

# **Implementasi Model EfficientNetB0 Pada Pembuatan Aplikasi Desktop Untuk Identifikasi Hama Tanaman Sawi Berbasis Deep Learning**

**Jose Julian Hidayat<sup>1\*</sup>, Muhammad Dikaisa Ibnu Amin<sup>2</sup>, Endah Kurnia Fitri<sup>3</sup>,  
Ariska Nur Angraini<sup>4</sup>, Cindy Setyowati<sup>5</sup>, Aditya Pratama Werdana<sup>6</sup>**

<sup>1,2,3,4,5,6</sup>Teknik Informatika, Fakultas Teknik, Universitas Pelita Bangsa, Bekasi, Indonesia.

Dikirimkan: 23-11-2025

Diterbitkan: 30-11-2025

## **Keywords:**

Aplikasi Desktop;  
Deep Learning;  
Deteksi Hama;  
EfficientNetB0;  
Tanaman Sawi.

## **E-mail Penulis**

### **korespondensi:**

[josejulianhidayat@gmail.com](mailto:josejulianhidayat@gmail.com)

**Abstrak.** Deteksi hama pada tanaman sawi merupakan langkah penting untuk menjaga kualitas hasil pertanian. Penelitian ini mengembangkan aplikasi desktop untuk mengidentifikasi daun sawi ada hama atau tanpa hama menggunakan metode deep learning berbasis EfficientNetB0. Dataset yang digunakan berasal dari CAISIM dan terdiri dari dua kelas citra daun sawi. Model dilatih menggunakan binary cross-entropy serta teknik data augmentation untuk meningkatkan generalisasi. Hasil pengujian menunjukkan bahwa model EfficientNetB0 mampu mencapai akurasi validasi sebesar 97,65% dengan nilai loss 0,0544, sehingga sangat efektif dalam membedakan daun sawi yang terinfeksi hama dan yang sehat. Model terlatih kemudian diintegrasikan ke dalam aplikasi desktop berbasis Python, sehingga pengguna dapat melakukan deteksi secara langsung melalui antarmuka yang sederhana dan mudah digunakan. Penelitian ini membuktikan bahwa integrasi EfficientNetB0 dalam aplikasi desktop dapat menjadi solusi praktis untuk mendukung sistem deteksi hama pada pertanian modern.

**Abstract.** Detecting pests on mustard plants is an important step in maintaining the quality of agricultural yields. This study developed a desktop application to identify mustard leaves with or without pests using the EfficientNetB0-based deep learning method. The dataset used was sourced from CAISIM and consisted of two classes of mustard leaf images. The model was trained using binary cross-entropy and data augmentation techniques to improve generalisation. Test results show that the EfficientNetB0 model is capable of achieving a validation accuracy of 97.65% with a loss value of 0.0544, making it highly effective in distinguishing between cabbage leaves that are infected with pests and those that are healthy. The trained model was then integrated into a Python-based desktop application, allowing users to perform detection directly through a simple and easy-to-use interface. This research proves that the integration of EfficientNetB0 into desktop applications can be a practical solution to support pest detection systems in modern agriculture.

## **1. Pendahuluan**

Salah satu komoditas *hortikultura* yang sangat menguntungkan dan banyak dikonsumsi oleh masyarakat Indonesia adalah Sawi [1]. Namun, hama sering merusak daun tanaman Sawi, menyebabkan kualitas dan kuantitas hasil panen menurun [2]. Proses deteksi hama secara manual melalui pengamatan visual masih sering digunakan. Namun, metode ini memiliki beberapa keterbatasan, termasuk membutuhkan pengalaman tenaga ahli, waktu yang lama, dan kemungkinan kesalahan dalam identifikasi [3]. Oleh karena itu, solusi kontemporer diperlukan untuk mengidentifikasi hama secara cepat, akurat, dan efektif [4].

Proses identifikasi hama dan penyakit tanaman menjadi lebih mudah berkat kemajuan *AI* dan *Computer Vision*. Metode *deep learning*, terutama *Convolutional Neural Network (CNN)*, telah terbukti lebih efektif dalam pengolahan citra karena kemampuan untuk mengekstraksi informasi visual secara otomatis. *EfficientNetB0* adalah salah satu arsitektur *CNN* yang memiliki kelebihan dalam hal efisiensi komputasi dan akurasi tinggi [5]. Oleh karena itu, cocok digunakan untuk keperluan deteksi hama pada daun tanaman.

