



Classification of Herbal Leaves Using Support Vector Machine (SVM)

Yakub Takandiwa^{1*}, Pingky Alfa Ray Leo Lede²

^{1,2} Universitas Kristen Wira Wacana Sumba
yakubtakandiwa2002@gmail.com^{1*}, pingky.leo.lede@unkriswina.ac.id²

Abstract

Indonesia is a country with high biodiversity, including various types of herbal leaves with potential use as traditional medicine. Manual identification of herbal leaves often encounters challenges due to morphological similarities among species and the limited availability of experts, thereby necessitating a fast and accurate technology-based classification method. This study aims to classify 10 types of herbal leaves using the Support Vector Machine (SVM) algorithm with a Radial Basis Function (RBF) kernel. The dataset consists of 3,500 leaf images (350 images per class), from which color features (HSV), texture features (Gray Level Co-occurrence Matrix/GLCM), and shape features (area, perimeter, and aspect ratio) were extracted. The research process includes preprocessing, feature extraction, data splitting into training and testing sets, model training, and performance evaluation. Evaluation was conducted using a confusion matrix, with accuracy as the primary metric due to the balanced class distribution. Precision, recall, and F1-score were employed as supporting evaluation metrics. The results indicate that the SVM model with an RBF kernel successfully classified the 10 types of herbal leaves with an accuracy of 81.29%. Based on per-class analysis, the highest performance was achieved in the Papaya class with an F1-score of 90.00%, followed by Jambu Biji (89.36%) and Pandan (87.14%). In contrast, the lowest performance was observed in the Aloe Vera class with an F1-score of 65.71% and Lime with 70.00%. The model achieved an average precision of 81.16%, recall of 80.73%, and F1-score of 80.94%. Misclassifications primarily occurred among classes with high morphological similarity, such as Aloe Vera, which was frequently misclassified as Pandan (9 cases) and Basil (5 cases). The system has been implemented as a Graphical User Interface (GUI) application that allows users to upload leaf images and obtain classification results along with information regarding their herbal benefits within 1–2 seconds.

Keywords: Classification, Confusion Matrix, Herbal Leaves, RBF Kernel, Support Vector Machine (SVM)

1. Introduction

Indonesia is recognized as one of the countries with the richest biodiversity in the world and is classified as a megabiodiversity country, referring to nations with exceptionally high levels of biological diversity. According to data from the Health Research and Development Agency (Balitbangkes), more than 30,000 plant species grow in Indonesia, of which approximately 9,600 species are known to have medicinal properties, and over 300 species are routinely utilized as traditional herbal medicine or *jamu*. Among various plant parts, leaves are the most frequently used because they contain diverse bioactive compounds such as flavonoids, alkaloids, tannins, and essential oils. These compounds function as antioxidants, antibacterial agents, and anti-inflammatory substances that are highly valuable in healthcare applications. Despite their widespread use, the identification of herbal leaves is still largely performed manually. This approach presents several limitations. The limited number of experts restricts access to reliable knowledge regarding herbal plant identification. In addition, morphological similarities among species often cause confusion, particularly for non-experts. For instance, basil leaves (*Ocimum sanctum*) are frequently mistaken for sweet basil (*Ocimum basilicum*), even though they possess different medicinal properties. Such misidentification may lead to serious consequences, including reduced treatment effectiveness or unintended side effects due to inappropriate usage. In the era of digital transformation, there is an urgent need to develop technology-based systems capable of identifying herbal leaves in a more objective, fast, and accurate manner. Machine learning provides a promising solution, as it can recognize complex patterns from leaf image data through feature extraction processes involving color, texture, and shape characteristics. Several previous studies have demonstrated the potential of this approach. [1] applied Gray Level Co-occurrence Matrix (GLCM) texture extraction combined with the Support Vector Machine (SVM) algorithm to classify clove leaves and achieved relatively high accuracy. Similarly, [2] reported that herbal rhizome classification using color, shape, and texture feature extraction with the SVM algorithm achieved an accuracy of up to 77.6%, although visual similarity among rhizome types remained a challenge. However, most previous studies remain limited in terms of dataset size, variety of leaf types, and algorithm parameter configuration. These limitations raise an important question regarding whether existing approaches can be generalized to classify multiple types of herbal leaves with high morphological similarity. Furthermore,

many studies primarily focus on algorithmic accuracy without emphasizing the practical implementation of the system to assist the public in real-world herbal leaf identification. Therefore, this study proposes the implementation of the Support Vector Machine (SVM) algorithm for the classification of herbal leaves using a large image dataset. The classification process is based on the extraction of color features (HSV), texture features (GLCM), and shape features (area, perimeter, and aspect ratio). The study focuses on 10 types of herbal leaves and evaluates the classification performance in terms of accuracy, precision, recall, and F1-score. In addition to performing classification, the developed system provides textual information regarding the medicinal benefits of each identified leaf, derived from scientific literature and credible health-related sources rather than clinical trials. This research is expected to contribute academically by providing an evaluation of SVM performance in multi-class herbal leaf classification using comprehensive feature extraction. Practically, it aims to assist the public in identifying herbal leaves more accurately and efficiently, reduce misidentification risks, and support educational development in computer vision and data science. Furthermore, the results of this study may serve as a foundation for developing technology-based applications in phytotherapy and herbal pharmaceutical industries.

