

Analisis Perbandingan Arsitektur *Convolutional Neural Network* (CNN) untuk Deteksi Karat Pada Daun Tanaman Kopi

Muhammad Rayhan Nuansa Adha^{a,1} Nadia Sukesu Sianipar^{a,2} Peksyaji^{a,3}

^a Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Indonesia

¹ peksyaji@ui.ac.id ² nadia.sukesu@ui.ac.id ³ muhammad.rayhan03@ui.ac.id

INFORMASI ARTIKEL

NPM 2006571053
NPM 2106700776
NPM 2006522322

Kata Kunci: CNN, Deteksi karat, Daun Kopi, ResNet101, MobileNet, DenseNet121

ABSTRAK

Kopi memegang peranan penting dalam kehidupan masyarakat. Delapan dari sepuluh orang dewasa di Indonesia minum rata-rata 2,5 cangkir kopi per hari dan sembilan dari sepuluh rumah tangga mempunyai kopi di dapur mereka. Indonesia adalah negara penghasil kopi terbesar keempat di dunia. Namun, jika dibandingkan dengan 3 negara lainnya, hasil produksi kopi Indonesia masih terbilang kecil. Banyak faktor yang membuat hal ini terus berlanjut, salah satunya adalah karat daun kopi yang disebabkan oleh jamur *Hemileia broadatrix* yang banyak menyerang daerah penghasil kopi. Jika penanganan terhadap penyakit ini lambat, maka penyakit pada satu pohon dapat menular pada pohon lainnya. Jika tidak dideteksi sejak dini, hal ini dapat sangat mengurangi hasil kopi atau lebih buruk lagi, memusnahkan seluruh perkebunan yang berdampak pada para petani. Penelitian ini mengusulkan metodologi untuk mendeteksi penyakit karat daun kopi menggunakan visi komputer dan menggunakan beberapa arsitektur *Convolutional Neural Networks* (CNN) yang telah dilatih sebelumnya. Secara khusus, penelitian ini menggunakan tiga model, yaitu *ResNet101V2*, *MobileNetV2*, dan *DenseNet121*. Mereka memiliki akurasi pelatihan masing-masing sebesar 81,83%, 85,5%, 86,88% dan akurasi validasinya masing-masing sebesar 89,45%, 85,55%, 92,19%. Dari tiga CNN tersebut, *DenseNet121* mencapai akurasi pengujian tertinggi sebesar 92,19% dan juga unggul dalam metrik evaluasi lainnya seperti *Precision*, *Recall*, dan *F1-score*. Secara keseluruhan, hasil penelitian ini merupakan satu langkah lebih dekat untuk mencapai peningkatan produksi kopi dan penghidupan petani.

Copyright © 2024.
All rights reserved.

I. Pendahuluan

Kopi memainkan peran penting dalam kehidupan masyarakat. Karena biasanya dikonsumsi di pagi hari, kebanyakan orang mengatakan bahwa mereka dapat mengandalkan kopi untuk memberi energi dan membantu mereka menjalani hari. Aroma kopi dikatakan menenangkan, dan rasa kopi yang enak justru membuat rileks [1].

Indonesia baru mampu memasok sebesar 4% untuk pemenuhan kebutuhan kopi Uni Eropa, hal ini sangat disayangkan padahal pasar yang tersedia amatlah besar. Menurut peneliti bidang pangan dari Center for Indonesia Policy Studies (CIPS), setidaknya ada dua hal yang menjadi penyebab kurangnya produktivitas kopi di Indonesia. Pertama karena pohon pada usia yang sudah tua akan sangat rentan

terhadap serangan penyakit, kemudian kedua karena terdapat karat pada daun kopi yang disebabkan oleh jamur bernama *Hemileia broadatrix* yang dapat menyebar melalui angin, hujan, atau kontak fisik sehingga mengubah warna daun dari hijau cerah menjadi kuning kecokelatan. Penyakit pada penyebab kedua ini membuat daun menjadi busuk dan rontok dengan sendirinya, sehingga sangat menurunkan kualitas dan kuantitas buah kopi. Penyakit ganas dan sangat menular ini membuat tanaman tidak dapat digunakan, menyebabkan kerugian besar sebesar 30% hingga 50% pada produksi kopi yang berdampak pada penghidupan tanaman tersebut [2].

Karat kopi adalah penyakit kopi yang paling signifikan dampaknya terhadap perekonomian global, dan kopi adalah komoditas pertanian paling berharga yang diperdagangkan secara internasional. Di negara-negara yang perekonomiannya sangat bergantung pada ekspor kopi, penyakit karat mempunyai pengaruh yang sangat besar terhadap petani kopi, industri jasa dan pendukung, dan bahkan lembaga perbankan. Bahkan sedikit penurunan hasil kopi atau sedikit kenaikan biaya produksi dapat menimbulkan dampak yang signifikan [3].

Dalam studi yang dilakukan oleh Marcos A., Silva Rodovalho N., dan Backes A. (2019), mereka mengusulkan penggunaan *Convolutional Neural Network* (CNN) untuk mendeteksi sinyal infeksi karat pada daun kopi. Mereka membuat dataset yang terdiri dari 159 gambar daun kopi dengan kamera digital beresolusi 12 megapiksel, kemudian gambar-gambar tersebut diberi anotasi secara manual oleh ahli. Dataset ini digunakan untuk mengevaluasi CNN. Hasilnya menunjukkan bahwa pendekatan mereka dapat mendeteksi infeksi dengan tingkat presisi yang tinggi, yang dibuktikan oleh nilai koefisien *Dice* yang tinggi. CNN yang dilatih mencapai akurasi 95% dengan hanya menggunakan gambar berukuran 40x40 piksel [4].

