

ANALISIS EFEKTIFITAS ALGORITMA MOBILENETV3-LARGE DAN EFFICIENTNET-B0 UNTUK KLASIFIKASI CITRA PENYAKIT DAUN JERUK

Alfian Rachmad Dianto¹, Fetty Tri Anggraeny², Hendra Maulana³

^{1,2,3} Informatika, Uaniversitas Pembangunan Nasional Veteran Jawa Timur; Jl. Raya Rungkut Madya, Surabaya, Indonesia.

Keywords:

MobileNetV3-Large;
EfficientNet-B0, Klasifikasi,
Citra, Analisis

Correspondent Email:

20081010011@student.upnja
tim.ac.id



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstrak. Meningkatnya konsumsi jeruk (*Citrus spp.*) di masyarakat menjadikan budidaya jeruk sebagai komoditas penting, namun rentan terhadap serangan penyakit yang dapat menyebabkan gagal panen. Penelitian ini bertujuan untuk mengevaluasi kinerja dan efisiensi algoritma *MobileNetV3-Large* dan *Efficient Net-B0* dalam mengklasifikasi citra penyakit daun jeruk, baik dari data mandiri maupun data terbuka seperti *Kaggle*. Metode yang digunakan adalah pendekatan kuantitatif eksperimental dengan pengujian variasi *hyperparameter*, *optimizer*, dan skenario rasio data pelatihan, validasi, serta pengujian. Evaluasi model dilakukan menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa kedua arsitektur mampu melakukan klasifikasi citra secara efektif, dengan hasil terbaik diperoleh pada konfigurasi rasio data 70-20-10, *optimizer RMSprop*, dan *learning rate* 0,1 menggunakan *early stopping*. Konfigurasi alternatif yang juga direkomendasikan adalah rasio 60-30-10 dengan *optimizer Adam* dan *epoch* 15 atau 30. Temuan ini menunjukkan bahwa *MobileNetV3-Large* dan *EfficientNet-B0* dapat diandalkan untuk sistem klasifikasi penyakit daun jeruk berbasis *website* atau aplikasi *mobile*, terutama pada kondisi data terbatas dan distribusi kelas yang tidak seimbang

Abstract. The increasing consumption of citrus (*Citrus spp.*) in the community makes citrus cultivation an important commodity, but it is vulnerable to disease attacks that can cause crop failure. This study aims to evaluate the performance and efficiency of the *MobileNetV3-Large* and *EfficientNet-B0* algorithms in classifying citrus leaf disease images, both from independent data and open data such as *Kaggle*. The method used is an experimental quantitative approach by testing *hyperparameter* variations, *optimizers*, and training, validation, and testing data ratio scenarios. Model evaluation is done using *Accuracy*, *Precision*, *Recall*, and *F1-score* metrics. The results showed that both architectures were able to perform image classification effectively, with the best results obtained in the configuration of data ratio 70-20-10, *optimizer RMSprop*, and *learning rate* 0.1 using *early stopping*. An alternative configuration that is also recommended is the 60-30-10 ratio with the *Adam* optimizer and 15 or 30 epochs. These findings show that *MobileNetV3-Large* and *EfficientNet-B0* are reliable for web-based or mobile application-based citrus leaf disease classification systems, especially under conditions of limited data and unbalanced class distribution.

1. PENDAHULUAN

Jeruk (*Citrus spp.*) merupakan salah satu komoditas buah yang banyak dikonsumsi oleh masyarakat karena kandungan nutrisinya yang tinggi, terutama sebagai sumber vitamin C. Selain dikonsumsi secara langsung, jeruk juga dimanfaatkan sebagai bahan baku pembuatan parfum dan pengharum ruangan [1]. Berdasarkan data Badan Statistik Nasional (2023), Provinsi Jawa Timur tercatat sebagai produsen jeruk tertinggi di Indonesia, mengungguli 37 provinsi lainnya. Selain itu, jeruk menempati posisi ketiga sebagai komoditas buah dengan nilai produksi tertinggi setelah pisang dan mangga. Namun demikian, di tengah tren peningkatan produksi secara umum di Jawa Timur, masih terdapat beberapa daerah yang tercatat mengalami penurunan produksi pada tahun 2022.

Penurunan produksi jeruk dapat disebabkan oleh berbagai faktor seperti luas lahan, produktivitas panen, dan harga jual di pasar [2]. Salah satu tantangan utama dalam budidaya jeruk adalah serangan penyakit yang dapat menyebabkan kegagalan panen [3]. Gejala penyakit ini umumnya terlihat pada daun, berupa bercak-bercak berwarna kuning atau cokelat [1]. Oleh karena itu, identifikasi penyakit melalui citra daun dinilai sebagai metode yang cepat dan akurat karena adanya perbedaan karakteristik visual yang jelas antara daun yang sehat dan yang terinfeksi [4].

