

## Implementasi Convolution Neural Network (CNN) untuk Deteksi Penyakit pada Daun Jagung Berbasis Citra Digital

Imam Wirabowo<sup>1</sup>, Indah Susilawati<sup>2</sup>

<sup>1,2</sup>Informatika, Teknologi Informasi, Universitas Mercu Buana Yogyakarta

<sup>1</sup>imamwirabowo4@gmail.com, <sup>2</sup>indah@mercubuana-yogya.ac.id

### Abstract

*Corn is Indonesia's second major food commodity that frequently faces leaf diseases such as Blight, Common Rust, and Gray Leaf Spot. Manual disease identification relies on subjective observation and is ineffective for large-scale applications. This research develops an efficient corn leaf disease detection system using Convolutional Neural Network (CNN) for low-specification devices. The dataset comprises 2,960 corn leaf images across three disease categories from academic publications, open datasets, and direct collection. Preprocessing includes resizing to 224×224 pixels, normalization, and data augmentation. The CNN model features sequential architecture with three Conv2D-MaxPooling2D blocks, flatten layer, 128-neuron dense layer with ReLU, dropout 0.5, and softmax output layer. Eight parameter combinations were evaluated using validation splits (0.2, 0.3), zoom ranges (0.2, 0.4), and epochs (20, 50). Optimal results achieved validation accuracy of 90.03% and loss of 0.2574 with validation split 0.2, zoom range 0.2, and 20 epochs. Confusion matrix demonstrates balanced performance across disease classes with average precision, recall, and F1-score of 0.90. The model proves efficient, accurate, and suitable for implementation on computationally limited devices, providing practical solutions for farmers in early corn disease detection.*

*Keywords: corn leaf disease, CNN, digital image*

### Abstrak

Jagung merupakan komoditas pangan utama kedua di Indonesia yang sering menghadapi masalah penyakit pada daun seperti *Blight*, *Common Rust*, dan *Gray Leaf Spot*. Identifikasi penyakit secara manual masih mengandalkan pengamatan langsung yang bersifat subjektif dan kurang efektif untuk skala luas. Penelitian ini bertujuan mengembangkan sistem deteksi penyakit daun jagung berbasis *Convolutional Neural Network* (CNN) yang efisien untuk perangkat dengan spesifikasi komputasi rendah. Dataset yang digunakan terdiri dari 2.960 citra daun jagung dengan tiga kategori penyakit yang diperoleh dari publikasi akademik, dataset terbuka, dan pengambilan langsung. *Preprocessing* meliputi *resizing* ke 224×224 piksel, normalisasi, dan augmentasi data sederhana. Model CNN dibangun dengan arsitektur *sequential* berlapis yang terdiri dari tiga blok *Conv2D* dan *MaxPooling2D*, *flatten layer*, *dense layer* 128 neuron dengan *ReLU*, *dropout* 0.5, dan *output layer* dengan aktivasi *softmax*. Evaluasi dilakukan dengan delapan kombinasi parameter *validation split* (0.2 dan 0.3), *zoom range* (0.2 dan 0.4), dan *epoch* (20 dan 50). Hasil terbaik diperoleh pada kombinasi *validation split* 0.2, *zoom range* 0.2, dan *epoch* 20 dengan akurasi validasi 90.03% dan *loss* 0.2574. *Confusion matrix* menunjukkan performa seimbang pada ketiga kelas penyakit dengan *precision*, *recall*, dan *F1-score* rata-rata 0.90. Model ini terbukti efisien, akurat, dan cocok untuk implementasi pada perangkat dengan keterbatasan komputasi, memberikan solusi praktis bagi petani dalam deteksi dini penyakit tanaman jagung.

Kata kunci: penyakit daun jagung, CNN, citra digital



## 1. Pendahuluan

Jagung merupakan salah satu komoditas pangan utama di Indonesia, berfungsi sebagai sumber karbohidrat kedua setelah padi [1]. Peran pentingnya dalam sektor pangan, pakan ternak, dan industri menjadikan jagung sebagai tanaman yang dibudidayakan secara luas dan bernilai ekonomi tinggi [2]. Namun demikian, dalam proses budidayanya, tanaman ini kerap menghadapi tantangan berupa serangan penyakit, terutama pada bagian daunnya. Penyakit seperti *Blight*, *Common Rust*, dan *Gray Leaf Spot* menjadi ancaman serius karena dapat menurunkan produktivitas dan kualitas hasil panen [3][4]. *Blight* disebabkan oleh jamur *exserohilum turcicum*, *common rust* oleh *puccinia sorghi*, dan *gray leaf spot* oleh *cercospora zeaemaydis*, yang semuanya mengganggu proses fotosintesis tanaman [4]. Faktor lingkungan seperti kelembaban tinggi dan suhu ekstrem mempercepat penyebarannya, sehingga upaya deteksi dini sangat penting untuk mengendalikan dampaknya.