Dalam penelitian yang dilakukan oleh Javierto, D., Martin, J., dan Villaverde, J. (2021), mereka mengembangkan sistem yang menggunakan Raspberry Pi 4 Model B untuk menangkap gambar daun kopi di lingkungan dengan pencahayaan terkontrol. Mereka kemudian menggabungkan YOLOv3 dengan MobileNetv2 dan melatih model ini menggunakan kumpulan data publik. Mereka berhasil mendeteksi penyakit daun kopi dengan akurasi mencapai 90% [5].

Dalam penelitian lainnya, Yasin, M., dan Maki, W. (2022) menggunakan pendekatan yang berbeda untuk mengklasifikasikan penyakit tanaman kopi. Dengan menganalisis foto daun, mereka menunjukkan bahwa pengolahan gambar dapat membantu mendeteksi dan mengobati penyakit tanaman. Mereka menggunakan pendekatan *Color Moments* dalam ruang warna RGB dan YCrCb untuk mengekstraksi informasi warna, serta *Gray Level Co-occurrence Matrix* (GLCM) untuk mengekstraksi informasi tekstur. Untuk klasifikasinya, mereka menggunakan algoritma *K-nearest neighbor* (KNN). Penelitian mereka menunjukkan bahwa model ini mencapai akurasi sebesar 81,3% [6].

Dalam studi yang dilakukan oleh Ayikpa dkk. (2022), mereka bertujuan untuk mengembangkan metode yang efektif dalam mengidentifikasi dan mengklasifikasikan penyakit daun kopi dengan memanfaatkan metode pembelajaran mesin klasik dan pembelajaran mendalam. Mereka membangun sebuah *Convolutional Neural Network* (CNN) khusus dan mengujinya menggunakan dataset daun kopi Arabika JMuBEN. Hasil dari metode pembelajaran mesin klasik menunjukkan akurasi antara 81,03% hingga 100%, dengan *Support Vector Machine* (SVM) dan *Random Forest* menunjukkan kinerja terbaik. Namun, metode pembelajaran mendalam memberikan hasil yang lebih unggul, dengan akurasi antara 97,37% hingga 100%. CNN khusus menunjukkan performa terbaik dengan nilai *loss* terendah sebesar 0,013 [7].

Pendekatan unik untuk klasifikasi penyakit tanaman secara *real-time* diperkenalkan oleh Falaschetti dkk. (2022), yang menggunakan *Convolutional Neural Network* (CNN) pada platform OpenMV Cam

H7 Plus yang hemat biaya dan daya. Mereka menggunakan dataset yang ditingkatkan dari PlantVillage dan dataset ESCA untuk melatih CNN, yang kemudian diimplementasikan pada kamera visi mesin berdaya rendah dengan kemampuan pemrograman Python. Pengguna dapat melihat hasil klasifikasi kamera secara real-time pada layar LCD. Eksperimen menunjukkan bahwa detektor gambar berbasis CNN, dengan akurasi sekitar 98,10% dan 95,24%, dapat diimplementasikan dengan sukses pada sistem yang memiliki keterbatasan sumber daya [8].

Penelitian oleh Kumar dkk. (2020) menggunakan foto yang telah diproses sebelumnya dari daun kopi yang sehat maupun yang terkena penyakit seperti penampang daun, karat daun, bercak *cercospora*, dan *phoma*. Mereka menggunakan beberapa algoritma CNN pada dataset yang telah mengalami augmentasi data dan ekstraksi fitur menggunakan model *Inception V3* yang telah dilatih sebelumnya. *Radial Basis Function Neural Network* mencapai akurasi sebesar 95% tanpa nilai *loss*, sementara CNN dengan augmentasi data mencapai akurasi 95% dengan nilai *loss* 0,10%. CNN yang menggunakan augmentasi data dan *transfer learning* dengan arsitektur *Inception V3* mencapai akurasi tertinggi sebesar 97,61% dengan nilai *loss* 0,3. Hasil ini diperoleh melalui berbagai teknik yang digunakan [9].

Penelitian ini bertujuan untuk membandingkan kinerja 3 model *Convolutional Neural Network* (CNN) yang banyak digunakan untuk mendeteksi karat daun kopi. Dengan menggunakan kumpulan data karat daun kopi yang seragam, penelitian ini akan membandingkan dan menilai *accuracy*, *precision*, *recall*, dan *F1-score* dari berbagai model CNN. Temuan penelitian ini dapat membantu dalam menentukan metode terbaik berbasis CNN untuk mendeteksi karat daun kopi, yang dapat berdampak langsung pada pertanian untuk diagnosis dini dan pengendalian penyakit ini.

II. Tinjauan Pustaka

2.1 Kopi Robusta

Kopi robusta (*Coffea canephora*) adalah salah satu dari dua spesies utama kopi yang dibudidayakan secara komersial selain Arabika (*Coffea arabica*). Kopi ini berasal dari Afrika Barat dan Tengah, namun sekarang juga banyak dibudidayakan di Vietnam, Indonesia, dan Brasil. Kopi robusta dikenal memiliki rasa yang lebih kuat, pahit, dan keras dengan nada *earthy* atau *nutty*. Kandungan kafein robusta lebih tinggi, berkisar antara 2.2-2.7% dibandingkan Arabika yang hanya 1.2-1.5%, menjadikan robusta lebih tahan terhadap hama dan penyakit. Tanaman ini tumbuh dengan baik di iklim panas dan lembap pada ketinggian 200-800 meter di atas permukaan laut, lebih produktif, dan umumnya lebih mudah dibudidayakan dibandingkan Arabika.

Daun kopi robusta memiliki karakteristik yang khas. Daunnya lebih besar dan tebal dengan bentuk elips atau oval, serta berwarna hijau tua yang menunjukkan kandungan klorofil tinggi, esensial untuk fotosintesis. Permukaan daun robusta lebih kasar dan kaku dibandingkan daun Arabika. Daun kopi robusta yang sehat biasanya berwarna hijau tua, kuat, elastis, dan memiliki permukaan yang bersih tanpa tanda-tanda serangan hama atau penyakit.