Seiring dengan perkembangan teknologi komputasi, khususnya *Artificial Intelligence* (AI) menjadi solusi potensial dalam mendeteksi penyakit tanaman berbasis citra daun. Dalam perkembangannya, bidang AI telah melahirkan cabang ilmu baru yaitu *Deep Learning*, yang memungkinkan pembelajaran fitur secara otomatis dan lebih akurat dari data citra yang kompleks [5]. Namun, masih terbatasnya pengujian algoritma *deep learning* yang kompatibel dengan perangkat *mobile* menjadi tantangan tersendiri [6]. Padahal, mayoritas petani mengandalkan perangkat *mobile* dalam kegiatan produksi. Oleh karena itu, pengembangan solusi berbasis *mobile* sangat diperlukan untuk meningkatkan efisiensi, efektivitas, produktivitas, dan kualitas hasil pertanian sekaligus mengurangi beban kerja petani [7].

Beberapa arsitektur *deep learning* telah dirancang untuk mendukung komputasi ringan

pada perangkat *mobile*, seperti *MobileNetV3-Large* dan *EfficientNet-B0*. Penelitian Pramuditha et al. (2023) mengembangkan sistem klasifikasi penyakit pada tanaman stroberi menggunakan algoritma *Convolutional Neural Network* (CNN) berbasis arsitektur *MobileNetV3-Large* dan *EfficientNet-B0*. Hasil pengujian yang dilakukan menunjukkan bahwa *MobileNet V3 Large* menghasilkan akurasi terbaik mencapai 92,14% dengan pengaturan *hyperparameter optimizer RMSProp*, *epochs* 70, dan *learning rate* 0,0001. Model ini juga mencapai *precision* 92,81%, *recall* 92,14%, dan *F1-Score* 92,25%, sehingga menunjukkan kinerja yang lebih unggul dibandingkan *Efficient Net-B0*.

Berdasarkan latar belakang yang telah diuraikan, penelitian ini berfokus pada pengujian perbandingan kinerja *MobileNet V3 Large* dan *Efficient Net-B0* dalam mengklasifikasikan citra penyakit daun jeruk. Penekanan utama penelitian terletak pada evaluasi efisiensi kedua arsitektur dalam konteks pengembangan aplikasi berbasis *mobile* untuk menunjang kegiatan produksi petani. Diharapkan hasil penelitian ini dapat memberikan kontribusi positif bagi sektor pertanian jeruk, khususnya di Provinsi Jawa Timur, dengan menghadirkan solusi berbasis *deep learning* yang akurat dan mudah diakses.

2. TINJAUAN PUSTAKA

2.1. Machine Learning

Teknologi *machine learning* merupakan alat bantu komputasional untuk mendeteksi objek secara otomatis [8]. *Machine learning* terdiri dari tiga cabang yang dibedakan berdasarkan tingkat kontrol manusia, yaitu *supervised learning* (pembelajaran diawasi), *unsupervised learning* (pembelajaran tanpa pengawasan), dan *reinforcement learning* (pembelajaran penguatan). Fokus pengembangannya dilakukan pada *supervised learning*, dimana proses klasifikasi merupakan salah satu contoh penerapannya [9].

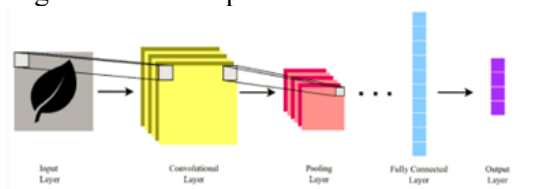
2.2. Deep Learning

Deep learning merupakan teknik lanjutan dalam pengembangan *machine learning* yang memanfaatkan algoritma yang meniru mekanisme kerja sistem saraf otak manusia [10]. Penerapan *deep learning*

difokuskan untuk mendukung otomatisasi klasifikasi objek pada citra yang sedang diuji. *Deep learning* merupakan bagian dari *machine learning*, sehingga teknologi ini memiliki kemampuan untuk mengenali pola gambar yang kompleks serta menghasilkan prediksi yang akurat berdasarkan data yang tersedia [11].

