

Comparison of EfficientNet-B0 and ResNet-50 for Detecting Diseases in Cocoa Fruit

Ni Putu Maylianti^{1*}, I Gusti Ngurah Lanang Wijayakusuma^{2*}, I Putu Chandra Arta Wiguna^{3*}

* Matematika, Universitas Udayana

ilohmeilin90@gmail.com¹, lanang_wijaya@unud.ac.id², putucandraw@gmail.com³

Article Info

Article history:

Received 2024-11-12

Revised 2024-11-25

Accepted 2025-01-16

Keyword:

Cocoa Diseases,
Classification,
Convolution Neural Network,
EfficientNet-B0,
ResNet 50.

ABSTRACT

Cocoa is a plant that is very susceptible to disease. One of the diseases that often attacks cocoa is black spots on the fruit. Detecting diseases in cocoa fruit is usually done manually by experts, which has limitations in providing information and is very expensive. This study proposes a model for detecting cocoa fruit diseases based on deep learning, namely convolution neural networks (CNN). This study compares CNN architectures, namely EfficientNetB0 and ResNet50 because these two architectures are very popular. EfficientNetB0 is known to be efficient in utilizing model parameters and the ability to achieve high accuracy, while ResNet50 uses Residual block recognition which allows deeper and more accurate model training. The dataset used is 3344 healthy cocoa fruit images, 943 black pod rot images and 103 pod borer images. From this study, the results for the accuracy of both methods are equally superior with an accuracy of 96% while for the precision of the EfficientNetB0 architecture is superior to ResNet50 with a value of 95.7% while for recall and f1-score ResNet50 is superior with a recall value of 94.7% and f1-score 93.3%. Based on the Confusion Matrix, it can be seen that ResNet50 is able to predict pod borer accurately so it can be concluded that in this study ResNet 50 is superior to EfficientNetB0. However, ResNet50 requires more parameters than EfficientNetB0 so ResNet50 requires a very large amount of data and when using a small amount of data EfficientNetB0 is more suitable for use.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Menurut data Badan Pusat Statistik tahun 2020 produksi kakao di Indonesia tercatat mencapai 667,3 ribu ton, produksi tersebut mengalami penurunan setiap tahunnya[1]. Penurunan produksi kakao ini disebabkan oleh penyakit yang menyerang kakao. Untuk menanggulangi penyakit yang menyerang kakao dibutuhkan deteksi penyakit sejak dini. Namun, deteksi penyakit selama ini masih menggunakan pengamatan system pakar dimana metode ini memiliki kelemahan salah satunya adalah keterbatasan akses terhadap ahli, keterbatasan informasi yang diperoleh serta biaya yang relatif tinggi

Artificial Intelligence(AI) banyak diimplementasikan kedalam berbagai sektor seperti peternakan [2], kedokteran

[3], pertanian [4], dan lainnya. Salah satu teknologi yang memberikan pengaruh yang sangat besar atau yang berkembang sangat cepat di berbagai fitur seperti ekstraksi fitur, interaksi fitur, dan representasi data adalah *deep learning*[5]. *Deep learning* adalah *machine learning* yang didasari oleh jaringan saraf tiruan yang memiliki banyak lapisan tersembunyi yang dapat belajar dan merepresentasikan karakteristik data secara otomatis. Implementasi *Convolutional Neural Networks* (CNN) menjadi salah satu keunggulan dari *deep learning* dalam ilmu komputer. Semakin meningkatnya penggunaan algoritma CNN dalam teknik *deep learning* dikarenakan beberapa faktor penting ekstraksi fitur secara manual tidak lagi dibutuhkan serta kemampuan melatih kembali model pada

data perkenalana objek baru sehingga model dapat dibuat di atas model yang sudah ada. Algoritma CNN terbagi atas 3 jenis lapisan (atau blok penyusun): lapisan konvolusional (*convolutional layer*), lapisan pooling (*pooling layer*), dan diikuti lapisan terhubung penuh (*fully connected layer*) [6].

Beberapa penelitian yang dilakukan telah menunjukkan implementasi CNN untuk mengklasifikasi penyakit pada tanaman. Penelitian sebelumnya telah berhasil membuktikan penggunaan *convolution neural network* (CNN) untuk mengklasifikasikan penyakit pada buah jeruk. Data yang digunakan pada penelitian ini diperoleh dari website Kaggle dengan total 1593 gambar buah sakit dan 1466 buah sehat yang menghasilkan akurasi sebesar 93,75[7].