Namun, daun kopi robusta juga rentan terhadap kondisi buruk seperti serangan karat daun dan tungau merah. Karat daun kopi, yang disebabkan oleh jamur *Hemileia vastatrix*, ditandai dengan bintik-bintik kuning atau oranye yang berkembang menjadi coklat atau merah, menyebabkan daun rontok dan mengurangi efisiensi fotosintesis serta produksi buah. Bertrand dan Guyot (2021) mencatat bahwa "Karat daun kopi adalah salah satu penyakit utama yang mempengaruhi produktivitas kopi robusta" [10]. Serangan tungau merah (*Oligonychus coffeae*) ditandai dengan bintik-bintik kuning kecil yang berubah menjadi coklat atau hitam, membuat daun layu atau kering. Souza dan Furtado (2017) menyatakan bahwa "Tungau merah adalah hama signifikan yang dapat menyebabkan kerusakan parah

pada daun kopi, mengurangi hasil panen" [11]. Kedua kondisi ini dapat merusak daun, mengurangi efisiensi fotosintesis, dan pada akhirnya mempengaruhi pertumbuhan tanaman dan hasil panen.

2.2 Image Classification

Dalam konteks klasifikasi gambar, piksel gambar dapat langsung digunakan sebagai masukan ke jaringan saraf *feed-forward* standar. Jaringan ini berfungsi dengan menghubungkan setiap piksel gambar ke neuron-neuron dalam lapisan tersembunyi, yang kemudian memproses informasi untuk menghasilkan klasifikasi akhir. Namun, meskipun pendekatan ini dapat diterapkan, penggunaan piksel gambar secara langsung untuk jaringan saraf *feed-forward* standar menghadapi beberapa tantangan utama.

Patch gambar yang relatif kecil pun dapat memiliki ribuan piksel. Misalnya, gambar berukuran 32x32 piksel saja sudah mengandung 1024 piksel. Ketika piksel-piksel ini digunakan sebagai input ke jaringan saraf, setiap piksel memerlukan parameter bobot koneksi yang harus dilatih. Dengan demikian, jumlah total parameter dalam jaringan tersebut menjadi sangat besar, yang dapat menyebabkan beberapa masalah signifikan:

- a. Kompleksitas Model
Menurut teori dimensi VC (*Vapnik-Chervonenkis*), jumlah parameter yang besar dalam model meningkatkan kompleksitasnya. Kompleksitas ini mencerminkan kapasitas model untuk mempelajari berbagai pola dalam data. Semakin kompleks modelnya, semakin besar potensi untuk *overfitting*, yaitu ketika model terlalu baik dalam mencocokkan data pelatihan namun berkinerja buruk pada data uji atau data baru.
- b. Kebutuhan Sampel Pelatihan
Model yang lebih kompleks memerlukan lebih banyak sampel pelatihan untuk menghindari *overfitting*. Dengan sejumlah besar parameter, model memerlukan data pelatihan yang cukup banyak untuk memastikan bahwa bobot yang dihasilkan dapat digeneralisasi dengan baik ke data baru.

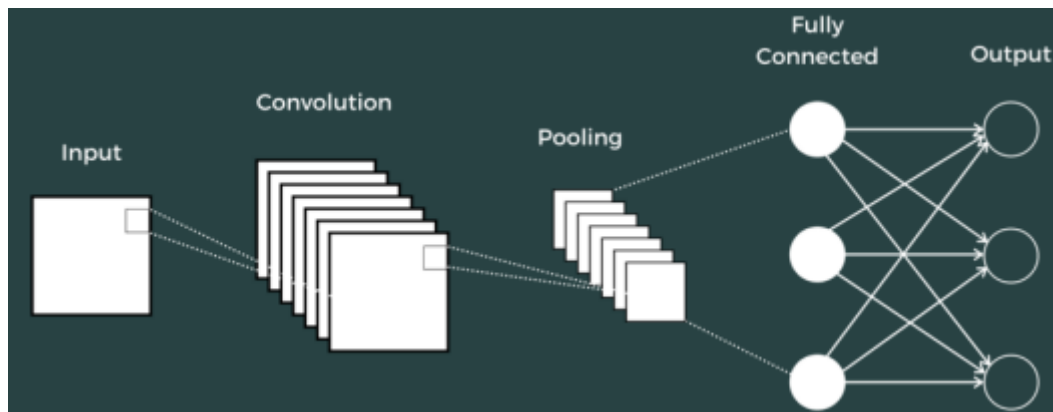
Untuk mengatasi masalah ini, model *Convolutional Neural Network* (CNN) menawarkan solusi yang lebih efisien. CNN menggabungkan bobot ke dalam filter kernel yang jauh lebih kecil. Filter ini bergerak melintasi gambar (konvolusi) untuk mendeteksi fitur-fitur penting seperti tepi, sudut, dan tekstur pada berbagai posisi dalam gambar. Beberapa keuntungan utama dari pendekatan CNN meliputi:

- a. Pengurangan Parameter
Dengan menggunakan filter yang lebih kecil, jumlah parameter yang perlu dilatih dalam model CNN jauh lebih sedikit dibandingkan dengan jaringan saraf yang sepenuhnya terhubung. Ini menyederhanakan model pembelajaran dan mengurangi risiko *overfitting*.
- b. Peningkatan Efisiensi
CNN lebih efisien dalam hal komputasi karena mengurangi jumlah perhitungan yang diperlukan untuk setiap lapisan. Proses konvolusi memungkinkan jaringan untuk fokus pada fitur-fitur lokal dalam gambar, yang kemudian digabungkan untuk memahami pola yang lebih kompleks.
- c. Ketangguhan Terhadap Variasi
CNN lebih tangguh dalam menghadapi variasi dalam data gambar, seperti pergeseran, skala, dan rotasi. Karena filter bergerak melintasi seluruh gambar, CNN dapat mengenali fitur yang sama di berbagai posisi dan skala, meningkatkan kemampuan generalisasi model.

