

Klasifikasi Tanaman Herbal Berdasarkan Tekstur Daun Menggunakan Backpropagation Berbasis Citra

Meilan Ezperanza Florentiana Letik¹, Franki Yusuf Bisilisin^{2*}

¹Sekolah Tinggi Manajemen Informatika Kupang (STIKOM Uyelindo Kupang, email: meylanletik02@gmail.com

² Sekolah Tinggi Manajemen Informatika Kupang (STIKOM Uyelindo Kupang, email: aenk.funk@gmail.com

*corresponding author)

Info Artikel

Diajukan: 05 Juli 2024

Diterima: 08 Juli 2024

Diterbitkan: 30 Juli 2024

Kata Kunci

Backpropagation;
Klasifikasi;
Citra;
Gambar;
Tanaman Herbal.

Keywords:

Backpropagation;
Classification;
Citra;
Image;
Herbal Plant.



Lisensi: cc-by-sa

Copyright © 2024 by Author. Published by
Faatuatua Media Karya

Abstrak

Indonesia memiliki sekitar 30.000 dari total 40.000 jenis tanaman herbal dunia, menjadikannya negara dengan jumlah tanaman herbal terbanyak kedua setelah Brazil. Tanaman herbal telah lama digunakan dalam pengobatan tradisional karena manfaat alaminya. Namun, pemanfaatan tanaman herbal di Indonesia masih terbatas akibat kurangnya pemahaman masyarakat. Penelitian ini bertujuan meningkatkan pemanfaatan tanaman herbal dengan mengklasifikasikan jenis tanaman berdasarkan tekstur daun menggunakan backpropagation. Ekstraksi ciri tekstur dilakukan dengan metode GLCM menggunakan empat fitur utama: contrast, correlation, energy, dan homogeneity. Data penelitian terdiri dari 200 citra, masing-masing 10 sampel dari 20 jenis tanaman herbal yang digunakan. menggunakan jaringan saraf tiruan dengan algoritma backpropagation. Evaluasi kinerja dilakukan dengan confusion matrix untuk menguji akurasi prediksi. Dengan 160 data latih dan 40 data uji, serta konfigurasi jaringan dengan 10 hidden layer, pelatihan sebanyak 5000 epoch, dan learning rate 0,01, diperoleh akurasi tertinggi 85% dan rata-rata 80,5%. Hasil menunjukkan model backpropagation mencapai akurasi baik dalam klasifikasi jenis tanaman herbal.

Abstract

Indonesia has around 30,000 of the world's total 40,000 types of herbal plants, making it the country with the second largest number of herbal plants after Brazil. Herbal plants have long been used in traditional medicine because of their natural benefits. However, the use of herbal plants in Indonesia is still limited due to a lack of public understanding. This research aims to increase the use of herbal plants by classifying plant types based on leaf texture using backpropagation. Texture feature extraction is carried out using the GLCM method using four main features: contrast, correlation, energy and homogeneity. The research data consists of 200 images, each with 10 samples from 20 types of herbal plants used. using artificial neural networks with the backpropagation algorithm. Performance evaluation is carried out using a confusion matrix to test prediction accuracy. With 160 training data and 40 test data, as well as a network configuration with 10 hidden layers, 5000 epochs of training, and a learning rate of 0.01, the highest accuracy was 85% and an average of 80.5%. The results show that the backpropagation model achieves good accuracy in classifying herbal plant types.

1. PENDAHULUAN

Indonesia diakui sebagai negara kedua setelah Brazil dengan jumlah tanaman herbal terbanyak[1]. Di dunia terdapat sekitar 40.000 jenis tanaman herbal, dan sekitar 30.000 di antaranya ditemukan di Indonesia, dan dari jumlah tersebut, sekitar 7.500 tanaman herbal yang telah diketahui memiliki manfaat atau khasiat, namun hanya 1.200 jenis tanaman herbal yang digunakan oleh masyarakat.

Tanaman herbal merupakan salah satu tumbuhan yang bermanfaat dalam mencegah dan mengobati penyakit. Di antara beberapa bagian tanaman herbal yaitu daun, akar, buah, bunga, dan

kulit kayu, daun merupakan salah satu bagian dari tanaman herbal yang berguna sebagai bahan baku dalam pembuatan obat tradisional. Setiap daun memiliki karakteristik yang unik seperti bentuk, tekstur dan warna. Menurut Jana Wäldchen dan Patrick Mäder, dalam mengklasifikasi daun, bentuk dan tekstur yang seringkali menjadi pembeda utama yang digunakan. Karena minimnya pengetahuan masyarakat terkait tanaman herbal sehingga menyebabkan kurangnya pemanfaatan dari tanaman herbal tersebut [2]. Maka dibutuhkan sebuah sistem yang akurat dalam mengidentifikasi tanaman herbal berdasarkan tekstur daun.

