasi: LR x Batch Size x Dropout Rate)	8	50
7	80:20	Baseline (LR 0,001 Bs 32 Dr 0,5)	1	50
8	80:20	Grid Search (8 kombinasi: LR x Batch Size x Dropout Rate)	8	50
9	90:10	Baseline (LR 0,001 Bs 32 Dr 0,5)	1	50
10	90:10	Grid Search (8 kombinasi: LR x Batch Size x Dropout Rate)	8	50

3.6 Evaluasi

Evaluasi sistem dilakukan untuk mengetahui seberapa baik performa model *Convolutional Neural Network* (CNN) berbasis arsitektur VGG16 dalam

mengklasifikasikan penyakit daun jagung. Setelah proses pelatihan model selesai dilakukan pada setiap skenario eksperimen, model kemudian diuji menggunakan data uji (*test set*) yang sebelumnya tidak digunakan selama proses pelatihan. Pengujian ini bertujuan untuk melihat kemampuan model dalam melakukan generalisasi terhadap data baru yang belum pernah dipelajari sebelumnya. Hasil prediksi model kemudian dibandingkan dengan label sebenarnya untuk menghasilkan *confusion matrix*. *Confusion matrix* digunakan untuk melihat distribusi prediksi benar dan salah pada setiap kelas klasifikasi. Melalui matriks ini dapat diketahui jumlah data yang berhasil diklasifikasikan dengan benar maupun yang mengalami kesalahan klasifikasi. Selain itu, *confusion matrix* juga membantu dalam mengidentifikasi pola kesalahan yang mungkin terjadi, misalnya apakah model sering keliru membedakan antara dua jenis penyakit tertentu yang memiliki karakteristik visual yang mirip.

Selain menggunakan *confusion matrix*, evaluasi performa model juga dilakukan menggunakan beberapa metrik kuantitatif, yaitu *accuracy*, *precision*, *recall*, dan *F1-score*. Penggunaan beberapa metrik evaluasi ini bertujuan agar analisis performa model tidak hanya bergantung pada satu indikator saja. Dengan demikian, kemampuan model dapat dianalisis secara lebih komprehensif dari berbagai aspek kinerja klasifikasi. *Accuracy* digunakan untuk mengukur proporsi jumlah prediksi yang benar terhadap seluruh data pengujian. Nilai ini memberikan gambaran umum mengenai tingkat keberhasilan model dalam melakukan klasifikasi secara keseluruhan. Namun, *accuracy* saja tidak selalu cukup untuk menggambarkan performa model secara detail, terutama pada kasus klasifikasi

multikelas. Oleh karena itu digunakan pula metrik *precision* dan *recall*. *Precision* menunjukkan seberapa banyak prediksi positif yang benar-benar sesuai dengan label sebenarnya, sedangkan *recall* menunjukkan seberapa banyak data dari suatu kelas yang berhasil dikenali oleh model.

Untuk memberikan keseimbangan antara *precision* dan *recall*, digunakan metrik *F1-score* yang merupakan rata-rata harmonik dari kedua nilai tersebut. Nilai *F1-score* memberikan gambaran performa model yang lebih seimbang terutama ketika terdapat perbedaan distribusi data antar kelas. Dengan menggunakan kombinasi beberapa metrik evaluasi tersebut, performa model CNN pada setiap skenario eksperimen dapat dianalisis secara lebih objektif dan komprehensif. Agar evaluasi lebih jelas, pada penelitian ini digunakan beberapa rumus metrik evaluasi sebagai berikut.

1. *Accuracy* menunjukkan proporsi prediksi yang benar terhadap seluruh data pengujian.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.1)$$

2. *Precision* mengukur tingkat ketepatan model dalam memprediksi suatu kelas.

$$Precision = \frac{TP}{TP+FP} \quad (3.2)$$

3. *Recall* menunjukkan kemampuan model dalam menemukan seluruh data yang benar dari suatu kelas..

$$Recall = \frac{TP}{TP+FN} \quad (3.3)$$

4. *F1-score* merupakan rata-rata harmonik dari precision dan recall.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3.4)$$

Keterangan:

TP (True Positive) : Data yang diprediksi benar sebagai kelas tertentu
 TN (True Negative) : Data yang diprediksi benar sebagai bukan kelas tersebut
 FP (False Positive) : Data yang salah diprediksi sebagai kelas tersebut
 FN (False Negative) : Data yang seharusnya kelas tersebut tetapi diprediksi kelas lain

Dalam penelitian ini, karena klasifikasi terdiri dari empat kelas (*multiclass*), perhitungan *precision*, *recall*, dan *F1-score* dilakukan untuk masing-masing kelas, kemudian dirata-ratakan. Pendekatan ini penting agar evaluasi tidak bias terhadap kelas tertentu saja. *Confusion matrix* membantu melihat pola kesalahan klasifikasi secara visual, misalnya apakah model sering tertukar antara *Northern Leaf Blight* dan *Gray Leaf Spot*. Sementara itu, *precision* dan *recall* memberikan gambaran yang lebih rinci mengenai kualitas prediksi pada tiap kelas, sehingga evaluasi tidak hanya bergantung pada nilai akurasi keseluruhan.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Preprocessing

Sebelum masuk ke proses pelatihan model, ada beberapa tahap persiapan data yang perlu dilakukan terlebih dahulu. Tahap ini penting karena citra dari dataset yang digunakan tidak selalu dalam kondisi yang seragam ukuran piksel berbeda-beda, rentang nilai intensitasnya pun belum tentu konsisten. Kalau langsung dimasukkan ke model VGG16 tanpa diproses dulu, hasilnya tidak akan optimal. Oleh karena itu, *preprocessing* menjadi langkah awal yang tidak bisa dilewatkan sebelum eksperimen dimulai.

4.1.1 Hasil Resize dan Normalisasi Citra

Arsitektur VGG16 yang digunakan dalam penelitian ini membutuhkan input citra dengan ukuran tetap, yaitu 224×224 piksel. Karena dataset yang digunakan berisi citra dengan berbagai ukuran asli yang tidak seragam, proses *resize* dilakukan terlebih dahulu pada seluruh citra agar memenuhi spesifikasi input tersebut. Proses ini dilakukan menggunakan fungsi *load_img* dari Keras dengan parameter `target_size=(224, 224)`. Setelah *resize*, seluruh nilai piksel dinormalisasi dari rentang 0–255 menjadi 0–1 dengan cara membagi setiap nilai piksel dengan 255. Normalisasi ini dilakukan dengan tujuan mempercepat konvergensi saat proses pelatihan berlangsung, sekaligus menjaga kestabilan pembaruan bobot di setiap iterasi. Tanpa normalisasi, perbedaan skala nilai piksel antar citra dapat menyebabkan gradien yang tidak stabil selama *backpropagation*. Secara visual,

tampilan citra tidak mengalami perubahan yang mencolok pada warna dan tekstur daun masih terlihat jelas. Perbedaannya hanya pada dimensi dan rentang nilai yang sudah disesuaikan dengan kebutuhan model. Hasil dari proses *resize* dan normalisasi dapat dilihat pada gambar berikut.



Gambar 4.1 Perbandingan citra sebelum dan sesudah resize serta normalisasi

4.1.2 Hasil Augmentasi Data Latih

Salah satu tantangan dalam penelitian ini adalah jumlah dataset yang relatif terbatas, yaitu hanya 1.500 citra yang dibagi ke dalam empat kelas. Dengan jumlah sebesar itu, terutama pada skenario pembagian data 90:10 di mana data uji hanya tersisa 150 citra, risiko *overfitting* menjadi cukup nyata. Model bisa saja "hafal" data latih tapi gagal menggeneralisasi pada data baru. Untuk mengatasi hal tersebut, augmentasi data diterapkan khusus pada data latih. Augmentasi dilakukan secara acak menggunakan ImageDataGenerator dari Keras dengan beberapa variasi transformasi, meliputi rotasi gambar hingga 20 derajat, zoom acak sebesar 20%, serta horizontal flip atau pencerminan gambar secara horizontal. Semua transformasi ini dipilih karena relevan secara agronomis daun jagung di lapangan bisa saja terlihat dari berbagai sudut dan jarak pengambilan gambar yang berbeda.

Satu hal yang perlu dicatat, augmentasi hanya diterapkan pada data latih, bukan pada data uji. Hal ini disengaja agar evaluasi model tetap mencerminkan kondisi nyata seberapa baik model mengenali citra yang belum pernah dilihat sebelumnya, bukan versi yang sudah dimanipulasi. Jika augmentasi juga diterapkan ke data uji, hasil evaluasi bisa jadi tidak mencerminkan performa model yang sebenarnya. Contoh hasil augmentasi yang dihasilkan selama proses pelatihan dapat dilihat pada Gambar 4.2. Terlihat bahwa meski citra mengalami rotasi dan pencerminan, karakteristik visual penyakit seperti bercak, warna, dan tekstur permukaan daun masih dapat dikenali dengan jelas. Ini menunjukkan bahwa augmentasi yang diterapkan tidak merusak informasi diagnostik yang relevan dalam citra.

Contoh Citra Setelah Preprocessing dan Augmentasi



Gambar 4.2 Contoh hasil augmentasi data latih (rotasi, zoom, dan horizontal flip)

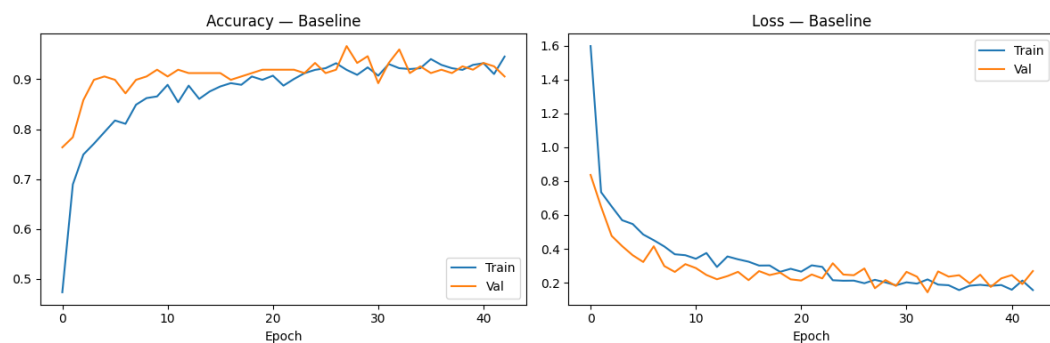
4.2 Hasil Eksperimen

Eksperimen dalam penelitian ini dirancang dengan dua pendekatan pelatihan yang dijalankan pada lima variasi rasio pembagian data, sehingga total terdapat 45 skenario pengujian. Pendekatan pertama adalah *baseline*, di mana model VGG16 dilatih menggunakan *learning rate* sebesar 0,001 *batch size* 32, dan *dropout rate* 0,5, dengan maksimum 50 epoch. Pendekatan kedua adalah *hyperparameter tuning* menggunakan metode *Grid Search*, yang menguji kombinasi tiga parameter yaitu *learning rate* (0,001 dan 0,0005), *batch size* (16 dan 32), serta *dropout rate* (0,3 dan 0,5) yang menghasilkan 8 kombinasi per rasio. Kombinasi dengan *F1-score* tertinggi pada data uji kemudian dipilih sebagai konfigurasi optimal, sebagaimana prinsip kerja *Grid Search* yang secara sistematis mengevaluasi seluruh kombinasi *hyperparameter* yang telah ditentukan untuk menemukan konfigurasi terbaik (Sholeh et al., 2025). Mekanisme *early stopping* dengan *patience* 10 epoch diterapkan pada kedua pendekatan untuk mencegah *overfitting* sekaligus menghemat waktu komputasi.

Pada setiap skenario, model dievaluasi menggunakan *confusion matrix* dan *classification report* yang mencakup nilai *precision*, *recall*, dan *F1-score* per kelas. Keempat kelas yang diklasifikasikan adalah *Gray Leaf Spot* (bercak daun), *Common Rust* (karat daun), *Northern Leaf Blight* (hawar daun), dan *Healthy* (daun sehat). Hasil dari masing-masing rasio diuraikan secara berurutan pada sub-bab berikut, dengan tabel lengkap memuat seluruh 9 skenario (1 *baseline* + 8 *Grid Search*) per rasio.

