

Penerapan *Convolutional Neural Network* Dalam Klasifikasi Daun Tanaman Obat Menggunakan Pendekatan *Transfer Learning*

Herliya Yolanda^{1*}, Lukman Hakim², Satriansyah³, Tri Hasanah Bimastari Aviani⁴

^{1,2,3,4} Informatika, Fakultas Ilmu Teknik, Universitas Bina Insan, Indonesia

Info Artikel

Riwayat Artikel:

Diterima : **12 Mei 2025**

Direvisi : **26 Juli 2025**

Diterbitkan : **5 Desember 2025**

Kata Kunci:

Daun Tanaman Obat,

Klasifikasi,

VGG-16.

Keywords:

Medicinal Plant,

Classification,

VGG-16,

ABSTRAK

Daun tanaman obat memiliki nilai penting dalam bidang kesehatan dan pengobatan tradisional karena kandungan senyawa bioaktifnya yang dapat digunakan untuk mengobati berbagai penyakit. Namun, pengenalan dan klasifikasi daun tanaman obat sering kali menjadi tantangan karena perbedaan visual yang sulit dikenali secara manual. Kesalahan dalam identifikasi dapat menghambat pengembangan obat-obatan herbal dan berpotensi merugikan pengguna. Oleh karena itu, diperlukan metode yang efektif untuk mengklasifikasikan daun tanaman obat secara akurat. Dataset yang digunakan dalam penelitian ini terdiri dari 3.500 gambar yang dibagi menjadi data latih, validasi, dan uji. Proses pelatihan model dilakukan dengan menggunakan arsitektur VGG-16. Hasil dari pelatihan menunjukkan bahwa model mampu mencapai akurasi sebesar 97%. Evaluasi model dilakukan menggunakan *confusion matrix*, yang menunjukkan bahwa model dapat mengklasifikasikan dengan sangat baik antara 10 kelas dalam klasifikasi daun tanaman obat. Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan klasifikasi tanaman obat yang lebih efektif dan efisien dapat menjadikannya alat yang potensial untuk mendukung pengambilan keputusan dalam tugas klasifikasi daun obat. Penelitian ini tidak hanya berfokus pada pengembangan model, tetapi juga memberikan wawasan tentang pentingnya penggunaan teknologi deep learning dalam bidang kesehatan, khususnya dalam klasifikasi daun tanaman obat.

ABSTRACT

Medicinal plant leaves hold significant value in health and traditional medicine due to their bioactive compounds, which can be used to treat various diseases. However, identifying and classifying medicinal plant leaves remains challenging due to subtle visual differences that are difficult to recognize manually. Misidentification can hinder the development of herbal medicines and potentially pose risks to users. Therefore, an effective method is needed to accurately classify medicinal plant leaves. The dataset used in this study consists of 3,500 images, which are divided into training, validation, and test sets. The model training process is conducted using the VGG16 architecture, which is known for its effectiveness in feature extraction from images. The training results indicate that the model achieves an accuracy of 97%. Model evaluation is performed using a confusion matrix, which demonstrates that the model effectively distinguishes between 10 classes of medicinal plant leaves. The findings of this study are expected to contribute to the development of a more effective and efficient medicinal plant classification system, making it a potential tool to support decision-making in medicinal leaf classification tasks. This research not only focuses on model development but also highlights the importance of deep learning technology in healthcare, particularly in medicinal plant leaf classifications.

Penulis Korespondensi:

Herliya Yolanda,

Program Studi Informatika,

Universitas Bina Insan

Email: herliyayolanda6@gmail.com

This is an open access article under the [CC BY-SA](#) license



1. PENDAHULUAN

Kemajuan teknologi di era modern memungkinkan pengoptimalan dalam berbagai sektor, termasuk sektor pertanian, untuk meminimalisir kesalahan identifikasi penyakit pada tanaman dengan memanfaatkan teknologi Artificial Intelligence (AI). AI merupakan cabang ilmu komputer yang memungkinkan mesin (komputer) untuk

melakukan tugas seperti manusia, bahkan lebih efisien. Salah satu cabang AI adalah machine learning, yang terus berkembang menjadi deep learning [1] .

