

Identifikasi Penyakit Daun Jeruk Siam Menggunakan *Convolutional Neural Network* (CNN) dengan Arsitektur EfficientNet

Burhan Syarif Acarya¹, Amri Muhaimin^{2✉}, Kartika Maulida Hindrayani³

^{1,2,3} Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia

Informasi Artikel	ABSTRAK
Riwayat Artikel Diserahkan : 14-03-2024 Direvisi : 22-03-2024 Diterima : 25-03-2024	Jeruk siam menjadi salah satu komoditas hortikultura yang memegang peranan utama dalam sektor pertanian Indonesia dengan jumlah produksi yang mencapai 2 juta ton setiap tahunnya. Namun, produksi jeruk siam rentan terhadap serangan hama dan penyakit, terutama pada bagian daun. Penyakit yang umum terjadi termasuk <i>Blackspot Leaf</i> , <i>Canker Leaf</i> , <i>Greening Leaf</i> , <i>Powdery Mildew</i> , dan <i>Citrus Leafminer</i> . Pada umumnya identifikasi penyakit pada tanaman jeruk dilakukan secara manual sehingga penentuan penyakit cenderung subyektif. Oleh karena itu, diperlukan solusi otomatis dalam mendeteksi penyakit pada daun jeruk. Tujuan penelitian yaitu untuk mengidentifikasi penyakit yang menyerang daun jeruk menggunakan metode <i>deep learning</i> yaitu CNN dengan arsitektur EfficientNetB3. Dataset yang digunakan adalah citra penyakit daun jeruk yang diambil langsung dari kebun jeruk yang dibagi menjadi 6 kelas seperti pada penyakit yang disebutkan di atas. Hasil penelitian menggunakan skenario <i>epoch</i> 10 dengan <i>optimizer</i> Adam memperoleh hasil akurasi terbaik yaitu 0,98 (98%).
Kata Kunci: CNN, EfficientNet, Identifikasi, Penyakit Daun Jeruk Siam	ABSTRACT <i>The siam orange is one of the main horticultural commodities in the Indonesian agricultural sector, with a production volume reaching 2 million tons annually. However, siam orange production is vulnerable to pest and disease attacks, especially on the leaves. Common diseases include Blackspot Leaf, Canker Leaf, Greening Leaf, Powdery Mildew, and Citrus Leafminer. Typically, disease identification in citrus plants is done manually, leading to subjective determination of diseases. Therefore, an automated solution is needed to detect diseases on orange leaves. The research aims to identify diseases affecting orange leaves using deep learning methods, specifically CNN with EfficientNetB3 architecture. The dataset used consists of images of citrus leaf diseases taken directly from orange groves, divided into 6 classes corresponding to the mentioned diseases. The research results using a scenario of 10 epochs with Adam optimizer obtained the best accuracy result of 0.98 (98%).</i>
Keywords : CNN, EfficientNet, Identification, Siam Orange Leaf Diseases	
Corresponding Author : Amri Muhaimin Sains Data, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia Jl. Rungkut Madya No.1, Gn. Anyar, Gn. Anyar, Surabaya Email: amri.muhaimin.stat@upnjatim.ac.id	

PENDAHULUAN

Jeruk (*Citrus sinensis* Osbeck) yaitu buah yang berasal dari benua Asia, seperti India, Indo-Cina, dan Indonesia (Sitanggang, 2021). Jeruk siam (*Citrus nobilis* var. *macrocarpa*) merupakan komoditas hortikultura yang populer dan sering ditemui di Indonesia. Tanaman jeruk siam mampu berproduksi dan tumbuh dengan baik di berbagai jenis lahan, baik dataran rendah maupun tinggi, serta dapat ditanam di lahan persawahan maupun tegalan (Aluhariandu et al., 2016). Jeruk

ini menjadi pilihan utama dalam kegiatan budidaya karena memberikan nilai ekonomi yang tinggi, serta diminati oleh berbagai kelompok masyarakat sebagai sumber vitamin C (Kristiandi et al., 2021). Di Indonesia, hasil produksi jeruk siam tergolong sangat besar yaitu mencapai 2 juta ton per tahunnya. Tercatat hasil produksi jeruk siam dari tahun 2019 sampai tahun 2022, yaitu pada 2019 sebesar 2.444.516 ton, 2020 sebesar 2.593.834 ton, 2021 sebesar 2.401.064, dan 2022 sebesar 2.717.447 ton. Pada tahun 2019 hingga 2020 produksi jeruk siam mengalami peningkatan dan terjadi penurunan produksi pada tahun 2021, dan kemudian pada 2022 terjadi peningkatan lagi (Badan Pusat Statistik Indonesia, 2023).

