



SISTEM KLASIFIKASI BERBASIS *ANDROID* UNTUK PENYAKIT BUAH KAKAO MENGGUNAKAN *CNN NASNET-MOBILE*

Gregorius Albertus Setu Gado¹, Putri Noraisya Primandari²

^{1,2} Teknik Informatika, Universitas 17 Agustus 1945 Surabaya
 Surabaya, Jawa Timur, Indonesia 60118
gregoriusalbertus23@gmail.com, puterinoraisya@untag-sby.ac.id

Abstract

Cocoa is an important commodity in Indonesia that is susceptible to pathogen-induced diseases. These diseases reduce fruit quality and are difficult to recognise at an early stage. This research uses a Convolutional Neural Network (CNN) method with a transfer learning approach and NASNet-Mobile architecture to facilitate the classification of cocoa fruit diseases. The data consisted of 2000 images of diseased and non-diseased cocoa pods divided into four classes, namely Cocoa Pod Rot (Black Pod), Fruit Sucking Ladybugs (Helopeltis sp), Fruit Borer (Pod Borer) and Normal. Training was conducted for 25 epochs using Google Colab. The best model produced 99.11% training accuracy, 96.14% validation, and 94.88% testing. The model was implemented into an Android device and field tested with 93.33% accuracy, 98.5% recall, 57.1% precision, and 71.6% F1-score. This system is effective in helping early detection of cocoa pod disease in a practical, efficient manner without reducing the accuracy value.

Keywords: Classification, CNN, NASNet-Mobile, Plant Disease, Transfer Learning

Abstrak

Kakao merupakan komoditas penting di Indonesia yang rentan terhadap penyakit yang diakibatkan patogen. Penyakit ini menurunkan kualitas buah dan sulit dikenali pada tahap awal. Penelitian ini menggunakan metode *Convolutional Neural Network* (CNN) dengan pendekatan *transfer learning* dan arsitektur *NASNet-Mobile* untuk memudahkan klasifikasi penyakit buah kakao. Data terdiri dari 2000 citra buah kakao berpenyakit dan tidak berpenyakit yang terbagi dalam empat kelas yaitu Busuk Buah Kakao (*Black Pod*), Kepik Penghisap Buah (*Helopeltis sp*), dan Penggerek Buah (*Pod Borer*) dan Normal. Pelatihan dilakukan selama 25 *epoch* menggunakan *Google Colab*. Model terbaik menghasilkan akurasi pelatihan 99,11%, validasi 96,14%, dan pengujian 94,88%. Model diimplementasikan ke dalam perangkat *android* dan diuji di lapangan dengan akurasi 93,33%, *recall* 98,5%, presisi 57,1%, dan *f1-score* 71,6%. Sistem ini efektif membantu deteksi dini penyakit buah kakao secara praktis, efisien tanpa mengurangi nilai akurasi.

Kata kunci: CNN, Klasifikasi, *NASNet-Mobile*, Penyakit Tanaman, *Transfer Learning*

1. PENDAHULUAN

Tanaman kakao dengan nama latin *Theobroma cacao*, adalah pohon kecil yang biasanya tumbuh antara 4 hingga 8 meter tingginya. Daun-daunnya berbentuk oval, berwarna hijau tua, dan berukuran sekitar 15-30cm panjangnya. Buah kakao memiliki bentuk oval atau lonjong, berwarna kuning, oranye, atau merah ketika masak. Buah dari tanaman kakao menjadi komoditas ekspor yang cukup besar bagi Indonesia. Komoditi kakao di Indonesia merupakan salah satu komoditi yang memberikan kesempatan usaha dan kerja, sebagai sumber kehidupan bagi jutaan penduduk yang bergerak aktif dalam kegiatan produksi, pengolahan hasil, pemasaran[1]. Tanaman kakao yang normal menghasilkan buah yang berkualitas bagus, akan tetapi tanaman kakao

rentan terkena penyakit yang diakibatkan jamur, hama ataupun bakteri. Petani umumnya yang berada di daerah terpencil sulit mengidentifikasi dan mengklasifikasikan penyakit dengan tepat karena minimnya pengetahuan akan jenis-jenis penyakit yang menyerang tanaman kakao, sehingga penanganan pun menjadi kurang tepat. Beberapa penyakit yang sering terdapat pada tanaman kakao yaitu busuk buah, kanker batang, kepik penghisap buah, antraknosa, upas dan penyakit-penyakit lain seperti penyakit-penyakit akar, belang daun, tunas bengkak, penyakit sapu, busuk buah *monilia*, dan *vascular streak dieback* [2]. Penelitian kali ini berfokus untuk mengembangkan sebuah sistem berbasis klasifikasi citra untuk penyakit Busuk Buah (*Black pod*), Kepik Penghisap

Buah (*Helopeltis sp.*), dan Penggerek Buah (*Pod Borer*), dan satu kelas yang akan mewakili kakao normal bebas penyakit. Klasifikasi berbasis citra pada makhluk hidup didasarkan pada ciri visual pada objek yang diklasifikasikan, seperti pada studi kesegaran ikan bandeng menggunakan *Convolutional Neural Network* (CNN) [3] dimana perubahan visual yang terjadi seiring waktu seperti perubahan warna mata dan insang dapat diklasifikasikan, sehingga perubahan visual pada kulit buah kakao yang terinfeksi penyakit akan sangat efektif untuk dikenali perbedaan fitur visualnya secara otomatis menggunakan metode visi komputer.

