

Klasifikasi Penyakit Daun Tomat Dengan Metode Convolutional Neural Network Menggunakan Arsitektur Inception-V3

Andi Nurdin¹, Dhian Satria Yudha Kartika², Abdul Rezha Efrat Najaf³

^{1,2,3}Universitas Pembangunan Nasional "Veteran" Jawa Timur, Jl. Rungkut Madya No.1, Gn. Anyar, Surabaya 6029, Indonesia

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 14-07-2024

Revisi Akhir: 20-08-2024

Diterbitkan Online: 05-09-2024

KATA KUNCI

Daun Tomat

Convolutional Neural Network

Klasifikasi

Inception V3

KORSPONDENSI

E-mail: andinuridin.ndyy@gmail.com

ABSTRACT

Tomato is an important food crop in the world, including Indonesia, which often faces leaf diseases such as mosaic virus, target spot, bacterial spot, yellow leaf curl virus, late blight, leaf mold, early blight, spider mites, and septoria leaf spot. These diseases are difficult to recognize manually, so Deep Learning technology, especially Convolutional Neural Network (CNN), is used for the classification process. This research uses Inception V3 transfer learning model and image preprocessing and data augmentation. Testing is done with various optimizers (Adam, SGD, RMS Prop) to find the best model. Tests were conducted using confusion matrix to determine the level of accuracy produced by the CNN model. The results show an accuracy of 93.8% with the optimal model using Optimizer Adam.

1. PENDAHULUAN

Tanaman tomat adalah salah satu komoditas *hortikultura* yang memiliki nilai ekonomi tinggi dan banyak dikonsumsi masyarakat Indonesia. Produksi tomat pada tahun 2021 sebesar 1,11 juta ton, tetapi pada tahun 2022 meningkat sebanyak 2.454 ton, atau 0,21%, menjadi 1,12 juta ton, berdasarkan data yang dirilis oleh BPS [1]. Salah satu faktor mengapa tomat masih menjadi komoditas panen terbesar di Indonesia adalah iklimnya yang ideal untuk budidaya tomat. Tanaman tomat cocok tumbuh baik di dataran rendah maupun dataran tinggi Indonesia.

Dalam budidaya tomat, masalah seperti hama dan penyakit sering terjadi, yang dapat mengurangi kualitas dan jumlah hasil panen. *Patogen* atau *mikroorganisme parasit*, inang, dan lingkungan adalah tiga penyebab penyakit tanaman. Ketika inang tanaman lemah, *patogen* di sekitarnya bersifat *virulen* atau memiliki daya infeksi tinggi, dan kondisi lingkungan yang mendukung penyebaran penyakit memungkinkan tanaman untuk terserang penyakit. Bahkan kerugian gagal panen dapat menyebabkan tomat menjadi turun sebesar 50% [2]. Kegagalan pada tanaman tomat tampak pada tomat yang mulai layu dan menghitam ketika

terkena hama. Serangan hama yang menyerang tanaman tomat dapat menghambat pertumbuhan tanaman. Jika hama tidak segera ditangani, akan menyebabkan kerugian yang lebih besar karena kerusakan yang ditimbulkan lebih parah [3].

Pengetahuan tentang penyakit tanaman dan metode penanggulangannya diperlukan untuk mengendalikan penyakit atau digunakan untuk penelitian terkait penyakit tanaman. Para petani di Indonesia rata-rata menghabiskan 40% biaya pengendalian hama dan penyakit. Petani biasanya menggunakan *pestisida* seperti *insektisida*, *fungisida*, dan *bakterisida* secara bersamaan dan berulang kali dalam jangka waktu yang lama [4]. Salah satu cara untuk mengendalikan hama dan penyakit tomat adalah dengan mendeteksi dan menentukan penyakit secara dini.

