

IMPLEMENTASI METODE CONVOLUTIONAL NEURAL NETWORK (CNN) DALAM KLASIFIKASI TANAMAN HERBAL BERBENTUK RIMPANG PADA SISTEM INFORMASI BERBASIS WEBSITE

Wulan Adira Nasution, Debi Yandra Niska, Said Iskandar Al Idrus, Zulfahmi Indra, Insan Taufik

Ilmu Komputer, Universitas Negeri Medan
Jalan Willem Iskandar, Pasar V Medan Estate, Percut Sei Tuan,
Deli Serdang, Sumatera Utara, Indonesia
wandhira18@gmail.com

ABSTRAK

Tanaman rimpang merupakan salah satu jenis tanaman herbal yang banyak dimanfaatkan dalam bidang kesehatan dan pengobatan tradisional. Tanaman rimpang seperti jahe, kunyit, lengkuas, kencur, dan temulawak merupakan tanaman herbal yang banyak dimanfaatkan dalam pengobatan tradisional. Namun, kemiripan bentuk dan warna antar jenis rimpang sering kali menyulitkan proses identifikasi secara kasat mata. Kesalahan identifikasi tidak hanya menurunkan efektivitas pengobatan, tetapi juga dapat memicu risiko kesehatan serius, seperti keracunan, gangguan pencernaan, hingga reaksi alergi pada individu yang sensitif terhadap senyawa *bioaktif* tertentu. Penelitian ini bertujuan membangun sistem klasifikasi tanaman rimpang secara otomatis menggunakan metode *Convolutional Neural Network* (CNN) yang diimplementasikan dalam sistem berbasis *website*. Selain melakukan klasifikasi, sistem ini juga menyajikan informasi mengenai karakteristik masing-masing tanaman rimpang guna meningkatkan pemahaman dan kewaspadaan pengguna dalam pemanfaatannya. Model dilatih menggunakan 50 *epoch* dengan skema pembagian data 80% untuk training, 10% validasi, dan 10% testing. Pengujian dilakukan sebanyak tiga kali dengan variasi nilai *learning rate* yang berbeda-beda. Hasil terbaik diperoleh pada pengujian kedua dengan *learning rate* 0.001 dengan akurasi sebesar 92,73%.

Kata kunci : Klasifikasi, Rimpang, *Convolutional Neural Network*

1. PENDAHULUAN

Tumbuhan merupakan salah satu bentuk organisme hidup yang berperan penting dalam kehidupan manusia. Seiring dengan berjalannya waktu, tumbuhan telah dimanfaatkan untuk berbagai keperluan, terutama dalam bidang kesehatan dan pengobatan. Tumbuhan yang memiliki khasiat penyembuhan dikenal sebagai tumbuhan herbal.

Tanaman herbal sering disebut sebagai apotek hidup. Hal ini merujuk pada pemanfaatan sebagian tanah untuk ditanami tanaman obat yang dapat digunakan untuk kepentingan sehari-hari, seperti untuk pengobatan penyakit ringan atau sebagai bahan baku masakan [1]. Di Indonesia, pengobatan tradisional menggunakan tanaman herbal telah menjadi bagian penting dalam mengatasi berbagai masalah kesehatan dan digunakan secara luas oleh masyarakat.

Seiring dengan manfaat yang ditawarkan, terdapat permasalahan yang sering terjadi saat pengonsumsi tanaman herbal. Permasalahan tersebut adalah kesulitan dalam mengidentifikasi jenis tanaman herbal karena memiliki kemiripan fisik yang mirip antar *spesies*. Salah satu jenis tanaman herbal yang sulit diidentifikasi adalah tanaman herbal berbentuk rimpang.

Tanaman rimpang mempunyai karakteristik dan ciri-ciri yang hampir sama dalam tekstur, bentuk, dan warna [2]. Hal tersebut membuat banyak orang mengalami kesulitan dalam membedakan tanaman rimpang karena bentuknya yang mirip [3]. Kesulitan dalam mengidentifikasi tanaman rimpang yang tepat

dapat berisiko menimbulkan dampak negatif bagi kesehatan seperti keracunan, gangguan pencernaan, atau reaksi alergi.

Penelitian tentang kesulitan identifikasi pernah dilakukan dengan mengumpulkan sebanyak 56 responden untuk menguji kemampuan membedakan rimpang. Hasilnya, 12 orang menjawab benar, 16 orang ragu-ragu, dan 28 orang salah [3]. Kesalahan dalam identifikasi ini semakin berisiko karena beberapa tanaman juga diketahui dapat memicu reaksi alergi pada individu yang sensitif. Sebagai contoh, paparan bubuk jahe yang dihirup dapat memicu reaksi alergi yang dimediasi oleh *IgE* [4].