2. Literature Review

2.1. Herbal Leaves

Herbal leaves are widely used in traditional medicine due to their rich content of secondary metabolites, such as flavonoids, alkaloids, and phenolic compounds. These bioactive substances contribute to various medicinal properties, including antipyretic, analgesic, antitumor, antiviral, antifungal, and anti-inflammatory effects. According to [3], the pharmacological potential of herbal leaves plays a significant role in supporting preventive and curative healthcare practices, particularly in countries with strong traditional medicine cultures such as Indonesia.

2.2. Feature Extraction

Feature extraction is a crucial stage in image processing that aims to identify and extract distinctive characteristics from images to support pattern recognition and classification processes. In leaf image classification, extracted features generally include color, texture, and shape attributes. Through feature extraction, raw image data consisting of pixel values are transformed into numerical parameters that can be effectively processed by classification algorithms. Texture features in this study are extracted using the Gray Level Co-occurrence Matrix (GLCM) method. GLCM represents the frequency of occurrence of pixel pairs with specific gray-level values at a defined spatial distance and orientation. From the GLCM matrix, four primary statistical parameters are commonly derived: contrast, which measures intensity variation between neighboring pixels; correlation, which indicates the linear relationship between pixel pairs; homogeneity, which reflects uniformity within the texture; and energy, which represents textural regularity. [4] emphasize that GLCM-based texture features are effective in distinguishing objects based on grayscale distribution patterns.

2.3. Machine Learning

Machine Learning (ML) is a subfield of Artificial Intelligence (AI) that enables systems to learn patterns from data through algorithmic processes without explicit programming instructions. ML algorithms can automatically analyze data, recognize patterns, and perform predictions or classifications on new data instances. According to [5], machine learning techniques are widely applied in image recognition tasks due to their ability to handle high-dimensional and complex datasets. In machine learning, there are two types of data: training data and test data. Training data is used to train algorithms, while test data serves to evaluate the performance of trained algorithms when faced with new data that has never been included in training data [6].

2.4. Classification

Classification is one of the primary tasks in machine learning, aiming to assign data instances to predefined categories or classes. Various algorithms can be employed for classification, ranging from probabilistic methods such as Naïve Bayes to more complex models such as Neural Networks. These algorithms utilize different strategies, including probability estimation, distance measurement, decision tree construction, or optimal boundary determination between classes [5]. The selection of an appropriate classification algorithm depends on the characteristics of the dataset and the problem domain.

2.5. Support Vector Machine (SVM)

SVM is a powerful classification method for separating data into specific classes by forming optimal boundary lines (hyperplanes). A hyperplane is a line (in 2 dimensions), plane (in 3 dimensions), or higher dimensional space that separates data by a maximum margin. With the use of the kernel, SVM is able to handle non-linear data [7].

The hyperplane in SVM is mathematically expressed as:

$$w \cdot x + b = 0 \quad (1)$$

where w is the weight vector, x is the input feature vector, and b is the bias term.

To ensure correct separation between two classes, each training sample must satisfy the maximum margin condition:

$$y_i(w \cdot x_i + b) \geq 1 \quad (2)$$

where x_i represents the i -th training data point and $y_i \in \{+1, -1\}$ denotes the class label. The use of $+1$ and -1 labels simplify the mathematical formulation and ensures symmetry in optimization.

After the training process, SVM determines support vectors, corresponding coefficients α_i , and bias b . Classification of a test sample is performed using the decision function:

$$f(x) = \sum_{i=1}^{n_{SV}} (\alpha_i y_i) \cdot K(x_i, x) + b \quad (3)$$

where n_{SV} denotes the number of support vectors, α_i are the learned coefficients, y_i are the support vector labels, $K(x_i, x)$ is the kernel function, and b is the bias. The sign of $f(x)$ determines the class membership of the test data.

2.6. Radial Basis Function (RBF) Kernel

The Radial Base Function (RBF) kernel is one of the kernels commonly used in the Support Vector Machine (SVM) algorithm to handle data that cannot be separated linearly. In herbal leaf image classification, features such as color, texture, and shape often have complex patterns and similarities between classes, making linear separation less effective. [7] stated that the use of the RBF kernel in SVM was able to improve the model's ability to capture non-linear relationships in herbal leaf image data compared to other classification methods.

$$K(x_i, x_j) = \exp(-\gamma |x_i - x_j|^2) \quad (4)$$

where x_i is the training data feature vector, x_j is the test data feature vector, and γ is the kernel parameter that regulates the amount of influence of each support vector. RBF kernel values are in the range of 0 to 1, where values close to 1 indicate a high degree of data similarity, while values close to 0 indicate large differences between data.