4.2.1 Eksperimen Rasio 50:50

Pada rasio 50:50, dataset dibagi secara merata sehingga data latih berjumlah 602 citra dan data uji sebanyak 750 citra. Rasio ini menjadi kondisi paling terbatas dalam penelitian ini karena model hanya mendapat separuh data untuk belajar. Pada skenario *baseline* ($LR=0,001$, $Bs=32$, $Dr=0,5$), model berhenti lebih awal akibat mekanisme *Early Stopping* pada epoch ke-43 dan menghasilkan akurasi 93,60% dengan *F1-score* 93,61%. Training history pada Gambar 4.3 menunjukkan kurva *train* dan *validation* yang berhimpitan rapat sepanjang pelatihan, mengindikasikan model tidak mengalami *overfitting* signifikan meski data latih relatif terbatas.



Gambar 4.3 Grafik training history rasio 50:50

Proses Grid Search pada rasio ini menguji 8 kombinasi *learning rate*, *batch size*, dan *dropout rate*. Hasil selengkapnya ditunjukkan pada Tabel 4.1. Kombinasi GS-5 ($LR=0,0005$ | $Bs=16$ | $Dr=0,3$) tercatat sebagai konfigurasi terbaik dengan akurasi 93,47% dan *F1-score* 93,52%, sedikit melampaui sebagian kombinasi lain namun secara keseluruhan masih berada di bawah skenario *baseline* (*F1-score* 93,61%), dengan selisih yang sangat kecil (-0,09%). Ini menjadikan rasio 50:50 satu-satunya rasio dalam penelitian ini di mana hyperparameter tuning tidak

berhasil melampaui performa *baseline*, meski selisihnya berada dalam margin variasi training yang wajar dan tidak dapat dikatakan signifikan secara praktis.

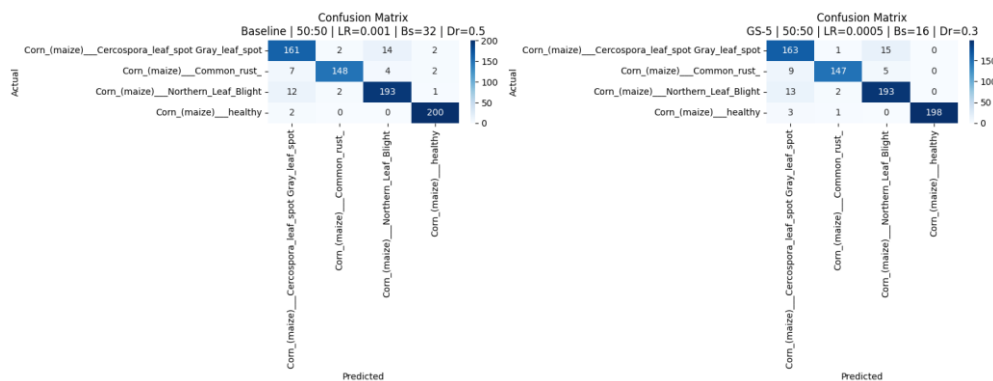
Tabel 4.1 Hasil evaluasi model pada rasio 50:50

Skenario	LR	Batch Size	Dropout	Accuracy	Precision	Recall	F1-Score
Baseline	0,001	32	0,5	0,9360	0,9366	0,9360	0,9361
GS-1	0,001	16	0,3	0,9333	0,9356	0,9333	0,9339
GS-2	0,001	16	0,5	0,9107	0,9232	0,9107	0,9126
GS-3	0,001	32	0,3	0,9320	0,9336	0,9320	0,9325
GS-4	0,001	32	0,5	0,9173	0,9210	0,9173	0,9183
GS-5	0,0005	16	0,3	0,9347	0,9365	0,9347	0,9352
GS-6	0,0005	16	0,5	0,9307	0,9321	0,9307	0,9310
GS-7	0,0005	32	0,3	0,9307	0,9339	0,9307	0,9315
GS-8	0,0005	32	0,5	0,9280	0,9289	0,9280	0,9283

Pola lain yang menarik dari Tabel 4.1 adalah pengaruh *dropout rate*: pada hampir seluruh kombinasi *learning rate* dan *batch size* yang sama, *dropout* 0,3 secara konsisten menghasilkan performa lebih baik dibandingkan *dropout* 0,5 (misalnya GS-1 vs GS-2, GS-3 vs GS-4, GS-5 vs GS-6, dan GS-7 vs GS-8). Pola ini mengindikasikan bahwa regularisasi yang terlalu kuat ($Dr=0,5$) kurang sesuai untuk ukuran data latih pada rasio ini, karena terlalu banyak neuron yang dinonaktifkan justru menghambat proses pembelajaran fitur yang relevan.

Ditinjau dari performa per kelas pada kombinasi terbaik (GS-5), kelas *Healthy* memperoleh *F1-score* tertinggi sebesar 99%, mengonfirmasi bahwa karakteristik visual daun sehat berwarna hijau merata tanpa lesi paling mudah

dibedakan dari ketiga kelas penyakit lainnya. Kelas *Common Rust* juga menunjukkan performa kuat dengan *precision* 97%, menandakan model sangat jarang salah mengenali kelas lain sebagai *Common Rust*. Sebaliknya, kelas *Gray Leaf Spot* tercatat sebagai kelas dengan *F1-score* terendah, yaitu 89%, dengan 15 dari 179 citra justru terklasifikasi sebagai *Northern Leaf Blight* berdasarkan *confusion matrix* pada Gambar 4.4. Kekeliruan ini cukup masuk akal secara visual, karena kedua penyakit tersebut sama-sama menimbulkan bercak memanjang pada permukaan daun, sehingga model lebih sulit membedakannya dibandingkan kelas *Healthy* atau *Common Rust* yang memiliki ciri visual lebih khas.

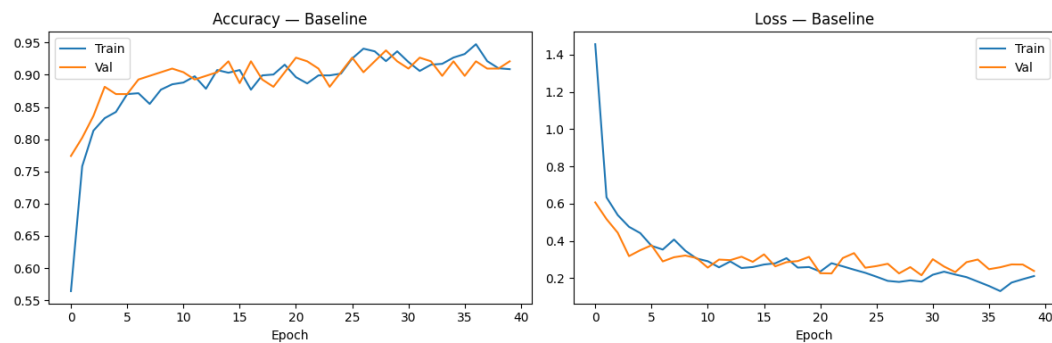


Gambar 4.4 Confusion matrix rasio 50:50 - baseline & GS (kombinasi terbaik)

4.2.2 Eksperimen Rasio 60:40

Pada rasio 60:40, data latih bertambah menjadi 723 citra dan data uji menjadi 600 citra. Skenario *baseline* (LR=0,001, Bs=32, Dr=0,5) menghasilkan akurasi 93,50% dengan *F1-score* 93,53%, berhenti lebih awal pada epoch ke-40 akibat *Early Stopping*. Performa ini sedikit lebih rendah dibandingkan *baseline* pada rasio 50:50 (93,61%) meski porsi data latih lebih besar, kemungkinan dipengaruhi oleh faktor keacakan inisialisasi bobot yang berbeda pada setiap sesi

pelatihan, sebagaimana lazim terjadi pada proses *training deep learning* yang bersifat stokastik.



Gambar 4.5 Grafik training history baseline rasio 60:40

Berbeda dengan rasio 50:50, proses Grid Search pada rasio 60:40 berhasil melampaui *baseline*. Kombinasi GS-7 (LR=0,0005 | Bs=32 | Dr=0,3) tercatat sebagai konfigurasi terbaik dengan akurasi 93,67% dan *F1-score* 93,74%, meningkat +0,21% dibandingkan *baseline*. Hasil selengkapnya ditunjukkan pada Tabel 4.2. Menariknya, kombinasi dengan *learning rate* 0,001 dan *batch size* 16 (GS-1) justru menghasilkan performa paling rendah pada rasio ini (*F1-score* 89,36%), jauh di bawah kombinasi lainnya — pola ini berbeda dari rasio 50:50 di mana kombinasi serupa (GS-1) justru termasuk yang terbaik kedua. Hal ini menunjukkan bahwa interaksi antar hyperparameter dapat bersifat spesifik terhadap rasio data, sehingga konfigurasi optimal pada satu rasio tidak selalu dapat digeneralisasi langsung ke rasio lainnya.

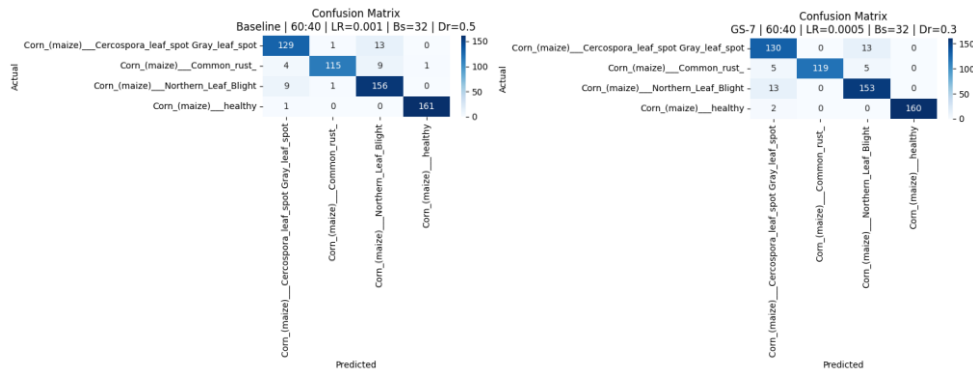
Tabel 4.2 Hasil evaluasi model pada rasio 60:40

Skenario	LR	Batch Size	Dropout	Accuracy	Precision	Recall	F1-Score
Baseline	0,001	32	0,5	0,9350	0,9371	0,9350	0,9353

GS-1	0,001	16	0,3	0,8917	0,9023	0,8917	0,8936
GS-2	0,001	16	0,5	0,9233	0,9250	0,9233	0,9237
GS-3	0,001	32	0,3	0,9317	0,9339	0,9317	0,9322
GS-4	0,001	32	0,5	0,9133	0,9213	0,9133	0,9148
GS-5	0,0005	16	0,3	0,9200	0,9252	0,9200	0,9204
GS-6	0,0005	16	0,5	0,9267	0,9311	0,9267	0,9269
GS-7	0,0005	32	0,3	0,9367	0,9391	0,9367	0,9374
GS-8	0,0005	32	0,5	0,9300	0,9313	0,9300	0,9304

Konsisten dengan rasio 50:50, *dropout* 0,3 kembali mengungguli *dropout* 0,5 pada mayoritas pasangan kombinasi yang sebanding (GS-1 vs GS-2 menjadi pengecualian, namun GS-3 vs GS-4, GS-5 vs GS-6, dan GS-7 vs GS-8 tetap menunjukkan keunggulan $Dr=0,3$). Pola ini memperkuat indikasi bahwa regularisasi *dropout* yang lebih ringan secara umum lebih sesuai untuk karakteristik dataset penelitian ini.