Faktor yang mempengaruhi produksi jeruk siam, diantaranya yaitu luas lahan, kesuburan tanah, kondisi iklim, umur tanaman, kondisi fisik tanaman, jumlah pohon, modal dan tenaga kerja (Setiadi et al., 2023). Selain itu, ada faktor utama yang menyebabkan turunnya hasil produksi jeruk yaitu serangan hama dan penyakit (Lestari et al., 2018), terutama pada daun jeruk. Daun rentan terhadap serangan jamur, bakteri, dan virus, termasuk *Blackspot Leaf*, *Canker Leaf*, *Greening Leaf*, *Powdery Mildew*, dan *Citrus Leafminer*. Sejumlah penyakit ini memiliki ciri khas tertentu pada daun, seperti bintik hitam (Ladaniya, 2023), cincin kuning kehijauan (Berk, 2016), bintik asimetris (Vashisth & Kadyampakeni, 2020), tepung putih (Puspito et al., 2018), telur, dan larva (Dpi, 2022). Pada umumnya petani atau masyarakat dalam mendeteksi penyakit pada tanaman jeruk dilakukan secara manual yaitu dengan melihatnya secara langsung berdasarkan pengamatannya sehingga penentuan penyakit tersebut bersifat subyektif. Maka dari itu, diperlukan upaya untuk mengembangkan solusi yang dapat secara otomatis mengenali penyakit pada tanaman jeruk agar terhindar dari kesalahan dan kekhawatiran bagi petani dan masyarakat (Febrinanto et al., 2018). Pada penelitian ini hanya berfokus pada objek tanaman jeruk yaitu berupa daun jeruk yang terkena penyakit (*Blackspot Leaf*, *Canker Leaf*, *Greening Leaf*, *Powdery Mildew*, dan *Citrus Leafminer*) dan daun jeruk sehat (*Healthy Leaf*). Dalam menghadapi perkembangan teknologi, inovasi dalam bidang pengolahan citra digital menjadi semakin relevan. Pengolahan citra digital melibatkan *Computer Vision*, sebuah ilmu yang memungkinkan komputer mengenali objek (Wijaya Kusuma et al., 2023). *Computer Vision* dapat menyelesaikan tugas dengan menggunakan metode untuk mengakuisisi, memproses, menganalisis, dan mengekstrak data berdimensi tinggi dari gambar, video, teks, dan lainnya (Riyantoko et al., 2021). Metode Convolutional Neural Network (CNN) dalam *Computer Vision* memungkinkan deteksi fitur citra yang lebih akurat (Abdullah et al., 2022). Arsitektur CNN, seperti EfficientNet, dirancang khusus untuk menangkap fitur-fitur signifikan dari gambar, menjadikannya ideal untuk tugas klasifikasi gambar, deteksi objek, dan pengenalan gambar (Idhom et al., 2023).

Beberapa penelitian terdahulu telah mengeksplorasi deteksi penyakit pada tanaman jeruk. Penelitian tentang penggunaan CNN dengan arsitektur Alexnet dalam menganalisis citra buah jeruk berhasil mencapai Tingkat akurasi sebesar 94,34%, dengan total 1790 data yang terbagi menjadi empat kelas (*Blackspot*, *Canker*, *Greening*, dan *Healthy*) (Swasono et al., 2023). Selanjutnya, penelitian mengenai citra daun jeruk menggunakan arsitektur MobileNet pada platform mobile mencapai tingkat akurasi 98% dengan membagi gambar menjadi tiga kelas (Normal, HLB, dan CTV), menggunakan optimasi RMSprop, learning rate 0,001, dan melakukan 15 epoch (Riswandi et al., 2021). Penelitian lainnya tentang deteksi penyakit daun jeruk siam, menggunakan CNN dengan gambar terbagi menjadi tiga kelas (CVPD, embun jelaga, dan ulat peliang) dan epoch 35, menghasilkan akurasi 94% (Kurniawati et al., 2023). Penelitian terkait arsitektur EfficientNet juga telah dilakukan untuk identifikasi 29 jenis penyakit pada tanaman dan buah jeruk. Pemanfaatan deep learning EfficientNetB3 dalam mengidentifikasi 29 jenis penyakit tanaman, dengan tahapan training menggunakan 57.067 gambar, validation 3.170 gambar, dan testing 3.171 gambar, mendapatkan hasil akurasi mencolok sebesar 99% (Karno et al., 2022). Sementara pada penelitian mengenai sistem diagnosis penyakit buah jeruk dengan basis CNN EfficientNet, diperoleh akurasi lebih dari 95%, melebihi VGG-U-Net yang hanya mencapai akurasi 87,66% (Huang et al., 2023). Temuan ini mengindikasikan bahwa model CNN, termasuk EfficientNet, memiliki potensi tinggi dengan akurasi di atas 90%. Sehingga tujuan dari penelitian ini adalah untuk mengenali penyakit yang menyerang daun jeruk siam menggunakan CNN dengan arsitektur EfficientNet dan juga