Daun tanaman obat memiliki nilai penting dalam bidang kesehatan dan pengobatan tradisional karena kandungan senyawa bioaktifnya yang dapat digunakan untuk mengobati berbagai penyakit [2], [3]. Namun, pengenalan dan klasifikasi daun tanaman obat sering kali menjadi tantangan karena perbedaan visual yang sulit dikenali secara manual. Kesalahan dalam identifikasi dapat menghambat pengembangan obat-obatan herbal dan berpotensi merugikan pengguna. Oleh karena itu, diperlukan metode yang efektif untuk mengklasifikasikan daun tanaman obat secara akurat.

Pemanfaatan teknologi informasi dan kecerdasan buatan (AI) telah menjadi landasan inovasi dalam berbagai bidang, termasuk sektor pertanian dan kesehatan, untuk mengatasi berbagai tantangan. Salah satu implementasi signifikan dari AI adalah deep learning berbasis citra, yang telah terbukti mampu melakukan identifikasi dan klasifikasi objek secara cepat, akurat, dan efisien [4]. Dalam konteks tanaman obat, teknologi ini dapat menjadi solusi untuk mengatasi kesulitan dalam pengenalan dan pengklasifikasian jenis daun tanaman obat yang memiliki karakteristik visual yang sering kali sulit dibedakan secara manual. Teknologi ini memungkinkan petani, peneliti, dan industri farmasi untuk memperoleh informasi yang relevan dengan lebih mudah, sehingga mendukung pengembangan obat berbasis herbal.

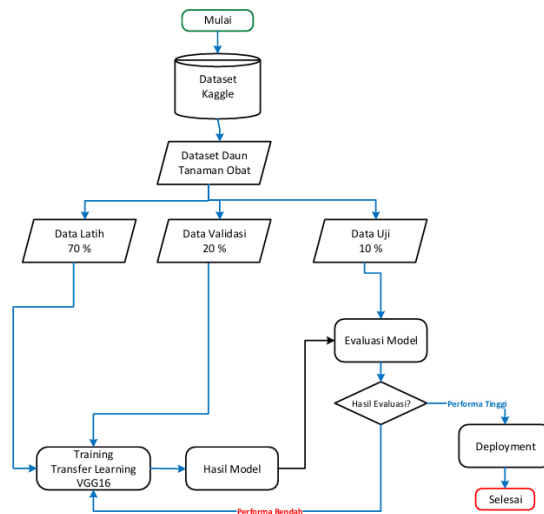
Convolutional Neural Network (CNN) merupakan metode deep learning yang dirancang khusus untuk analisis data berbasis citra [5], [6]. CNN bekerja dengan menggunakan lapisan-lapisan konvolusi untuk mengekstraksi pola-pola visual, mulai dari fitur sederhana seperti garis hingga fitur yang lebih kompleks seperti bentuk atau tekstur [7], [8]. Salah satu arsitektur CNN yang telah terbukti efektif adalah VGG16, yang memiliki struktur jaringan sederhana namun dalam, terdiri dari 16 lapisan yang berperan dalam mempelajari pola-pola visual secara hierarkis [9]. Kelebihan utama VGG16 terletak pada kemampuan ekstraksi fitur yang mendetail, meskipun struktur jaringannya lebih ringan dibandingkan arsitektur yang lebih kompleks [10]. Hal ini menjadikan VGG16 cocok untuk klasifikasi gambar pada dataset berskala kecil hingga menengah.

Dalam penelitian ini, sistem klasifikasi dirancang untuk mengidentifikasi jenis daun tanaman obat menggunakan metode CNN dengan arsitektur VGG16, yang diperkuat oleh pendekatan transfer learning. Pendekatan ini memungkinkan pemanfaatan bobot yang telah dipelajari dari dataset berskala besar sebelumnya, seperti ImageNet, untuk diterapkan pada dataset baru dengan jumlah data yang terbatas. Transfer learning tidak hanya mempercepat proses pelatihan model, tetapi juga meningkatkan akurasi dalam mengenali pola-pola spesifik pada citra daun tanaman obat. Dengan sistem ini, diharapkan proses identifikasi daun tanaman obat dapat dilakukan secara lebih efisien, mendukung eksplorasi kandungan bioaktif, dan memberikan kontribusi signifikan terhadap inovasi dalam pengembangan produk berbasis tanaman obat.