Penelitian sebelumnya menunjukkan bahwa *CNN* sangat baik dalam klasifikasi gambar daun tanaman. Sebagai contoh, sebuah penelitian yang dilakukan untuk mengidentifikasi hama daun di tanaman hidroponik menemukan bahwa metode *CNN* dapat mendeteksi empat jenis daun berbeda: daun kriting, daun kutu kebuluran, daun sehat, dan daun terkena septoria. Dengan menggunakan 183 data pelatihan dan 49 data validasi, penelitian tersebut menemukan bahwa jarak pengambilan gambar, pencahayaan, dan kejernihan citra sangat memengaruhi akurasi metode *CNN* [6]. Hasil ini menunjukkan bahwa penggunaan *CNN* sangat mungkin diterapkan untuk mendeteksi hama daun sawi.

Penelitian ini menggunakan arsitektur *EfficientNetB0* untuk mengembangkan sistem deteksi hama pada daun sawi. Dataset yang digunakan adalah *CAISIM*, yang mengandung gambar daun sawi dalam dua kategori utama: daun dengan hama dan daun tanpa hama. Pelatihan model dilakukan melalui tahapan *preprocessing*, augmentasi data, dan perancangan arsitektur jaringan sesuai kebutuhan klasifikasi [7]. Penelitian ini tidak hanya berfokus pada pengembangan model, itu juga berfokus pada penerapan sistem aplikasi dalam dunia nyata.

Pengembangan aplikasi *desktop* ini memungkinkan deteksi hama pada tanaman sawi dengan lebih cepat, efisien, dan akurat. Untuk mendapatkan hasil prediksi secara otomatis, pengguna hanya perlu mengunggah gambar daun ke aplikasi. Diharapkan penelitian ini dapat membantu praktik pertanian kontemporer, meningkatkan produktivitas, dan membantu petani menemukan serangan hama sejak dini.

## 2. Metode Penelitian

### 2.1. Desain Penelitian

Penelitian ini menggunakan metode eksperimental dengan penerapan teknik *deep learning* berbasis *Convolutional Neural Network (CNN)* untuk melakukan klasifikasi hama pada daun sawi. Model yang digunakan adalah *EfficientNetB0* yang diintegrasikan ke dalam aplikasi berbasis *desktop*. Tahapan penelitian meliputi pengumpulan dataset, *preprocessing* data, perancangan model, pelatihan model, evaluasi performa, dan implementasi dalam aplikasi *desktop*.

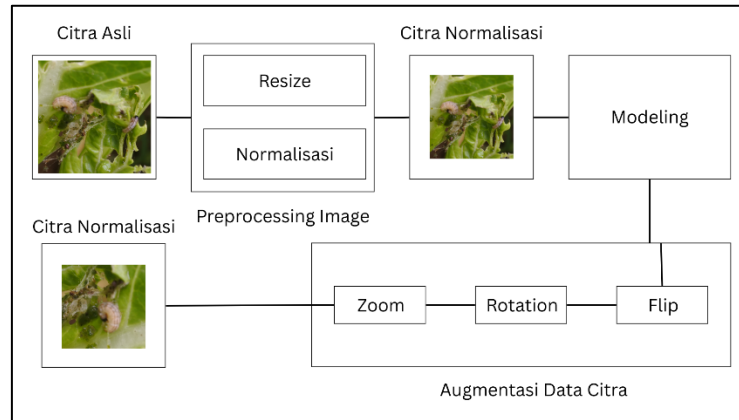
### 2.2. Data Collection

Data yang digunakan dalam penelitian ini berasal dari himpunan gambar daun sawi *CAISIM* dari dua kelas utama: Data Sawi Ada Hama dan Data Sawi Tanpa Hama. 1.065 gambar total digunakan, terdiri dari 852 gambar untuk proses pelatihan dan 213 gambar untuk proses validasi. Gambar tersebut diambil dengan berbagai kondisi pencahayaan, sudut pengambilan, dan tingkat kejernihan gambar, sehingga dapat mencerminkan berbagai kondisi lapangan. Kemudian, dataset ini diolah menggunakan fungsi *image\_dataset\_from\_directory()* pada *TensorFlow*, dengan ukuran gambar yang diseragamkan menjadi 224 x 224 piksel dan format warna *RGB*. Dengan menggunakan dataset ini, model dapat lebih baik mempelajari cara membedakan karakteristik visual antara daun sawi yang sehat dan daun yang terinfeksi hama.

### 2.3. Preprocessing dan Augmentasi Data

Sebelum digunakan pada tahap pelatihan model, proses *preprocessing* dilakukan untuk memastikan bahwa semua gambar dalam dataset memiliki format yang sama [8]. Sebelum diubah menjadi ukuran 224 x 224 piksel untuk memenuhi persyaratan arsitektur *EfficientNetB0*, setiap gambar diubah ke format *RGB* dan diubah menjadi *tensor numerical* agar jaringan saraf dapat menanganinya [9]. Untuk meningkatkan stabilitas pelatihan, citra juga dinormalisasi sehingga nilai piksel berada dalam rentang yang sesuai [10]. Sehingga model dapat mempelajari pola visual secara konsisten, tahapan *preprocessing* ini bertujuan untuk mengurangi perbedaan teknis antar gambar [11].

