

Image recognition based on combination of RGB–HSV color and shape features using PCA and K-nearest neighbor

Hadiq¹, Solehatin², Much Aziz Muslim³, Dwika Ananda Agustina Pertiwi⁴

^{1,2} Information Technical, Sekolah Tinggi Ilmu Komputer PGRI Banyuwangi, Indonesia

³Department of Computer Science, Universitas Negeri Semarang, Indonesia

⁴Faculty Technology Management and Business, Universiti Tun Hussein Onn Malaysia, Malaysia

Article Info

Article history:

Received May 11, 2026

Revised June 2, 2026

Accepted June 15, 2026

Keywords:

Flower image recognition

PCA

HSV

RGB

Shape features

K-nearest neighbor

ABSTRACT

This study aims to accurately separate flower objects from complex backgrounds in artificial intelligence (AI)-based plant management systems. Previous studies have shown limitations in preprocessing techniques and often did not explicitly report classification accuracy. To address these issues, the proposed framework consists of eight stages. The process begins with image size standardization to 150×150 pixels, followed by low-pass filtering and image sharpening to enhance object boundaries. Segmentation is then performed sequentially using RGB and HSV color models to achieve more precise object separation. Subsequently, Sobel edge detection and thinning are applied to extract geometric features, such as distances between petal tips and flower perimeter measurements. The extracted features are optimized using Principal Component Analysis (PCA), which reduces the original ten attributes to four principal components, thereby eliminating data redundancy before classification using the Euclidean distance-based K-Nearest Neighbor (KNN) algorithm. Experimental results show that PCA preserves 86.7% of the original data variance, while the proposed system achieves an overall average classification accuracy of 88% at $k = 5$. Specifically, the recognition accuracies obtained for the four flower categories were 93.33% for Flower A, 86.67% for Flower B, 90.00% for Flower C, and 80.00% for Flower D. The main contribution of this research is the integration of intensive preprocessing techniques, the combination of RGB-HSV color features with geometric shape features, and PCA-based feature optimization, which collectively improve the stability and computational efficiency of KNN classification.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Hadiq

Information Technical,

Sekolah Tinggi Ilmu Komputer PGRI Banyuwangi,

Jalan Achmad Yani Nomor 80 Taman Baru Banyuwangi, Jawa Timur, Indonesia.

Email: hadiq@stikombanyuwangi.ac.id

<https://doi.org/10.52465/joscecx.v7i2.111>

1. INTRODUCTION

Automatic flower classification has become a crucial requirement in artificial intelligence (AI)-based plant management systems. However, accurately separating flower objects from complex backgrounds remains a significant challenge, highlighting the urgent need for the development of automated and accurate plant

management systems in the AI era [1]]. Previous studies have applied various approaches, including thresholding based on HSV saturation histograms, Otsu thresholding for shape feature extraction, and the integration of 3D scanning with RGB imagery [2],[3].

Although these studies have contributed to the advancement of this field, many of them exhibit several limitations. For instance, some studies do not explicitly report classification success rates, while others provide only technical evaluation metrics without clearly presenting classification accuracy. Furthermore, the use of raw images without adequate filtering and preprocessing often makes it difficult for the system to distinguish flowers from leaves [4]. Consequently, classification systems continue to face challenges in accurately differentiating flower and leaf structures when relying solely on raw images without sufficient filtering and segmentation processes.

To address these limitations, this study proposes a more integrated approach through intensive preprocessing stages, including linear filtering (low-pass filtering) to smooth image data and image sharpening techniques to enhance object boundaries [5], [6]. Unlike conventional methods that typically rely on a single color model, the proposed system employs a two-stage segmentation process using RGB and HSV color models sequentially to achieve more precise object separation [7].

In addition to standard color features, this research extracts comprehensive geometric shape features through Sobel edge detection and thinning processes. These features include unique attributes such as the distances between petal tips, the distances from petal tips to the pistil, and the flower perimeter, enabling a more detailed representation of flower morphology [5], [8], [9].

Color-based segmentation methods have been widely explored, particularly using Red Green Blue (RGB) and Hue Saturation Value (HSV) color spaces. These color models provide complementary information that can enhance segmentation performance when combined. Previous studies have demonstrated the effectiveness of thresholding-based segmentation across various domains, including medical and agricultural imaging [8]. In addition, common segmentation techniques such as edge-based, region-based, and pixel-based methods have been applied, with thresholding remaining a widely used approach due to its simplicity and computational efficiency [9].

The primary scientific contribution and novelty of this research lie in the application of Principal Component Analysis (PCA) as an optimization technique to address data redundancy arising from the combination of color features (RGB-HSV) and shape features [10], [11]. PCA is employed to reduce the original ten attributes into four principal components (PC1–PC4), which retain 86.7% of the total data variance. This dimensionality reduction process produces a more stable and efficient feature representation, thereby improving the effectiveness of classification using the K-Nearest Neighbor (KNN) algorithm [12]-[13].

