

PENGEMBANGAN SISTEM DETEKSI REMPAH-REMPAH OTOMATIS MENGGUNAKAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)

Pristya Haliza Ramadhanti^{1*}, Clara Dian Anggraini², Mochamad Wahyudi³, Sumanto⁴, Ade Surya Budiman⁵

^{1, 2, 3, 4, 5}Program Studi Informatika, Fakultas Teknik dan Informatika
Universitas Bina Sarana Informatika, Indonesia

ABSTRACT

Penelitian ini membahas pembangunan model Convolutional Neural Network (CNN) untuk klasifikasi gambar rempah-rempah secara otomatis. Proses pelatihan dilakukan di Google Colab, dengan menambahkan variasi gambar (augmentasi data) agar model bisa mengenali lebih banyak bentuk. Dataset gambar diproses menggunakan *ImageDataGenerator* dengan augmentasi data berupa rotasi, zoom, flipping, dan translasi guna meningkatkan variasi data latih. Model CNN dibangun secara *sequential* dengan lapisan Conv2D, MaxPooling2D, Flatten, Dense, dan Dropout, serta dilatih menggunakan fungsi *Categorical Cross Entropy* dan aktivasi *Softmax*. Hasil pelatihan menunjukkan akurasi data latih mencapai 95–99%, sementara akurasi validasi meningkat hingga 65%. Meskipun terjadi lonjakan *loss* di awal epoch, performa model membaik pada epoch selanjutnya. Evaluasi prediksi terhadap data uji menunjukkan akurasi tinggi dengan tingkat kepercayaan di atas 90% untuk sebagian besar kelas. Namun, confusion matrix mengindikasikan adanya kesulitan dalam membedakan kelas yang memiliki kemiripan visual tinggi, seperti jahe, temulawak, dan lengkuas. Secara keseluruhan, model menunjukkan performa yang baik dan dapat dijadikan dasar dalam pengembangan sistem pengenalan rempah berbasis citra.

Kata kunci: Rempah-rempah, Klasifikasi gambar, CNN, Augmentasi, Confusion Matrix.

ABSTRACT

This research discusses the construction of a Convolutional Neural Network (CNN) model for automatic classification of spice images. The training process is carried out on Google Colab, by adding image variations (data augmentation) so that the model can recognize more shapes. The image dataset is processed using ImageDataGenerator with data augmentation in the form of rotation, zoom, flipping, and translation to increase the variety of training data. The CNN model is built sequentially with Conv2D, MaxPooling2D, Flatten, Dense, and Dropout layers, and trained using Categorical Cross Entropy function and Softmax activation. The training results show that the accuracy of the training data reaches 95-99%, while the validation accuracy increases to 65%. Although there was a spike in loss in the early epochs, the model performance improved in the later epochs. Evaluation of predictions against test data showed high accuracy with confidence levels above 90% for most classes. However, the confusion matrix indicated difficulty in distinguishing classes with high visual similarity, such as ginger, curcuma xanthorrhiza and sand ginger. Overall, the model performed well and can be used as a basis for developing an image-based spice recognition system.

Keywords: Spices, Image classification, CNN, Augmentation, Confusion Matrix

1. PENDAHULUAN

Rempah-rempah adalah salah satu kekayaan alam Indonesia yang punya peran penting dalam kehidupan sehari-hari, terutama untuk menambah aroma, rasa, dan warna

pada masakan[1]. Sejak dulu, rempah-rempah memiliki nilai ekonomi tinggi dan menjadi komoditas penting dalam perdagangan global, terutama oleh bangsa Eropa. Meskipun punya nilai tinggi dan sering digunakan, beberapa jenis rimpang masih susah dibedakan karena bentuknya mirip. Biasanya, orang mengenalnya lewat gambar atau meminta bantuan ahli, tapi cara ini butuh waktu yang cukup lama[2].

Metode *Convolutional Neural Network* (CNN) termasuk salah satu metode dalam deep learning yang merupakan pengembangan dari *Artificial Neural Network* (ANN). CNN biasanya digunakan untuk pengolahan data berupa gambar Metode *Convolutional Neural Network* (CNN) digunakan untuk mempermudah klasifikasi rempah-rempah secara visual dan meningkatkan akurasi hasilnya[3].