Pengklasifikasian melibatkan pengelompokan organisme berdasarkan ciri-ciri khusus, CNN memungkinkan model mengekstraksi fitur penting seperti bentuk dan corak bulu melalui lapisan konvolusi, sebelum akhirnya memetakan ciri-ciri tersebut ke dalam kelas yang sesuai pada lapisan *fully connected* [4]. Temuan ini menunjukkan bahwa CNN merupakan pendekatan yang efektif dalam mendukung proses klasifikasi berdasarkan karakteristik visualnya. Penelitian ini bertujuan untuk membantu petani kakao dalam mengklasifikasikan 3 penyakit dan buah normal yang sudah disebutkan sebelumnya. Sistem diharapkan mampu menyelesaikan masalah klasifikasi penyakit yang dilakukan secara manual dimana klasifikasi kerap baru dapat dilakukan saat penyakit sudah mencapai stadium tertentu. *Supervised learning* berbasis CNN [5] mampu menangkap fitur visual kompleks dari organisme biologis dan memetakannya secara akurat ke dalam kelas target, menjadikannya sangat relevan untuk sistem klasifikasi berbasis citra.

Penelitian yang dilakukan oleh [6] menunjukkan bahwa penggunaan *Convolutional Neural Network* (CNN) mampu mengklasifikasikan penyakit pada tanaman padi hanya berdasarkan citra daun, dengan memanfaatkan *dataset* kecil dari *Kaggle* dan skenario *epoch* dimana secara konseptual dan teknis hampir sama dengan penelitian ini, menunjukkan potensi besar untuk digunakan pada deteksi penyakit secara visual di kalangan petani. CNN adalah salah satu pendekatan yang menjanjikan, karena kemampuannya dalam mengklasifikasikan dan mendeteksi objek dalam citra dengan akurasi yang tinggi menggunakan kemajuan teknologi komputasi, seperti penggunaan *deep learning* [7] dan pengolahan citra digital. Struktur CNN yang terdiri dari *feature learning layer* dan *fully-connected layer* berperan dalam mengekstraksi pola visual dari gambar mentah. Lapisan awal bertugas menangkap pola dasar seperti tepi dan tekstur, *feature learning layer* [8] bertugas menangkap pola penting dari citra mentah menggunakan filter dan *max pooling*, sedangkan *fully connected layer* memetakan hasil ekstraksi tersebut ke dalam label kerusakan spesifik. Hal ini memperkuat alasan pemanfaatan CNN dalam tugas deteksi visual berbasis citra, termasuk dalam penelitian ini yang menggunakan *transfer learning* dari arsitektur *NasNet-Mobile* untuk klasifikasi penyakit tanaman berbasis kerusakan pada permukaan citra. *Transfer Learning*

memperkuat efektivitas penggunaan CNN. Klasifikasi citra daun tanaman dengan menggunakan arsitektur *VGG19* [9] yang dilatih ulang melalui *transfer learning*, serta ditambahkan teknik *dropout* dapat mengurangi *overfitting* yang meningkatkan akurasi. Hal ini juga diterapkan dalam penelitian oleh [10], yang memanfaatkan *transfer learning* dengan arsitektur *MobileNetV2* dan *EfficientNet-B0* untuk mengenali ekspresi wajah dalam rangka mendeteksi potensi perkelahian. Model *pretrained* yang digunakan ditransfer dari domain umum dan di-*fine-tune* untuk tugas klasifikasi ekspresi. Pendekatan ini memungkinkan pemanfaatan fitur-fitur yang telah dipelajari sebelumnya secara lebih cepat dan efisien, terutama ketika *dataset* yang digunakan terbatas.

Arsitektur CNN yang digunakan pada penelitian ini ialah arsitektur *NASNet-Mobile*. Arsitektur CNN berbasis *transfer learning* sudah pernah diterapkan dalam penelitian pada penyakit buah kakao seperti pada penelitian yang dilakukan oleh [11], penelitian dilakukan menggunakan beberapa model sekaligus seperti *CenterNet ResNet50 V2*, *EfficientDet D0*, *SSD MobileNet V2*, dan *SSD ResNet50 V1 FPN* dan menggunakan aplikasi seluler untuk mengambil gambar buah kakao secara *realtime* sehingga dapat dijadikan referensi untuk melakukan penelitian dengan topik yang serupa namun menggunakan arsitektur CNN yang berbeda yaitu *NASNet-Mobile*. Pada penelitian [12] yang membahas penggunaan *dataset* kecil secara efektif dalam klasifikasi cacat permukaan baja menggunakan metode CNN *NASNet-Mobile* dan penggunaan *dataset* kecil, diketahui bahwa peneliti hanya menggunakan 300 gambar per kelas (total 1800 gambar untuk enam jenis cacat) dari *dataset* *Severstal*. Meskipun jumlah data jauh lebih kecil dibandingkan *dataset* lain, model yang digunakan mampu mencapai akurasi pelatihan yang baik menunjukkan bahwa pendekatan ini efisien dalam situasi dengan data terbatas. Hal ini menjadi solusi penting bagi peneliti yang tidak memiliki akses terhadap data besar.

Penelitian oleh [13] menunjukkan bahwa *NASNet-Mobile* merupakan salah satu arsitektur CNN yang sangat cocok untuk implementasi di lingkungan pertanian dengan keterbatasan komputasi. Dalam studi tersebut, *NASNet-Mobile* digunakan untuk klasifikasi penyakit daun tomat dengan memanfaatkan *framework TensorFlow* dan *Keras*, yang mendukung proses *transfer learning*, optimasi model, dan konversi ke format *deployment* seperti *TensorFlow Lite*. Arsitektur ini menunjukkan akurasi tinggi dengan ukuran model yang ringkas dan jumlah parameter yang rendah. Penelitian ini membuktikan bahwa arsitektur ringan seperti *NASNet-Mobile* dapat menjaga keseimbangan antara efisiensi dan akurasi, serta sangat ideal untuk diterapkan langsung di lapangan oleh petani melalui perangkat *Android* atau *edge computing* tanpa memerlukan sumber daya besar.