Untuk menyelesaikan permasalahan diatas, dibutuhkan sebuah sistem yang dapat membantu dalam mengklasifikasikan tanaman herbal dengan mudah dan akurat dengan menggunakan metode pengolahan citra digital yang paling menjanjikan dalam pengolahan citra yaitu metode *Convolutional Neural Network* (CNN). *Convolutional Neural Network* (CNN) termasuk kedalam arsitektur deep learning yang paling umum digunakan untuk tugas-tugas pengolahan citra. [5]. CNN biasanya bekerja dengan cara mempelajari fitur-fitur penting dari gambar dan menggeneralisasi fitur-fitur tersebut untuk menyelesaikan berbagai masalah, seperti pengenalan objek [6].

Dengan menggunakan metode CNN, sistem ini diharapkan mampu membedakan tanaman rimpang berdasarkan bentuk, warna, dan tekstur. Selain itu, diharapkan mampu memudahkan akses informasi bagi pengguna agar dapat mencegah terjadinya

kesalahan konsumsi yang dapat menimbulkan risiko kesehatan serius, seperti keracunan, iritasi, maupun reaksi alergi. Dengan demikian, sistem ini tidak hanya meningkatkan efektivitas pemanfaatan tanaman herbal, tetapi juga mendukung keamanan dalam penggunaan obat tradisional.

2. TINJAUAN PUSTAKA

2.1. Tanaman Herbal

Tanaman herbal adalah tumbuhan obat yang memiliki manfaat tambahan dalam pengobatan tradisional terhadap berbagai penyakit. Tanaman herbal ini mengandung metabolit sekunder yang merupakan senyawa dengan efek obat yang khas [7]. Senyawa-senyawa tersebut dapat bermanfaat untuk pencegahan, penyembuhan penyakit, serta menjalankan fungsi biologis tertentu [1].

2.2. Tanaman Rimpang

Rimpang atau yang dalam bahasa Latin disebut *rhizome*, merupakan bagian modifikasi yang berasal dari batang tumbuhan yang berkembang di bawah tanah serta dapat membentuk tunas dan akar baru dari setiap ruasnya [8]. Rimpang umumnya digunakan sebagai bahan dalam berbagai bidang sektor industri seperti makanan/minuman, kosmetik, dan sebagai bahan farmasi [9]. Berikut adalah contoh tanaman herbal berbentuk rimpang:



Gambar 1. Jenis - Jenis Tanaman Rimpang

2.3. Machine Learning

Machine Learning adalah cabang ilmu yang mengkaji cara pembuatan model kecerdasan buatan yang dapat belajar dari data. Model-model ini umumnya disebut sebagai model *machine learning* yang mampu menyimpan dan mengubah pengetahuan mereka melalui proses pembelajaran dari data. *Machine learning* biasanya digunakan untuk menganalisis data dalam bentuk tabel, teks, suara, dan lainnya. Selain itu, *machine learning* menyimpan pengetahuan mereka dalam bentuk parameter, yang berupa aturan atau nilai numerik (*integer* atau *float*)[10].

2.4. Deep Learning

Deep Learning dirancang menyerupai sistem pada otak manusia, di mana *neuron-neuron* saling

terhubung dan akan *Deep Learning* merupakan teknik pembelajaran dengan menggunakan jaringan saraf tiruan yang terdiri dari beberapa lapisan. *Deep Learning* dirancang menyerupai sistem pada otak manusia, di mana *neuron-neuron* saling terhubung dan akan membentuk jaringan *neuron* yang cukup kompleks [11]. Sistem tersebut mampu belajar serta beradaptasi dengan jumlah data beskala besar, serta mampu menyelesaikan permasalahan kompleks yang sulit dikerjakan menggunakan algoritma pembelajaran mesin lainnya [12].

2.5. Pengolahan Citra Digital

Pengolahan citra merupakan teknik yang mengubah gambar atau citra menjadi format digital dan menerapkan berbagai operasi untuk memperbaiki kualitas citra atau memperoleh informasi penting darinya. Pada dasarnya, pengolahan citra adalah bentuk pemrosesan sinyal yang menggunakan gambar sebagai input dan menghasilkan output berupa gambar yang telah diproses atau fitur tertentu dari gambar tersebut [13]. Citra digital merepresentasikan visual yang disimpan kedalam bentuk digital dan tersusun atas piksel-piksel yang memiliki nilai numerik tertentu [14].

2.6. Convolutional Neural Network (CNN)

Algoritma Convolutional Neural Network (CNN) adalah salah satu algoritma yang digunakan untuk melakukan tugas-tugas pengolahan pada data seperti pengolahan gambar ataupun suara. *Algoritma CNN* biasanya digunakan dalam proses pengklasifikasian data dengan pendekatan *Supervised Learning* yang didalamnya terdiri dari data latih dan data uji untuk mengklasifikasikan data ke kategori yang sesuai [5].

Arsitektur CNN yang akan digunakan ada penelitian ini *arsitektur* yaitu *Xception*. *Xception* (*Xtreme of Inception*) merupakan arsitektur CNN yang mengadopsi teknik *depthwise separable convolution*[15]. Model ini diinisialisasi menggunakan bobot pralatih menggunakan dataset dari *ImageNet* [16].

2.7. Confusion Matrix

Confusion matrix merupakan suatu metode evaluasi kinerja yang digunakan untuk menilai efektivitas suatu sistem klasifikasi.