Selanjutnya penelitian yang difokuskan pada penggunaan CNN yaitu VGG16 dan ResNet 50 untuk mendeteksi penyakit tanaman dan gulma secara dini. Pada penelitian ini diperoleh hasil bahwa ResNet50 mencapai akurasi yang lebih tinggi dari VGG16 yaitu 96,2% [8].

Penelitian lainnya dengan mengklasifikasikan buah kakao sehat dan terserang penyakit menggunakan *deep learning*. Data yang digunakan dalam penelitian ini diperoleh dari website Kaggle dengan jumlah data *pest attack, black pod attack*, dan *healthy* masing masing 1000 data yang kemudian akan diklasifikasikan menggunakan model *efficientnetB0, MobileNetV3Small, NasNetMobile* dan diperoleh hasil performa *EfficientNetB0* lebih unggul dalam mengklasifikasi penyakit kakao dengan akurasi sebesar 94% [9].

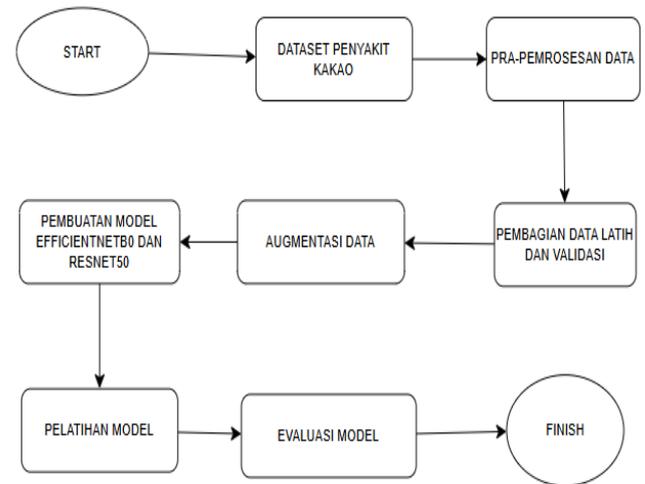
Pada penelitian menggunakan arsitektur ResNet18, yang telah terbukti efektif dalam mengidentifikasi gambar buah kakao, membedakan antara sampel sehat dan yang terinfeksi. Model mencapai akurasi prediksi 83% ketika dilatih dengan total 1596 gambar, menunjukkan kemampuannya dalam mendeteksi penyakit [10]. Penelitian [11] mengevaluasi lima model untuk mengklasifikasikan penyakit kakao yaitu model CNN Kustom, VGG16, *EfficientNetB0*, ResNet50, dan LeNet-5 dengan menggunakan dataset 3344 dataset kakao sehat, 943 black pod rot dan 103 pod borer. Model CNN Kustom mengungguli yang lain, mencapai akurasi, presisi, dan daya ingat 91,79%, bersama dengan skor F1 82,08% .

Berdasarkan beberapa penelitian yang telah dilakukan menunjukkan bahwa CNN terbukti efektif dalam tugas klasifikasi penyakit tanaman dimana *EfficientNetB0* dan *ResNet50* menunjukkan performa yang lebih unggul sehingga peneliti ingin membandingkan *EfficientNetB0* dan *ResNet50* untuk mendeteksi penyakit pada buah kakao. Penelitian ini bertujuan untuk mengevaluasi performa kedua model tersebut menggunakan data citra buah kakao yang terbagi ke dalam citra buah kakao sehat, terserang *black pod rot* dan terserang *pod borer*. Penelitian ini diharapkan memberikan wawasan mengenai model yang akurat dalam deteksi penyakit pada buah kakao sehingga memberikan kontribusi dalam pengembangan teknologi berbasis kecerdasan buatan yang mendukung sektor pertanian serta dapat menjadi Solusi dalam membantu petani kakao mendeteksi penyakit sehingga meminimalkan kerugian akibat serangan penyakit.

II. METODE

A. Tahapan Penelitian

Tahapan yang dapat dilaksanakan pada penelitian ini terdiri atas beberapa langkah, seperti pengumpulan dataset penyakit pada tanaman kakao, pra-pemrosesan data, pembagian dataset, augmentasi data, pembuatan model arsitektur CNN yaitu model *EfficientNetB0* dan model *ResNet50*, kemudian melatih model yang sudah dibangun sebelumnya dan yang terakhir adalah proses mengevaluasi model . Rancangan penelitian tersedia dalam Gambar 1.