Penelitian ini menggunakan metode *Convolutional Neural Network* (CNN) dengan arsitektur VGG16 dan pendekatan *transfer learning* untuk melakukan klasifikasi daun tanaman obat. Pemanfaatan *Transfer learning* digunakan untuk meningkatkan akurasi model pada kondisi jumlah data yang terbatas, dengan memanfaatkan bobot dari model yang telah dilatih sebelumnya. Dalam tahap evaluasi, kinerja model akan diukur menggunakan metrik akurasi, presisi, recall, dan F1-score untuk menilai efektivitas sistem klasifikasi yang dibangun.

2. METODE PENELITIAN

Rancangan penelitian digunakan untuk mengklasifikasi daun tanaman obat dengan pendekatan *transfer learning* arsitektur VGG16 . Rancangan penelitian dapat dilihat pada Gambar 1.



Gambar 1. Rancangan Penelitian

Gambar 1. merupakan rancangan sistem klasifikasi daun tanaman obat dengan pendekatan *transfer learning* menggunakan arsitektur VGG16.

2.1 Dataset

Dataset diambil pada website penyedia dataset public yaitu kaggle, dataset berjumlah 3.500 data, dibagi menjadi 2450 data latih (train), 700 untuk data validasi, dan 350 data uji (test) serta jumlah kelas sebanyak 10 (Sepuluh) kelas antara lain daun belimbing wuluh, jambu biji, jeruk nipis, kemangi, lidah buaya, nangka, pandan, pepaya, seledri, dan daun sirih. Sampel dataset dapat dilihat pada Tabel 1 berikut ini.

Tabel 1. Sampel Data

No	Nama Kelas	Sampel Data	Data Latih	Data Val	Data Uji
1	Belimbing Wuluh		245	70	35
2	Jambu Biji		245	70	35
3	Jeruk Nipis		245	70	35
4	Kemangi		245	70	35
5	Lidah Buaya		245	70	35
6	Nangka		245	70	35
7	Pandan		245	70	35
8	Pepaya		245	70	35
9	Seledri		245	70	35
10	Sirih		245	70	35
TOTAL			2450	700	350

2.2 Training dengan arsitektur VGG16

Pada tahap *training* menggunakan arsitektur VGG16. VGG16 merupakan salah satu arsitektur *deep learning* yang terdiri dari 16 lapisan, yaitu 13 lapisan konvolusional dan 3 lapisan *fully connected* [11]. Arsitektur VGG-16 dirancang dengan pendekatan berbasis blok, di mana setiap blok terdiri dari beberapa lapisan konvolusional berukuran 3 x 3 dengan langkah (stride) 1. Fungsi aktivasi ReLU digunakan untuk memastikan hanya nilai-nilai positif yang diteruskan ke lapisan berikutnya, sehingga membantu proses pembelajaran menjadi lebih efisien. Setelah setiap blok konvolusi, diterapkan lapisan *max pooling* berukuran 2 x 2 dengan stride 2 untuk mengurangi dimensi fitur dan kompleksitas komputasi. Selain itu, pada bagian *fully connected*, dapat ditambahkan teknik *dropout* untuk meminimalkan risiko *overfitting* dan mempercepat proses pelatihan model. Struktur model pada penelitian ini dapat dilihat pada Gambar 2.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[None, 150, 150, 3]	0
block1_conv1 (Conv2D)	(None, 150, 150, 64)	1792
block1_conv2 (Conv2D)	(None, 150, 150, 64)	36928
block1_pool (MaxPooling2D)	(None, 75, 75, 64)	0
block2_conv1 (Conv2D)	(None, 75, 75, 128)	73856
block2_conv2 (Conv2D)	(None, 75, 75, 128)	147584
block2_pool (MaxPooling2D)	(None, 37, 37, 128)	0
block3_conv1 (Conv2D)	(None, 37, 37, 256)	295168
block3_conv2 (Conv2D)	(None, 37, 37, 256)	590880
block3_conv3 (Conv2D)	(None, 37, 37, 256)	590880
block3_pool (MaxPooling2D)	(None, 18, 18, 256)	0
block4_conv1 (Conv2D)	(None, 18, 18, 512)	1180160
block4_conv2 (Conv2D)	(None, 18, 18, 512)	2359808
block4_conv3 (Conv2D)	(None, 18, 18, 512)	2359808
block4_pool (MaxPooling2D)	(None, 9, 9, 512)	0
block5_conv1 (Conv2D)	(None, 9, 9, 512)	2359808
block5_conv2 (Conv2D)	(None, 9, 9, 512)	2359808
block5_conv3 (Conv2D)	(None, 9, 9, 512)	2359808
block5_pool (MaxPooling2D)	(None, 4, 4, 512)	0
flatten (Flatten)	(None, 8192)	0
dense (Dense)	(None, 512)	4194816
dense_1 (Dense)	(None, 10)	5130
Total params: 18,914,634		
Trainable params: 18,914,634		
Non-trainable params: 0		