Teknologi sekarang telah mencapai kemampuan untuk mendeteksi bentuk dan warna daun dengan canggih. Dengan *machine learning*, teknologi tersebut dapat mengelompokkan dan memprediksi data. *Machine learning* sendiri merupakan cabang dari kecerdasan buatan atau *artificial intelligence* yang melibatkan penggunaan model matematika pada sampel data, yang dikenal sebagai data latih, untuk membuat prediksi atau mengambil keputusan dalam berbagai tugas. Dalam beberapa tahun terakhir, metode *Convolutional Neural Network* (CNN)

telah muncul sebagai solusi untuk mengatasi tantangan klasifikasi gambar. *Convolutional Neural Network* (CNN) adalah jaringan saraf tiruan yang terinspirasi dari korteks visual hewan yang dapat mengidentifikasi pola kompleks dalam data visual dengan efisiensi tinggi. Keunggulan *Convolutional Neural Network* (CNN) terletak pada kemampuannya untuk mempelajari fitur-fitur penting dari data secara otomatis melalui proses konvolusi, yang memungkinkan pengenalan pola yang lebih akurat dan efisien dibandingkan dengan metode tradisional [5]. Selain itu, CNN memiliki struktur yang mendalam dan *hierarkis*, yang memfasilitasi pembelajaran fitur dari tingkat rendah hingga tingkat tinggi, serta penggunaan parameter yang lebih sedikit yang mengurangi risiko *overfitting*. CNN juga menunjukkan adaptabilitas yang luar biasa dalam berbagai kondisi dan aplikasi, menjadikannya pilihan yang tepat untuk penelitian yang membutuhkan analisis citra yang kompleks dan variatif [6].

Transfer learning adalah sebuah teknik dalam *machine learning* yang memungkinkan peningkatan kinerja pada tugas tertentu dengan memanfaatkan pengetahuan yang telah diperoleh dari model yang sebelumnya dilatih pada tugas atau domain yang berbeda. Dalam implementasi *transfer learning* pada *Convolutional Neural Network* (CNN), lapisan-lapisan fitur yang telah dipelajari oleh satu CNN dapat digunakan kembali untuk menginisialisasi jaringan CNN lainnya. Teknik ini terbukti memiliki dampak signifikan dalam meningkatkan kinerja pada tugas target dan telah menjadi alat penting dalam berbagai aplikasi [7]. Proses *transfer learning* antar CNN biasanya dikenal dengan istilah *fine-tuning*, di mana lapisan-lapisan fitur yang ditransplantasikan dari CNN sumber hanya disesuaikan dengan menggunakan data target. *Fine-tuning* bertujuan untuk menyempurnakan lapisan fitur yang telah dipelajari oleh CNN sumber agar lebih cocok dengan tugas target. Dalam *fine-tuning* lapisan fitur pada CNN sumber dapat digunakan secara keseluruhan atau hanya sebagian, tergantung pada kecocokan dengan tugas target tersebut. Setelah itu, lapisan-lapisan ini akan disesuaikan dan diperbarui dengan menggunakan dataset gambar yang baru pada tugas klasifikasi yang berbeda [8]. Dengan memanfaatkan pengetahuan yang telah diperoleh sebelumnya, *transfer learning* dapat membantu meningkatkan kinerja tugas target pada CNN yang baru.

Berdasarkan uraian di atas, penelitian ini bertujuan untuk mengembangkan sistem pengenalan otomatis yang dapat mendeteksi dan mendiagnosis penyakit tomat berdasarkan citra digital daun tomat menggunakan metode *Convolutional Neural Network* (CNN). Pemilihan metode *Convolutional Neural Network* (CNN) dalam penelitian ini adalah karena kemampuannya yang telah terbukti dalam berbagai studi untuk menghasilkan hasil yang unggul dalam tugas-tugas seperti klasifikasi gambar, deteksi objek, dan segmentasi. Penelitian ini menggunakan data citra daun tomat yang terdiri dari 10 kelas data gambar, yaitu *mosaic virus*, *target spot*, *bacterial spot*, *yellow leaf curl virus*, *late blight*, *leaf mold*, *early blight*, *spider mites*, *septoria leaf spot*, dan daun tomat sehat. Pengujian dilakukan dengan menggunakan *confusion matrix* dengan nilai yang akan diukur yaitu akurasi, presisi, *recall*, dan *f1-score*. Dalam penelitian ini akan membandingkan tiga jenis algoritma *optimizer* yaitu Adam, SGD, dan RMSProp. Penelitian ini diharapkan dapat memberikan manfaat bagi petani tomat, peneliti, dan masyarakat umum dalam hal pengendalian penyakit tomat.

2. TINJAUAN PUSTAKA

2.1. *Convolutional Neural Network* (CNN)

Convolutional Neural Network (CNN) adalah jenis jaringan saraf tiruan yang dirancang khusus untuk mengolah data yang memiliki struktur grid, seperti gambar. CNN meniru cara otak manusia memproses informasi visual, membuatnya sangat efektif dalam mengenali pola dan fitur dalam gambar [9].