		Actual Value	
		(Positive) 1	(Negative) 0
Predicted Value	(Positive) 1	TP (True Positive)	FP (False Positive)
	(Negative) 0	FN (False Negative)	TN (True Negative)

Gambar 2. Confusion matrix

Confusion matrix terdiri dari beberapa *matrix* yang digunakan untuk mengevaluasi model seperti, akurasi, *precision*, *recall* dan *f1-score*. *Accuracy* digunakan untuk mengukur seberapa banyak prediksi yang benar, baik positif maupun negatif.

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

Precision (Presisi) berguna untuk mengukur ketepatan prediksi positif yang dibuat oleh model.

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

Recall digunakan untuk mengukur seberapa baik model menemukan semua kasus positif yang sebenarnya.

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

F1-Score: F1 score merupakan hasil gabungan antara *precision* dan *recall*.

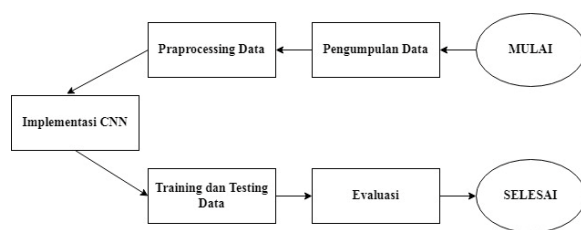
$$F1 - \text{Score} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (4)$$

2.8. Website

Website adalah sekumpulan halaman digital yang memuat beragam informasi seperti gambar, teks, video, suara, atau gabungan dari berbagai elemen tersebut. *Website* terhubung melalui internet sehingga dapat diakses oleh siapa pun yang terhubung ke jaringan internet [17]. Dalam pengembangan *website*, diperlukan beberapa bahasa pemrograman untuk membangun satu halaman *website*. Bahasa pemrograman tersebut meliputi HTML, CSS, dan JavaScript.

3. METODE PENELITIAN

Tahapan penelitian yang dilakukan dapat dilihat dibawah ini:



Gambar 3. Tahapan penelitian

3.1. Pengumpulan Data

Tahapan pertama adalah mengumpulkan gambar rimpang tanaman herbal dari penjual di pasar menggunakan kamera Samsung A23 dengan format .jpg, dengan memperhatikan pencahayaan, sudut, dan latar belakang yang bersih untuk memastikan konsistensi dataset.

3.2. Preprocessing Data

Proses ini meliputi pemberian label pada 10 kategori, penghilangan background, dan *resize* citra

menjadi 224x224 piksel. Augmentasi data dilakukan dengan *flipping*, *zooming*, dan variasi lainnya. Data kemudian dibagi menjadi 80% *training*, 10% *validation* dan 10% *testing*.

3.3. Implementasi CNN

Pada tahap ini, arsitektur CNN dibangun menggunakan *framework* seperti *TensorFlow*. Dimulai dengan layer konvolusi untuk mengekstraksi fitur, diikuti dengan layer aktivasi ReLU untuk menambah *non-linearitas*. Layer *pooling* (seperti *max pooling*) digunakan untuk mereduksi dimensi fitur, kemudian dilanjutkan dengan layer *fully connected* untuk integrasi fitur. Terakhir, layer *output* dengan fungsi aktivasi *softmax* atau *sigmoid* digunakan untuk menghasilkan *probabilitas* klasifikasi.

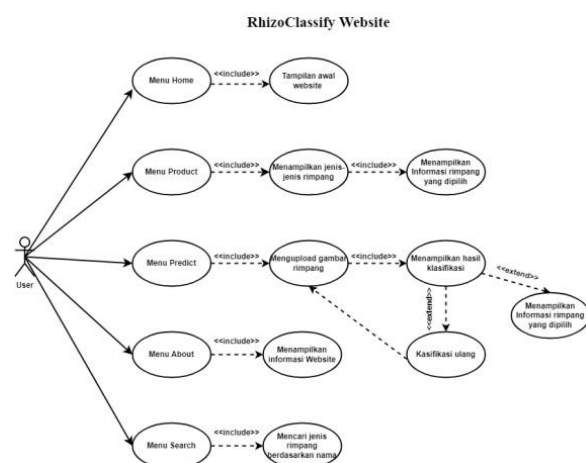
3.4. Training dan Testing Data

Langkah selanjutnya adalah melatih model menggunakan data *training* yang telah diproses sebelumnya. Proses *training* bertujuan untuk menyesuaikan bobot dan bias model agar dapat mengenali pola-pola yang ada dalam dataset. Selama proses *training*, parameter model disesuaikan berdasarkan *feedback* dari fungsi *loss* yang dihitung dari perbandingan hasil prediksi dengan label yang sebenarnya. Setelah *training* selesai, model diuji menggunakan data *testing* yang tidak pernah dilihat sebelumnya untuk mengevaluasi kinerja akhirnya.