Penelitian ini menggunakan berbagai teknik augmentasi data untuk meningkatkan kemampuan generalisasi model dan menghindari *overfitting* [12]. Teknik augmentasi seperti *RandomFlip*, *RandomRotation*, dan *RandomZoom* secara acak mengubah orientasi, sudut rotasi, dan tingkat pembesaran gambar. Modul keras digunakan untuk meningkatkan karena transformasi diterapkan secara otomatis selama pelatihan dengan *sequential()*. Metode ini membuat model dapat mengidentifikasi variasi kondisi citra yang lebih luas, seperti perbedaan pencahayaan atau posisi daun, yang menghasilkan kinerja yang lebih baik pada data nyata.



Gambar 1. Prapemrosesan dan Augmentasi Citra

## 2.4. Arsitektur Pemodelan

Arsitektur model yang digunakan dalam penelitian ini adalah *EfficientNetB0* yang berperan sebagai *feature extractor utama*. Model dibangun menggunakan pendekatan *Functional API*, diawali dengan input berukuran  $224 \times 224 \times 3$  yang kemudian diproses melalui lapisan *data augmentation* meliputi *RandomFlip*, *RandomRotation*, dan *RandomZoom*. Setelah augmentasi, citra diteruskan ke tahap *preprocessing* menggunakan fungsi *eff\_preprocess* sebelum memasuki *backbone EfficientNetB0* dengan parameter *include\_top=False* dan bobot awal *ImageNet*. Keluaran dari *EfficientNetB0* kemudian dirangkum melalui lapisan *GlobalAveragePooling2D*, dilanjutkan dengan lapisan *Dense 256 (ReLU)* dan *Dropout 0.5*, kemudian *Dense 64 (ReLU)* dan *Dropout 0.3* untuk memperkuat generalisasi model. Tahap akhir menggunakan *Dense(1, activation='sigmoid')* sebagai *output* untuk melakukan klasifikasi biner antara daun sawi ada hama dan tanpa hama. Struktur arsitektur ini dirancang untuk menghasilkan keseimbangan antara akurasi tinggi dan efisiensi komputasi.

## 2.5. Evaluasi dan Implementasi

Evaluasi model dilakukan menggunakan *validation dataset* yang terdiri dari 213 citra daun sawi untuk mengukur performa model dalam mengklasifikasikan dua kategori, yaitu daun sawi ada hama dan daun sawi tanpa hama. Pengujian dilakukan menggunakan metrik akurasi, *loss*, *Confusion Matrix*, dan *Classification Report* untuk memastikan keandalan prediksi model *EfficientNetB0*. Hasil evaluasi menunjukkan performa yang konsisten dan akurat, sehingga model dianggap layak untuk diterapkan dalam penggunaan praktis. Setelah melalui tahap evaluasi, model yang telah dilatih kemudian diintegrasikan ke dalam sebuah aplikasi desktop berbasis Python (*Tkinter*) yang dirancang agar pengguna dapat mengunggah citra daun sawi dan memperoleh hasil prediksi secara langsung. Sistem melakukan *preprocessing* otomatis, menjalankan model untuk menghasilkan keluaran, serta menampilkan status kondisi daun dalam antarmuka yang sederhana dan mudah dipahami. Implementasi ini bertujuan untuk memberikan solusi praktis dalam deteksi hama pada tanaman sawi, sehingga dapat membantu petani dan pengguna umum dalam melakukan identifikasi dini secara cepat tanpa memerlukan keahlian teknis.

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

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

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

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

## 3. Hasil dan Pembahasan

### 3.1. Hasil Penelitian

Pelatihan model dilakukan menggunakan arsitektur *EfficientNetB0* dengan dua tahap, yaitu *transfer learning* dan *fine-tuning*. Pada tahap awal, *backbone EfficientNetB0* dibekukan dan hanya lapisan klasifikasi yang dilatih selama 15 *epoch*. Hasil pelatihan menunjukkan peningkatan akurasi dari *epoch* ke *epoch* dengan nilai *loss* yang terus menurun, menandakan bahwa model mampu mempelajari pola daun secara efektif. Tahap kedua dilakukan dengan

membuka sebagian lapisan *EfficientNetB0* untuk *fine-tuning* selama 5 *epoch* sehingga model dapat menyesuaikan bobot lebih optimal terhadap dataset daun sawi.