Literatur yang relevan menunjukkan bahwa metode *Convolutional Neural Network* (CNN) dalam pemrosesan citra dapat menjadi solusi untuk mengatasi kesulitan dalam mengidentifikasi rempah-rempah. Salah satu penelitian berjudul “Deteksi Jenis Rempah-Rempah Menggunakan Metode *Convolutional Neural Network* Secara *RealTime*” ditunjukkan oleh Sanjaya dan Nurraharjo. Penelitian ini menggunakan CNN untuk mengenali 12 jenis rempah-rempah dengan jumlah data yang cukup besar[4]. Hasilnya menunjukkan bahwa CNN memiliki potensi untuk digunakan dalam pengenalan rempah secara *real-time*, meskipun rata-rata akurasi hanya mencapai 60%. Hal ini menegaskan bahwa pemilihan dataset yang tepat sangat penting agar hasilnya lebih akurat. Penelitian lain yang berjudul “Implementasi *Convolutional Neural Network* Untuk Klasifikasi Tanaman Rimpang Secara Virtual” oleh Darmatasia dan Syafar juga menggunakan CNN untuk mengklasifikasikan rempah-rempah khas Indonesia. Hasilnya, akurasi saat pelatihan sangat tinggi yaitu 99%, namun saat diuji kembali (validasi), akurasi turun menjadi 60%[5]. Hal ini menunjukkan bahwa meskipun CNN efektif, tetap ada tantangan karena tampilan rempah-rempah yang mirip satu sama lain. Penelitian ini menegaskan pentingnya peran CNN dalam membantu membedakan rempah-rempah yang memiliki bentuk atau warna yang hampir sama.

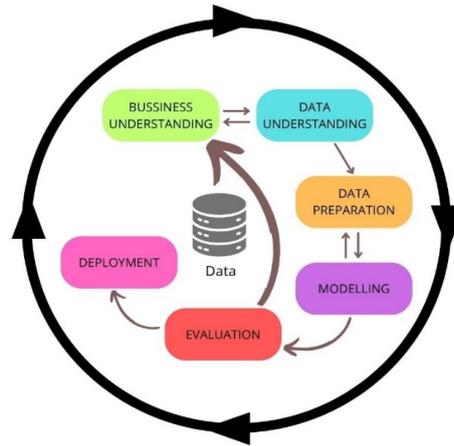
Dari pembahasan yang telah disampaikan, penelitian ini memiliki tujuan untuk menciptakan sebuah sistem yang dapat memudahkan pengguna dalam mengenali jenis rempah, terutama rimpang, secara cepat, mudah, dan efisien.

Tabel 1. Perbandingan Penelitian Terdahulu

No	Penulis (Tahun)	Metode	Objek Klasifikasi	Akurasi	Kelebihan	Kelemahan
1	Sanjaya & Nurraharjo (2023)	CNN	12 jenis rempah	60%	Real-time, jumlah data besar	Akurasi rendah karena variasi data
2	Darmatasia & Syafar (2023)	CNN	Tanaman rimpang	99% (train), 60% (val)	Akurasi tinggi saat pelatihan	Overfitting
3	Penelitian ini (2025)	CNN + Transfer Learning	Jahe, kunyit, kencur, lengkuas, temulawak	95–99% (train), 65% (val), >90% (test)	Akurasi uji tinggi, augmentasi dan fine-tune	Salah klasifikasi pada kelas yang mirip
4	Shorten & Khoshgoftaar (2019)	CNN + Augmentasi	Beragam dataset citra	-	Ulasan lengkap teknik augmentasi data	Tidak fokus pada rempah atau rimpang
5	Mohanty (2016)	CNN (AlexNet, GoogLeNet)	Penyakit tanaman berbasis citra daun	Hingga 99.35%	Deteksi visual tanaman akurat	Bergantung pada kualitas dataset
6	Minh (2019)	CNN	Tanaman obat tradisional (Asia Tenggara)	>90%	Cakupan spesies tanaman lokal	Belum diuji pada lingkungan riil
7	Kamilaris & Prenafeta-Boldú (2018)	Deep Learning	Aplikasi pertanian (umum)	-	Survei lengkap bidang pertanian & DL	Hanya survei, bukan eksperimen