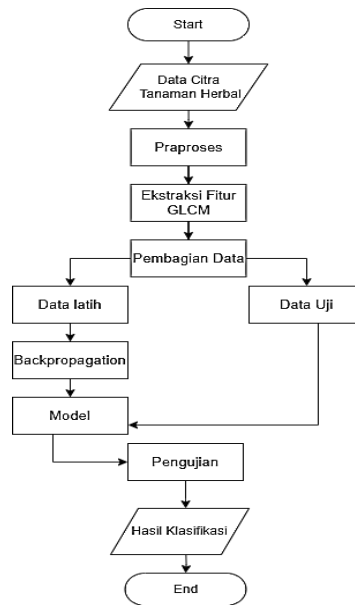
Melalui pendekatan teknologi, identifikasi jenis tanaman herbal berdasarkan ekstraksi tekstur daun dapat dilakukan melalui proses klasifikasi dengan pengolahan citra. Metode ini dikenal sebagai pengenalan pola daun, yang fokus pada analisis karakteristik struktur daun[3]. Pada penelitian Achmad dkk dengan judul "Identifikasi Jenis Jerawat Berdasarkan Tekstur Menggunakan GLCM dan *Backpropagation*", data yang digunakan dibagi menjadi 120 data latih dan 18 data uji. Akurasi identifikasi jenis jerawat mencapai 56,67% dengan 10000 *epoch* dan nilai *error* 0,01. Kendala terjadi karena data yang terbatas, pola jerawat yang mirip, dan kesulitan jaringan dalam identifikasi jenis jerawat, menyebabkan nilai akurasi masih di bawah 70%[4]. Pada penelitian ini ekstraksi ciri tekstur dan ciri bentuk menggunakan GLCM dengan 4 parameter yaitu *contrast*, *correlation*, *energy* dan *homogeneity*. GLCM (*Gray level co-occurrence matrix*) merupakan teknik ekstraksi fitur statistik orde kedua yang menggunakan matriks kookurensi untuk menggambarkan hubungan antara piksel-piksel dalam citra digital. Hal ini dilakukan dengan menghitung kemunculan nilai keabuan yang berbeda dan seberapa sering kombinasi tersebut terjadi dalam jarak tertentu pada citra[5]. Dalam klasifikasi citra teknik yang digunakan yaitu *Artificial Neural Network* (ANN), terdiri dari *neuron-neuron* yang saling terhubung, melibatkan bobot, lapisan, lapisan *neuron*, masukan, keluaran dan lapisan tersembunyi. Dalam ANN, metode-metode yang termasuk di dalamnya yaitu *multilayer perceptron*, *radial basis function networks*, *convolutional neural networks*, *long short-term memory*, *backpropagation*, *gated recurrent unit*, *feedforward* dan *neural networks*. Salah satu metode yang akan digunakan dalam metode penelitian ini yaitu *backpropagation*.

Backpropagation merupakan model algoritma yang pertama kali diperkenalkan oleh Paul Werbos pada tahun 1974, kemudian dikemukakan kembali oleh David Parker di tahun 1982 dan kemudian dipopulerkan oleh Rumelhart dan McClelland pada tahun 1986[6]. *Backpropagation* terdiri dari tiga lapisan, yaitu lapisan *input* dimana data dimasukkan ke dalam jaringan, lapisan tersembunyi yang berfungsi untuk memproses data, dan lapisan *output* tempat hasil akhir, dihasilkan berdasarkan *input* yang diberikan[7]. *Backpropagation* digunakan dalam pembelajaran mesin untuk mengkalkulasi gradien dari fungsi kesalahan terhadap bobot dalam jaringan saraf tiruan. Algoritma ini berperan dalam menyesuaikan bobot jaringan dengan tujuan meningkatkan akurasi hasil *output*. *Backpropagation* melakukan proses pembelajaran terhadap riwayat data sebelumnya, sehingga dapat mengenali pola data dan mengklasifikasikan data yang akan datang. Untuk melakukan klasifikasi, dilakukan dengan dua proses, yaitu proses pelatihan dan pengujian.