2.7. Python Programming Language

Python is a high-level programming language developed by Guido van Rossum and released for the first time in 1991. Python is an interpretive programming language that is multiplatform, easy to learn, and has a clear and easy to understand syntax. This language can be used for various purposes and is supported by many large open-source libraries and communities [8]. Python can be used through various operating system services, such as Windows (python), Linux, Android (Google Play), MacOS, and so on [9].

2.8. Previous Studies

Several previous studies have explored the application of machine learning methods in plant and herbal leaf classification. [1] implemented the Support Vector Machine (SVM) algorithm to classify clove leaf types based on texture features. Their study demonstrated that SVM was capable of achieving high classification accuracy using texture-based feature extraction, highlighting the effectiveness of SVM in distinguishing leaf characteristics derived from grayscale distribution patterns. [7] conducted a comparative study on herbal leaf classification using K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) algorithms with Fourier Descriptor features. Their findings indicated that both algorithms were effective in performing classification tasks. However, SVM showed more stable performance, while KNN offered a simpler implementation with results that were relatively comparable to SVM. Similarly, [10] investigated herbal leaf classification using KNN combined with Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) feature extraction methods. The study utilized a dataset consisting of 450 images across 15 types of leaves. The results revealed that the HOG + KNN combination achieved an accuracy of 92.67%, while LBP + KNN reached 88.67%, demonstrating that shape and texture features significantly contribute to classification performance. Beyond plant classification, Ranti et al. (2022) [5] compared several machine learning algorithms, including Naïve Bayes, KNN, and SVM, for breast cancer classification. Their results showed that SVM achieved the highest accuracy among the evaluated methods, further confirming its robustness and strong generalization capability in classification problems. In addition, [2] applied the Support Vector Machine method for herbal medicinal plant classification using GLCM texture features. The study employed Randomized Search Cross-Validation for parameter optimization with the RBF kernel. The best-performing model achieved 83.9% accuracy on training data and 77.6% on testing data. These findings emphasize the importance of parameter tuning and kernel selection in improving SVM classification performance.

3. Research Method

All paragraphs must be justified alignment. With justified alignment, both sides of the paragraph are straight.

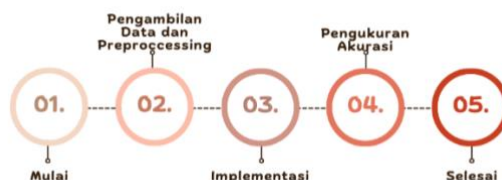


Fig. 1: Research Flow

Figure 1 displays the flow of research carried out by researchers. Starting from data retrieval and preprocessing, at this stage a dataset of herbal leaves is collected to extract features from the image. Consists of 10 types of herbal leaves which are taken and then extracted and then stored in the form of CSV (comma separate value) files. The next stage is implementing SVM to carry out classification on herbal leaves. The next stage is measuring the accuracy of both methods, namely SVM, to determine its accuracy.

3.1. Data Collection and Preprocessing

Data collection was carried out by collecting datasets from open sources on the site: <https://data.mendeley.com/datasets/s82j8dh4rr/1>. In the dataset there are 10 types of herbal leaves, namely Starfruit, Jambu Biji, Lime, Basil, Aloe Vera, Jackfruit, Pandan, Papaya, Celery and Betel, each of which has its own folder so that each folder has 350 leaf images and the entire dataset has $350 \times 10 = 3500$ leaf images. Then, each folder name will be used as a label.



Fig. 2: Example of a Collection of Herbal Leaf Datasets

Then a collection of herbal leaf datasets will extract the features which are then used for classification. The leaf features taken are HSV color (mean value (mean) and standard deviation (std) from Hue, Saturation and Value), GLCM texture (contrast, homogeneity, energy, correlation) and shape (area, perimeter, aspect ratio).

Table 1: Extraction Results Features

h_mean	v_mean	h_std	v_std	contrast	..	energy	correlation	area	perimeter	aspect_ratio	label
17.14	231.44	41.87	64.50	209.68	..	0.5941	0.9784	16129	508	1.0	Belimbing Wuluh
7.16	235.39	19.23	53.31	120.66	..	0.8651	0.9826	16129	508	1.0	Jambu Biji
8.28	229.53	19.07	59.67	123.76	..	0.8272	0.9859	16129	508	1.0	Jeruk Nipis
7.58	230.83	18.47	59.92	126.18	..	0.8433	0.9852	16129	508	1.0	Kemangi
4.42	239.56	14.52	51.22	141.61	..	0.9050	0.9769	16129	508	1.0	Lidah Buaya
17.14	231.44	41.87	64.50	209.68	..	0.5941	0.9784	16129	508	1.0	Nangka
7.16	235.39	19.23	53.31	120.66	..	0.8651	0.9826	16129	508	1.0	Pandan
8.28	229.53	19.07	59.67	123.76	..	0.8272	0.9859	16129	508	1.0	Pepaya
7.58	230.83	18.47	59.92	126.18	..	0.8433	0.9852	16129	508	1.0	Seledri
4.42	239.56	14.52	51.22	141.61	..	0.9050	0.9769	16129	508	1.0	Sirih