Dengan demikian, CNN tidak hanya menyederhanakan model pembelajaran tetapi juga meningkatkan kecepatan dan ketangguhan jaringan dalam memecahkan masalah klasifikasi gambar. Pendekatan ini menjadikan CNN sebagai pilihan yang lebih efektif dan efisien dibandingkan dengan jaringan saraf tradisional yang sepenuhnya terhubung.

2.3 Convolutional Neural Network (CNN)

Convolution Neural Network merupakan salah satu dari pembelajaran mendalam dan mempunyai struktur yang dapat mengolah data dua dimensi dan mempunyai lapisan masuk, lapisan keluar dan lapisan tersembunyi yang berisi *convolutional layer*, *pooling layer*, dan *fully connected layers* [12]. Berikut model CNN yang dapat dilihat pada Gambar 2.3



Gambar 2.3 Model *Convolution Neural Network* [13]

Convolution Layer merupakan lapisan dan blok utama dari CNN yang digunakan dalam menghitung *output* dari *neuron* yang terhubung ke *local region* dari masukan citra [14]. *Convolution Layer* dianggap sebagai filter bagian *input* untuk meningkatkan pola yang diberikan. Menggunakan *filter* pada setiap *neuron* dengan cara menggeser untuk perkalian antara region kecil yang terhubung ke masukan citra dan dilanjutkan dalam melakukan operasi konvolusi pada keluaran lapisan.

Pooling Layer merupakan salah satu proses lapisan untuk menurunkan ukuran citra untuk memudahkan operasi pengolahan citra dan mempercepat komputasi untuk lapisan berikutnya [15]. *Pooling layer* ini bertujuan untuk mengurangi jumlah parameter dan proses komputasi, serta mencegah terjadinya akurasi tinggi pada prediksi pada data *training* namun tidak dapat memprediksi data *testing* atau dikenal dengan *overfitting*. Ada beberapa teknik untuk melakukan proses ini salah satunya yaitu mengambil nilai *maximum* dari sub citra atau disebut *Max pooling*. *Max pooling* ini dapat memecah output dari *relu* menjadi beberapa *grid* yang kecil lalu menggunakan nilai *maxsimum* dari setiap *grid* untuk membangun matriks citra yang direduksi.

Full Connected Layer salah satu lapisan yang terdiri beberapa *node*. Jumlah layer ini digunakan untuk menggabungkan ekstrasi fitur ke dalam kelas yang diperbarui dan di ulangi dalam dalam perkalian acak sehingga memiliki bobot dan bias untuk pendekatan kelas. Proses ini bisa membutuhkan waktu yang lama tergantung pada banyaknya fitur yang dihasilkan [14].

2.3.1 ResNet101V2

ResNet101V2 adalah varian dari arsitektur *Residual Networks (ResNet)* yang dirancang untuk meningkatkan performa dalam klasifikasi gambar dengan 101 lapisan dalam. Model ini mengatasi masalah *vanishing gradient* melalui penggunaan blok residual, di mana koneksi pendek memungkinkan sinyal input melompati beberapa lapisan, menjaga informasi penting tetap utuh. *ResNet101V2* memperkenalkan *pre-activation residual block*, yang melakukan aktivasi dan

normalisasi batch sebelum konvolusi, meningkatkan stabilitas dan efisiensi pelatihan dibandingkan dengan versi sebelumnya. Dengan kedalaman dan struktur ini, *ResNet101V2* mampu menangkap fitur kompleks dari data gambar, menjadikannya sangat efektif untuk tugas-tugas pengenalan pola dan klasifikasi gambar.

2.3.2 *MobileNetV2*

MobileNetV2 adalah arsitektur jaringan saraf konvolusi yang dirancang untuk perangkat dengan keterbatasan sumber daya seperti ponsel dan perangkat edge. Diperkenalkan oleh Google, *MobileNetV2* menggunakan blok residual dengan koneksi *shortcut* dan memperkenalkan konsep *inverted residuals* dan *linear bottlenecks*. Inverted residuals memanfaatkan ekspansi dan kompresi fitur untuk mengurangi jumlah operasi komputasi, sementara *linear bottlenecks* menjaga informasi penting tetap utuh dengan lapisan aktivasi ReLU hanya diterapkan pada ruang dimensi tinggi. Pendekatan ini menghasilkan model yang lebih efisien dan cepat, dengan performa tinggi dalam klasifikasi gambar dan deteksi objek, sekaligus mengurangi kebutuhan daya dan memori.

2.3.3 *DenseNet121*

DenseNet121 adalah arsitektur jaringan saraf konvolusional yang dikenal dengan konektivitas padat antar lapisan, di mana setiap lapisan menerima input dari semua lapisan sebelumnya. Diperkenalkan oleh Huang et al., *DenseNet121* memiliki 121 lapisan dan menggunakan koneksi langsung ini untuk meningkatkan aliran informasi dan gradien di seluruh jaringan, mengatasi masalah *vanishing gradient* dan mempromosikan pemanfaatan ulang fitur. Setiap lapisan dalam *DenseNet* menambahkan fitur baru sambil menggabungkan semua fitur sebelumnya, menghasilkan representasi yang lebih kaya dan lebih efisien. Struktur ini memungkinkan model untuk mencapai kinerja yang tinggi dengan lebih sedikit parameter dan komputasi dibandingkan dengan arsitektur tradisional, menjadikannya sangat efektif untuk tugas-tugas seperti klasifikasi gambar dan deteksi objek.