Arsitektur CNN terdiri dari beberapa lapisan utama, yaitu lapisan konvolusi, lapisan *pooling*, dan lapisan *fully connected*. Lapisan konvolusi bertugas untuk mendeteksi fitur-fitur penting dalam gambar melalui filter yang bergerak melintasi gambar, menghasilkan peta fitur yang menunjukkan kehadiran fitur tersebut. Lapisan *pooling* berfungsi untuk mengurangi dimensi data dan mengurangi kompleksitas perhitungan dengan mengambil nilai maksimum atau rata-rata dari bagian kecil peta fitur. Akhirnya, lapisan *fully connected* bertindak sebagai pengklasifikasi yang menggabungkan semua informasi yang diekstraksi untuk menghasilkan output akhir [10].

Convolutional Neural Network (CNN) telah banyak digunakan dalam berbagai aplikasi, seperti pengenalan wajah, klasifikasi objek, dan analisis citra medis. Keunggulan CNN terletak pada kemampuannya untuk belajar dan mengenali fitur-fitur yang kompleks secara otomatis dari data gambar yang besar [11].

2.2. *Confusion Matrix*

Confusion matrix adalah alat penting dalam evaluasi model klasifikasi dalam pembelajaran mesin. Matriks ini menampilkan ringkasan prediksi model terhadap label aktual dalam berbagai kelas atau kategori. Dalam matriks dua dimensi ini, setiap baris mewakili contoh dalam kelas aktual, dan setiap kolom mewakili contoh dalam kelas yang diprediksi. Elemen diagonal matriks mewakili prediksi yang benar, sementara elemen di luar diagonal menunjukkan kesalahan klasifikasi. Dengan menganalisis nilai-nilai seperti *true positive*, *false positive*, *true negative*, dan *false negative* dalam matriks, praktisi dapat memperoleh wawasan mendalam tentang performa model, termasuk kekuatan dan kelemahannya [12].

Tabel 1. Confusion Matrix

		Nilai Aktual	
		Label <i>Positive</i> (1)	<i>Negative</i> (0)
Nilai Prediksi	<i>Positive</i> (1)	<i>True Positive</i> (TP)	<i>False Positive</i> (FP)
	<i>Negative</i> (0)	<i>False Negative</i> (FN)	<i>True Negative</i> (TN)

Melalui Tabel 1, bisa didapatkan nilai untuk mengetahui performa dari model yang dihasilkan, antara lain yaitu :

1. Akurasi merupakan rasio dari prediksi yang benar dari keseluruhan data sesuai dengan label atau kelasnya.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \dots \dots \dots (1)$$

2. Presisi merupakan rasio prediksi yang benar dari keseluruhan hasil yang diprediksi positif. Presisi

menjelaskan berapa persentase data yang sesuai kelasnya dari seluruh data yang diprediksi sesuai kelasnya.

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots(2)$$

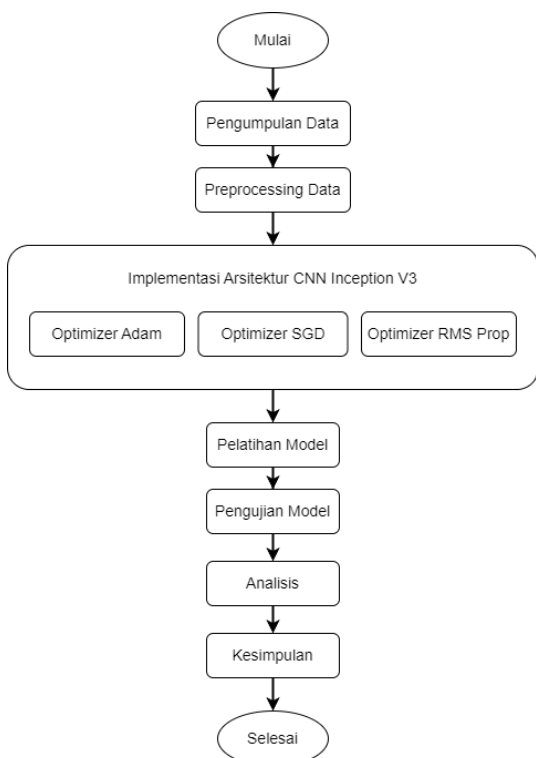
3. *Recall* merupakan rasio dari prediksi yang benar dari keseluruhan data yang memang benar. *Recall* menjelaskan berapa persentase data yang diprediksi sesuai kelas dari keseluruhan data yang sesuai kelasnya.