Selama ini, identifikasi penyakit pada daun jagung masih banyak mengandalkan pengamatan langsung oleh petani atau ahli pertanian. Meskipun cara ini bersifat praktis, hasil identifikasi seringkali bersifat subjektif, sangat bergantung pada pengalaman individu, dan kurang efektif untuk diterapkan dalam skala lahan yang luas. Keterlambatan dalam proses identifikasi juga membuat tindakan pencegahan dan penanganan menjadi tidak optimal, sehingga berisiko menurunkan hasil produksi secara signifikan. Dalam konteks ini, penggunaan teknologi pengolahan citra digital berbasis kecerdasan buatan, khususnya *Convolutional Neural Network* (CNN), menjadi alternatif solusi yang menjanjikan [5]. Penerapan teknologi berbasis *deep learning* telah menjadi pendekatan yang populer dalam mendeteksi penyakit tanaman, termasuk penyakit pada daun jagung.

CNN merupakan salah satu teknik *deep learning* yang telah terbukti mampu mengekstraksi fitur visual dari citra secara otomatis dan akurat [6]. Teknologi ini telah banyak dimanfaatkan di berbagai bidang, termasuk pertanian, untuk mengidentifikasi penyakit tanaman secara otomatis melalui citra digital. Salah satu keunggulan CNN adalah kemampuannya dalam mengenali pola visual kompleks, bahkan dari gambar yang memiliki gejala serupa antar jenis penyakit. Gambar daun yang diambil melalui kamera atau perangkat sensor optik akan diolah oleh sistem untuk mengenali pola-pola khusus dari penyakit tertentu. Keberhasilan sistem ini sangat bergantung pada kualitas gambar yang digunakan. Unsur teknis seperti

ketajaman, pencahayaan, dan sudut pengambilan gambar bisa mempengaruhi hasil ekstraksi fitur oleh model [7]. Setiap gambar digital tersusun dari piksel-piksel kecil, di mana masing-masing piksel menyimpan informasi mengenai warna dan tingkat kecerahan yang membentuk keseluruhan tampilan gambar [8][9]. Kelebihan lain dari penggunaan citra digital adalah fleksibilitasnya untuk diintegrasikan ke dalam sistem berbasis perangkat keras berdaya rendah [8].

Namun, penerapan CNN dalam deteksi penyakit tanaman juga menghadapi beberapa tantangan, khususnya terkait keterbatasan sumber daya perangkat keras dan sensitivitas terhadap kualitas gambar [10][11]. Untuk mengatasi hal ini, teknik augmentasi data sering digunakan agar model dapat mengenali lebih banyak variasi citra dan menghasilkan performa yang lebih stabil terhadap data baru [12].

Berbagai penelitian telah dilakukan untuk mengembangkan model deteksi penyakit pada daun jagung berbasis CNN. Kusumastuti, Dwi Putra, dan Yudam (2024) memanfaatkan arsitektur *EfficientNet* dan berhasil mencapai akurasi 97,77% dalam mengklasifikasi *Blight*, *Common Rust*, dan *Gray Leaf Spot* [11]. Meskipun sangat akurat, model ini membutuhkan daya komputasi tinggi sehingga kurang cocok untuk diterapkan pada perangkat dengan spesifikasi rendah. Sebaliknya, penelitian oleh Azizah (2023) menggunakan arsitektur *AlexNet* yang lebih ringan dan mampu mencapai akurasi 90%, meskipun performanya menurun pada kondisi pencahayaan yang tidak ideal [1]. Hal ini menunjukkan bahwa efisiensi model dan ketahanannya terhadap variasi citra di lapangan masih menjadi tantangan.

Penelitian lain oleh Alwan, Handayan, dan Faisal (2024) mengombinasikan CNN dengan algoritma *YOLOv8* untuk mendeteksi penyakit pada daun jagung. Hasilnya cukup menjanjikan dengan akurasi mencapai 95%, serta kemampuan deteksi yang cepat dan efisien [13]. Namun, model ini cenderung mengalami *overfitting* jika data latih kurang bervariasi, serta sangat bergantung pada kualitas gambar input. Sementara itu, Widiyanto, Utami, dan Ariatmanto (2023) mengembangkan model CNN standar dengan akurasi 93,85%, namun tidak mengoptimalkan teknik augmentasi data, yang padahal penting untuk meningkatkan ketahanan model terhadap variasi citra [3]. Di sisi lain, Imam Sya'bullah M. (2021) menunjukkan bahwa penggunaan format warna HSV dalam CNN dapat

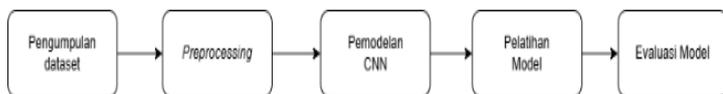
meningkatkan akurasi hingga 95,69%, namun pendekatan ini memerlukan proses pra-pemrosesan tambahan dan tidak dioptimalkan untuk perangkat dengan spesifikasi rendah [5].

Berdasarkan berbagai studi tersebut, terlihat bahwa masih terdapat gap dalam penelitian, yaitu pada kebutuhan model yang tidak hanya akurat tetapi juga ringan, efisien, dan tangguh terhadap variasi kualitas gambar serta kondisi perangkat keras yang terbatas. Beberapa penelitian lebih berfokus pada pencapaian akurasi tinggi tanpa memperhatikan aspek adaptabilitas model di kondisi lapangan nyata. Selain itu, teknik augmentasi data belum dimanfaatkan secara optimal dalam sebagian besar penelitian, padahal teknik ini dapat meningkatkan generalisasi model terhadap variasi pencahayaan, sudut pengambilan gambar, dan latar belakang [14][15].