2.3. Convolutional Neural Network

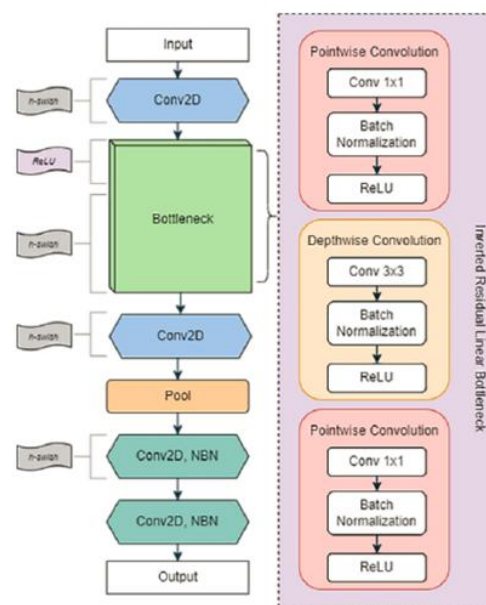
Convolutional Neural Network (CNN) adalah algoritma yang mampu secara otomatis mempelajari fitur-fitur dari data citra [12]. Meskipun proses otomatisasi ini berjalan secara mandiri, pengawasan manusia tetap diperlukan untuk melakukan evaluasi model agar memperoleh hasil yang optimal. Pembelajaran fitur citra dilakukan melalui proses ekstraksi menggunakan kernel berukuran tertentu untuk membentuk *convolutional layer* tanpa menghilangkan informasi penting dari data asli. Setelah *convolutional layer* terbentuk, data kemudian diproses melalui *pooling layer* guna menyederhanakan komputasi, dan diakhiri dengan *fully connected layer* yang berfungsi sebagai dasar dalam proses klasifikasi.



Gambar 1. Cara kerja CNN [6]

2.4. MobileNetV3-Large

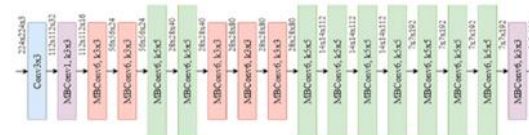
Mobile Net V3 merupakan arsitektur *deep learning* yang efisien dan dirancang untuk perangkat dengan sumber daya terbatas. *MobileNetV3* memiliki dua varian, yaitu *Mobile V3-Large* dan *MobileNetV3-Small*. Dalam penelitian ini, digunakan arsitektur *Mobile Net V3-Large* untuk tugas klasifikasi citra penyakit pada daun jeruk. Arsitektur ini dirancang melalui pendekatan *Network Architecture Search* (NAS), sebuah metode pencarian otomatis yang efisien dalam menentukan struktur jaringan optimal [13]. *Mobile Net V3-Large* juga mengintegrasikan beberapa teknik untuk meningkatkan efisiensi dan akurasi model, terutama pada perangkat dengan sumber daya terbatas.



Gambar 2. Ilustrasi Arsitektur *MobileNetV3-Large* [6]

2.5. *EfficientNet-B0*

Efficient Net merupakan salah satu arsitektur *Convolutional Neural Network* (CNN) yang dirancang secara inovatif untuk mengoptimalkan keseimbangan antara akurasi dan efisiensi melalui pendekatan *compound scaling* yaitu metode sistematis dalam penyesuaian tiga dimensi utama jaringan yaitu kedalaman (*depth*), lebar (*width*), dan resolusi input [14]. Versi dasarnya, yaitu *Efficient Net-B0*, menggunakan blok *Mobile Inverted Bottleneck Convolution* (MBConv), yang merupakan pengembangan dari arsitektur *Mobile Net V2*.



Gambar 3. Ilustrasi Arsitektur EfficientNet-B0
[6]

2.6. Matrik Evaluasi

Confusion matrix merupakan metode evaluasi yang digunakan untuk mengukur kinerja model klasifikasi dengan membandingkan hasil prediksi dan data aktual. matriks ini terdiri dari empat komponen utama: *True True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN),

yang masing-masing merepresentasikan keberhasilan dan kesalahan prediksi model dalam mengklasifikasikan data. Beberapa metrik turunan yang umum digunakan meliputi *accuracy*, yaitu proporsi prediksi benar dari seluruh data; *precision*, yang mengukur ketepatan model dalam memprediksi kelas positif; serta *recall*, yang menunjukkan seberapa baik model dalam mengenali seluruh data positif. Untuk menyeimbangkan *precision* dan *recall*, digunakan *F1-score* sebagai metrik gabungan yang bermanfaat terutama saat data tidak seimbang [15].

3. METODE PENELITIAN

Penelitian ini mengadopsi pendekatan kuantitatif eksperimental dengan tahapan sistematis, meliputi (1) pengumpulan data sekunder dari Badan Pusat Statistik Nasional dan provinsi Jawa Timur, (2) pemahaman dan penyusunan studi literatur dengan kriteria jurnal terakreditasi *scopus* atau minimal SINTA 2, (3) proses pencarian data, (4) persiapan data set, serta (5) merancang model klasifikasi, (6) penyusunan dan pengujian model skenario, dan (7) evaluasi model klasifikasi. Desain sistem klasifikasi penelitian ini dimulai dengan pengumpulan data dari dataset melalui *resizing* (224x224 piksel) dan data *augmentation* (rotasi dan *flipping*). Selanjutnya, dilakukan perancangan model klasifikasi menggunakan arsitektur *MobileNetV3-Large* dan *EfficientNet-B0*. Model diuji dengan berbagai skenario melalui perubahan *hyperparameter* dan *optimizer*. Tahap akhir berupa evaluasi performa model menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-score* untuk menilai efektivitas deteksi penyakit daun jeruk seperti pada gambar.