2.4 Augmentasi

Augmentasi merupakan teknik yang penting dari *computer vision* dalam manipulasi sebuah citra gambar sehingga mesin mampu mengenali berbagai citra gambar sekaligus menambah data secara tidak langsung [16]. Flip merupakan salah satu jenis dari augmentasi yang digunakan untuk mengubah posisi citra gambar secara vertikal ataupun horizontal. Salah satu jenis dari augmentasi lainnya yaitu random rotation yang berfungsi untuk merotasi sebuah citra, tetapi rotasi tersebut diberikan nilai derajat yang bisa kita tentukan sehingga citra gambar akan di rotasi maksimal derajat yang telah kita tentukan.

2.5 Evaluasi Model

Evaluasi suatu algoritma *machine learning* merupakan bagian esensial untuk mengukur kinerja suatu model *machine learning* yang telah dibuat. Pengukuran evaluasi dalam penelitian ini menggunakan model *confusion matrix* yang dapat dilihat dalam Tabel 5.10. Model *confusion matrix* terdiri dari *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN).

		<i>Actual</i>	
		<i>Positive</i>	<i>Negative</i>
<i>Predicted</i>	<i>Positive</i>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
	<i>Negative</i>	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

Tabel 2.5 *Confusion Matrix*

Evaluasi dapat dilakukan dengan menggunakan beberapa cara, seperti akurasi, presisi, *recall*, dan *F-Score*. Akurasi merupakan proporsi data benar (*true*) positif dan negatif dari data aktual. Dua pengukuran evaluasi untuk mengukur performa banyaknya data yang benar dari keseluruhan data aktual adalah presisi dan *recall*

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

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

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

F-Score merupakan pembobotan rata-rata dari presisi dan *recall* yang bergantung pada fungsi bobot β .

$$F - Score = F_{\beta} = (1 + \beta^2) \frac{(Recall)(Precision)}{\beta^2(Precision + Recall)}$$

Dengan $\beta = 1$, maka diperoleh *F-Score* standar sebagai berikut

$$F - Score = F_1 = 2 \times \left(\frac{Recall \times Precision}{Recall + Precision} \right)$$

(Dalianis *et al.* 2018)

III. Data dan Metodologi

3.1 Data

Data yang digunakan merupakan data sekunder yang bersumber dari situs *Kaggle*. Dataset tersebut berisi 1560 gambar dengan 769 gambar diantaranya merupakan daun kopi robusta dengan tungai dan bercak yang menandakan keberadaan karat pada daun kopi dan 791 sisanya merupakan gambar daun tanpa gejala tersebut yang menunjukkan daun yang sehat. Selain itu, dataset ini juga mencakup anotasi mengenai kondisi (sehat dan tidak sehat) dan tingkat keparahan penyakit (luas daun dengan bercak). Pada penelitian ini, kami akan mengklasifikasikan gambar daun kopi robusta yang sehat diberi label 0 dan gambar daun kopi robusta yang terinfeksi penyakit atau tidak sehat diberi label 1.

3.2 Metode

Pada penelitian ini, kami menggunakan metode *Convolutional Neural Network (CNN)* dalam melakukan klasifikasi gambar daun kopi robusta dengan beberapa arsitektur seperti *ResNet101v2*, *MobileNetv2*, dan *DenseNet121* sebagai *base model*. Berikut adalah tahapan-tahapan dalam melakukan klasifikasi gambar tersebut :

a. Persiapan data

Penelitian ini dimulai dengan persiapan dataset gambar yang dikategorikan ke dalam dua kelas. Dataset ini diatur dalam direktori yang terstruktur untuk memudahkan proses pelatihan dan validasi. Data dibagi menjadi dua subset: subset pelatihan (80%) dan subset validasi (20%). Pembagian ini bertujuan untuk mengevaluasi model secara akurat dengan menggunakan data yang tidak pernah dilihat model selama proses pelatihan.

Untuk meningkatkan generalisasi model, kami menggunakan teknik augmentasi data melalui ImageDataGenerator dari TensorFlow. Teknik augmentasi ini meliputi *rescaling* nilai piksel gambar untuk menormalkan data, serta berbagai transformasi seperti rotasi, pergeseran, skala, dan *flipping* secara horizontal. Teknik ini membantu model untuk mengenali fitur penting dalam gambar meskipun terjadi variasi pada input gambar.

b. Pembuatan Model

Dalam penelitian ini, kami memilih tiga arsitektur model *deep learning* yang berbeda yaitu :

1. *ResNet101V2*: Model ini menggunakan *residual blocks*, yang memungkinkan pelatihan jaringan sangat dalam tanpa mengalami masalah *vanishing gradient*. Setiap residual block terdiri dari beberapa lapisan konvolusi dan *shortcut connections* yang menghubungkan input dengan output block.
2. *MobileNetV2*: Model ini dirancang untuk efisiensi dengan menggunakan *inverted residuals* dan *linear bottlenecks*. Arsitektur ini mengurangi jumlah parameter dan operasi yang diperlukan, membuatnya ideal untuk aplikasi *mobile* dan *embedded*.
3. *DenseNet121*: Model ini menghubungkan setiap lapisan ke semua lapisan sebelumnya dalam blok yang disebut *dense blocks*. Pendekatan ini memungkinkan aliran gradien yang lebih baik dan *reuse* fitur yang lebih efisien, sehingga membantu mengurangi *overfitting*.