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

$$Accuracy = \frac{128 + 84}{124 + 84 + 5 + 0} \quad (2)$$

$$Accuracy = \frac{208}{203} = 0.9765 \text{ atau } 97.65\% \quad (3)$$

Proses evaluasi dilakukan menggunakan 213 citra validasi. Model menunjukkan performa yang sangat baik dengan tingkat akurasi validasi sebesar 97,65% dan nilai *loss* sebesar 0,0544, sesuai hasil evaluasi pada laporan. *Confusion Matrix* memperlihatkan bahwa sebagian besar citra dapat diklasifikasikan dengan benar, baik untuk kelas “Data Sawi Ada Hama” maupun “Data Sawi Tanpa Hama”. *Classification Report* juga menunjukkan nilai *precision*, *recall*, dan *F1-score* yang tinggi pada kedua kelas, menandakan bahwa model tidak hanya akurat tetapi juga seimbang dalam mengenali kedua kategori daun.

$$Precision = \frac{TP}{TP + FP} = \frac{124}{124 + 5} = \frac{124}{129} = 0.9612 \text{ atau } 96.12\% \quad (4)$$

$$Recall = \frac{TP}{TP + FN} = \frac{124}{124 + 0} = \frac{124}{124} = 1 \text{ atau } 100\% \quad (5)$$

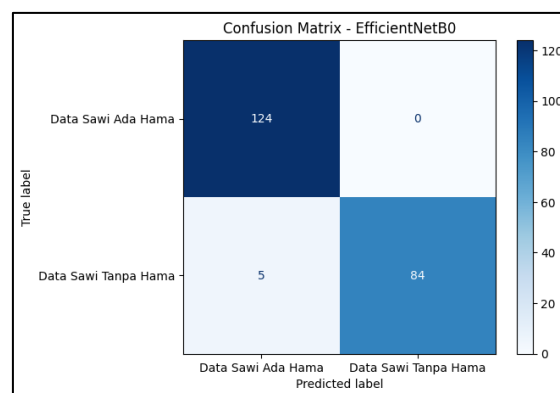
$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

$$F1 - Score = 2 \times \frac{0.9612 \times 1.0}{0.9612 + 1.0} = 0.9800 \text{ atau } 98\% \quad (7)$$

Selain pengujian pada dataset, model juga diuji menggunakan beberapa citra uji manual dengan kondisi pencahayaan dan sudut pengambilan gambar yang berbeda. Berdasarkan pengujian tersebut, model mampu menghasilkan prediksi yang konsisten dan sesuai dengan kondisi daun, sehingga membuktikan kemampuan generalisasi model yang baik. Hasil ini mendukung bahwa *EfficientNetB0* merupakan arsitektur yang tepat digunakan pada penelitian deteksi hama daun sawi.

### 3.2. Pembahasan

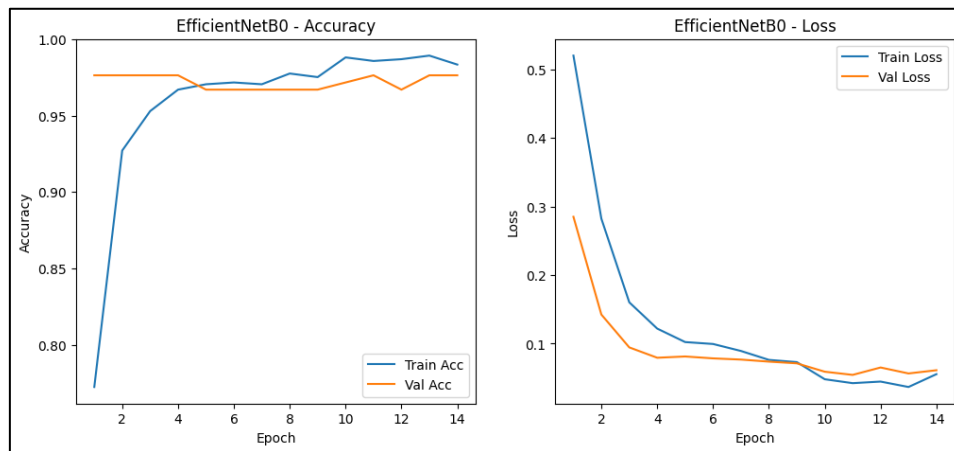
Hasil pelatihan dan evaluasi menunjukkan bahwa penggunaan *EfficientNetB0* memberikan performa yang sangat baik untuk tugas klasifikasi daun sawi. Keberhasilan model dipengaruhi oleh beberapa faktor utama, antara lain penggunaan data *augmentation* yang membantu model mengenali variasi citra, penggunaan fungsi aktivasi *ReLU* pada lapisan *dense*, dan pemilihan *binary cross-entropy* sebagai fungsi *loss* yang sesuai untuk klasifikasi biner. Selain itu, penggunaan fungsi *eff\_preprocess* pada *EfficientNetB0* turut memastikan bahwa citra *input* berada dalam format yang sesuai dengan standar *pretrained* model, sehingga memperkuat stabilitas pelatihan.