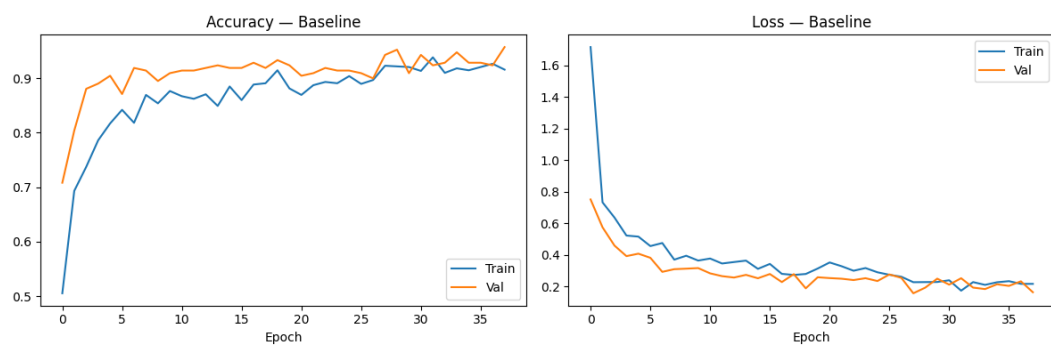
Ditinjau dari performa per kelas pada kombinasi terbaik (GS-7), kelas *Healthy* kembali mencatat *F1-score* tertinggi (99%), sementara kelas *Common Rust* mencapai *precision* sempurna 100%, tidak ada satu pun citra dari kelas lain yang salah diklasifikasikan sebagai *Common Rust*. Kelas *Gray Leaf Spot* tetap menjadi yang paling menantang dengan *F1-score* 89%, mengulangi pola yang sama seperti pada rasio 50:50. Berdasarkan *confusion matrix* pada Gambar 4.6, kekeliruan klasifikasi *Gray Leaf Spot* sebagian besar tertukar dengan *Northern Leaf Blight*, mengonfirmasi bahwa kedua kelas penyakit ini secara konsisten menjadi pasangan kelas yang paling sulit dibedakan oleh model di seluruh rasio yang diuji sejauh ini.



Gambar 4.6 Confusion matrix rasio 60:40 - baseline & GS (kombinasi terbaik)

4.2.3 Eksperimen Rasio 70:30

Pada rasio 70:30 data latih berjumlah 1050 citra dan data uji sebanyak 450 citra. Skenario *baseline* (LR=0,001, Bs=32, Dr=0,5) menghasilkan akurasi 92,44% dengan *F1-score* 92,51%, berhenti pada epoch ke-38 akibat *Early Stopping*. Performa ini sedikit lebih rendah dibandingkan dua rasio sebelumnya, meski data latih jauh lebih banyak dari rasio 50:50 maupun 60:40 sekali lagi mengindikasikan bahwa variasi inisialisasi bobot turut mempengaruhi hasil akhir pelatihan, bukan murni fungsi dari jumlah data.



Gambar 4.7 Grafik training history baseline rasio 70:30

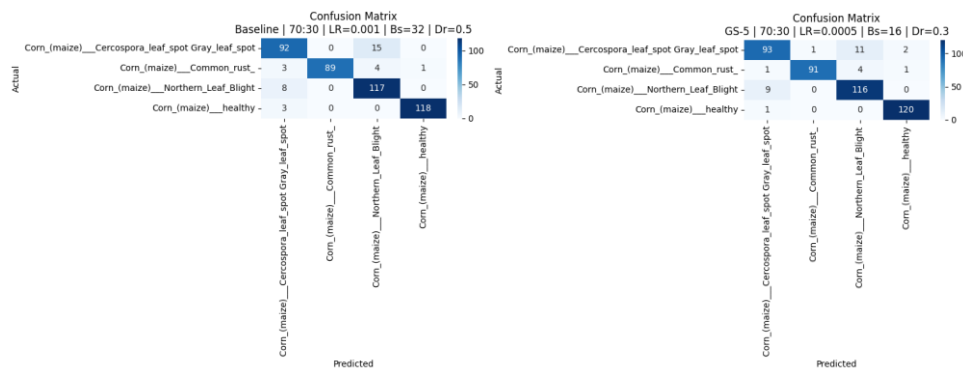
Proses Grid Search pada rasio ini kembali menunjukkan peningkatan dibandingkan *baseline*. Kombinasi GS-5 (LR=0,0005 | Bs=16 | Dr=0,3) tercatat

sebagai konfigurasi terbaik dengan akurasi 93,33% dan *F1-score* 93,34%, meningkat +0,83% dibandingkan *baseline* peningkatan paling besar yang tercatat sejauh ini dibandingkan rasio 50:50 dan 60:40. Hasil selengkapnya ditunjukkan pada Tabel 4.3.

Tabel 4.3 Hasil evaluasi model pada rasio 70:30

Skenario	LR	Batch Size	Dropout	Accuracy	Precision	Recall	F1-Score
Baseline	0,001	32	0,5	0,9244	0,9275	0,9244	0,9251
GS-1	0,001	16	0,3	0,9267	0,9286	0,9267	0,9273
GS-2	0,001	16	0,5	0,9111	0,9155	0,9111	0,9121
GS-3	0,001	32	0,3	0,9178	0,9204	0,9178	0,9187
GS-4	0,001	32	0,5	0,9222	0,9248	0,9222	0,9230
GS-5	0,0005	16	0,3	0,9333	0,9341	0,9333	0,9334
GS-6	0,0005	16	0,5	0,9200	0,9224	0,9200	0,9205
GS-7	0,0005	32	0,3	0,9200	0,9268	0,9200	0,9205
GS-8	0,0005	32	0,5	0,9289	0,9309	0,9289	0,9292

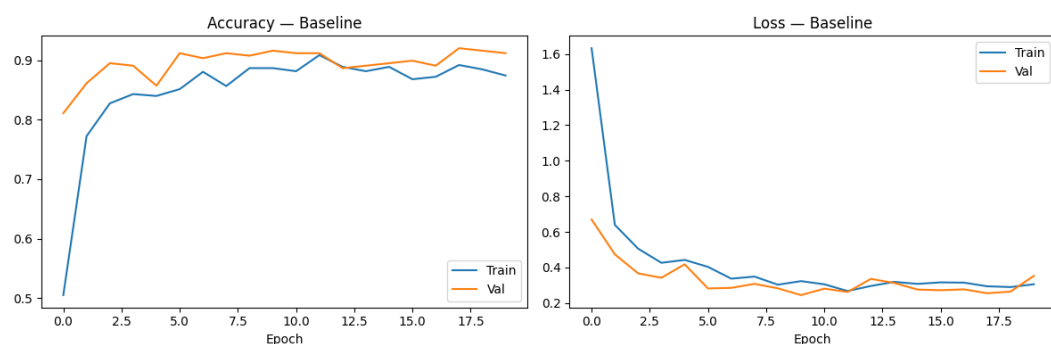
Performa per kelas pada kombinasi terbaik (GS-5) menunjukkan peningkatan paling mencolok pada kelas *Common Rust*, dengan *F1-score* naik dari sekitar 91% pada *baseline* menjadi 96%. Kelas *Healthy* tetap konsisten dengan *F1-score* tertinggi (98%), sementara *Gray Leaf Spot* kembali menjadi kelas paling sulit dengan *F1-score* 88% melanjutkan pola yang sudah teramati pada dua rasio sebelumnya. Berdasarkan *confusion matrix* pada Gambar 4.8, sebagian besar kekeliruan klasifikasi *Gray Leaf Spot* tetap tertukar dengan *Northern Leaf Blight*, mengonfirmasi bahwa pasangan kelas ini secara konsisten menjadi yang paling menantang di seluruh rasio.



Gambar 4.8 Confusion matrix rasio 70:30 - baseline & GS (kombinasi terbaik)

4.2.4 Eksperimen Rasio 80:20

Pada rasio 80:20, data latih mencapai 962 citra dan data uji berkurang menjadi 300 citra. Skenario *baseline* (LR=0,001, Bs=32, Dr=0,5) menghasilkan akurasi 91,33% dengan *F1-score* 91,37%, berhenti jauh lebih awal pada epoch ke-20 dibandingkan tiga rasio sebelumnya. Berhentinya pelatihan yang lebih cepat ini kemungkinan terjadi karena data validasi yang lebih sedikit (20% dari data latih) membuat nilai *validation loss* lebih fluktuatif antar-epoch, sehingga mekanisme *Early Stopping* terpicu lebih cepat.



Gambar 4.9 Grafik training history rasio baseline rasio 80:20

Pada rasio ini, Grid Search menghasilkan temuan yang berbeda dari dua rasio sebelumnya: kombinasi terbaik justru menggunakan *learning rate default* (0,001), bukan 0,0005. Kombinasi GS-1 (LR=0,001 | Bs=16 | Dr=0,3) tercatat

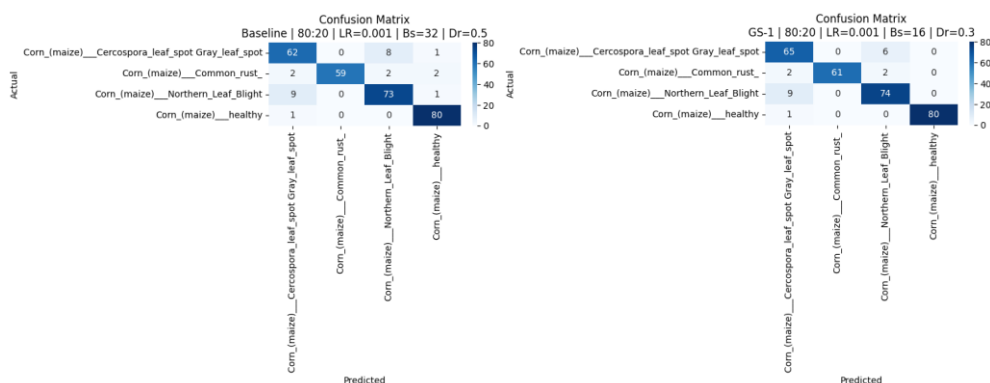
sebagai konfigurasi terbaik dengan akurasi 93,33% dan *F1-score* 93,42%, meningkat +2,05% dibandingkan *baseline* peningkatan terbesar kedua dalam penelitian ini. Hasil selengkapnya ditunjukkan pada Tabel 4.4. Pola ini mengindikasikan bahwa pada porsi data latih yang lebih besar, *learning rate* yang lebih besar (0,001) dapat bekerja optimal selama diimbangi dengan *batch size* yang lebih kecil (16) dan regularisasi *dropout* yang ringan (0,3).

Tabel 4.4 Hasil evaluasi model pada rasio 80:20

Skenario	LR	Batch Size	Dropout	Accuracy	Precision	Recall	F1-Score
Baseline	0,001	32	0,5	0,9133	0,9154	0,9133	0,9137
GS-1	0,001	16	0,3	0,9333	0,9361	0,9333	0,9342
GS-2	0,001	16	0,5	0,9167	0,9216	0,9167	0,9174
GS-3	0,001	32	0,3	0,9300	0,9328	0,9300	0,9307
GS-4	0,001	32	0,5	0,9267	0,9288	0,9267	0,9268
GS-5	0,0005	16	0,3	0,9200	0,9232	0,9200	0,9211
GS-6	0,0005	16	0,5	0,9133	0,9195	0,9133	0,9146
GS-7	0,0005	32	0,3	0,9133	0,9155	0,9133	0,9139
GS-8	0,0005	32	0,5	0,9267	0,9290	0,9267	0,9275

Performa per kelas pada kombinasi terbaik (GS-1) menunjukkan kelas *Common Rust* mencapai *precision* sempurna 100%, sementara *recall*-nya juga tinggi (94%), menghasilkan *F1-score* 97%. Kelas *Healthy* tetap konsisten di posisi teratas dengan *F1-score* 99%. Kelas *Gray Leaf Spot* mengalami sedikit peningkatan dibandingkan rasio-rasio sebelumnya, dengan *F1-score* 88%, namun tetap menjadi kelas dengan performa terendah. Berdasarkan Gambar 4.10, kekeliruan klasifikasi

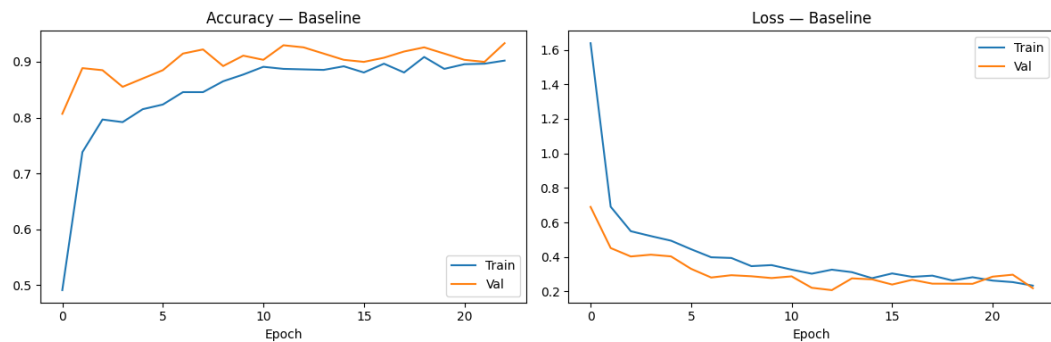
pada kelas ini didominasi oleh citra yang terprediksi sebagai *Northern Leaf Blight*, melanjutkan pola yang konsisten teramati di seluruh rasio.