NASNet-Mobile sendiri telah terbukti efektif tidak hanya dalam klasifikasi penyakit tanaman, tetapi juga dalam pengenalan makhluk hidup non-tumbuhan seperti yang

ditunjukkan pada penelitian [14] dengan memanfaatkan bobot *pretrained* dari *ImageNet* (*transfer learning*), model mampu mencapai akurasi tinggi menggunakan optimizer *RMSprop*, menjadikannya yang tertinggi di antara lima *optimizer* yang diuji. Model CNN ini dilengkapi dengan proses *pre-processing* citra yang meliputi augmentasi dan *rescaling*, serta evaluasi berbasis *confusion matrix* dan *classification report*. Temuan ini menegaskan bahwa *NASNetMobile* bersifat fleksibel dan efisien untuk diaplikasikan dalam klasifikasi berbagai objek biologis, baik dalam ranah pertanian maupun peternakan. Akhir dari sistem akan di-*deploy* pada *android*. Proses *deployment* dilakukan di *Android* karena *Android* merupakan salah satu *platform* yang penggunaanya terus meningkat, hal ini sejalan dengan penelitian yang dilakukan oleh [15] yang menggunakan pelatihan pada *Google Collab* dan hasil klasifikasi di perangkat *Android*. Kombinasi *transfer learning* dan juga perangkat *android* dapat menunjang efisiensi.

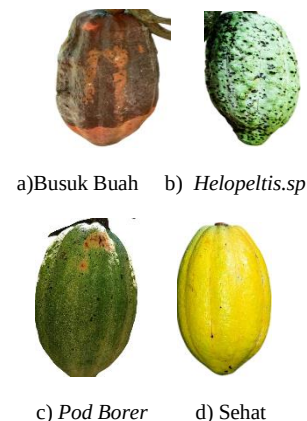
2. METODE PENELITIAN

2.1 Metode pengumpulan data, instrumen penelitian, dan metode pengujian

Data yang digunakan dalam penelitian ini diperoleh melalui metode pengumpulan data primer dan sekunder. Data primer diambil melalui pengambilan citra-citra buah kakao secara langsung di perkebunan kakao, sedangkan data sekunder diambil melalui platform penyedia *dataset* seperti *Roboflow* dan juga *Kaggle*. *Dataset* yang diperoleh terdiri dari beberapa kategori citra, antara lain buah kakao sehat, serta yang mengalami gangguan penyakit seperti *Black Pod*, *Helopeltis sp*, dan *Pod Borer*. Instrumen penelitian yang digunakan berupa perangkat keras (*hardware*) seperti laptop dengan spesifikasi yang mendukung komputasi intensif. Perangkat lunak (*software*) berupa bahasa pemrograman *Python* yang dijalankan di *Google Colaboratory* (*Google Colab*), sebuah platform komputasi berbasis *cloud* yang memungkinkan pelatihan model machine learning secara efisien. Selain itu, digunakan pustaka pendukung seperti *TensorFlow/Keras*, *NumPy*, dan *Matplotlib* untuk proses pelatihan dan visualisasi data.

Tahapan pengujian, antara lain membagi *dataset* ke dalam data pelatihan, validasi, dan pengujian (*train*, *validation*, dan *test*) untuk menghindari *overfitting* dan mengevaluasi performa model secara objektif. Evaluasi performa dilakukan menggunakan metrik evaluasi seperti akurasi, presisi, *recall*, dan *f1-score* untuk mengukur kemampuan model dalam mengklasifikasikan gambar sesuai label penyakitnya. *Dataset* berjumlah 2.000 citra buah kakao, terbagi secara seimbang ke dalam empat kelas, masing-masing berisi 500 gambar. Kelas-kelas tersebut mencakup *Black Pod*, *Helopeltis sp*, *Pod Borer*, dan Normal. Dengan distribusi data yang merata, proses pelatihan model menjadi lebih optimal, memungkinkan CNN dengan metode *transfer learning* menggunakan *NASNet-Mobile* untuk mengenali pola penyakit secara lebih akurat. *Dataset* ini telah melalui

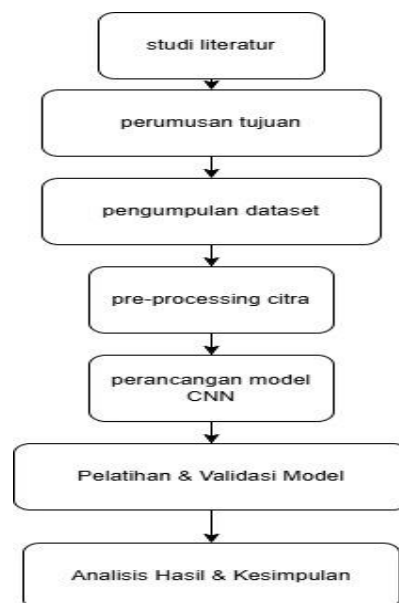
tahap *preprocessing* yaitu menghapus latar belakang untuk memastikan kualitas gambar yang baik sebelum digunakan dalam pelatihan model, sehingga meningkatkan efektivitas seperti pada Gambar 1



Gambar 1. Citra yang dipre-processing

2.2 Tahapan penelitian

Setiap tahapan menghasilkan keluaran yang menjadi dasar untuk tahapan berikutnya. Tahap-tahap penelitian akan dilampirkan pada Gambar 2 sebagai rancangan tahapan penelitian

