

Klasifikasi Penyakit Daun Pada Tanaman Kopi Menggunakan *Convolutional Neural Network* Dengan Arsitektur *MobileNet*

1st Alfian Ikhwanda
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
alfanikhwanda@student.telkomuniversity.ac.id

2nd Rita Magdalena
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
ritamagdalenat@telkomuniversity.ac.id

3rd Sofia Saidah
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
sofiasaidahsfi@telkomuniversity.ac.id

Abstrak — Tanaman kopi merupakan salah satu komoditas unggulan di Indonesia yang berkontribusi besar terhadap perekonomian nasional, baik melalui ekspor maupun konsumsi domestik. Namun, produktivitas kopi sering terancam oleh penyakit daun, seperti karat daun (*Hemileia vastatrix*), bercak daun (*Phoma Costaricensis*), dan penggerek daun (*Leucoptera Coffeella*), yang dapat menyebabkan penurunan kualitas dan hasil panen secara signifikan. Identifikasi penyakit daun secara manual oleh petani atau ahli pertanian seringkali memerlukan waktu dan keahlian yang khusus, sehingga rentan terhadap kesalahan. Penelitian ini bertujuan untuk mengembangkan sistem klasifikasi otomatis penyakit daun pada tanaman kopi menggunakan *Convolutional Neural Network* (CNN) dengan arsitektur *MobileNet*. Arsitektur *MobileNet* dipilih karena efisiensinya dalam menangani data dengan sumber daya komputasi terbatas, seperti perangkat seluler. Proses klasifikasi dimulai dengan pengumpulan data dari dataset Kaggle, yang mencakup gambar daun sehat dan yang terinfeksi penyakit. Data ini melalui tahap *preprocessing*, meliputi penyesuaian ukuran citra menjadi 224x224 piksel. Model dilatih menggunakan parameter utama seperti *optimizer*, *learning rate*, *batch size*, dan *epoch*, untuk menghasilkan klasifikasi yang optimal. Penelitian dilakukan dengan menggunakan 1664 dataset, dengan pembagian 400 data daun sehat, 460 data karat daun, 484 data bercak daun dan 320 data penggerek daun. Dataset yang digunakan untuk pengujian yaitu sebesar 333 data. Melalui serangkaian pengujian dengan empat skenario yang berbeda, diperoleh konfigurasi optimal model dengan hasil terbaik yaitu Akurasi 99.549%, Loss 0.058, Presisi 95%, Recall 95%, F1-score 95%. Dengan parameter terbaik adalah

optimizer RMSprop, *learning rate 0.0001*, *epoch 125* dan *batch size 16*.

Kata kunci: Daun Kopi, *Convolutional Neural Network* (CNN), *MobileNet*, Penyakit Daun, Citra Daun Tanaman Kopi.

I. PENDAHULUAN

Tanaman kopi adalah salah satu komoditas unggulan di Indonesia dengan produksi tinggi dan berkontribusi penting bagi perekonomian nasional. Selain minyak dan gas, kopi adalah komoditas ekspor Indonesia yang signifikan dan berkontribusi pada pendapatan devisa negara. Pasar kopi di dalam negeri juga masih cukup besar [1]. Namun, tanaman kopi sering mengalami serangan penyakit yang dapat menyebabkan penurunan produksi dan kualitas biji kopi, seperti serangan penyakit daun. Beberapa penyakit utama yang sering ditemui pada daun tanaman kopi antara lain *Leaf Rust* (*Hemileia vastatrix*), *Leaf Blight* (*Phoma Costaricensis*), dan *Leaf Miner* (*Leucoptera Coffeella*) [2].

Jenis-jenis penyakit pada daun tanaman kopi ditentukan secara manual oleh ahli pertanian dengan mengamati gejala pada daun. Metode tradisional ini membutuhkan waktu dan keahlian khusus, sehingga pengembangan sistem klasifikasi otomatis penyakit daun kopi dengan teknologi *computer vision* adalah solusi yang menjanjikan. Identifikasi visual juga dapat menimbulkan kesalahan, terutama bagi petani yang kurang berpengalaman [3].

Penelitian sebelumnya menunjukkan hasil yang baik dalam identifikasi penyakit daun tanaman kopi menggunakan CNN dengan arsitektur *EfficientNet-B0* (91% akurasi) [4], *ResNet50* (99,89%) dan *MobileNet* (97,01%) [5]. Metode *transfer learning* dan model *deep learning* dengan jaringan *MobileNetV2* mencapai akurasi 99,93% dalam mendeteksi penyakit daun kopi robusta [6]. Klasifikasi hama pada citra daun kopi menggunakan CNN arsitektur *Alexnet* menghasilkan akurasi 81,6% [7]. Klasifikasi penyakit daun

kopi menggunakan CNN dengan model EfficientNet memberikan akurasi 97% dan loss 0,1328 pada data pelatihan, dan akurasi 0,97% dan loss 0,1328 pada data pengujian [2]. Penelitian ini bertujuan mengembangkan sistem klasifikasi penyakit daun kopi menggunakan CNN arsitektur MobileNet yang efisien di perangkat seluler dan memberikan akurasi yang baik dalam mendeteksi penyakit daun kopi.

Teknologi computer vision dan machine learning memungkinkan pengembangan sistem klasifikasi penyakit daun yang lebih akurat dan efisien. CNN efektif mengekstrak fitur visual [8], namun membutuhkan sumber daya komputasi besar. MobileNet, arsitektur CNN untuk perangkat seluler, dapat menjadi solusi. MobileNet dirancang untuk perangkat seluler agar efisien dalam ukuran dan kecepatan. MobileNet menggunakan lapisan konvolusi depthwise dan pointwise [9]. Selain itu, MobileNet juga menawarkan beberapa varian arsitektur dengan skala yang berbeda, sehingga memungkinkan untuk menyesuaikan model dengan sumber daya komputasi yang tersedia.