3.2. Implementation of SVM

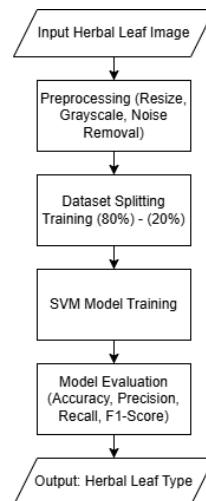


Fig. 3 SVM Implementation Flow

The herbal leaf classification process using the Support Vector Machine (SVM) method begins with collecting images of herbal leaves from various types of plants such as betel, basil, pandan, papaya and celery. The image obtained then goes through the preprocessing stage, namely uniforming the image size (resize), changing the color image to grayscale, and reducing lighting or noise interference so that the image quality is better for analysis. Next, feature extraction is carried out to take the characteristics of each leaf. Features used in this research include color characteristics with HSV space, texture characteristics using the Gray Level Co-occurrence Matrix (GLCM) method in the form of contrast, correlation, energy and homogeneity, as well as shape characteristics in the form of area, perimeter and aspect ratio. The results of feature extraction are then represented in the form of numerical data and divided into two parts, namely training data of 80% and test data of 20%. Trainer data is used to build the SVM model, while test data is used to measure model performance. At the training stage, SVM works by looking for the best hyperplane that can separate types of leaves based on the features they have. Kernel selection is an important step, where the Linear kernel is used for linear data, while the RBF kernel is more suitable for non-linear data such as herbal leaves which have visual similarities between classes. After the model is trained, the evaluation stage is carried out using a confusion matrix to calculate accuracy, precision, recall and F1-score values. The final result is the predicted output of herbal leaf types which can be used as additional information regarding the classification and efficacy of each leaf.

At this stage, the SVM model has been trained using the herbal leaf training dataset. The training results produce a support vector, coefficient value ($\alpha_i y_i$) and bias (b). The following is an example of SVM calculations in carrying out classification.

To illustrate the classification mechanism, consider a binary classification scenario involving two herbal leaf classes: Belimbing Wuluh (+1) and Jambu Biji (-1). For simplification, assume that two texture features (contrast and energy) are extracted from each sample. The class labels +1 and -1 serve only as categorical encodings.

During the training phase, the Support Vector Machine (SVM) algorithm determines the optimal separating hyperplane by identifying support vectors, computing their corresponding coefficients ($\alpha_i y_i$), estimating the bias term (b), and optimizing the Radial Basis Function (RBF) kernel parameter (γ). In this example, the training process produces 7 support vectors, with a bias value (b): -0.1737 and kernel parameter (γ): 0.1111.

Given a new test sample represented by the feature vector:

$$x = [10.0, 230.0, 22.0, 58.0, 135.0, 0.78, 0.979, 13000, 1.01]$$

the classification is performed using the SVM decision function:

$$f(x) = \sum_{i=1}^{n_{sv}} (\alpha_i y_i) \cdot K(x_i, x) + b \quad (4)$$

where the RBF kernel is defined as:

$$K(x_i, x) = e^{-\gamma \|x_i - x\|^2} \quad (5)$$

The kernel function measures the similarity between the test sample and each support vector, producing values in the range [0,1]. Each similarity value is weighted by its corresponding coefficient ($\alpha_i y_i$), and the results are summed together with the bias term.

Based on the computed contributions of all support vectors, the decision function yields:

$$f(x) = 1.0803$$

Since $f(x) > 0$ the test sample is classified as Class +1 (Belimbing Wuluh).

3.3. SVM Performance Evaluation

The performance of the Support Vector Machine (SVM) model was evaluated using a confusion matrix. In multi-class classification, the confusion matrix is represented as a K×K table, where K denotes the number of classes (in this study, 10 herbal leaf classes). Rows correspond to the true class labels, while columns represent the predicted class labels. Diagonal elements indicate correctly classified samples for each class, whereas off-diagonal elements represent misclassifications. For each class i , performance metrics are computed using a one-vs-rest approach. The evaluation components are defined as follows:

1. TP_i = the number of samples from class i correctly predicted as class i .
2. FP_i = the number of samples from other classes incorrectly predicted as class i
3. FN_i = the number of samples from class i incorrectly predicted as other classes
4. TN_i = the number of samples that do not belong to class i and are not predicted as class i .

$$Accuracy = \frac{\sum_{i=1}^K TP_i}{N} \quad (6)$$

where N represents the total number of test samples.

In addition to accuracy, class-level performance is evaluated using precision, recall, and F1-score. Precision for class i is calculated as:

$$Precision_i = \frac{TP_i}{TP_i + FP_i} \quad (7)$$

which measures the proportion of correctly predicted samples among all samples predicted as class i .

Recall (or sensitivity) for class i is defined as:

$$Recall_i = \frac{TP_i}{TP_i + FN_i}$$

which indicates the proportion of correctly identified samples among all actual samples of class i. The F1-score, which represents the harmonic mean of precision and recall, is computed as:

$$F1_i = 2 \cdot \frac{Precision_i \cdot Recall_i}{Precision_i + Recall_i} \tag{5}$$

Since the dataset used in this study contains a balanced number of samples for each class, accuracy is considered the primary evaluation metric. Precision, recall, and F1-score are reported as complementary metrics to provide a more detailed analysis of class-wise performance.

