

Perbandingan Arsitektur CNN untuk Klasifikasi Penyakit Pada Buah Mangga Studi Kasus: MobileNetV3, EfficientNetB3, dan ResNet-18

Andita Dani Achmad^{1*}, Arysespajayadi², Sofyan Tandungan³, Muh. Takdir Muslihi⁴, Andi Triska Muliana⁵

¹Informatika, Universitas Fajar

²Teknologi Industri, Politeknik Negeri Media Kreatif

³Teknik Informatika, Universitas Hasanuddin

⁴Teknik Listrik dan Instalasi, AK-Manufaktur Bantaeng

⁵Teknik Listrik, Politeknik Kotabaru

Email: ¹anditadaniachmad@gmail.com, ²arysespajayadi@gmail.com,

³sofyantandungan@unhas.ac.id, ⁴mtakdir.muslihi@gmail.com, ⁵anditriskamuliana@poltekab.ac.id

*Penulis Korespondensi

Abstract—Mango fruit diseases, such as anthracnose and stem-end rot, can significantly reduce crop quality and yield. Conventional disease identification is often subjective and difficult to implement on a large scale. This study aims to compare the performance of three Convolutional Neural Network (CNN) architectures, MobileNetV3, EfficientNetB3, and ResNet-18 in classifying mango fruit diseases based on digital images. The method employed is a quantitative experiment involving dataset collection, image preprocessing, and model training using the Mango Fruit DDS dataset, which includes five classes: *Alternariose*, *Anthracnose*, *Aspergillus Rot*, *Healthy*, and *Stem End Rot*. The results indicate that EfficientNetB3 provides the most optimal performance, achieving the highest validation accuracy of 0.9698 (best accuracy of 0.9984) and superior computational efficiency compared to the other models. These findings conclude that EfficientNetB3 is the most suitable architecture for developing accurate and efficient mango disease detection systems in real-world agricultural environments.

Intisari— Penyakit pada buah mangga, seperti antraknosa dan busuk ujung batang, secara signifikan dapat menurunkan kualitas dan hasil panen. Identifikasi penyakit secara konvensional sering kali bersifat subjektif dan sulit diterapkan dalam skala besar. Penelitian ini bertujuan untuk membandingkan kinerja tiga arsitektur Convolutional Neural Network (CNN), yaitu MobileNetV3, EfficientNetB3, dan ResNet-18, dalam mengklasifikasikan penyakit pada buah mangga berbasis citra digital. Metode yang digunakan adalah eksperimen kuantitatif dengan tahapan meliputi pengumpulan dataset, pra-pemrosesan citra, serta pelatihan model menggunakan dataset *Mango Fruit DDS* yang mencakup lima kelas: *Alternariose*, *Anthracnose*, *Aspergillus Rot*, *Healthy*, dan *Stem End Rot*. Hasil penelitian menunjukkan bahwa EfficientNetB3 memberikan performa paling optimal dengan akurasi validasi tertinggi mencapai 0,9698 (akurasi terbaik 0,9984) serta efisiensi komputasi yang unggul dibandingkan model lainnya. Temuan ini menyimpulkan bahwa EfficientNetB3 merupakan arsitektur yang paling sesuai untuk pengembangan sistem deteksi penyakit mangga yang akurat dan efisien pada lingkungan pertanian nyata

Kata Kunci— Mangga, Klasifikasi Penyakit, CNN, MobileNetV3, EfficientNetB3, ResNet-18

I. PENDAHULUAN

Buah mangga (*Mangifera indica* L.) merupakan salah

satu komoditas hortikultura unggulan di negara-negara tropis, termasuk Indonesia, dengan kontribusi ekonomi yang signifikan pada sektor pertanian dan ekspor. Namun, produktivitas dan kualitas buah mangga sangat dipengaruhi oleh berbagai penyakit seperti *anthracnose*, *powdery mildew*, dan *stem end rot* yang dapat menurunkan hasil panen secara drastis apabila tidak dideteksi sejak dini [1].

Masing-masing patogen penyakit mangga menunjukkan manifestasi klinis yang berbeda. Antraknosa, yang disebabkan oleh *Colletotrichum gloeosporioides*, ditandai dengan munculnya lesi berwarna coklat hingga hitam dengan pola penyebaran tidak teratur pada permukaan buah [2]. *Alternariose* atau bintik hitam merupakan hasil infeksi jamur sejak fase buah muda yang baru terlihat selama pematangan [3]. Rotasi *Aspergillus* adalah penyakit pascapanen yang memicu pembusukan jaringan buah akibat infeksi *Aspergillus niger* [4]. Di sisi lain, stem end rot (SER) merupakan penyakit kritis yang menginfeksi ujung tangkai buah dan merambat ke jaringan internal, disebabkan oleh patogen seperti *Lasiodiplodia theobromae* [4].

Metode identifikasi penyakit mangga secara tradisional umumnya mengandalkan observasi visual oleh ahli patologi tanaman. Pendekatan ini memiliki sejumlah kelemahan, di antaranya bersifat subjektif, membutuhkan keahlian tinggi, dan kurang aplikatif untuk skala besar. Selain itu, sifat buah mangga yang bersifat klimakterik meningkatkan kerentanan terhadap pembusukan pascapanen, sehingga metode deteksi yang cepat dan akurat menjadi sangat esensial [5].

Kemajuan dalam bidang computer vision dan deep learning khususnya Convolutional Neural Networks (CNN), telah memfasilitasi otomatisasi dalam klasifikasi penyakit tanaman berbasis citra digital. Arsitektur CNN mampu melakukan ekstraksi fitur visual secara otomatis dan telah terbukti unggul dibandingkan metode berbasis fitur manual (hand-crafted features) dalam berbagai studi klasifikasi penyakit tanaman [6], [7]. Beberapa penelitian terdahulu juga menegaskan bahwa CNN dapat mencapai akurasi tinggi dalam klasifikasi penyakit daun dan buah pada berbagai komoditas pertanian [8].

Beragam arsitektur CNN telah dikembangkan untuk meningkatkan performa sekaligus efisiensi komputasi. MobileNetV3 merupakan arsitektur ringan yang dioptimalkan untuk perangkat dengan keterbatasan sumber

daya melalui pemanfaatan depthwise separable convolution dan mekanisme squeeze-and-excitation [9], [10]. EfficientNet memperkenalkan pendekatan compound scaling yang secara sistematis menyeimbangkan kedalaman, lebar, dan resolusi jaringan untuk mencapai performa optimal dengan parameter yang efisien [11]. Sementara itu, ResNet dengan mekanisme residual learning tetap menjadi pilihan populer sebagai baseline karena stabilitas pelatihan dan kemampuannya mengatasi masalah vanishing gradient [12].