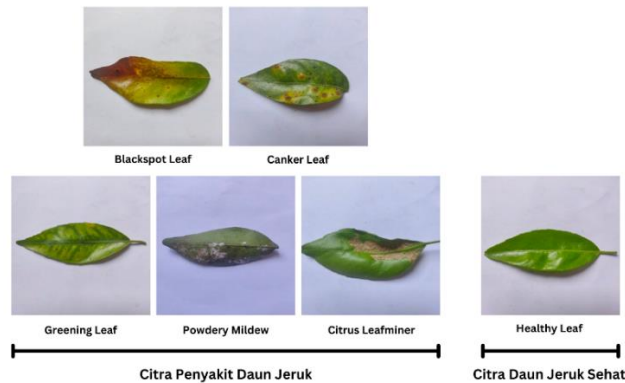
METODE PENELITIAN

The diagram illustrates the workflow for developing an EfficientNet model, consisting of five main stages:

- Pengumpulan Data** (Data Collection): Represented by a camera icon and a grid of various bird images.
- Preprocessing Data**: Represented by a dashed box around a landscape image and a Venn diagram with three overlapping circles labeled R (Red), G (Green), and B (Blue).
- Pembuatan Model EfficientNet** (Model Creation): Shown as a complex neural network architecture with multiple layers of MBConv and Conv modules, including channel counts and kernel sizes (e.g., 14x14x12, 22x12x3).
- Pelatihan Model EfficientNet** (Model Training): Represented by a gear icon and a bar chart showing performance metrics over time.
- Evaluasi Model EfficientNet** (Model Evaluation): Represented by a 2x2 confusion matrix with cells labeled TP (True Positive), FP (False Positive), FN (False Negative), and TN (True Negative).

Pada Gambar 1. Tahapan Penelitian dimulai dari pengumpulan data yang dilakukan di Kebun Jeruk Siam, Kecamatan Semboro, Kabupaten Jember, kemudian *preprocessing data* yaitu melakukan normalisasi gambar dan pengaturan skala warna menjadi RGB, pembuatan model EfficientNet dilakukan secara *pretrained*, pelatihan model EfficientNet untuk melakukan skenario pengujian, dan evaluasi model EfficientNet untuk mengetahui tingkat akurasi dari model dan mengevaluasi performa model dalam proses klasifikasi.

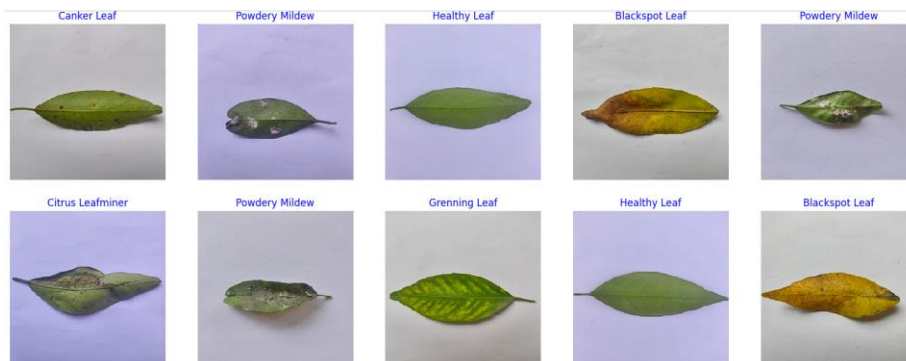
Pada penelitian ini menggunakan peralatan seperti buku gambar A3 putih dan kamera *handphone* dengan resolusi kamera sebesar 12 MP. Pengambilan gambar dilakukan dari pukul 12.00 sampai 14.00 untuk memastikan pencahayaan optimal, sehingga citra dapat terlihat terang dan jelas. Daun jeruk siam yang terkena penyakit dipetik dari pohon dan difoto langsung dengan latar belakang buku gambar A3 putih. Jumlah data citra daun jeruk siam yang digunakan yaitu sebanyak 480 data yang terbagi menjadi 3, yaitu data latih (*train*), data validasi (*validation*), dan data uji (*test*). Pada penelitian ini terbagi menjadi 6 kelas, yakni citra daun jeruk siam sehat (*Healthy*), citra penyakit kanker (*Canker Leaf*), citra penyakit jamur pathogen (*Blackspot Leaf*), citra penyakit gangguan kloroplas (*Greening Leaf*), citra penyakit jamur tepung putih (*Powdery Mildew*), dan citra penyakit oleh ngengat kecil (*Citrus Leafminer*) seperti pada Gambar 2. di bawah ini.