2. METODE PENELITIAN

Penelitian ini menerapkan metode CRISP-DM (*Cross-Industry Standard Process for Data Mining*), yang sering digunakan sebagai acuan dalam menyelesaikan berbagai permasalahan di perusahaan maupun lembaga riset. Metode ini terdiri dari enam langkah utama, yaitu *Understanding business*, *understanding data*, *preparation*, *modeling*, *evaluation*, dan *deployment*[6]. Dalam konteks penelitian ini, metode CRISP-DM digunakan untuk membangun sistem yang mampu mengenali jenis-jenis rempah, khususnya rimpang, dengan bantuan teknologi pembelajaran mesin (*machine learning*), yaitu *Convolutional Neural Network* (CNN).



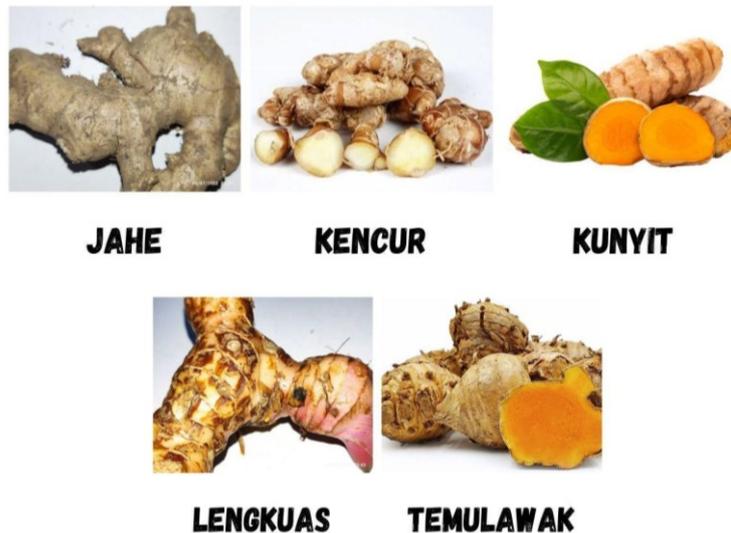
Gambar 1. Metodologi penelitian

A. Business Understanding

Tujuan dari penelitian ini adalah menciptakan sistem berbasis kecerdasan buatan yang mampu mengenali jenis rempah secara otomatis dari gambar, sehingga proses identifikasi bisa dilakukan dengan lebih cepat dan akurat.

B. Data Understanding

Penelitian ini menggunakan data berupa gambar lima jenis rempah, dataset berupa kumpulan gambar rempah-rempah jenis rimpang seperti jahe, kunyit, lengkuas, kencur dan temulawak yang diperoleh dari situs Kaggle.com.



Gambar 2. Dataset rempah

C. Data Preparation

Tahap ini berfokus pada pengolahan data agar siap digunakan dalam pelatihan model. Jumlah total gambar yang digunakan dalam dataset adalah 1.800 gambar. Gambar-gambar yang telah dikumpulkan diubah ukurannya menjadi 224x224 piksel guna menyamakan dimensi input model CNN dan mempercepat proses pelatihan. Setiap citra dinormalisasi dengan membagi nilai pikselnya menjadi rentang 0–1. Dataset kemudian dipisahkan menjadi dua bagian: Data pelatihan (data training) sebanyak 80% dan Data validasi sebanyak 20%. Untuk meningkatkan variasi data dan mengurangi risiko overfitting, diterapkan teknik augmentasi gambar seperti rotation, zoom, dan flipping secara horizontal. Seluruh proses ini dilakukan menggunakan pustaka Python seperti TensorFlow, ImageDataGenerator, dan NumPy.