Gambar 1. Diagram Alur Penelitian

B. Dataset Penyakit Kakao

Pada penelitian ini dataset yang digunakan diperoleh melalui website Kaggle [12] yang dapat diakses pada : <https://www.kaggle.com/datasets/zaldyjr/cacao-diseases>. Dataset ini terdiri dari gambar buah kakao sehat sebanyak 3344 citra, gambar penyakit *black pod rot* sebanyak 943 citra dan gambar penyakit *pod borer* sebanyak 103 data. Google Drive digunakan untuk menyimpan dataset ini dan akan diimpor ke google colaboratory untuk diproses. Gambar telah diubah ukurannya menjadi 224x224 pixel yang kemudian akan dipelajari oleh *EfficientNetB0* dan *ResNet50*. Namun , dikarenakan jumlah data penyakit *pod borer* jauh lebih sedikit dibandingkan dengan data penyakit *black pod rot* dan buah kakao sehat sehingga penting untuk melakukan augmentasi pada data penyakit *pod borer* sehingga jumlah data menjadi seimbang dan mengurangi terjadinya *overfitting*.

C. Pra-pemrosesan Data

Langkah awal dalam menjalankan penelitian ini yakni melakukan instalasi library TensorFlow dan Keras Untuk membangun model *EfficientNetB0* dan *ResNet50*, melatih model dan mengevaluasi model CNN (Convolutional Neural Network) yang digunakan. ImageDataGenerator untuk melakukan augmentasi dan normalisasi gambar, Shutil atau distutils untuk menyalin dataset dari direktori sumber ke direktori lokal, serta Matplotlib untuk menampilkan citra

hasil augmentasi atau untuk visualisasi data. Selanjutnya, Dataset gambar yang berisi citra penyakit buah kakao disalin dari direktori Google Drive ke direktori lokal untuk persiapan pelatihan model. Direktori yang berisi citra kemudian disimpan dalam variabel `image_path` untuk memudahkan akses ke citra dalam proses berikutnya, setelah mendefinisikan path dataset, gambar akan dimuat dari direktori tersebut menggunakan pustaka seperti TensorFlow atau Keras. Kemudian lakukan normalisasi data dengan membagi nilai piksel citra setiap gambar dengan 255 agar berada dalam kisaran 0 - 1 supaya mempermudah proses pelatihan model.

D. Pembagian Dataset

Dalam proses ini, dataset dibagi menjadi dua bagian, dengan 80% dimanfaatkan untuk pelatihan, sementara 20% data lainnya digunakan untuk validasi. Model akan dilatih menggunakan mayoritas data dan diuji dengan data yang tidak terlihat selama pelatihan. Pembagian dataset akan disajikan pada Tabel 1.

TABEL 1.
DATA PENYAKIT KAKAO

Kelas	Data Latih	Data Validasi	Total
Healthy	2675	669	3344
Black pod rot	754	189	943
Pod borer	82	21	103

E. Augmentasi Data

Untuk meningkatkan kualitas model Machine Learning, khususnya dalam identifikasi jenis penyakit tanaman kakao menggunakan Convolutional Neural Network (CNN), dilakukan beberapa teknik augmentasi data pada kelas *pod borer*. Pertama, gambar dirotasi sebesar 10 derajat untuk mengubah perspektif. Selanjutnya, gambar digeser secara horizontal dan vertikal sebesar 30% sampai 70% dari ukuran sebenarnya, guna menciptakan variasi letak objek dalam citra. Hal ini digunakan untuk menyeimbangkan kelas *pod borer* dengan menambahkan lebih banyak data pelatihan pada kelas tersebut sehingga mengurangi *overfitting*.

F. Convolution Neural Network

Convolutional Neural Network merupakan salah satu jenis algoritma supervised yang cara kerjanya adalah menerima input berupa gambar. CNN sangat cocok untuk pemrosesan dengan input berupa citra. karena pada CNN biasa digunakan untuk mengklasifikasikan citra[13]. CNN adalah suatu metode machine learning yang dikembangkan dari Multi Layer Perceptron (MLP) yang didesain untuk mengolah data 2 dimensi.

G. EfficientNetB0

EfficientNet sendiri salah satu arsitektur transfer learning pada algoritma CNN yang dikembangkan oleh Google pada tahun 2019 lalu [14][15]. EfficientNetB0 adalah arsitektur jaringan saraf yang canggih dalam ranah deep learning. Ini

terdiri dari urutan lapisan yang dirancang khusus untuk menyeimbangkan akurasi dan efisiensi komputasi.