Beyond segmentation, feature extraction plays a critical role in improving classification accuracy. Prior studies have utilized combinations of color and shape features—such as mean color values, standard deviation, area, perimeter, and compactness—to improve robustness against illumination variability [14]-[15]. Other approaches have incorporated dimensionality reduction techniques, such as Principal Component Analysis (PCA), and texture-based methods to enhance feature representation. Although these methods have shown promising results, several limitations remain, including incomplete preprocessing pipelines, unclear reporting of classification performance, and limited evaluation metrics [16]- [17].

Furthermore, some existing studies focus only on segmentation or classification without effectively integrating both processes into a unified framework. In other cases, the use of raw image data without adequate preprocessing reduces the reliability of extracted features and negatively impacts classification accuracy [18], [19]. These limitations highlight the need for a more comprehensive approach that integrates preprocessing, segmentation, feature extraction, and feature optimization in a coherent pipeline.

Based on these gaps, this study proposes an approach to improve flower image recognition accuracy by integrating RGB and HSV color features with shape descriptors, followed by feature optimization using Principal Component Analysis (PCA). The implementation of this method has been empirically proven to improve flower image recognition performance, achieving a significant average classification accuracy of 88% at $k = 5$. These results demonstrate the effectiveness of the proposed approach and indicate superior performance compared with previous studies, providing a more reliable solution for automated flower classification.

2. METHOD

The research process consists of eight stages. First, a flower image is provided as the input. The second stage involves image filtering to smooth the original image, thereby supporting subsequent processing stages

and improving image detection accuracy. The third stage is image segmentation, which is performed to separate the flower image (object) from the background. In this segmentation process, two color models are employed: Hue Saturation Value (HSV) and Red Green Blue (RGB).

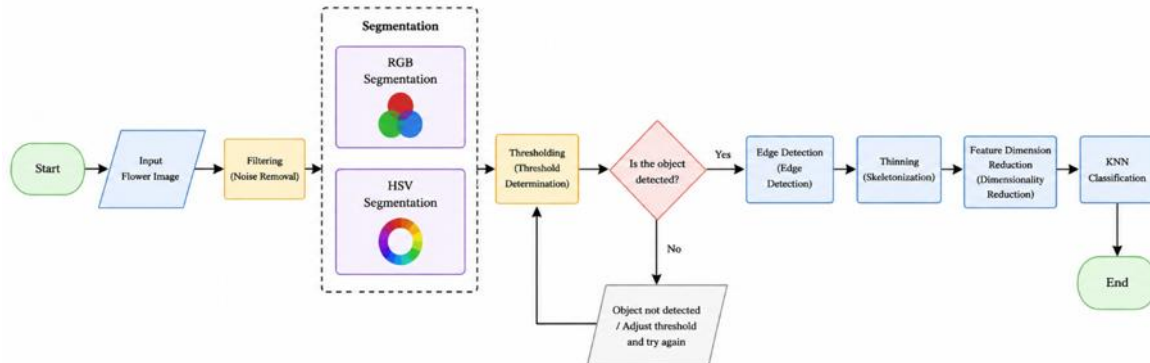


Figure 1. Flower image recognition stages

The fourth stage involves thresholding, which converts the image into a grayscale image based on a predetermined threshold value. The fifth stage is edge detection, which is used to highlight detailed regions of the image and enhance blurred details. The sixth stage is the thinning process, which identifies key points by detecting the longest and shortest points on the flower petals. The seventh stage involves dimensionality reduction using the Principal Component Analysis (PCA) method to optimize data representation before feature classification. In the final stage, classification is performed using the K-Nearest Neighbor (KNN) algorithm with ten criteria applied the flower features [20]. The research methodology stages are illustrated in Figure 1.

Input Flower Image

The first stage is determining the input image of the flower with supported compression formats in the study, which are limited to several types: JPG, BMP, and GIF. In addition to the image format, there are also several specific requirements for capturing flower images. First, the flower image must be taken in a perpendicular position so that the pistil is located at the center. Second, the image must not contain any cropped parts of the flower, and the left, right, top, and bottom boundaries of the image should align precisely with the edges of the flower petals. Third, only one flower should be present in the image. Fourth, the flower should be more visually prominent than the background color. Fifth, the number of petals must be clearly visible and not overlapping or clustered. Sixth, the optimal size of the input image is limited to a maximum resolution of 1300×1300 pixels and a minimum resolution of 80×80 pixels.

After acquiring the input image, a resizing process is performed to standardize the image used during the learning phase of the system to a size of 150×150 pixels. The purpose of setting this size is to accelerate the recognition process. This study utilizes a dataset that combines primary data collected directly by the researchers and stored in an internal database with secondary data obtained from images used in previous studies. Although the public availability of the primary dataset is not explicitly specified, the secondary data refer to widely validated benchmark datasets reported in the literature, such as the "17 Flowers" dataset, which has been used in studies involving 120 flower images.

For the experimental evaluation, a total of 50 test samples are distributed across four flower categories: Flower A (15 samples), Flower B (15 samples), Flower C (10 samples), and Flower D (10 samples). Each image sample undergoes a feature extraction process to generate ten feature attributes that combine color characteristics and geometric shape parameters. These features are subsequently processed using Principal Component Analysis (PCA) for dimensionality reduction prior to classification. The integration of primary and secondary datasets is conducted in a structured manner to ensure the validity, reliability, and objectivity of the experimental results, thereby providing a robust evaluation of the proposed method.