Gambar 4.10 Confusion matrix rasio 80:20 - baseline & GS (kombinasi terbaik)

4.2.5 Eksperimen Rasio 90:10

Rasio 90:10 adalah konfigurasi dengan data latih terbanyak, yaitu 1.081 citra, sementara data uji hanya 150 citra. Skenario *baseline* (LR=0,001, Bs=32, Dr=0,5) menghasilkan akurasi 90,67% dengan *F1-score* 90,73%, berhenti pada epoch ke-23 akibat *Early Stopping*. Performa ini menjadi yang terendah di antara seluruh rasio untuk skenario *baseline*, meski memiliki porsi data latih terbanyak. Hal ini sejalan dengan temuan penelitian sebelumnya yang menunjukkan bahwa data uji yang terlalu sedikit (hanya 150 citra, atau sekitar 37-38 citra per kelas) membuat estimasi performa model menjadi kurang stabil dan lebih sensitif terhadap kesalahan klasifikasi pada masing-masing citra individual (Bichri et al., 2024).



Gambar 4.11 Grafik training history baseline rasio 90:10

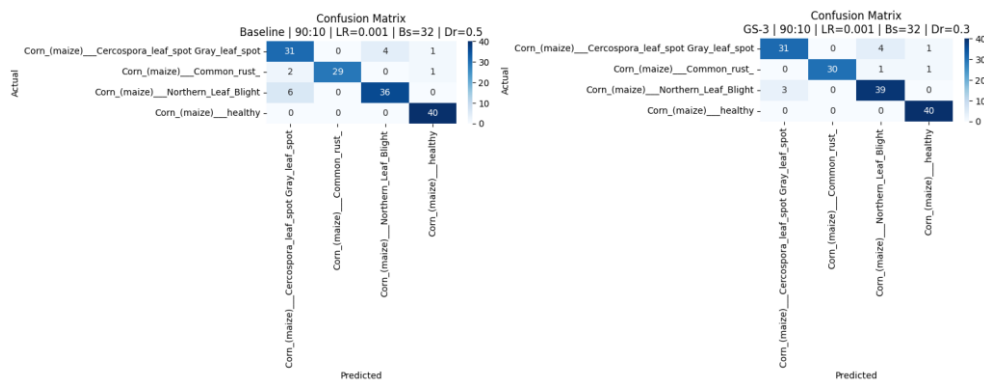
Grid Search pada rasio ini memberikan peningkatan paling besar di antara seluruh rasio yang diuji. Kombinasi GS-3 (LR=0,001 | Bs=32 | Dr=0,3) tercatat sebagai konfigurasi terbaik dengan akurasi 93,33% dan *F1-score* 93,31%, meningkat +2,58% dibandingkan *baseline*. Hasil selengkapnya ditunjukkan pada Tabel 4.5. Menariknya, kombinasi terbaik pada rasio ini menggunakan parameter yang identik dengan *baseline* kecuali pada *dropout rate* (0,3 dibandingkan 0,5 pada *baseline*), menunjukkan bahwa penyesuaian regularisasi saja, tanpa mengubah *learning rate* atau *batch size*, sudah cukup untuk memberikan peningkatan performa yang signifikan pada rasio dengan data uji paling terbatas ini.

Tabel 4.5 Hasil evaluasi model pada rasio 90:10

Skenario	LR	Batch Size	Dropout	Accuracy	Precision	Recall	F1-Score
Baseline	0,001	32	0,5	0,9067	0,9101	0,9067	0,9073
GS-1	0,001	16	0,3	0,9133	0,9175	0,9133	0,9142
GS-2	0,001	16	0,5	0,9067	0,9080	0,9067	0,9067
GS-3	0,001	32	0,3	0,9333	0,9343	0,9333	0,9331
GS-4	0,001	32	0,5	0,9267	0,9283	0,9267	0,9270
GS-5	0,0005	16	0,3	0,9200	0,9229	0,9200	0,9209

GS-6	0,0005	16	0,5	0,9267	0,9284	0,9267	0,9269
GS-7	0,0005	32	0,3	0,9267	0,9273	0,9267	0,9266
GS-8	0,0005	32	0,5	0,9133	0,9175	0,9133	0,9142

Performa per kelas pada kombinasi terbaik (GS-3) menunjukkan kelas *Healthy* mencapai *recall* sempurna 100% seluruh 40 citra daun sehat pada data uji berhasil diklasifikasikan dengan benar. Kelas *Common Rust* kembali mencatat *precision* sempurna 100%. Kelas *Gray Leaf Spot* tetap menjadi yang paling menantang dengan *F1-score* 89%, konsisten dengan pola yang teramati di keempat rasio sebelumnya. Berdasarkan Gambar 4.12, dari 36 citra *Gray Leaf Spot* pada data uji, terdapat kekeliruan klasifikasi ke kelas *Northern Leaf Blight*, mengonfirmasi sekali lagi bahwa kedua kelas penyakit ini secara konsisten menjadi pasangan yang paling sulit dibedakan oleh model di seluruh rasio pembagian data yang diuji dalam penelitian ini.



Gambar 4.12 Confusion matrix rasio 90:10 - baseline & GS (kombinasi terbaik)

4.3 Perbandingan Hasil Model

Setelah seluruh 45 skenario eksperimen dijalankan pada kelima rasio pembagian data, bagian ini menyajikan perbandingan menyeluruh untuk melihat

pola umum yang muncul lintas rasio. Tujuannya bukan sekadar mencari angka tertinggi, tapi lebih ke memahami pola apakah semakin besar rasio data latih selalu menghasilkan model yang lebih baik, dan apakah Grid Search secara konsisten memberikan peningkatan dibanding *baseline*. Ke-45 skenario tersebut terdiri dari 5 skenario *baseline* dan 40 skenario Grid Search yang menguji kombinasi tiga hyperparameter (*learning rate, batch size, dropout rate*) pada masing-masing rasio. Tabel 4.6 merangkum hasil keseluruhan dari semua skenario, dengan baris bertanda kuning menunjukkan kombinasi terbaik (F1-score tertinggi) pada masing-masing rasio.

Tabel 4.6 Perbandingan Baseline & Grid Search pada seluruh rasio

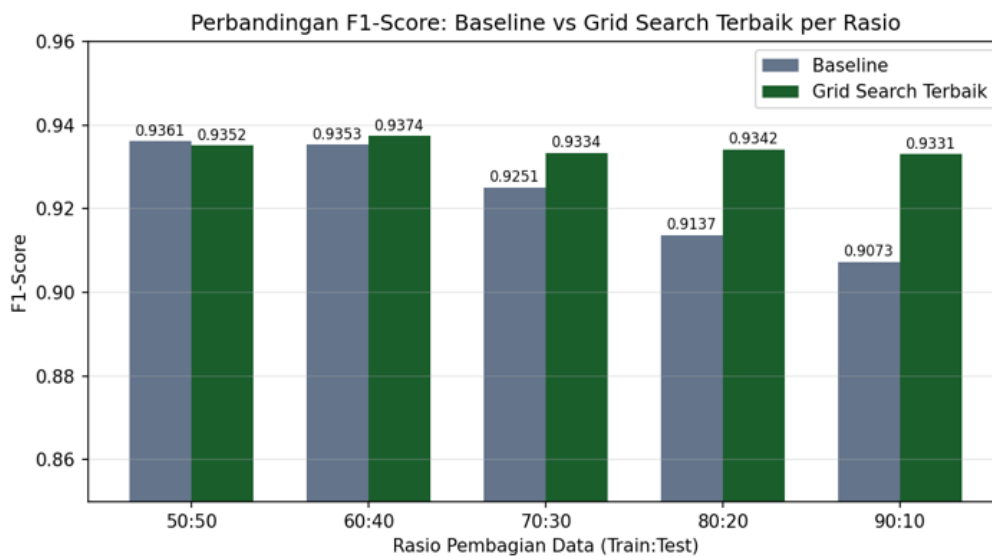
No	Skenario	Rasio	LR	Bs	Dr	Accuracy	Precision	Recall	F1-Score
1	Baseline	50:50	0,001	32	0,5	0,9360	0,9366	0,9360	0,9361
2	GS-1	50:50	0,001	16	0,3	0,9333	0,9356	0,9333	0,9339
3	GS-2	50:50	0,001	16	0,5	0,9107	0,9232	0,9107	0,9126
4	GS-3	50:50	0,001	32	0,3	0,9320	0,9336	0,9320	0,9325
5	GS-4	50:50	0,001	32	0,5	0,9173	0,9210	0,9173	0,9183
6	GS-5	50:50	0,0005	16	0,3	0,9347	0,9365	0,9347	0,9352
7	GS-6	50:50	0,0005	16	0,5	0,9307	0,9321	0,9307	0,9310
8	GS-7	50:50	0,0005	32	0,3	0,9307	0,9339	0,9307	0,9315
9	GS-8	50:50	0,0005	32	0,5	0,9280	0,9289	0,9280	0,9283
10	Baseline	60:40	0,001	32	0,5	0,9350	0,9371	0,9350	0,9353
11	GS-1	60:40	0,001	16	0,3	0,8917	0,9023	0,8917	0,8936
12	GS-2	60:40	0,001	16	0,5	0,9233	0,9250	0,9233	0,9237
13	GS-3	60:40	0,001	32	0,3	0,9317	0,9339	0,9317	0,9322
14	GS-4	60:40	0,001	32	0,5	0,9133	0,9213	0,9133	0,9148
15	GS-5	60:40	0,0005	16	0,3	0,9200	0,9252	0,9200	0,9204
16	GS-6	60:40	0,0005	16	0,5	0,9267	0,9311	0,9267	0,9269
17	GS-7	60:40	0,0005	32	0,3	0,9367	0,9391	0,9367	0,9374