TABEL II
ARSITEKTUR EFFICIENTNETB0

Operator	Input	Kernel	Layers/Repeat
Conv	224 x 224 x 32	3 x 3	1
MBConv1	112 x 112 x 16	3 x 3	1
MBConv6	112 x 112 x 24	3 x 3	2
MBConv6	56 x 56 x 40	5 x 5	2
MBConv6	28 x 28 x 80	3 x 3	3
MBConv6	28 x 28 x 112	5 x 5	3
MBConv6	14 x 14 x 192	5 x 5	4
MBConv6	7 x 7 x 320	3 x 3	1
Conv	7 x 7 x 1280	-	1

H. ResNet50

ResNet, juga dikenal sebagai "*shortcut connections*", memungkinkan model untuk melewati lapisan. ResNet50, sebuah CNN dengan 50 lapisan, telah sebelumnya dilatih pada kumpulan data yang besar.

TABEL III.
ARSITEKTUR RESNET50

Operator	Output	Kernel	Stride
Conv2D	112x112x64	7x7	2
MaxPooling	56x56x64	3x3	2
Conv Block 1	56x56x256	1x1,3x3,1x1	1
Identity Block x2	56x56x256	1x1,3x3,1x1	1
Conv Block 2	28x28x512	1x1,3x3,1x1	2
Identity Block x3	28x28x512	x3	1
Conv Block 3	14x14x1024	1x1,3x3,1x1	2
Identity Block x5	14x14x1024	1x1,3x3,1x1	1
Conv Block 4	7x7x2048	1x1,3x3,1x1	2
Identity Block x2	7x7x2048	1x1,3x3,1x1	1
Average Pooling	1x1x2048	7x7	-
FC	1000	-	-

I. Pelatihan Model

Untuk meningkatkan kinerja model pada input yang tidak terlihat, serangkaian hiperparameter yang dapat dikonfigurasi harus didefinisikan secara eksplisit sebelum melatih jaringan saraf konvolusional. Tabel III menunjukkan hiperparameter yang digunakan dalam eksperimen dan menerapkan metrik akurasi guna mengukur performa selama pelatihan.

TABEL III
KONFIGURASI PELATIHAN

Hyperparameter	Value
Learning rate	0.0001
optimizer	Adam
Max epoch	30
Batch size	32
Loss	sparse_categorical_crossentropy

J. Evaluasi Model

Pada proses evaluasi, dilakukan evaluasi terlebih dahulu pada data validasi untuk memantau bagaimana tingkat perkembangan selama pelatihan. Setelah itu, kemudian akan dilakukan evaluasi pada data uji yang baru. Proses evaluasi dilakukan dengan menghitung berbagai metrik seperti akurasi, presisi, recall, dan F1-skor. Tujuannya adalah untuk menilai kinerja model klasifikasi pada setiap kelas tanaman kakao. Confusion matrix yakni alat visualisasi yang digunakan untuk menunjukkan jumlah prediksi yang benar dan yang salah untuk setiap kelas. Untuk melihat perhitungan matematis yang terkait, silakan merujuk ke persamaan yang berikut:

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

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

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

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

Di mana:

- TP: True Positives (benar positif)
- TN: True Negatives (benar negatif)
- FP: False Positives (positif salah)
- FN: False Negatives (negatif salah)

Untuk memberikan gambaran bagaimana model bekerja, akan ditampilkan juga grafik akurasi dan loss masing masing model selama melakukan pelatihan. Selain itu juga akan ditunjukkan hasil akurasi, recall, precision dan f1-score yang diperoleh dari kedua model selama pelatihan.