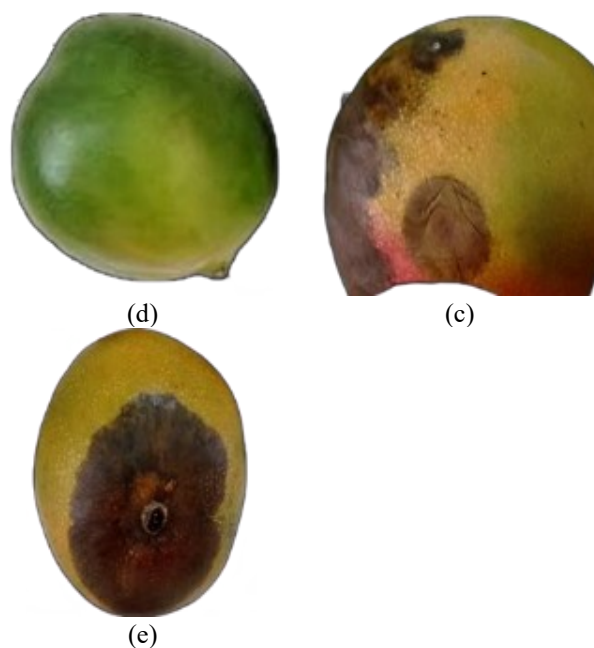
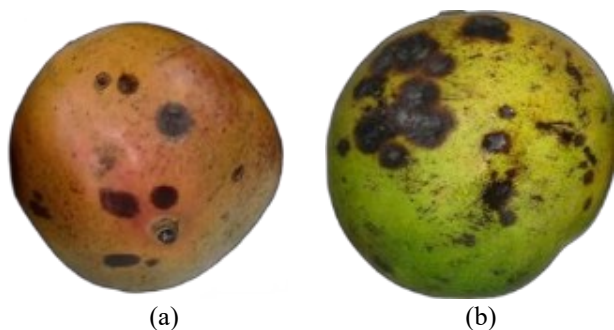
Meskipun sejumlah penelitian telah menerapkan CNN untuk klasifikasi penyakit tanaman, mayoritas di antaranya berfokus pada peningkatan akurasi tanpa memberikan perhatian yang komprehensif terhadap aspek efisiensi komputasi. Selain itu, kajian komparatif yang secara spesifik mengevaluasi kinerja arsitektur CNN ringan dan menengah dalam klasifikasi penyakit buah mangga masih sangat terbatas [13].

Oleh karena itu, terdapat kebutuhan mendesak untuk melakukan analisis komparatif yang tidak hanya mempertimbangkan akurasi, tetapi juga efisiensi komputasi dalam konteks implementasi praktis.

Adapun kontribusi utama penelitian ini meliputi penyajian analisis komparatif kinerja tiga arsitektur CNN populer pada dataset penyakit buah mangga, pengevaluasian trade-off antara akurasi dan efisiensi komputasi pada masing-masing model, dan pemberian rekomendasi arsitektur optimal untuk implementasi sistem deteksi penyakit mangga dalam konteks pertanian praktis.

II. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimen kuantitatif dengan memanfaatkan metode *Deep Learning* berbasis Convolutional Neural Network (CNN) untuk melakukan klasifikasi penyakit pada tanaman mangga berdasarkan citra digital. Fokus utama penelitian adalah membandingkan kinerja tiga arsitektur CNN, yaitu ResNet-18, MobileNetV3-Large, dan EfficientNetB3, dari sisi akurasi klasifikasi dan efisiensi komputasi. Tahapan penelitian meliputi pengumpulan dataset, pra-pemrosesan citra, perancangan dan pelatihan model, serta evaluasi performa.



Gambar. 1 Contoh gambar tiap kelas pada Mango Fruit DDS Dataset. (a) alternariose, (b) anthracnose, (c) aspergillus rot, (d) healthy, dan (e) stem end rot

A. Dataset Penelitian

Dataset yang digunakan dalam penelitian ini berasal dari dua sumber terbuka yang telah banyak digunakan dalam penelitian bidang pertanian dan visi computer, yaitu Mango Fruit DSS Dataset berisi citra buah mangga dengan berbagai jenis penuakit dan kerusakan fisik [14]. Gambar 1 menunjukkan contoh citra dari masing-masing kelas pada dataset Mango Fruit DSS yang digunakan dalam penelitian ini. Dataset terdiri dari lima kategori penyakit dan kondisi buah mangga, yaitu: (a) alternariose, yang ditandai dengan bercak hitam kecil yang menyebar pada permukaan kulit; (b) anthracnose, yang menunjukkan lesi berwarna gelap dengan pola tidak beraturan; (c) aspergillus rot, yang ditandai dengan pembusukan jaringan buah disertai perubahan warna menjadi coklat kehitaman; (d) healthy, yaitu buah mangga dalam kondisi normal tanpa gejala penyakit, serta (e) stem end rot, yang ditandai dengan pembusukan pada bagian ujung tangkai yang meluas ke jaringan buah. Seluruh data citra kemudian dibagi menjadi dua bagian, yaitu data pelatihan (training set) dan data validasi (validation set) dengan proporsi yang seimbang antar kelas untuk menghindari bias kelas selama proses pembelajaran model.

B. Pra-pemrosesan Data

Pra-pemrosesan citra dilakukan untuk memastikan kualitas data dan kesesuaian format input dengan arsitektur CNN yang digunakan. Tahapan pra-pemrosesan meliputi:

- 1) *Resize* citra ke ukuran input standar sesuai kebutuhan model.
- 2) Normalisasi nilai piksel ke rentang [0,1] untuk mempercepat konvergensi selama pelatihan.
- 3) *Augmentasi* data berupa rotasi acak, *flipping* horizontal dan vertikal, serta penyesuaian tingkat kecerahan citra untuk meningkatkan variasi data dan mengurangi risiko

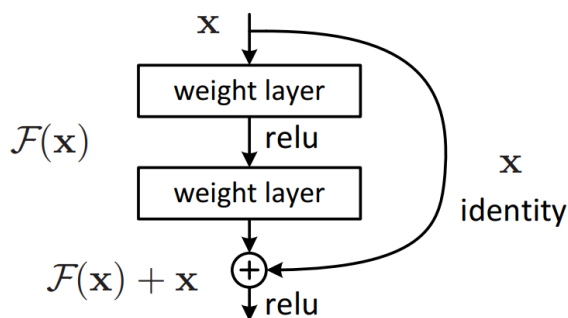
overfitting.