18	GS-8	60:40	0,0005	32	0,5	0,9300	0,9313	0,9300	0,9304
19	Baseline	70:30	0,001	32	0,5	0,9244	0,9275	0,9244	0,9251
20	GS-1	70:30	0,001	16	0,3	0,9267	0,9286	0,9267	0,9273
21	GS-2	70:30	0,001	16	0,5	0,9111	0,9155	0,9111	0,9121
22	GS-3	70:30	0,001	32	0,3	0,9178	0,9204	0,9178	0,9187
23	GS-4	70:30	0,001	32	0,5	0,9222	0,9248	0,9222	0,9230
24	GS-5	70:30	0,0005	16	0,3	0,9333	0,9341	0,9333	0,9334
25	GS-6	70:30	0,0005	16	0,5	0,9200	0,9224	0,9200	0,9205
26	GS-7	70:30	0,0005	32	0,3	0,9200	0,9268	0,9200	0,9205
27	GS-8	70:30	0,0005	32	0,5	0,9289	0,9309	0,9289	0,9292
28	Baseline	80:20	0,001	32	0,5	0,9133	0,9154	0,9133	0,9137
29	GS-1	80:20	0,001	16	0,3	0,9333	0,9361	0,9333	0,9342
30	GS-2	80:20	0,001	16	0,5	0,9167	0,9216	0,9167	0,9174
31	GS-3	80:20	0,001	32	0,3	0,9300	0,9328	0,9300	0,9307
32	GS-4	80:20	0,001	32	0,5	0,9267	0,9288	0,9267	0,9268
33	GS-5	80:20	0,0005	16	0,3	0,9200	0,9232	0,9200	0,9211
34	GS-6	80:20	0,0005	16	0,5	0,9133	0,9195	0,9133	0,9146
35	GS-7	80:20	0,0005	32	0,3	0,9133	0,9155	0,9133	0,9139
36	GS-8	80:20	0,0005	32	0,5	0,9267	0,9290	0,9267	0,9275
37	Baseline	90:10	0,001	32	0,5	0,9067	0,9101	0,9067	0,9073
38	GS-1	90:10	0,001	16	0,3	0,9133	0,9175	0,9133	0,9142
39	GS-2	90:10	0,001	16	0,5	0,9067	0,9080	0,9067	0,9067
40	GS-3	90:10	0,001	32	0,3	0,9333	0,9343	0,9333	0,9331
41	GS-4	90:10	0,001	32	0,5	0,9267	0,9283	0,9267	0,9270
42	GS-5	90:10	0,0005	16	0,3	0,9200	0,9229	0,9200	0,9209
43	GS-6	90:10	0,0005	16	0,5	0,9267	0,9284	0,9267	0,9269
44	GS-7	90:10	0,0005	32	0,3	0,9267	0,9273	0,9267	0,9266
45	GS-8	90:10	0,0005	32	0,5	0,9133	0,9175	0,9133	0,9142

Dari tabel di atas, beberapa hal langsung terlihat. Grid Search berhasil mengungguli *baseline* pada empat dari lima rasio, yaitu 60:40, 70:30, 80:20, dan 90:10. Satu-satunya pengecualian ada di rasio 50:50, di mana *baseline* justru lebih unggul tipis dengan selisih F1-score sebesar 0,09 poin persen. *Dropout rate* 0,3 secara konsisten menjadi bagian dari kombinasi terbaik di seluruh lima rasio tanpa terkecuali, sementara *learning rate* dan *batch size* optimal bervariasi tergantung rasio.

Tabel 4.7 Rekap perbandingan Baseline & Grid Search Terbaik pada seluruh rasio

Rasio	F1 Baseline	F1 GS Terbaik	Delta	Keterangan
50:50	0,9361	0,9352 (GS-5: LR=0,0005 Bs=16 Dr=0,3)	-0,0009	Baseline lebih baik
60:40	0,9353	0,9374 (GS-7: LR=0,0005 Bs=32 Dr=0,3)	+0,0021	GS lebih baik
70:30	0,9251	0,9334 (GS-5: LR=0,0005 Bs=16 Dr=0,3)	+0,0083	GS lebih baik
80:20	0,9137	0,9342 (GS-1: LR=0,001 Bs=16 Dr=0,3)	+0,0205	GS lebih baik
90:10	0,9073	0,9331 (GS-3: LR=0,001 Bs=32 Dr=0,3)	+0,0258	GS lebih baik



Gambar 4.13 Grafik perbandingan F1-Score Baseline vs Grid Search Terbaik pada seluruh rasio

Berdasarkan Tabel 4.7 dan Gambar 4.13, terdapat beberapa pola penting yang dapat diamati. Pertama, hyperparameter tuning melalui Grid Search berhasil meningkatkan performa model pada empat dari lima rasio yang diuji (60:40, 70:30, 80:20, dan 90:10), dengan peningkatan *F1-score* berkisar antara +0,21% hingga +2,58%. Hanya pada rasio 50:50 performa Grid Search sedikit di bawah *baseline*, dengan selisih yang sangat kecil (-0,09%) dan dapat dianggap berada dalam batas variasi training yang wajar.

Kedua, terdapat kecenderungan bahwa semakin besar porsi data latih, semakin besar pula manfaat yang diperoleh dari hyperparameter tuning. Hal ini tercermin dari peningkatan F1-score yang terus membesar seiring rasio data latih yang lebih besar: dari +0,21% pada rasio 60:40, +0,83% pada rasio 70:30, +2,05% pada rasio 80:20, hingga +2,58% pada rasio 90:10. Pola ini cukup masuk akal secara teoritis: pada konfigurasi *baseline* yang menggunakan parameter *default*

tanpa optimasi, model cenderung membutuhkan penyesuaian hyperparameter yang lebih cermat ketika data uji yang tersedia untuk validasi performa semakin terbatas, sehingga ruang untuk perbaikan melalui tuning menjadi lebih besar.

Ketiga, ditinjau dari kombinasi hyperparameter yang terpilih sebagai konfigurasi terbaik di setiap rasio, *dropout rate* 0,3 selalu menjadi pilihan optimal di seluruh lima rasio tanpa terkecuali, mengonfirmasi temuan pada masing-masing sub-bab sebelumnya bahwa regularisasi yang lebih ringan secara konsisten lebih sesuai untuk karakteristik dataset penelitian ini. Sebaliknya, *learning rate* dan *batch size* optimal bervariasi antar rasio: *learning rate* 0,0005 terpilih pada rasio 50:50, 60:40, dan 70:30, sementara *learning rate* 0,001 (sama dengan *baseline*) terpilih pada rasio 80:20 dan 90:10. Variasi ini menunjukkan bahwa kombinasi hyperparameter yang optimal bersifat spesifik terhadap proporsi data, sehingga proses Grid Search pada setiap rasio tetap diperlukan dan tidak dapat digantikan dengan satu konfigurasi tunggal yang diasumsikan berlaku universal.

Keempat, dari sisi performa per kelas, pola yang konsisten teramati di seluruh 45 skenario adalah kelas *Healthy* secara konsisten memperoleh *F1-score* tertinggi (berkisar 97-99%), kelas *Common Rust* secara konsisten memperoleh *precision* tertinggi (97-100%), sementara kelas *Gray Leaf Spot* secara konsisten menjadi kelas dengan performa terendah (88-89%) di seluruh rasio yang diuji. Kekeliruan klasifikasi pada kelas *Gray Leaf Spot* hampir selalu tertukar dengan kelas *Northern Leaf Blight*, yang secara visual memang memiliki kemiripan berupa bercak atau lesi memanjang pada permukaan daun. Konsistensi pola ini di seluruh rasio data mengindikasikan bahwa kesulitan membedakan kedua kelas tersebut

bersifat fundamental pada karakteristik citra dataset, bukan disebabkan oleh keterbatasan jumlah data latih maupun pemilihan hyperparameter tertentu.

4.4 Pembahasan

Sistem Dari seluruh 45 skenario eksperimen yang telah dijalankan pada lima rasio pembagian data, ada beberapa hal yang perlu dibahas lebih dalam, bukan sekadar melaporkan angka, tapi mencoba memahami mengapa hasilnya bisa seperti itu. Ada beberapa pertanyaan yang menarik untuk dijawab, yaitu apakah hyperparameter tuning benar-benar membantu, kenapa kelas *Gray Leaf Spot* selalu paling sulit, dan bagaimana posisi hasil penelitian ini dibanding penelitian lain yang sudah ada sebelumnya.

Soal pengaruh Grid Search, jawabannya adalah ya, tapi dengan catatan. Dari lima rasio yang diuji, Grid Search berhasil meningkatkan performa di empat rasio (60:40, 70:30, 80:20, dan 90:10) dengan besaran delta yang bervariasi. Temuan ini sejalan dengan Sholeh et al. (2025) yang menyatakan bahwa optimasi *hyperparameter* menggunakan *Grid Search* terbukti mampu meningkatkan performa model secara signifikan dibandingkan penggunaan parameter *default*. Peningkatan terbesar terjadi di rasio 90:10 sebesar +2,58 poin persen, diikuti rasio 80:20 sebesar +2,05 poin persen, rasio 70:30 sebesar +0,83 poin persen, dan rasio 60:40 sebesar +0,21 poin persen. Hanya di rasio 50:50 Grid Search tidak berhasil melampaui *baseline*, dengan selisih yang sangat kecil yaitu 0,09 poin persen selisih yang berada dalam batas variasi training yang wajar.

Kalau ditelusuri lebih jauh, penyebab utama Grid Search bisa meningkatkan performa erat kaitannya dengan kesesuaian antara *learning rate* dan jumlah data latih. Terlihat bahwa pada rasio dengan data latih yang lebih sedikit (50:50, 60:40, 70:30), *learning rate* yang terpilih sebagai konfigurasi terbaik justru yang lebih kecil yaitu 0,0005. Sebaliknya, pada rasio dengan data latih yang lebih besar (80:20 dan 90:10), *learning rate default* 0,001 (sama dengan *baseline*) yang terpilih sebagai bagian dari konfigurasi terbaik. Morales-Hernández et al. (2023) menjelaskan bahwa sensitivitas model terhadap nilai *hyperparameter* bergantung pada karakteristik data dan kompleksitas tugas klasifikasi, sehingga tidak ada satu nilai yang berlaku optimal untuk semua kondisi. Pola ini mengindikasikan bahwa pemilihan *learning rate* yang optimal bergantung pada skala data latih yang tersedia, bukan sesuatu yang bisa diasumsikan tanpa proses pencarian sistematis.

Satu hal yang juga konsisten teramati di seluruh rasio adalah pengaruh *dropout rate*. Berbeda dengan *learning rate* dan *batch size* yang konfigurasi optimalnya bervariasi antar rasio, *dropout rate* 0,3 selalu menjadi pilihan terbaik di kelima rasio tanpa terkecuali. Cardoza et al. (2022) menyatakan bahwa konfigurasi *hyperparameter* yang tidak tepat, termasuk regularisasi yang berlebihan, dapat berdampak langsung pada stabilitas dan kualitas hasil pelatihan model CNN. Pada penelitian ini, *dropout* 0,5 yang digunakan pada *baseline* tampaknya terlalu agresif untuk ukuran *dense layer* yang relatif kecil (128 neuron), sehingga terlalu banyak neuron yang dinonaktifkan justru menghambat proses pembelajaran fitur yang relevan dari representasi VGG16.

Terkait pengaruh rasio pembagian data, hasilnya tidak semudah yang dibayangkan. Secara intuitif, lebih banyak data latih semestinya menghasilkan model yang lebih baik, tapi kenyataannya tidak selalu begitu pada skenario *baseline*. *Baseline* di rasio 50:50 justru menghasilkan F1-score tertinggi di antara semua *baseline* (93,61%), mengalahkan rasio 70:30, 80:20, bahkan 90:10 yang masing-masing memiliki porsi data latih lebih besar namun F1-score *baseline*-nya lebih rendah (92,51%, 91,37%, dan 90,73%). Fenomena seperti ini bisa terjadi karena beberapa alasan, yaitu variasi acak dalam inisialisasi bobot, distribusi kelas pada *batch* yang berbeda di tiap sesi pelatihan, hingga ukuran data uji yang semakin kecil pada rasio dengan porsi data latih besar sehingga evaluasi menjadi lebih tidak stabil.

Namun setelah *Grid Search* diterapkan, polanya menjadi lebih jelas. skenario dengan data latih terbanyak (90:10) menghasilkan peningkatan performa paling besar (+2,58 poin persen), sementara skenario dengan data latih tersedikit (50:50) justru menjadi satu-satunya yang tidak terbantu oleh tuning. Ini mengisyaratkan bahwa kualitas hyperparameter punya peran yang tidak kalah penting dari jumlah data latih itu sendiri dengan konfigurasi yang tepat, data latih yang banyak dapat dimaksimalkan, namun tanpa konfigurasi yang baik, data yang banyak pun tidak banyak membantu.

Dari sisi performa per kelas, ada dua temuan yang konsisten di seluruh 45 skenario tanpa terkecuali. Pertama, kelas *Healthy* selalu mendapat *F1-score* tertinggi, berkisar antara 97% hingga 99%. Ini cukup masuk akal karena daun sehat

memiliki karakteristik visual yang paling khas, berupa warna hijau merata dan permukaan halus tanpa bercak, sehingga mudah dikenali meski data latih terbatas.