III. HASIL DAN PEMBAHASAN

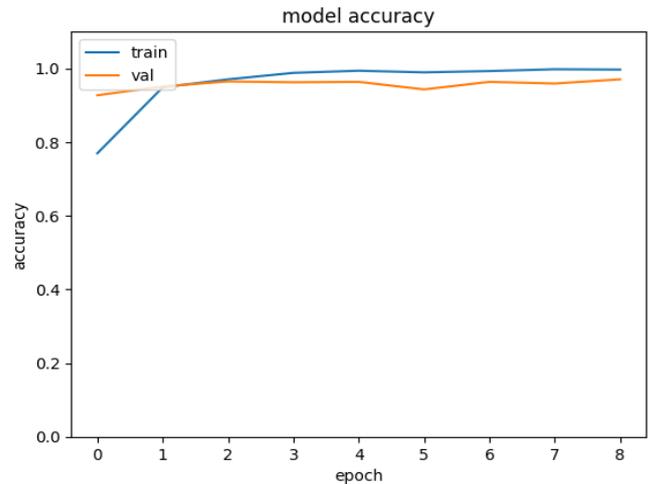
A. Hasil dan Akurasi

Hasil penelitian yang telah dilakukan peneliti akan ditunjukkan pada bagian ini. Hasil akurasi arsitektur EfficientNetB0 dan ResNet 50 akan ditampilkan pada Tabel IV dengan pembagian data 80:20 dan jumlah epoch sebanyak 30 epoch dilengkapi dengan total parameter pada masing masing arsitektur CNN.

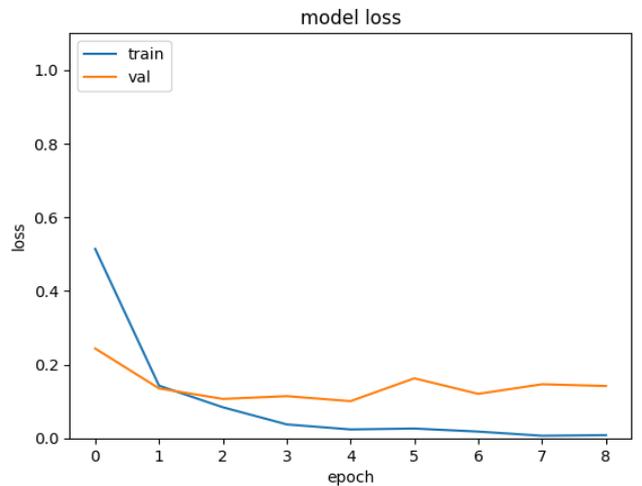
TABEL IV
HASIL AKURASI MODEL CNN

Arsitektur CNN	Total Parameter	Akurasi
EfficientNetB0	4,053,414	96%
ResNet50	23,593,859	96%

Tabel IV dapat disimpulkan bahwa kedua arsitektur CNN memperoleh akurasi tertinggi sebesar 96% .

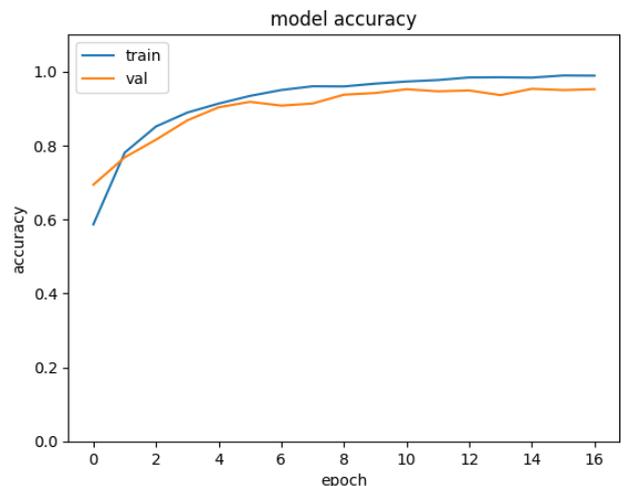


Gambar 2. Kurva akurasi EfficientNetB0

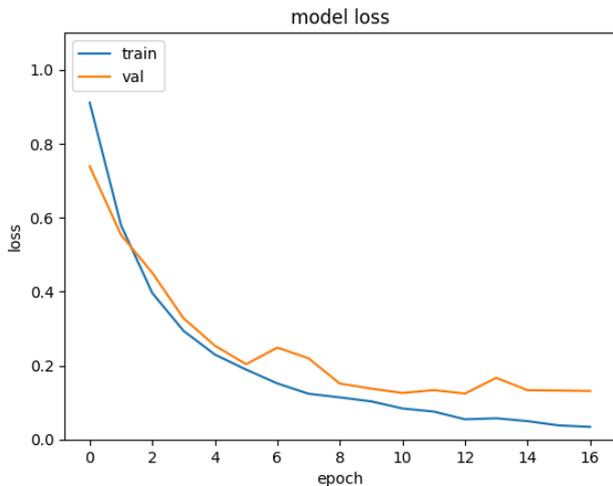


Gambar 3. Kurva Model Loss EfficientNetB0

Untuk kurva akurasi dan loss arsitektur ResNet50 ditampilkan pada Gambar 4 dan pada Gambar 5.