Gambar 3. 1 Metode Penelitian

4. HASIL DAN PEMBAHASAN

4.1. Data Asli Dengan Algoritma *MobileNetV3-Large*

Hasil pengujian pada lima *batch* menunjukkan bahwa tidak seluruh skenario mencapai jumlah *epoch* maksimum yang telah ditetapkan, meskipun beberapa skenario berhasil mencapainya. Kombinasi paling optimal pada tahap ini ditemukan pada *batch* pertama, yaitu dengan penggunaan *optimizer Adam* dan *learning rate* 0,1 yang menghasilkan akurasi sebesar 97,56% dan *F1-score* 97,6%. Kombinasi ini menggunakan rasio pembagian data 70% untuk pelatihan, 20% untuk validasi, dan 10% untuk pengujian, serta membutuhkan waktu komputasi sebesar 911,63 detik (sekitar 15,19 menit), dengan capaian *epoch* maksimum sebanyak 30. Untuk memastikan validitas hasil, visualisasi menggunakan *confusion matrix* dilakukan dan menunjukkan bahwa hampir seluruh prediksi sesuai dengan label aktual, hanya terdapat satu kesalahan klasifikasi, sehingga mendukung konsistensi antara hasil evaluasi dan kurva pelatihan.

Pada analisis distribusi kombinasi 10% teratas, sebagian besar konfigurasi menggunakan rasio data 60-30-10, yang mengindikasikan bahwa performa terbaik dapat dicapai meskipun proporsi data latih lebih kecil asalkan didukung oleh porsi validasi yang memadai. Selain itu, *learning rate* 0,1 mendominasi kombinasi terbaik, yang menunjukkan efektivitasnya dalam mempercepat proses pelatihan, khususnya pada arsitektur *MobileNetV3-Large*. Dari sisi *optimizer*, *Adam* digunakan pada enam kombinasi terbaik, diikuti oleh *RMSprop* pada lima kombinasi, sedangkan SGD hanya muncul pada satu konfigurasi. Hal ini menunjukkan bahwa pada tahap ini, *Adam* dan *RMSprop* merupakan *optimizer* yang paling konsisten dalam menghasilkan performa model yang optimal.

4.2. Data Asli Dengan Algoritma *EfficientNet-B0*

Pengujian menggunakan algoritma *EfficientNet-B0* pada data asli menunjukkan bahwa sebagian besar skenario menghasilkan nilai akurasi dan metrik evaluasi yang maksimal, meskipun tidak ada satupun percobaan yang mencapai batas *epoch*

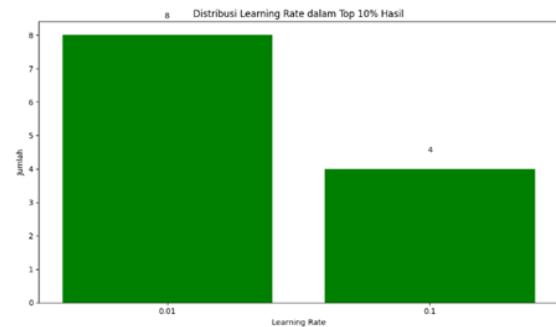
maksimum yang telah ditentukan. Hal ini menunjukkan efektivitas penerapan *early stopping* dalam mencegah *overfitting* atau *underfitting* ketika validasi loss tidak mengalami perbaikan lebih lanjut. Kombinasi terbaik dicapai pada skenario dengan rasio pembagian data 70% latih, 20% validasi, dan 10% uji, dengan optimizer *RMSprop*, learning rate 0,1, dan jumlah *epoch* aktual sebanyak 6 dari maksimum 15, menghasilkan akurasi, *precision*, *recall*, dan *F1-score* masing-masing sebesar 100%, dengan waktu komputasi sekitar 213 detik. Visualisasi melalui *confusion matrix* menunjukkan akurasi prediksi yang sangat tinggi, memperkuat hasil dari metrik evaluasi.

Distribusi kombinasi optimal 10% teratas didominasi oleh rasio pembagian data 60-30-10 sebanyak tujuh kombinasi, dan sisanya menggunakan rasio 70-20-10, menandakan bahwa proporsi data validasi yang lebih besar dapat memberikan kontribusi positif terhadap stabilitas pelatihan model. Pada aspek learning rate, nilai 0,1 digunakan pada delapan dari sepuluh kombinasi teratas, menunjukkan bahwa learning rate yang relatif tinggi tetap efektif jika didukung data dan konfigurasi yang sesuai. Dalam hal *optimizer Adam* menunjukkan performa tertinggi dengan tujuh kombinasi optimal, diikuti *RMSprop* dengan lima kombinasi, sedangkan SGD menunjukkan performa yang kurang kompetitif. Hasil ini menegaskan bahwa *optimizer Adam* dan *RMSprop* merupakan *optimizer* yang paling sesuai untuk skenario klasifikasi citra penyakit daun jeruk menggunakan *EfficientNet-B0* pada data asli.