II. KAJIAN TEORI

1. Kopi

Tanaman kopi merupakan komoditas perkebunan bernilai ekonomi tinggi. Dua spesies utama yang dibudidayakan adalah kopi arabika dan kopi robusta [10]. Kopi arabika lebih diminati karena cita rasanya, namun lebih rentan terhadap penyakit. Kopi robusta lebih tahan dan memiliki kandungan kafein yang lebih tinggi [10]. Kopi tumbuh optimal di daerah tropis dengan suhu dan curah hujan tertentu, ketinggian 800-2000 mdpl, serta tanah subur dengan pH 6-7 [11]. Pengendalian penyakit daun kopi dapat dilakukan dengan pemilihan varietas tahan penyakit, praktik budidaya yang tepat, serta penggunaan fungisida yang bijaksana [11].

a. Leaf Rust

Penyakit *Leaf Rust* disebabkan oleh jamur *Hemileia vastatrix* dan merupakan masalah serius bagi tanaman kopi, terutama arabika [12]. Gejalanya berupa bercak kuning pada daun yang kemudian berubah menjadi coklat dan menyebabkan daun rontok [12]. Perkembangan penyakit dipengaruhi oleh suhu, kelembaban, curah hujan, dan sinar matahari [13]. Siklus serangan berulang setiap tahun jika tidak ada pengendalian [13].



GAMBAR 1
Penyakit daun kopi *Leaf Rust*

b. Leaf Blight

Leaf Blight, atau hawar daun, adalah penyakit yang menyerang tanaman kopi dan disebabkan oleh jamur bernama *Phoma costaricensis*. Faktor-faktor seperti kelembaban tinggi, suhu dingin, dan lingkungan yang memiliki tingkat cuaca angin yang kencang dapat meningkatkan risiko tanaman terdampak penyakit ini [15]. Gambar dibawah ini merupakan daun yang terkena penyakit hawar daun [14].



GAMBAR 2
Penyakit daun kopi *Leaf Blight*

c. Leaf Miner

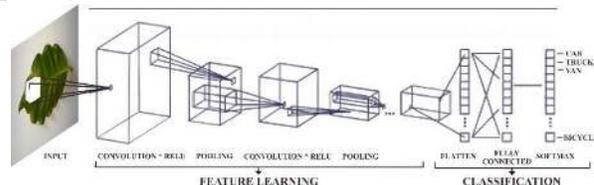
Leaf miner adalah larva dari serangga yang menggerogoti jaringan daun pada tanaman kopi, menyebabkan kerusakan yang dapat terlihat dalam bentuk terowongan atau bercak-bercak pada permukaan daun. Serangan dari leaf miner ini sangat mengganggu proses fotosintesis, yang merupakan aktivitas penting bagi pertumbuhan dan perkembangan tanaman [16]. Gambar berikut ini merupakan daun yang terkena penyakit *leaf miner* [14].



GAMBAR 3
Penyakit daun kopi *Leaf Miner*

2. Convolutional Neural Network

Convolutional Neural Network (CNN) adalah arsitektur jaringan saraf tiruan yang dirancang untuk tugas-tugas terkait pemrosesan gambar dan visual. CNN efektif dalam mengenali pola visual dan banyak digunakan dalam aplikasi seperti pengenalan wajah, klasifikasi objek, dan segmentasi gambar. Arsitektur CNN terdiri dari beberapa jenis lapisan, termasuk lapisan konvolusi, lapisan aktivasi, dan lapisan pooling [17].

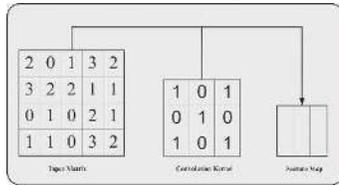


GAMBAR 4
Arsitektur CNN

a. Convolution layer

Convolution layer berperan penting dalam Convolutional Neural Networks (CNN) untuk pengolahan

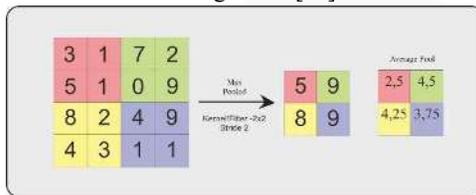
citra digital. Lapisan ini menerapkan filter untuk menangkap fitur penting dari data input seperti tepi, warna, dan tekstur gambar. Filter ini bergerak melewati gambar dan melakukan operasi konvolusi, menghasilkan feature map yang mengidentifikasi pola penting [18]. Proses konvolusi juga mengurangi dimensi data, membuat proses lebih efisien [19].



GAMBAR 5
Komponen Convolution Layer

b. Pooling Layer

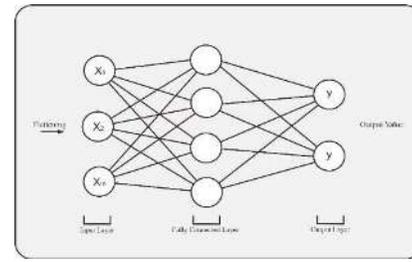
Lapisan pooling, seperti max pooling dan average pooling, mengurangi dimensi spasial dari feature map dengan mengambil beberapa bagian tertentu menjadi satu nilai representatif. Hal ini mengurangi beban komputasi dan meningkatkan efisiensi sistem. Penerapan pooling layer juga dapat meningkatkan performa sistem dalam mengenali objek [20]. Lapisan pooling juga berfungsi memberikan ketahanan terhadap noise atau gangguan pada gambar input. Lapisan ini efektif dalam mempertahankan informasi penting meskipun ada perubahan kecil dalam gambar [21].