Gambar 2. Struktur Model VGG16

Berdasarkan Gambar 2, arsitektur VGG16 terdiri dari 5 blok utama yang masing-masing berisi beberapa layer konvolusi (Conv2D) dengan ukuran kernel 3x3 dan dilanjutkan oleh satu layer max pooling 2x2 untuk mengurangi dimensi spasial. Model ini menerima input citra berukuran 150x150 piksel dengan 3 channel warna (RGB). Blok pertama terdiri dari dua layer konvolusi dengan 64 filter, diikuti oleh max pooling, menghasilkan output berukuran 75x75x64. Blok kedua menggunakan dua layer konvolusi dengan 128 filter dan pooling, menghasilkan output 37x37x128. Blok ketiga memiliki tiga layer konvolusi dengan 256 filter, dilanjutkan pooling dengan output 18x18x256. Blok keempat dan kelima masing-masing memiliki tiga layer konvolusi dengan 512 filter, dengan output akhir sebelum fully connected layer adalah 4x4x512. Output ini kemudian diratakan (flatten) menjadi vektor 1 dimensi berukuran 8192, yang dihubungkan ke layer dense dengan ukuran filter 512, dan akhirnya ke layer output dengan 10 filter sebagai representasi dari jumlah kelas. Total parameter yang dilatih dalam model ini adalah 18.914.634, menunjukkan bahwa seluruh layer bersifat trainable, sehingga setiap bagian jaringan berperan aktif dalam proses pembelajaran.

2.3 Evaluasi Model

Setelah model dilatih, dilakukan evaluasi menggunakan data uji untuk mengukur performa model. Evaluasi model menggunakan *confusion matrix*. *Confusion matrix* menyajikan visualisasi tingkat kesalahan atau kebingungan algoritma dalam membedakan tiap kelas yang berbeda, serta bersifat independen terhadap jenis algoritma klasifikasi yang digunakan. [12]. *Confusion matrix* dapat dilihat pada Tabel 2 [13].

Tabel 2. Confusion Matrix

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

Evaluasi dengan confusion matrix menghasilkan nilai akurasi, presisi, *recall*, dan *F1-Score* untuk mengukur performa model.

1. Akurasi

Akurasi merupakan perbandingan antara jumlah prediksi yang benar dengan total keseluruhan data. Nilai akurasi dapat dihitung menggunakan Persamaan (1) [14].

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

2. Presisi

Presisi merupakan rasio antara jumlah prediksi positif yang benar dengan seluruh prediksi yang diklasifikasikan sebagai positif. Nilai presisi dapat dihitung menggunakan Persamaan (2) [14].

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

3. Recall

Recall atau sensitivitas merupakan ukuran seberapa banyak data positif yang berhasil diprediksi dengan tepat dari seluruh data positif yang ada. Nilai recall dihitung menggunakan Persamaan (3) [14].

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

4. F1-Score

F1-Score merupakan rata-rata harmonik dari Presisi dan Recall [15]. Nilai *F1-Score* dapat dihitung menggunakan persamaan (4) [14].

$$\text{F1-Score} = 2 * \frac{(\text{Presisi} * \text{Recall})}{(\text{Presisi} + \text{Recall})} \quad (4)$$

2.4 Deployment

Setelah mendapat model dengan performa tinggi maka model akan siap dipakai dan dibundel dalam ekstensi .h5.