Kedua yang lebih menarik, kelas *Gray Leaf Spot* konsisten menjadi yang paling sulit diklasifikasikan di seluruh skenario dengan F1-score yang tidak pernah melampaui 89% bahkan pada skenario terbaik sekalipun. Ini bukan soal jumlah data atau konfigurasi model, karena di seluruh rasio dan konfigurasi pun hasilnya tidak bergerak jauh dari kisaran 88-89%. Penyebab paling logis adalah kemiripan visual antara *Gray Leaf Spot* dan *Northern Leaf Blight*. Keduanya menghasilkan bercak memanjang sejajar tulang daun dengan warna abu-abu kecoklatan, dan dalam kondisi tertentu, terutama pada stadium infeksi awal, batas visualnya sangat tipis. *Confusion matrix* pada hampir seluruh skenario menunjukkan bahwa sebagian besar kesalahan prediksi kelas ini memang berpindah ke *Northern Leaf Blight* dan sebaliknya.

Dari sisi kelas *Common Rust*, performa model cukup baik dan menunjukkan peningkatan yang paling signifikan setelah *Grid Search* diterapkan pada beberapa rasio, dengan *precision* yang secara konsisten mencapai 97-100% di seluruh skenario. Ini kemungkinan karena pustula oranye kecoklatan yang menjadi ciri khas penyakit ini memiliki pola warna yang cukup unik dan mudah dibedakan dari ketiga kelas lainnya oleh model CNN yang bekerja pada level fitur warna dan tekstur.

Skenario terbaik secara keseluruhan dari seluruh 45 skenario adalah Grid Search pada rasio 60:40 dengan kombinasi GS-7 (*learning rate* 0,0005, *batch size* 32, *dropout rate* 0,3), yang menghasilkan akurasi 93,67% dan *F1-score* 93,74%

sebagaimana telah dibahas secara rinci pada Gambar 4.6 di subbab 4.2.2. Langkah terakhir dalam pembahasan ini adalah menempatkan hasil penelitian ini dalam konteks yang lebih luas yaitu membandingkannya dengan beberapa penelitian sebelumnya yang juga mengerjakan klasifikasi penyakit daun jagung berbasis citra. Tabel 4.8 merangkum perbandingan tersebut.

Tabel 4.8 Perbandingan hasil penelitian ini dengan penelitian terdahulu

Peneliti (Tahun)	Metode	Dataset	Jumlah Kelas	Akurasi
Rachmad (2023)	CNN scratch	PlantVillage	4	97,20%
Fadhilla et al. (2023)	CNN dasar	Akuisisi lapangan	4	94,50%
Pratama & Pristyanto (2023)	VGG16	PlantVillage + lokal	4	98,10%
Wahyuningrum et al. (2021)	DenseNet-169	PlantVillage	4	99,10%
Penelitian ini (2026)	VGG16 + Grid Search	Maize Diseases Detection	4	93,67%

Dari tabel tersebut, akurasi terbaik yang diperoleh penelitian ini adalah 92,67%, yang memang berada di bawah beberapa penelitian sebelumnya. Rachmad (2023) mencapai 97,20%, Pratama dan Pristyanto (2023) dengan VGG16 berhasil meraih 98,10%, dan Wahyuningrum et al. (2021) dengan DenseNet-169 bahkan mencapai 99,10%. Namun, hasil penelitian ini tetap lebih tinggi dibandingkan Fadhilla et al. (2023) yang menggunakan CNN dasar pada data akuisisi lapangan langsung (94,50%).

Namun ada beberapa hal penting yang perlu dipertimbangkan sebelum langsung menyimpulkan bahwa hasil penelitian ini lebih rendah. Pertama, dataset yang digunakan berbeda. Sebagian besar penelitian terdahulu menggunakan

PlantVillage yang memiliki puluhan ribu citra dengan kondisi yang sangat terkontrol, sementara penelitian ini menggunakan dataset *Maize Diseases Detection* yang hanya berisi 1.500 citra. Perbandingan langsung antar dataset yang berbeda tentu tidak bisa sepenuhnya apel ke apel.

Kedua, kontribusi utama penelitian ini bukan semata-mata pada angka akurasi tertinggi, melainkan pada analisis sistematis pengaruh dua variabel sekaligus yaitu rasio pembagian data dan kombinasi tiga hyperparameter (*learning rate, batch size, dropout rate*) terhadap performa model, melalui total 45 skenario eksperimen yang terstruktur. Penelitian-penelitian sebelumnya umumnya hanya menguji satu atau dua konfigurasi tanpa eksplorasi yang terstruktur pada kombinasi beberapa hyperparameter sekaligus.

Ketiga, arsitektur VGG16 yang digunakan dalam penelitian ini memang bukan yang paling kompleks. Penelitian Wahyuningrum et al. (2021) yang menggunakan DenseNet-169 secara arsitektur jauh lebih dalam dan kompleks. Dengan dataset yang terbatas seperti yang digunakan di sini, memilih VGG16 justru merupakan keputusan yang disengaja untuk menjaga keseimbangan antara kapasitas model dan efisiensi komputasi sesuai dengan salah satu celah penelitian yang diidentifikasi di Bab 2.

Dengan mempertimbangkan semua faktor tersebut, hasil 93,67% pada dataset yang jauh lebih kecil dan dengan pendekatan yang lebih terstruktur bisa dianggap cukup kompetitif. Yang lebih penting, penelitian ini berhasil menunjukkan bahwa optimasi hyperparameter secara sistematis melalui Grid

Search pada tiga parameter sekaligus mampu meningkatkan performa model VGG16 secara konsisten pada empat dari lima rasio yang diuji, dan bahwa pemilihan kombinasi hyperparameter yang tepat bergantung pada kondisi data yang ada, bukan sesuatu yang bisa diasumsikan begitu saja tanpa proses pencarian yang terstruktur.

4.5 Integrasi Sains dan Islam

Sebagai mahasiswa di lingkungan UIN, ada satu hal yang selalu diingatkan sejak awal kuliah bahwa ilmu pengetahuan dan nilai-nilai keislaman bukan dua hal yang berdiri terpisah. Keduanya bisa dan seharusnya saling memperkuat. Penelitian ini memang berbicara tentang kecerdasan buatan dan klasifikasi citra, tapi kalau ditelusuri lebih dalam, ada benang merah yang menghubungkannya dengan ajaran Islam, khususnya soal perintah untuk mengamati, berpikir, dan memanfaatkan akal dalam mengelola apa yang ada di bumi.

Di Bab 1 sudah disebutkan QS. An-Nahl ayat 11 sebagai salah satu landasan filosofis penelitian ini. Ayat tersebut berbunyi:

يُنَبِّتُ لَكُمْ بِهِ الزَّرْعَ وَالزَّيْتُونَ وَالنَّخِيلَ وَالْأَعْنَابَ وَمِنْ كُلِّ الثَّمَرَاتِ إِنَّ فِي ذَلِكَ لَآيَةً لِّقَوْمٍ يَتَفَكَّرُونَ ﴿١١﴾

“Dengan (air hujan) itu Dia menumbuhkan untukmu tanaman-tanaman, zaitun, kurma, anggur, dan segala macam buah-buahan. Sungguh, pada yang demikian itu benar-benar terdapat tanda (kebesaran Allah) bagi orang-orang yang memikirkan.” (Q.S. An-Nahl [16]: 11)

Ayat ini menyebut tanaman secara eksplisit seperti zaitun, kurma, anggur, dan buah-buahan lainnya sebagai tanda kebesaran Allah yang seharusnya mendorong manusia untuk berpikir. Perintah untuk berpikir atau tafakkur di sini

bukan sekadar kontemplasi pasif, melainkan bentuk keterlibatan aktif dengan alam seperti mengamati, mencari pola, dan memahami hukum yang berlaku di dalamnya. Dalam konteks penelitian ini, proses mengamati citra daun jagung, mengidentifikasi pola visual penyakit, dan membangun model yang mampu mengenalinya secara otomatis adalah salah satu bentuk konkret dari *tafakkur* yang dimaksud ayat tersebut.

Jagung sebagai komoditas pertanian yang dibahas dalam penelitian ini memang tidak disebut langsung di Al-Qur'an. Namun perintah untuk menjaga dan mengelola tanaman sebagai sumber pangan yang dalam bahasa agama adalah bagian dari tugas kekhalifahan manusia di bumi bersifat umum dan mencakup seluruh jenis tanaman yang dibutuhkan manusia. Allah berfirman dalam QS. Al-An'am ayat 99:

وَهُوَ الَّذِي أَنْزَلَ مِنَ السَّمَاءِ مَاءً فَأَخْرَجْنَا بِهِ نَبَاتَ كُلِّ شَيْءٍ فَأَخْرَجْنَا مِنْهُ خَضِرًا نُخْرِجُ مِنْهُ حَبًّا مُتَرَاكِبًا ﴿٩٩﴾ ...

“Dan Dialah yang menurunkan air dari langit, lalu Kami tumbuhkan dengan air itu segala macam tumbuh-tumbuhan, maka Kami keluarkan dari tumbuh-tumbuhan itu tanaman yang menghijau, Kami keluarkan dari tanaman yang menghijau itu butir yang banyak...” (Q.S. Al-An'am [6]: 99)

Ayat ini menggambarkan bagaimana Allah menumbuhkan tanaman dari air hujan hingga menghasilkan biji yang bermanfaat. Di sinilah letak relevansinya dengan penelitian ini yang mana jagung adalah tanaman penghasil biji (*habban mutarakiban*) yang menjadi salah satu sumber pangan strategis. Ketika penyakit menyerang daun jagung dan mengganggu proses fotosintesis hingga menurunkan hasil panen, upaya mendeteksinya sedini mungkin adalah bagian dari tanggung jawab manusia untuk menjaga keberlanjutan sumber pangan tersebut.

Dalam perspektif Islam, merawat dan menjaga produktivitas tanaman pangan bukan sekadar urusan ekonomi atau teknologi semata. Ada dimensi ibadah di dalamnya. *Rasulullah Shallallahu 'Alaihi wa Sallam* bersabda:

مَا مِنْ مُسْلِمٍ يَغْرِسُ غَرْسًا أَوْ يَزْرَعُ زَرْعًا فَيَأْكُلُ مِنْهُ طَيْرٌ أَوْ إِنْسَانٌ أَوْ بَيْمَةٌ إِلَّا كَانَ لَهُ بِهِ صَدَقَةٌ

“Tidaklah seorang muslim menanam pohon atau tanaman, lalu dimakan oleh burung, manusia, atau hewan ternak darinya, melainkan hal itu menjadi sedekah baginya.” (H.R. Bukhari No. 2320 dan Muslim No. 1553)

Hadits ini menegaskan bahwa menjaga keberlangsungan tanaman agar tetap sehat dan produktif merupakan bentuk amal yang bernilai sedekah jariyah. Jika menanam saja sudah bernilai ibadah, maka membangun sistem yang membantu mendeteksi penyakit tanaman sejak dini agar hasilnya bisa dinikmati lebih banyak orang tentu tidak kalah nilainya di sisi Allah.

4.5.1 Tujuan dan Manfaat Integrasi

Integrasi antara sains dan Islam dalam penelitian ini bukan sekadar pemenuhan syarat akademis di lingkungan UIN. Lebih dari itu, ada tiga dimensi manfaat yang menjadi tujuan sekaligus pertanggungjawaban dari penelitian ini. Manfaat yang tertuju kepada Allah sebagai pencipta, kepada manusia sebagai pengguna, dan kepada alam sebagai objek yang dijaga.

Hablum minallah Setiap upaya ilmiah yang dilakukan dengan niat ikhlas, metode yang jujur, dan tujuan yang bermanfaat bagi kemaslahatan umat adalah bentuk ibadah dan pengabdian kepada Allah. Penelitian ini dibangun di atas dasar tafakkur yaitu merenungi dan mengkaji ciptaan Allah berupa tanaman jagung

beserta penyakitnya secara ilmiah. Proses mengamati, mengolah data, dan menghasilkan pengetahuan baru merupakan wujud syukur atas akal yang dikaruniakan Allah kepada manusia. QS. Al-Isra' ayat 36 mengingatkan bahwa setiap klaim harus dapat dipertanggungjawabkan. Dalam penelitian ini, pertanggungjawaban itu diwujudkan melalui pengujian sistematis, pelaporan hasil yang jujur, dan pengakuan atas keterbatasan yang ada.