Setiap model diinisiasi dengan menggunakan arsitektur yang telah ditentukan dan *pretrained weights* dari *ImageNet* untuk meningkatkan konvergensi pelatihan. Kami kemudian menambahkan beberapa lapisan *dense* di atas lapisan konvolusi dasar untuk menyelesaikan tugas klasifikasi biner.

c. Penambahan Lapisan dan Kompilasi Model

Setelah menginisiasi model dasar, kami menambahkan lapisan *pooling global* untuk meratakan output dari lapisan konvolusi. Di atasnya, kami menambahkan lapisan *dense* dengan 512 unit dan fungsi aktivasi ReLU untuk menangkap fitur non-linear. Untuk mencegah *overfitting*, kami menambahkan lapisan *dropout* dengan *dropout rate* 0.4. Terakhir, lapisan output dengan satu neuron dan fungsi aktivasi sigmoid ditambahkan untuk menghasilkan probabilitas kelas.

Setiap model kemudian dikompilasi menggunakan Adam *optimizer*, yang dikenal efektif untuk berbagai tugas pembelajaran mendalam, dan fungsi *loss binary crossentropy*, yang sesuai untuk tugas klasifikasi biner. Metrik akurasi juga digunakan untuk memantau kinerja model selama pelatihan.

d. Pelatihan Model

Model dilatih menggunakan data pelatihan selama 100 *epoch*. Selama proses pelatihan, augmentasi data yang telah disebutkan sebelumnya diterapkan untuk meningkatkan generalisasi model. Proses pelatihan memanfaatkan pembagian data menjadi *batch-batch* kecil untuk efisiensi komputasi dan penggunaan memori.

Untuk setiap *epoch*, model dilatih pada seluruh *batch* data pelatihan dan kemudian dievaluasi pada data validasi. Ini memungkinkan pemantauan performa model secara terus-menerus dan deteksi dini apabila terjadi *overfitting*.

e. Evaluasi Model

Setelah pelatihan, setiap model dievaluasi menggunakan data validasi untuk mendapatkan metrik akurasi dan *loss*. Evaluasi ini memberikan gambaran langsung mengenai seberapa baik model dapat menggeneralisasi pada data yang belum pernah dilihat sebelumnya.

Untuk memberikan gambaran yang lebih komprehensif mengenai performa model, kami juga menghitung metrik evaluasi tambahan seperti *precision*, *recall*, dan *F1-score*. *Precision* mengukur proporsi prediksi positif yang benar, *recall* mengukur proporsi sebenarnya positif yang terprediksi dengan benar, dan *F1-score* merupakan rata-rata harmonik dari *precision* dan *recall*, memberikan gambaran keseimbangan antara kedua metrik tersebut.

IV. Hasil dan Diskusi

Dengan menggunakan dataset gambar daun kopi robusta yang terdiri dari 769 gambar daun dengan penyakit dan 791 gambar daun yang sehat seperti yang terlihat pada Gambar 4.1. Langkah pertama, kami melakukan pra-proses data, yakni melakukan augmentasi data melalui ImageDataGenerator dari TensorFlow. Teknik augmentasi ini meliputi *rescaling* nilai piksel gambar, rotasi, pergeseran, skala, dan *flipping* secara horizontal. Hasil dari augmentasi data ini menghasilkan dataset yang lebih bervariasi. Selanjutnya, kami membagi dataset yang kami miliki ke dalam data latih dan data validasi dengan proporsi 80:20. Artinya, dalam subset data latih terdapat 1248 gambar dan 312 gambar lainnya masuk ke dalam subset validasi.



Gambar 4.1. Daun Kopi Robusta Beserta Label

Setelah dataset kami siap untuk melakukan pelatihan dan pengujian model, kami menyiapkan model yang akan kami gunakan dalam penelitian ini, yakni *ResNet101V2*, *MobileNetV2*, dan *DenseNet121*. Dari model tersebut, kami menambahkan lapisan *pooling global*, lapisan *dense*

dengan 512 unit dan fungsi aktivasi ReLU, lapisan *dropout* dengan *dropout rate* 0.4, dan lapisan output dengan satu neuron dan fungsi aktivasi sigmoid. Kami menggunakan Adam *optimizer* untuk mengompilasi model dan menggunakan fungsi *loss binary crossentropy*. Ringkasan dari tiap model dapat dilihat pada Gambar 4.2 sampai Gambar 4.4.

Model: "sequential_1"		
Layer (type)	Output Shape	Param #
=====		
sequential (Sequential)	(None, 224, 224, 3)	0
resnet101v2 (Functional)	(None, 7, 7, 2048)	42626560
flatten (Flatten)	(None, 100352)	0
dense (Dense)	(None, 1024)	102761472
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 512)	524800
dropout_1 (Dropout)	(None, 512)	0
batch_normalization (Batch Normalization)	(None, 512)	2048
dense_2 (Dense)	(None, 1)	513
=====		
Total params: 145915393 (556.62 MB)		
Trainable params: 103287809 (394.01 MB)		
Non-trainable params: 42627584 (162.61 MB)		

Gambar 4.2. Model ResNet101V2

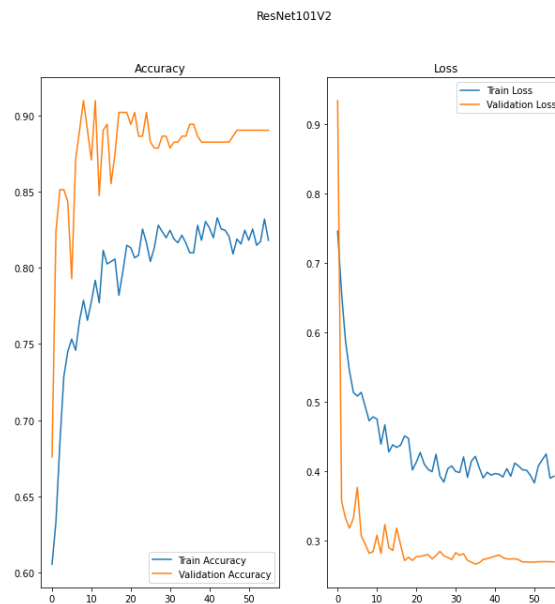
Model: "sequential_1"		
Layer (type)	Output Shape	Param #
=====		
sequential (Sequential)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Function)	(None, 7, 7, 1280)	2257984
flatten (Flatten)	(None, 62720)	0
dense (Dense)	(None, 1024)	64226304
dropout (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 512)	524800
dropout_1 (Dropout)	(None, 512)	0
batch_normalization (Batch Normalization)	(None, 512)	2048
dense_2 (Dense)	(None, 1)	513
=====		
Total params: 67,011,649		
Trainable params: 64,752,641		
Non-trainable params: 2,259,008		