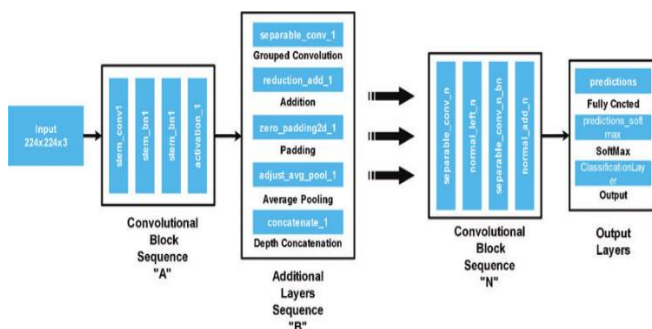


Gambar 2. Alur Penelitian

Beberapa tahapan ini diperlukan untuk membuat acuan sebagai solusi pemecahan masalah yang dapat terjadi. Tahapan metode penelitian memegang peran penting untuk memastikan keakuratan dan validitas hasil yang diperoleh. Penelitian dimulai dengan pengumpulan dan pengolahan *dataset* yang berisi citra buah kakao dengan berbagai kondisi, termasuk penyakit seperti *Black Pod*, *Helopeltis sp*, *Pod Borer* dan kakao Normal. Setelah itu, tahap *preprocessing* seperti augmentasi, normalisasi, dan teknik peningkatan kualitas gambar diterapkan untuk

meningkatkan performa model yaitu model yang sudah terintegrasi dengan *NASNet-Mobile*.

NASNet-Mobile adalah arsitektur CNN yang dikembangkan oleh Google dengan pendekatan *Neural Architecture Search* (NAS), yaitu teknik otomatis untuk menemukan struktur jaringan terbaik. Arsitektur ini menggunakan *Normal Cell* dan *Reduction Cell*, yang memungkinkan pembelajaran fitur lebih optimal dengan efisiensi tinggi. Dalam penelitian ini, *NASNet-Mobile* digunakan sebagai *backbone* untuk ekstraksi fitur, dengan beberapa lapisan tambahan, termasuk *fully connected layer* dengan aktivasi sigmoid, guna mendukung klasifikasi multilabel pada citra buah kakao. Gambar 3 akan menampilkan arsitektur *NASNet-Mobile*.



Gambar 3. Arsitektur *NASNet-Mobile*

ReLU (*Rectified Linear Unit*) digunakan sebagai fungsi aktivasi dalam arsitektur model ini karena keunggulannya dalam mengatasi masalah *vanishing gradient* yang sering terjadi pada fungsi aktivasi seperti sigmoid atau tanh. *ReLU* bekerja dengan mengubah semua nilai negatif menjadi nol dan mempertahankan nilai positif secara linear, aktivasi *ReLU* dirumuskan dengan:

$$f(x) = \max(0, x)$$

Klasifikasi yang digunakan ialah klasifikasi *multi-label*, sehingga fungsi aktivasi yang digunakan ialah fungsi aktivasi sigmoid. Fungsi aktivasi sigmoid digunakan pada lapisan *output* karena model ini menerapkan klasifikasi multilabel, di mana satu gambar buah kakao dapat memiliki lebih dari satu jenis penyakit. Sigmoid mengubah *output* jaringan menjadi nilai probabilitas antara 0 dan 1, yang memungkinkan setiap kelas dievaluasi secara independen. Fungsi ini dirumuskan sebagai:

$$\sigma(x) = \frac{1}{1 + e^x}$$

Keunggulan utama sigmoid dalam klasifikasi multilabel adalah kemampuannya menangani label yang tidak eksklusif, berbeda dengan *softmax* yang lebih cocok untuk klasifikasi *multiclass*. Dengan sigmoid, model dapat menentukan kemungkinan setiap penyakit pada gambar kakao tanpa membatasi prediksi hanya pada satu kelas. Dalam penelitian ini, pemilihan *hyperparameter* dilakukan untuk mengoptimalkan kinerja model CNN berbasis *NASNet-Mobile* dalam klasifikasi penyakit pada buah

kakao. Beberapa *hyperparameter* utama yang digunakan akan diterangkan pada Tabel 1

Tabel 1. *Hyperparameter*

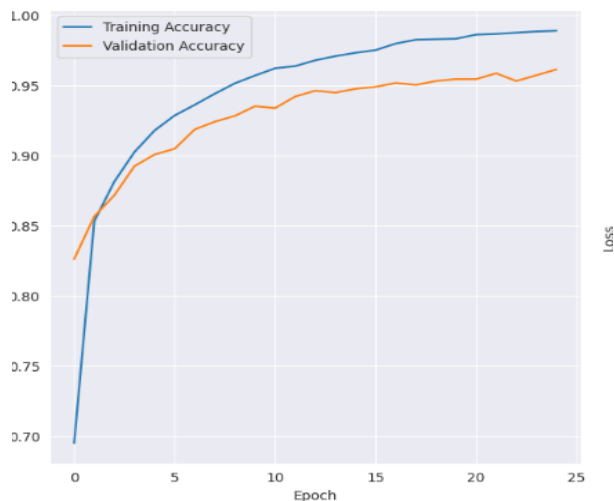
<i>hyperparameter</i>	<i>value</i>
Arsitektur	<i>NASNet-Mobile</i>
Ukuran Citra <i>Input</i>	224 x 224 x 3
<i>Batch</i>	64
Fungsi Aktivasi	ReLU, Sigmoid
<i>Learning Rate</i>	0.001 (<i>default Adam</i>)
<i>Loss Function</i>	<i>Binary Crossentropy</i>
<i>Epoch</i>	25
<i>Dropout</i>	0,2 (20%)

Pemilihan *hyperparameter* ini didasarkan pada eksperimen dan validasi model, dengan tujuan mendapatkan akurasi yang tinggi serta generalisasi yang baik terhadap data uji.