Tabel 2. Jumlah Dataset Rempah

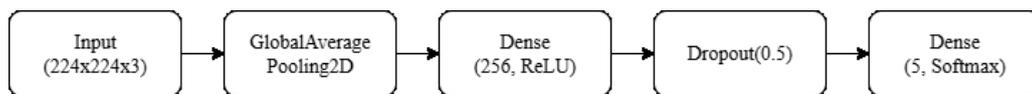
No	Nama Rimpang	Data
1	Jahe	360
2	Kencur	360
3	Kunyit	360
4	Lengkuas	360
5	Temulawak	360
Jumlah Data		1800

D. Modelling

Model klasifikasi dibangun dengan memanfaatkan arsitektur CNN menggunakan pendekatan transfer learning. Salah satu algoritma yang digunakan adalah Convolutional Neural Network (CNN), yang merupakan pengembangan dari Multilayer Perceptron (MLP) yaitu terdiri dari lapisan konvolusi, lapisan pooling, dan diakhiri dengan lapisan fully connected. CNN dirancang khusus untuk memproses data berdimensi dua, seperti citra dan sinyal suara. Arsitektur CNN terinspirasi dari cara kerja sel-sel saraf di otak manusia[7]. Dalam hal ini, digunakan model ResNet50 yang telah dilatih sebelumnya (pre-trained) pada dataset ImageNet. Model ini kemudian di fine-tune dengan menambahkan beberapa lapisan baru di bagian output untuk disesuaikan dengan lima kelas rempah yang ingin dikenali. Model dikompilasi menggunakan optimizer Adam, yang terkenal karena kemampuannya dalam mempercepat proses konvergensi, dan fungsi loss categorical crossentropy yang sesuai

untuk klasifikasi lebih dari dua kelas. Proses pelatihan dilakukan sebanyak 10 epoch menggunakan platform Google Colab.

Arsitektur CNN yang digunakan dalam penelitian ini menggabungkan model ResNet50 sebagai feature extractor, kemudian ditambahkan beberapa layer custom seperti GlobalAveragePooling2D, Dense dengan 256 neuron dan ReLU, Dropout 0.5, dan Dense output layer dengan Softmax 5 kelas. Diagram arsitektur model ditampilkan pada Gambar 3 sebagai berikut:



Gambar 3. Arsitektur Model CNN

E. *Evaluation*

Pada tahap evaluasi ini dilakukan dengan menggunakan berbagai metrik seperti akurasi, presisi, recall, F1-score, dan MCC (Matthews Correlation Coefficient). Selain itu, juga digunakan analisis confusion matrix untuk mengetahui jenis kesalahan apa saja yang muncul dari hasil prediksi model

F. *Deployment*

Tahap akhir adalah implementasi sederhana dari sistem klasifikasi. Sistem dikembangkan di Google Colab dan memungkinkan pengguna untuk mengunggah gambar rempah melalui antarmuka berbasis kode. Setelah gambar diunggah, sistem akan menampilkan hasil prediksi secara otomatis lengkap dengan nama kelas dan persentase keyakinan model terhadap gambar tersebut. Ke depannya, sistem ini dapat dikembangkan lebih lanjut ke dalam aplikasi berbasis web atau mobile agar lebih mudah diakses oleh masyarakat dan dapat dijalankan secara online.

3. HASIL DAN PEMBAHASAN

Proses dimulai dengan impor library yang dibutuhkan, seperti numpy, matplotlib.pyplot, dan modul dari tensorflow.keras seperti ImageDataGenerator, Sequential, serta berbagai jenis lapisan jaringan saraf tiruan Convolutional Neural Network (CNN). Setelah itu, pengaturan awal dan persiapan dataset dilakukan dengan menggunakan ImageDataGenerator yang tidak hanya mereskalakan piksel gambar ke rentang [0,1], tetapi juga melakukan augmentasi data berupa rotation, zoom, flipping, dan pergeseran gambar.