Oleh karena itu, penelitian ini bertujuan untuk mengembangkan model CNN yang ringan dan efisien untuk klasifikasi penyakit pada daun jagung, dengan memanfaatkan teknik augmentasi data sederhana agar model lebih tangguh terhadap variasi kondisi citra. Sistem ini dirancang agar dapat dijalankan pada perangkat dengan spesifikasi rendah yang umum digunakan oleh petani. Dengan demikian, diharapkan hasil dari penelitian ini dapat memberikan kontribusi nyata bagi sektor pertanian, khususnya dalam mendeteksi penyakit tanaman secara cepat, akurat, dan dapat diakses oleh kalangan petani secara luas.

**2. Metode Penelitian**

Penelitian ini dilakukan dengan pendekatan eksperimen menggunakan model berbasis *Convolutional Neural Network* (CNN) untuk mengidentifikasi penyakit pada daun jagung melalui citra digital. CNN dipilih karena kemampuannya dalam mengenali pola-pola kompleks pada gambar, seperti tekstur, warna, dan bentuk yang menjadi indikator utama dalam klasifikasi gejala penyakit tanaman. Dengan karakteristik tersebut, model CNN diharapkan mampu memberikan hasil klasifikasi yang lebih akurat dan konsisten. Rangkaian tahapan penelitian ini digambarkan secara visual pada Gambar 1.



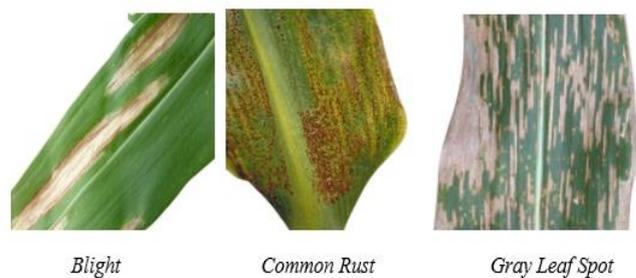
Gambar 1. Blok Diagram Alur Penelitian

Penelitian ini memanfaatkan perangkat keras berupa laptop Acer 4349 dengan RAM 12 GB, SSD 500 GB, kartu grafis Intel HD Graphics, dan prosesor Intel Celeron B800 1.50 GHz dual-core, yang meskipun spesifikasinya sederhana, masih mampu menjalankan pelatihan dan pengujian model skala kecil hingga menengah dengan dukungan komputasi awan. Untuk pengumpulan data secara manual, digunakan *smartphone* Redmi Note 12 Pro dengan RAM 8 GB

dan kamera 108 MP yang menghasilkan gambar berkualitas tinggi. Di sisi perangkat lunak, pengembangan dilakukan pada *platform* Windows 10 64-bit menggunakan *Google Colab* dan *Jupyter Notebook*, yang mendukung pemrosesan data besar dan kompatibel dengan pustaka pembelajaran mesin seperti *TensorFlow* dan *Keras*, sementara *Python* dipilih sebagai bahasa pemrograman utama karena fleksibilitas dan ekosistemnya yang luas.

**2.1. Pengumpulan Data**

Data yang digunakan dalam penelitian ini merupakan citra daun jagung yang dikategorikan berdasarkan tiga jenis penyakit utama, yaitu *Blight*, *Common Rust*, dan *Gray Leaf Spot*. Tahap awal yang dilakukan adalah pengumpulan data sebanyak 2.960 citra, yang diperoleh melalui dua metode, yaitu dari dataset terbuka dan pengambilan gambar secara langsung. Sebagian besar data diperoleh dari dataset terbuka *Kaggle*, sebanyak 2.660 citra, dengan rincian 900 citra untuk kelas *Blight*, 950 citra untuk *Common Rust*, dan 810 citra untuk *Gray Leaf Spot*. Contoh data dari sumber *Kaggle* dapat dilihat pada Gambar 2.



Gambar 2. Dataset Penyakit Daun Jagung dari Kaggle

Sementara itu, 300 citra lainnya diperoleh melalui proses pengambilan gambar secara langsung di lahan pertanian jagung yang berlokasi di Majalengka. Pengambilan gambar ini dilakukan oleh peneliti dengan menggunakan kamera *smartphone* Redmi Note 12 Pro, yang menghasilkan citra beresolusi rata-rata 2160 x 3840 piksel, dengan ukuran file berkisar antara 1,76 MB hingga 2,5 MB per gambar. Dalam kegiatan ini, peneliti didampingi oleh Bapak Eno, seorang petani jagung setempat yang memberikan dukungan teknis dan akses ke lokasi lahan, serta turut membantu dalam mengenali kondisi umum tanaman di lapangan. Selain itu, penelitian ini juga mendapatkan pendampingan akademik dari Dr. Ir. Dian Astriani, S.P., M.P., selaku ahli dalam bidang hama, penyakit, dan proteksi pada tanaman, yang memberikan arahan ilmiah dalam proses identifikasi dan klasifikasi jenis penyakit daun jagung secara tepat. Jumlah data hasil pengambilan langsung adalah 100 citra untuk masing-masing kelas penyakit, yaitu *Blight*, *Common Rust*, dan *Gray Leaf Spot*. Contoh



dibandingkan untuk mengetahui konfigurasi mana yang memberikan akurasi tertinggi dengan *loss* yang paling rendah.