3.5. Evaluasi Model

Langkah terakhir adalah mengevaluasi kinerja model CNN yang telah dilatih. Evaluasi dapat dilakukan menggunakan metrik seperti akurasi, *presisi*, *recall*, atau *F1-score*. Hal ini dilakukan untuk menganalisis hasil evaluasi dan memahami seberapa baik model CNN dalam mengklasifikasikan gambar.

3.6. Perancangan Sistem



Gambar 4. Use Case Diagram

Diagram *Use Case RhizoClassify Website* menggambarkan interaksi pengguna dengan fitur utama sistem, seperti *Home*, *Product*, *Predict*, *About*

serta *Search*. Pengguna dapat melakukan klasifikasi atau melihat detail lebih lanjut tentang rimpang yang diklasifikasi.

4. HASIL DAN PEMBAHASAN

4.1. Pengumpulan Data

Data tanaman herbal berbentuk rimpang dikumpulkan secara langsung menggunakan kamera ponsel Samsung A23. Data yang dikumpulkan terdiri dari 10 label untuk tanaman rimpang. Jumlah keseluruhan citra yang diambil sebanyak 500 citra yang terdiri dari 50 citra pada setiap label.

4.2. *Praprocessing* Data

Pada tahapan ini dilakukan beberapa langkah untuk memproses citra. Tahapan tersebut terdiri dari:

a. Pelabelan Data

Pelabelan data dilakukan dengan memberikan label manual pada setiap gambar sesuai jenis tanaman rimpang untuk memastikan keakuratan dan mempersiapkan data pelatihan model CNN.

b. Penghapusan Gambar pada *Background*

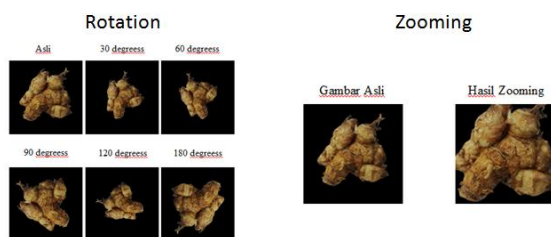
Pada tahap penghapusan *background*, latar belakang gambar tanaman herbal rimpang dihapus dan diubah menjadi hitam.

c. Mengubah Ukuran Gambar

Semua gambar diubah ukurannya menjadi 500x500 piksel untuk memastikan konsistensi input pada model CNN, memudahkan proses pelatihan, dan meningkatkan akurasi klasifikasi.

d. *Augmentasi*

Proses *augmentasi* berguna untuk meningkatkan variasi pada gambar. Proses tersebut meliputi pemutaran rotasi dan *zooming*.



Gambar 5. *Augmentasi*

e. Pengolahan Data Input

Pada tahap pengolahan data input, ketidakseimbangan data diatasi dengan *augmentasi* sehingga setiap label memiliki 100 gambar, menghasilkan total 1000 gambar. Ditambahkan juga label "random" untuk citra non-tanaman rimpang, menjadikan total data 1100 gambar.

4.3. Pembagian Data

Data akan dibagi kedalam 3 bagian yaitu data latih dan data uji/validasi. Sebanyak 80% data akan digunakan untuk melatih model, 10% untuk validasi dan 10% sisanya untuk proses pengujian.

4.4. Implementasi CNN

Berikut ini arsitektur model CNN yang akan dibangun:

Model: "sequential"

Layer (type)	Output Shape	Param #
xception (Functional)	(None, 7, 7, 2048)	20861480
conv2d_4 (Conv2D)	(None, 7, 7, 32)	589856
max_pooling2d (MaxPooling2D)	(None, 3, 3, 32)	0
dropout (Dropout)	(None, 3, 3, 32)	0
flatten (Flatten)	(None, 288)	0
dense (Dense)	(None, 11)	3179

 Total params: 21,454,515
 Trainable params: 593,035
 Non-trainable params: 20,861,480

Gambar 6. Arsitektur Model CNN

Pada tahap pembangunan arsitektur model, digunakan model *Xception* sebagai *feature extractor*. Model *Xception* ini dipilih karena sudah dilatih sebelumnya (*pretrained*) menggunakan dataset besar *ImageNet*, sehingga mampu mengenali berbagai pola fitur visual secara efektif. Input gambar disesuaikan berukuran 224x224 piksel dengan tiga saluran warna (RGB).

Setelah itu, ditambahkan *Layer Conv2D* yang digunakan untuk mengekstraksi fitur tambahan dengan 32 filter kernel 3x3. Kemudian dilanjutkan dengan dengan layer *MaxPool2D* yang digunakan untuk mengurangi dimensi data atau *feature map* sehingga mempercepat pelatihan. *Dropout layer* juga diterapkan untuk mencegah *overfitting* dengan cara menonaktifkan sebagian *neuron* saat pelatihan.

Selanjutnya, output dari *layer* sebelumnya diratakan menjadi *vector* 1 dimensi menggunakan *layer Flatten* agar dapat diproses oleh *layer fully connected (Dense)*. Kemudian, ditambahkan *layer Dense* dengan 11 *neuron* yang masing-masing merepresentasikan satu label. Fungsi aktivasi *softmax* ditambahkan untuk menghasilkan probabilitas tiap kelas, sehingga model dapat memilih kelas dengan nilai probabilitas tertinggi sebagai hasil prediksi.