Gambar 2. Citra Penyakit dan Citra Sehat pada Daun Jeruk Siam

Preprocessing Data

Pada tahap ini dilakukan normalisasi ukuran gambar sebelum dilakukan *training* menjadi ukuran 256 x 256 pixel dan melakukan pengaturan skala warna menjadi RGB (*red, green, blue*). Normalisasi ukuran gambar menggunakan bantuan *software resize pixel* yang ada di Google dan pengaturan skala warna menjadi RGB menggunakan perintah konversi warna yang ada di dalam program Google Colab. Tujuan dilakukan tahapan ini yaitu untuk memastikan konsistensi ukuran gambar pada saat melakukan *input* pada proses *training model*, efisiensi komputasi, serta interpretasi yang tepat terhadap informasi warna dalam model *neural network*. Berikut ini merupakan hasil *preprocessing data* pada Gambar 3.



Gambar 3. Preprocessing Data

Pembuatan Model EfficientNet

EfficientNet merupakan bagian arsitektur dari Convolutional Neural Network (CNN) yang dikembangkan oleh Quoc V. Le dan Mingxin Tan pada tahun 2019 (Tan & Le, 2019). Arsitektur ini menggabungkan konvolusi *pointwise* dan *depthwise*. Ada tiga dimensi penskalaan pada EfficientNet, yaitu resolusi (*resolution*), lebar (*width*), dan kedalaman (*depth*). EfficientNet terdiri dari tujuh blok, dengan masing-masing blok mengandung beberapa sub-blok. Untuk menjaga konsistensi dalam menyesuaikan lebar, kedalaman, dan resolusi jaringan, EfficientNet mengaplikasikan sebuah koefisien gabungan ϕ untuk menskalakan aspek-aspek tersebut secara merata. Persamaan (1) hingga (3) di bawah ini menggambarkan notasi yang digunakan untuk kedalaman, lebar, dan resolusi:

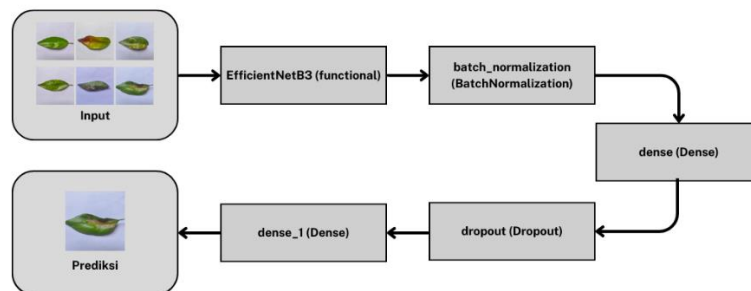
$$d = \alpha^\phi \quad (1)$$

$$w = \beta^\phi \quad (2)$$

$$r = \gamma^\phi \text{ s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \quad (3)$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

Koefisien α, β, γ dapat disesuaikan melalui pencarian grid terkecil. Sementara itu, koefisien ϕ merupakan faktor yang ditentukan pengguna untuk mengatur dan mengalokasikan sumber daya yang digunakan untuk penskalaan model. Sementara itu, α, β, γ digunakan untuk menentukan bagaimana cara menambahkan tambahan sumber daya untuk kedalaman (*depth*), lebar (*width*), dan resolusi (*resolution*). Dalam penelitian ini, penulis memanfaatkan arsitektur EfficientNetB3. Berikut adalah deskripsi model EfficientNetB3 yang digunakan dalam penelitian ini.



Gambar 4. Model EfficientNetB3

Pada Gambar 4. Di atas terdapat model EfficientNetB3. Model EfficientNetB3 ini untuk meningkatkan efisiensi model CNN dalam penggunaan sumber daya komputasi, dengan mengoptimalkan jumlah parameter yang dibutuhkan untuk mencapai akurasi yang tinggi dalam klasifikasi penyakit daun jeruk siam. Penulis menggunakan model EfficientnetB3 secara *pre-trained* (yang telah dilatih sebelumnya) dimana model sudah dilatih sebelumnya menggunakan dataset ImageNet yaitu dataset gambar besar yang terdiri dari jutaan gambar yang dikelompokkan ke dalam ribuan kategori atau kelas yang berbeda. Adapun untuk jumlah parameter serta penjelasan lapisan terdapat pada Tabel.1 di bawah ini.