3. HASIL DAN ANALISIS

3.1 Hasil Pelatihan Klasifikasi Daun Tanaman Obat

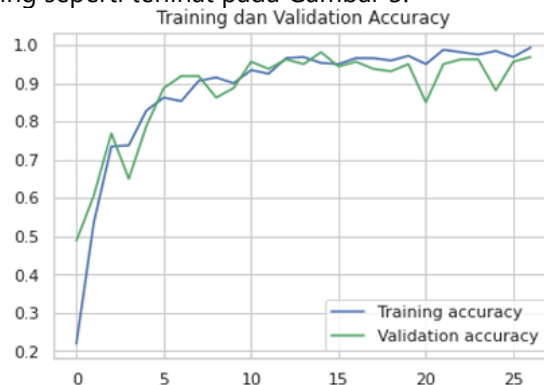
Pada penelitian ini, proses melatih model pembelajaran mendalam menggunakan dataset daun tanaman obat selama 100 epoch. pelatihan model pembelajaran mendalam dilakukan menggunakan dataset daun tanaman obat selama 100 epoch. Pelatihan ini bertujuan untuk meningkatkan akurasi model dalam mengklasifikasikan data secara tepat, sekaligus menurunkan nilai loss yang menunjukkan seberapa baik model mampu menyesuaikan diri dengan data pelatihan. Hasil pelatihan pada saat 5 epoch terakhir dapat dilihat hasilnya pada Tabel 3.

Tabel 3. Tampilan 5 Terakhir dari Hasil Pelatihan

Epoch	Akurasi Training	Loss Training	Accuracy Validasi	Loss Validasi
1	0.9781	0.0710	0.9812	0.0497

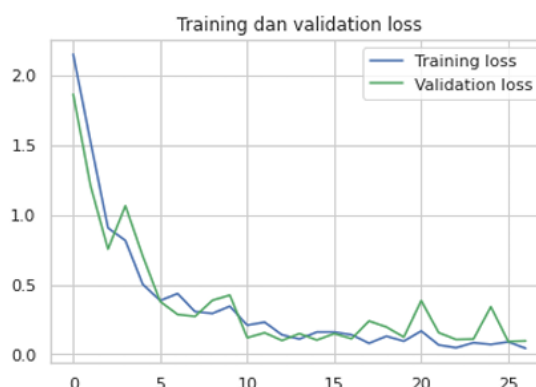
Epoch	Akurasi Training	Loss Training	Accuracy Validasi	Loss Validasi
2	0.9719	0.0709	0.9875	0.0438
3	0.9686	0.1049	0.9750	0.0674
4	0.9771	0.0942	0.9563	0.1272
5	0.9673	0.0975	0.9563	0.1647

Pada lima epoch terakhir dari 100 epoch pelatihan, performa model menunjukkan tren akurasi yang tetap tinggi, meskipun terdapat beberapa indikasi fluktuasi dalam nilai loss. Akurasi pelatihan pada epoch ke-96 hingga ke-100 berkisar antara 96.73% hingga 97.81%, dengan nilai tertinggi tercapai pada epoch ke-96 sebesar 97.81%. Namun, pada epoch ke-98 dan ke-100, terjadi sedikit penurunan akurasi pelatihan, masing-masing menjadi 96.86% dan 96.73%. Setelah melakukan proses pelatihan atau *training*, model yang telah dibentuk akan diujikan dengan data testing seperti terlihat pada Gambar 3.