Filtering

The second stage involves a filtering process, which is carried out to smooth the original image so that it can better support the subsequent stages and improve the accuracy of image detection. The method used is linear filtering with a low-pass filter, which retains low-frequency components while removing high-

frequency components, followed by an image sharpening process to enhance the distinction between the foreground and background. In general, the computation is performed using a sliding window approach, where a filter in the form of an $m \times n$ matrix is shifted across the image until the center of the filter covers all pixels. The low-pass filter is based on averaging the pixel values with its 8 neighboring pixels. In this study, low-pass filtering is applied using a 3×3 window, as illustrated in Figure 2.

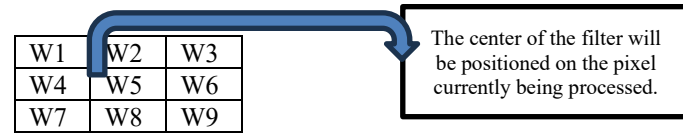


Figure 2. Windows 3x3

The low-pass filter is based on averaging the pixel values with its eight neighboring pixels. In this study, the low-pass filtering used applies a $1/9$ filtering formula, as shown in Equation (1).

$$g(m,n) = w_1f(m-1,n-1) + w_2f(m-1,n) + w_3f(m-1,n+1) + w_4f(m,n-1) + w_5f(m,n) + w_6f(m,n+1) + w_7f(m+1,n-1) + w_8f(m+1,n) + w_9f(m+1,n+1) \quad (1)$$

Segmentation

In the third stage, the segmentation process is carried out twice, using two color models: Red, Green, and Blue (RGB) and Hue, Saturation, and Value (HSV). The first segmentation stage uses the RGB model, which is performed based on values calculated from dividing the image (150×150) into 25 regions. From these regions, the RGB values are computed by taking the red, green, and blue components of the flower and calculating their average by summing the three color components and dividing by three. If the dominant color component of the flower produces a value greater than or equal to 128, then the segmentation process assigns the background a value of 255 (black), while the flower retains its original color. After obtaining the segmentation result from the RGB process, segmentation is further refined using the HSV model. In the HSV model, the hue value ranges from 0 to 360, while the value component represents the intensity of the color. Similar to saturation, the value component ranges from 0 to 1. If the value is 0, the color becomes black, whereas if the value is 1, the presence of black in the color is eliminated. Saturation represents the intensity or purity of the color, with a range from 0 to 1.

Thresholding

In the fourth stage, the thresholding process is performed to convert the image into a grayscale image using a predetermined threshold value. In this process, the flower is assigned a value of 0 (white), while the remaining areas are assigned black.

Edge Detection

In the fifth stage, an edge detection process is carried out to extract the edges of objects within the image. This process highlights the detailed parts of the image and enhances blurred regions. Blurring may occur due to errors or effects during the image acquisition process. A point (x,y) is considered an edge if it has a significant difference compared to its neighboring pixels. In this study, edge detection is applied to the segmentation results using the Sobel method on each RGB component with gray-level values ranging from 0 to 255. The values of each RGB component are summed, then rounded to integer values. If the rounded value exceeds 255, it is capped at 255.

Thinning

In the sixth stage, the thinning process is performed using skeleton binarization. The transformation of the skeleton into a binary form is carried out in a single image scan. If the analyzed pixel has at least one of its four neighbors (p_0, p_2, p_4, p_6) with a value lower than its gray value, or if all eight of its neighbors have values lower than its gray value, then the binary value is changed to 1; otherwise, it becomes 0.

Step 1: Data preparation is carried out by forming a set S using Equation (2):

$$(\Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M) S = \Gamma_1, \Gamma_2, \Gamma_3, \dots, \Gamma_M \quad (2)$$

Step 2: The mean value (Ψ) is computed using Equation (3):

$$\Psi = \frac{1}{M} \sum_{n=1}^M \Gamma_n \quad (3)$$

Step 3: The difference vector (Φ_i) from the training data (Γ_i) is calculated using Equation (4):

$$(\Psi)\Phi_i = \Gamma_i - \Psi \quad (4)$$

Step 4: The covariance matrix (C) is determined using Equation (5):

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (5)$$

Step 5: The eigenvalues (λ) and eigenvectors of the covariance matrix (C) are computed using Equation (6):

$$C x v_i = \lambda_i x v_i \quad (6)$$

Step 6: The eigenface values (μ) are determined using Equation (7):

$$\mu_i = \sum_{k=1}^M v_{ik} \Phi_k \quad i = 1, \dots, M \quad (7)$$

Feature Dimension Reduction

Feature dimension reduction was performed using Principal Component Analysis (PCA) to optimize data representation prior to the classification stage. This process began with the normalization of ten initial attributes, comprising color features (RGB and HSV) and geometric shape features, through mean-centering to ensure that each feature contributed equally to the analysis.