Berdasarkan permasalahan yang ada, akan dilakukan klasifikasi tanaman herbal berdasarkan tekstur daun menggunakan *backpropagation* berbasis citra. Pada penelitian ini diharapkan dapat membantu masyarakat untuk mengetahui pemanfaatan tanaman herbal sebagai obat tradisional.

2. METODE PENELITIAN

Penelitian ini menggunakan metode klasifikasi menggunakan jaringan *neural Backpropagation* dan analisis GLCM untuk mengekstrak dan mengolah fitur tekstur daun tanaman herbal. Data citra daun diambil untuk menghasilkan matriks GLCM, yang kemudian digunakan sebagai *input* untuk pelatihan model *Backpropagation*. Metode ini bertujuan untuk mengidentifikasi pola tekstur yang dapat membedakan jenis-jenis tanaman herbal berdasarkan karakteristik visual daun. Proses klasifikasi dapat dilihat melalui *flowchart* pada gambar 1.



Gambar 1. Flowchart Prosedur Penelitian

2.1 Tanaman Herbal

Tanaman herbal merupakan jenis tanaman yang dapat digunakan sebagai alternatif dalam proses penyembuhan secara alami[8]. Beberapa kategori tanaman herbal dapat dibedakan berdasarkan bagian tanaman yang dimanfaatkan. Ada tanaman herbal yang memanfaatkan daunnya, seperti daun salam, daun sirih, dan daun randu. Selain itu, terdapat tanaman herbal yang mengambil bagian batangnya, seperti kayu manis, brotowali, dan pulasari. Jenis lain mencakup tanaman herbal yang menggunakan buahnya, seperti jeruk nipis, ketumbar, dan belimbing wuluh. Selanjutnya, ada juga tanaman herbal yang memanfaatkan bijinya, seperti kecubung, pinang, dan pala. Sementara itu, beberapa tanaman herbal mengambil bagian akarnya, seperti pepaya, aren, dan pulai pandak. Terdapat pula tanaman herbal yang memanfaatkan umbi atau rimpangnya, seperti kencur dan jahe[1]. Selain sebagai obat herbal, tanaman herbal juga menjadi salah satu bahan baku pembuatan obat kimiawi, namun semakin meningkatnya kesadaran terhadap resiko terkait bahan-bahan kimia yang ada pada obat modern, sehingga membuka mata akan pentingnya obat-obatan tradisional. Ada beberapa faktor yang menjadi alasan masyarakat menggunakan tanaman herbal sebagai obat, yaitu tanaman herbal mudah ditemukan dilingkungan sekitar, tanaman herbal juga cenderung lebih alami dan minim efek samping, selain itu harga obat-obatan kimia juga tergolong lebih mahal. Tetapi, biasanya penggunaan tanaman herbal dalam pengobatan seringkali memerlukan waktu yang cukup lama sehingga efeknya tidak segera terlihat

2.2 Gray Level Co-occurrence Matrix (GLCM)

GLCM merupakan pendekatan klasik yang efektif untuk mengekstraksi fitur tekstur pada citra. Aplikasi dari GLCM melibatkan penggunaannya dalam berbagai konteks seperti pengenalan citra, segmentasi citra, pengambilan citra, klasifikasi citra, dan analisis tekstur[9]. berikut tahapan dalam mengekstraksi fitur dengan menggunakan GLCM:

1. mengonversi citra dari format warna RGB ke citra dengan skala abu-abu.
2. menentukan nilai intensitas maksimum untuk citra skala abu-abu.
3. membentuk matriks koordinasi untuk arah 0°, 45°, 90°, 135° dengan nilai intensitas maksimum sebesar 1.
4. menghitung nilai setiap elemen dalam matriks melibatkan perhitungan jumlah kemunculan urutan nilai I di sekitar setiap elemen matriks.
5. normalisasi pada citra dilakukan dengan membagi nilai setiap elemen dalam matriks citra dengan jumlah total nilai dari seluruh matriks.
6. Ekstraksi ciri, yaitu:
 - a. Contrast

$$Contrast = \sum_{i=1}^n \sum_{j=1}^n (i - j)^2 p(i, j) \dots\dots\dots(1)$$

Dimana:

N = Jumlah nilai piksel dalam citra, nilai "n" mencakup jumlah baris dan kolom dalam citra.

(i - j) = posisi piksel pada citra.

(i - j)² = Perbedaan kuadrat intensitas antara piksel i dan j. Rumus ini mengukur perbedaan intensitas antar pasangan piksel.

p(i,j) = Merupakan probabilitas kemunculan pasangan intensitas i dan j. Probabilitas ini mengukur seberapa sering pasangan intensitas tertentu muncul dalam citra.

b. Correlation

$$Correlation = \frac{\sum_{i=1}^n \sum_{j=1}^n (i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i \sigma_j} \dots\dots\dots(2)$$

Dimana :

n = Tingkat keabu-abuan pada citra

σ_i dan σ_j = Deviasi standar dari intensitas piksel i dan j.