Gambar 2. *Confusion Matrix*

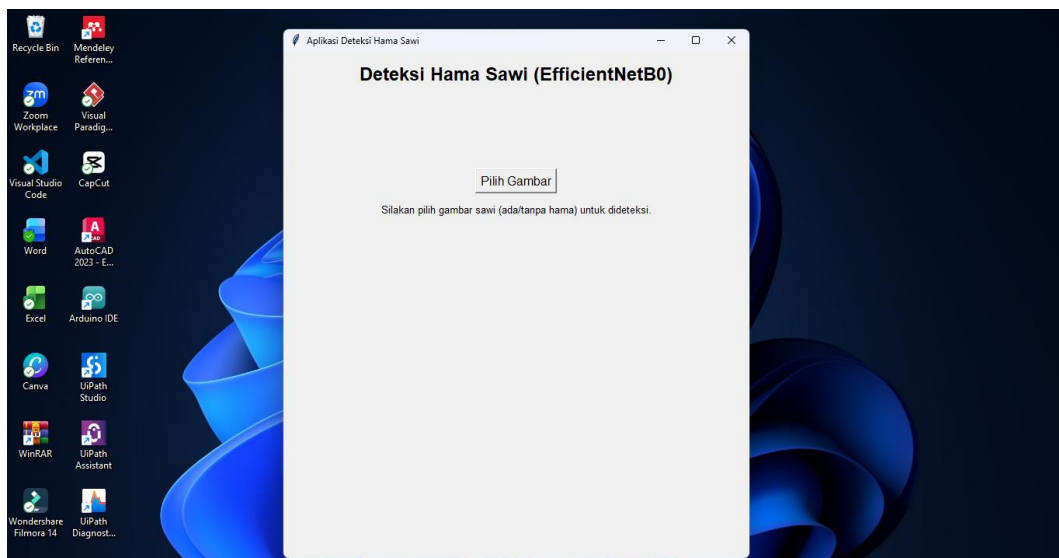
*Confusion Matrix* menunjukkan bahwa kesalahan prediksi sangat sedikit, sehingga risiko misklasifikasi pada data baru relatif rendah. Hal ini berarti model mampu melihat perbedaan visual antara daun yang terinfeksi hama dan daun sehat dengan cukup baik. Namun demikian, seperti penelitian serupa pada tanaman hidroponik yang

menyoroti pengaruh kualitas citra terhadap hasil klasifikasi, performa model pada penelitian ini juga sangat bergantung pada kualitas input, seperti pencahayaan, ketajaman gambar, serta jarak pengambilan foto. Oleh karena itu, citra dengan kualitas buruk dapat berpotensi menurunkan akurasi prediksi.



Gambar 3. Kurva Pelatihan Model *EfficientNetB0*

Grafik akurasi dan loss menunjukkan bahwa proses pelatihan model *EfficientNetB0* berlangsung secara stabil dan efektif. Pada grafik akurasi, terlihat bahwa nilai *train accuracy* meningkat tajam dari *epoch* awal kemudian terus naik hingga mendekati 1.00, sedangkan *validation accuracy* berada pada rentang stabil sekitar 0.96–0.98, yang menandakan bahwa model mampu melakukan generalisasi dengan baik pada data validasi. Sementara itu, grafik loss memperlihatkan penurunan *train loss* yang konsisten dari nilai awal yang tinggi menuju nilai rendah di bawah 0.1, diikuti oleh *validation loss* yang turun dengan pola yang hampir serupa. Kurva *validation loss* yang relatif stabil tanpa lonjakan signifikan menunjukkan bahwa model tidak mengalami *overfitting* secara berarti. Secara keseluruhan, kedua grafik tersebut mengindikasikan bahwa model berhasil mencapai performa yang optimal dengan keseimbangan antara akurasi tinggi dan nilai loss yang rendah, sehingga layak digunakan dalam proses deteksi hama pada daun sawi.