The covariance matrix analysis revealed the presence of data redundancy caused by strong correlations among features, particularly within the color components. Based on the computation of eigenvalues and eigenvectors, four principal components (PC1–PC4) were selected, collectively preserving the majority of the information contained in the original dataset. This selection was further supported by scree plot analysis, which indicated an elbow point at PC4, suggesting that the inclusion of additional components would provide only marginal informational gain.

The implementation of PCA effectively eliminated feature redundancy and minimized distortion in Euclidean distance calculations, thereby enhancing computational efficiency and improving the stability and classification performance of the K-Nearest Neighbor (KNN) algorithm.

KNN Classification

The final stage of the study is the classification of flower images using the K-Nearest Neighbor (KNN) algorithm. The classification is performed based on feature vectors obtained from PCA dimensionality reduction. The distance between the test image data and the training image data is calculated using the Euclidean distance, as shown in Equation (8).

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (8)$$

The K-Nearest Neighbor (KNN) algorithm was implemented as the final stage for flower image classification, utilizing the PCA-reduced feature vectors as input data. In this study, each flower image was represented by four principal components (PC1–PC4), which served as the feature space for classification. The similarity between test samples and training samples was determined using the Euclidean distance metric, which measures the proximity of feature vectors based on their underlying characteristics. By calculating the Euclidean distance between the PCA-transformed feature representations, the KNN classifier identified the nearest neighboring samples and assigned the class label according to the majority voting principle. This approach enables effective classification while benefiting from the reduced dimensionality and minimized feature redundancy achieved through PCA.

The class of the test image is determined based on the majority class of the K nearest neighbors with the smallest distance values. The value of K is determined through a testing process to obtain the best

classification performance. The system subsequently identifies a predefined number of nearest neighbors, denoted as K , based on the smallest Euclidean distance values. The class label of the test image is then assigned according to the majority class among these neighboring samples. This majority voting mechanism enables the classifier to determine the most likely category for each test instance.

The selection of an appropriate K value is critical to the overall performance of the classification system. A very small K value tends to make the classifier highly sensitive to noise and outliers, whereas a very large K value may blur the decision boundaries between classes and reduce classification accuracy. To determine the optimal parameter, experiments were conducted using several K values ($K = 1, 3, 5, 7, \text{ and } 9$). The experimental results indicated that $K = 5$ provided the best classification performance, achieving the highest overall accuracy and the most balanced classification results across all flower categories.

3. RESULTS AND DISCUSSIONS

The research methodology consists of several stages, including flower image acquisition as the input, image filtering, and image segmentation using the Hue Saturation Value (HSV) and Red Green Blue (RGB) color models. The subsequent stages involve thresholding, edge detection, and the thinning process. Dimensionality reduction is then performed using the Principal Component Analysis (PCA) method to optimize feature representation. Finally, classification is carried out using the K -Nearest Neighbor (KNN) algorithm to identify and categorize the flower images.

Input Flower Image

The first stage involves acquiring the input image, followed by a resizing process to standardize the image used during the system learning phase to a size of 150×150 pixels. The purpose of setting this size is to accelerate the recognition process. The results of the resizing process are shown in Figure 3 for images originally captured at a large size, and in Figure 4 for images captured at a smaller size. The input image data in this study consists of primary data collected directly and stored in a database. Meanwhile, the test images are obtained from both primary data and secondary data derived from images used in previous studies.



Original Image (650×650 pixels)



Resized Image (150×150 pixels)

Figure 3. Example of the resizing process from a large image size



Original Image (83×81 pixels)



Resized Image (150×150 pixels)

Figure 4. Result of the resizing process from a small image size

Filtering

In the second stage, the filtering process is carried out by scanning from the coordinate origin (0,0) along the x-axis to the y-axis, ending at the coordinate (150,150). From the scanning results of the RGB values, the minimum and maximum values are extracted for each color component: red, green, and blue. There are six criteria used based on these color components, namely: minimum red value, minimum green value, minimum blue value, maximum red value, maximum green value, and maximum blue value. The results of the RGB value extraction process are shown in Figure 5.

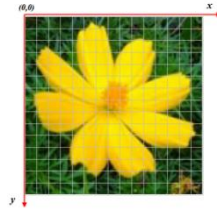


Figure 5. Result of the RGB value extraction process

From the available flower input data, the results of the low-pass filtering calculation are applied to each RGB component with gray-level values ranging from 0 to 255. The values of each RGB component are summed according to the low-pass filtering formula, then each resulting value is rounded to an integer. If the rounded value exceeds 255, it is capped at 255. The results are shown in Figure 6.