Gambar 3. Hasil Akurasi Data Train

Gambar 3. menunjukkan perbandingan akurasi antara data pelatihan dan validasi selama proses pelatihan model pada 25 epoch. Pada awal pelatihan (epoch < 5), akurasi pelatihan meningkat secara signifikan, menunjukkan bahwa model dengan cepat belajar dari data pelatihan. Akurasi validasi juga meningkat dengan pola yang serupa, meskipun mengalami fluktuasi yang lebih besar. Setelah melewati epoch ke 5, baik akurasi pelatihan maupun validasi mulai mendekati nilai maksimum, yaitu sekitar 0,95 hingga 1,0. Fluktuasi kecil pada akurasi validasi setelah epoch ke-10 menunjukkan adanya variasi kinerja model pada data validasi, namun secara keseluruhan tetap konsisten dengan tren akurasi pelatihan. Pada epoch terakhir, kedua kurva hampir sejajar dan menunjukkan akurasi mendekati 1,0, yang mencerminkan kinerja model yang sangat baik. Hasil ini menunjukkan bahwa model berhasil mempelajari pola-pola dalam data tanpa mengalami overfitting yang signifikan, mengingat akurasi pelatihan dan validasi relatif stabil dan serupa. Namun, evaluasi lebih lanjut dengan data pengujian diperlukan untuk memastikan generalisasi model terhadap data baru yang tidak terlihat selama pelatihan. Evaluasi selanjutnya adalah menggunakan grafik loss pada pelatihan seperti pada Gambar 4.



Gambar 4. Hasil Loss Data Train

Gambar 4 menunjukkan perbandingan antara *loss* pelatihan dan *loss* validasi selama 25 epoch. Pada awal pelatihan (epoch 0–5), terdapat penurunan yang signifikan pada *loss* pelatihan dan validasi, dari nilai awal lebih dari 2,0 menuju nilai di bawah 0,5. Hal ini menunjukkan bahwa model dengan cepat belajar mengurangi kesalahan dalam memprediksi output pada data pelatihan dan validasi. Setelah epoch ke-5, laju penurunan

menjadi lebih lambat, dengan nilai loss mendekati nol. Mulai dari epoch ke-10 hingga akhir pelatihan, loss pelatihan dan validasi berada pada level yang stabil dan rendah, dengan fluktuasi kecil pada loss validasi. Fluktuasi ini mencerminkan variabilitas kinerja model pada data validasi, yang bisa jadi disebabkan oleh keragaman dalam dataset validasi itu sendiri. Secara umum, grafik ini menunjukkan bahwa model memiliki performa pelatihan yang baik, ditandai dengan loss pelatihan yang konsisten rendah. Tidak ada tanda-tanda overfitting yang signifikan, karena loss validasi tetap sejalan dengan loss pelatihan sepanjang pelatihan. Untuk mengonfirmasi kinerja ini, evaluasi lebih lanjut terhadap data pengujian diperlukan untuk memastikan kemampuan model dalam menggeneralisasi pada data yang belum terlihat sebelumnya.

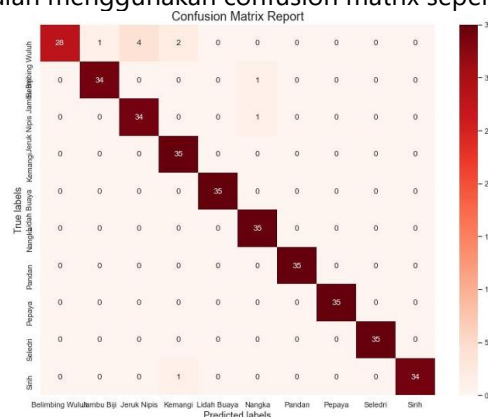
3.2 Evaluasi Hasil Klasifikasi

Setelah melakukan proses pelatihan atau training, model ini akan diujikan dengan data testing dan menghasilkan classification Report yang terdiri dari precision, recall, f1-score dan accuracy seperti terlihat pada Gambar 5.

	precision	recall	f1-score	support
Belimbing Wuluh	0.97	0.92	0.95	38
Jambu Biji	1.00	0.97	0.99	37
Jeruk Nipis	0.89	0.94	0.91	34
Kemangi	0.92	0.97	0.94	34
Lidah Buaya	1.00	1.00	1.00	36
Nangka	1.00	0.95	0.97	38
Pandan	1.00	1.00	1.00	36
Pepaya	1.00	0.97	0.99	37
Seledri	0.97	1.00	0.99	35
Sirih	0.97	1.00	0.99	35
accuracy			0.97	360
macro avg	0.97	0.97	0.97	360
weighted avg	0.97	0.97	0.97	360