Tabel 1. Deskripsi Model EfficientNetB3

Layer	Output Shape	Parameters
input_1 (InputLayer)	224, 224, 3	0
efficientnetb3 (Functional)	1536	10783535
batch_normalization (BatchNormalization)	1536	6144
dense (Dense)	256	393472
dropout (Dropout)	256	0
dense_1 (Dense)	6	1524

Setelah membuat model EfficientNetB3 secara *pre-trained*, kemudian penulis menambahkan beberapa lapisan untuk menyesuaikan dengan dataset yang digunakan. Untuk *input* (masukan), gambar dibuat menjadi 224x224 pixel dengan konversi warna RGB (red, green, blue). Penggunaan *BatchNormalization* bertujuan untuk mempercepat proses pelatihan dan meningkatkan stabilitas serta konsistensi model selama proses tersebut. Di dalam lapisan Dense terdapat *regularizer* L2 dan L1 yang bertujuan untuk mencegah terjadinya *overfitting*, kemudian terdapat juga *activation ReLU* yang bertujuan untuk mengatasi masalah gradien yang menghilang dan konvergensi yang cepat. *Dropout* juga digunakan untuk mencegah terjadinya *overfitting*. *Dense_1* merupakan layer *fully connected* terakhir dalam model, dengan jumlah unit yang setara dengan jumlah kelas yang ingin diprediksi yaitu sejumlah 6. Jadi, total parameter yang digunakan dalam model ini yaitu 11184693 parameter.

HASIL DAN PEMBAHASAN

Pelatihan Model EfficientNet

Pada saat pelatihan Model EfficientNet ini, dilakukan dua skenario pengujian. Pertama yaitu menggunakan parameter dengan *epoch* berjumlah 5 dan 10. Kedua menggunakan percobaan

beberapa optimasi yaitu menggunakan Adam dan RMSprop. Pengujian ini dilaksanakan untuk mengevaluasi parameter performa dan akurasi model, serta untuk menentukan model yang memberikan hasil terbaik. Jumlah data yang digunakan dalam skenario pengujian ini yaitu untuk jumlah data latih (*train*) yaitu 480 (masing-masing kelas berjumlah 80). Data validasi dan data uji berasal dari data pelatihan yang telah dilakukan *splitting* (pemisahan data) yaitu menjadi 80:20, sehingga jumlah data validasi (*validation*) yaitu 384 (masing-masing kelas berjumlah 64), dan data uji (*test*) yaitu 96 (masing-masing kelas berjumlah 16).

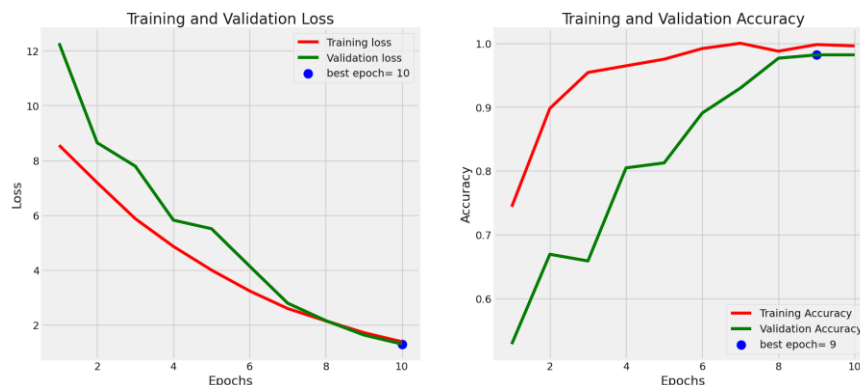
Skenario Pengujian

Skenario pertama yaitu melakukan pengujian menggunakan parameter dengan *epoch* berjumlah 5 dan 10. *Epoch* berfungsi untuk mengatur jumlah perulangan iterasi dalam proses pembaruan bobot jaringan. Penggunaan *epoch* berperan dalam mengurangi kesalahan (*error*) pada proses pelatihan model. Skenario kedua yaitu melakukan pengujian menggunakan parameter dengan optimasi Adam dan RMSprop. Pengujian ini bertujuan untuk memahami bagaimana penggunaan optimizer memengaruhi tingkat akurasi model.

Tabel 2. Hasil Skenario Pengujian Menggunakan Parameter *Epoch* dan *Optimizer*

Epoch	Optimizer	Train Loss	Train Accuracy	Validation Loss	Validation Accuracy	Test Loss	Test Accuracy
5	Adam	4,0	0,88	3,85	0,94	3,82	0,93
10	Adam	1,35	0,95	1,30	0,97	1,30	0,98
5	RMSprop	2,85	0,97	3,30	0,89	3,09	0,92
10	RMSprop	1,41	0,95	1,41	0,98	1,37	0,97

Dari hasil Tabel 2 di atas, terlihat hasil pengujian dengan skenario parameter *epoch* dan *optimizer* yang paling baik yaitu pada skenario pengujian menggunakan *epoch* 10 dengan *optimizer* Adam. Terlihat bahwa hasil *Train Loss* 1,35; *Train Accuracy* 0,95; *Validation Loss* 1,30; *Validation Accuracy* 0,97; *Test Loss* 1,30; dan *Test Accuracy* 0,98.