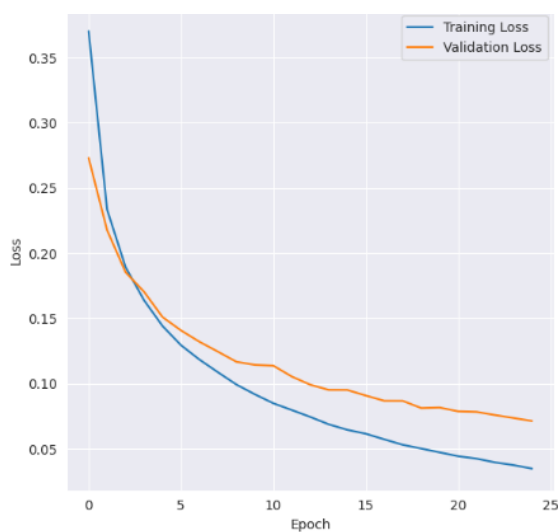
3. HASIL DAN PEMBAHASAN

Hasil pelatihan menunjukkan bahwa model yang dikembangkan berhasil mencapai tingkat akurasi yang tinggi pada data uji, yang mengindikasikan bahwa pendekatan yang digunakan terbukti efektif dalam mengidentifikasi dan membedakan jenis-jenis penyakit kakao secara tepat. Untuk memperkuat dan memperjelas hasil eksperimen yang telah dilakukan, disajikan beberapa grafik visualisasi yang merepresentasikan proses pelatihan dan evaluasi model. Grafik pertama menampilkan akurasi pelatihan dan validasi terhadap jumlah *epoch*, yang memberikan gambaran mengenai seberapa baik model mempelajari pola dari data latih dan seberapa mampu model tersebut melakukan generalisasi terhadap data yang belum pernah dilihat sebelumnya.

Sementara itu, grafik kedua menunjukkan *kurva loss*, yang menggambarkan tingkat kesalahan model dalam memprediksi label yang benar selama proses pelatihan berlangsung. Penurunan nilai *loss* yang konsisten dari waktu ke waktu menunjukkan bahwa model mengalami proses pembelajaran yang stabil dan bertahap menuju hasil yang optimal. Dengan demikian, proses perkembangan performa model dapat diamati secara lebih komprehensif, sekaligus memperkuat kesimpulan bahwa arsitektur yang digunakan memiliki kapabilitas yang kuat dalam klasifikasi citra penyakit buah kakao. Grafik akan ditampilkan pada Gambar 4 dan Gambar 5.



Gambar 4. Grafik akurasi pelatihan



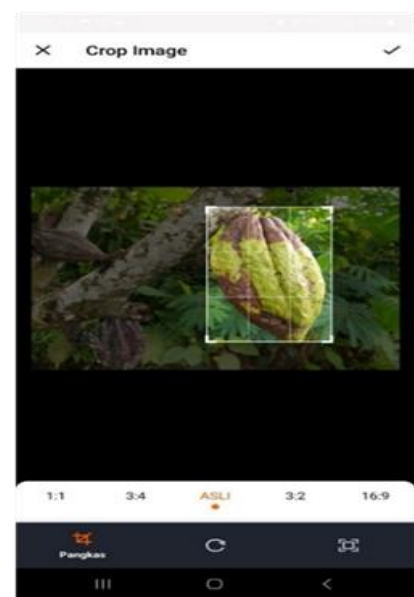
Gambar 5. Grafik Kurva Loss

Akurasi yang diperoleh pada saat pelatihan model dengan 25 epoch ialah akurasi sebesar 99% pada akurasi *training*, dan akurasi *validation* sebesar 96%. Pelatihan menggunakan *Google Collab*, bahasa pemrograman *Python* dan juga library *TensorFlow* yang dapat diakses secara publik. Akurasi pelatihan yang terus naik diiringi akurasi validasi yang juga terus naik menandakan model tidak mengalami *overfitting*, model *transfer learning* dengan algoritma *NASNet-Mobile* telah membantu sistem memahami pola dari setiap kelas pada *dataset*. Peneliti juga menggunakan *precision*, *Recall*, dan *F1-score* sebagai metrik evaluasi kinerja model pada *Google Collab* dimana menghasilkan nilai *precision* sebesar 94,9%, *Recall* sebesar 94,9% dan *F1-score* sebesar 94,9%. Nilai *precision*, *Recall*, dan *F1-score* masing-masing akan ditampilkan pada Tabel 2.

Tabel 2. Informasi Metrik Evaluasi pada *Google Collab*

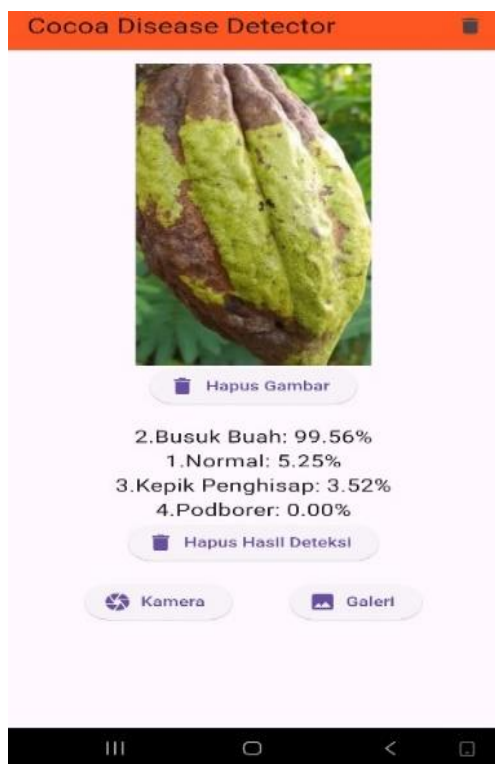
Kelas	<i>precision</i>	<i>Recall</i>	<i>F1-1 Score</i>
Normal	95%	94%	94%
Blackpod	95%	92%	93%
<i>Helopeltis sp</i>	96%	96%	96%
PodBorer	94%	98%	96%