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(3)$$

4. *F1-Score*, adalah rasio yang membandingkan antara nilai rata-rata presisi dan *recall*.

$$F1 = \frac{2 \times Recall \times Precision}{Recall + Precision} \dots\dots\dots(4)$$

3. METODOLOGI



Gambar 1. Alur Penelitian

Alur penelitian untuk identifikasi penyakit pada daun tomat dapat dilihat pada Gambar 1, ada beberapa tahapan dalam melakukan identifikasi dengan menggunakan model *transfer learning* pada metode *Convolutional Neural Network* (CNN). Tahap awal dimulai dari pengumpulan data yang digunakan, kemudian melakukan *preprocessing data* yaitu merubah ukuran citra dan membagi dataset yang digunakan ke data *training*, *validation* dan data *testing*. selanjutnya dilakukan pengujian model arsitektur *Convolution Neural Network* (CNN) menggunakan *transfer learning Inception-V3*. Tahap selanjutnya adalah proses klasifikasi dengan *transfer learning* menggunakan arsitektur *Inception-V3*, kemudian hasil dari klasifikasi akan evaluasi menggunakan tabel confusion matrik untuk mengetahui model

CNN yang terbaik dalam mengidentifikasi penyakit pada daun tomat.

3.1. Dataset

Data yang digunakan untuk identifikasi penyakit daun tomat pada penelitian ini adalah data citra daun tomat yang didapatkan dari website kaggle dengan nama pemilik Kaustubh [13]. Data citra yang digunakan dibagi menjadi 10 kelas yaitu data citra daun sehat, data citra daun penyakit *mosaic virus*, data citra daun penyakit *target spot*, data citra daun penyakit *bacterial spot*, data citra daun penyakit *yellow leaf curl virus*, data citra daun penyakit *late blight*, data citra daun penyakit *leaf mold*, data citra daun penyakit *early blight*, data citra daun penyakit *spider mites*, dan data citra daun penyakit *septoria leaf spot*. Data yang ditemui sudah terbagi menjadi 2 folder yaitu folder *train* model dan folder *testing* yang terlihat pada Tabel 2.

Tabel 2. Detail Dataset Citra Daun Tomat

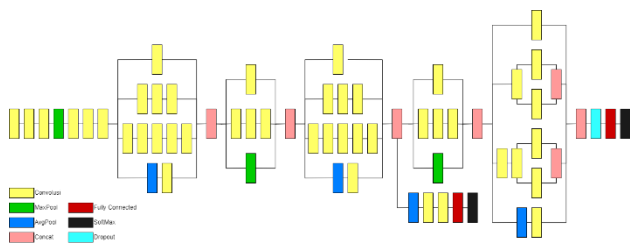
Sampel Data	Data Train Model	Data Testing
Daun Sehat	1000	100
Mosaic Virus	1000	100
Target Spot	1000	100
Bacterial Spot	1000	100
Yellow Leaf Curl Virus	1000	100
Late Blight	1000	100
Leaf Mold	1000	100
Early Blight	1000	100
Spider Mites	1000	100
Septoria Leaf Spot	1000	100
Total	11000	

3.2. Preprocessing Data

Tahap persiapan atau *preprocessing* data adalah mengolah data asli supaya bisa siap digunakan pada pada model klasifikasi. *Preprocessing* data pada penelitian ini adalah merubah ukuran citra ke ukuran 224x224 piksel untuk meningkatkan nilai akurasi proses klasifikasi. Data dari folder *train* model pada *dataset* kemudian dibagi menjadi 2 bagian untuk dimasukkan kedalam model klasifikasi, perincian pembagian data ini adalah 80% untuk data training yaitu data yang digunakan untuk melatih model klasifikasi, 20% untuk data *validation* yaitu data untuk memvalidasi model klasifikasi untuk menghindari terjadinya *overfitting*, dan untuk data testing menggunakan data dari folder *testing* yaitu data yang digunakan untuk menguji ketepatan model klasifikasi.

3.3. Transfer Learning

Transfer learning adalah teknik di mana model arsitektur CNN menggunakan pengetahuan yang telah diperoleh dari dataset sebelumnya untuk klasifikasi pada dataset baru. Dengan memiliki lapisan *convolution* dan *pooling* yang lebih dalam daripada model CNN standar, *transfer learning* mampu mengekstraksi lebih banyak fitur tekstur dari citra dan menghasilkan representasi yang lebih kaya dari informasi visual.