Tahapan ini bertujuan meningkatkan kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

C. Arsitektur Model CNN

Penelitian ini mengevaluasi tiga arsitektur Convolutional Neural Network (CNN) yang merepresentasikan kompromi berbeda antara kapasitas representasi dan efisiensi komputasi, yaitu ResNet-18, MobileNetV3-Large, dan EfficientNetB3. Pemilihan ketiga model tersebut dilakukan untuk memperoleh perbandingan yang komprehensif antara arsitektur berorientasi performa, arsitektur ringan untuk komputasi terbatas, serta arsitektur dengan strategi penskalaan yang sistematis [15].

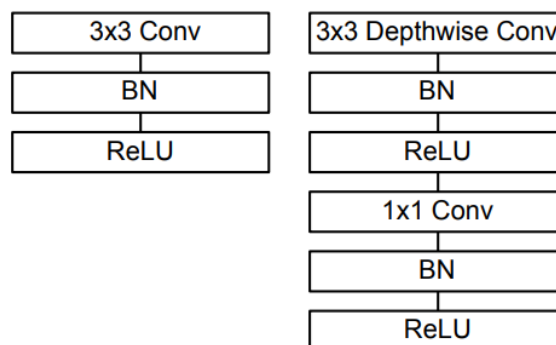
1) ResNet-18 menggunakan mekanisme residual connection (skip connection) yang memformulasikan pembelajaran sebagai fungsi residual terhadap masukan blok, proses fungsi diperlihatkan pada gambar 2. Desain ini terbukti mempermudah optimasi pada jaringan yang lebih dalam dan membantu menjaga aliran gradien selama pelatihan, sehingga mengurangi permasalahan degradation yang umum terjadi pada CNN dalam [15].



Gambar. 2 Mekanisme residual pada ResNet.

- 2) MobileNetV3-Large dirancang untuk kebutuhan komputasi yang efisien, dengan memanfaatkan prinsip arsitektur ringan MobileNet yang berbasis depthwise separable convolution untuk menekan jumlah operasi dan parameter. Selain itu, MobileNetV3 menggabungkan elemen desain yang dioptimalkan melalui pencarian arsitektur (NAS) dan memasukkan modul atensi kanal seperti squeeze-and-excitation guna meningkatkan kualitas representasi fitur tanpa peningkatan biaya komputasi yang signifikan [16, 17, 18]. Susunan arsitektur ini dapat dilihat pada tabel 1.
- 3) EfficientNetB3 menerapkan pendekatan compound scaling, yaitu strategi penskalaan yang menyeimbangkan kedalaman (depth), lebar (width), dan resolusi input secara serampak melalui koefisien terpadu. Pendekatan ini memungkinkan peningkatan akurasi yang kompetitif sekaligus mempertahankan efisiensi parameter dan komputasi dibandingkan penskalaan konvensional yang hanya menambah salah satu dimensi arsitektur [19]. Perbedaan antara konvolusi standar dan depthwise separable convolution yang menjadi dasar efisiensi pada arsitektur ringan dapat dilihat pada gambar 3.

Setiap arsitektur digunakan sebagai *feature extractor*, kemudian dimodifikasi pada lapisan klasifikasi akhir agar sesuai dengan jumlah kelas yang digunakan dalam penelitian.



Gambar. 3 Kiri: lapisan konvolusi standar yang diikuti batch normalization dan aktivasi ReLU. Kanan: konvolusi depthwise separable yang terdiri dari lapisan depthwise dan pointwise, kemudian dilanjutkan dengan batch normalization dan aktivasi ReLU.

Tabel 1. Spesifikasi untuk MobileNetV3-Large.

Input	Operator	exp size	#out	SE	NL	S
$224^2 \times 3$	conv2d	-	16	-	HS	2
$112^2 \times 16$	bneck, 3x3	16	16	-	RE	1
$112^2 \times 16$	bneck, 3x3	64	24	-	RE	2
$56^2 \times 24$	bneck, 3x3	72	24	-	RE	1
$56^2 \times 24$	bneck, 5x5	72	40	√	RE	2
$28^2 \times 40$	bneck, 5x5	120	40	√	RE	1
$28^2 \times 40$	bneck, 5x5	120	40	√	RE	1
$28^2 \times 40$	bneck, 5x5	240	80	-	HS	2
$14^2 \times 80$	bneck, 3x3	200	80	-	HS	1
$14^2 \times 80$	bneck, 3x3	184	80	-	HS	1
$14^2 \times 80$	bneck, 3x3	184	80	-	HS	1
$14^2 \times 80$	bneck, 3x3	480	112	√	HS	1
$14^2 \times 112$	bneck, 3x3	672	112	√	HS	1
$14^2 \times 112$	bneck, 5x5	672	160	√	HS	2
$7^2 \times 160$	bneck, 5x5	960	160	√	HS	1
$7^2 \times 160$	bneck, 5x5	960	160	√	HS	1
$7^2 \times 160$	conv2d, 1x1	-	960	-	HS	1
$7^2 \times 160$	pool, 7x7	-	-	-	-	1
$1^2 \times 960$	conv2d 1x1, NBN	-	1280	-	HS	1
$1^2 \times 1280$	conv2d 1x1, NBN	-	K	-	-	1

SE menunjukkan apakah terdapat modul Squeeze-and-Excite pada blok tersebut. NL menunjukkan jenis fungsi nonlinier yang digunakan. Pada tabel 1, HS berarti h-swish dan RE berarti ReLU. NBN berarti tanpa batch normalization. S menunjukkan nilai stride.

D. Konfigurasi Pelatihan Model

Untuk memastikan bahwa perbandingan kinerja antar arsitektur bersifat adil (fair comparison), seluruh model dilatih menggunakan konfigurasi pelatihan yang identik pada setiap

eksperimen, sehingga perbedaan performa yang diamati terutama berasal dari karakteristik arsitektur, bukan dari variasi hiperparameter. Pada penelitian ini, ketiga arsitektur (ResNet-18, MobileNetV3-Large, dan EfficientNetB3) digunakan sebagai feature extractor, kemudian lapisan klasifikasi disesuaikan terhadap jumlah kelas pada dataset. Proses optimisasi parameter dilakukan menggunakan optimizer Adam dengan learning rate sebesar 0,001, yang dipilih karena kemampuan adaptifnya dalam mengatur pembaruan bobot dan kecenderungannya menghasilkan konvergensi stabil pada tugas klasifikasi citra. Pelatihan dijalankan dengan batch size 16 selama 25 epoch untuk menyeimbangkan kebutuhan komputasi dan kecukupan iterasi pembelajaran. Fungsi objektif yang digunakan adalah categorical cross-entropy, sesuai untuk skenario klasifikasi multikelas, karena menghukum prediksi yang menyimpang dari distribusi label sebenarnya dan mendorong pemisahan representasi antar kelas. Selama pelatihan, nilai loss dan akurasi pada data pelatihan maupun validasi dipantau secara berkala guna mengevaluasi dinamika konvergensi, sekaligus mengidentifikasi indikasi overfitting apabila terjadi divergensi antara performa pelatihan dan validasi. Dengan konfigurasi ini, evaluasi yang dihasilkan tidak hanya merefleksikan kemampuan prediktif model, tetapi juga memberikan dasar yang konsisten untuk membandingkan stabilitas pelatihan dan generalisasi antar arsitektur.