GAMBAR 6
Maxpooling dan Averagepooling pada input 4 × 4

c. Fully Connected (FC) Layers

Fully connected layer atau lapisan dense berperan penting dalam jaringan saraf untuk pengolahan data dan pengenalan pola. Fungsinya adalah menghubungkan dan menginterpretasikan fitur dari lapisan sebelumnya. Setiap neuron terhubung dengan neuron di lapisan berikutnya, menciptakan jaringan yang kompleks dan kuat dalam menghasilkan prediksi akhir. Lapisan ini penting untuk klasifikasi pengenalan gambar. Namun, fully connected layer rentan terhadap overfitting. Penggunaan teknik optimasi seperti dropout dapat membantu mengurangi risiko overfitting dan meningkatkan kinerja pada data baru [22].



GAMBAR 7
Komponen Fully connected layer

d. Rectified Linear Unit (RELU)

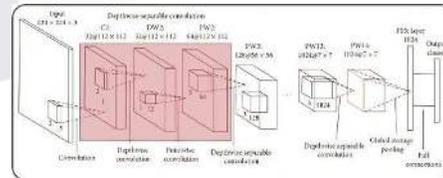
ReLU adalah fungsi aktivasi penting dalam CNN yang memperkenalkan non-linearitas. ReLU memungkinkan jaringan mempelajari pola kompleks [23]. ReLU dan variannya meningkatkan konvergensi dan kinerja model, mempercepat pelatihan dan meningkatkan akurasi model [24].

e. Softmax

Fungsi softmax mengubah output neuron menjadi distribusi probabilitas yang dapat diinterpretasikan di antara berbagai kelas. Softmax memungkinkan model memberikan estimasi yang lebih jelas tentang seberapa besar kemungkinan suatu input termasuk dalam masing-masing kelas. Hal ini sangat berguna dalam pengambilan keputusan untuk tugas klasifikasi [19]. Penggunaan softmax dalam perhitungan acak membantu meningkatkan efisiensi pemrosesan, terutama pada aplikasi perangkat keras [25].

3. MobileNet

MobileNet adalah arsitektur CNN yang dirancang untuk pengenalan gambar dan video pada perangkat dengan sumber daya terbatas. MobileNet fokus untuk menciptakan model yang ringkas dan efisien agar dapat dijalankan pada perangkat dengan kemampuan komputasi terbatas. Arsitektur MobileNet memberikan kinerja tinggi dengan hyperparameter yang membuat model lebih kecil dan proses perhitungan cepat. Untuk mencapai efisiensi tinggi, MobileNet menggunakan teknik "depthwise separable convolutions" yang memecah proses konvolusi menjadi Depthwise convolution dan Pointwise convolution [9].



GAMBAR 8
Proses Arsitektur MobileNet

4. Citra Digital

Citra digital adalah gambar yang dihasilkan atau direkam oleh perangkat digital dan disimpan dalam bentuk piksel. Setiap piksel mewakili warna atau intensitas cahaya pada titik tertentu dan disimpan dalam bentuk kode digital atau bit. Model warna yang paling umum adalah model RGB.

Standar warna, seperti sRGB atau Adobe RGB, dirancang agar tampilan warna di layar digital konsisten dan mendekati warna asli [26].

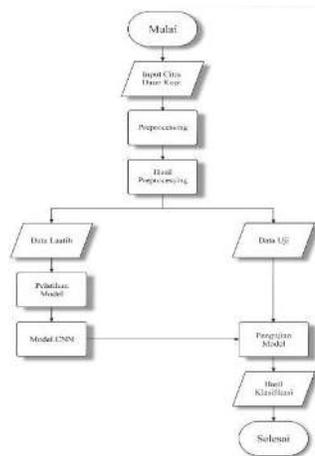
5. Citra RGB

Citra RGB adalah salah satu bentuk representasi gambar digital yang sangat umum digunakan. Model ini didasarkan pada kombinasi tiga warna dasar, yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*), yang dikenal dengan sistem warna aditif. Setiap piksel pada citra RGB memiliki nilai intensitas untuk masing-masing warna tersebut, yang jika digabungkan dapat menghasilkan berbagai warna lain. Misalnya, intensitas setiap kanal warna biasanya berkisar antara 0 hingga 255 dalam format 8-bit, sehingga memungkinkan terciptanya lebih dari 16 juta warna yang berbeda [19].

III. PERENCANAAN SISTEM

1. Alur Sistem

Dalam bab ini, penulis akan menjelaskan mengenai perancangan sistem yang digunakan penulis untuk klasifikasi penyakit daun pada tanaman kopi menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur MobileNet. Proses ini dimulai dari pengumpulan data hingga menghasilkan klasifikasi yang akurat. Gambar dibawah ini merupakan *flowchart* yang digunakan penulis dalam perancangan sistem.



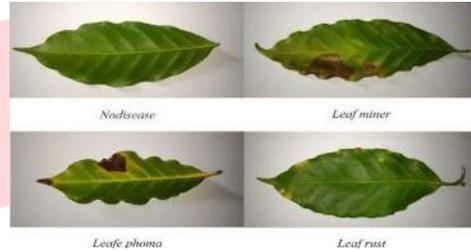
GAMBAR 9
Flowchart sistem

Berdasarkan gambar *flowchart* diatas, tahapan pengembangan sistem klasifikasi penyakit daun pada tanaman kopi yang dimulai dari input citra daun yang diperoleh dari web www.kaggle.com sebagai data awal. Kemudian citra tersebut melalui tahap *pre-processing*, yang meliputi proses *resize* untuk menyesuaikan ukuran citra agar seragam. Hasil *pre-processing* kemudian dibagi menjadi dua bagian, yaitu data latih dan data uji. Model CNN dilatih menggunakan data untuk dihasilkan parameter yang paling optimal. Sementara itu, data uji digunakan untuk menguji model yang telah dilatih guna mengevaluasi performanya. Pada tahap akhir, hasil pengujian memberikan output berupa klasifikasi kondisi daun kopi, seperti kondisi daun sehat atau

terkena penyakit daun, yang diantaranya *leaf miner*, *leaf phoma* dan *leaf rust*.