Gambar 2. Arsitektur Inception-V3

Kedalaman lapisan arsitektur pada *Transfer Learning* yang membuat metode ini bisa mengatasi *overfitting* pada data yang sedikit. Penelitian ini akan menggunakan metode *Transfer Learning Inception-V3*. Gambar 2 adalah susunan arsitektur dari model *Transfer Learning Inception-V3*, pada arsitektur ini memiliki lapisan yang lebih dalam dibandingkan dengan arsitektur CNN sederhana.

3.4. Optimizer

Algoritma optimasi bertujuan untuk meminimalkan kesalahan, menemukan bobot optimal dan memaksimalkan tingkat akurasi. Penelitian ini menggunakan 3 jenis *optimizer* yaitu Adam, SGD, dan RMS Prop.

1. Adam (*Adaptive Moment Estimation*)

Adam menggabungkan keunggulan RMSprop dan momentum. Ini menghitung gradien adaptif serta momentum *gradien* untuk mengoptimalkan efisiensi pelatihan. Adam secara efektif mengelola laju pembelajaran dan menyediakan penyesuaian yang adaptif untuk setiap parameter berdasarkan estimasi kedua momen gradien (*mean* dan *varians gradien*) [14].

2. Stochastic Gradient Descent (SGD)

SGD adalah optimasi dasar yang digunakan dalam pelatihan model deep learning. Ini menghitung gradien dari fungsi kerugian terhadap setiap batch data pelatihan dan melakukan langkah berlawanan dengan gradien untuk memperbarui bobot model. SGD sering digunakan karena kesederhanaannya, meskipun cenderung memiliki kecepatan konvergensi yang lebih lambat dibandingkan optimasi lainnya [15].

3. RMSprop

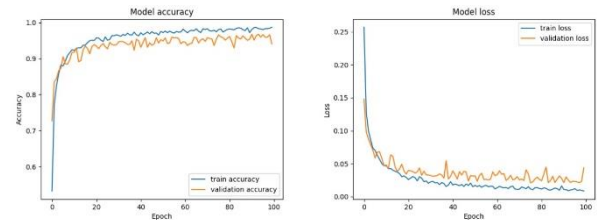
RMSprop adalah optimasi yang mengadaptasi laju pembelajaran untuk setiap parameter berdasarkan perubahan *gradien* dalam iterasi terakhir. Ini membantu mengatasi masalah laju pembelajaran yang tidak konsisten dalam SGD dengan menyesuaikan laju pembelajaran secara adaptif untuk setiap parameter. RMSprop efektif untuk pelatihan *deep neural networks* dan membantu mempercepat *konvergensi* pelatihan [14].

Penelitian ini akan melakukan analisa tiga algoritma optimasi mana yang terbaik dalam melakukan klasifikasi daun yang sehat dan daun yang memiliki penyakit. Proses analisa untuk mencari model yang terbaik pada penelitian ini menggunakan metode pengujian *Confusion Matrix*. Pengujian metode *Confusion Matrix* menggunakan hasil data yang diprediksi oleh model klasifikasi dengan data yang sesungguhnya.

4. HASIL DAN PEMBAHASAN

Penelitian ini melakukan beberapa percobaan sesuai tahapan pada kerangka penelitian, percobaan pertama adalah dengan menggunakan *optimizer* Adam untuk klasifikasi citra daun, kemudian dilakukan percobaan dengan menggunakan arsitektur model *transfer learning Inception-V3*.

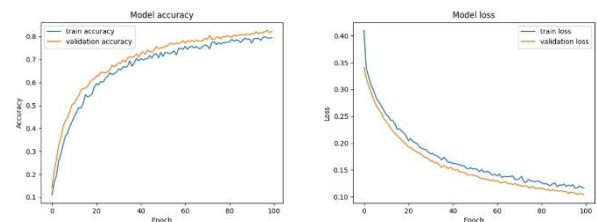
4.1. Hasil Klasifikasi dengan Optimizer Adam



Gambar 3. Grafik Akurasi dan Loss dengan Optimizer Adam

Hasil klasifikasi pada data *training* dan data *validation* menggunakan *optimizer* Adam menunjukkan performa yang baik seperti yang ditunjukkan pada Gambar 3 nilai akurasi pada data *training* dan data *validation* meningkat secara stabil dan nilai loss pada data *training* dan data *validation* menurun secara stabil. Grafik pada Gambar 3 menunjukkan nilai akurasi dan *loss* pada data *training* dan data validasi tidak memiliki perbedaan yang lebar.