μ_i dan μ_j = Nilai rata-rata dari intensitas piksel i dan j.

c. Energy

$$Energy = \sum_{i=1}^n \sum_{j=1}^n p(i,j)^2 \dots\dots\dots(3)$$

d. Homogeneity

$$Homogeneity = \sum_{i=1}^n \sum_{j=1}^n \frac{p(i,j)}{1 + (i - j)^2} \dots\dots\dots(4)$$

2.3 K-Fold Cross Validation

Cross Validation atau validasi silang adalah teknik penambangan data yang bertujuan untuk meningkatkan akurasi hasil dengan membagi data menjadi dua subset, yaitu data pelatihan dan data pengujian. Salah satu metode *cross validation* yang paling umum digunakan adalah *K-fold cross validation*. *K-fold cross validation* secara acak membagi data menjadi K subset atau "lipatan". Setiap iterasi proses ini menggunakan satu lipatan sebagai data uji dan lipatan sisanya sebagai data pelatihan untuk melatih model. Proses ini diulang sebanyak K kali, dan setiap lipatan hanya digunakan satu kali sebagai data uji. Tujuan utamanya adalah untuk mengevaluasi performa model secara keseluruhan dengan menghindari bias yang mungkin timbul dari pembagian awal data[10].

2.4 Backpropagation

Backpropagation adalah suatu algoritma pembelajaran yang digunakan untuk mengurangi tingkat *error* pada suatu model dengan menyesuaikan bobot berdasarkan perbedaan antara *output* yang dihasilkan dan target yang diinginkan[11]. *Backpropagation* melibatkan tiga tahap penting dalam proses pembelajarannya, yakni perambatan maju yang menghasilkan *output* jaringan, perambatan balik yang menilai kesalahan prediksi dengan membandingkannya dengan target yang diinginkan, dan perubahan bobot serta bias untuk mengoreksi dan meningkatkan kinerja jaringan. Pelatihan dengan *backpropagation* dilakukan melalui Langkah-langkah berikut ini:

1. Inisialisasi bobot dan bias dengan bilangan acak kecil.
2. Apabila kondisi berhenti belum terpenuhi atau bernilai salah, proses akan berlanjut dari langkah 3 hingga langkah 10.
3. Untuk setiap data latih, proses akan dilakukan mulai dari langkah 4 hingga langkah 9.
4. Setiap unit masukan menerima sinyal dan meneruskan melalui seluruh *unit* pada *hidden layers* (lapisan tersembunyi).
5. Hitung semua keluaran di unit tersembunyi (Z_j)

$$z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \dots\dots\dots(5)$$

Setelah itu, fungsi aktivasi (sigmoid) digunakan untuk menghitung keluaran pada *hidden layer*.

$$z_j = f(z_{in_j})$$

6. Hitung semua jaringan di *output layer* (Y_k).

$$y_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} \dots\dots\dots(6)$$

Kemudian menerapkan fungsi aktivasi sigmoid untuk menghitung sinyal *output*.

$$y_k = f(y_in_k)$$

Proses pelatihan akan berhenti ketika nilai MSE yang dihasilkan lebih kecil atau sama dengan nilai MSE target, atau jika jumlah iterasi telah mencapai batas maksimum. Jika salah satu dari kondisi ini terpenuhi, proses pelatihan akan dihentikan. Jika tidak, akan dilakukan Langkah berikut.

- Setiap unit *output* di y , kalikan dengan turunan dari fungsi aktivasi untuk menghitung nilai kesalahan .

$$\delta_k = (t_k - y_k) f'(y_in_k) \dots\dots\dots(7)$$

Faktor δ_k ini digunakan untuk menghitung koreksi *error* (ΔW_{jk}) yang akan digunakan untuk memperbarui bobot W_{jk} .

$$\Delta W_{jk} = \alpha \delta_k z_j \dots\dots\dots(8)$$

Dihitung perubahan bias untuk memperbaharui ΔW_{0k} .

$$\Delta W_{0k} = \alpha \delta_k$$

- Pada langkah ini, setiap unit tersembunyi ($z_{j,j} = 1, \dots, p$) melakukan penjumlahan *input* delta yang telah diberi bobot.