2. Dataset

Pada penelitian ini, dataset yang digunakan oleh penulis diperoleh melalui platform Kaggle dengan judul *Coffee Leaf Diseases* oleh pengguna Badasstechie yang dapat diakses secara daring. Dataset ini memiliki total berjumlah 1664 gambar yang terdiri dari empat kategori, yaitu daun yang sehat (*nodisease*) 400 gambar, daun yang terinfeksi oleh (*leaf miner*) 460 gambar, daun yang menunjukkan gejala phoma (*leaf phoma*) 484 gambar, dan daun dengan penyakit (*leaf rust*) 320 gambar.

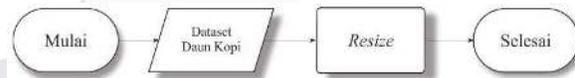


GAMBAR 10

Citra daun kopi *Nodisease*, *leaf miner*, *leaf phoma* dan *leaf rust*

3. Pre-Processing

Tahap preprocessing adalah langkah awal yang sangat penting untuk mempersiapkan citra daun kopi sebelum masuk ke tahapan selanjutnya. Pada tahap ini, citra daun kopi yang akan digunakan dimasukkan sebagai data awal. Langkah pertama adalah mengubah ukuran citra (*resize*) 224 x 224 agar memiliki dimensi yang seragam. Proses ini dilakukan untuk memastikan semua gambar memiliki ukuran yang sama sehingga lebih mudah diproses oleh sistem, serta mengurangi risiko kesalahan akibat perbedaan ukuran citra.



GAMBAR 11
Diagram blok tahap *Pre-processing*

4. Pelatihan Dan Pengujian Model

Setelah melakukan pre-processing, tahap selanjutnya adalah pelatihan model menggunakan CNN dengan arsitektur MobileNet. Tahapan pelatihan model mencakup dua proses utama, yaitu pelatihan dan pengujian model. Proses ini bertujuan untuk membangun model klasifikasi penyakit daun pada tanaman kopi dengan performa yang optimal. Pada proses pelatihan, data latih digunakan untuk melatih model CNN agar mampu mengenali pola dan fitur penting pada citra daun kopi. Model dilatih menggunakan parameter-parameter utama seperti optimizer, learning rate, batch size, dan epochs. Tujuan utama proses ini adalah mengajarkan model untuk menghasilkan klasifikasi yang akurat berdasarkan data citra yang diberikan.

Setelah pelatihan selesai, model yang telah dilatih diuji menggunakan data uji yang telah dipisahkan sebelumnya. Proses pengujian ini bertujuan untuk mengevaluasi performa model dalam mengklasifikasikan citra baru secara akurat. Hasil pengujian meliputi evaluasi metrik seperti akurasi, precision, recall, F1 score, dan loss untuk menilai kemampuan model dalam mengenali kategori daun kopi, termasuk kondisi sehat (nodisease), leaf miner, leaf phoma, dan leaf rust.

5. Parameter Performasi

Untuk mengevaluasi kinerja model yang telah dilatih, dilakukan pengukuran dengan menggunakan beberapa parameter performa. Parameter ini memberikan gambaran tentang kemampuan model dalam melakukan klasifikasi secara akurat dan efektif. Lima parameter utama yang digunakan adalah sebagai berikut:

a. Akurasi

Akurasi digunakan untuk mengukur sejauh mana model dapat melakukan prediksi yang benar terhadap seluruh data uji. Akurasi memberikan gambaran umum kinerja model secara keseluruhan.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Dimana TP (*True Positive*) merupakan jumlah prediksi benar untuk kelas positif. TN (*True Negative*) merupakan jumlah prediksi benar untuk kelas negatif. FP (*False Positive*) merupakan jumlah prediksi salah untuk kelas negatif (diprediksi sebagai positif). FN (*False Negative*) merupakan jumlah prediksi salah untuk kelas positif (diprediksi sebagai negatif).

b. Loss

Loss mengukur rata-rata kesalahan model selama pelatihan atau pengujian. Loss digunakan untuk mengetahui seberapa jauh prediksi model dari nilai aktual. Nilai loss yang lebih rendah menunjukkan performa model yang lebih baik.

Rumus loss tergantung pada fungsi loss yang digunakan, seperti *categorical cross-entropy* untuk klasifikasi multi-kelas:

$$H(p, q) = -\frac{1}{N} \sum_{i=1}^N \sum_k p_{i,k} \log q_{i,k} \quad (2)$$

c. Precision

Precision mengukur akurasi prediksi untuk kelas positif, yaitu seberapa banyak dari prediksi positif yang benar-benar positif.

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

Dimana TP (*True Positive*) merupakan jumlah prediksi benar untuk kelas positif. FP (*False Positive*) merupakan

jumlah prediksi salah untuk kelas negatif (diprediksi sebagai positif).

d. F1 Score

F1 score adalah rata-rata harmonik antara *precision* dan *recall*. Parameter ini digunakan untuk menyeimbangkan *trade-off* antara *precision* dan *recall*, terutama jika distribusi kelas tidak seimbang.

$$F1\ Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (4)$$

e. Recall

Recall (atau sensitivitas) mengukur seberapa banyak dari seluruh kelas positif yang berhasil dikenali oleh model.