E. Evaluasi Performa Model

Evaluasi performa model dilakukan secara kuantitatif untuk menilai ketepatan klasifikasi dan pola kesalahan antar kelas. Untuk setiap kelas didefinisikan true positive (TP), true negative (TN), false positive (FP), dan false negative (FN). Berdasarkan komponen tersebut, metrik evaluasi dihitung sebagai berikut:

- 1) Accuracy, untuk mengukur tingkat ketepatan klasifikasi secara keseluruhan.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- 2) Precision, untuk mengukur ketepatan prediksi pada masing-masing kelas.

$$Precision = \frac{TP}{TP + FP}$$

- 3) Recall, untuk mengukur kemampuan model dalam mendeteksi setiap kelas secara benar.

$$Recall = \frac{TP}{TP + FN}$$

- 4) F1-score, sebagai nilai keseimbangan antara precision dan recall.

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

Selain metrik agregat, digunakan confusion matrix untuk menganalisis pola kesalahan antar kelas secara lebih rinci, termasuk kecenderungan misclassification pada pasangan kelas yang memiliki kemiripan visual. Untuk aspek efisiensi, dilakukan pengukuran waktu inferensi pada CPU dan/atau throughput dalam frame per second (FPS) dengan prosedur yang konsisten (ukuran batch dan resolusi input tetap). Dengan demikian, evaluasi tidak hanya menilai performa prediksi, tetapi juga memberikan gambaran trade-off antara akurasi dan biaya komputasi sebagai dasar

pemilihan arsitektur yang paling sesuai untuk penerapan di lingkungan pertanian nyata dengan keterbatasan sumber daya.

III. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil eksperimen serta pembahasan terhadap kinerja tiga arsitektur Convolutional Neural Network (CNN), yaitu ResNet-18, MobileNetV3-Large, dan EfficientNetB3, dalam melakukan klasifikasi penyakit pada tanaman mangga. Evaluasi dilakukan berdasarkan metrik akurasi, precision, recall, F1-score, waktu pelatihan, serta efisiensi inferensi pada CPU.

A. Hasil Pelatihan dan Validasi Model

Proses pelatihan dilakukan selama 25 epoch dengan konfigurasi hiperparameter yang seragam pada seluruh arsitektur, sehingga perbedaan dinamika pembelajaran yang diamati terutama merefleksikan karakteristik masing-masing model. Secara umum, ketiga model menunjukkan indikasi konvergensi yang baik, ditandai oleh tren penurunan training loss dan validation loss serta peningkatan training accuracy dan validation accuracy seiring bertambahnya epoch (Gambar 4-9). Pola ini mengindikasikan bahwa proses optimisasi berjalan efektif dan model mampu mempelajari representasi fitur yang relevan untuk tugas klasifikasi penyakit pada buah mangga.

Pada ResNet-18, kurva loss memperlihatkan penurunan yang konsisten pada fase awal pelatihan, namun validation loss menunjukkan fluktuasi yang lebih nyata dibanding training loss pada beberapa epoch (Gambar 4). Selisih yang sesekali melebar antara kedua kurva tersebut mengindikasikan adanya kecenderungan overfitting ringan, yakni model semakin baik menyesuaikan diri terhadap data latih tetapi peningkatan generalisasi pada data validasi tidak selalu sejalan. Meskipun demikian, kurva akurasi memperlihatkan bahwa validation accuracy tetap meningkat dan stabil pada kisaran tinggi setelah beberapa epoch awal (Gambar 5), yang menunjukkan bahwa overfitting yang terjadi tidak bersifat dominan, tetapi lebih berupa ketidakstabilan lokal pada proses pembaruan bobot.

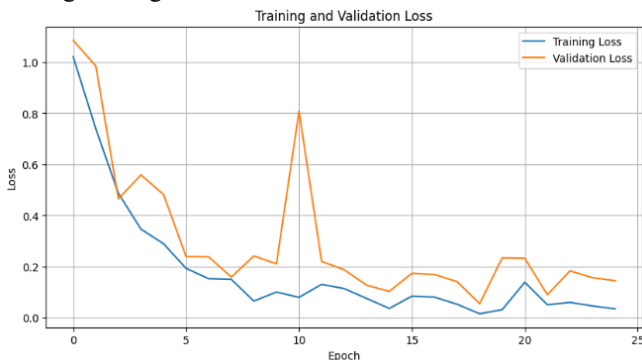
Model MobileNetV3-Large memperlihatkan dinamika pembelajaran yang relatif seimbang antara data pelatihan dan validasi. Kurva training loss dan validation loss cenderung bergerak parallel dengan jarak yang tidak terlalu jauh pada sebagian besar epoch, meskipun terdapat beberapa fluktuasi validation loss pada titik-titik tertentu (Gambar 6). Konsistensi ini diperkuat oleh kurva akurasi yang menunjukkan validation accuracy mengikuti tren training accuracy dengan deviasi yang terbatas (Gambar 7). Secara metodologis, pola tersebut mengindikasikan bahwa model mampu mempertahankan generalisasi yang cukup baik, sekaligus menunjukkan bahwa arsitektur yang lebih ringan dapat mencapai performa validasi tinggi pada data ini.

Sementara itu, EfficientNetB3 menunjukkan karakteristik konvergensi yang paling stabil. Kurva loss memperlihatkan penurunan tajam pada fase awal dan kemudian memasuki fase stabil dengan jarak yang kecil antara training loss dan validation loss pada sebagian besar epoch (Gambar 8). Kurva akurasi juga menunjukkan bahwa validation accuracy meningkat dengan cepat dan berada sangat dekat dengan

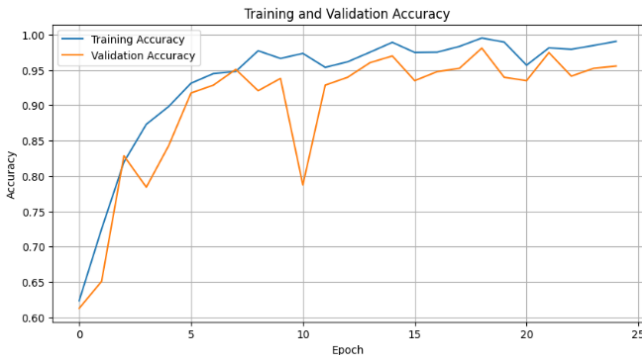
training accuracy hingga akhir pelatihan (Gambar 9). Pola ini secara kuat mengindikasikan kemampuan generalisasi yang lebih baik dan stabilitas pelatihan yang tinggi, karena model tidak hanya mencapai akurasi pelatihan yang tinggi tetapi juga mempertahankan performa validasi secara konsisten tanpa fluktuasi ekstrem.