4.3. Data Kaggle Dengan Algoritma *MobileNetV3-Large*

Pada tahap pengujian dengan algoritma *MobileNetV3-Large* menggunakan data dari Kaggle, diperoleh hasil bahwa kombinasi rasio data 70% pelatihan, 20% validasi, dan 10% pengujian serta learning rate sebesar 0,1 tetap menjadi konfigurasi yang menghasilkan performa optimal. Beberapa skenario berhasil mencapai jumlah *epoch* maksimal tanpa memerlukan mekanisme *early stopping*, menandakan kestabilan pelatihan pada konfigurasi tertentu. Salah satu skenario terbaik menghasilkan akurasi dan *F1-score* di atas 96%, meskipun kurva pembelajarannya

menunjukkan fluktuasi yang mengindikasikan potensi *overfitting* jika pelatihan dilanjutkan terlalu lama. Namun, visualisasi melalui *confusion matrix* menguatkan bahwa model tetap mampu mengklasifikasikan citra dengan tingkat kesalahan yang sangat rendah.



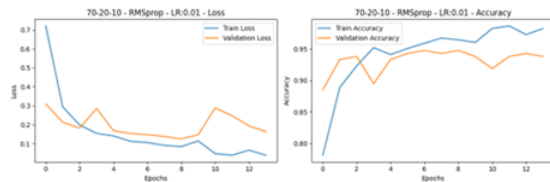
Gambar 1. Distribusi *Learning Rate* Pada Data Kaggle dengan *MobileNetV3-Large*

Distribusi dari 10% kombinasi terbaik menunjukkan keseimbangan antara penggunaan rasio 70-20-10 dan 60-30-10. Perbedaan signifikan muncul pada learning rate, di mana nilai 0,01 lebih sering muncul pada kombinasi terbaik dibandingkan 0,1, menunjukkan bahwa variasi data mempengaruhi sensitivitas terhadap parameter ini. Selain itu, dari sisi optimizer, Adam dan SGD masing-masing digunakan dalam lima kombinasi optimal, sedangkan *RMSprop* hanya dalam dua, yang menunjukkan bahwa efektivitas optimizer sangat bergantung pada karakteristik dataset yang digunakan.

4.4. Data Kaggle Dengan Algoritma *EfficientNet-B0*

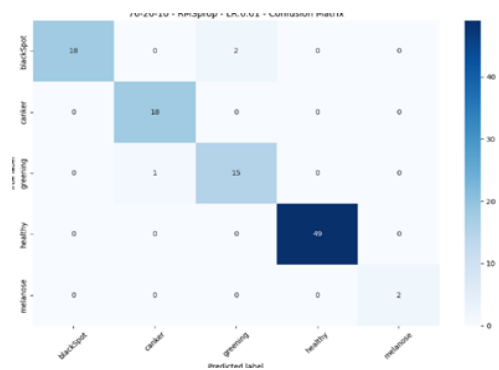
Hasil optimal dari pengujian algoritma *EfficientNet-B0* terhadap data Kaggle menunjukkan di mana kombinasi paling optimal ditemukan pada batch ke-3. Dalam konteks penelitian, proses pengujian algoritma memerlukan pengulangan skenario secara sistematis agar dapat memperoleh hasil yang benar-benar representatif serta memahami dampak dari setiap variabel yang diubah. Temuan pada batch ke-3 menunjukkan performa terbaik, meskipun pada tahap-tahap sebelumnya performa optimal justru diperoleh dari batch ke-1 dan ke-2. Hal ini menegaskan pentingnya jumlah uji coba yang memadai, karena jika jumlah uji terbatas, maka hasil akhir

berpotensi mengandung bias saat diimplementasikan secara nyata.



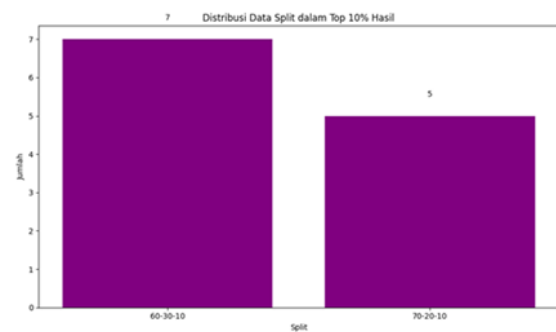
Gambar 2. Kurva Pembelajaran Model Teratas Data Kaggle dengan *EfficientNet-B0*

Kurva pembelajaran model pada Gambar 2 menunjukkan bahwa pelatihan dihentikan pada *epoch* ke-14 akibat aktivasi fungsi *early stopping*. Hal ini disebabkan oleh nilai *validation loss* yang tidak menunjukkan perbaikan signifikan setelah mengalami lonjakan, meskipun train loss terus menurun. Maka dari itu, untuk mencegah terjadinya *overfitting*, pelatihan dihentikan lebih awal karena model telah menunjukkan akurasi yang tinggi.