Model dikompilasi menggunakan *optimizer Adam* agar mampu menyesuaikan laju pembelajaran selama proses pelatihan berlangsung. Fungsi *loss categorical_crossentropy* digunakan untuk klasifikasi multikelas dengan format label *one-hot encoding*, yang berguna untuk meminimalkan kesalahan dalam prediksi.

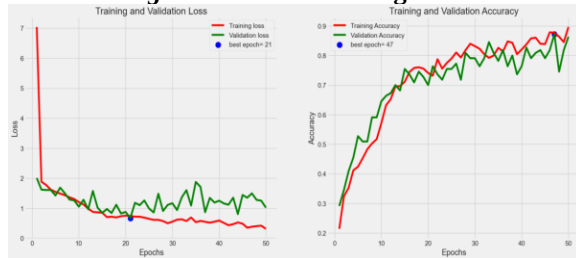
4.5. Hyperparameter Model

Model akan diuji menggunakan 50 *epoch* dengan 3 pengujian menggunakan *learning rate* berbeda yaitu, 0.01, 0.001, 0.0001 untuk mencari nilai akurasi terbaik pada proses pengujian. Berikut adalah hyperparameter yang digunakan:

Tabel 1. Hyperparameter

No	Hyperparameter	Nilai
1.	Learning Rate	0.01, 0.001, 0.0001
2.	Split data	80% Training, 10% Validation, 10% Testing
3.	Batch Size	64
4.	Layers	6
5.	Epoch	50
6.	Optimizer	Adam

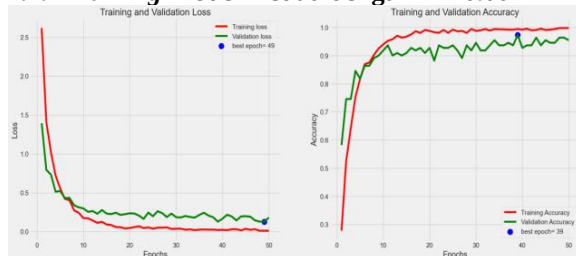
4.6. Training Model Pertama dengan LR 0.01



Gambar 7. Grafik Accuracy dan Loss Training Pertama

Grafik diatas menunjukkan bahwa nilai *loss* menurun seiring bertambahnya *epoch*. Hal ini menandakan bahwa model semakin belajar dengan baik. Sementara itu, grafik *Training and Validation Accuracy* menunjukkan peningkatan akurasi pada data pelatihan dan validasi. Hasil akurasi yang didapatkan pada proses *training* model pertama yaitu sebesar 86.36%.

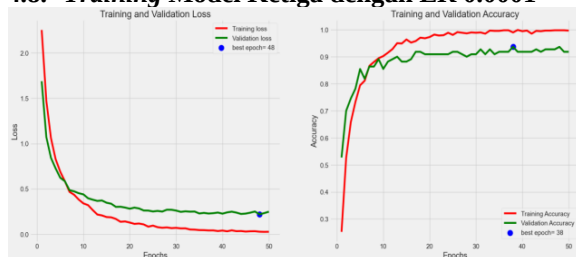
4.7. Training Model Kedua dengan LR 0.001



Gambar 8. Grafik Accuracy dan Loss Training Kedua

Gambar grafik diatas menunjukkan penurunan *loss* yang signifikan di awal pelatihan, menandakan penyesuaian bobot yang efektif. Akurasi juga meningkat, dengan performa terbaik pada *epoch* ke-39 dan memperoleh hasil akurasi yang diperoleh sebesar 92,73%.

4.8. Training Model Ketiga dengan LR 0.0001



Gambar 9. Grafik Accuracy dan Loss Training Ketiga

Grafik pada *training* ketiga juga menunjukkan peningkatan akurasi secara bertahap selama pelatihan, yang mencerminkan kemampuan model dalam mempelajari pola data dengan baik. Meskipun akurasi akhir pada data uji sebesar 90,00% sedikit lebih rendah dibanding grafik sebelumnya, hasil ini tetap menunjukkan kinerja yang baik. Performa terbaik tercapai pada *epoch* ke-38.

Berdasarkan ketiga pengujian yang sudah dilakukan, maka diperoleh hasil sebagai berikut:

Tabel 2. Hasil Akurasi

Pengujian	Learning Rate	Hasil Akurasi
1.	0.01	86.36%
2.	0.001	92.73%
3.	0.0001	90.00%

Dari Tabel 3. dapat disimpulkan bahwa nilai *learning rate* 0.001 memberikan hasil akurasi terbaik, yaitu sebesar 92,73%. Hal ini menunjukkan bahwa pemilihan *learning rate* yang tepat sangat berpengaruh terhadap performa model, di mana *learning rate* yang terlalu besar atau kecil dapat menurunkan akurasi prediksi.