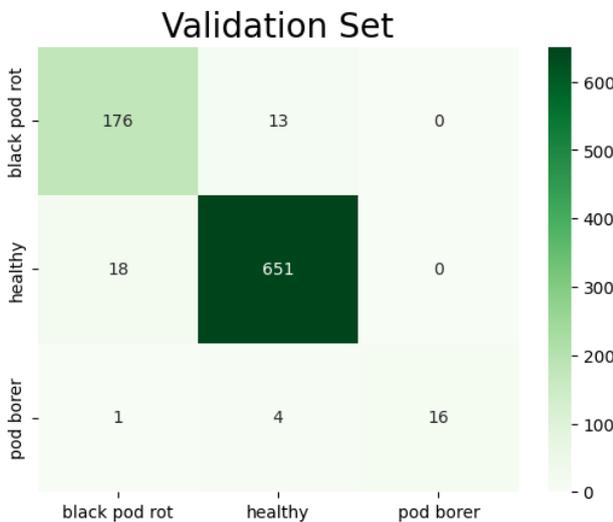
Gambar 4. Kurva akurasi ResNet50



Gambar 5. Kurva Model Loss ResNet50

B. Confusion Matrix

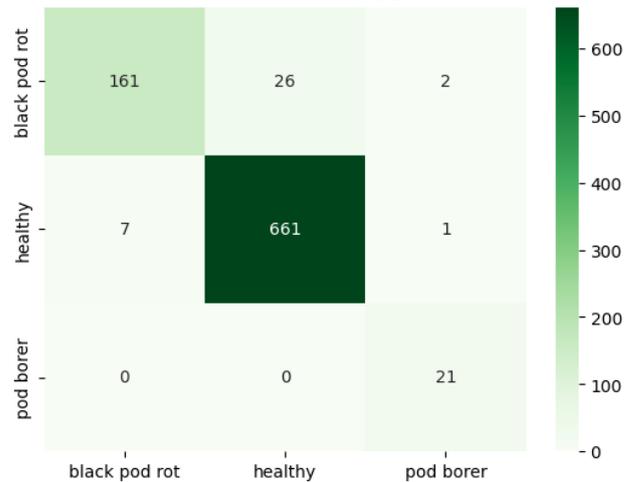
Persebaran dari data uji kedalam bentuk tabel matriks disebut *Confusion Matrix*. Dalam penelitian ini, *Confusion Matrix* akan ditampilkan dalam 3 kelas dengan label hasil prediksi akan ditampilkan pada sumbu X dan label data asli ditampilkan pada sumbu Y. Sebuah gambar dikatakan dapat diprediksi dengan tepat apabila sumbu X dan Y bertemu pada tempat yang sama. Semakin tinggi nilai dari hasil prediksi akan ditandai dengan warna hijau tua dan apabila semakin rendah hasil prediksi akan ditandai dengan warna hijau muda hingga putih.



Gambar 6. Confusion matrix EfficientNetB0

Gambar 6 diatas dapat disimpulkan bahwa EfficientNetB0 memiliki prediksi kurang tepat dalam beberapa kelas seperti dataset yang seharusnya merupakan *black pod rot* diprediksi sebagai *healthy* sebanyak 13 dan dataset yang merupakan *healthy* diprediksi sebagai *black pod rot* sebanyak 18. serta dataset yang merupakan *pod borer* diprediksi sebagai *healthy* sebanyak 4 dan *black pod rot* sebanyak 1.

Validation Set



Gambar 7. Confusion matrix ResNet50

Gambar 7 diatas dapat disimpulkan bahwa ResNet50 memiliki prediksi kurang tepat dalam beberapa kelas seperti dataset yang seharusnya merupakan *black pod rot* diprediksi sebagai *healthy* sebanyak 26 dan diprediksi sebagai *pod borer* sebanyak 2 dan dataset yang merupakan *healthy* diprediksi sebagai *black pod rot* sebanyak 7 dan 1 gambar diprediksi sebagai *pod borer*.