Model yang dilatih telah mampu menampilkan probabilitas penyakit yang telah didefinisikan sebelumnya yaitu *Black Pod*, *Helopeltis sp*, *Pod Borer*, dan probabilitas seberapa besar buah mendekati definisi buah normal dengan label Normal. Model telah berhasil diimplementasikan ke dalam perangkat *Android* pada Gambar 6 menampilkan desain *user interface*



Gambar 6. Citra Pada Android

Citra kakao yang akan mengalami proses klasifikasi menggunakan sistem akan di-input-kan melalui menu *input* gambar atau langsung menggunakan kamera untuk memotret citra kakao yang ingin diidentifikasi. Citra yang diperoleh kemudian akan diproses oleh sistem untuk mengidentifikasi jenis penyakit yang mungkin menyerang buah tersebut secara otomatis dan cepat, sehingga pengguna dapat memperoleh informasi diagnosis secara instan dan akurat. Hasil klasifikasi ini juga dapat dijadikan sebagai referensi awal bagi petani atau pengguna dalam mengambil tindakan penanganan yang sesuai terhadap penyakit yang terdeteksi. Gambar 7 akan ditampilkan hasil klasifikasi dari citra buah kakao yang sudah di-input-kan ke sistem.



Gambar 7. Hasil Klasifikasi Model Pada Android

Model menghasilkan probabilitas untuk setiap kelas, probabilitas ini berkisar 0 sampai 1 lalu probabilitas akan ditampilkan dalam bentuk *presentase*, dimana semakin tinggi nilainya semakin yakin model bahwa gambar tersebut termasuk dalam kelas tertentu. Gambar 8 menunjukkan hasil klasifikasi citra setelah dilakukan *crop (focus)*, sedangkan Gambar 9 menunjukkan hasil klasifikasi tanpa dilakukan *crop*.



Gambar 8. Citra yang difokuskan (*cropping*)



Gambar 9. Citra yang tidak difokuskan

Model kesulitan mengklasifikasikan penyakit dengan benar karena jarak yang terlalu jauh, dan citra tidak berfokus pada buah kakao, sehingga model salah mengklasifikasikan *Blackpod* sebagai *Pod Borer*. Selanjutnya adalah pengujian, pengujian pada data test adalah langkah penting dalam proses evaluasi sistem, dengan melakukan pengujian data test, peneliti dapat mengetahui apakah model mengenali pola yang telah dipelajari selama proses pelatihan saat model sudah terintegrasi ke dalam perangkat *Android*. Berikut ditampilkan informasi hasil klasifikasi pada data uji yang pada Tabel 3.

Tabel 3. Pengujian Pada Data Uji di Lapangan

<i>citra</i>	<i>prediksi</i>	<i>Label benar</i>	<i>status</i>
Citra 1	Busuk Buah: 99%	BusukBuah	Benar
Citra 2	<i>Helopeltis</i> : 96%	<i>Helopeltis</i>	Benar
Citra 3	Busuk Buah: 91%	Busuk Buah	Benar
Citra 4	<i>Pod Borer</i> :56%	<i>Pod Borer</i>	Benar
Citra 5	Busuk Buah: 99%	Busuk Buah	Benar
Citra 6	PodBorer: 91%	Busuk Buah	Salah
Citra 7	Busuk Buah: 99%	Busuk Buah	Benar
Citra 8	PodBorer: 67%	<i>Pod Borer</i>	Benar
Citra 9	<i>Helopeltis</i> : 86%	Busuk Buah	Salah
Citra 10	Busuk Buah: 97%	Busuk Buah	Benar
Citra 11	Busuk Buah: 88%	Busuk Buah	Benar
Citra 12	Busuk Buah: 99%	Busuk Buah	Benar
Citra 13	Busuk Buah: 99%	Busuk Buah	Benar

<i>citra</i>	<i>prediksi</i>	<i>Label benar</i>	<i>status</i>
Citra 14	<i>Helopeltis</i> : 84%	<i>Helopeltis</i>	Benar
Citra 15	<i>Helopeltis</i> : 99%	<i>Helopeltis</i>	Benar
Citra 16	Busuk Buah: 99%	Busuk Buah	Benar
Citra 17	Busuk Buah: 99%	Busuk Buah	Benar
Citra 18	Busuk Buah: 99%	Busuk Buah	Benar
Citra 19	Busuk Buah: 99%	Busuk Buah	Benar
Citra 20	Busuk Buah: 88%	Busuk Buah	Benar
Citra 21	<i>Helopeltis</i> : 91%	<i>Helopeltis</i>	Benar
Citra 22	<i>Pod Borer</i> : 94%	<i>Pod Borer</i>	Benar
Citra 23	<i>Pod Borer</i> : 75%	<i>Pod Borer</i>	Benar
Citra 24	Busuk Buah: 71%	Busuk Buah	Benar
Citra 25	Normal: 99%	Normal	Benar
Citra 26	Normal: 94%	Normal	Benar
Citra 27	Normal: 99%	Normal	Benar
Citra 28	Normal: 96%	Normal	Benar
Citra 29	<i>Helopeltis</i> : 97%	<i>Helopeltis</i>	Benar
Citra 30	<i>Helopeltis</i> : 99%	<i>Helopeltis</i>	Benar