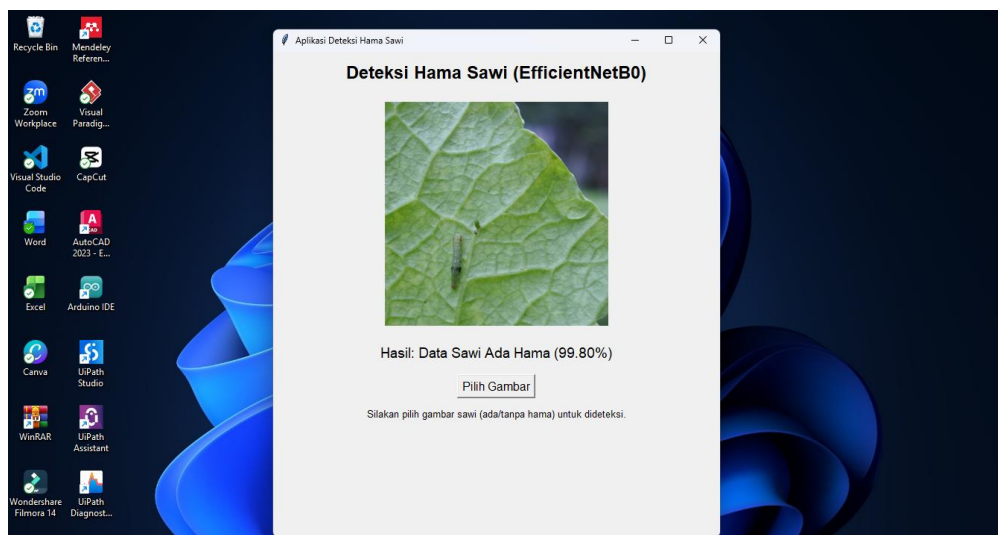
Gambar 4. Tampilan Aplikasi Desktop

Black Box Testing dilakukan dengan menguji fitur utama aplikasi, seperti fungsi unggah gambar, proses deteksi otomatis, tampilan hasil prediksi, serta respons aplikasi terhadap berbagai jenis input citra. Hasil pengujian menunjukkan bahwa aplikasi mampu menampilkan keluaran sesuai yang diharapkan untuk setiap skenario uji, termasuk ketika diberikan gambar daun sawi sehat maupun terinfeksi hama, sehingga memastikan bahwa aplikasi tidak hanya akurat secara algoritmik, tetapi juga berfungsi dengan baik saat digunakan oleh pengguna.

Tabel 1. Hasil Uji *Blaxbox Testing*

No	Skenario Uji	Input	Output yang Diharapkan	Output yang Dihasilkan	Status
1	Mengunggah gambar daun sawi	File gambar (.jpg/.png)	Gambar tampil pada area preview aplikasi	Gambar tampil dengan benar	Berhasil
2	Melakukan deteksi hama	Gambar daun sawi sehat	Output: “Tanpa Hama”	Aplikasi menampilkan “Tanpa Hama”	Berhasil
3	Melakukan deteksi hama	Gambar daun sawi terinfeksi hama	Output: “Ada Hama”	Aplikasi menampilkan “Ada Hama”	Berhasil
4	Input file bukan gambar	File PDF / TXT	Aplikasi menolak dan menampilkan pesan kesalahan	Aplikasi menampilkan peringatan invalid file	Berhasil
5	Tidak ada file yang dipilih	Klik tombol deteksi tanpa input	Aplikasi menampilkan peringatan “silakan pilih gambar”	Aplikasi menampilkan pesan peringatan	Berhasil
6	Menguji stabilitas aplikasi	Mengunggah beberapa gambar secara bergantian	Aplikasi tetap stabil dan memproses setiap gambar	Aplikasi berjalan stabil tanpa error	Berhasil

Gambar 5. tersebut menunjukkan tampilan aplikasi desktop deteksi hama sawi berbasis *EfficientNetB0* yang berhasil menampilkan citra daun sawi terinfeksi hama pada area antarmuka utama.