C. Precision, Recall, dan F1-Score

Pada bagian ini akan dihitung parameter *Precision*, *Recall*, dan *F1-Score* dari tabel *Confusion Matrix* berdasarkan kelas masing-masing.

	precision	recall	f1-score	support
black pod rot	0.90	0.93	0.92	189
healthy	0.97	0.97	0.97	669
pod borer pest	1.00	0.76	0.86	21
accuracy			0.96	879
macro avg	0.96	0.89	0.92	879
weighted avg	0.96	0.96	0.96	879

Gambar 8. Classification report EfficientNetB0

Berdasarkan Gambar 8 dapat disimpulkan nilai *precision* tertinggi diperoleh oleh *pod borer* sedangkan nilai *precision* terendah diperoleh oleh *black pod rot* yaitu 0,90. Parameter *Recall* tertinggi terjadi pada kelas *healthy* dengan nilai 0,97 sedangkan terendah pada kelas *pod borer* yaitu 0,76. Sedangkan untuk nilai *f1-score* tertinggi terjadi pada kelas *healthy* dengan nilai 0,97 dan terendah pada kelas *pod borer* dengan nilai 0,86. Seperti yang dapat dilihat pada confusion matrix pada Gambar 6, terdapat 169 kelas *black pod rot*, dimana 176 data telah diprediksi dengan tepat sebagai kelas *black pod rot*. Namun 13 gambar diprediksi dengan tidak akurat, yaitu kelas *healthy*. Pada Gambar 6 juga terdapat 669 gambar kelas *healthy* dimana 651 data dapat diprediksi secara tepat akan tetapi, ada 18 data yang diprediksi tidak akurat

yaitu pada kelas *black pod rot*. Kemudian untuk 21 data kelas *pod borer* hanya 16 gambar yang prediksinya tepat dan 5 gambar tidak akurat dalam prediksi yaitu 4 pada kelas *healthy* dan 1 pada kelas *black pod rot*.

	precision	recall	f1-score	support
black pod rot	0.96	0.85	0.90	189
healthy	0.96	0.99	0.97	669
pod borer pest	0.88	1.00	0.93	21
accuracy			0.96	879
macro avg	0.93	0.95	0.94	879
weighted avg	0.96	0.96	0.96	879

Gambar 9. Classification report ResNet50

Berdasarkan Gambar 9 dapat disimpulkan nilai *precision* tertinggi diperoleh oleh *black pod rot* dan *healthy* sedangkan nilai *precision* terendah diperoleh oleh *pod borer* yaitu 0,88. Parameter *Recall* tertinggi terjadi pada kelas *pod borer* dengan nilai 1,00 sedangkan terendah pada kelas *black pod rot* yaitu 0,85. Sedangkan untuk nilai *f1-score* tertinggi terjadi pada kelas *healthy* dengan nilai 0,97 dan terendah pada kelas *black pod rot* dengan nilai 0,90. dapat dilihat confusion matrix pada Gambar 7, ada 189 data citra *black pod rot*, yang mana 161 data dapat diprediksi dengan tepat sebagai *black pod rot*. Akan tetapi, ada 28 data yang tidak akurat dalam prediksi, yaitu 26 pada kelas *healthy* dan 2 pada kelas *pod borer*. Begitupun 669 data pada kelas *healthy* banyak nya data yang dapat diprediksi secara akurat sebanyak 661 data sedangkan 8 data lainnya diprediksi secara tidak akurat yaitu 7 data pada kelas *black pod rot* dan 1 data pada kelas *healthy*.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, implementasi CNN dengan arsitektur EfficientNetB0 dan ResNet50 dalam mengklasifikasi citra penyakit buah kakao dengan 3 kelas yaitu *healthy*, *black pod rot* dan *pod borer* bekerja dengan sangat baik. Kedua arsitektur CNN ini dengan pembagian dataset 80% untuk data latih dan 20% untuk data validasi sama sama menunjukkan hasil yang terbaik dengan akurasi 96%. Untuk *precision* arsitektur EfficientNetB0 lebih unggul daripada ResNet50 dengan nilai 95,7% sedangkan untuk *recall* dan *f1-score* ResNet50 lah yang lebih unggul dengan nilai *recall* 94,7% dan *f1-score* 93,3%. Berdasarkan *Confusion Matrix* terlihat bahwa ResNet50 mampu memprediksi *pod borer* secara tepat sehingga dapat disimpulkan bahwa didalam penelitian ini ResNet 50 lebih unggul dibandingkan dengan EfficientNetB0. Akan tetapi ResNet50 membutuhkan parameter yang lebih banyak dari EfficientNetB0 sehingga ResNet50 memerlukan jumlah data yang sangat banyak dan ketika menggunakan jumlah

data yang sedikit EfficientNetB0 lebih cocok untuk digunakan.