Hablum minannas Secara langsung, penelitian ini ditujukan untuk membantu petani jagung dalam mengidentifikasi penyakit daun secara lebih cepat dan objektif dibandingkan pengamatan visual manual yang selama ini dilakukan. Deteksi dini yang lebih akurat berpotensi mengurangi kerugian akibat gagal panen dan membantu petani mengambil keputusan pengendalian penyakit yang lebih tepat sasaran. Pada skala yang lebih luas, model klasifikasi yang dikembangkan ini dapat menjadi pondasi bagi pengembangan sistem deteksi otomatis yang bisa diakses oleh dinas pertanian maupun masyarakat umum untuk memberikan manfaat yang tidak terbatas pada satu pihak saja.

Hablum minal'alam Deteksi penyakit tanaman yang lebih akurat dan tepat waktu secara tidak langsung berkontribusi pada pelestarian ekosistem pertanian. Ketika penyakit diidentifikasi lebih awal, penggunaan fungisida dan pestisida dapat dilakukan secara lebih terukur hanya pada tanaman yang benar-benar membutuhkan, bukan secara masif dan tidak terkontrol. Pengurangan penggunaan bahan kimia berlebihan berdampak positif pada kesehatan tanah, kualitas air, dan keanekaragaman hayati di lingkungan sekitar lahan pertanian. Ini sejalan dengan prinsip Islam tentang larangan berbuat kerusakan di bumi sebagaimana ditegaskan

dalam QS. Al-Baqarah ayat 11 bahwa manusia tidak boleh membuat kerusakan di muka bumi, termasuk kerusakan ekologis akibat penggunaan bahan kimia yang tidak perlu.

Ketiga dimensi manfaat ini saling berkaitan dan tidak bisa dipisahkan satu sama lain. Manfaat untuk alam mendukung keberlanjutan sumber pangan bagi manusia, manfaat untuk manusia merupakan realisasi dari tugas kekhalifahan yang dibebankan Allah, dan keduanya bersama-sama merupakan bentuk ibadah dan penghambaan kepada Allah yang menciptakan semuanya. Inilah yang dalam pendekatan keilmuan UIN disebut sebagai integrasi sains dan Islam yang utuh bukan sekadar menambahkan ayat di skripsi, tapi menjadikan nilai-nilai Islam sebagai ruh yang menjiwa seluruh proses dan tujuan penelitian.

Yang membuat penelitian ini terasa relevan secara spiritual bukan hanya pada objek tanaman jagung, tapi juga pada pendekatannya. *Convolutional Neural Network* bekerja dengan cara mengekstrak fitur visual dari citra secara bertingkat, dari tepi dan warna di lapisan awal hingga pola yang lebih kompleks di lapisan yang lebih dalam. Proses ini pada dasarnya adalah upaya manusia untuk meniru atau lebih tepatnya memformalisasi cara pandang yang sebenarnya sudah diajarkan Al-Qur'an sejak lama yaitu amati, kenali tanda-tandanya, dan ambil pelajaran darinya.

Dalam seluruh proses penelitian ini, mulai dari memilih arsitektur model, menentukan kandidat *hyperparameter*, hingga menganalisis hasilnya semua keputusan didasarkan pada data dan pengujian yang terstruktur, bukan asumsi atau perkiraan semata. Pendekatan ini sejalan dengan semangat Islam yang sangat

menghargai ilmu yang berbasis bukti dan mendorong umatnya untuk tidak hanya menerima sesuatu begitu saja tanpa pemahaman. QS. Al-Isra' ayat 36 menegaskan hal ini:

وَلَا تَقْفُ مَا لَيْسَ لَكَ بِهِ عِلْمٌ إِنَّ السَّمْعَ وَالْبَصَرَ وَالْفُؤَادَ كُلُّ أُولَٰئِكَ كَانَ عَنْهُ مَسْئُولًا ﴿٣٦﴾

“Dan janganlah kamu mengikuti sesuatu yang tidak kamu ketahui. Sesungguhnya pendengaran, penglihatan, dan hati nurani, semua itu akan diminta pertanggungjawabannya.” (Q.S. Al-Isra' [17]: 36)

Ayat ini secara tidak langsung menuntut adanya pertanggungjawaban atas setiap klaim atau keputusan yang diambil. Dalam konteks penelitian ilmiah, ini terwujud melalui pengujian yang terstruktur, dokumentasi yang jelas, dan evaluasi yang objektif. Penelitian ini mencoba memenuhi tuntutan itu dengan menjalankan eksperimen secara sistematis, melaporkan hasil secara jujur termasuk ketika *Grid Search* tidak selalu lebih baik dari *baseline*, dan mengakui keterbatasan yang ada.

Semangat menuntut ilmu sebagai kewajiban juga ditegaskan oleh *Rasulullah Shalallahu Alaihi Wassalam* dalam hadits shahih yang diriwayatkan oleh Ibnu Majah:

طَلَبُ الْعِلْمِ فَرِيضَةٌ عَلَى كُلِّ مُسْلِمٍ

“Menuntut ilmu adalah kewajiban bagi setiap muslim”
(H.R. Ibnu Majah No. 224, dishahihkan oleh Al-Albani dalam Shahih Ibnu Majah)

Hadits ini diriwayatkan dari Anas bin Malik *radhiyallahu 'anhu* dan termaktub dalam Sunan Ibnu Majah (No. 224), serta telah dishahihkan oleh Syaikh Muhammad Nashiruddin Al-Albani dalam Shahih Ibnu Majah. Kewajiban

menuntut ilmu dalam hadits ini bersifat umum, mencakup ilmu agama maupun ilmu yang bermanfaat bagi kehidupan manusia. Penelitian di bidang kecerdasan buatan untuk pertanian termasuk penelitian ini adalah salah satu wujud nyata dari menuntut ilmu yang bermanfaat bagi kemaslahatan umat.

Pada akhirnya, membangun sistem deteksi penyakit tanaman berbasis kecerdasan buatan adalah salah satu bentuk ikhtiar atau usaha nyata yang dilakukan dengan sungguh-sungguh menggunakan akal dan ilmu yang Allah karuniakan. Hasilnya tidak harus sempurna, karena kesempurnaan hanya milik Allah. Tapi selama prosesnya dilakukan dengan niat yang benar, metode yang bertanggung jawab, dan tujuan yang bermanfaat bagi orang banyak khususnya para petani yang selama ini menghadapi tantangan identifikasi penyakit secara manual di lapangan, maka penelitian ini dalam kapasitasnya yang kecil semoga bisa menjadi bagian dari amal ilmu yang bermanfaat.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini bertujuan untuk menganalisis performa model klasifikasi berbasis arsitektur VGG16 dengan pendekatan *transfer learning* dalam mengklasifikasikan penyakit daun tanaman jagung ke dalam empat kelas, yaitu *Gray Leaf Spot*, *Common Rust*, *Northern Leaf Blight*, dan *Healthy*. Eksperimen dijalankan pada lima variasi rasio pembagian data dengan dua pendekatan pelatihan yaitu *baseline* dan *hyperparameter tuning* menggunakan *Grid Search* yang menguji kombinasi tiga parameter *learning rate* (0,001 dan 0,0005), *batch size* (16 dan 32), serta *dropout rate* (0,3 dan 0,5) menghasilkan delapan kombinasi per rasio, sehingga total terdapat 45 skenario eksperimen. Berdasarkan hasil eksperimen yang telah dilakukan, beberapa kesimpulan dapat ditarik sebagai berikut.

Pertama, model CNN berbasis VGG16 dengan pendekatan *transfer learning* terbukti mampu mengklasifikasikan penyakit daun jagung dengan baik pada dataset yang terbatas. Akurasi terbaik yang diperoleh adalah 93,67% dengan *F1-score* 93,74%, dicapai pada skenario *Grid Search* dengan rasio pembagian data 60:40 menggunakan kombinasi *learning rate* 0,0005, *batch size* 32, dan *dropout rate* 0,3. Hasil ini menunjukkan bahwa pemanfaatan bobot pretrained ImageNet pada VGG16 efektif dalam mengekstrak fitur visual penyakit daun jagung meskipun data yang tersedia hanya berjumlah 1.500 citra.

Kedua, *hyperparameter tuning* metode *Grid Search* terbukti memberikan dampak positif terhadap performa model pada empat dari lima rasio yang diuji (Sholeh et al., 2025). Peningkatan paling signifikan terjadi pada rasio 90:10, di mana *Grid Search* meningkatkan *F1-score* sebesar 2,58 poin persen dari *baseline*. Satu-satunya pengecualian terjadi pada rasio 50:50, di mana *baseline* sedikit lebih unggul dengan selisih yang sangat kecil (0,09 poin persen) selisih yang masih berada dalam batas variasi training yang wajar dan tidak dapat dikatakan signifikan secara praktis.

Ketiga, pemilihan *learning rate* optimal bergantung pada kondisi data yang tersedia dan tidak bisa diasumsikan begitu saja. Morales-Hernández et al. (2023) menjelaskan bahwa tidak ada satu nilai hyperparameter yang berlaku optimal untuk semua kondisi, karena sensitivitasnya bergantung pada karakteristik data dan kompleksitas tugas yang dikerjakan. Rasio dengan data latih yang lebih sedikit, seperti 50:50, 60:40, dan 70:30, cenderung lebih cocok dengan *learning rate* yang lebih kecil yaitu 0,0005, sementara rasio dengan data latih lebih besar, seperti 80:20 dan 90:10, justru lebih optimal dengan *learning rate default* 0,001. Sebaliknya, *dropout rate* 0,3 secara konsisten menjadi pilihan optimal di seluruh lima rasio tanpa terkecuali, menunjukkan bahwa regularisasi yang lebih ringan secara umum lebih sesuai untuk karakteristik dataset penelitian ini dibandingkan *dropout* 0,5 yang digunakan pada *baseline*. Temuan ini mengonfirmasi bahwa proses pencarian hyperparameter secara sistematis seperti *Grid Search* terbukti relevan dan diperlukan, bukan sekadar memilih nilai *default*.

Keempat, dari sisi performa per kelas, kelas *Gray Leaf Spot* konsisten menjadi yang paling sulit diklasifikasikan di seluruh 45 skenario dengan F1-score yang tidak pernah melampaui 0,89. Kemiripan visual antara *Gray Leaf Spot* dan *Northern Leaf Blight* menjadi faktor utama yang menyebabkan model sering ragu di antara keduanya dikarenakan keduanya menghasilkan bercak memanjang sejajar tulang daun dengan warna abu-abu kecoklatan. Sebaliknya, kelas *Healthy* dan *Common Rust* secara konsisten mendapat *F1-score* dan *precision* tertinggi karena karakteristik visualnya yang lebih khas dan mudah dibedakan.

5.2 Saran

Penelitian ini tentu masih memiliki keterbatasan yang bisa menjadi peluang untuk dikembangkan lebih lanjut. Ada beberapa hal yang rasanya sayang kalau tidak dicoba pada penelitian berikutnya. Yang paling mendesak adalah soal dataset. Jumlah 1.500 citra yang digunakan dalam penelitian ini tergolong sangat terbatas untuk tugas klasifikasi citra yang melibatkan empat kelas sekaligus. Penelitian selanjutnya sebaiknya menggunakan dataset yang lebih besar dan beragam, idealnya yang mencakup citra dari kondisi lapangan nyata dengan variasi pencahayaan, sudut pengambilan, dan latar belakang yang lebih heterogen. Ini penting agar model yang dihasilkan lebih *robust* ketika diterapkan di kondisi sebenarnya.