Gambar 5. Grafik Loss dan Accuracy pada Training dan Validation

Pada Gambar 5 menampilkan hasil akurasi (*accuracy*) dan nilai kerugian (*loss*) pada data pelatihan (*training*) dan data validasi (*validation*). Grafik *loss* digunakan untuk mengukur seberapa besar kesalahan pada model yang sedang dievaluasi. Nilai *loss* yang baik yaitu jika mendekati 0. Terlihat bahwa model EfficientNet dengan *epoch* 10 dan *optimizer* Adam menghasilkan nilai *loss* untuk data latih (*train*) dan data validasi (*validation*) masing-masing yaitu 1,35 dan 1,30. Sedangkan untuk grafik akurasi digunakan untuk mengetahui seberapa baik model yang digunakan. Akurasi yang baik yaitu jika mendekati 100%. Terlihat bahwa model EfficientNet dengan *epoch* 10 dan *optimizer* Adam menghasilkan akurasi untuk data latih (*train*) dan data validasi (*validation*) masing-masing yaitu 0,95 dan 0,97.

Evaluasi Model EfficientNet

Setelah melakukan proses pengujian model menggunakan beberapa skenario pengujian dan ditemukan hasil yang terbaik, langkah selanjutnya adalah mengevaluasi model menggunakan data uji (*test*). Proses ini bertujuan untuk mengukur berbagai metrik kinerja seperti akurasi *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. *Accuracy* menghitung tingkat ketepatan klasifikasi model. *Precision* menghitung presentase dari prediksi positif yang akurat dari semua prediksi positif. *Recall* menghitung presentase positif yang benar dari keseluruhan data yang sebenarnya positif. *F1-Score* adalah perbandingan nilai rata-rata antara *Precision* dan *Recall*. Confusion matrix digunakan untuk memperoleh detail yang lebih dalam mengenai performa model. Rumus *confusion matrix* yaitu sebagai berikut:

$$Accuracy = \frac{TN+TP}{TN+FP+FN+TP} \quad (4)$$

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

$$Precision = \frac{TP}{TP+FN} \quad (6)$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (7)$$

TP (*True Positive*) merupakan jumlah data positif yang berhasil diklasifikasikan dengan benar oleh model, sedangkan FN (*False Negative*) adalah jumlah data positif yang tidak terdeteksi dengan benar sebagai negatif oleh model. FP (*False Positive*) merujuk pada jumlah data negatif yang keliru diklasifikasikan sebagai positif oleh model, dan TN (*True Negative*) adalah jumlah data negatif yang berhasil diklasifikasikan dengan benar oleh model. *F1-Score* mewakili rata-rata *precision* dan *recall*, menunjukkan kinerja dari model dalam klasifikasi yang relatif akurat.

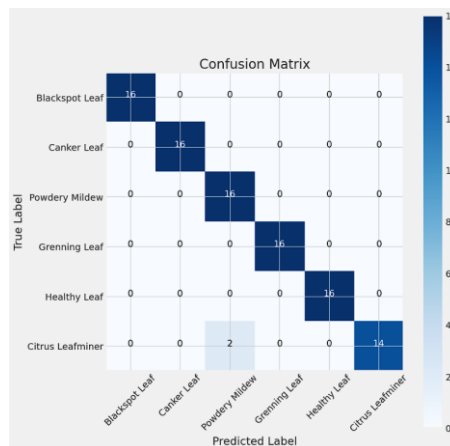
Evaluasi Confusion Matrix

Confusion Matrix adalah suatu metode visualisasi yang digunakan untuk mengevaluasi performa model dalam proses klasifikasi pada *machine learning*. *Confusion Matrix*, juga dikenal sebagai *error matrix*, memberikan gambaran tentang seberapa baik atau buruk model dalam memprediksi kelas dari data yang diuji. Berikut adalah tabel *Confusion Matrix* yang menunjukkan evaluasi performa model terhadap data uji (*test*).

Tabel 3. Hasil Pengukuran dengan Confusion Matrix

	Precision	Recall	F1-Score	Support
Blackspot Leaf	1,00	1,00	1,00	16
Canker Leaf	1,00	1,00	1,00	16
Powdery Mildew	0,89	1,00	0,94	16
Greening Leaf	1,00	1,00	1,00	16
Healthy Leaf	1,00	1,00	1,00	16
Citrus Leafminer	1,00	0,88	0,93	16
Accuracy			0,98	96
Macro AVG	0,98	0,98	0,98	96
Weighted AVG	0,98	0,98	0,98	96

Pada Tabel 3. di atas terlihat bahwa hasil pengukuran data uji (*testing*) dengan *confusion matrix*. Untuk melihat bagaimana data uji (*testing*) bisa diprediksi dengan benar maupun salah bisa dilihat pada *confusion matrix* di bawah.