Data-data yang digunakan sebagai data uji merupakan citra kakao yang diambil langsung oleh peneliti, dengan demikian data-data tersebut bersifat data primer. Citra yang akan dijadikan data *test* tidak pernah digunakan oleh peneliti selama proses pelatihan baik sebagai data *training* maupun *validation*. Persentase terbesar mewakili seberapa besar keyakinan model bahwa citra terklasifikasi penyakit tertentu. Hasil pengujian model yang terintegrasi ke dalam perangkat *Android* yang dilampirkan pada Tabel 3 sudah mencakup Label Sebenarnya (Label Benar), Hasil Klasifikasi (Prediksi), Status Prediksi (Status) apakah benar atau salah, dan citra kakao yang ingin diklasifikasikan itu sendiri. Hasil pengujian akan digunakan untuk menghitung akurasi sebagai bahan evaluasi model yang terintegrasi ke perangkat *Android*, perhitungan akurasi menggunakan rumus:

$$\text{Akurasi} = \frac{\text{Jumlah Klasifikasi Benar}}{\text{Total Data Uji}} \times 100\%$$

Sesuai formula di atas maka akurasi model yang digunakan pada uji coba perangkat *Android* sebagai media untuk melakukan proses klasifikasi akan dihitung sebagai berikut:

$$\begin{aligned} \text{Akurasi} &= \frac{28 \text{ (data diklasifikasikan benar)}}{30 \text{ (total data uji)}} \times 100\% \\ &= 93,33\% \end{aligned}$$

metrik lain juga digunakan sebagai alat evaluasi yaitu metrik *Recall*, metrik *Recall* dipilih karena metrik *Recall* mengukur seberapa banyak label yang sebenarnya positif berhasil dideteksi oleh model (*True Positif* dibandingkan

dengan *True Positif* dan *False Negatif*). Metrik ini dipilih agar memastikan bahwa model mampu membaca penyakit yang ada sebaik mungkin, jika menggunakan metrik *Presisi*, dimana dihitung juga berapa label yang terklasifikasi *False Positif*, ini akan menjadikan nilai *Presisi* menjadi rendah karena banyak label akan terklasifikasi sebagai *False positif* walaupun akurasi sangat-sangat kecil contohnya nilai 0,01%.

$$\text{Recall} = \frac{\text{Jumlah TP}}{\text{Jumlah TP} + \text{Jumlah FN}} \times 100\%$$

$$\text{Recall} = \frac{68 \text{ (dari 120 label yang ada pada 30 citra)}}{68+1} \times 100\%$$

$$\text{Recall} = 98,5\%$$

sedangkan akurasi *Presisi* didapat dengan formula sebagai berikut:

$$\text{Presisi} = \frac{\text{Jumlah TP}}{\text{Jumlah TP} + \text{Jumlah FP}} \times 100\%$$

$$\begin{aligned} \text{Presisi} &= \frac{68 \text{ (dari 120 label yang ada pada 30 citra)}}{68+51} \times 100\% \\ \text{Presisi} &= 57,1\% \end{aligned}$$

F1-score adalah rata-rata harmonik dari *Presisi* dan *Recall*, yang digunakan untuk menyeimbangkan keduanya.

$$F1 - \text{Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \times 100\%$$

$$F1 - \text{Score} = 2 \times \frac{0.571 \times 0.985}{0.571 + 0.985} \times 100\%$$

$$F1 - \text{Score} = 71,6\%$$

Dengan demikian akurasi model dalam memprediksi penyakit sebuah citra kakao yang di-input-kan menggunakan perangkat *Android* memiliki nilai sebesar 93,33%, lalu dalam kalkulasi menggunakan metrik *Recall*, didapat nilai akurasi sebesar 98,5%, *Presisi* sebesar 57,1%, dan *F1-score* sebesar 71,6%. Hal ini menunjukkan model mampu membaca penyakit dengan baik dengan sedikit citra dimana model tidak mampu mendeteksi keberadaan penyakit. *Presisi* yang lebih rendah (57.1%) menunjukkan bahwa model menghasilkan cukup banyak *False Positives* dibandingkan dengan jumlah *True Positif*. Artinya, model sering salah mengidentifikasi label positif. *F1-score* (71.6%) mencerminkan keseimbangan antara *Presisi* dan *Recall*. Dalam kasus ini, meskipun *Recall* tinggi, *Presisi* yang lebih rendah menurunkan nilai *F1-Score*.

4. KESIMPULAN

Berdasarkan hasil penelitian mengenai deteksi penyakit pada buah kakao menggunakan model CNN dengan arsitektur *NASNet-Mobile* yang telah diimplementasikan ke dalam aplikasi *Android*, diperoleh beberapa kesimpulan yaitu akurasi pelatihan 99,11% menunjukkan bahwa model dapat mengenali pola pada data pelatihan dengan sangat baik, akurasi validasi 96,14% mengindikasikan model memiliki kemampuan generalisasi yang baik terhadap data yang tidak dilihat selama proses pelatihan, akurasi pada data

uji model 94,88% memperlihatkan bahwa model tetap memiliki performa tinggi pada data baru, meskipun sedikit menurun dibandingkan akurasi validasi. Penurunan ini dapat disebabkan oleh distribusi data uji yang berbeda dari data pelatihan. Metrik lain seperti *Recall* menghasilkan nilai 94,9%, *Presisi* 94,9% dan *F1-score* 94,9%. Akurasi pada 30 Data Uji menggunakan Aplikasi *Android* 93,33% dan metrik *Recall* sebesar 98,5% menunjukkan bahwa implementasi model ke dalam aplikasi *Android* cukup baik, meskipun ada sedikit penurunan yang mungkin disebabkan oleh kualitas gambar masukan sehingga banyak label terklasifikasi *false positif* dengan walaupun dengan nilai akurasi yang cukup kecil. Metrik lain seperti *Presisi* menghasilkan nilai 57,1% dan *F1-score* 71,6%.