Dalam proses evaluasi performa model, digunakan metode *confusion matrix*, yaitu sebuah tabel yang menggambarkan kemampuan model dalam mengklasifikasikan data secara tepat serta mengidentifikasi berbagai jenis kesalahan prediksi yang terjadi. Melalui *confusion matrix*, dapat diketahui bagaimana distribusi prediksi benar dan salah untuk setiap kelas, sehingga memudahkan analisis lebih mendalam terhadap kinerja model dalam mengidentifikasi masing-masing jenis penyakit pada daun jagung. Untuk melengkapi analisis ini, digunakan pula sejumlah parameter evaluasi, yaitu *accuracy*, *precision*, *recall*, dan *F1-score*. Keempat metrik ini dihitung berdasarkan nilai-nilai yang diperoleh dari *confusion matrix*, dan dirumuskan dalam persamaan (1) hingga (4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (3)$$

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (4)$$

Keterangan:

TP (*True Positive*): Kasus positif yang diprediksi dengan benar.

TN (*True Negative*): Kasus negatif yang diprediksi dengan benar.

FP (*False Positive*): Kasus negatif yang salah diprediksi sebagai positif.

FN (*False Negative*): Kasus positif yang salah diprediksi sebagai negatif.

### 3. Hasil dan Pembahasan

Penelitian ini menghasilkan model klasifikasi penyakit daun jagung menggunakan pendekatan deep learning berbasis arsitektur *Convolutional Neural Network* (CNN). Dataset terdiri dari 2.960 citra yang terbagi ke dalam tiga kelas penyakit utama, yaitu *Blight*, *Common Rust*, dan *Gray Leaf Spot*. Sebanyak 2.660 citra diperoleh dari dataset terbuka *Kaggle*, sementara 300 citra lainnya diambil secara langsung di lahan pertanian jagung di Majalengka dengan bantuan petani setempat menggunakan kamera *smartphone* Redmi Note 12 Pro. Seluruh citra kemudian melalui tahap *preprocessing* seperti konversi warna, *resizing* menjadi 224×224 piksel, normalisasi nilai piksel ke rentang 0–1, serta augmentasi seperti rotasi, *flipping*, dan *zooming*.

Model CNN dibangun dengan tiga blok Conv2D dan MaxPooling2D, dilanjutkan dengan proses *flattening*, satu *dense layer* dengan 128 neuron dan aktivasi ReLU, serta dropout sebesar 0.5 untuk mencegah *overfitting*. *Output layer* menggunakan aktivasi softmax untuk klasifikasi multi-kelas. Proses pelatihan menggunakan batch size 32 dan data dibagi ke dalam data pelatihan dan validasi dengan rasio 80:20 atau 70:30. Delapan kombinasi parameter *preprocessing* (P1–P8) diuji untuk mencari konfigurasi paling optimal, dengan variasi pada *validation split*, *zoom range*, dan jumlah *epoch*, yang dirangkum dalam Tabel 1.

Tabel 1. Tabel Parameter Perbandingan Model

Kode Perbandingan	Validation Split	Zoom Range	Epoch
P1	0.2	0.2	20
P2	0.2	0.4	20
P3	0.3	0.2	20
P4	0.3	0.4	20
P5	0.2	0.2	50
P6	0.2	0.4	50
P7	0.3	0.2	50
P8	0.3	0.4	50

#### 3.1. Hasil Pengujian Berdasarkan Validation Split

Percobaan ini dilakukan untuk mengevaluasi dampak penggunaan dua nilai *validation split*, yaitu 0.2 dan 0.3, terhadap akurasi dan los model. *Validation split* berperan dalam menentukan pembagian data yang digunakan untuk validasi, sehingga mempengaruhi terhadap stabilitas serta kemampuan generalisasi model. Bagian ini menjadi sangat penting, terutama ketika model diterapkan pada perangkat dengan komputasi rendah ataupun terbatas. Pada percobaan ini, digunakan total data sebanyak 2.960. Jika *validation split* yang digunakan dengan nilai 0.2, data pelatihan berjumlah 2.368 data atau 80% dari total data, sementara 592 data atau 20% sisanya digunakan untuk validasi. Jika nilai *validation split*nya menjadi 0.3, data pelatihan akan berjumlah 2.072 data atau 70% , dan data validasi sebanyak 888 data atau 30% dari total data. Perbedaan pembagian ini bertujuan untuk menguji performa model dalam berbagai skenario dan mengamati pengaruhnya terhadap pengaruhnya terhadap akurasi serta loss. Hasil pengujian yang telah dilakukan menunjukkan detail performa model terdapat pada Tabel 2.