4. Results and Discussion

4.1. Analysis

The implementation of herbal leaf classification in this study began with a preprocessing stage, in which image data were processed to extract color (HSV), texture (GLCM), and shape features (area, perimeter, and aspect ratio). The purpose of this process was to transform raw image data into numerical feature representations suitable for training the SVM model. Each of the 3,500 images generated a corresponding feature vector along with its class label. The extracted data were stored in CSV format and used as input for model training. The preprocessing and feature extraction procedures were implemented using the Python programming language along with relevant scientific computing libraries.

	A	B	C	D	E	F	G	H	I	J	K	L	
1	h_mean_s	mean_v	mean_h	std_s	std_v	std_contrast	homogeneity	energy	correlation	area	perimeter	aspect_ratio	label
2	16.89434814453125	27.549041748046875	231.47991943359375	41.495725566295704	76.30033542662537	64.41214648714							
3	7.124114990234375	16.400253295898438	235.42723083496094	19.058902568092908	45.86782056558626	53.26132138589							
4	8.171707153320312	26.147506713867188	229.62196350097656	18.842428702778192	62.73997108016134	59.557805695858							
5	5.787658669140625	22.183197021484375	230.76593017578125	18.56010916290716	56.4125519301098	60.0441878516535							
6	4.401153564453125	15.604660034179688	239.5938720703125	14.397606456190415	52.71971036504764	51.089528283128							
7	7.9317474365234375	25.401092529296875	229.3173065185547	18.26546448311315	60.32678305032443	60.011275263373							
8	6.1268768310546875	24.301193237304688	234.00027465820312	16.262815155634744	66.04122998108133	56.4343709652							
9	7.662704467734375	27.217071533203125	228.47325134277344	17.932959890925034	65.72435372765732	62.9051120141							
10	8.552886962890625	28.494110107421875	227.37380981445312	19.341130416081956	67.00446882307396	63.49255380386							
11	8.833999633789062	27.466079711914062	226.26898193359375	18.626948550716378	60.87409694811195	61.2713581307							
12	9.0579833984375	30.254608154296875	223.1147003173828	18.233629242079456	63.28346562579829	65.11573005615749							
13	14.945816040039062	34.213531494140625	234.57391357421875	39.34122136510123	84.94357974989511	51.42267054104							
14	7.060211181640625	27.897048950195312	229.39471435546875	17.162121892011537	69.65097776377233	63.04026446633							
15	10.655685424804688	39.731109619140625	219.35243225097656	20.119690204461484	77.4148260474518	68.22647469270							
16	10.243850708007812	43.31379699707031	218.8717498779297	19.069802016958466	83.0194048674478	68.3254722914131							
17	8.90098571773438	40.192169189453125	221.58868408203125	18.41913168067804	85.03437216312268	69.854416499734							
18	8.587738037109375	38.9947509765625	222.233642578125	17.914524798759242	83.12885936932739	69.08232319268383							
19	7.842315673828125	35.381011962890625	225.5125274658203	16.990432959961662	78.24199756312039	64.551431151577							
20	8.808202017106375	30.215501720724375	227.154238847772438	18.600772936307203	84.3084873883527	60.618005627076							

Fig. 4 Extraction Results

Following preprocessing, the SVM model was trained using 80% of the dataset, while the remaining 20% was reserved for testing. Feature normalization was performed using Standard Scaler to ensure consistent feature scaling. Model validation employed 5-fold cross-validation, and hyperparameter optimization was conducted using GridSearchCV. The evaluated parameters included the kernel type (linear and RBF), regularization parameter (C = 0.1, 1, 10, 100), and gamma values (0.001, 0.01, 0.1, 1, scale, auto). Model performance during tuning was assessed based on accuracy. The final trained model was saved as svm_herbal_model_fixed.pkl for deployment and further evaluation.