4.9. Evaluasi Model

Tahap selanjutnya adalah melakukan evaluasi model dengan *confusion matrix* untuk menilai kinerja prediksi. Pengujian dilakukan menggunakan 110 data citra. *Confusion matrix* berikut merupakan hasil dari *training* kedua yang merupakan *training* dengan tingkat akurasi tertinggi.

		Confusion Matrix										
Actual	giring	7	0	0	0	0	0	0	0	0	0	0
	hitam	0	10	0	0	0	0	0	0	0	0	0
	jahe	0	0	13	1	0	0	0	1	0	0	0
	kencur	0	0	0	12	0	0	1	0	0	0	0
	kunci	0	0	0	0	10	0	0	0	0	0	0
	kunyit	0	1	0	0	0	7	0	0	0	0	0
	kunyitp	0	0	0	1	0	0	4	0	0	0	0
	lengkuas	0	0	0	0	0	0	0	13	0	0	0
	mangga	0	0	0	0	0	0	0	0	9	0	0
	random	0	0	0	0	0	0	1	0	0	8	0
	temulawak	0	0	0	1	0	1	0	0	0	0	9
		Predicted										
		giring	hitam	jahe	kencur	kunci	kunyit	kunyitp	lengkuas	mangga	random	temulawak

Gambar 10. Evaluasi Model

Model yang dibangun berhasil memprediksi gambar yang benar sebanyak 102 citra dan gambar yang salah sebanyak 8 citra.

$$\begin{aligned}
 \text{akurasi} &= \frac{7 + 10 + 13 + 12 + 10 + 7 + 4 + 13 + 9 + 8 + 9}{110} \times 100\% \\
 \text{akurasi} &= \frac{102}{110} \times 100\% = 92.73\%
 \end{aligned}$$

Selanjutnya untuk hasil dari nilai *precision*, *recall*, dan *f1-score* untuk setiap kelas dapat dilihat dari tabel *classification report* dibawah ini:

Tabel 4. *Classification Report*

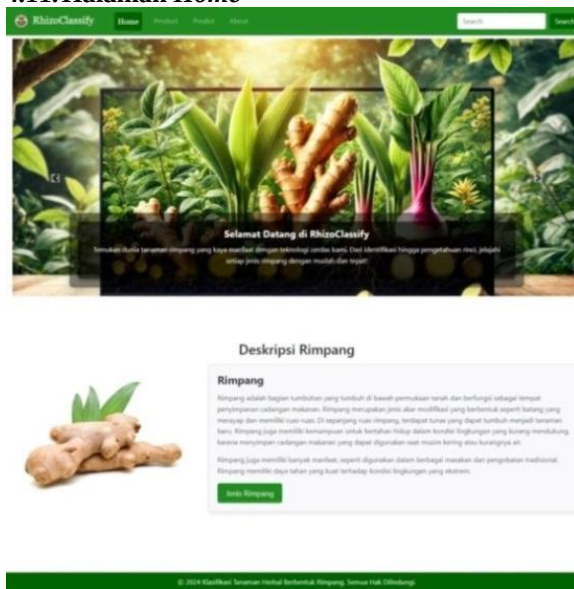
Label	Precision	Recall	F1-score
Giring	1.00	1.00	1.00
Hitam	0.91	1.00	0.95
Jahe	1.00	0.87	0.93
Kencur	0.80	0.92	0.86
Kunci	1.00	1.00	1.00
Kunyit	0.88	0.88	0.88
kunyitp	0.67	0.80	0.73
Lengkuas	0.93	1.00	0.96
Mangga	1.00	1.00	1.00
Random	1.00	0.89	0.94
temulawak	1.00	0.82	0.90
Rata-Rata	0.93	0.93	0.92
Accuracy	92.73%		

Berdasarkan Tabel 3, hasil pengujian model klasifikasi menunjukkan nilai rata-rata *precision* dan *recall* sebesar 0,93, serta *F1-score* sebesar 0,92. Selain itu, akurasi keseluruhan yang diperoleh sebesar 92,73%, yang menunjukkan bahwa model sudah cukup baik dalam mengenali dan membedakan berbagai jenis tanaman rimpang.

4.10. Integrasi Model CNN kedalam Website

Setelah model CNN dilatih, model disimpan dalam format .h5 dan diintegrasikan ke dalam *website* menggunakan *Flask* sebagai *backend*. Pengguna dapat mengunggah gambar rimpang dan melihat hasil prediksi langsung melalui antarmuka *web*.

4.11. Halaman Home

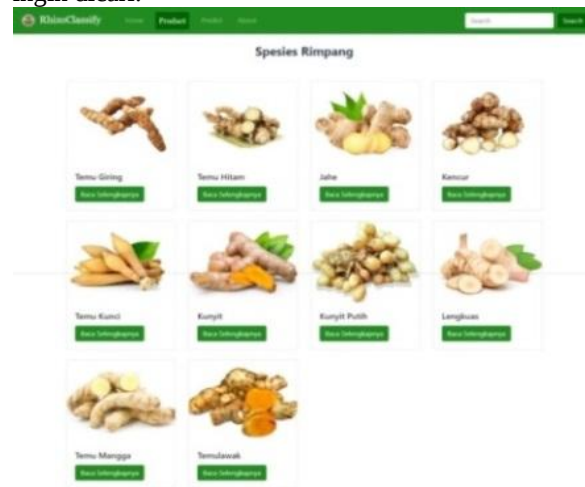


Gambar 11. Tampilan Menu Home

Menu *Home* adalah halaman utama yang muncul saat membuka *website*. Menu ini berisi tombol navigasi dan deskripsi singkat tanaman rimpang.