Matriks Input :						
125	125	200	130	210	150	150
125	125	200	130	210	150	150
185	185	150	130	150	100	100
180	180	155	135	145	105	105
180	180	150	130	135	105	105
185	185	150	140	125	110	110
135	135	145	130	135	145	145
135	135	145	130	135	145	145

Matriks Output :					
157.8	152.8	167.8	151.1	152.2	
165	154.4	156.1	139.4	135	
171.7	155	142.2	126.1	116.7	
171.7	156.1	140.6	125.6	116.1	
160.6	149.4	137.8	128.3	123.9	
150	143.9	137.2	132.8	132.8	

Filter :			
	1	1	1
0.11	1	1	1
	1	1	1

(a) Input Matrix (b) Output Matrix (c) Filter

Figure 6. Example of low-pass filter matrix calculation on RGB

The results of the matrix calculation using low-pass filtering can be seen in Equation (2), and the results of the filtering process are shown in Figure 7.

$$\begin{aligned}
 g(3,3) &= ((125 * 1) + ((125 * 1) + (200 * 1) + ((125 * 1) + ((125 * 1) + \dots \dots \dots 2 \\
 &\quad + ((200 * 1) + (185 * 1) + (185 * 1) + (200 * 1)) / 9 \\
 &= 1420 \\
 &= 157,8
 \end{aligned}$$



(a) Original Image (b) Filtered Image

Figure 7. Result of the Image Filtering Proses

Segmentation

In the third stage, the segmentation process is performed. The first segmentation is carried out using the Red, Green, and Blue (RGB) model, where the input image is taken from the result of the filtering process. The values are calculated based on dividing the 150 × 150 image into 25 regions, and the RGB values are computed for each region. For each region, the red, green, and blue components of the flower are averaged by summing the three color values and dividing by three. If the dominant flower color produces a value greater than or equal to 128, then the segmentation process assigns the background a value of 255 (black), while the flower retains its original color. The results of the RGB value calculations in the segmentation process are shown in Figure 8.



Figure 8. Result of the RGB image segmentation process

The results of the Red, Green, and Blue (RGB) segmentation are then followed by segmentation using Hue, Saturation, and Value (HSV). In the HSV segmentation stage, the input image is taken from the result of

the RGB segmentation. The values are calculated based on dividing the 150×150 image into 25 regions, and the HSV values are computed for each region. The Hue value ranges from 0 to 360, while the Value component in the HSV model ranges from 0 to 1. If the Value is 0, the color becomes black; whereas if the Value is 1, the presence of black in the color is eliminated. The Saturation component also ranges from 0 to 1. The results of the HSV value calculations in the segmentation process are shown in Figure 9.



Figure 9. Result of the HSV image segmentation process

Thresholding

In the fourth stage, the thresholding process is performed based on the segmentation results by dividing the 150×150 image into 25 regions. In this process, the flower is assigned a value of 0 (white). The results of the RGB value calculations in the segmentation process can be seen in Figure 10.



Figure 10. Result of the thresholding process image

Edge Detection

In the fifth stage, edge detection is performed. In this study, edge detection is applied to the results of the segmentation process using the Sobel method on each RGB component with gray-level values ranging from 0 to 255. The values of each RGB component are summed, then rounded to integer values. If the rounded value exceeds 255, it is capped at 255. The results of the matrix calculations for edge detection using the Sobel method are shown in Equation (3), and the results of the edge detection process are presented in Figure 11.

$$G(x) = ((125 * -1) + ((125 * -2) + (200 * -1) + (185 * 1) + (185 * 2) + (150 * 1))) = 130$$

$$G(y) = ((125 * -1) + ((125 * -2) + (185 * -1) + (200 * 1) + (200 * 2) + (150 * 1))) = 190$$

$$\text{Sobel Result } (x) + g(y) = 130 + 190 \dots\dots\dots 3$$

$$= 320$$

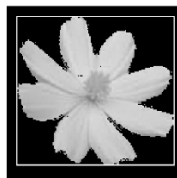


Figure 11. Result of the edge detection process image

Thinning

Next, the determination of the longest and shortest points on the flower is carried out starting from the coordinate $(x,y)(x, y)(x,y)$, specifically at $(0,75)(0, 75)(0,75)$, which represents the midpoint of the coordinate axis in a 150×150 pixel image. The tracing process begins when the first point is found and continues to subsequent points in a clockwise direction based on the 8-connected component approach. The results of this process are illustrated as white lines derived from the edge detection output. The longest point is determined when no longer a longer point can be found during the tracing process, indicating that the traced points are becoming shorter. Conversely, the shortest point is identified when no shorter point can be found and the traced points tend to become longer. The matrix calculation is the same as in the edge detection process. The results of the thinning process are shown in Figure 12.

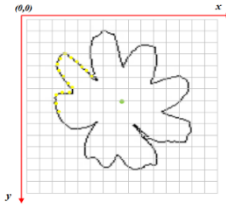


Figure 12. Result of determining the longest and nearest points

Feature Dimension Reduction

After completing all preprocessing, segmentation, and feature extraction stages, a flower image feature vector is obtained consisting of 10 (ten) attributes: Red, Green, Blue, Hue, Saturation, Value, the distance between leaf tips, the distance from leaf tips to the pistil, the number of petals, and the flower perimeter. This feature vector is then reduced in dimensionality using the Principal Component Analysis (PCA) method to eliminate redundancy among features and improve the efficiency of the classification process.

All training data are organized into a feature matrix X of size $M \times N$, where M represents the number of flower image samples and $N=10$ is the number of initial features. Each row of the matrix represents one flower image, while each column represents a single feature. Before performing dimensionality reduction, the feature data are normalized by subtracting the mean value of each feature to ensure that each feature contributes equally in the PCA process.