Tujuannya adalah memperbesar variasi data pelatihan agar model mampu mengenali lebih banyak bentuk dan posisi rempah-rempah. Data pelatihan diambil dari folder dataset/train, sedangkan data validasi berasal dari dataset/valid.

```
[11] train_datagen = ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    rotation_range=30,          # Lebih tinggi agar lebih bervariasi
    zoom_range=0.3,           # Zoom in/out lebih besar
    width_shift_range=0.2,     # Geser horizontal
    height_shift_range=0.2,    # Geser vertikal
    horizontal_flip=True,      # Membalik gambar secara horizontal
    vertical_flip=True,        # Membalik gambar secara vertikal
    brightness_range=[0.7, 1.3], # Variasi kecerahan
    fill_mode='nearest'        # Mengisi piksel kosong setelah augmentasi
)
training_set = train_datagen.flow_from_directory('/content/dataset/train',
                                                target_size=(64, 64),
                                                batch_size=32,
                                                class_mode='categorical') # Changed to categorical

Found 1440 images belonging to 5 classes.
```

```
[12] test_datagen = ImageDataGenerator(rescale=1./255)
test_set = test_datagen.flow_from_directory('/content/dataset/val',
                                           target_size=(64, 64),
                                           batch_size=32,
                                           class_mode='binary')
```

```
Found 360 images belonging to 5 classes.
```

Gambar 4. Proses Augmentasi

Model CNN dibangun dengan menggunakan pendekatan Sequential, yang terdiri dari beberapa lapisan: dimulai dari Conv2D dan MaxPooling2D untuk mengekstraksi fitur visual dari gambar, kemudian Flatten untuk meratakan hasil konvolusi menjadi vektor satu dimensi, dilanjutkan dengan Dense sebagai fully connected layer, serta Dropout untuk mencegah overfitting. Penelitian ini memakai metode Categorical Cross Entropy untuk melatih keluaran probabilitas dari CNN terhadap gambar rempah. Pada bagian akhir arsitektur CNN, digunakan fungsi aktivasi Softmax karena fungsi ini paling sesuai dan umum digunakan untuk tugas klasifikasi dengan banyak label[8].

```
[13] base_model = ResNet50(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

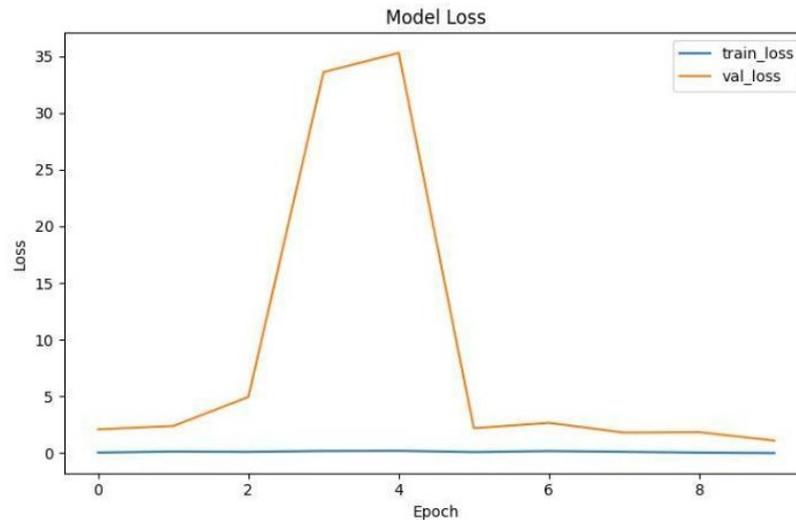
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/resnet50/resnet50\_weights\_tf\_dim\_ordering\_tf\_data\_format.h5
94765736/94765736 ————— 4s 0us/step
```

```
[14] classifier = tf.keras.Sequential([ #buat Sequential
    base_model,
    tf.keras.layers.GlobalAveragePooling2D(),
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dense(5, activation='softmax') # Changed to 5 (multi-class)
])
```