Secara keseluruhan, analisis kurva pelatihan–validasi menunjukkan bahwa ketiga arsitektur berhasil mempelajari pola klasifikasi dengan baik, namun dengan karakteristik yang berbeda: ResNet-18 cenderung menunjukkan indikasi overfitting ringan melalui fluktuasi validation loss, MobileNetV3-Large memperlihatkan keseimbangan pelatihan–validasi yang kompetitif, dan EfficientNetB3 memberikan stabilitas serta kedekatan kurva pelatihan–validasi yang paling baik, yang merefleksikan generalisasi yang lebih kuat.

Berdasarkan dinamika kurva loss dan accuracy pada data pelatihan serta validasi (Gambar 4–9), dapat disimpulkan bahwa seluruh arsitektur mencapai konvergensi yang memadai, namun menunjukkan karakteristik generalisasi yang berbeda. EfficientNetB3 memperlihatkan kestabilan konvergensi dan kesenjangan pelatihan–validasi yang paling kecil, sedangkan ResNet-18 dan MobileNetV3-Large menunjukkan fluktuasi validasi pada beberapa epoch yang mengindikasikan adanya perbedaan sensitivitas terhadap variasi data. Oleh karena itu, evaluasi selanjutnya difokuskan pada perbandingan kuantitatif melalui metrik akurasi validasi dan waktu pelatihan untuk menilai trade-off antara performa prediktif dan efisiensi komputasi pada masing-masing arsitektur.



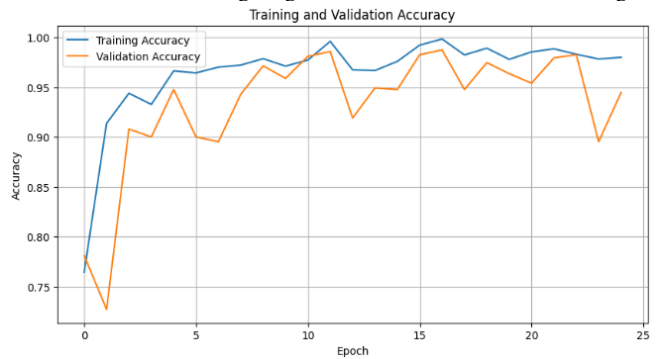
Gambar. 4 Grafik Training dengan Validation Loss Resnet-18



Gambar. 5 Grafik Training dengan Validation Accuracy Resnet-18



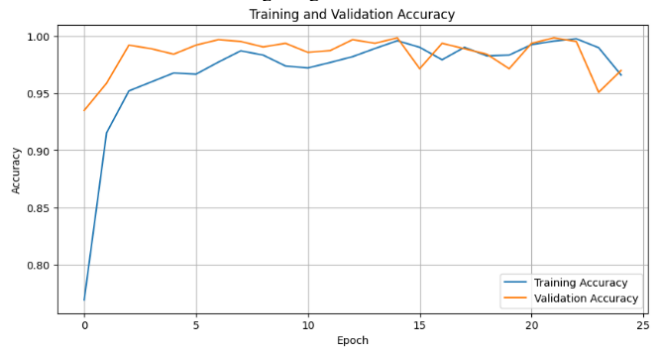
Gambar. 6 Grafik Training dengan Validation Loss MobileNetV3-Large



Gambar. 7 Grafik Training dengan Validation Accuracy MobileNetV3-Large



Gambar. 8 Grafik Training dengan Validation Loss EfficientNetB3



Gambar. 9 Grafik Training dengan Validation Accuracy EfficientNetB3

B. Perbandingan Akurasi dan Waktu Pelatihan

Perbandingan performa kuantitatif antar model dievaluasi melalui indikator train loss, validation loss, training accuracy, validation accuracy, best validation accuracy, serta waktu pelatihan. Kombinasi metrik ini digunakan untuk menilai tidak hanya kemampuan model dalam mempelajari pola pada data latih, tetapi juga daya generalisasi pada data validasi serta biaya komputasi yang

dibutuhkan selama proses pembelajaran. Berdasarkan Tabel 2, ResNet-18 menunjukkan nilai train loss (0,0349) dan validation loss (0,0145) yang paling rendah dibandingkan model lainnya, disertai training accuracy yang tinggi ($\approx 0,9905$). Temuan ini mengindikasikan proses optimisasi yang stabil pada data latih dan validasi. Sementara itu, MobileNetV3-Large memperlihatkan train loss (0,0552) yang masih relatif rendah, namun validation loss tercatat jauh lebih tinggi (0,3162). Pola ini mengisyaratkan adanya ketidaksesuaian antara distribusi prediksi pada data latih dan validasi, yang dapat berkaitan dengan sensitivitas model terhadap variasi data validasi atau keterbatasan representasi fitur pada kondisi tertentu. Di sisi lain, EfficientNetB3 memiliki train loss (0,0918) dan validation loss (0,0932) yang lebih tinggi dibanding ResNet-18, namun tidak menunjukkan lonjakan validation loss setajam MobileNetV3-

Large, sehingga mengindikasikan generalisasi yang lebih konsisten.

Selanjutnya, pada Tabel 3 terlihat bahwa EfficientNetB3 menghasilkan validation accuracy tertinggi (0,9698) serta best validation accuracy terbaik (0,9984), diikuti oleh ResNet-18 (0,9556; best 0,981) dan MobileNetV3-Large (0,9444; best 0,9873). Hasil ini memperkuat indikasi bahwa pendekatan arsitektural EfficientNetB3 mampu mengekstraksi fitur yang lebih diskriminatif untuk membedakan kelas penyakit secara lebih efektif pada data yang tidak dilatih secara langsung. Dengan kata lain, keunggulan EfficientNetB3 lebih tercermin pada performa validasi (generalization performance), bukan semata pada capaian akurasi pelatihan.