The mean value of the features is calculated using Equation (2), and each feature vector is then centered by subtracting its mean as shown in Equation (3). The covariance matrix C is subsequently computed using Equation (4). The results of the covariance matrix calculation indicate a relatively high correlation among several color features, particularly between the RGB and HSV components. This suggests the presence of feature redundancy that can be reduced using PCA.

The eigenvalues and eigenvectors are obtained from the covariance matrix using Equation (5). The eigenvectors are sorted in descending order based on their eigenvalues. The largest eigenvalues represent the principal components that capture the greatest variance in the flower image data. Table 1 and Figure 13 present the results of the PCA eigenvalue calculations and their distribution.

Table 1. Eigenvalues and variance from PCA results

Principal Component	Eigenvalue	Variance (%)	Cumulative Variance (%)
PC1	4,18	41,8	41,8
PC2	2,36	23,6	65,4
PC3	1,42	14,2	79,6
PC4	0,71	7,1	86,7
PC5	0,54	5,4	92,1
PC6	0,36	3,6	95,7
PC7	0,23	2,3	98,0
PC8	0,12	1,2	99,2
PC9	0,06	0,6	99,8
PC10	0,02	0,2	100,0

Based on Table 1, it can be observed that the first to the fourth principal components (PC1–PC4) have dominant variance contributions compared to the other components. The cumulative variance explained by the first four principal components reaches 86.7%, while the remaining components contribute relatively small amounts of variance. Figure 13 illustrates the distribution of eigenvalues in the form of a scree plot. It can be seen that there is a significant decrease in eigenvalues from PC1 to PC3, followed by a gradual flattening after PC4. This pattern indicates the presence of an elbow point at PC4, suggesting that adding more components beyond PC4 does not provide a significant increase in explained variance.

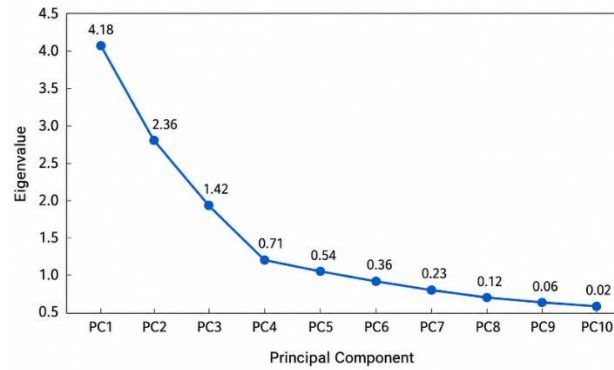


Figure 13. Distribution of PCA eigenvalues

Based on the analysis of cumulative variance in Table 1 and the eigenvalue decreasing pattern in Figure 13, the first four principal components (PC1–PC4) were selected as the result of feature dimensionality reduction. These four principal components are able to preserve 86.7% of the variance in the flower image data, thus reducing the feature dimension from the original 10 features to 4 main features without significant information loss. The resulting reduced feature vector is then used in the classification stage using the K-Nearest Neighbor (KNN) algorithm.

Although eigenvalue analysis and the scree plot are used to determine the number of retained principal components, each principal component generated by PCA is essentially a linear combination of all original features. Therefore, to understand the contribution of each original feature to the selected principal components, further analysis is required through PCA loading values. This analysis aims to identify the most dominant features forming each retained principal component, so that the interpretation of dimensionality reduction results is not only numerical but also carries semantic meaning regarding the characteristics of flower images. Table 2 presents the feature loadings of the four principal components.

Table 2. Feature Loadings on the four principal components (PC1–PC4)

Fitur	PC1	PC2	PC3	PC4
Red	0,82	0,21	0,08	0,04
Green	0,79	0,25	0,10	0,06
Blue	0,76	0,18	0,12	0,05
Hue	0,74	0,32	0,09	0,07
Saturation	0,71	0,28	0,11	0,09
Value	0,69	0,30	0,14	0,08
Distance between leaf tips	0,12	0,81	0,18	0,09
Distance from leaf tips to pistil	0,15	0,78	0,22	0,11
Number of petals	0,09	0,24	0,83	0,19
Flower perimeter	0,11	0,27	0,79	0,21

In Table 2, it can be observed that the first principal component (PC1) is dominated by color features in both RGB and HSV color spaces, with the highest loading values on Red, Green, Blue, Hue, Saturation, and Value. This indicates that color variation is the main factor distinguishing flower images in the dataset used in this study. The second principal component (PC2) has the largest contribution from the distance between leaf tips and the distance from leaf tips to the pistil, representing the geometric characteristics and structural shape of the flower. Meanwhile, the third principal component (PC3) is dominated by the number of petals and flower perimeter, which describe the complexity and size of the flower object.

The fourth principal component (PC4) has a relatively small contribution and serves as a complementary component capturing variance not represented by the previous three components. These results indicate that the combination of color features (RGB–HSV) and shape features provides complementary contributions in the PCA-based dimensionality reduction process, ensuring that the reduced feature set still preserves important information relevant for the classification stage. Based on the dimensionality reduction results using Principal Component Analysis (PCA), each flower image is represented by four principal components (PC1–PC4), which retain 86.7% of the total data variance.