Algoritma *NASNet-Mobile* dapat membantu model CNN dalam proses pemahaman pola citra pada setiap kelas dengan baik. Secara keseluruhan, model memiliki performa yang baik dan stabil dalam mendeteksi penyakit pada buah kakao, baik di lingkungan pengujian model maupun implementasi pada aplikasi *android* sehingga dapat menjawab permasalahan yang peneliti angkat yaitu bagaimana menciptakan sebuah sistem berbasis aplikasi *android* yang dapat mengklasifikasikan penyakit pada buah kakao yaitu dengan menggunakan CNN dan metode *transfer learning* algoritma *NASNet-Mobile*.

DAFTAR PUSTAKA

- [1] A. Gozali and H. Sultan, "ANALISIS PEMASARAN KAKAO DI DESA KAYUBOKO KECAMATAN PARIGI BARAT KABUPATEN PARIGI MOUTONG Marketing Analysis Of Cocoa Kayuboko Village Parigi Barat Sub district Parigi Moutong Regency," *J. Agrotekbis*, vol. 11, no. 4, pp. 856–865, 2023.
- [2] Anderson Matitaputty, Handry R.D. Amanupunyo, and Wilhelmina Rumahlewang, "KERUSAKAN TANAMAN KAKAO (*Theobroma cacao* L.) AKIBAT PENYAKIT PENTING DI KECAMATAN TANIWEL KABUPATEN SERAM BAGIAN BARAT," 2014, 2AD.
- [3] M. M. I. Al-Ghiffary, C. A. Sari, E. H. Rachmawanto, N. M. Yacoob, N. R. D. Cahyo, and R. R. Ali, "Milkfish Freshness Classification Using Convolutional Neural Networks Based on Resnet50 Architecture," *Advance Sustainable Science Engineering and Technology*, vol. 5, no. 3, p. 0230304, Oct. 2023, doi: 10.26877/asset.v5i3.17017.
- [4] W. Bismi and M. Qomaruddin, "Jurnal Informatika dan Rekayasa Perangkat Lunak Klasifikasi Citra Genus *Panthera* Menggunakan Pendekatan Deep learning Berbasis Convolutional Neural Network (CNN)," vol. 5, no. 2, pp. 172–179, 2023.
- [5] U. Sri Rahmadhani and N. Lysbetti Marpaung, "Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN," vol. 8, no. 2, 2023.
- [6] Muhammad Rijal, Yani Andi Muhammad, and Rahman Abdul, "DETEKSI CITRA DAUN UNTUK KLASIFIKASI PENYAKIT PADI MENGGUNAKAN PENDEKATAN DEEP LEARNING DENGAN MODEL CNN," *Jurnal Teknologi Terpadu*, vol. 10, pp. 56–62, 2024.
- [7] L. Trihardianingsih, A. Sunyoto, and T. Hidayat, "Classification of Tea Leaf Diseases Based on ResNet-50 and Inception V3," *Sinkron*, vol. 8, no. 3, pp. 1564–1573, Jul. 2023, doi: 10.33395/sinkron.v8i3.12604.
- [8] I. K. Trisiawan and Y. Yuliza, "Penerapan Multi-Label Image Classification Menggunakan Metode Convolutional Neural Network (CNN) Untuk Sortir Botol Minuman," *Jurnal Teknologi Elektro*, vol. 13, no. 1, p. 48, Feb. 2022, doi: 10.22441/jte.2022.v13i1.009.
- [9] Pratama M. Duta, Gustriansyah Rendra, and Purnamasari Evi, "KLASIFIKASI PENYAKIT DAUN PISANG MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN)," *Jurnal Teknologi Terpadu*, vol. 10, pp. 1–6, 2024.
- [10] N. M. K. K. Handayani, E. Y. Hidayat, M. Naufal, and P. L. W. E. Putra, "Pengenalan Ekspresi Wajah Menggunakan Transfer Learning MobileNetV2 dan EfficientNet-B0 dalam Memprediksi Perkelahian," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 1, p. 106, Jan. 2024, doi: 10.30865/mib.v8i1.7048.
- [11] S. Kumi, D. Kelly, J. Woodstuff, R. K. Lomotey, R. Orji, and R. Deters, "Cocoa Companion: Deep Learning-Based Smartphone Application for Cocoa Disease Detection," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 87–94. doi: 10.1016/j.procs.2022.07.013.
- [12] A. Khebli, Y. Kateb, H. Megloul, S. Aguib, M. Salah, and K. Touhami, "Classifying Surface Fault in Steel Strips Using a Customized NasNet-Mobile CNN and Small Dataset," 2024. [Online]. Available: <https://www.researchgate.net/publication/378876350>
- [13] A. Ghodekar and A. Kumar, "LightLeafNet: Lightweight and Efficient NASNetMobile Architecture for Tomato Leaf Disease Classification," 2023. [Online]. Available: <https://ssrn.com/abstract=4570347>

- [14] D. D. Nur Cahyo, M. Anwar Fauzi, J. Tri Nugroho, and K. Kusrini, "Analisis Perbandingan Optimizer pada Arsitektur NASNetMobile Convolutional Neural Network untuk Klasifikasi Ras Kucing," *J Teknol*, vol. 15, no. 2, pp. 171–177, Jan. 2023, doi: 10.34151/jurtek.v15i2.4025.
- [15] R. Setiyo Budiawan and B. Hartono, "Pengembangan Sistem Pendeteksi Jenis Sayuran dengan Metode CNN Berbasis *Android*," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 5, no. 1, pp. 62–72, 2023.