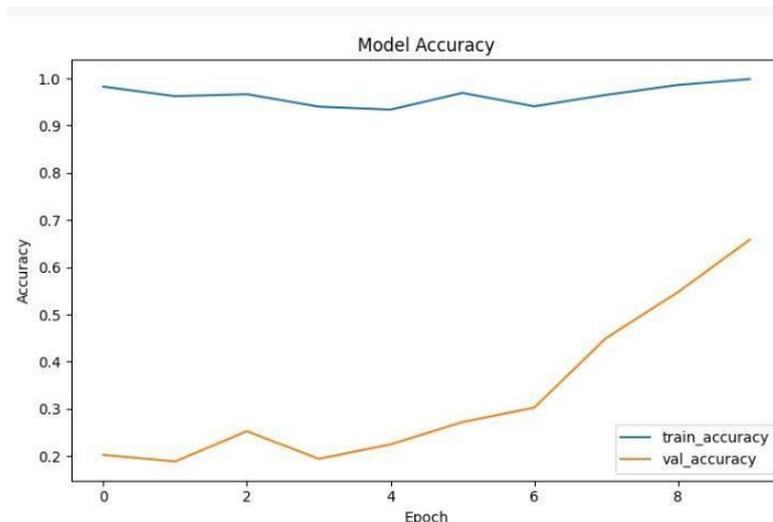
```
[15] classifier.compile( #mengompilasi model (mengevaluasi akurasi model)
    loss='categorical_crossentropy', #multi-class
    optimizer='adam',
    metrics=['accuracy']
)
```

Gambar 5. Kompilasi model

Proses pelatihan dilakukan selama 10 epoch, di mana model menunjukkan performa yang cukup baik berdasarkan hasil evaluasi grafik dan prediksi gambar. Grafik akurasi menunjukkan bahwa akurasi data latih cenderung tinggi dan stabil pada kisaran 95–99%, sementara akurasi validasi meningkat secara bertahap hingga mencapai sekitar 65% di akhir pelatihan. Meskipun sempat terjadi lonjakan drastis pada grafik loss validasi di awal epoch, hal tersebut berhasil dikoreksi pada epoch selanjutnya, menandakan adanya perbaikan dalam proses pembelajaran model.



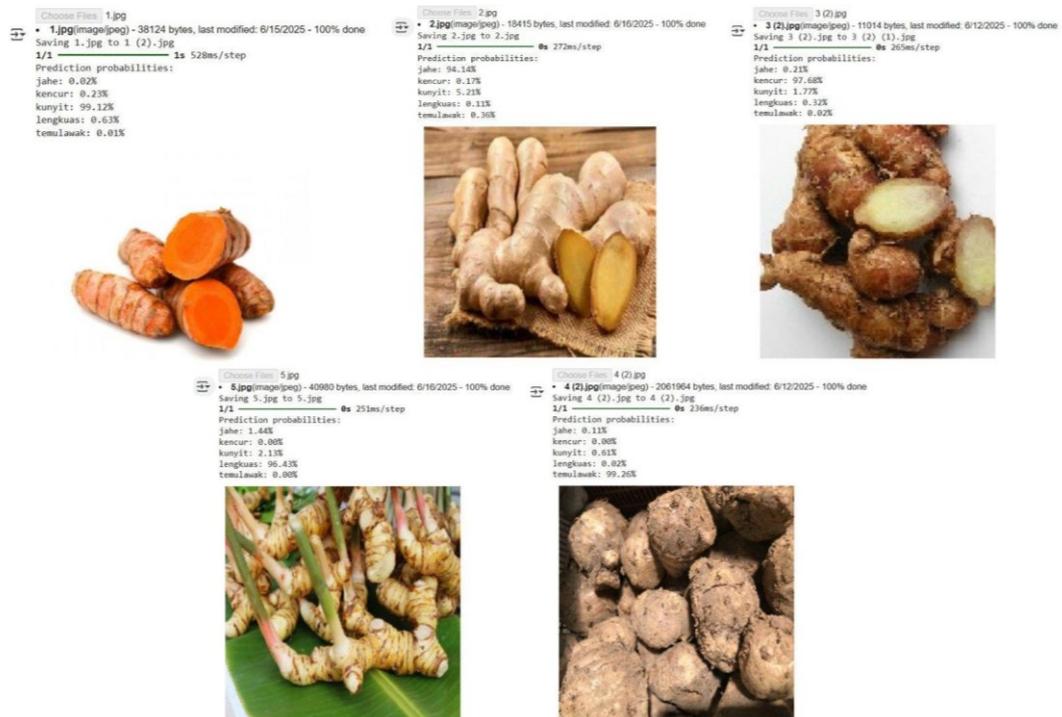
Gambar 6. Grafik model train loss dan val loss