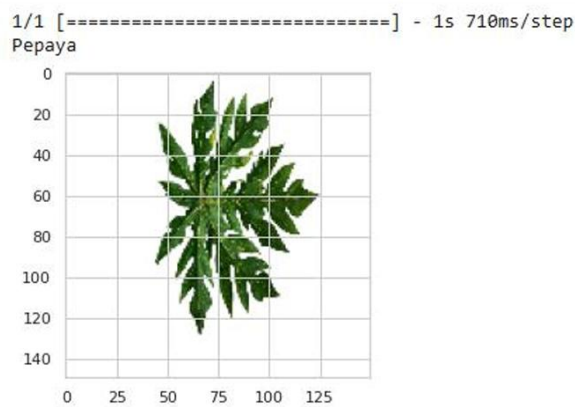
Gambar 5. Classification Report

Berdasarkan Gambar 5, menunjukkan performa yang sangat baik dalam mengklasifikasikan berbagai jenis daun tanaman obat. Secara keseluruhan, model mencapai akurasi sebesar 97%, dengan nilai precision, recall, dan f1-score rata-rata (*macro average*) yang juga sebesar 97%. Hal ini menunjukkan bahwa model memiliki kemampuan yang cukup baik dalam mengidentifikasi setiap kelas dengan tingkat kesalahan yang rendah. Selain itu, *weighted average* yang mencapai nilai yang sama dengan *macro average* mengindikasikan bahwa distribusi kelas dalam dataset relatif seimbang, sehingga tidak ada bias signifikan terhadap kelas tertentu. Evaluasi selanjutnya adalah menggunakan confusion matrix seperti pada Gambar 6.



Gambar 6. Confusion Matrix

Berdasarkan Gambar 5, model menunjukkan performa yang sangat baik dengan mayoritas prediksi berada pada diagonal utama, yang mengindikasikan prediksi yang benar. Sebagian besar kelas memiliki Tingkat klasifikasi yang sempurna, dengan jumlah prediksi yang benar hampir mendekati jumlah sampel sebenarnya. Namun, terdapat beberapa kelas yang mengalami kesalahan klasifikasi dalam jumlah kecil, terutama pada kelas Belimbing Wuluh dan Sirih. Hasil pengujian dengan data gambar langsung secara individual dapat dilihat pada Gambar 7.



Gambar 7. Implementasi pada Data Baru

Berdasarkan Gambar 7, menunjukkan hasil klasifikasi citra daun dengan label "Pepaya." Berdasarkan representasi visual, citra tersebut memperlihatkan daun dengan ciri khas daun pepaya, termasuk bentuk yang menjari dengan bagian tepi daun yang bergerigi. Proses klasifikasi dilakukan melalui suatu model pembelajaran mesin yang memprediksi label dengan tingkat akurasi yang didukung oleh waktu pemrosesan sebesar 1s 710ms per langkah.

4. KESIMPULAN

Hasil penelitian menunjukkan bahwa model klasifikasi yang dibangun memiliki performa sangat baik dengan tingkat akurasi yang tinggi, mencerminkan kemampuan dalam memprediksi secara akurat dan meminimalkan kesalahan dalam identifikasi jenis daun tanaman obat. Dengan akurasi mencapai 97%, model menunjukkan keandalan tinggi dalam membedakan antar kelas, menjadikannya alat yang potensial untuk mendukung proses klasifikasi daun obat. Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi arsitektur Convolutional Neural Network (CNN) lain seperti ResNet, Inception, atau DenseNet untuk mengetahui sejauh mana model-model tersebut dapat meningkatkan performa klasifikasi. Selain itu, penerapan teknik *fine-tuning* pada beberapa lapisan akhir model pralatih juga dapat menjadi alternatif untuk mengoptimalkan hasil, khususnya jika tersedia dataset yang lebih besar dan beragam.