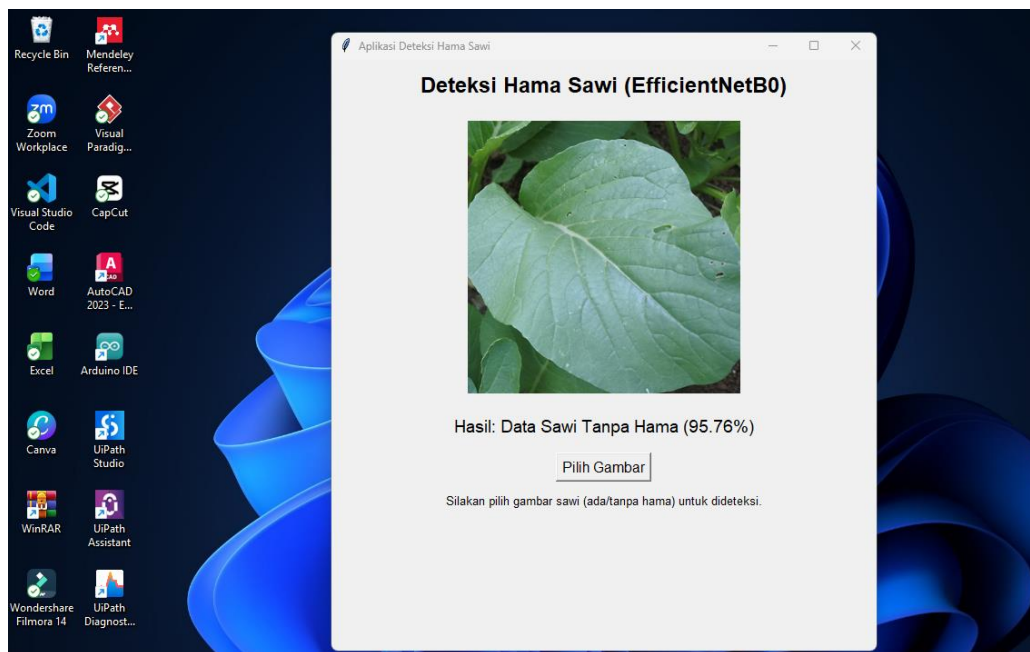


Gambar 5. Hasil Uji 1

Aplikasi desktop deteksi hama sawi berbasis *EfficientNetB0* pada gambar 6. tersebut menunjukkan bahwa sistem berhasil memproses citra daun sawi yang diunggah pengguna dan menampilkannya pada area tampilan aplikasi. Berdasarkan hasil prediksi, aplikasi mengklasifikasikan citra tersebut sebagai “Data Sawi Tanpa Hama” dengan tingkat keyakinan 95.76%, yang menandakan bahwa model mampu mengenali karakteristik daun sehat dengan sangat baik. Proses unggah gambar, analisis model, dan penampilan keluaran juga berjalan lancar tanpa kesalahan, sehingga membuktikan bahwa aplikasi mampu memberikan hasil deteksi yang akurat dan responsif pada kondisi daun sawi yang tidak terinfeksi hama.

Integrasi model ke dalam aplikasi desktop berbasis Python membuktikan bahwa model tidak hanya bekerja baik pada tahap pelatihan, tetapi juga dapat diterapkan dalam penggunaan sehari-hari. Aplikasi mampu memproses gambar yang diunggah pengguna, melakukan preprocessing otomatis, menjalankan prediksi menggunakan model *EfficientNetB0*, dan menampilkan hasil secara cepat. Hal ini menunjukkan bahwa sistem yang dikembangkan tidak hanya memiliki performa akademik yang baik tetapi juga memiliki nilai praktis yang tinggi bagi petani maupun pengguna umum.





Gambar 6. Hasil Uji 2

#### 4. Kesimpulan

Penelitian ini berhasil mengembangkan sebuah sistem deteksi hama pada daun sawi menggunakan metode *deep learning* berbasis arsitektur *EfficientNetB0*. Model dilatih menggunakan dataset *CAISIM* yang terdiri dari dua kelas, yaitu daun sawi ada hama dan tanpa hama, dengan proses *preprocessing*, augmentasi data, serta tahapan *transfer learning* dan *fine-tuning* yang dilakukan secara sistematis. Hasil evaluasi menunjukkan bahwa model mampu mencapai tingkat akurasi validasi sebesar 97,65% dengan nilai loss 0,0544, yang menandakan bahwa model memiliki kemampuan generalisasi yang sangat baik dalam membedakan daun sawi sehat dan yang terinfeksi hama.

Selain pengembangan model, penelitian ini juga berhasil mengimplementasikan sistem ke dalam sebuah aplikasi *desktop* berbasis *Python*, yang mampu melakukan deteksi hama secara otomatis melalui tampilan antarmuka yang sederhana dan mudah digunakan. Pengujian menggunakan metode *Black Box Testing* menunjukkan bahwa aplikasi mampu menjalankan seluruh fungsinya dengan baik, mulai dari pemilihan gambar, penampilan citra, hingga penyajian hasil prediksi dengan tingkat keyakinan yang tinggi. Dengan demikian, sistem ini dapat menjadi solusi praktis untuk membantu petani, peneliti, maupun pengguna umum dalam mengidentifikasi kondisi daun sawi secara cepat, akurat, dan efisien.

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan acuan untuk pengembangan penelitian selanjutnya. Pertama, diperlukan penambahan variasi data pada dataset, baik dari segi kondisi daun, pencahayaan, maupun jenis hama, agar model dapat beradaptasi pada kondisi lapangan yang lebih luas. Kedua, penggunaan arsitektur *deep learning* yang lebih kompleks atau teknik *ensemble* dapat dipertimbangkan untuk meningkatkan performa model, terutama dalam kondisi citra yang sulit atau berkualitas rendah. Ketiga, aplikasi *desktop* yang telah dikembangkan dapat ditingkatkan dengan penambahan fitur seperti penyimpanan riwayat deteksi, deteksi secara realtime melalui kamera, atau integrasi dengan pertanian berbasis web dan mobile agar lebih mudah diakses. Terakhir, uji coba langsung pada lingkungan pertanian nyata sangat disarankan untuk memvalidasi kinerja model dan memastikan bahwa sistem benar-benar efektif digunakan oleh petani di lapangan.