Tabel 2. Tabel Hasil Pengujian Berdasarkan Validation Split

Kode Perbandingan	Validation Split	Akurasi Training	Akurasi Validasi	Loss Training	Loss Validasi
P1	0.2	91.60%	90.03%	0.2121	0.2547
P2	0.2	89.99%	85.47%	0.2582	0.3589
P5	0.2	96.28%	92.40%	0.1098	0.2815

P6	0.2	94.26%	89.36%	0.1494	0.3693	P4	20	91.65%	89.53%	0.2314	0.2513
P3	0.3	90.83%	86.94%	0.2398	0.3014	P5	50	96.28%	92.40%	0.1098	0.2815
P4	0.3	91.65%	89.53%	0.2314	0.2513	P6	50	94.26%	89.36%	0.1494	0.3693
P7	0.3	91.00%	90.83%	0.2358	0.2534	P7	50	91.00%	90.83%	0.2358	0.2534
P8	0.3	94.16%	90.99%	0.1605	0.2237	P8	50	94.16%	90.99%	0.1605	0.2237

Pada kombinasi perbandingan dengan *validation split* 0.2, model secara umum menunjukkan akurasi validasi yang cukup tinggi. Seperti pada kombinasi P1 yang mencapai akurasi validasi sebesar 90.03% dengan *loss* 0.2547, hanya dengan menggunakan 20 *epoch*, menunjukkan hasil yang efisien untuk komputasi rendah. Sementara itu, kombinasi P5 menunjukkan hasil yang lebih tinggi dengan nilai sebesar 92.40% namun memerlukan 50 *epoch*, yang berarti memerlukan komputasi yang lebih besar dan beresiko *overfitting* lebih tinggi, seperti yang terlihat pada selisih antara akurasi pelatihan dan validasi.

Pada kombinasi perbandingan yang menggunakan *validation split* 0.3, model terlihat mempunyai stabilitas validasi yang lebih baik pada kombinasi tertentu, seperti pada P8 yang mencapai akurasi 90.99% dan *loss* validasi terendah 0.2237. Namun, seluruh kombinasi yang dengan *validation split* 0.3 cukup baik hanya dalam menggunakan 50 *epoch*, yang memerlukan komputasi yang lebih besar.

Secara keseluruhan, *validation split* 0.2 lebih sesuai untuk kondisi dengan komputasi rendah, karena mampu memberikan hasil akurasi yang kompetitif dalam waktu pelatihan lebih singkat dan beban komputasi yang lebih ringan.

### 3.2. Hasil Pengujian Berdasarkan Epoch

Jumlah *epoch* adalah parameter yang menunjukkan berapa kali seluruh data pelatihan diproses oleh model selama *training*. Parameter ini berperan dalam menentukan berapa lama durasi pelatihan serta sejauh mana model memahami pola pada data dalam mempelajarinya. Pada pengujian ini menggunakan dua nilai *epoch*, yaitu 20 dan 50, dengan tujuan diterapkan dua nilai ini adalah untuk menilai pengaruhnya terhadap akurasi, *loss*, serta efisiensi proses komputasi. Hasil pengujian yang telah dilakukan dirangkum secara rinci pada Tabel 3.

Tabel 3. Tabel Hasil Pengujian Berdasarkan Epoch

Kode Perbandingan	Epoch	Akurasi Training	Akurasi Validasi	Loss Training	Loss Validasi
P1	20	91.60%	90.03%	0.2121	0.2547
P2	20	89.99%	85.47%	0.2582	0.3589
P3	20	90.83%	86.94%	0.2398	0.3014

Pada kombinasi perbandingan dengan 20 *epoch*, model seperti P1 berhasil mencapai akurasi validasi 90.03% dengan *loss* validasi 0.2547, yang tergolong baik dalam pelatihan dengan durasi yang pendek. Kombinasi P4 juga menjadi salah satu kombinasi dengan *loss* validasi yang tergolong rendah yaitu 0.2513, mendekati performa dengan *epoch* yang lebih tinggi, namun dalam waktu yang lebih singkat.

Pada kombinasi dengan 50 *epoch* secara umum menghasilkan akurasi yang lebih tinggi, seperti kombinasi P5 yang mencapai 92.40% dan 90.99% yang dicapai oleh kombinasi P8, serta memiliki *loss* pada pelatihan yang lebih kecil. Namun, peningkatan ini dengan resiko pada penggunaan waktu dan komputasi yang lebih besar. Selain itu kombinasi seperti P6 menunjukkan adanya potensi *overfitting*, dengan ditandai oleh akurasi pelatihan sebesar 94.26% namun validasi lebih rendah sebesar 89.36% serta *loss* validasi tinggi yaitu 0.3693.

Secara keseluruhan, penggunaan 20 *epoch* lebih baik untuk penerapan pada perangkat rendah dengan komputasi terbatas, karena mampu menghasilkan performa yang baik dengan waktu pelatihan yang lebih efisien.