4.12. Menu Product

Pada menu *product* terdapat daftar nama tanaman rimpang yang ada pada *website*. Disetiap daftar juga terdapat tombol “Baca selengkapnya” yang berisi informasi setiap tanaman rimpang yang ingin dicari.



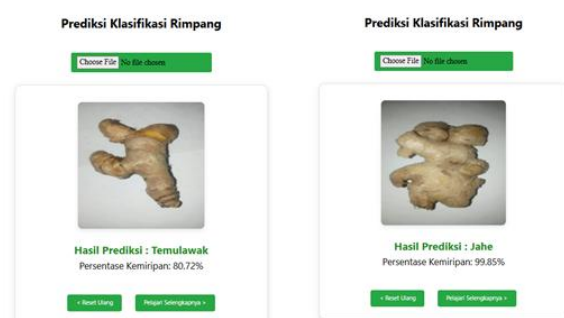
Gambar 12. Tampilan Menu Product

4.13. Menu Predict



Gambar 13. Tampilan Menu Predict

Menu ini berguna untuk memprediksi jenis tanaman rimpang dengan cara mengunggah gambar ke sistem. Sistem dirancang untuk menampilkan hasil prediksi berdasarkan persentase kemiripan. Jika persentase kemiripan di atas 60%, sistem akan menampilkan satu label rimpang. Namun, jika persentase kemiripan di bawah 60%, sistem akan menampilkan dua label rimpang dengan nilai kemiripannya.

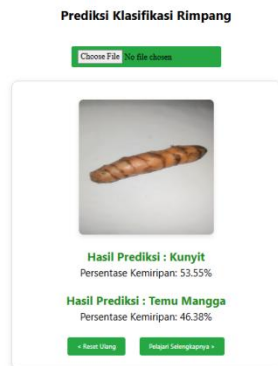


Uji Coba 1

Uji Coba 2

Gambar 14. Tampilan Hasil Predict diatas 60%

Gambar 14. menunjukkan dua contoh hasil prediksi yang dilakukan melalui dua kali percobaan berbeda, masing-masing dengan persentase kemiripan lebih dari 60%. Pada setiap percobaan, sistem hanya menampilkan satu label rimpang sesuai hasil analisis.



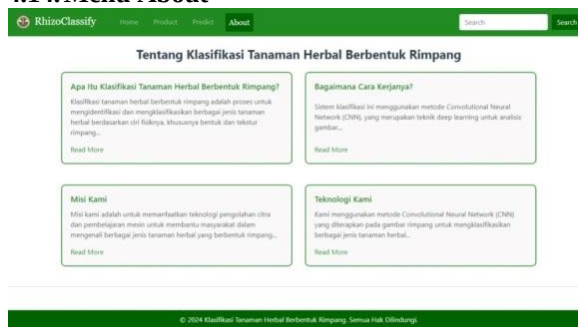
Gambar 15. Tampilan Hasil *Predict* dibawah 60%

Gambar 15. memperlihatkan hasil prediksi dengan persentase kemiripan di bawah 60%, sehingga sistem menampilkan dua label rimpang yang paling mendekati beserta nilai kemiripannya. Tujuannya agar pengguna tetap mendapat alternatif jenis rimpang terdekat. Setelah prediksi muncul, akan tersedia tombol “Pelajari Selengkapnya” untuk melihat informasi detail mengenai rimpang tersebut.



Gambar 16. Tampilan Menu Informasi *Product*

4.14. Menu About



Gambar 17. Tampilan Menu *About*

Menu ini berisi informasi mengenai *website*. Menu ini menjelaskan tentang aplikasi yang dibuat.

5. KESIMPULAN DAN SARAN

Algoritma Convolutional Neural Network (CNN) dengan menggunakan arsitektur *Xception* berhasil dibangun dan menunjukkan *performa* yang baik dalam mengidentifikasi jenis-jenis tanaman rimpang. Data yang digunakan sebanyak 100 sampel setiap label/kelas. Model ini dilatih menggunakan 50 epoch dengan skema pembagian data 80% untuk *training*, 10% untuk *validation* dan 10% untuk *testing*. Pengujian dilakukan 3 kali dengan *learning rate* yang berbeda-beda. Hasil terbaik diperoleh pada pengujian kedua dengan learning rate 0.001. Evaluasi pada data uji menghasilkan akurasi sebesar 92.73%, yang membuktikan bahwa *algoritma* CNN yang dibangun berhasil mengklasifikasikan tanaman rimpang dengan akurasi yang cukup tinggi. Adapun saran yang dapat diberikan untuk pengembangan lebih lanjut yaitu memperbanyak jumlah sampel dengan variasi kondisi pencahayaan, sudut pandang, dan latar belakang. Selain itu, pengujian berbagai konfigurasi hyperparameter seperti jumlah epoch, learning rate, dan optimizer juga perlu dilakukan untuk meningkatkan *performa* model.