Terkait kelas *Gray Leaf Spot* yang konsisten menjadi tantangan terbesar di seluruh 45 skenario yang telah diuji, ada pendekatan yang masih belum dicoba dan berpotensi membantu, yaitu menerapkan teknik *fine-tuning* pada layer konvolusi

VGG16 yang dalam penelitian ini sepenuhnya dibekukan (*frozen*). Membuka beberapa *layer* terakhir dari *convolutional base* untuk ikut dilatih kemungkinan bisa membantu model menangkap fitur yang lebih spesifik terhadap karakteristik visual penyakit jagung yang ada pada dataset ini, bukan hanya mengandalkan fitur generik dari ImageNet. Eksperimen perbandingan jumlah blok yang dibekukan pada *convolutional base* VGG16 dapat menjadi topik tersendiri yang layak dieksplorasi lebih lanjut, mengingat penelitian ini belum menyentuh aspek tersebut.

Dari sisi *hyperparameter tuning*, penelitian ini telah mengoptimalkan tiga parameter sekaligus, yaitu *learning rate*, *batch size*, dan *dropout rate*, melalui Grid Search dengan delapan kombinasi per rasio. Meski demikian, ruang pencarian ini masih dapat diperluas lebih jauh. Penelitian selanjutnya bisa mempertimbangkan parameter tambahan seperti jumlah neuron pada *dense layer*, jumlah *epoch maksimum* yang lebih besar, atau pemilihan *optimizer* selain Adam, misalnya RMSprop atau SGD dengan *momentum*. Selain itu, mengingat Grid Search menguji seluruh kombinasi secara *exhaustive* sehingga waktu komputasi meningkat signifikan seiring bertambahnya jumlah parameter, metode pencarian yang lebih efisien seperti *Random Search* atau *Bayesian Optimization* juga bisa menjadi alternatif yang lebih hemat waktu ketika ruang parameter yang dieksplorasi semakin besar di penelitian mendatang.

Terakhir, hasil model yang sudah dibangun dalam penelitian ini idealnya tidak berhenti sebagai laporan akademik saja. Ada potensi yang cukup nyata untuk dikembangkan menjadi sistem yang lebih aplikatif, misalnya dalam bentuk aplikasi mobile yang bisa digunakan langsung oleh petani di lapangan untuk

mengidentifikasi penyakit daun jagung hanya dengan memotret daunnya. Dengan akurasi diatas 93%, fondasi modelnya sudah cukup untuk dijadikan titik awal pengembangan ke arah sana. Tentu masih perlu banyak penyempurnaan, tapi setidaknya jarak antara penelitian akademis dan manfaat nyata di lapangan tidak lagi terlalu jauh.

DAFTAR PUSTAKA

- Anand, A. (2025). Leaf Disease Detection System Through Advanced AI and Machine Learning Integration. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.5018301>
- Anikster, Y., Eilam, T., Bushnell, W. R., & Kosman, E. (n.d.). *Spore dimensions of Puccinia species of cereal hosts as determined by image analysis*.
- Azizah, Q. N. (2023). Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network AlexNet. *sudo Jurnal Teknik Informatika*, 2(1), 28–33. <https://doi.org/10.56211/sudo.v2i1.227>
- Baireddy, S., Lee, D.-Y., Gongora-Canul, C., Cruz, C. D., & Delp, E. J. (2022a). *Leaf Tar Spot Detection Using RGB Images* (arXiv:2205.00952). arXiv. <https://doi.org/10.48550/arXiv.2205.00952>
- Baireddy, S., Lee, D.-Y., Gongora-Canul, C., Cruz, C. D., & Delp, E. J. (2022b). *Leaf Tar Spot Detection Using RGB Images* (arXiv:2205.00952). arXiv. <https://doi.org/10.48550/arXiv.2205.00952>
- Baksh, H., Thottempudi, K., Khatal, M. M., G S, S., Das, N., Alam, M. A., Karanwal, R., & P, P. (2024). Advancing Agriculture through Artificial Intelligence, Plant Disease Detection Methods, Applications, and Limitations. *Journal of Advances in Biology & Biotechnology*, 27(8), 730–739. <https://doi.org/10.9734/jabb/2024/v27i81191>
- Bichri, H., Chergui, A., & Hain, M. (2024). Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets. *International Journal of Advanced Computer Science and Applications*, 15(2). <https://doi.org/10.14569/IJACSA.2024.0150235>
- Cardoza, I., García-Vázquez, J. P., Díaz-Ramírez, A., & Quintero-Rosas, V. (2022). Convolutional Neural Networks Hyperparameter Tunning for Classifying Firearms on Images. *Applied Artificial Intelligence*, 36(1), 2058165. <https://doi.org/10.1080/08839514.2022.2058165>
- Duklan, N., Kumar, S., Maheshwari, H., Singh, R., Sharma, S. D., & Swami, S. (2024). CNN Architectures for Image Classification: A Comparative Study Using ResNet50V2, ResNet152V2, InceptionV3, Xception, and MobileNetV2. *International Journal of Electronics and Communication Engineering*, 11(9), 11–21. <https://doi.org/10.14445/23488549/IJECE-V11I9P102>

- Fraiwan, M., Faouri, E., & Khasawneh, N. (2022). Classification of Corn Diseases from Leaf Images Using Deep Transfer Learning. *Plants*, 11(20), 2668. <https://doi.org/10.3390/plants11202668>
- Ihtada, F. K., Alfianita, R., & Aziz, O. Q. (2025). Aspect-based Multilabel Classification of E-commerce Reviews Using Fine-tuned IndoBERT. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*. <https://doi.org/10.22219/kinetik.v10i1.2088>
- Iswantoro, D., & Handayani Un, D. (2022). Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Ilmiah Universitas Batanghari Jambi*, 22(2), 900. <https://doi.org/10.33087/jiubj.v22i2.2065>
- Khoiri, S., Abdiatun, A., Muhlisa, K., Amzeri, A., & Megasari, D. (2021). The Incidence and Severity of Downy Mildew Disease on Local Madurese Maize Crops in Sumenep district, East Java, Indonesia. *Agrologia*, 10(1). <https://doi.org/10.30598/ajibt.v10i1.1295>
- Khoiriyah, S., Rusmana, R., & Laila, A. (2024). Pengaruh Waktu Penyiangan dan Pemangkasan terhadap Pertumbuhan dan Hasil Tanaman Jagung Manis. *Jurnal Ilmiah Membangun Desa dan Pertanian*, 9(4), 359–368. <https://doi.org/10.37149/jimdp.v9i4.1163>
- Luo, N., Meng, Q., Feng, P., Qu, Z., Yu, Y., Liu, D. L., Müller, C., & Wang, P. (2023). China can be self-sufficient in maize production by 2030 with optimal crop management. *Nature Communications*, 14(1), 2637. <https://doi.org/10.1038/s41467-023-38355-2>
- Mawarni, R., Wulaningrum, R., & Helilintar, R. (2023). *1-3 Universitas Nusantara PGRI Kediri*
- Morales-Hernández, A., Van Nieuwenhuyse, I., & Rojas Gonzalez, S. (2023). A survey on multi-objective hyperparameter optimization algorithms for machine learning. *Artificial Intelligence Review*, 56(8), 8043–8093. <https://doi.org/10.1007/s10462-022-10359-2>
- Murphy, B. J., Wilson, T. M., Ziebarth, E. A., Bowerbank, C. R., & Carlson, R. E. (2024). Authentication of Fennel, Star Anise, and Anise Essential Oils by Gas Chromatography (GC/MS) and Stable Isotope Ratio (GC/IRMS) Analyses. *Plants*, 13(2), 214. <https://doi.org/10.3390/plants13020214>
- Mutia Fadhilla, Suryani, D., Labellapansa, A., & Gunawan, H. (2023). Corn Leaf Diseases Recognition Based on Convolutional Neural Network. *IT Journal*

- Peña, D., Martín, C., Fernández-Rodríguez, D., Terrón-Sánchez, J., Vicente, L. A., Albarrán, Á., Rato-Nunes, J. M., & López-Piñeiro, A. (2023). Medium-Term Effects of Sprinkler Irrigation Combined with a Single Compost Application on Water and Rice Productivity and Food Safety. *Plants*, 12(3), 456. <https://doi.org/10.3390/plants12030456>
- Pratama, A. Y., & Pristyanto, Y. (2023). CLASSIFICATION OF CORN PLANT DISEASES USING VARIOUS CONVOLUTIONAL NEURAL NETWORK. *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, 9(1), 49–56. <https://doi.org/10.33480/jitk.v9i1.4258>
- Rachmad, A., Fuad, M., & Rochman, E. M. S. (2023). Convolutional Neural Network-Based Classification Model of Corn Leaf Disease. *Mathematical Modelling of Engineering Problems*, 10(2), 530–536. <https://doi.org/10.18280/mmep.100220>
- Rusnadi Padjung, Muh. Farid, Hari Iswoyo, Muhammad Farid Maricar, Ifayanti Ridwan Saleh, Ahmad Fauzan Adzima, & Amin Nur. (2024). Pertumbuhan dan Produksi Beberapa Varietas Tanaman Jagung (*Zea mays* L.) pada Berbagai Dosis NPK: Growth and Production of Several Varieties of Corn Plants (*Zea mays* L.) at Various Doses of NPK. *Perbal: Jurnal Pertanian Berkelanjutan*, 12(1), 42–54. <https://doi.org/10.30605/perbal.v12i1.3173>
- Septiandi, L. A., Yuniarno, E. M., & Zaini, A. (2021). Deteksi Kedipan dengan Metode CNN dan Percentage of Eyelid Closure (PERCLOS). *Jurnal Teknik ITS*, 10(1), A56–A63. <https://doi.org/10.12962/j23373539.v10i1.61174>
- Setiawan, W., Rochman, E. M. S., Satoto, B. D., & Rachmad, A. (2022). Machine Learning and Deep Learning for Maize Leaf Disease Classification: A Review. *Journal of Physics: Conference Series*, 2406(1), 012019. <https://doi.org/10.1088/1742-6596/2406/1/012019>
- Sholeh, M., Lestari, U., & Andayati, D. (2025). Hyperparameter Optimization Using Grid Search and Random Search to Improve the Performance of Prediction Models with Decision Trees. *Jurnal Riset Multidisiplin Dan Inovasi Teknologi*, 3(03), 453–464. <https://doi.org/10.59653/jimat.v3i03.2025>
- Simran, A., Shijin Kumar, P. S., & Bachu, S. (2021). Content Based Image Retrieval Using Deep Learning Convolutional Neural Network. *IOP Conference Series: Materials Science and Engineering*, 1084(1), 012026. <https://doi.org/10.1088/1757-899X/1084/1/012026>

- Tri Wahyuningrum, R., Kusumaningsih, A., Putra Rajeb, W., & Eddy Purnama, I. K. (2021). Classification of Corn Leaf Disease Using the Optimized DenseNet-169 Model. *2021 The 9th International Conference on Information Technology: IoT and Smart City*, 67–73. <https://doi.org/10.1145/3512576.3512588>
- Tuncel, G., & Gunturk, B. (2024). A Fuzzy Multi-Criteria Decision-Making Approach for Agricultural Land Selection. *Sustainability*, 16(23), 10509. <https://doi.org/10.3390/su162310509>
- Vankudoth, R. (2025). *AI-Powered System for Detecting and Classifying Plant Diseases using Image Processing Techniques*. In Review. <https://doi.org/10.21203/rs.3.rs-7218586/v1>
- Yu, H., Liu, J., Chen, C., Heidari, A. A., Zhang, Q., Chen, H., Mafarja, M., & Turabieh, H. (2021). Corn Leaf Diseases Diagnosis Based on K-Means Clustering and Deep Learning. *IEEE Access*, 9, 143824–143835. <https://doi.org/10.1109/ACCESS.2021.3120379>
- Zhang, K., Mason, A. S., Farooq, M. A., Islam, F., Quezada-Martinez, D., Hu, D., Yang, S., Zou, J., & Zhou, W. (2021). Challenges and prospects for a potential allohexaploid Brassica crop. *Theoretical and Applied Genetics*, 134(9), 2711–2726. <https://doi.org/10.1007/s00122-021-03845-8>