### 3.3 Hasil Pengujian Berdasarkan Zoom Range

*Zoom range* adalah salah satu teknik augmentasi data yang digunakan untuk meningkatkan generalisasi model dengan memperbesar atau memperkecil citra input secara acak. Dalam pengujian ini akan dilakukan dengan menggunakan dua variasi *zoom range*, yaitu 0.2 dan 0.4, untuk mengevaluasi dampaknya terhadap akurasi *loss*. Hasil pengujian yang telah dilakukan dicantumkan secara rinci pada Tabel 4.

Tabel 4. Tabel Hasil Pengujian Berdasarkan Zoom Range

Kode Perbandingan	Zoom Range	Akurasi Training	Akurasi Validasi	Loss Training	Loss Validasi
P1	0.2	91.60%	90.03%	0.2121	0.2547
P3	0.2	90.83%	86.94%	0.2398	0.3014
P5	0.2	96.28%	92.40%	0.1098	0.2815
P7	0.2	91.00%	90.83%	0.2358	0.2534
P2	0.4	89.99%	85.47%	0.2582	0.3589

P4	0.4	91.65%	89.53%	0.2314	0.2513
P6	0.4	94.26%	89.36%	0.1494	0.3693
P8	0.4	94.16%	90.99%	0.1605	0.2237

Pada kombinasi perbandingan dengan *zoom range* 0.2, sebagian model menunjukkan performatas validasi yang lebih stabil dan efisien. Model dengan kombinasi P1 dan P7 mevapai akurasi validasi diatas 90%, dengan *loss* validasi yang rendah seperti P7 dengan akurasi 90.83% dan *loss*-nya 0.2534. Kombinasi P5 bahkan memiliki akurasi validasi tertinggi yaitu 92.40% meskipun dengan *loss* validasi yang lebih tinggi dari kombinasi P7.

Pada kombinasi yang menggunakan *zoom range* 0.4, seperti kombinasi P2 dan P6 cenderung memiliki akurasi lebih rendah seperti P2 dengan akurasi yang hanya 85.47% dan P6 yang *loss* validasinya tinggi sebesar 0.3693. Meskipun kombinasi P8 menunjukkan hasil yang baik dengan nilai akurasi 90.99% dan nilai *loss* validasi mencapai 0.2237, peningkatan pada performa ini tampak tidak konsisten dan muncul dengan kompleksitas augmentasi yang lebih besar.

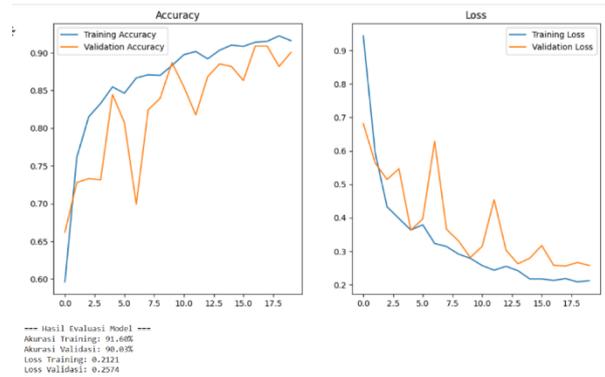
Dari segi efisiensi terhadap komputasi, augmentasi dengan *zoom range* 0.2 lebih ringan serta memberikan hasil validasi yang stabil dan baik. Oleh karena itu, kombinasi dengan *zoom range* 0.2 lebih baik diterapkan pada sistem dengan komputasi terbatas tanpa mengorbankan performa secara signifikan.

### 3.4 Analisis dan Pembahasan

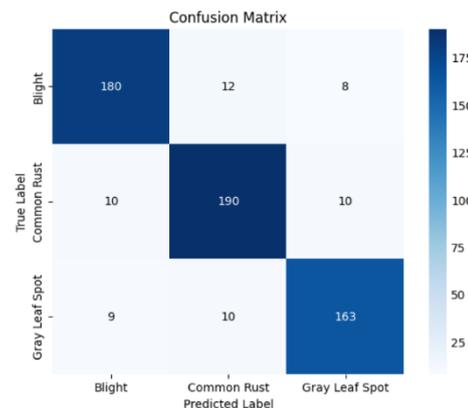
Berdasarkan hasil pengujian terhadap berbagai variasi kombinasi parameter data augmentasi dan jumlah *epoch*, kombinasi P1 menunjukkan hasil dengan performa yang sangat baik dan paling seimbang jika diterapkan untuk *device* dengan komputasi yang rendah dengan kombinasi *zoom range* 0.2, *validation split* 0.2, dan jumlah *epoch* 20 ini menghasilkan akurasi pelatihan sebesar 91.60% dan akurasi validasi sebesar 90.03%, dengan nilai *loss* pelatihan 0.2121 dan *loss* validasi 0.2574. Selisih yang kecil antara akurasi pelatihan dan validasi menunjukkan bahwa model mampu melakukan generalisasi dengan baik terhadap data yang belum pernah dilihat sebelumnya, serta tidak mengalami *overfitting* yang berlebihan. Hal ini menjadi salah satu indikator penting dalam memilih model yang optimal namun efisien, terutama untuk diterapkan pada perangkat dengan komputasi rendah.