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

Dimana TP (*True Positive*) merupakan jumlah prediksi benar untuk kelas positif. FN (*False Negative*) merupakan jumlah prediksi salah untuk kelas positif (diprediksi sebagai negatif).

IV. HASIL DAN ANALISIS

Bab ini berisi hasil dan analisis dari pengujian sistem pada klasifikasi citra daun menggunakan arsitektur CNN MobileNet. Pengujian sistem ini bertujuan untuk mendapatkan hasil performansi terbaik pada klasifikasi citra daun kopi dengan mengevaluasi parameter akurasi, *loss*, *recall*, *presisi*, dan *F1-score*.

1. Skenario Pengujian Sistem

Dalam tugas akhir ini, akan dilakukan 4 skenario pengujian. Setiap pengujian memiliki tujuan untuk mendapatkan hasil kinerja sistem terbaik. Hasil dari setiap pengujian akan digunakan kembali pada pengujian selanjutnya. Tujuannya adalah untuk mendapatkan hasil yang paling optimal. Berikut ini rencana pengujian yang akan dilakukan:

a. Skenario pertama

Skenario pengujian pertama melibatkan proses evaluasi terhadap berbagai *optimizer*. Tujuannya adalah mengidentifikasi *optimizer* yang menghasilkan akurasi tertinggi, yang kemudian akan digunakan pada tahapan pengujian selanjutnya.

b. Skenario kedua

Skenario kedua pengujian ini difokuskan pada *learning rate*. Beberapa nilai *learning rate* akan diuji menggunakan *optimizer* terbaik pada skenario pertama untuk menentukan nilai yang *learning rate* terbaik.

c. Skenario ketiga

Skenario pengujian ketiga yaitu proses evaluasi terhadap *epoch* menggunakan *optimizer* dan *learning rate* terbaik yang diperoleh dari skenario sebelumnya. Tujuannya

adalah mengidentifikasi *epoch* yang mendapatkan nilai dengan akurasi tertinggi.

d. Skenario keempat

Skenario pengujian keempat yaitu proses evaluasi terhadap *batch size* untuk mendapatkan nilai yang terbaik menggunakan *optimizer*, *learning rate* dan *epoch* terbaik yang diperoleh dari skenario sebelumnya.

2. Hasil Pengujian

Data yang akan diuji adalah citra tanaman kopi yang berjumlah 1.664 citra dan telah dilakukan *resize* dengan ukuran 224 x 224 piksel. Terdapat beberapa parameter awal yang akan diukur selama pengujian meliputi *optimizer*, *learning rate*, *epoch*, dan *batch size*. Untuk melihat parameter awal pada pengujian pertama, dapat dilihat pada Tabel 1.

TABEL 1
Parameter Awal

Ukuran Citra	224 × 224 pixel
<i>Optimizer</i>	Adam
<i>Learning rate</i>	0,0001
<i>Epoch</i>	100
<i>Batch size</i>	32

Pada parameter *optimizer*, lima *optimizer* yang akan diterapkan adalah Adam, Nadam, Adamax, RMSprop dan Stochastic Gradient Descent (SGD). Untuk parameter *learning rate*, terdapat tiga variasi nilai yang akan diuji, yaitu 0.001, 0.0001 dan 0,00001. Pada parameter *epoch*, akan digunakan variasi nilai sebanyak empat, yaitu 25, 50, 75 dan 100. Sedangkan pada parameter *batch size*, akan digunakan lima variasi, yakni 4, 8, 16, 32, dan 64. Proses pelatihan model akan memanfaatkan 80% dari total 1664 data, sementara proses pengujian akan menggunakan sisa dari proses pelatihan yaitu 20%. Pengujian ini bertujuan untuk mendapatkan nilai akurasi, *loss*, *precision*, *recall*, dan *F1-score*.

a. Skenario 1: Pengujian Terhadap *Optimizer*

Pada skenario pertama ini akan dilakukan pengujian terhadap *optimizer* dengan parameter awal seperti pada Table 1 Pengujian ini dilakukan dengan tujuan untuk mendapatkan *optimizer* yang paling efisien untuk mengklasifikasi citra daun kopi. *Optimizer* yang akan diuji yaitu Adam, Nadam, Adamax, RMSprop dan SGD. Hasil dari pengujian *optimizer* tersebut dapat dilihat pada Tabel 2.

TABEL 2
Hasil Pengujian Terhadap *optimizer*

<i>Optimizer</i>	Akurasi	<i>Loss</i>	<i>Pres</i>	<i>Recall</i>	<i>FI Score</i>
Adam	97.444%	0.156	94%	94%	94%
Nadam	96.992%	0.153	94%	94%	94%
Adamax	91.729%	0.304	91%	91%	91%
RMSprop	97.895%	0.133	94%	94%	94%
SGD	71.729%	0.787	78%	78%	77%

Tabel 2 Menunjukkan *optimizer* RMSprop merupakan *optimizer* dengan nilai keseluruhan terbaik yaitu akurasi 97.895%, *loss* 0.133, *precision* 94%, *recall* 94%, dan *F1-score* 94%. *Optimizer* RMSprop (*Root Mean Square Propagation*) bekerja dengan menyesuaikan *learning rate* secara adaptif berdasarkan rata-rata eksponensial dari kuadrat gradien sebelumnya. RMSprop lebih unggul dari yang lain, menjadikan *optimizer* ini menjadi pilihan yang sangat baik untuk digunakan dalam pengujian selanjutnya.