REFERENSI

- [1] R. Soekarta, N. Nurdjan, and A. Syah, "Klasifikasi Penyakit Tanaman Tomat Menggunakan Metode Convolutional Neural Network (CNN)," *Insect (Informatics Secur. J. Tek. Inform.*, vol. 8, no. 2, pp. 143–151, 2023.
- [2] Sarno, "Pemanfaatan Tanaman Obat (Biofarmaka) Sebagai Produk unggulan Masyarakat Desa Depok Banjarnegara," *Abdimas Unwahas*, vol. 4, no. 2, pp. 73–78, 2019.
- [3] W. Irma, F. Farida, H. Purwanto, R. T. Syurya, and A. Maltia, "PKM Optimalisasi Sumber Belajar Rumah Tanam Herbal Medicine Berbasis Teknologi Iot Di Sekolah Alam Rumbai," *J. Pengabd. Untuk Mu NegeRI*, vol. 8, no. 3, pp. 300–306, 2024.
- [4] N. Nova *et al.*, "Systematic Review: Pemanfaatan Deep Learning untuk Diagnosis Penyakit Menggunakan MRI," *J. Penelit. Inov.*, vol. 5, no. 2, pp. 839–852, 2025.
- [5] R. Ardiansyah, A. C. Fajarulloh, B. Aprilio, and S. Putra, "Penggunaan CNN Untuk Menentukan Jumlah Kalori Pada Sayuran Dan Buah Menggunakan Image Processing," *Pros. Semin. Nas. Teknol. DAN SAINS TAHUN*, vol. 4, pp. 571–579, 2025.
- [6] A. Maya, K. Putri, and A. F. Rozi, "Implementasi Convolutional Neural Network Dalam Menentukan Tingkat Kematangan Mentimun dan Tomat Berdasarkan Warna Kulit," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 5, pp. 10388–10394, 2024.
- [7] I. Nurhasanah, "Klasifikasi Tingkat Kematangan Buah Anggur Transfigurasi Menggunakan Metode Convolutional Neural Network," Universitas Hasanuddin, 2023.

- [8] D. Kurniadi, R. M. Shidiq, and A. Mulyani, "Perbandingan Penggunaan Optimizer dalam Klasifikasi Sel Darah Putih Menggunakan Convolutional Neural Network," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 14, no. 1, pp. 77–86, 2025, doi: 10.22146/jnteti.v14i1.17162.
- [9] Andrianto, I. Tahyudin, and G. Karyono, "Perbandingan Efficientnet , Visual Geometry Group 16, dan Residual Network 50 Untuk Klasifikasi Kendaraan Bermotor," *Build. Informatics, Technol. Sci.*, vol. 6, no. 3, pp. 1995–2004, 2024, doi: 10.47065/bits.v6i3.6450.
- [10] R. Klangbunrueang, P. Pookduang, W. Chansanam, and T. Lunrasri, "AI-Powered Lung Cancer Detection: Assessing VGG16 and CNN Architectures for CT Scan Image Classification," *Informatics*, vol. 12, no. 18, pp. 1–30, 2025, doi: <https://doi.org/10.3390/informatics12010018>.
- [11] N. L. W. Rahayu, N. Gunantara, and M. Sudarma, "Klasifikasi Jajanan Khas Bali Untuk Preservasi Pengetahuan Kuliner Lokal Menggunakan Arsitektur VGG-16," *SINTECH J. |*, vol. 7, no. 1, pp. 1–14, 2024.
- [12] Arnita, F. Marpaung, F. Aulia, Nita Suryani, and R. C. Nabila, *Computer Vision Dan Pengolahan Citra Digital*. Surabaya, Jawa Timur: Pustaka Aksara, 2022.
- [13] N. Fadhillah and A. Desiani, "Klasifikasi Nyeri Punggung Bawah Menggunakan Algoritma K-Nearest Neighbor Dan Support Vector Machine," *J. Tek. Elektro dan Komputasi Vol.*, vol. 7, no. 1, pp. 91–98, 2025.
- [14] A. Fahrizal, D. Rusirawan, and L. Lidyawati, "Pemodelan Produksi Energi Pembangkit Listrik Tenaga Surya 1000 WP dengan Algoritma Naive Bayes," *J. Tekno Insentif*, vol. 16, no. 2, pp. 105–118, 2022.
- [15] A. H. Bik, F. T. Anggraeny, and E. Y. Puspaningrum, "Klasifikasi Penyakit Ginjal Menggunakan Algoritma Hibrida CNN-ELM," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 3, pp. 3836–3844, 2024.