Gambar 6. Hasil Prediksi oleh Confusion Matrix

Pada Gambar 6. menunjukkan hasil prediksi yang diperoleh oleh model menggunakan skenario pengujian dengan pengaturan *epoch* 10 dan *optimizer* Adam, mendapatkan hasil yang sangat baik yaitu mendapatkan akurasi sebesar 0,98 (98%). Terlihat dari 6 kelas (*Blackspot Leaf*, *Canker Leaf*, *Powdery Mildew*, *Greening Leaf*, *Healthy Leaf*, dan *Citrus Leafminer*) hampir semua data diprediksi dengan benar. Masing-masing kelas sudah bisa diprediksi dengan benar oleh model dengan jumlah data sebanyak 16 data uji (*testing*), hanya saja terdapat 1 kelas yaitu *Citrus Leafminer* yang diprediksi benar oleh model sejumlah 14 dan yang diprediksi salah sejumlah 2.

KESIMPULAN DAN SARAN

Penelitian yang menggunakan model CNN dengan arsitektur EfficientNetB3 untuk identifikasi penyakit daun jeruk siam telah berhasil melakukan prediksi gambar dengan baik. Penelitian ini menggunakan skenario pengujian dengan menggunakan parameter *epoch* 5, 10 dan *optimizer* Adam, RMSprop. Didapatkan hasil bahwa skenario pengujian menggunakan *epoch* 10 dengan *optimizer* Adam menghasilkan akurasi yang paling tinggi yaitu mencapai 0,98 (98%). Selain itu, penggunaan Model EfficientNet secara *pre-trained* juga bisa berpengaruh terhadap hasil akurasi karena model sebelumnya sudah dilatih menggunakan dataset ImageNet.

Berdasarkan simpulan dari penelitian ini, maka masih ditemukan permasalahan yaitu terdapat gambar yang masih diprediksi salah dalam pengklasifikasian. Oleh karena itu, untuk penelitian yang akan datang diharapkan untuk melakukan optimalisasi terhadap model yang sudah dibuat atau melakukan beberapa uji coba skenario yang lebih banyak agar mendapatkan model yang lebih optimal. Selain itu, diharapkan untuk menggunakan model EfficientNet secara *scratch* (membangun model dari awal dan mengatur bobot secara mandiri) untuk mengetahui apakah hasil akurasi model bisa baik seperti pada model *pre-trained*.

REFERENSI

- Abdullah, S. H., Magdalena, R., & Fu'adah, R. Y. N. (2022). Klasifikasi Diabetic Retinopathy Berbasis Pengolahan Citra Fundus Dan Deep Learning. *Journal of Electrical and System Control Engineering*, 5(2), 84–90. <https://doi.org/10.31289/jesce.v5i2.5659>
- Aluhariandu, V. E., Tariningsih, D., Fajar, D. P., & Lestari, K. (2016). Analisis USAhatani Jeruk Siam Dan Faktor “Faktor Yang Mempengaruhi Penerimaan Petani (Studi Kasus Di Desa Bayung Gede Kecamatan Kintamani Kabupaten Bangli). *Agrimeta*, 6(12), 90058. <https://www.neliti.com/publications/90058/>
- Badan Pusat Statistik Indonesia. (2023). Statistik Indonesia 2023. In D. D. Statistik (Ed.), *Badan Pusat Statistik* (Vol. 1101001). Badan Pusat Statistik. <https://www.bps.go.id/publication/2020/04/29/e9011b3155d45d70823c141f/statistik-indonesia-2020.html>
- Berk, Z. (2016). Chapter 5 - Diseases and pests. In Z. Berk (Ed.), *Citrus Fruit Processing* (pp. 83–93).