1. Classification Performance

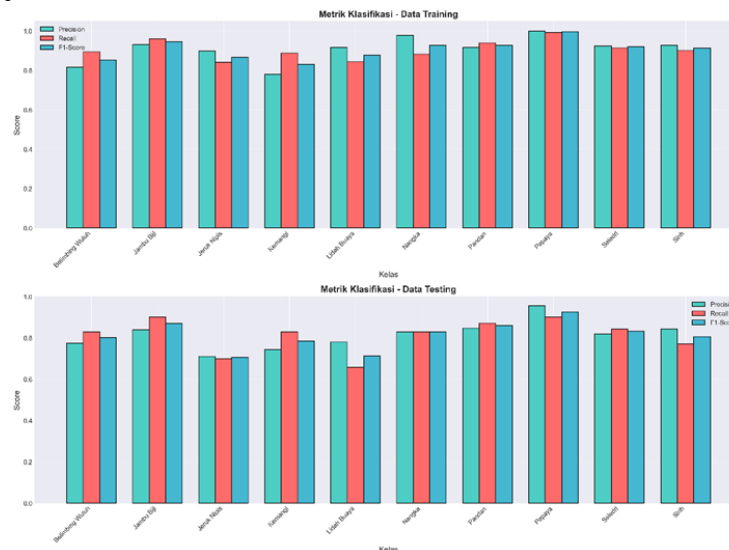


Fig. 5 Training Model Classification Metrics

On the training data, the SVM model demonstrated strong performance across all classes. Precision values ranged from 0.78 to 1.00, with the Papaya class achieving perfect precision (1.00), indicating no false positive predictions for that class. Recall values ranged from 0.84 to 1.00, showing that most samples were correctly identified. The F1-score, representing the harmonic mean of precision and recall, ranged

from 0.83 to 1.00, reflecting a well-balanced classification performance. Overall, these high metric values indicate that the model successfully captured discriminative patterns from the extracted leaf features.

When evaluated on the testing data, a moderate decrease in performance was observed, which is expected in generalization scenarios. Precision values ranged from 0.71 to 0.95, while recall values varied between 0.65 and 0.90. The most noticeable performance decline occurred in classes such as Lime and Aloe Vera, likely due to morphological similarities with other leaf types. The F1-score on testing data ranged from 0.70 to 0.93, indicating that the model maintained relatively stable performance on unseen data. The difference between training and testing results suggests that the model does not merely memorize the training data but demonstrates acceptable generalization capability.

2. Learning Curve Analysis.

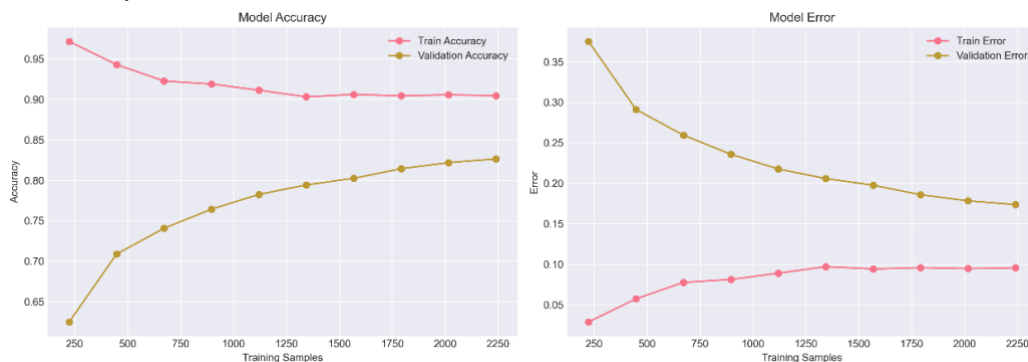


Fig. 6 SVM Model Performance Chart

The learning curve analysis further illustrates the model’s behavior as the training dataset size increases. With a small number of training samples, the training accuracy was very high (approximately 0.97), whereas validation accuracy was relatively low (around 0.62), indicating potential overfitting. As the number of training samples increased to approximately 2,250 instances, training accuracy gradually stabilized around 0.90, while validation accuracy improved to approximately 0.82. This pattern indicates that increasing the amount of training data helps the model learn more generalized patterns and reduces overfitting.

A similar trend was observed in the error curves. Initially, training error was low (around 0.03) and increased slightly to approximately 0.09 as more data were incorporated. Conversely, validation error decreased significantly from approximately 0.38 to 0.17. The relatively stable gap between training and validation error suggests that the model achieves a balanced trade-off between bias and variance, indicating stable and reliable classification performance.

4.2. Implementation

At this stage, the trained model was deployed into a graphical user interface (GUI)-based application for practical herbal leaf classification. When the program is executed, the system automatically loads the pre-trained machine learning model from the specified directory. Once successfully initialized, the system indicates that the model is ready for classification.