Dari aspek efisiensi komputasi, waktu pelatihan EfficientNetB3 (641 detik) merupakan yang paling tinggi, hampir dua kali lipat dibanding ResNet-18 (307 detik) dan MobileNetV3-Large (313 detik). Hal ini menunjukkan adanya trade-off yang jelas antara performa generalisasi dan biaya komputasi: EfficientNetB3 membutuhkan waktu pelatihan lebih besar, namun memberikan performa validasi terbaik. Dengan demikian, pemilihan model untuk implementasi perlu mempertimbangkan kebutuhan aplikasi: EfficientNetB3 lebih sesuai untuk skenario yang memprioritaskan ketepatan dan stabilitas prediksi, sedangkan ResNet-18 dan MobileNetV3-Large lebih kompetitif untuk konteks dengan keterbatasan waktu pelatihan atau sumber daya komputasi.

Tabel 2. Perbandingan Training dan Validasi dari Models (Train Loss, Val. Loss, dan Train Acc)

No.	Model	Train Loss	Val. Loss	Train Acc
1	ResNet-18	0.0349	0.0145	0.9905
2	MobileNet-V3-Large	0.0552	0.3162	0.9799
3	EfficientNetB3	0.0918	0.0932	0.966

Tabel 3. Perbandingan Training dan Validasi dari Models (Val. Acc, Training Time, dan Best Val. Acc)

No.	Model	Val. Acc	Training Time	Best Val. Acc
1	ResNet-18	0.9556	307	0.981

2	MobileNet-V3-Large	0.9444	313	0.9873
3	EfficientNetB3	0.9698	641	0.9984

C. Evaluasi Klasifikasi Berdasarkan Metrik Precision, Recall, dan F1-Score

Evaluasi kinerja model pada penelitian ini diperluas melalui metrik precision, recall, dan F1-score pada setiap kelas, karena akurasi agregat saja belum cukup untuk menggambarkan kualitas prediksi secara rinci pada tingkat kelas. Precision merefleksikan ketepatan prediksi suatu kelas (kemampuan menekan false positive), sedangkan recall merepresentasikan kelengkapan deteksi (kemampuan menekan false negative). F1-score digunakan sebagai ukuran komposit yang menyeimbangkan keduanya, sehingga lebih representatif ketika terdapat trade-off antara precision dan recall pada kelas tertentu.

Berdasarkan Tabel 4, ResNet-18 menunjukkan performa yang tinggi dan relatif stabil pada sebagian besar kelas, dengan F1-score berada pada rentang 0,979–0,992. Namun demikian, terlihat adanya penurunan recall pada kelas Healthy (recall = 0,959) meskipun precision mencapai 1,000, yang mengindikasikan bahwa model cenderung konservatif dalam memprediksi kelas Healthy. Pola ini menunjukkan adanya sebagian sampel Healthy yang terklasifikasi ke kelas penyakit (meningkatnya false negative pada Healthy), sehingga pada konteks aplikasi inspeksi kualitas, kondisi tersebut berpotensi meningkatkan over-rejection (buah sehat dianggap sakit) dan berdampak pada efisiensi sortasi. Pada MobileNetV3-Large, performa tampak cukup kompetitif namun memperlihatkan variasi yang lebih jelas antar kelas. Kelas Stem end rot memperoleh hasil sempurna (precision = 1,000, recall = 1,000, F1-score = 1,000) yang menunjukkan kemampuan diskriminasi yang sangat baik pada kelas tersebut. Akan tetapi, beberapa kelas lain menunjukkan penurunan yang lebih nyata, terutama pada Aspergillus Rot (recall = 0,942, F1-score = 0,963) dan Anthracnose (precision = 0,949, F1-score = 0,974). Pola ini mengindikasikan bahwa pada kelas-kelas dengan kemiripan tekstur atau gejala visual yang lebih halus, model berarsitektur ringan dapat mengalami keterbatasan dalam menangkap fitur diskriminatif secara konsisten, meskipun tetap menawarkan keuntungan efisiensi komputasi.

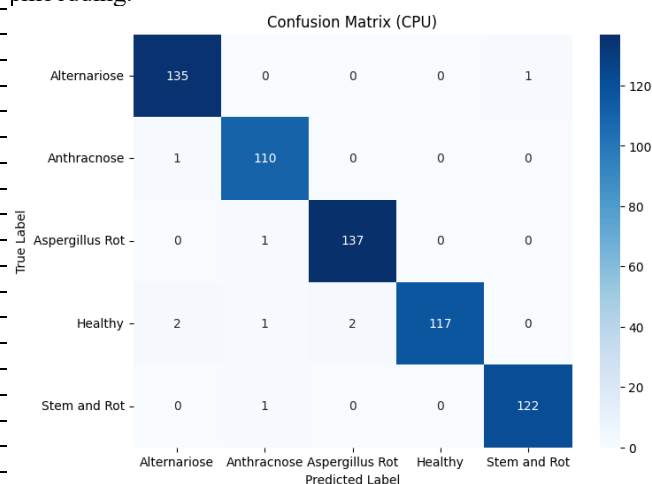
Sementara itu, EfficientNetB3 memberikan performa paling unggul secara keseluruhan, ditunjukkan oleh nilai F1-score yang tinggi dan merata pada seluruh kelas (rentang 0,989–1,000) serta akurasi tertinggi (0,994). EfficientNetB3 bahkan mencapai nilai sempurna pada kelas Anthracnose (precision = 1,000, recall = 1,000, F1-score = 1,000) dan mempertahankan stabilitas yang sangat baik pada kelas lainnya, termasuk Healthy (precision = 0,992, recall = 0,992). Konsistensi ini mengindikasikan bahwa EfficientNetB3 mampu membangun representasi fitur yang lebih diskriminatif dan memiliki generalization yang lebih kuat untuk membedakan karakteristik visual antar kelas penyakit secara lebih akurat. Secara keseluruhan, analisis berbasis precision–recall–F1 memperkuat kesimpulan bahwa EfficientNetB3 merupakan kandidat terbaik untuk skenario yang memprioritaskan ketepatan dan stabilitas prediksi lintas kelas, sedangkan ResNet-18 menunjukkan

performa yang kuat namun perlu perhatian khusus pada sensitivitas kelas Healthy. MobileNetV3-Large tetap relevan sebagai alternatif untuk implementasi pada perangkat terbatas, dengan catatan bahwa kelas-kelas tertentu (misalnya Aspergillus Rot) memerlukan strategi mitigasi untuk menekan kesalahan deteksi dan meningkatkan konsistensi performa.