Gambar 4.3. Model MobileNetV2

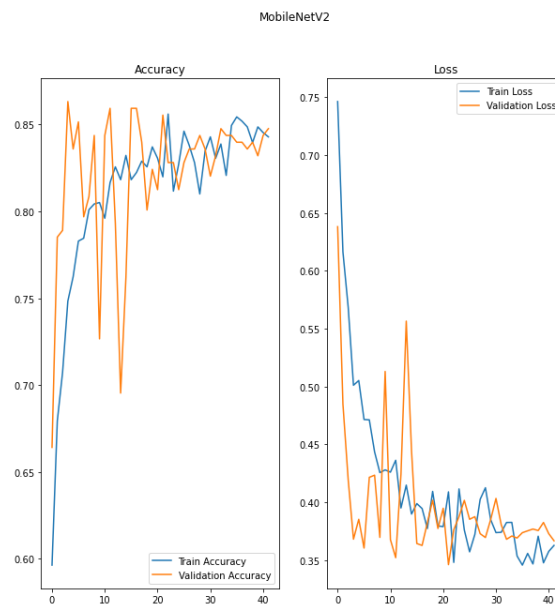
Model: "sequential_3"		
Layer (type)	Output Shape	Param #
=====		
sequential_2 (Sequential)	(None, 224, 224, 3)	0
densenet121 (Functional)	(None, 7, 7, 1024)	7037504
flatten_1 (Flatten)	(None, 50176)	0
dense_3 (Dense)	(None, 1024)	51381248
dropout_2 (Dropout)	(None, 1024)	0
dense_4 (Dense)	(None, 512)	524800
dropout_3 (Dropout)	(None, 512)	0
batch_normalization_1 (Batch Normalization)	(None, 512)	2048
dense_5 (Dense)	(None, 1)	513
=====		
Total params: 58,946,113		
Trainable params: 51,907,585		
Non-trainable params: 7,038,528		

Gambar 4.4. Model DenseNet121

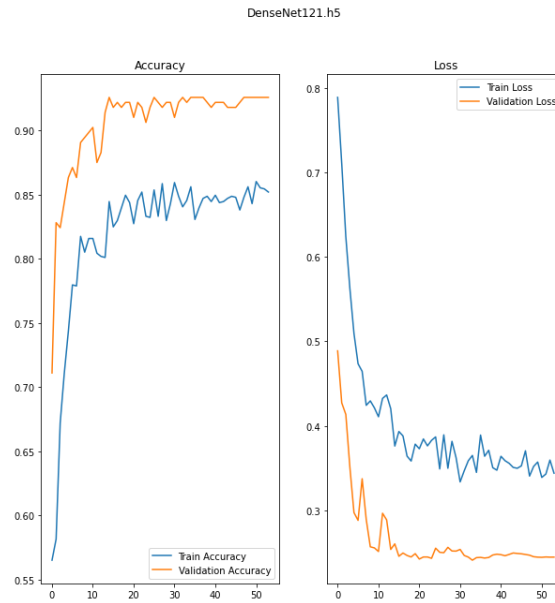
Selanjutnya dilakukan pelatihan dan pengujian tiap model dan hasil perhitungan akurasi serta *loss* tiap model divisualisasikan, seperti Gambar 4.5 sampai Gambar 4.7. Terlihat bahwa grafik akurasi ResNet101V2 selisih antara data latih dan visualisasi yang cukup jauh, sekitar 10%. Untuk model MobileNetV2 memiliki akurasi yang cukup mirip antara data latih dan data validasinya, begitu pula untuk *loss* yang dihasilkan. Terakhir, model DenseNet121 menjadi model dengan akurasi pada data latih terbaik dan memiliki selisih yang tidak terlalu jauh dengan akurasi data validasi.



Gambar 4.5. Evaluasi Model ResNet101V2



Gambar 4.6. Evaluasi Model MobileNetV2



Gambar 4.7. Evaluasi Model DenseNet121

Setiap model dibandingkan hasil evaluasinya dengan beberapa metrik evaluasi, seperti akurasi, *precision*, *recall*, dan *F1-score*. Perbandingan hasil evaluasi tiap model terlihat seperti yang ditampilkan pada Tabel 4.1. Dari tabel tersebut, terlihat bahwa *DenseNet121* memiliki akurasi terbaik dan memiliki metrik evaluasi lain yang terbaik juga apabila dibandingkan dengan *ResNet101V2* dan *MobileNetV2*.

Tabel 4.1. Perbandingan Metrik Evaluasi tiap Model

Model	Kelas	Akurasi	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<i>ResNet101V2</i>	Sehat	0,8945	0,91	0,87	0,89
	Tidak Sehat		0,88	0,92	0,90
<i>MobileNetV2</i>	Sehat	0,8555	0,89	0,80	0,84
	Tidak Sehat		0,83	0,91	0,87
<i>DenseNet121</i>	Sehat	0,9219	0,91	0,94	0,92
	Tidak Sehat		0,94	0,91	0,92

Dengan tingginya metrik evaluasi yang dihasilkan oleh model *DenseNet121*, kami menyarankan model ini untuk digunakan dalam memprediksi penyakit pada daun kopi robusta.