$$\delta_in_j = \sum_{k=1}^m \delta_k w_{jk} \dots\dots\dots(9)$$

Untuk menghitung informasi *error*, hasilnya kemudian dikalikan dengan turunan dari fungsi aktivasi yang digunakan.

$$\delta_j = f'(Zin) = \begin{cases} 1 & \text{jika } Zin_j > 0 \\ 0 & \text{jika } Zin_j \leq 0 \end{cases} \dots\dots\dots(10)$$

faktor δ_j ini digunakan untuk menghitung koreksi kesalahan (ΔV_{ij}) yang akan digunakan untuk memperbarui bobot V_{ij} .

$$\Delta V_{ij} = \alpha \delta_j x_i$$

selain itu, koreksi kesalahan bias ΔV_{0j} juga dihitung, yang nantinya akan digunakan untuk memperbaharui V_{0j} .

$$\Delta V_{0j} = \alpha \delta_j \dots\dots\dots(11)$$

- Setiap unit *output* dilakukan perubahan bobot dan bias.

$$w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta W_{jk} \dots\dots\dots(12)$$

Setiap unit tersembunyi dilakukan perubahan bobot dan bias.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij} \dots\dots\dots(13)$$

2.5 Confusion Matrix

Confusion Matrix adalah suatu tabel yang menggambarkan jumlah data uji yang diklasifikasikan dengan benar (*true positive dan true negative*) dan jumlah data uji yang salah (*false positive dan false negative*)[12]. Model *confusion matrix* dapat dilihat pada gambar 2.

		Predicted	
		Positive	Negative
actual	Positive	TP	FP
	Negative	FN	TN

Gambar 2. *Confusion Matrix*

1. Sensitivitas

Sensitivitas mengukur kemampuan model untuk mengidentifikasi semua kejadian positif. Hal ini mengukur kemampuan model untuk mendeteksi kasus positif yang sebenarnya.

$$Sensivitas = \frac{TP}{TP+FN}$$

2. Spesifisitas

Spesifisitas mengukur kemampuan model untuk mengidentifikasi semua kejadian negatif. Hal ini mengukur kemampuan model untuk membedakan antara kasus positif dan negatif.

$$Spesifisitas = \frac{TN}{TN+FP} \dots\dots\dots(15)$$

3. Akurasi

Akurasi mengukur seberapa baik suatu model mengklasifikasikan semua data dengan benar, baik positif maupun negatif.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots(16)$$

3. HASIL DAN ANALISIS

3.1 Implementasi Sistem

Klasifikasi jenis tanaman herbal berdasarkan tekstur daun dilakukan menggunakan algoritma *backpropagation*. Tahap pertama adalah *preprocessing data*, di mana dataset diubah menjadi ukuran lebih kecil dan dikonversi ke citra *grayscale*. Setelah itu, ekstraksi fitur dilakukan menggunakan GLCM. Data yang telah diproses dan diekstraksi fiturnya digunakan untuk melatih model *backpropagation*, yang kemudian diuji dengan data tes.

1. Data Citra

Data yang digunakan dalam penelitian ini yaitu citra daun tanaman herbal dari 20 jenis tanaman, dengan masing-masing jenis tanaman diambil sampel sebanyak 10 citra dengan ukuran setiap citranya 3024 x 4032 piksel, sehingga total terdapat 200 citra. Untuk mengidentifikasi daun berdasarkan tekstur, citra diambil dari sisi depan dan belakang daun dengan rotasi 0°, 45°, 90°, 135° dan 180°. Pengambilan citra dilakukan menggunakan kamera dengan jarak 30 cm antara kamera dan objek dengan menggunakan *background* putih, kemudian citra disimpan dalam format file *.jpg

2. Praproses Citra

Pada tahap praproses, dari 200 citra yang telah diperoleh, setiap citra memiliki ukuran asli 3024 x 4032 piksel. Karena ukuran piksel yang terlalu besar untuk proses analisis lebih lanjut, dilakukan *resizing* citra sehingga memiliki ukuran yang lebih kecil, yaitu 300 x 300 piksel. *Resize* ini membantu mengurangi dimensi citra tanpa mengorbankan informasi visual utama. Selanjutnya, citra-citra tersebut dikonversi ke citra keabuan (*grayscale*). Konversi keabuan dilakukan untuk menghilangkan informasi warna dan menjadikan citra dalam representasi yang lebih sederhana, namun tetap mempertahankan informasi tekstur yang penting.