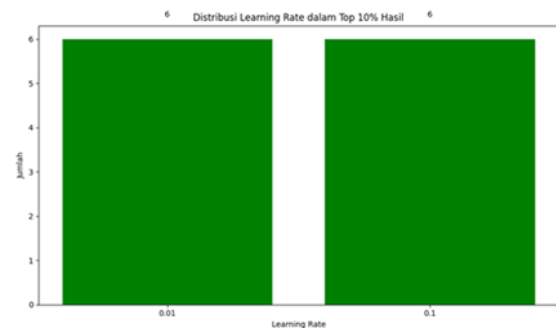
Gambar 3. *Confusion Matriks* Model Teratas Data Kaggle dengan *EfficientNet-B0*

Berdasarkan visualisasi hasil prediksi model optimal melalui confusion matrix pada Gambar 3, diperoleh performa klasifikasi yang baik. Meskipun demikian, terjadi ketidakseimbangan dalam prediksi pada kelas *healthy* dan *melanose*, yang kemungkinan besar disebabkan oleh distribusi data yang tidak seimbang antar kelas. Akibatnya, model cenderung lebih sering memprediksi ke salah satu kelas dominan, sehingga menimbulkan potensi bias klasifikasi.



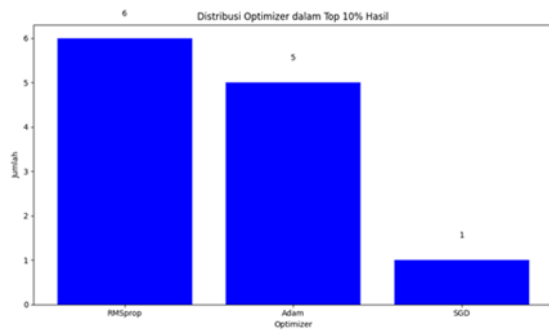
Gambar 4. Distribusi Rasio Data Pada Data Kaggle dengan *EfficientNet-B0*

Gambar 4 menunjukkan distribusi rasio pembagian data pada 10% kombinasi skenario paling optimal. Diketahui bahwa rasio 60% data latih, 30% data validasi, dan 10% data uji digunakan sebanyak tujuh kali, lebih tinggi dibandingkan rasio 70%-20%-10% yang hanya muncul lima kali. Hal ini menunjukkan bahwa model mampu belajar secara efektif dengan proporsi data latih yang lebih rendah selama jumlah data keseluruhan mencukupi dan distribusi antar kelas memiliki perbedaan yang signifikan.



Gambar 5. Distribusi *Learning Rate* Pada Data Kaggle dengan *EfficientNet-B0*

Distribusi penggunaan *learning rate* ditunjukkan pada Gambar 5, yang memperlihatkan keseimbangan penggunaan antara nilai 0,1 dan 0,01, masing-masing sebanyak enam kombinasi. Ini mengindikasikan bahwa kedua nilai *learning rate* tersebut mampu menghasilkan performa optimal pada tahap ini dan sesuai dengan karakteristik data serta algoritma yang digunakan.



Gambar 6. Distribusi Optimizer Pada Data Kaggle dengan *EfficientNet-B0*

Gambar 6 menampilkan distribusi penggunaan *optimizer*, di mana algoritma RMSprop dan Adam mendominasi penggunaan dalam 10% kombinasi terbaik, sementara SGD menunjukkan frekuensi penggunaan yang rendah. Temuan ini konsisten dengan hasil pada tahap-tahap sebelumnya

4.5. Evaluasi Perbandingan

Tabel 1. Perbandingan Kombinasi dari Model paling Optimal

No	Batch	Jenis Data	Algoritma	Pengujian Skenario	Top Kombinasi (Rasio Data, Optimizer, Learning Rate, Max Epoch)
1	1	Data Asli	MobileNet V3-Large	Skenario ke-1 dari tabel 4.1, skenario ke-2 dari tabel 4.2, skenario ke-3 dari tabel 4.4, skenario ke-4 dari tabel 4.5.	70-20-10-Adam-0,1-30
	2				
	3				
	4				
	5				
2	1	Data Kaggle	EfficientNet-B0		70-20-10-RMSprop-0,1-15 (<i>early stopping</i> aktif)
	2				
	3				
	4				
	5				
3	1	Data Kaggle	MobileNet V3-Large		70-20-10-RMSprop-0,1-15
	2				
	3				
	4				
	5				
4	1	Data Kaggle	EfficientNet-B0		70-20-10-RMSprop-0,01-30 (<i>early stopping</i> aktif)
	2				
	3				
	4				
	5				