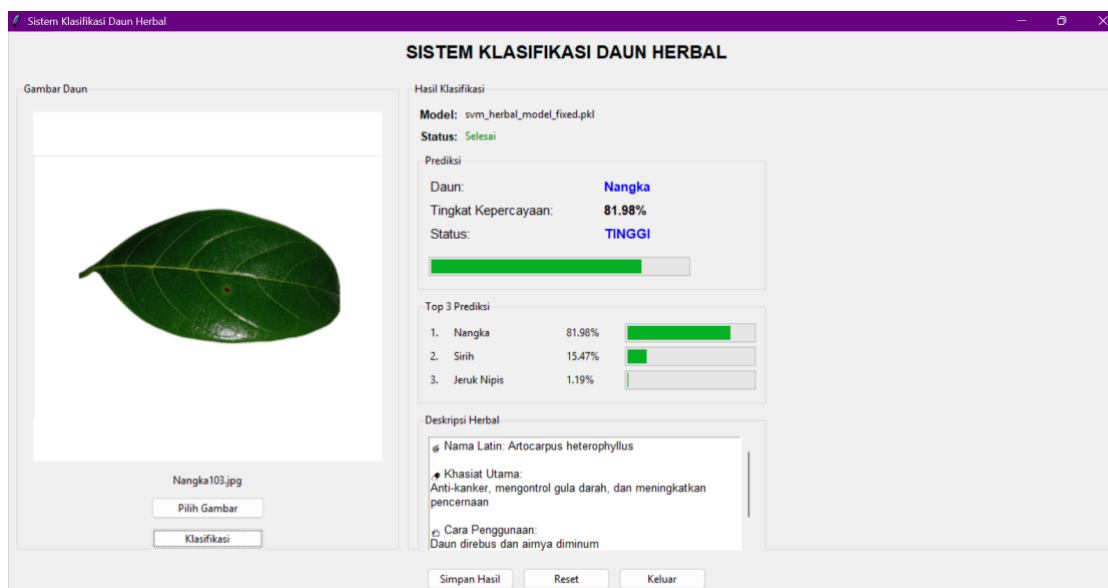


Fig. 7: GUI Classification of Herbal Leaves

Users begin the classification process by selecting an image file (JPG, PNG, or BMP format). The selected image is displayed within the interface for preview. Upon initiating the classification process, the system performs feature extraction by generating 13 numerical features

consisting of color features (6 HSV components), texture features (4 GLCM descriptors), and shape features (area, perimeter, and aspect ratio). Feature scaling is then applied before feeding the feature vector into the trained SVM model for prediction.

The predicted output, originally represented in numerical class format, is mapped into the corresponding herbal leaf name for user interpretability. The system presents the main prediction along with its confidence score (percentage). To enhance interpretability, confidence levels are categorized into qualitative levels (e.g., very high, high, moderate, and low). Additionally, the top three predicted classes are displayed to provide comparative insight into alternative classifications.

Beyond classification, the system provides a brief description of each identified herbal leaf, including its scientific name, primary medicinal benefits, usage method, and important notes. The entire classification process—from image selection to result display—typically requires approximately 1–2 seconds, demonstrating that the system operates efficiently while maintaining informative output suitable for both general users and professionals.

4.3. SVM Model Evaluation

Based on the confusion matrix obtained from model testing, the following confusion matrix represents the classification results for 10 herbal leaf classes.

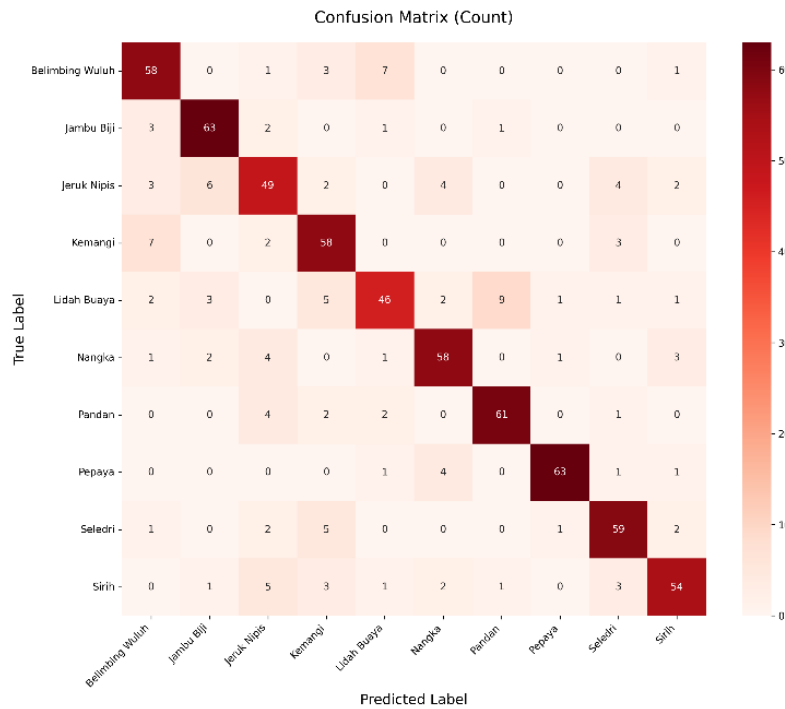


Fig. 8: Confusion Matrix

1. Performance Metrics

$$Accuracy = \frac{\sum_{i=1}^K TP_i}{N}$$

$$TP_{total} = 58 + 63 + 49 + 58 + 46 + 58 + 61 + 63 + 59 + 54 = 569$$

$$Accuracy = \frac{569}{700} = 0.8129 \approx 81.29\%$$

2. Complete Evaluation Metrics Per Class

Tabel 2: Metrics Per Class

Kelas	TP	FP	FN	Precision (%)	Recall (%)	F1-Score (%)
Belimbing Wuluh	58	12	17	82.86	77.33	80.00
Jambu Biji	63	8	7	88.73	90.00	89.36
Jeruk Nipis	49	21	21	70.00	70.00	70.00
Kemangi	58	12	12	82.86	82.86	82.86
Lidah Buaya	46	24	24	65.71	65.71	65.71
Nangka	58	12	12	82.86	82.86	82.86
Pandan	61	9	9	87.14	87.14	87.14
Pepaya	63	7	7	90.00	90.00	90.00
Seledri	59	11	11	84.29	84.29	84.29