b. Skenario 2: Pengujian Terhadap *Learning Rate*

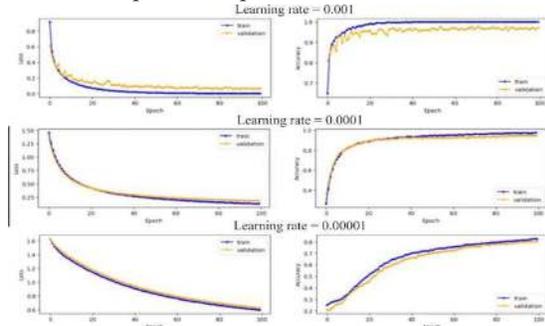
Pada skenario kedua ini akan dilakukan pengujian terhadap *learning rate* dengan *epoch* 100, *batch size* 32 dan menggunakan *optimizer* terbaik yang sudah diperoleh dari skenario satu yaitu RMSprop. Pengujian ini dilakukan dengan tujuan untuk mendapatkan *learning rate* yang paling efisien. Nilai yang akan diuji adalah 0.001, 0.0001, dan 0.00001. Hasil dari pengujian *learning rate* tersebut dapat dilihat pada Tabel 3 berikut ini.

TABEL 3
Hasil dari pengujian *learning rate*

<i>Learning Rate</i>	Akurasi	<i>Loss</i>	<i>Pres</i>	<i>Recall</i>	<i>FI Score</i>
0,001	100.000	0.001	97%	97%	97%
0,0001	97.895	0.133	94%	94%	94%
0,00001	83.158	0.594	82%	82%	82%

Hasil dari pengujian diatas, *learning rate* 0.001 memperoleh nilai keseluruhan yang sangat bagus. Namun berdasarkan analisis perbandingan grafik evaluasi, model dengan *learning rate* 0.0001 menunjukkan performa yang lebih baik dibandingkan dengan *learning rate* 0.001. Performa tersebut terlihat dari akurasi validasi yang lebih tinggi, *loss* validasi yang lebih rendah, serta kurva *loss* dan akurasi pada grafik yang lebih stabil dan berdekatan antara data training dan validasi. Dengan demikian, *learning rate*

0.0001 lebih optimal untuk dijadikan sebagai *learning rate* untuk skenario pengujian berikutnya. Grafik perbandingan untuk *learning rate* 0.001, *learning rate* 0.0001 dan *learning rate* 0.00001 dapat dilihat pada Gambar 12 berikut.



GAMBAR 12

Grafik perbandingan untuk *learning rate* 0.001, *learning rate* 0.0001 dan *learning rate* 0.00001

c. Skenario 3: Pengujian Terhadap Epoch

Pada skenario ketiga ini akan dilakukan pengujian terhadap *epoch* dengan menggunakan *optimizer* RMSprop, *learning rate* 0.0001, dan *batch size* awal yaitu 32. Pengujian ini dilakukan dengan tujuan untuk mendapatkan *epoch* yang paling efisien. Nilai *epoch* yang akan diuji adalah 75, 100, 125 dan 150. Hasil dari pengujian *epoch* dapat dilihat pada Tabel 4.4 berikut ini.

TABEL 4
Hasil dari pengujian epoch

Epoch	Akurasi	Loss	Precision	Recall	FI Score
75	95.789	0.171	94%	94%	94%
100	97.594	0.130	94%	94%	94%
125	98.346	0.102	96%	95%	95%
150	99.398	0.071	94%	93%	93%

Tabel 4 menunjukan bahwa *epoch* 125 memberikan hasil terbaik secara signifikan, ditandai dengan akurasi (98.346) dan *loss* (0.102). Meskipun memiliki nilai akurasi yang rendah dan *loss* yang lebih tinggi dibandingkan dengan *epoch* 150, namun dari nilai metrik presisi, *recall*, dan *fi-Score*, *epoch* 125 mendapatkan nilai yang relatif lebih tinggi. Maka dari itu, *epoch* 125 akan digunakan untuk pengujian pada skenario selanjutnya.

d. Skenario 4: Pengujian Terhadap Batch Size

Pada skenario keempat ini akan dilakukan pengujian terhadap *batch size* dengan menggunakan *optimizer* RMSprop, *learning rate* 0.0001, dan *epoch* 125. Pengujian ini dilakukan dengan tujuan untuk mendapatkan *epoch* yang paling efisien. Nilai *batch size* yang akan diuji adalah 8, 16, 32 dan 64. Hasil dari pengujian *batch size* dapat dilihat pada Tabel 5 berikut ini.

TABEL 5
Hasil dari pengujian batch size

Batch Size	Akurasi	Loss	Precision	Recall	FI score
8	99.398	0.042	94%	94%	94%
16	99.549	0.058	95%	95%	95%
32	99.248	0.090	94%	93%	94%
64	98.195	0.132	92%	92%	92%

Tabel 5 menunjukkan bahwa kinerja model menurun seiring peningkatan *batch size*. Akurasi dan *loss* terbaik dicapai pada *batch size* 8, sementara metrik lainnya yaitu *precision*, *recall* dan *fi-score* dicapai pada *batch size* 16. Penurunan kinerja pada *batch size* besar mengindikasikan kesulitan menemukan minimum lokal optimal dan kurangnya generalisasi. *Batch size* 16 menunjukkan kinerja terbaik dengan penggunaan memori yang efisien. Oleh karena itu, *batch size* 16 dipilih sebagai konfigurasi optimal untuk model.