Berdasarkan Tabel 1 dapat dijelaskan bahwa rasio pembagian data sebesar 70% untuk data latih, 20% untuk data validasi, dan 10% untuk data uji merupakan kombinasi yang paling optimal, karena seluruh model mencapai kinerja terbaik pada konfigurasi tersebut. Dari sisi optimizer, sebagian besar model menunjukkan performa terbaik ketika menggunakan RMSprop. Selain itu, nilai learning rate sebesar 0,1 secara umum menghasilkan hasil yang optimal. Sementara itu, variasi jumlah epoch menunjukkan kelebihan dan kekurangan masing-masing akibat pengaruh mekanisme *early stopping*. Oleh karena itu, berdasarkan hasil evaluasi tersebut, dilakukan penyusunan kombinasi parameter yang direkomendasikan pada Tabel 2.

Tabel 2. Kombinasi Baru Berdasarkan Top Kombinasi

No	Rasio Pembagian Data	Optimizer	Learning Rate	Max Epoch
1	70% data latih, 20% data validasi, 10% data uji	RMSprop	0,1	15 atau 30

Selain mempertimbangkan kombinasi dengan performa tertinggi, analisis distribusi kombinasi hyperparameter pada 10% teratas hasil pengujian juga dilakukan, sebagaimana disajikan dalam Tabel 2.

Tabel 3. Distribusi Tertinggi Rasio Data, *Learning Rate, Optimizer*

No	Jenis Data	Algoritma	Distribusi Rasio Data Tertinggi	Distribusi Learning Rate Tertinggi	Distribusi Optimizer Tertinggi
1	Data Asli	MobileNetV3-Large	70-20-10	0,1	Adam
2		EfficientNet-B0	60-30-10	0,1	Adam
3	Data Kaggle	MobileNetV3-Large	Seimbangan	0,01	Adam/SGD
4		EfficientNet-B0	60-30-10	Seimbangan	RMSprop

Dari distribusi tersebut, dilakukan penyusunan konfigurasi baru yang ditampilkan pada Tabel 3, yakni pembagian data sebesar 60% untuk latih, 30% untuk validasi, dan 10%

untuk uji, dengan optimizer Adam, learning rate 0.1, serta jumlah epoch sebanyak 15 atau 30. Penggunaan early stopping tetap disarankan guna menghindari risiko overfitting maupun underfitting pada proses pelatihan model.

Tabel 4. Kombinasi Baru Berdasarkan Distribusi *Hyperparameter*

No	Rasio Pembagian Data	Optimizer	Learning Rate	Max Epoch
1	60% data latih, 30% data validasi, 10% data uji	Adam	0,1	15 atau 30

5. KESIMPULAN

Berdasarkan hasil penelitian yang melibatkan pengujian pada dua jenis data, yaitu data asli yang diperoleh secara mandiri dan data sekunder dari *platform Kaggle*, serta analisis dua algoritma klasifikasi yaitu *MobileNetV3-Large* dan *EfficientNet-B0*, diperoleh kesimpulan bahwa kedua algoritma tersebut menunjukkan kinerja yang baik dalam klasifikasi penyakit pada daun jeruk, dengan mayoritas nilai metrik evaluasi, termasuk akurasi, berada di atas 90%, yang juga didukung oleh visualisasi *confusion matrix*. Pengujian dilakukan melalui 24 skenario yang diulang sebanyak lima kali sehingga total mencapai 120 eksperimen untuk setiap kombinasi data dan algoritma, dengan optimalisasi pelatihan menggunakan *pre-trained model* dan fungsi *early stopping*. Penelitian ini menghasilkan kombinasi *hyperparameter* optimal berdasarkan distribusi 10% tertinggi dari skenario yang diuji, yang menunjukkan efisiensi dalam pengolahan data dengan jumlah kelas yang tidak seimbang. Disarankan agar penggunaan kombinasi ini tetap dilengkapi dengan *pre-trained model* dan *early stopping* agar tetap relevan saat diaplikasikan pada jenis citra lain, dengan catatan memperhatikan karakteristik data yang serupa dengan studi ini. Untuk pengembangan selanjutnya, direkomendasikan pembuatan aplikasi berbasis mobile yang memanfaatkan salah satu dari kedua algoritma tersebut, serta perluasan pengujian dengan dataset dari tanaman lain atau kondisi data yang berbeda namun terbatas, guna memperkaya validitas dan cakupan penelitian

serta menambah kontribusi ilmiah di bidang klasifikasi citra.

UCAPAN TERIMA KASIH

Penulis mengucapkan banyak terima kasih kepada orang tua, dosen pembimbing serta dosen penguji dan orang-orang terdekat penulis yang selalu mendukung secara moril atau materil untuk penyelesaian penelitian ini.