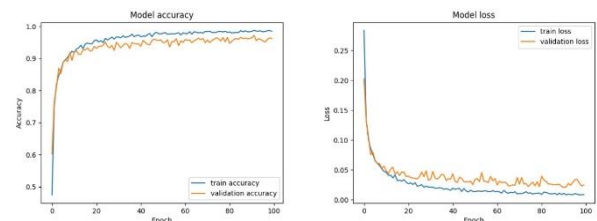
4.2. Hasil Klasifikasi dengan Optimizer SGD



Gambar 4. Grafik Akurasi dan Loss dengan Optimizer SGD

Hasil klasifikasi pada data *training* dan data *validation* menggunakan *optimizer* SGD menunjukkan performa yang baik seperti yang ditunjukkan pada Gambar 4 nilai akurasi pada data *training* dan data *validation* meningkat secara stabil dan nilai *loss* pada data *training* dan data *validation* menurun secara stabil. Tetapi hasil grafik Gambar 4 menunjukkan nilai akurasi yang masih rendah dan nilai *loss* yang tinggi.

4.3. Hasil Klasifikasi dengan Optimizer RMS Prop



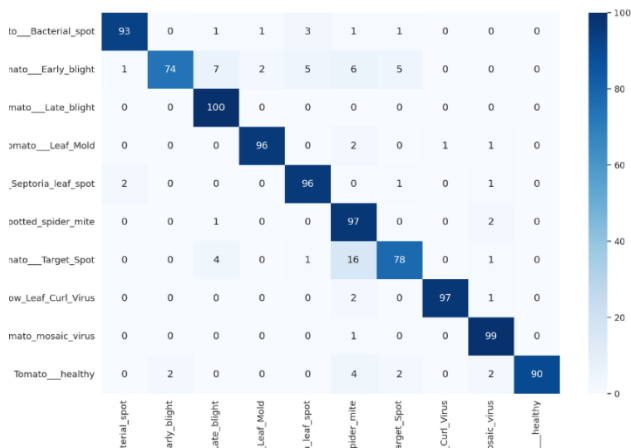
Gambar 5. Grafik Akurasi dan Loss dengan Optimizer RMS Prop

Hasil klasifikasi pada data *training* dan data *validation* menggunakan *optimizer* RMS Prop menunjukkan performa yang baik seperti yang ditunjukkan pada Gambar 5 nilai akurasi pada

data *training* dan data *validation* meningkat secara stabil dan nilai *loss* pada data *training* dan data *validation* menurun secara stabil. Grafik pada Gambar 5. menunjukkan nilai akurasi dan *loss* pada data *training* dan data validasi tidak memiliki perbedaan yang lebar.

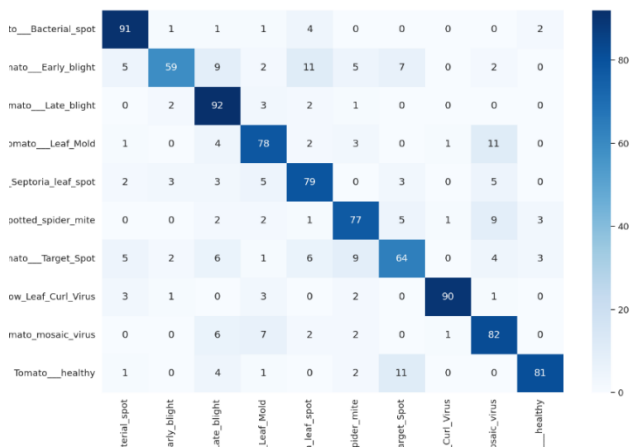
4.4. Hasil Pengujian menggunakan Confusion Matrix

Tahap terakhir adalah melakukan pengujian model klasifikasi pada data testing dan dianalisa menggunakan *Confusion Matrix*. Hasil pengujian dengan menggunakan *Confusion Matrix* dapat dilihat pada Gambar 6, Gambar 7, dan Gambar 8 untuk menghitung tingkat akurasi dari pengujian menggunakan metode ini adalah dengan cara membandingkan nilai data yang diprediksi benar dengan seluruh jumlah data yang digunakan.