3. Analisis Hasil

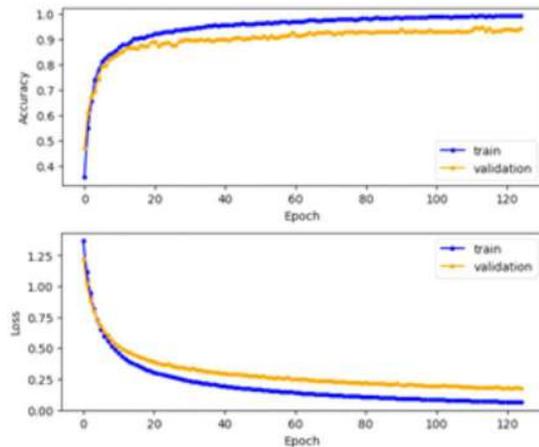
Dari pengujian terhadap keempat skenario tersebut, dapat disimpulkan bahwa konfigurasi optimal untuk model klasifikasi citra daun kopi ini adalah menggunakan *optimizer* RMSprop, *learning rate* 0.0001, *epoch* 125, dan *batch size* 16. RMSprop adalah sebagai *optimizer* yang paling optimal, sementara *learning rate* 0.0001 memberikan keseimbangan terbaik antara kecepatan konvergensi dan stabilitas model. *Epoch* 125 memungkinkan model untuk belajar secara optimal tanpa mengalami *overfitting* dan *batch size* 16 menawarkan keseimbangan terbaik antara kinerja dan efisiensi penggunaan memori. Tabel 6 merupakan skenario terbaik yang diperoleh dari pengujian ini.

TABEL 6
Skenario Terbaik

Ukuran Citra	224 × 224
Optimizer	RMSprop
Learning Rate	0.0001
Epoch	125
Batch Size	16

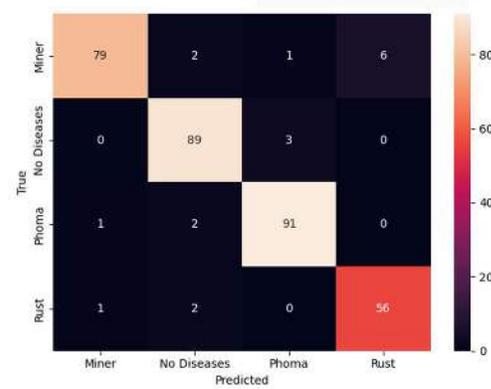
Konfigurasi ini menghasilkan kinerja terbaik dengan akurasi mencapai 99.549% dan *loss* 0.058. Pada grafik

akurasi tidak mengalami *overfitting*, menunjukkan peningkatan yang stabil dan konsisten dikarenakan penggunaan kombinasi parameter yang optimal, yang dapat disimpulkan bahwa pemilihan parameter yang telah di uji efektif dalam mengoptimalkan kinerja model dan mencapai akurasi yang tinggi. Hasil grafik akurasi dan loss ditunjukkan dari Gambar 13 berikut.



GAMBAR 13
Grafik Akurasi dan Loss

Pada Gambar 14 dibawah terlihat bahwa *confusion matrix* menunjukkan kinerja model yang sangat baik dalam mengklasifikasikan citra daun kopi, di mana sebagian besar citra diklasifikasikan dengan benar, ditunjukkan oleh nilai tinggi pada diagonal utama yang mewakili *true positive* setiap kelas. Total untuk citra yang diklasifikasi adalah 333 citra yang terbagi menjadi 4 kelas yaitu *Nodiseases* 91, *Rust* 56, *Phoma* 94, *Miner* 92.



GAMBAR 14
Confusion Matrix

Meskipun terdapat beberapa kesalahan klasifikasi *minor* (*false positive* dan *false negative*) di luar diagonal, jumlahnya relatif kecil, menunjukkan kemampuan model yang baik dalam membedakan kelas-kelas citra daun kopi. Model mungkin sedikit lebih sulit membedakan antara kelas "*Miner*" dan "*No Diseases*", tetapi secara keseluruhan, model

menunjukkan performa yang sangat baik dengan akurasi tinggi dalam klasifikasi citra daun kopi.

V. KESIMPULAN

Penelitian ini berhasil membangun model klasifikasi citra daun kopi menggunakan arsitektur CNN *MobileNet* dengan dataset yang dikumpulkan. Model ini mampu mengklasifikasikan empat kelas penyakit daun kopi, yaitu "*Miner*", "*No Diseases*", "*Phoma*", dan "*Rust*". Melalui serangkaian pengujian dengan empat skenario yang berbeda, diperoleh konfigurasi optimal model dengan menggunakan *optimizer* RMSprop, *learning rate* 0.0001, *epoch* 125, dan *batch size* 16. Konfigurasi ini menghasilkan kinerja terbaik dengan akurasi mencapai 99.549% dan loss 0.058, menunjukkan bahwa model telah mempelajari data dengan baik. Hasil ini menunjukkan bahwa arsitektur *MobileNet* memiliki performa yang sangat baik dalam mengklasifikasikan penyakit daun pada tanaman kopi.

Analisis grafik akurasi menunjukkan bahwa model tidak mengalami *overfitting* dan mampu melakukan generalisasi dengan baik pada data baru. Hal ini terlihat dari peningkatan akurasi yang stabil dan konsisten selama proses pelatihan, serta tidak adanya kesenjangan yang besar antara akurasi data *training* dan validasi. *Confusion matrix* juga memperkuat analisis ini, menunjukkan bahwa model memiliki performa yang sangat baik dalam mengklasifikasikan citra daun kopi, terutama untuk kelas "*Phoma*" dan "*Rust*" yang memiliki kesalahan klasifikasi sangat minim.

Meskipun model sedikit lebih sulit membedakan antara kelas "*Miner*" dan "*No Diseases*", namun secara keseluruhan tetap menunjukkan akurasi yang tinggi. Jumlah data yang berbeda-beda pada setiap kelas tidak menghalangi model untuk mempelajari dan mengklasifikasikan citra dengan baik. Hasil penelitian ini menunjukkan potensi besar penggunaan arsitektur CNN *MobileNet* dalam mendiagnosis penyakit daun kopi secara otomatis dan akurat.