DAFTAR PUSTAKA

- [1] F. R. Lestari, I. Purwanti, N. Purnama, A. M. Sajiah, L. B. Aksara, dan J. T. Informatika, "IDENTIFIKASI PENYAKIT TANAMAN JERUK SIAM MENGGUNAKAN METODE M-SVM," 2019.
- [2] A. Setiadi, A. Gafaruddin, dan A. Slamet, "Faktor-Faktor yang Mempengaruhi Produksi Usahatani Jeruk Manis Siam Madu (*Citrus sinensis nobilis*) di Desa Tanea Kecamatan Konda Kabupaten Konawe Selatan," *Jurnal Ilmiah Membangun Desa dan Pertanian*, vol. 8, no. 3, hlm. 88–95, Mei 2023, doi: 10.37149/jimdp.v8i3.79.
- [3] F. Rahayu Lestari, J. Yusmah Sari, dan I. Purwanti Ningrum Purnama, DETEKSI PENYAKIT TANAMAN JERUK SIAM BERDASARKAN CITRA DAUN MENGGUNAKAN SEGMENTASI WARNA RGB-HSV. 2018. [Daring]. Tersedia pada: www.ti.eng.uho.ac.id
- [4] A. Setiyorini dan J. Yusmah Sari, "Perbaikan Kualitas Citra Untuk Klasifikasi Daun Menggunakan Metode Fuzzy K-Nearest Neighbor," *ULTIMATICS*, vol. IX, no. 2, hlm. 129, 2017, [Daring]. Tersedia pada: www.theplantlist.org,
- [5] Ulfah Nur Oktaviana, Ricky Hendrawan, Alfian Dwi Khoirul Annas, dan Galih Wasis Wicaksono, "Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model Terlatih Resnet101," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, hlm. 1216–1222, Des 2021, doi: 10.29207/resti.v5i6.3607.
- [6] D. A. Pramudhita, F. Azzahra, K. Arfat, R. Magdalena, dan S. Saidah, "Strawberry Plant Diseases Classification Using CNN Based on MobileNetV3-Large and EfficientNet-B0 Architecture ARTICLE INFO ABSTRACT," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, vol. 9, no. 3, hlm. 522–534, 2023, doi: 10.26555/jiteki.v9i3.26341.
- [7] B. U. Rusydi dan M. Rusli, "PEMANFAATAN TEKNOLOGI PERTANIAN DAN PENGARUHNYA TERHADAP

- PENDAPATAN PETANI,” *Journal of Regional Economics*, 2022.
- [8] D. Immanuel Salintohe, I. Alwiah Musdar, T. Informatika, dan S. Kharisma Makassar, “IMPLEMENTASI MACHINE LEARNING UNTUK MENGIDENTIFIKASI TANAMAN HIAS PADA APLIKASI TIERRA,” *JTRISTE*, vol. 9, no. 1, hlm. 1–15, 2022.
- [9] N. Faulina, “Implementation Of Artificial Neural Network (ANN) Classification In Type 2 Diabetes Mellitus Cases,” *Sciencestatistics Journal of Statistics, Probability, and Its Application*, vol. 2, no. 2, hlm. 80–92, 2024, [Daring]. Tersedia pada: <https://scholar.ummetro.ac.id/index.php/scie ncestatistics/index>
- [10] F. Fitra Maulana dan N. Rochmawati, “Klasifikasi Citra Buah Menggunakan Convolutional Neural Network,” *JINACS - (Journal of Informatics and Computer Science)*, 2019.
- [11] K. S. Dewi Linda dan A. Dwi Hartanto, “Studi Literatur Mengenai Klasifikasi Citra Kucing Dengan Menggunakan Deep Learning: Convolutional Neural Network (CNN),” *Journal homepage: Journal of Electrical Engineering and Computer (JEECOM)*, vol. xx, No. xx, 2023, doi: 10.33650/jeecom.v4i2.
- [12] S. F. Chairunissa, “Klasifikasi Jenis Lebah Madu Menggunakan Metode Convolutional Neural Network,” *JITTER- Jurnal Ilmiah Teknologi dan Komputer*, vol. 4, no. 3, 2023.
- [13] A. Howard dkk., “Searching for MobileNetV3,” dalam *Searching for MobileNetV3*, 2019.
- [14] R. Retno Utami, U. Fadzilia Arifin, dan N. Putri, “The Application of Artificial Intelligence in Waste Classification as an Effort In Plastic Waste Management,” *Jurnal Informatika dan Teknologi Informasi*, vol. 21, no. 1, hlm. 1–13, 2024, doi:10.31515/telematika.v21i1.11977.
- [15] M. D. Purbolaksono, M. Irvan Tantowi, A. Imam Hidayat, dan A. Adiwijaya, “Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, hlm. 393–399, Apr 2021, doi: 10.29207/resti.v5i2.3008.