3. Ekstraksi Fitur GLCM

Setelah melewati tahap praproses, yang melibatkan proses *resize* dan konversi citra ke dalam *grayscale*, pada tahap ini setiap citra daun akan diekstraksi fiturnya untuk mencari nilai *contrast*, *correlation*, *energy* dan *homogeneity*.

a. Implementasi Ekstraksi Fitur GLCM

Implementasi fungsi *calculate_glcm_features (image)* dalam Python menghitung dan mengekstraksi fitur GLCM dari citra *grayscale*. Fungsi ini mendefinisikan empat sudut (0°, 45°, 90°, dan 135°) dan menggunakan modul *skimage.feature* untuk menghitung GLCM dengan jarak 1 piksel dan ukuran matriks 3x3. Setelah GLCM dihitung dan dinormalisasi, fitur tekstur seperti kontras, korelasi, energi, dan homogenitas diekstraksi dengan *graycoprops*. Hasil ekstraksi dari setiap sudut disimpan dalam *dictionary* dan kemudian digabungkan menjadi satu vektor fitur tunggal per citra. Vektor-vektor fitur ini dikumpulkan dalam list *glcm_feature_vectors* untuk analisis lanjut. Hasil ekstraksi terlihat pada tabel 1.

Tabel 1. Hasil ekstraksi fitur GLCM

No	Sudut	Contrast	Correlation	Energy	Homogeneity
1	0°	89.8577	0.9777	0.8974	0.9046
	45°	170.0205	0.9580	0.8960	0.9026
	90°	125.9259	0.9688	0.8971	0.9040
	135°	125.1881	0.9691	0.8967	0.9034
2	0°	103.5117	0.9743	0.8972	0.9045
	45°	119.3184	0.9705	0.8964	0.9038
	90°	109.1952	0.9729	0.8959	0.9039

	135°	168.3765	0.9584	0.8965	0.9020

	0°	40.1661	0.9811	0.9398	0.9477
200	45°	34.8599	0.9837	0.9397	0.9474
	90°	52.6442	0.9753	0.9396	0.9466
	135°	89.9933	0.9579	0.9386	0.9446

Setelah mendapatkan hasil ekstraksi fitur, langkah utama sebelum pelatihan model adalah normalisasi fitur GLCM, dengan menggunakan *Z-score standardization* untuk memastikan semua fitur memiliki skala yang sama. Ini mengurangi variabilitas antar fitur, mempercepat konvergensi model, dan meningkatkan stabilitas serta akurasi pelatihan. Untuk hasil normalisasi fitur terlihat pada tabel 2.

Tabel 2. Hasil normalisasi fitur GLCM

No	Sudut	Contrast	Correlation	Energy	Homogeneity
1	0°	0.0934	0.2345	-0.4298	-0.6823
	45°	0.3430	0.1512	-0.4282	-0.6513
	90°	0.3322	0.0944	-0.4267	-0.6714
	135°	-0.0483	0.2943	-0.4169	-0.6328
2	0°	0.2857	0.1727	-0.4336	-0.6884
	45°	-0.1016	0.3138	-0.4218	-0.6226
	90°	0.1514	0.1624	-0.4301	-0.4407
	135°	0.3290	0.1553	-0.4331	-0.6646
200
	0°	-0.6064	0.2968	0.3403	0.3547
	45°	-0.8424	0.4844	0.3527	0.3975
	90°	-0.4597	0.2006	0.3380	0.3362
	135°	-0.3559	0.1484	0.3334	0.3307

4. Pelatihan Model

Setelah normalisasi data, langkah berikutnya adalah pelatihan model dengan *backpropagation*. Pemilihan arsitektur jaringan, seperti jumlah neuron di lapisan tersembunyi dan fungsi aktivasi, sangat mempengaruhi kinerja model. *Backpropagation* memperbarui bobot dan bias berdasarkan error antara prediksi dan nilai aktual untuk meminimalkan error secara iteratif selama beberapa *epoch* hingga konvergensi. Bobot dan bias diinisialisasi dengan nilai acak dari distribusi normal standar, dikalikan dengan $\sqrt{2 / \text{input_size}}$ untuk lapisan tersembunyi dan $\sqrt{2 / \text{hidden_size}}$ untuk lapisan *output*, sementara bias diinisialisasi dengan nol. Fungsi aktivasi yang digunakan adalah ReLU untuk lapisan tersembunyi dan sigmoid untuk lapisan *output*. *Forward propagation* menghitung aktivasi lapisan tersembunyi dan *output* berdasarkan *input*. *Backpropagation* menghitung gradien bobot dan bias menggunakan *error* antara nilai target dan *output* prediksi, lalu memperbaruinya dengan metode *gradient descent* berdasarkan laju pembelajaran (*learning_rate*). *Neural network* menggunakan ReLU untuk lapisan tersembunyi dan *sigmoid* untuk lapisan *output*, dengan *backpropagation* untuk memperbarui bobot dan bias guna meningkatkan akurasi prediksi.