REFERENSI

1. Badan Pusat Statistik. (2023). *Statistik kopi Indonesia 2022* (Vol. 7). Jakarta: Badan Pusat Statistik. ISSN 2714-8505.
2. Rosadi, M. I., Hakim, L., & A., M. F. (2023). Classification of coffee leaf diseases using the convolutional neural network (CNN) EfficientNet model. *IAIC International Conference Series*, 4(1), 58–69. <https://doi.org/10.34306/conferenceseries.v4i1.627>
3. Aufar, Y., & Kaloka, T. P. (2022). Robusta coffee leaf diseases detection based on MobileNetV2 model. *International Journal of Electrical and Computer Engineering (IJECE)*, 12(6), 6675–6683. <https://doi.org/10.11591/ijece.v12i6.pp6675-6683>
4. Sabrina, S. A., & Maki, W. F. A. (2022). Klasifikasi penyakit pada tanaman kopi robusta berdasarkan citra

- daun menggunakan Convolutional Neural Network. *e-Proceeding of Engineering*, 9(3), 1919–1927. <https://doi.org/10.11591/epe.v9i3.12345>
5. Paulos, E. B., & Woldeyohannis, M. M. (2022). Detection and classification of coffee leaf disease using deep learning. In *Proceedings of the ICT4DA Conference* (pp. 1-7). IEEE. <https://doi.org/10.1109/ICT4DA56482.2022.9971300>
 6. AUFAR, Y., & Kaloka, T. P. (2022). Robusta Coffee Leaf Detection based on MobileNetV2. *Jurnal Internasional Teknik Elektro dan Komputer (IJECE)*, 12(6), 6675-6683. <https://doi.org/10.11591/ijece.v12i6.pp6675-6683>
 7. Irfansyah, D., Mustikasari, M., & Suroso, A. (2021). Arsitektur Convolutional Neural Network (CNN) Alexnet untuk klasifikasi hama pada citra daun tanaman kopi. *Jurnal Informatika: Jurnal Pengembangan IT (JPIT)*, 6(2). ISSN: 2477-5126, e-ISSN: 2548-9356.
 8. Esgario, J. G. M., Krohling, R. A., & Ventura, J. A. (2023). *Deep learning for classification and severity estimation of coffee leaf biotic stress*. Federal University of Espirito Santo.
 9. Howard, A. G., Wang, W., Zhu, M., Weyand, T., Chen, B., Andreetto, M., & Kalenichenko, D. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
 10. Rahardjo, Pudji. *Berkebun Kopi*. Penebar Swadaya, https://books.google.co.id/books?id=Qy0-DwAAQBAJ&pg=PA1&hl=id&source=gbs_toc_r&cad=2#v=onepage&q&f=false. Diakses pada [accessed Okt. 17, 2024].
 11. Nazir, Nani. "Mengenal Tanaman Kopi". BBPP Lembang, 28 Desember 2016. <https://bbpplembang.bppsdp.pertanian.go.id/publikasi-detail/1385>. Diakses pada [accessed Okt. 17, 2024].
 12. Sugiarti, L. (2019). Identifikasi hama dan penyakit pada tanaman kopi di kebun percobaan Fakultas Pertanian Universitas Winaya Mukti. *Jurnal Agrowiralodra*, 2(1), 16-22.
 13. Toko Tanaman. (2023). Pengendalian penyakit karat daun pada kopi (*Hemileia vastatrix*). *Toko Tanaman*. Diakses dari <https://blog.tokotanaman.com/pengendalian-penyakit-karat-daun-pada-kopi-hemileia-vastatrix/>
 14. www.kaggle.com
 15. Wahyuningtyas, B., Tritoasmoro, I. I., & Ibrahim, N. (2022). Identifikasi penyakit pada daun kopi menggunakan metode local binary pattern dan random forest. *e-Proceeding of Engineering*, 8(6), 2972-2992.
 16. O. G. Filho, "Coffee leaf miner resistance," pp. 110–117, 2006.
 17. O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*. Retrieved from <https://www.researchgate.net/publication/285164623>
 18. Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *Proceedings of the International Conference on Engineering and Technology (ICET2017)*, Antalya, Turkey. IEEE. <https://doi.org/10.1109/ICET.2017.978-1-5386-1949-0>
 19. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90. <https://doi.org/10.1145/3065386>
 20. Sun, M., Song, Z., Jiang, X., Pan, J., & Pang, Y. (2016). Learning pooling for convolutional neural network. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2016.10.049>
 21. Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(5), 611–629. <https://doi.org/10.1007/s13244-018-0639-9>
 22. Basha, S. H. S., Dubey, S. R., Pulabaigari, V., & Mukherjee, S. (2019). Impact of fully connected layers on performance of convolutional neural networks for image classification. *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2019.10.008>
 23. Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.
 24. Woo, S., & Lee, C. L. (2018). Decision boundary formation of deep convolution networks with ReLU. In *2018 IEEE* (pp. 1-6). IEEE. <https://doi.org/10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00-13>
 25. Yuan, Z., Li, J., Li, Z., Ding, C., Ren, A., Qiu, Q., Draper, J., & Wang, Y. (2017). Softmax regression design for stochastic computing based deep convolutional neural networks. In *Proceedings of the GLSVLSI '17 Conference* (pp. 467-472). ACM. <https://doi.org/10.1145/3060403.3060467>
 26. Poynton, C., & Funt, B. (2013). Perceptual uniformity in digital image representation and display. *Color Research and Application*, 00(00), 000-000. <https://doi.org/10.1002/col.2176>