Untuk mengembangkan penelitian selanjutnya, peneliti menyarankan untuk mencoba menggunakan pembagian data yang bervariasi untuk melihat bagaimana performa kedua arsitektur CNN tersebut. penelitian ini juga dapat dikembangkan dengan menambahkan lebih banyak data citra penyakit kakao yang dapat dikumpulkan melalui website maupun dapat diambil secara langsung.

DAFTAR PUSTAKA

- [1] Asiva Noor Rachmayani, "statistik-kakao-indonesia-2020," p. 6, 2015.
- [2] P. Tassinari *et al.*, "A computer vision approach based on deep learning for the detection of dairy cows in free stall barn," *Comput. Electron. Agric.*, vol. 182, no. May 2020, p. 106030, 2021, doi: 10.1016/j.compag.2021.106030.
- [3] M. Tsuneki, "Deep learning models in medical image analysis," *J. Oral Biosci.*, vol. 64, no. 3, pp. 312–320, 2022, doi: 10.1016/j.job.2022.03.003.
- [4] Z. Unal, "Smart Farming Becomes even Smarter with Deep Learning - A Bibliographical Analysis," *IEEE Access*, vol. 8, pp. 105587–105609, 2020, doi: 10.1109/ACCESS.2020.3000175.
- [5] L. Chen, *Deep Learning and Practice with MindSpore*. 2021. doi: 10.1007/978-981-16-2233-5.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436–444, May 2015, doi: 10.1038/nature14539.
- [7] P. Dhiman, V. Kukreja, and A. Kaur, "Citrus Fruits Classification and Evaluation using Deep Convolution Neural Networks: An Input Layer Resizing Approach," *2021 9th Int. Conf. Reliab. Infocom Technol. Optim. (Trends Futur. Dir. ICRITO 2021)*, pp. 1–4, 2021, doi: 10.1109/ICRITO51393.2021.9596357.
- [8] B. Gupta, S. Bomble, O. Gaikar, S. Chalekar, S. R. Vispute, and K. Rajeswari, "Convolutional Neural Networks for Detection of Crop Diseases and Weed," *2022 6th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2022*, pp. 1–5, 2022, doi: 10.1109/ICCUBEA54992.2022.10010772.
- [9] R. A. Godmalin, C. J. Aliac, and L. Feliscuzo, "Classification of Cacao Pod if Healthy or Attack by Pest or Black Pod Disease Using Deep Learning Algorithm," *4th IEEE Int. Conf. Artif. Intell. Eng. Technol. IICAIET 2022*, pp. 1–5, 2022, doi: 10.1109/IICAIET55139.2022.9936817.
- [10] R. Yanac Montesino, J. A. Rosales-Huamani, and J. L. Castillo-Sequera, "Detection of phytophthora palmivora in cocoa fruit with deep learning," *Iber. Conf. Inf. Syst. Technol. Cist.*, no. June, pp. 23–26, 2021, doi: 10.23919/CISTI52073.2021.9476279.
- [11] K. Sing Soh, E. Gubin Moug, K. John Julius Danker, J. A. Dargham, and A. Farzammia, "Cocoa Diseases Classification using Deep Learning Algorithm," *ITM Web Conf.*, vol. 63, p. 01014, 2024, doi: 10.1051/itmconf/20246301014.
- [12] "https://www.kaggle.com/datasets/zaldyjr/cacao-diseases."
- [13] H. Nisa', C., Puspaningrum, E. Y., & Maulana, "Penerapan Metode Convolutional Neural Network untuk Klasifikasi Penyakit Daun Apel pada Imbalanced Data," in *Prosiding Seminar Nasional Informatika Bela Negara*, 2020.
- [14] S. Armandito and T. B. Sasongko, "Comparison of EfficientNetB7 and MobileNetV2 in Herbal Plant Species Classification Using Convolutional Neural Networks," *J. Appl. Informatics Comput.*, vol. 8, no. 1, pp. 176–185, 2024, doi: 10.30871/jaic.v8i1.7927.
- [15] S. Armandito and T. Sasongko, "Comparison of EfficientNetB7 and MobileNetV2 in Herbal Plant Species Classification Using Convolutional Neural Networks", *JAIC*, vol. 8, no. 1, pp. 176-185, Jul. 2024.