5. K-fold Cross Validation

Berdasarkan *k-fold cross-validation* dengan k=5, data dibagi menjadi lima bagian yang sama. Setiap iterasi menggunakan satu bagian sebagai data uji dan empat bagian sebagai data latih. Pada setiap *fold*, data uji terdiri dari 40 citra yang mewakili 20 jenis tanaman herbal, dengan dua citra per kelas. Hasil *backpropagation* dengan *epoch* 300, 500, 1000, 3000, dan 5000 menunjukkan akurasi pelatihan dan pengujian yang bervariasi di setiap *fold*. akurasi untuk setiap *epoch* terlihat pada tabel 3.

Tabel 3. Akurasi pelatihan

Fold	Epoch	Training Accuracy	Testing Accuracy
1	300	0,96781250	0,96375000
	500	0,977500000	0,97250000
	1000	0,98281250	0,97500000
	3000°	0,98625000	0,96750000
	5000	0, 99531250	0, 97750000
2	300	0,97062500	0,97000000
	500	0,97812500	0,96750000

	1000	0,98625000	0,97750000
	3000	0,99187500	0,97375000
	5000	0,99312500	0,98500000
3	300	0,96562500	0,96375000
	500	0,97218750	0,97125000
	1000	0,98031250	0,97750000
4	3000	0,98562500	0,98250000
	5000	0,99250000	0,98625000
	300	0,97500000	0,96250000
5	500	0,97437500	0,97125000
	1000	0,97906250	0,97375000
	3000	0,99062500	0,97750000
5	5000	0,99093750	0,97625000
	300	0,96500000	0,96750000
	500	0,97781250	0,96375000
5	1000	0,98281250	0,97125000
	3000	0,98625000	0,97375000
	5000	0,99343750	0,97625000

Berdasarkan perbandingan *epoch*, *epoch* 5000 menunjukkan kinerja terbaik dengan akurasi tertinggi pada data latih dan data uji. Rata-rata akurasi pengujian mencapai 97.75% dari lima *fold*, dengan stabilitas antara 97.75% hingga 99.34% di setiap lipatan. *Epoch* 5000 dipilih karena memberikan kinerja optimal dan stabilitas yang tinggi pada pengujian, menunjukkan model mencapai tingkat pembelajaran yang baik tanpa meningkatkan kompleksitas secara berlebihan.

6. Pengujian Sistem

Setelah menyelesaikan proses *k-fold*, model akan diuji lebih lanjut menggunakan *confusion matrix* untuk memastikan kinerja sistem yang optimal dalam mengklasifikasikan jenis tanaman herbal berdasarkan tekstur daun. *Confusion matrix* ini menunjukkan sejauh mana model berhasil memprediksi kelas yang sesuai dengan dataset sebenarnya. Hasil pengujian *confusion matrix* dari 5 *fold* terlihat pada tabel 4.

Tabel 4. Hasil pengujian *confusion matrix*

Skenario	Sensifitas	spesifisitas	Akurasi
Fold 1	88,89%	50%	85%
Fold 2	82,76%	45,45%	72,5%
Fold 3	93,33%	30%	77,5%
Fold 4	83,33%	75%	82,5%
Fold 5	91,43%	40%	85%
Jumlah			82,5%

Berdasarkan hasil pengujian menggunakan matriks konfusi, *fold* ke-5 menunjukkan akurasi tertinggi sebesar 85% dalam sistem klasifikasi jenis tanaman herbal. Meskipun *fold* ke-1 juga mencapai akurasi 85%, pemilihan *fold* ke-5 didasarkan pada sensitivitas yang lebih tinggi (91,43%), menunjukkan bahwa model pada *fold* ke-5 mampu mengidentifikasi dengan baik sampel yang sebenarnya positif. Sedangkan *fold* ke-1 memiliki sensitivitas 88,89% dengan spesifisitas 50%. Dengan akurasi keseluruhan 82,5%, sistem klasifikasi menunjukkan kinerja yang baik pada model dalam mengklasifikasikan tanaman herbal.