V. Kesimpulan

Dengan menggunakan model *Convolutional Neural Network* (CNN), dapat dibangun model untuk memprediksi kesehatan daun kopi jenis robusta. Dari model yang diuji, model *DenseNet121* menghasilkan akurasi terbaik dengan nilai 0,9219. Selain akurasi yang baik, metrik evaluasi lain seperti *precision*, *recall*, dan *F1-score* dari model *DenseNet121* juga yang tertinggi daripada model lain yang diuji. Sehingga, kedepannya dapat digunakan model *DenseNet121* untuk memprediksi kesehatan daun kopi jenis robusta untuk meningkatkan produksi kopi dan penghidupan petani.

Referensi

- [1] “Top 3 Reasons Why People Drink Coffee,” <https://premiumwaters.com/blog/reasons-why-people-drink-coffee/>.
- [2] M. L. V. de Resende, E. A. Pozza, T. Reichel, and D. M. S. Botelho, “Strategies for Coffee Leaf Rust Management in Organic Crop Systems,” *Agronomy*, vol. 11, no. 9, p. 1865, Sep. 2021, doi: 10.3390/agronomy11091865.
- [3] P. A. Arneson, “Coffee rust,” *The Plant Health Instructor*, 2000, doi: 10.1094/PHI-I-2000-0718-02.
- [4] A. P. Marcos, N. L. Silva Rodovalho, and A. R. Backes, “Coffee Leaf Rust Detection Using Genetic Algorithm,” in *2019 XV Workshop de Visão Computacional (WVC)*, IEEE, Sep. 2019, pp. 16–20. doi: 10.1109/WVC.2019.8876934.
- [5] G. L. Manso, H. Knidel, R. A. Krohling, and J. A. Ventura, “A smartphone application to detection and classification of coffee leaf miner and coffee leaf rust,” Mar. 2019.
- [6] M. A. N. Yasin and W. F. Al Maki, “Coffee Plant Disease Classification Using K-Nearest Neighbor,” in *2022 10th International Conference on Information and Communication Technology (ICoICT)*, IEEE, Aug. 2022, pp. 240–245. doi: 10.1109/ICoICT55009.2022.9914843.
- [7] K. J. Ayikpa, K. J. Ayikpa, K. J. Ayikpa, D. Mamadou, P. Gouton, and K. J. Adou, “Experimental Evaluation of Coffee Leaf Disease Classification and Recognition Based on Machine Learning and Deep Learning Algorithms,” *Journal of Computer Science*, vol. 18, no. 12, pp. 1201–1212, Dec. 2022, doi: 10.3844/jcssp.2022.1201.1212.
- [8] L. Falaschetti, L. Manoni, D. Di Leo, D. Pau, V. Tomaselli, and C. Turchetti, “A CNN-based image detector for plant leaf diseases classification,” *HardwareX*, vol. 12, p. e00363, Oct. 2022, doi: 10.1016/j.ohx.2022.e00363.
- [9] M. Kumar, P. Gupta, P. Madhav, and Sachin, “Disease Detection in Coffee Plants Using Convolutional Neural Network,” in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, Jun. 2020, pp. 755–760. doi: 10.1109/ICCES48766.2020.9138000.
- [10] B. Bertrand and B. Guyot, “Coffee Quality and its Biological Basis,” *Annu Rev Food Sci Technol*, vol. 12, no. 1, pp. 121–139, 2021.
- [11] R. M. Souza and E. L. Furtado, “Coffee Leaf Rust and Coffee Berry Disease: New Strategies for Integrated Control,” *Front Plant Sci*, vol. 8, 2017.
- [12] A. Ridhovan and A. Suharso, “PENERAPAN METODE RESIDUAL NETWORK (RESNET) DALAM KLASIFIKASI PENYAKIT PADA DAUN GANDUM,” *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 1, pp. 58–65, Feb. 2022, doi: 10.29100/jupi.v7i1.2410.
- [13] M. Cornelis, “The Impact of Caffeine and Coffee on Human Health,” *Nutrients*, vol. 11, no. 2, p. 416, Feb. 2019, doi: 10.3390/nu11020416.

- [14] M. E. Al Rivan and S. Hartoyo, “Klasifikasi Isyarat Bahasa Indonesia Menggunakan Metode Convolutional Neural Network,” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 2, Aug. 2022, doi: 10.28932/jutisi.v8i2.4863.
- [15] N. D. Miranda, L. Novamizanti, and S. Rizal, “CONVOLUTIONAL NEURAL NETWORK PADA KLASIFIKASI SIDIK JARI MENGGUNAKAN RESNET-50,” *Jurnal Teknik Informatika (Jutif)*, vol. 1, no. 2, pp. 61–68, Dec. 2020, doi: 10.20884/1.jutif.2020.1.2.18.
- [16] A. Setiawan, “Augmentasi Data pada Computer Vision,” <https://medium.com/data-folks-indonesia/augmentasi-data-pada-computer-vision-45c5ebe10e8f>.

Lampiran

Nama	NPM	Peran dalam Kelompok	Persentase Keaktifan
Muhammad Rayhan Nuansa Adha	2006571053	Melakukan <i>Modelling</i> di python, membuat Data, Metodologi, dan Kesimpulan, serta merapihkan makalah.	100%
Nadia Sukesi Sianipar	2106700776	Melakukan <i>Preprocessing</i> di python, membuat Abstrak, Pendahuluan, dan Tinjauan Pustaka, serta merapihkan makalah.	100%
Peksyaji	2006522322	Melakukan <i>Evaluation Model</i> di python, membuat Hasil dan Diskusi, serta merapihkan Daftar Pustaka.	100%