KNN Classification

The resulting reduced feature vector is then used as input for the classification process using the K-Nearest Neighbor (KNN) algorithm with Euclidean distance calculation as shown in Equation (7). The experimental results show that the use of PCA-based features significantly improves flower image recognition performance compared to using the original high-dimensional features. Dimensionality reduction not only reduces computational complexity but also minimizes redundancy and high correlation among RGB and HSV color features, resulting in a more stable nearest-neighbor search process.

Furthermore, the integration of color and shape features in the PCA feature space provides better class separation, especially for flower types with similar colors but different morphological structures. This demonstrates that PCA successfully preserves important discriminative information required by KNN for accurate classification. Overall, these results confirm that the combination of PCA-based feature reduction and distance-based KNN classification is an effective approach for flower image recognition using RGB–HSV color features and shape characteristics.

Table 3. KNN accuracy results for various k values

k Value	Test Data Count	Correctly Classified Data	Accuracy (%)
1	50	40	80,00
3	50	43	86,00
5	50	44	88,00
7	50	42	84,00
9	50	41	82,00

Classification accuracy is evaluated by measuring the proportion of correctly classified test images against the total number of test images. The accuracy formula used in this study is defined in Equation (8):

$$\text{Accuracy} = (\text{Number of Correctly Classified Data} / \text{Total Data}) \times 100\% \quad (8)$$

The accuracy value in Table 3 was calculated using Equation (8) for each variation of k. For example, at k=5, from 50 test data items, 44 were correctly classified, yielding an accuracy of $(44/50) \times 100\% = 88\%$.

In the initial stage of classification evaluation, testing was conducted on several variations of the parameter k in the K-Nearest Neighbor (KNN) algorithm to determine the optimal value of k. This testing aims to examine the influence of the number of nearest neighbors on the accuracy level of flower image recognition. All test data were used simultaneously in this stage, with the input feature vector consisting of four principal components (PC1–PC4) obtained from Principal Component Analysis (PCA).

The results of the k-value variation testing are presented in Table 3. Based on the table, it can be observed that the value of k significantly affects classification performance. The value k=5 produces the highest accuracy compared to other k values. Therefore, k=5 is selected as the optimal parameter and used in the subsequent testing process.

After the optimal value of the parameter k has been determined, the next evaluation stage is conducted by analyzing the image recognition performance based on each flower class. In this stage, the classification process is carried out using the KNN algorithm with k=5, while the input features remain the four principal components obtained from PCA dimensionality reduction.

Table. KNN Recognition Results Based on Flower Type (k = 5)

Flower Type	Test Data Count	Correctly Classified Data	Accuracy (%)
Flower A	15	14	93,33
Flower B	15	13	86,67
Flower C	10	9	90,00
Flower D	10	8	80,00
Average	50	44	88,00

This per-class evaluation aims to determine the system's ability to recognize each type of flower specifically and to identify differences in accuracy levels among classes. The results of flower image recognition based on flower types are presented in Table 4.

Discussion

The results of the study show that the combination of RGB–HSV color feature extraction and shape features, which are then reduced using Principal Component Analysis (PCA), is able to produce an effective feature representation for flower image recognition. The dimensionality reduction from ten initial features to four principal components, which preserve 86.7% of the data variance, proves that most of the discriminative information in flower images can be represented in a lower-dimensional feature space. This condition

directly contributes to improving the stability and efficiency of the classification process using the K-Nearest Neighbor (KNN) algorithm.

The application of PCA in this study plays an important role in reducing redundancy among features, particularly between RGB and HSV components, which are naturally highly correlated. Without dimensionality reduction, these correlated features could distort Euclidean distance calculations in KNN, thereby reducing classification accuracy. By using PCA-based features, the distance between data points in the feature space becomes more representative of the visual differences between flower classes, ultimately improving KNN's ability to distinguish between categories.

The experimental results of varying the parameter k in KNN show that selecting an appropriate k value significantly affects image recognition performance. The value $k=5$ yields the highest accuracy because it balances sensitivity to noise and model generalization. A smaller k tends to make the system more sensitive to local variations and data errors, while a larger k can blur class boundaries due to the dominance of neighbors from other classes. This finding is consistent with the general characteristics of KNN in a low-dimensional feature space.

The analysis of recognition results based on flower types shows variations in accuracy across classes. Flower types with more consistent color and shape characteristics achieve higher recognition rates, while those with greater intra-class variation or visual similarity to other classes tend to have lower accuracy. This indicates that although RGB–HSV color features significantly contribute to class separation, shape features play an important role in reducing ambiguity for flowers with similar colors. Therefore, the combination of color and shape features is proven to be more effective than using either feature alone.

Although the results demonstrate good recognition performance with an average accuracy of 88%, limitations in dataset size and the number of flower classes still affect system generalization. The relatively small number of test samples in each class makes the accuracy sensitive to individual classification errors. Therefore, the results of this study primarily reflect the effectiveness of the proposed method rather than its capability to handle larger and more diverse datasets.