- Academic Press. <https://doi.org/https://doi.org/10.1016/B978-0-12-803133-9.00005-9>
- Dpi, N. S. W. (2022). *primefact Citrus leaf miner*. 126(March).
- Febrinanto, F. G., Dewi, C., & Wiratno, A. T. (2018). Implementasi Algoritme K-Means Sebagai Metode Segmentasi Citra Dalam Identifikasi Penyakit Daun Jeruk. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 2(11), 5375–5383.
- Huang, Z., Jiang, X., Huang, S., Qin, S., & Yang, S. (2023). An efficient convolutional neural network-based diagnosis system for citrus fruit diseases. *Frontiers in Genetics*, 14(August), 1–11. <https://doi.org/10.3389/fgene.2023.1253934>
- Idhom, M., Prasetya, D. A., Riyantoko, P. A., Fahrudin, T. M., & Sari, A. P. (2023). Pneumonia Classification Utilizing VGG-16 Architecture and Convolutional Neural Network Algorithm for Imbalanced Datasets. *TIERS Information Technology Journal*, 4(1), 73–82. <https://doi.org/10.38043/tiers.v4i1.4380>
- Karno, A., Hastomo, W., Wardhana, I., Sutarno, & Arif, D. (2022). 29 Jenis Penyakit Tanaman Menggunakan Deep Learning EfficientNetB3 Identifikasi. *Insearch: Information System Research Journal*, 2(2), 35–45.
- Kristiandi, K., Fertiasari, R., Yunita, N. F., Astuti, T. W., & Sari, D. (2021). Analisis Produktivitas Dan Luas Tanaman Jeruk Siam Sambas Tahun 2015-2020. *Mimbar Agribisnis: Jurnal Pemikiran Masyarakat Ilmiah Berwawasan Agribisnis*, 7(2), 1747. <https://doi.org/10.25157/ma.v7i2.5607>
- Kurniawati, D., Wulaningrum, R., & Sahertian, J. (2023). *Perancangan Sistem Deteksi Penyakit Daun Jeruk Siam*. 7, 1241–1247.
- Ladaniya, M. (2023). Chapter 4 - Preharvest factors. In M. Ladaniya (Ed.), *Citrus Fruit (Second Edition)* (Second Edi, pp. 113–144). Academic Press. <https://doi.org/https://doi.org/10.1016/B978-0-323-99306-7.00010-4>
- Lestari, F. R., Sari, J. Y., Sutardi, Purwanti, I., & Purnama, N. (2018). Deteksi Penyakit Tanaman Jeruk Siam Berdasarkan Citra Daun Menggunakan Segmentasi Warna RGB-HSV. *Teknologi Terapan Berbasis Kearifan Lokal (Snt2bkl)*, December, 276–283.
- Puspito, M. A., Hidayat, N., & Suprpto. (2018). Sistem Pendukung Keputusan Diagnosa Penyakit Tanaman Jeruk Menggunakan Metode Naive Bayes Classifier. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(7), 1–6.
- Riswandi, R., Jamiah, R., Mardhatillah, N., & Hamid, H. P. (2021). Klasifikasi Penyakit Pada Citra Daun Jeruk Menggunakan Arsitektur MobileNet berbasis Mobile Platform. *Jurnal Fokus Elektroda : Energi Listrik, Telekomunikasi, Komputer, Elektronika Dan Kendali*, 6(4), 212. <https://doi.org/10.33772/jfe.v6i4.19113>
- Riyantoko, P. A., Sugiarto, & Hindrayani, K. M. (2021). Facial Emotion Detection Using Haar-Cascade Classifier and Convolutional Neural Networks. *Journal of Physics: Conference Series*, 1844(1). <https://doi.org/10.1088/1742-6596/1844/1/012004>
- Setiadi, A., Gafaruddin, A., & Slamet, A. (2023). Faktor-Faktor yang Mempengaruhi Produksi Usahatani Jeruk Manis Siam Madu (*Citrus sinensis nobilis*) di Desa Tanea Kecamatan Konda Kabupaten Konawe Selatan. *Jurnal Ilmiah Membangun Desa Dan Pertanian*, 8(3), 88–95. <https://doi.org/10.37149/jimdp.v8i3.79>
- Sitanggang, K. D. (2021). *KULTUR ANTERA JERUK (I)*. CV Literasi Nusantara Abadi.
- Swasono, D. I., Wijaya, M. A. R., & Hidayat, M. A. (2023). Klasifikasi Penyakit pada Citra Buah Jeruk Menggunakan Convolutional Neural Networks (CNN) dengan Arsitektur Alexnet. *INFORMAL: Informatics Journal*, 8(1), 68. <https://doi.org/10.19184/isj.v8i1.38563>
- Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *36th International Conference on Machine Learning, ICML 2019, 2019-June*, 10691–10700.
- Vashisth, T., & Kadyampakeni, D. (2020). Chapter 49 - Diagnosis and management of nutrient constraints in citrus. In A. K. Srivastava & C. Hu (Eds.), *Fruit Crops* (pp. 723–737). Elsevier. <https://doi.org/https://doi.org/10.1016/B978-0-12-818732-6.00049-6>
- Wijaya Kusuma, W., Rizal Isnanto, R., Fauzi, A., & Korespondensi, P. (2023). Analisis Perbandingan Model Cnn Vgg16 Dan Densenet121 Menggunakan Kerangka Kerja Tensorflow Untuk Deteksi Jenis Hewan. *Jurnal Teknik Komputer*, 1(4), 141–147.