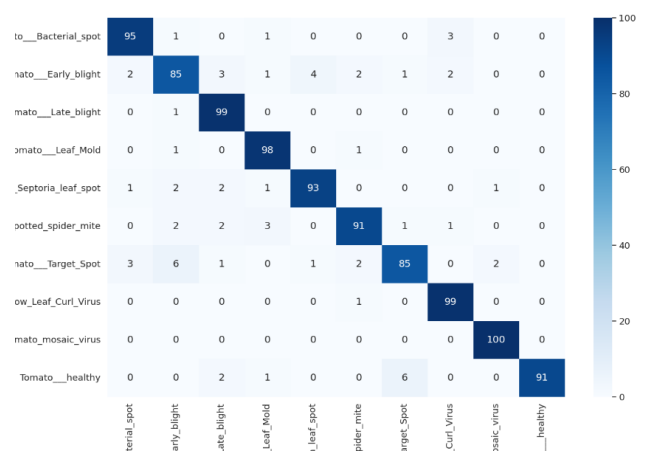
Gambar 6. Confusion Matrix pada Optimizer Adam

Hasil pengujian dengan *confusion matrix* menggunakan *optimizer* Adam pada gambar 6 tersebut berhasil mengklasifikasikan 938 gambar dari 1000 gambar yang diujikan dan gagal mengklasifikasikan pada 62 gambar.



Gambar 7. Confusion Matrix pada Optimizer SGD

Hasil pengujian dengan *confusion matrix* menggunakan *optimizer* SGD gambar 7 tersebut berhasil mengklasifikasikan 787 gambar dari 1000 gambar yang diujikan dan gagal mengklasifikasikan pada 213 gambar.



Gambar 8. Confusion Matrix pada Optimizer RMS Prop

Hasil pengujian dengan *confusion matrix* menggunakan *optimizer* RMS Prop gambar 8 tersebut berhasil mengklasifikasikan 936 gambar dari 1000 gambar yang diujikan dan gagal mengklasifikasikan pada 64 gambar.

Tabel 3. Perbandingan Hasil Pengujian

Optimizer	Akurasi (%)	Precision (%)	Recall (%)	F1-Score (%)
Adam	93,8%	93,9%	93,8%	93,7%
SGD	78,7%	79%	78,7%	78,6%
RMS Prop	93,6%	93,7%	93,6%	93,5%

Tabel 2 merupakan hasil perbandingan dari nilai akurasi yang didapatkan dengan menggunakan *Confusion Matrix*. Nilai akurasi tertinggi dihasilkan oleh *optimizer* Adam dengan nilai 93,8% dengan selisih nilai 0,2% dengan *optimizer* RMS Prop dan selisih nilai 15,1% dengan *optimizer* SGD. Serta untuk nilai *precision*, *recall*, *f1-score* tertinggi dihasilkan oleh *optimizer* Adam dengan nilai *precision* 93,9%, nilai *recall* 93,8%, dan nilai *f1-score* 93,7%.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil dari percobaan proses klasifikasi penyakit daun tanaman tomat dengan metode Convolutional Neural Network (CNN) menggunakan *transfer learning* Inception-V3 pada penelitian ini dapat diambil kesimpulan hasil klasifikasi menggunakan *optimizer* Adam, SGD, dan RMS Prop menghasilkan bahwa *optimizer* Adam memiliki hasil yang paling baik, dibuktikan dengan performa model yang stabil dan akurasi yang paling tinggi dengan jumlah 93,8%.