Dari hasil pelatihan model dengan menggunakan *k-fold* menghasilkan hasil pengujian jenis tanaman herbal dengan *confusion matrix* pada *fold* terbaik sebagai berikut:

7. KESIMPULAN

Klasifikasi jenis tanaman herbal berdasarkan tekstur daun menggunakan metode *backpropagation* berhasil diimplementasikan. Model yang digunakan dalam sistem klasifikasi menunjukkan kemampuan yang baik dalam mengidentifikasi tanaman herbal dan mendapatkan akurasi terbaik yaitu 85%. Selain itu, akurasi sistem secara rata-rata yang diperoleh sebesar 80,5%.

REFERENSI

- [1] P. Melati, "Penerapan K-Nearest Neighbor Pada Pengolahan Citra Digital Menggunakan Local Binary Pattern Untuk Klasifikasi Tanaman Herbal," 2020, [Online]. Available: <http://repository.uin-suska.ac.id/29658/>
- [2] M. Meiryama, S. Devella, and S. M. Adelfi, "Klasifikasi Daun Herbal Berdasarkan Fitur Bentuk dan Tekstur Menggunakan KNN," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 3, pp. 2573–2584, 2022, doi: 10.35957/jatisi.v9i3.2974.
- [3] V. Roviqoh, H. Dan, and S. Lukman, "Klasifikasi Citra Daun Menggunakan Metode Jaringan Syaraf Tiruan Backpropagation Berbasis Matlab," *Semin. Nas. Teknol. Inf. dan Komun. STI&K*, vol. 7, no. 1, 2023.
- [4] Y. F. Achmad, A. Yulfitri, and M. B. Ulum, "Identifikasi Jenis Jerawat Berdasarkan Tekstur Menggunakan GLCM dan Backpropagation," *J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer)*, vol. 20, no. 2, p. 139, 2021, doi: 10.53513/jis.v20i2.4747.
- [5] M. A. Abilisa, I. R. Magdalena, and Sofia Sa'idah, "Identifikasi Jenis Kulit Manusia Menggunakan Metode Glcm Dan Lvq Berbasis Android," *eProceedings ...*, vol. 8, no. 1, pp. 182–197, 2021, [Online]. Available: <https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/viewFile/14265/14049>
- [6] J. R. Simanungkalit, H. Havaluddin, H. S. Pakpahan, N. Puspitasari, and M. Wati, "Algoritma Backpropagation Neural Network dalam Memprediksi Harga Komoditi Tanaman Karet," *Ilk. J. Ilm.*, vol. 12, no. 1, pp. 32–38, 2020, doi: 10.33096/ilkom.v12i1.521.32-38.
- [7] G. Eka Saputra, A. Faisal, and D. Ahmad Apandi, "Perbandingan Penerapan Algoritma Neural Network Backpropagation Dengan Optimasi Algoritma Lbfgs Dan Sgd Untuk Prediksi Penyakit Jantung," *Semin. Nas. Teknol. Inf. dan Komun. STI&K*, vol. 4, no. 1, pp. 1–10, 2020.
- [8] A. Arifin, J. Hendyli, and D. E. Herwindiati, "Klasifikasi Tanaman Obat Herbal Menggunakan Metode Support Vector Machine," *Comput. J. Comput. Sci. Inf. Syst.*, vol. 5, no. 1, p. 25, 2021, doi: 10.24912/computatio.v1i1.12811.
- [9] A. Pariyandani, "Analisis Metode Naive Bayes dalam Pengklasifikasian Citra Ikan Berformalin dengan Memanfaatkan Ekstraksi Fitur GLCM," 2022, [Online]. Available: <https://repositori.uma.ac.id/handle/123456789/17174>
- [10] R. R. R. Arisandi, B. Warsito, and A. R. Hakim, "Aplikasi Naive Bayes Classifier (Nbc) Pada Klasifikasi Status Gizi Balita Stunting Dengan Pengujian K-Fold Cross Validation," *J. Gaussian*, vol. 11, no. 1, pp. 130–139, 2022, doi: 10.14710/j.gauss.v11i1.33991.
- [11] P. N. Napitupulu, A. R. Damanik, and J. E. Napitupulu, "Implementasi Algoritma Backpropagation Jaringan Syaraf Tiruan Untuk Prediksi Angka Harapan Hidup Di Kota Jambi," *J. JPILKOM (J. Penelit. Ilmu Komput.)*, vol. 1, no. 1, pp. 10–15, 2023.
- [12] D. Normawati and S. A. Prayogi, "Implementasi Naive Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J. Sains Komput. Inform. (J-SAKTI)*, vol. 5, no. 2, pp. 697–711, 2021.