Gambar 5 adalah grafik akurasi dan *loss* selama proses pelatihan. Grafik akurasi menunjukkan bahwa nilai akurasi dan validasi meningkat seiring bertambahnya *epoch*, dan bergerak mendekati titik konvergen pada akhir pelatihan. Hal ini mengidentifikasi bahwa model mempelajari secara stabil. Sementara itu, grafik *loss* menunjukkan penurunan yang konsisten, baik pada data pelatihan

ataupun validasi, meskipun terdapat fluktuasi kecil pada *loss* validasi. Namun, nilai akhir dari *loss* validasi tetap berada pada titik rendah, menandakan bahwa model mampu mempertahankan performa yang baik. Hal ini dapat menguntungkan bagi *deployment* model pada perangkat *edge* seperti *smartphone*, *Raspberry Pi*, atau *embedded system* lainnya yang memiliki keterbatasan pada komputasi.



Gambar 5. Grafik Akurasi Kombinasi P1



Gambar 6. Hasil Confusion Matrix Kombinasi P1

Evaluasi lebih mendetail terhadap performa model ditunjukkan melalui Gambar 6, yang menyajikan *confusion matrix* untuk kombinasi P1. Gambar ini menggambarkan jumlah prediksi benar dan salah untuk masing-masing kelas penyakit daun jagung, yaitu *Blight*, *Common Rust*, dan *Gray Leaf Spot*. Model berhasil mengklasifikasikan 180 dari 200 data *Blight* dengan benar, sementara 12 data salah diklasifikasikan sebagai *Common Rust* dan 8 data sebagai *Gray Leaf Spot*. Untuk kelas *Common Rust*, model mengklasifikasikan 190 dari 210 data dengan tepat, sedangkan sisanya terdiri dari 10 data yang salah diklasifikasikan sebagai *Blight* dan 10 data lainnya sebagai *Gray Leaf Spot*. Pada kelas *Gray Leaf Spot*, sebanyak 163 dari 182 data berhasil diklasifikasikan dengan benar, sementara terdapat 9 data yang diklasifikasikan sebagai *Blight* dan 10 data sebagai *Common Rust*.

Kesalahan klasifikasi yang terjadi tergolong kecil dan tersebar cukup merata di antara ketiga kelas, yang mengindikasikan bahwa model tidak menunjukkan kecenderungan bias terhadap satu kelas tertentu. Konsistensi performa ini juga tercermin dalam hasil *classification report* pada Tabel 5, di mana nilai *precision*, *recall*, dan *F1-score* untuk seluruh kelas berada di kisaran 0,89 hingga 0,90. Model menunjukkan akurasi keseluruhan sebesar 90,03 persen, dengan nilai rata-rata *macro* dan rata-rata *weighted* masing-masing sebesar 0,90. Hasil ini menegaskan bahwa model memiliki kinerja yang seimbang dan andal dalam mengklasifikasikan ketiga jenis penyakit daun jagung.

Tabel 5. Classification Report Kombinasi P1

Class	Precision	Recall	F1-Score	Support
Blight	0.90	0.90	0.90	200
Common Rust	0.89	0.90	0.90	210
Gray Leaf Spot	0.90	0.90	0.90	182
Accuracy			0.90	592
Macro avg	0.90	0.90	0.90	592
Weighted avg	0.90	0.90	0.90	592

Secara keseluruhan, kombinasi P1 terbukti menjadi pilihan paling tepat untuk implementasi pada perangkat dengan komputasi terbatas. Kombinasi ini menunjukkan kinerja yang ringan, akurat, dan stabil, sehingga ideal untuk digunakan dalam lingkungan yang memiliki keterbatasan sumber daya komputasi. Dengan jumlah *epoch* yang rendah, konfigurasi augmentasi data yang sederhana, serta performa yang baik, model ini tidak hanya efisien dalam penggunaan sumber daya, tetapi juga mampu memberikan prediksi yang akurat dan stabil. Hal ini menjadi nilai tambah yang penting bagi penerapan nyata di sektor pertanian, khususnya dalam mendukung proses deteksi dini penyakit daun jagung yang cepat dan tepat.

#### 4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem deteksi penyakit pada daun jagung berbasis *Convolutional Neural Network* (CNN) yang efisien untuk perangkat dengan spesifikasi komputasi rendah. Dari delapan kombinasi pengujian parameter, hasil terbaik diperoleh pada kombinasi P1 dengan *validation split* 0.2, *zoom range* 0.2, dan *epoch* 20 menghasilkan akurasi validasi 90.03% dan *loss* 0.2574 tanpa indikasi *overfitting*. Evaluasi lanjutan melalui *confusion matrix* dan *classification report* menunjukkan performa yang seimbang pada tiga kelas penyakit (*Blight*, *Common Rust*, dan *Gray Leaf Spot*) dengan rata-rata *precision*, *recall*, dan *F1-score* sebesar 0.90. Dengan arsitektur sederhana, preprocessing ringan, serta penggunaan augmentasi dasar, sistem ini terbukti akurat dan efisien, serta

potensial untuk diterapkan secara langsung dalam sektor pertanian.