Penelitian selanjutnya diharapkan dapat menggunakan dataset dengan jumlah yang lebih banyak, sehingga dapat menghasilkan performa model yang lebih baik, atau bisa dengan menggunakan data primer untuk pengujiannya. Dan pada penelitian selanjutnya *Optimizer* Adam dapat digunakan untuk penelitian selanjutnya apabila ingin membuat model dengan akurasi tinggi dengan data yang banyak.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik, “Produksi Tanaman Sayuran Menurut Provinsi dan Jenis Tanaman 2022,” Badan Pusat Statistik. Accessed: Mar. 02, 2024. [Online]. Available: <https://www.bps.go.id/id/statistics-table/3/ZUhFd1JtZzJWVVpQWTJsV05XTllhVmhRSzFoNFFUMDkjMw==/produksi-tanaman-sayuran-menurut-provinsi-dan-jenis-tanaman--2022.html?year=2022>
- [2] W. Pradana, “Gagal Panen, Petani di Lembang Babat Habis 25 Ribu Pohon Tomatnya,” detikNews. Accessed: Mar. 03, 2024. [Online]. Available: <https://news.detik.com/berita-jawa-barat/d-5869139/gagal-panen-petani-di-lembang-babat-habis-25-ribu-pohon-tomatnya>
- [3] P. Echo, “Atasi Hama yang Menyerang Tanaman Tomat Tanpa Pestisida Kimia,” Fakultas Pertanian dan Peternakan Universitas Muhammadiyah Kotabumi. Accessed: Mar. 03, 2024. [Online]. Available: <https://fpp.umko.ac.id/2022/01/03/atasi-hama-yang-menyering-tanaman-tomat-tanpa-pestisida-kimia/>
- [4] B. T. Wiryanta, *Bertanam Tomat*, Printing 1. Jakarta: Agromedia Pustaka, 2002.
- [5] O. A. Montesinos López, A. Montesinos López, and J. Crossa, “Convolutional Neural Networks,” in *Multivariate Statistical Machine Learning Methods for Genomic Prediction*, O. A. Montesinos López, A. Montesinos López, and J. Crossa, Eds., Cham: Springer International Publishing, 2022, pp. 533–577. doi: 10.1007/978-3-030-89010-0_13.
- [6] M. M. Taye, “Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions,” *Computation*, vol. 11, no. 3. MDPI, Mar. 01, 2023. doi: 10.3390/computation11030052.
- [7] Z. Zhao, L. Alzubaidi, J. Zhang, Y. Duan, and Y. Gu, “A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations,” *Expert Systems with Applications*, vol. 242. Elsevier Ltd, May 15, 2024. doi: 10.1016/j.eswa.2023.122807.
- [8] J. Gupta, S. Pathak, and G. Kumar, “Deep Learning (CNN) and Transfer Learning: A Review,” in *Journal of Physics: Conference Series*, Institute of Physics, 2022. doi: 10.1088/1742-6596/2273/1/012029.
- [9] M. Agil Izzulhaq, “Indonesian Journal of Mathematics and Natural Sciences Penerapan Algoritma Convolutional Neural Network Arsitektur ResNet50V2 Untuk Mengidentifikasi Penyakit Pneumonia,” 2024. [Online]. Available: <https://journal.unnes.ac.id/journals/JM/index>
- [10] A. Kholik, “KLASIFIKASI MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA TANGKAPAN LAYAR HALAMAN INSTAGRAM,” *JDMSI*, vol. 2, no. 2, pp. 10–20, 2021.
- [11] D. Gunawan and H. Setiawan, “Convolutional Neural Network dalam Analisis Citra Medis,” 2022.
- [12] K. M. Ting, “Confusion Matrix,” in *Encyclopedia of Machine Learning*, G. I. Sammut Claude and Webb, Ed., Boston, MA: Springer US, 2010, p. 209. doi: 10.1007/978-0-387-30164-8_157.
- [13] B. Kaustubh, “Tomato leaf disease detection,” Kaggle. Accessed: Mar. 27, 2024. [Online]. Available: <https://www.kaggle.com/datasets/kaustubhb999/tomato-leaf>
- [14] E. Hassan, M. Y. Shams, N. A. Hikal, and S. Elmougy, “The effect of choosing optimizer algorithms to improve computer vision tasks: a comparative study,” *Multimed Tools Appl*, vol. 82, no. 11, pp. 16591–16633, May 2023, doi: 10.1007/s11042-022-13820-0.
- [15] C. G. Desai and C. Desai, “Comparative Analysis of Optimizers in Deep Neural Networks,” 2020. [Online]. Available: www.ijisrt.com959

BIODATA PENULIS



Andi Nurdin

Mahasiswa Program Studi Sistem Informasi Universitas Pembangunan Nasional “Veteran” Jawa Timur, Surabaya.

Email: andinurdin.ndvy@gmail.com



Dhian Satria Yudha Kartika

Dosen Sistem Infomasi Universitas Pembangunan Nasional “Veteran” Jawa Timur, Surabaya.

Email: dhian.satria@upnjatim.ac.id



Abdul Rezha Efrat Najaf

Dosen Sistem Infomasi Universitas Pembangunan Nasional “Veteran” Jawa Timur, Surabaya.

Email: rezha.efrat.sifo@upnjatim.ac.id