	Sirih	54	16	16	77.14	77.14	77.14
Average Metrics:							
Precision:	$\frac{82.86 + 88.73 + 70.00 + 82.86 + 65.71 + 82.86 + 87.14 + 90.00 + 84.29 + 77.14}{10} = 81.16\%$						
Recall:	$\frac{77.33 + 90.00 + 70.00 + 82.86 + 65.71 + 82.86 + 87.14 + 90.00 + 84.29 + 77.14}{10} = 80.73\%$						
F1-Score	$\frac{80.00 + 89.36 + 70.00 + 82.86 + 65.71 + 82.86 + 87.14 + 90.00 + 84.29 + 77.14}{10} = 80.94\%$						

5. Conclusion

Based on the implementation and testing results of the herbal leaf classification system using the Support Vector Machine (SVM) method, several conclusions can be drawn as follows:

1. The herbal leaf classification system was successfully developed and is capable of classifying leaf images into 10 different herbal classes.
2. The Support Vector Machine (SVM) method provided reasonably good classification performance, achieving a testing accuracy of approximately 81.29%.
3. The confusion matrix indicates that several classes achieved high classification accuracy, while classes with visually similar characteristics tended to experience misclassification.
4. The developed system can be used as an initial supporting tool for identifying herbal leaf types based on image data.

References

- [1] S. Talib, S. Sudin, and M. Dzirkullah Suratin, "Penerapan Metode Support Vector Machine (Svm) Pada Klasifikasi Jenis Cengkeh Berdasarkan Fitur Tekstur Daun," *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, vol. 11, no. 1, pp. 26–34, 2024, doi: 10.30656/prosisko.v11i1.7911.
- [2] A. Arifin, J. Hendyli, and D. E. Herwindiati, "Klasifikasi Tanaman Obat Herbal Menggunakan Metode Support Vector Machine," *Computatio : Journal of Computer Science and Information Systems*, vol. 5, no. 1, p. 25, 2021, doi: 10.24912/computatio.v1i1.12811.
- [3] S. Harahap, "Alkaloid and Flavonoid Phytochemical Screening on Balakka Leaves (*Phyllanthus Emblica* L.)," *Formosa Journal of Science and Technology*, vol. 2, no. 8, pp. 2069–2082, 2023, doi: 10.55927/fjst.v2i8.5691.
- [4] W. S. Sari and C. A. Sari, "Klasifikasi Bunga Mawar Menggunakan Knn Dan Ekstraksi Fitur Glcm Dan Hsv," *Skanika*, vol. 5, no. 2, pp. 145–156, 2022, doi: 10.36080/skanika.v5i2.2951.
- [5] Muntiari Novita Ranti and Hanif Kharis Hudaiby, "Klasifikasi Penyakit Kanker Payudara Menggunakan Perbandingan Algoritma Machine Learning," *Jurnal Ilmu Komputer dan Teknologi*, vol. 3, no. 1, pp. 1–6, 2022, [Online]. Available: <http://creativecommons.org/licenses/by/4.0/%0Ahttp://ejournal.uhb.ac.id/index.php/IKOMTI>
- [6] E. Retnongsih and R. Pramudita, "Mengenal Machine Learning Dengan Teknik Supervised dan Unsupervised Learning Menggunakan Python," *BINA INSANI ICT JOURNAL*, vol. 7, no. 2, pp. 156–165, 2020, [Online]. Available: <https://www.python.org/>
- [7] P. R. Prayoga, P. Purnawansyah, T. Hasanuddin, and H. Darwis, "Klasifikasi Daun Herbal Menggunakan K-Nearest Neighbor dan Support Vector Machine dengan Fitur Fourier Descriptor," *Edumatic: Jurnal Pendidikan Informatika*, vol. 7, no. 1, pp. 160–168, 2023, doi: 10.29408/edumatic.v7i1.17521.
- [8] Milton Dapamudang, Fajar Hariadi, and Raynesta Mikaela Indri Malo, "Deteksi Pose Semaphore Berbasis Deep Learning Menggunakan Metode Multi-Layer Perceptron dan MediaPipe," *Jurnal ilmiah Sistem Informasi dan Ilmu Komputer*, vol. 5, no. 2, pp. 405–423, 2025, doi: 10.55606/juisik.v5i2.1386.
- [9] M. N. Fahmi, "Implementasi Mechine Learning menggunakan Python Library: Scikit-Learn (Supervised dan Unsupervised Learning)," *Sains Data Jurnal Studi Matematika dan Teknologi*, vol. 1, no. 2, pp. 87–96, Dec. 2023, doi: 10.52620/sainsdata.v1i2.31.
- [10] M. Meiriyama, S. Devella, and S. M. Adelfi, "Klasifikasi Daun Herbal Berdasarkan Fitur Bentuk dan Tekstur Menggunakan KNN," *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 3, pp. 2573–2584, 2022, doi: 10.35957/jatisi.v9i3.2974.