Gambar 7. Grafik model train accuracy dan val accuracy

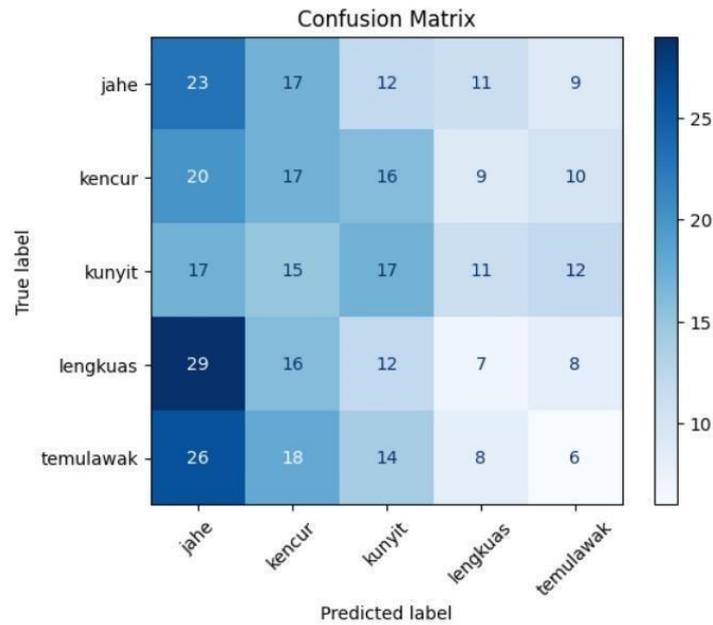
Selanjutnya, hasil prediksi model terhadap gambar uji menunjukkan tingkat akurasi yang sangat tinggi, dengan semua gambar berhasil dikenali secara tepat sesuai label aslinya dengan tingkat kepercayaan di atas 90%. Misalnya, gambar kunyit diprediksi dengan

probabilitas 99,12%, temulawak 99,26%, lengkuas 96,43%, kencur 97,68% dan jahe 94,14%. Keberhasilan ini menandakan bahwa model mampu mengenali ciri khas visual dari masing-masing jenis rempah dengan baik. Namun, perlu diwaspadai lonjakan loss yang terjadi pada awal pelatihan sebagai indikasi overfitting sementara, yang dapat diminimalkan dengan teknik seperti early stopping atau pengaturan ulang learning rate. Secara keseluruhan, model CNN ini telah menunjukkan kinerja yang memuaskan dalam tugas klasifikasi gambar rempah-rempah dan dapat digunakan sebagai dasar dalam sistem pengenalan otomatis berbasis citra.



Gambar 8. Hasil prediksi model CNN

Selain itu, digunakan juga confusion matrix, yaitu tabel yang menunjukkan jumlah prediksi benar dan salah untuk setiap kelas, sehingga dapat diketahui pola kesalahan yang sering dilakukan model.



Gambar 9. Confusion matrix

Dari matriks tersebut, terlihat bahwa model memiliki kesulitan dalam membedakan antara beberapa kelas, khususnya antara jahe, lengkuas, dan temulawak. Misalnya, dari 50 sampel temulawak, sebanyak 26 sampel justru diklasifikasikan sebagai jahe, yang menunjukkan tingkat kesalahan klasifikasi yang cukup tinggi pada kelas tersebut. Hal serupa terjadi pada lengkuas, di mana 29 sampel juga salah diklasifikasikan sebagai jahe. Hanya 23 dari 50 sampel jahe yang berhasil diklasifikasikan dengan benar, sementara sisanya tersebar ke kelas lain, seperti kencur (17), kunyit (11), dan temulawak (9). Hal ini menunjukkan bahwa fitur visual antara jahe, lengkuas, dan temulawak mungkin memiliki kemiripan yang tinggi, sehingga membingungkan model.