#### Daftar Rujukan

- [1] PRABOWO, Rachmat Udhi et al. AMPLIFIKASI SISTEM AGRIBISNIS KOMODITAS SAWI KABUPATEN MALANG (Studi Kasus Di Desa Pujon Kidul Kabupaten Malang). *AGRIBIOS*, [S.l.], v. 20, n. 2, p. 202-210, nov. 2022. ISSN 2723-7044. Available at: <<https://unars.ac.id/ojs/index.php/agribios/article/view/2361>>. Date accessed: 23 nov. 2025. doi: <https://doi.org/10.36841/agribios.v20i2.2361>.
- [2] Z. Zahrawati, S. Soedidjo, and H. Susanti, "Intensitas Serangan Hama Daun, Pertumbuhan dan Hasil Tanaman Sawi (*Brassica juncea* L.) yang Diaplikasi dengan Beberapa Konsentrasi dan Frekuensi Larutan Daun Galam (*Melaleuca cajuputi*)," *EnviroScientee*, vol. 17, no. 3, pp. 1–10, Nov. 2021. [Online]. Available: <https://media.neliti.com/media/publications/389246-none-0745391c.pdf>

- 
- [3] J. J. Hidayat, C. Setyowati, and A. P. Werdana, "Analisis Penyakit pada Daun Padi Menggunakan VGG-16 Transfer Learning dan Teknik Segmentasi K-Mean," *Jurnal Media Infotama*, vol. 21, no. 1, pp. 98–104, Apr. 2025. [Online]. Available: <https://jurnal.unived.ac.id/index.php/jmi/article/view/7313>
- [4] Hidayat, J. J., Setyowati, C., & Werdana, A. P. (2025, May). Perancangan sistem prediksi penyakit padatanaman padi berbasis image processing menggunakan algoritma VGG-16 transfer learning dan K-Means segmentation. *Journal of Practical Computer Science*, 5(1), 1–15. <https://jurnal.pelitabangsa.ac.id/index.php/jpcs/article/view/5759>
- [5] M. I. Burhanuddin, A. Syaifullah, S. A. P. Jaya, and M. G. Somoal, "Analisis Komparatif Model MobilenetV1 dan EfficientNetB0 dalam Klasifikasi Citra Empat Musim Menggunakan Transfer Learning," *JEKIN: Jurnal Teknik Informatika*, vol. 5, no. 2, 2025. [Online]. Available: <https://doi.org/10.58794/jekin.v5i2.1378>
- [6] U. Rahmalisa, M. Mardeni, Y. Yulisman, and A. Linarta, "Implementasi Algoritma Convolutional Neural Network untuk Deteksi Hama pada Tanaman Hidroponik," *Jurnal Informatika*, vol. 15, no. 2, 2023. [Online]. Available: <https://informatika.universitasdumai.ac.id/index.php/path/article/view/622>
- [7] A. B. Wicaksono and R. P. Hartati, "Analisis Performa Arsitektur CNN InceptionV3 dan VGG16 dalam Klasifikasi Deteksi Kanker Otak," *JUPI: Jurnal Ilmiah Penelitian dan Pembelajaran Informatika*, vol. 10, no. 2, pp. 6090–6097, 2024. [Online]. Available: <https://doi.org/10.29100/jupi.v10i2.6090>
- [8] Mariana Purba, "Klasifikasi Jenis Daun Tanaman Tropis Menggunakan Model ResNet50 Berbasis Transfer Learning dengan Dataset Tropical Plant Leaf", *JCOSIS*, vol. 2, no. 2, pp. 51–57, Oct. 2025.
- [9] "COMPUTER VISION MODERN", *yayasanpat*, vol. 11, no. 1, Sep. 2025, Accessed: Nov. 23, 2025. [Online]. Available: <https://penerbit.stekom.ac.id/index.php/yayasanpat/article/view/635>
- [10] S. Supandi, A. Ghofur, and F. Santoso, "Deteksi Kualitas Rumput Laut Menggunakan Metode Convolution Neural Network (CNN) Berdasarkan Citra Digital (Studi Kasus: Desa Alasmalang Kecamatan Raas Sumenep)," *RABIT: Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 10, no. 2, pp. 604–614, Jul. 2025. [Online]. Available: <https://jurnal.univrab.ac.id/index.php/rabit/article/download/6273/2441>
- [11] Ninuk wiliyani, T. . Khawa, dan S. Ramli, "Peningkatan Kontras Pada PreProcessing Gambar Permukaan Solar Panel dengan Histogram", *InnoTech*, vol. 2, no. 1, Jan 2025.
- [12] E. S. Budi, A. N. Chan, P. P. Alda, and M. A. F. Idris, "Optimasi Model Machine Learning untuk Klasifikasi dan Prediksi Citra Menggunakan Algoritma Convolutional Neural Network," *RESOLUSI: Rekayasa Teknik Informatika dan Informasi*, vol. 4, no. 5, pp. 502–509, May 2024. [Online]. Available: <https://djournals.com/resolusi>
-