DAFTAR PUSTAKA

- [1] Kumontoy, G. D., D. Deeng., and T. Mulianti, “Vol. 16 No. 3 / Juli - September 2023,” vol. 16, no. 3, pp. 1–20, 2023.
- [2] Asep Zainal Alfari and Enny Itje Sela, “Klasifikasi Rimpang Menggunakan Metode K-Nearest Neighbor dan Ekstraksi Ciri Gray Level Co-occurrence Matrix,” *J. Fasilkom*, vol. 14, no. 1, pp. 88–94, 2024, doi: 10.37859/jf.v14i1.6832.
- [3] Y. Feberian and D. Fitriati, “Klasifikasi Rimpang Menggunakan Convolution Neural Network,” *J. Informatics Adv. Comput.*, vol. 3, no. 1, pp. 10–14, 2022.
- [4] M. N. Ahnaf, Nasiroh, N. Aulia, N. L. M. Lestrari, M. Ngongo, and A. R. Hakim, “Jahe (*Zingiber Officinale*): Tinjauan Fitokimia, Farmakologi, dan Toksikologi,” *J. Ilmu Kedokt. dan Kesehat.*, vol. 11, pp. 1992–1998, 2024, doi: 10.36684/74-1-2022-98-101.
- [5] U. S. Rahmadhani and N. L. Marpaung, “Klasifikasi Jamur Berdasarkan Genus Dengan Menggunakan Metode CNN,” *J. Inform. J. Pengemb. IT*, vol. 8, no. 2, pp. 169–173, 2023, doi: 10.30591/jpit.v8i2.5229.
- [6] B. Raharjo, *Deep Learning dengan Python*. Semarang: Yayasan Prima Agus Teknik, 2022.
- [7] A. Sapitri, V. Asfianti, and E. D. Marbun, “Pengelolaan Tanaman Herbal Menjadi Simplicia sebagai Obat Tradisional,” *J. Abdimas Mutiara*, vol. 3, no. 1, pp. 94–102, 2022.
- [8] M. Mayasari, D. Iskandar Mulyana, M. Betty Yel, and S. Tinggi Ilmu Komputer Cipta Karya Informatika Jl Raden, “Komparasi Klasifikasi Jenis Tanaman Rimpang Menggunakan Principal Component Analisis, Support Vector Machine, K-Nearest Neighbor Dan Decision

- Tree,” *J. Tek. Inform. Kaputama*, vol. 6, no. 2, pp. 644–655, 2022.
- [9] Darmatasia and S. A. Muhammad, “Implementasi Convolutional Neural Network Untuk Klasifikasi Tanaman Rimpang Secara Virtual,” *J. INSTEK (Informatika Sains dan Teknol.*, vol. 8, no. 1, pp. 122–131, 2023.
- [10] G. N. Elwirehardja, T. Suparyanto, and B. Pardamean, *Machine Learning Untuk Pemula*. 2023.
- [11] A. Kholik, “Klasifikasi Menggunakan Convolutional Neural Network (Cnn) Pada Tangkapan Layar Halaman Instagram,” *J. Data Min. dan Sist. Inf.*, vol. 2, no. 2, p. 10, 2021, doi: 10.33365/jdmsi.v2i2.1345.
- [12] A. Raup, W. Ridwan, Y. Khoeriyah, S. Supiana, and Q. Y. Zaqiah, “Deep Learning dan Penerapannya dalam Pembelajaran,” *JlIP - J. Ilm. Ilmu Pendidik.*, vol. 5, no. 9, pp. 3258–3267, 2022, doi: 10.54371/jiip.v5i9.805.
- [13] Mangaras Yanu F., B. Yuwono, and Dessyanto Boedi P., *Dasar Pengolahan Citra Digital Edisi 2022*, 2nd ed., no. 1. Yogyakarta: UPN Veteran Yogyakarta, 2022.
- [14] R. Dijaya, *Buku Ajar Pengolahan Citra Digital*. Sidoarjo: UMSIDA PRESS, 2023.
- [15] A. E. Putra, M. F. Naufal, and V. R. Prasetyo, “Klasifikasi Jenis Rempah Menggunakan Convolutional Neural Network dan Transfer Learning,” *J. Edukasi dan Penelit. Inform.*, vol. 9, no. 1, p. 12, 2023, doi: 10.26418/jp.v9i1.58186.
- [16] D. P. Pamungkas and M. F. Amrulloh, “Analisis hasil klasifikasi penyakit daun bawang merah menggunakan cnn arsitektur exception,” vol. 10, no. 1, pp. 359–366, 2025.
- [17] O. Sari, Ani, A. Abdilah, and Sunarti, *Web Programming1*. 2022. doi: 10.1201/9781003316244-11.