Tabel 3. Tabel Klasifikasi Rempah-rempah

Kelas	Precision	Recall	F1-Score	Support
Jahe	0.83	0.46	0.59	50
Kencur	0.88	0.90	0.89	50
Kunyit	0.84	0.84	0.84	50
Lengkuas	0.71	0.42	0.53	50
Temulawak	0.54	0.52	0.53	50
Rata-rata	0.76	0.63	0.68	250

4. KESIMPULAN

Model CNN yang dikembangkan berhasil mengklasifikasikan lima jenis rempah yaitu: jahe, kunyit, kencur, lengkuas, dan temulawak dengan tingkat akurasi yang tinggi, terbukti dari hasil pengujian yang menunjukkan prediksi dominan di atas 94% untuk setiap gambar yang sesuai. Hal ini menunjukkan bahwa model mampu mengenali karakteristik visual rempah secara efektif, meskipun terdapat variasi bentuk dan pencahayaan pada gambar. Dengan demikian, sistem ini memiliki potensi besar untuk digunakan dalam identifikasi otomatis rempah-rempah, khususnya dalam aplikasi pertanian, perdagangan, atau edukasi berbasis teknologi.

DAFTAR PUSTAKA

- [1] A. Eka et al., “Klasifikasi Jenis Rempah Menggunakan Convolutional Neural Network dan Transfer Learning,” 2023.
- [2] C. Nisa and F. Candra, “Klasifikasi Jenis Rempah-Rempah Menggunakan Algoritma Convolutional Neural Network,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 1, pp. 78–84, Dec. 2023, doi: [10.57152/malcom.v4i1.1018](#).
- [3] R. Abdulhakim, Carudin, and B. Arif Dermawan, “Analisis dan Penerapan Algoritma Convolutional Neural Network untuk Klasifikasi Kendaraan Prioritas,” *Jurnal Sains dan Informatika*, vol. 7, no. 2, pp. 135–144, Dec. 2021, doi: [10.34128/jsi.v7i2.335](#).
- [4] M. Sanjaya and E. Nurraharjo, “Deteksi Jenis Rempah-Rempah Menggunakan Metode Convolutional Neural Network Secara Real Time,” 2023.
- [5] A. M. S. Darmatasia, “IMPLEMENTASI CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN RIMPANG SECARA VIRTUAL DARMATASIA,” vol. 8, no. 1, 2023.
- [6] A. A. Permana, R. Muttaqin, and A. Sunandar, “SISTEM DETEKSI API SECARA REAL TIME MENGGUNAKAN ALGORITMA YOU ONLY LOOK ONCE (YOLO) VERSI 8,” 2024. Accessed: Jun. 18, 2025. [Online]. Available: <https://universe.roboflow.com/>.
- [7] H. A. Pratiwi, M. Cahyanti, and M. Lamsani, “IMPLEMENTASI DEEP LEARNING FLOWER SCANNER MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK,” *Sebatik*, vol. 25, no. 1, pp. 124–130, Jun. 2021, doi: [10.46984/sebatik.v25i1.1297](#).

- [8] M. Farid Naufal and S. Ferdiana Kusuma, “PENDETEKSI CITRA MASKER WAJAH MENGGUNAKAN CNN DAN TRANSFER LEARNING,” vol. 8, no. 6, pp. 1293–1300, 2021, doi: [10.25126/jtiik.202185201](#).
- [9] G. Hinton, L. Deng, D. Yu, et al., “Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012. doi: [10.1109/MSP.2012.2205597](#)
- [10] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv preprint [arXiv:1409.1556](#), 2014.
- [11] A. Shorten and T.M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019. doi: [10.1186/s40537-019-0197-0](#)
- [12] M. Mohanty, A. Hughes, and A. Salathé, “Using Deep Learning for Image-Based Plant Disease Detection,” *Frontiers in Plant Science*, vol. 7, pp. 1419, 2016. doi: [10.3389/fpls.2016.01419](#)
- [13] H. Q. Minh et al., “Application of Convolutional Neural Networks in Identification of Medicinal Plant Species,” *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, 2019. doi: [10.14569/IJACSA.2019.0100665](#)
- [14] S. Kamilaris and F. Prenafeta-Boldú, “Deep Learning in Agriculture: A Survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018. doi: [10.1016/j.compag.2018.02.016](#)
- [15] H. Wang et al., “Image-Based Plant Disease Detection with Deep Learning: A Review,” *Plant Methods*, vol. 15, no. 1, 2019. doi: [10.1186/s13007-019-0479-9](#)