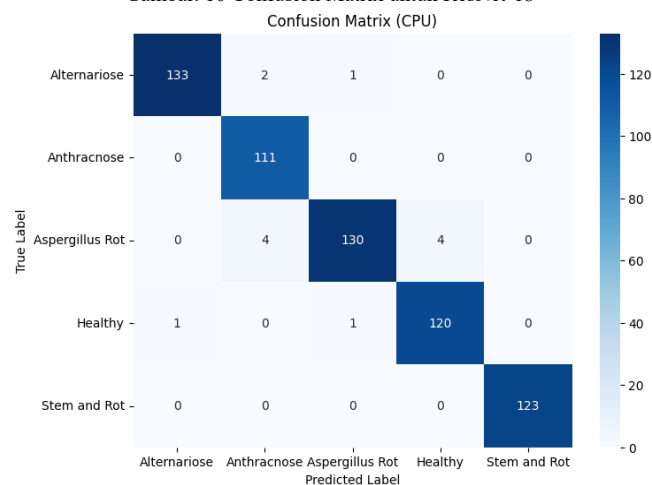
Tabel 4. Hasil Precision, Recall, dan F1-Score untuk Setiap Kelas pada Masing-masing Arsitektur CNN

No.	Model	Class	Precision	Recall	F1-Score
1	ResNet-18	Alternariose	0.978	0.993	0.985
		Anthraco-nose	0.973	0.991	0.982
		Aspergillus Rot	0.986	0.993	0.989
		Healthy	1.000	0.959	0.979
		Stem end rot	0.992	0.992	0.992
		Accuracy	0.986		
2	MobileNetV3	Alternariose	0.993	0.978	0.985
		Anthraco-nose	0.949	1.000	0.974
		Aspergillus Rot	0.985	0.942	0.963
		Healthy	0.968	0.984	0.976
		Stem end rot	1.000	1.000	1.000
		Accuracy	0.979		
3	EfficientNetB3	Alternariose	1.000	0.993	0.996
		Anthraco-nose	1.000	1.000	1.000
		Aspergillus Rot	0.993	0.986	0.989
		Healthy	0.992	0.992	0.992
		Stem end rot	0.984	1.000	0.992
		Accuracy	0.994		

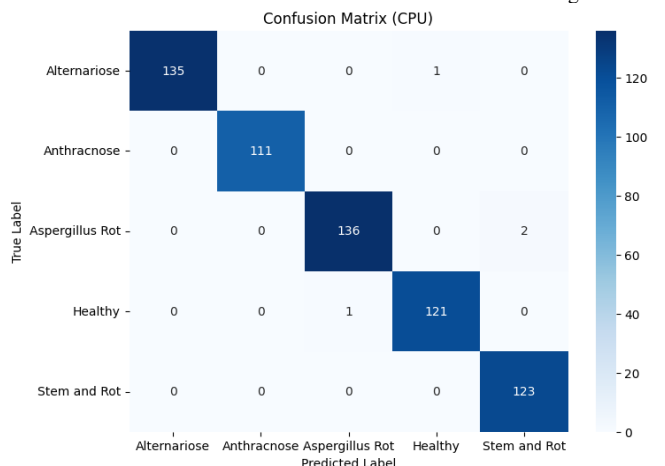
memerlukan pembedaan tekstur halus. Confusion Matrix dari EfficientNetB3 pada Gambar 12 menunjukkan prediksi yang paling terkonsentrasi pada diagonal utama, sehingga kesalahan antar-kelas lebih sedikit dibanding dua model lainnya. Temuan ini konsisten dengan interpretasi bahwa EfficientNetB3 memiliki kemampuan ekstraksi fitur yang lebih baik untuk membedakan kelas-kelas yang tampak mirip (mis. perbedaan pola lesi dan tekstur permukaan). Dengan kata lain, model ini lebih efektif membangun pemisahan fitur (feature separability) pada ruang embedding.



Gambar. 10 Confusion Matrix untuk ResNet-18



Gambar. 11 Confusion Matrix untuk MobileNetV3-Large



Gambar. 12 Confusion Matrix untuk EfficientNetB3

D. Analisis Confusion Matrix

Analisis confusion matrix digunakan untuk melihat pola kesalahan klasifikasi antar-kelas secara rinci, sehingga kita tidak hanya mengetahui nilai akurasi agregat, tetapi juga memahami pasangan kelas mana yang paling sering tertukar. Pada penelitian ini, pola kesalahan terutama muncul pada kelas penyakit yang memiliki kemiripan karakteristik visual, khususnya pasangan Anthracnose dan Aspergillus Rot. Sementara itu, EfficientNetB3 memperlihatkan prediksi yang lebih dominan pada diagonal utama (benar), menandakan jumlah kesalahan yang lebih sedikit dibandingkan ResNet-18 dan MobileNetV3-Large.

Berdasarkan confusion matrix ResNet-18 pada Gambar 10, masih tampak adanya salah klasifikasi antar kelas yang mirip secara visual. Pola ini mengindikasikan bahwa meskipun ResNet-18 stabil sebagai backbone, representasi fitur yang dipelajari masih relatif “overlap” untuk beberapa penyakit yang gejalanya sama-sama berupa bercak gelap/nekrosis pada permukaan buah (mis. Anthracnose vs Aspergillus Rot). Ketika ciri visual dominan berada pada tekstur/warna yang serupa, model cenderung memakai fitur global (kontras/warna) alih-alih fitur lokal yang lebih spesifik (batas lesi, pola sebaran, atau tekstur jamur).

Confusion Matrix MobileNetV3-Large pada Gambar 11 menunjukkan kecenderungan salah klasifikasi pada pasangan kelas yang mirip juga masih muncul. Secara konseptual, MobileNetV3 memang dirancang efisien, sehingga pada kasus-kasus borderline seperti gejala yang sangat mirip, pencahayaan buruk, atau lesi kecil, kapasitas representasinya dapat lebih cepat “kehilangan detail” dibanding backbone yang lebih kuat. Ini biasanya terlihat pada meningkatnya confusion di kelas-kelas yang

Tabel 5. Perbandingan Kecepatan Inferensi (FPS CPU) untuk Setiap Model

No.	Model	Waktu Pelatihan (detik)	Waktu Inference (detik)	Karakteristik Efisiensi
1	ResNet-18	307	31.271	Arsitektur stabil, namun kurang optimal untuk inferensi cepat pada CPU
2	MobileNet-V3- Large	313	55.272	Efisien dan seimbang antara akurasi dan kecepatan inferensi
3	EfficientNetB3	641	13.800	Akurasi tinggi dengan efisiensi komputasi terbaik dalam generalisasi

E. Analisis Efisiensi Komputasi

Analisis efisiensi komputasi dilakukan untuk menilai kelayakan implementasi model pada lingkungan operasional yang berpotensi memiliki keterbatasan sumber daya, sehingga evaluasi tidak hanya berfokus pada ketepatan klasifikasi tetapi juga pada biaya komputasi selama pelatihan dan inferensi. Berdasarkan pengukuran pada Tabel 5, EfficientNetB3 menunjukkan waktu pelatihan paling tinggi (641 detik), mengindikasikan kebutuhan komputasi yang lebih besar pada fase optimisasi parameter dibanding ResNet-18 (307 detik) dan MobileNetV3-Large (313 detik). Namun, pada fase inferensi berbasis CPU, EfficientNetB3 tercatat memiliki nilai inferensi paling rendah (13.8 detik) dibanding ResNet-18 (31.271 detik) dan MobileNetV3-Large (55.272 detik), yang menunjukkan potensi keunggulan dari sisi respons prediksi pada skenario implementasi, dengan catatan bahwa definisi metrik inferensi (mis. latency per citra atau total waktu untuk sejumlah sampel tertentu) perlu dinyatakan secara eksplisit agar interpretasi konsisten.