#### Daftar Rujukan

- [1] Q. N. Azizah, "Klasifikasi Penyakit Daun Jagung Menggunakan Metode Convolutional Neural Network AlexNet," *sudo J. Tek. Inform.*, vol. 2, no. 1, pp. 28–33, 2023, doi: 10.56211/sudo.v2i1.227.
- [2] Kementerian Pertanian, *Direktorat pakan direktorat jenderal peternakan dan kesehatan hewan kementerian pertanian 2022*. 2022.
- [3] B. Widiyanto, E. Utami, and D. Ariatanto, "Identifikasi Penyakit Tanaman Jagung Berdasarkan Citra Daun Menggunakan Convolutional Neural Network," *Techno.Com*, vol. 22, no. 3, pp. 599–608, 2023, doi: 10.33633/tc.v22i3.8425.
- [4] D. Ahumada, "Gray Leaf Spot in Corn," *NC State Ext. Publ.*, pp. 4–7, 2020, [Online]. Available: <https://content.ces.ncsu.edu/gray-leaf-spot-in-corn>
- [5] U. G. Mada, "INTISARI Klasifikasi Penyakit Daun Jagung menggunakan," pp. 0–1, 2021.
- [6] U. Kulsum and A. Cherid, "Penerapan Convolutional Neural Network Pada Klasifikasi Tanaman Menggunakan ResNet50," *Simkom*, vol. 8, no. 2, pp. 221–228, 2023, doi: 10.51717/simkom.v8i2.191.
- [7] S. Ahmed, M. B. Hasan, T. Ahmed, M. R. K. Sony, and M. H. Kabir, "Less is More: Lighter and Faster Deep Neural Architecture for Tomato Leaf Disease Classification," *IEEE Access*, vol. 10, pp. 68868–68884, 2022, doi: 10.1109/ACCESS.2022.3187203.
- [8] M. Khoiruddin, A. Junaidi, and W. A. Saputra, "Klasifikasi Penyakit Daun Padi Menggunakan Convolutional Neural Network," *J. Dinda Data Sci. Inf. Technol. Data Anal.*, vol. 2, no. 1, pp. 37–45, 2022, doi: 10.20895/dinda.v2i1.341.
- [9] M. Bagga and S. Goyal, "Image-based detection and classification of plant diseases using deep learning: State-of-the-art review," *Urban Agric. Reg. Food Syst.*, vol. 9, no. 1, 2024, doi: 10.1002/uar2.20053.
- [10] T. B. Sasongko, H. Haryoko, and A. Amrullah, "Analisis Efek Augmentasi Dataset dan Fine Tune pada Algoritma Pre-Trained Convolutional Neural Network (CNN)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 4, pp. 763–768, 2023, doi: 10.25126/jtiik.20241046583.
- [11] R. Kusumastuti, T. Dwi Putra, and Z. Zulfahmi Yudam, "Klasifikasi Citra Penyakit Daun Jagung Menggunakan Algoritma Cnn Efficientnet," *Multitek Indones.*, vol. 17, no. 2, pp. 143–153, 2024, doi: 10.24269/mtkind.v17i2.10085.
- [12] M. H. Zayd, M. W. Oktavian, D. G. T. Meranggi, J. A. Figo, and N. Yudistira, "Improvement of garbage classification using pretrained Convolutional Neural Network," *Teknologi*, vol. 12, no. 1, pp. 1–8, 2022, doi: 10.26594/teknologi.v0i0.2403.
- [13] D. N. Alwan, H. H. Handayan, and S. Faisal, "Implementasi YOLOv8 Untuk Deteksi Penyakit Daun Jagung Menggunakan Algoritma Convolutional Neural Network (CNN)," *Sci. Student J. Information, Technol. Sci.*, vol. 5, no. 2, 2024, [Online]. Available: <https://journal.ubpkarawang.ac.id/mahasiswa/index.php/ssj/article/view/1009>
- [14] M. Setiono and Supatman, "Klasifikasi Penyakit Antraknosa Citra Cabai Rawit Dengan Metode Convolutional Neural Network (CNN)," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 11, no. 2, pp. 308–320, 2024.
- [15] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," *Array*, vol.

- 16, no. November, p. 100258, 2022, doi: 10.1016/j.array.2022.100258.
- [16] C. R. Kotta, D. Paseru, and M. Sumampouw, "Implementasi Metode Convolutional Neural Network untuk Mendeteksi Penyakit Pada Citra Daun Tomat," *J. Pekommas*, vol. 7, no. 2, pp. 123–132, 2022, doi: 10.56873/jpkm.v7i2.4961.
- [17] A. T. R. Dzaky, "Deteksi Penyakit Tanaman Cabai Menggunakan Metode Convolutional Neural Network," *e-Proceeding Eng.*, vol. 8, no. 2, pp. 3039–3055, 2021, [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/14701/14478>
- [18] S. Sheila, I. Permata Sari, A. Bagas Saputra, M. Kharil Anwar, and F. Restu Pujianto, "Deteksi Penyakit Pada Daun Padi Berbasis Pengolahan Citra Menggunakan Metode Convolutional Neural Network (CNN)," *Multinetics*, vol. 9, no. 1, pp. 27–34, 2023, doi: 10.32722/multinetics.v9i1.5255.