Secara keseluruhan temuan ini menegaskan adanya trade-off antara biaya pelatihan dan efisiensi inferensi: arsitektur dengan kapasitas representasi lebih tinggi cenderung membutuhkan pelatihan lebih mahal, namun dapat tetap kompetitif (atau unggul) pada inferensi tergantung pada desain scaling dan efisiensi operasionalnya. Implikasinya, pemilihan model untuk deployment sebaiknya mempertimbangkan kebutuhan aplikasi: jika prioritasnya adalah akurasi dan kestabilan prediksi dengan inferensi yang efisien, EfficientNetB3 menjadi kandidat utama; sementara MobileNetV3-Large tetap relevan untuk perangkat terbatas ketika efisien model menjadi prioritas, dengan evaluasi tambahan untuk memastikan konsistensi metrik inferensi dan kondisi uji.

IV. KESIMPULAN

Berdasarkan hasil eksperimen, dapat disimpulkan bahwa pemilihan arsitektur CNN sangat berpengaruh

terhadap performa klasifikasi penyakit tanaman. ResNet-18 menawarkan stabilitas dan struktur yang sederhana, MobileNetV3-Large memberikan keseimbangan antara akurasi dan efisiensi, sementara EfficientNetB3 menunjukkan performa paling optimal baik dari sisi akurasi, generalisasi, maupun efisiensi komputasi.

Temuan ini menunjukkan bahwa akurasi validasi ResNet-18 yaitu 0.9556, akurasi validasi MobileNet-V3-Large yaitu 0.9444, dan EfficientNetB3 yaitu 0.9698 sehingga EfficientNetB3 merupakan arsitektur yang paling sesuai untuk diterapkan pada sistem klasifikasi penyakit tanaman mangga berbasis citra, terutama untuk aplikasi yang membutuhkan akurasi tinggi dan kecepatan inferensi yang baik. Hasil penelitian ini diharapkan dapat menjadi acuan bagi pengembangan sistem deteksi penyakit tanaman berbasis visi komputer di masa mendatang.

REFERENSI

- [1] J. G. Fernandez and J. I. Hormaza, "Pest and Diseases in Mango (Mangifera Indica L)", 2020. Tersedia: https://www.mango.org/wp-content/uploads/2020/08/Mango_Pests_and_Diseases_ENG.pdf
- [2] G. V. Benatar, Y. Nurhayati, and N. Febryani, "Identifikasi Colletotrichum asianum Penyebab Antraknosa Mangga Kultivar Golek di Indramayu", Media Pertanian, vol. 8, no. 1, pp. 1–13, Mei 2023.
- [3] S. Kumar, B. Gupta, N. Bhatia, and K. Anuradha, "An Approach for Diagnosing and Differentiating Mango Fruit Diseases Using Hybrid CNN+SVM Classifier", International Journal of Engineering Trends and Technology, vol. 73, no. 1, pp. 146–154, Januari 2025.
- [4] A. Pramesti, "Efektivitas Biosintesis Nanopartikel Perak (AgNPs) oleh Bacillus amyloliquefaciens EV13 dalam Menghambat Pertumbuhan Jamur Patogen Pascapanen Buah Mangga (Mangifera Indica)", Skripsi Sarjana Universitas Islam Negeri Syarif Hidayatullah Jakarta, 2023.
- [5] I. M. Arti, P. Asnur, R. Kurniasih, and E. P. Ramdan, "Identification, Pathogenesis, and Virulence Test of Fungus Causes Postharvest Disease of Gedong Gincu Mango from Pal Market, Cimanggis, Depok", Jurnal Pembelajaran dan Biologi Nukleus, vol. 8, no. 2, pp. 236–246, Juli 2022.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", nature, 27 Mei 2015. Tersedia: <https://www.nature.com/articles/nature14539>.
- [7] S. U. Khan, A. Alsuhaibani, A. Alabduljabbar, F. Almarshad, Y. N. Altherwy, and T. Akram, "A Review on Automated Plant Disease Detection: Motivation, Limitations, Challenges, and Recent Advancements for Future Research", Journal of King Saud University Computer and Information Sciences, vol. 37, no. 34, pp. 1–36, Mei 2025.
- [8] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification", Computational Intelligence and Neuroscience, vol. 2016, no. 1, Juni 2016.
- [9] A. Howard, M. Sandler, G. Chu, L.C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. A., "Searching for MobileNeV3", arXiv, November 2019. Tersedia: <https://arxiv.org/pdf/1905.02244>.
- [10] S. D. Kusumaningrum, "Ekstraksi Ciri untuk Klasifikasi Tipe Kulit Wajah Menggunakan Citra Foto", Thesis Universitas Islam Indonesia, 2024.
- [11] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks", arXiv, September 2020. Tersedia: <https://arxiv.org/pdf/1905.11946>.
- [12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition", arXiv, Desember 2015. Tersedia: <https://arxiv.org/pdf/1512.03385>.
- [13] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A Comparative Study of Fine-Tuning Deep Learning Models for Plant Disease Identification", Computers and Electronics in Agriculture, vol. 161, pp. 272–279, Juni 2019.
- [14] D. Faye, I. Diop, N. Mbaye, M. Diedhiou, and D. Dione, "SenMangoFruitDSS", Mendeley Data, November 2023. Tersedia: <https://data.mendeley.com/datasets/jvz9p9cbpw/4>.

- [15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proc. CVPR*, 2016.
- [16] A. G. Howard et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv:1704.04861*, 2017.
- [17] A. Howard et al., "Searching for MobileNetV3," *Proc. ICCV*, 2019.
- [18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," *Proc. CVPR*, 2018.
- [19] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *Proc. ICML*, 2019.