To position the contribution of this study within the existing body of research, a synthesis of related works is presented in Table 5. This comparison covers methods, features, and accuracy values reported in previous studies on flower or plant image recognition tasks.

Table 5. Synthesis of Related Research on Flower/Plant Image Recognition

Author (Year)	Method	Features	Dataset	Accuracy (%)
Rosyani & Hariansyah (2020)	Otsu + Naïve Bayes	Color histogram (RGB), shape features	Local flower dataset	82.00
Abdullah et al. (2023)	Nearest Mean Classifier	Shape and color combined features	Fruit/plant images	85.50
Inthiyaz et al. (2018)	PCA + Level Sets	PCA-fused color covariance and Gabor texture	Oxford Flower dataset	84.30
Nuraini et al. (2023)	KNN + PCA	Image-based features (fish species)	Fish image dataset	86.40
Zhang et al. (2021)	Attention mechanism + Multi-loss network	Deep CNN features	Flower image dataset	91.20
This Study (2025)	RGB-HSV + Shape + PCA + KNN	RGB, HSV, shape (10 features \rightarrow 4 PCs)	Primary flower dataset	88.00

Based on the synthesis in Table 5, the proposed method achieves competitive accuracy (88%) compared to several related studies that employ traditional handcrafted features and dimensionality reduction. Rosyani and Hariansyah [13] reported 82% accuracy using Otsu thresholding and Naïve Bayes without dimensionality reduction, while Abdullah et al. [15] achieved 85.5% using a nearest mean classifier with combined color and shape features. Inthiyaz et al. [19] applied PCA with Gabor texture features on the Oxford dataset, achieving 84.3%, whereas Nuraini et al. [20] combined KNN and PCA for fish species classification and reached 86.4% accuracy. The deep learning approach by Zhang et al. [4] achieved the highest accuracy of 91.2%, but at the cost of greater computational complexity and larger dataset requirements. These comparisons demonstrate that the proposed method, which integrates RGB–HSV color features with shape descriptors and applies PCA-based dimensionality reduction before KNN classification, offers a balanced trade-off between accuracy, computational efficiency, and dataset size. This confirms its

suitability as a lightweight yet effective solution for flower image recognition under constrained data conditions.

Overall, this study shows that a flower image recognition approach based on a combination of RGB–HSV color and shape features, reduced using PCA and classified using KNN, is a simple yet effective solution. This approach is suitable for image recognition systems with limited data and relatively low computational requirements. Future work can be extended by increasing the dataset size and number of flower classes, incorporating advanced texture features, and comparing KNN performance with other classification methods to improve system robustness.

4. CONCLUSION

This study successfully implemented a flower image recognition method based on a combination of RGB–HSV color features and shape features using dimensionality reduction through Principal Component Analysis (PCA) and classification using K-Nearest Neighbor (KNN). The dimensionality reduction results show that the four principal components generated by PCA are able to represent 86.7% of the data variance, thereby effectively reducing feature redundancy without losing important information.

The classification experiments demonstrate that the use of PCA-based features improves the stability and accuracy of flower image recognition, with the highest accuracy achieved using the KNN parameter $k=5$. Per-class evaluation on four flower types shows that the system is capable of recognizing flower images with good performance, achieving an average accuracy of 88%, although performance variations among classes still exist due to differences in visual characteristics and limitations in the amount of data.

Overall, this study proves that the combination of PCA and KNN is an effective and efficient approach for flower image recognition based on color and shape features, particularly under conditions of limited data and computational resources. Future work can focus on increasing the dataset size, exploring texture features, and applying other classification methods to improve system performance and generalization.

Although the proposed approach achieved promising results with an average classification accuracy of 88%, several limitations may affect the generalizability of the system, particularly regarding the relatively small dataset size and the limited number of flower classes included in the study. The small number of test samples makes the reported accuracy highly sensitive to individual classification errors. Furthermore, the system's performance still exhibits variations across different flower categories due to differences in visual characteristics and ambiguities among certain flower types.

In addition, the proposed method imposes relatively strict image acquisition requirements. The flower images must be captured from a perpendicular viewpoint with the pistil positioned near the center of the image, while the petals should remain fully visible without being cropped or overlapping. These constraints may limit the applicability of the system in real-world environments where image acquisition conditions are often uncontrolled. Therefore, future research should focus on expanding the dataset with a larger number of samples and flower species, incorporating advanced texture-based features to improve feature representation, and conducting comparative evaluations with alternative classification methods. Such improvements are expected to enhance the robustness, scalability, and practical applicability of the proposed flower classification system across a broader range of conditions and use cases.

REFERENCES

- [1] H. Hadiq, S. Solehatin, D. Djuniarto, M. A. Muslim, and S. N. Salahudin, "Comparison of the suitability of the otsu method thresholding and multilevel thresholding for flower image segmentation," *J. Soft Comput. Explor.*, vol. 4, no. 4, pp. 242–249, Dec. 2023, doi: 10.52465/josce.v4i4.266.
- [2] R. Kosasih, "Automatic Segmentation of Abdominal Aortic Aneurism (AAA) By Using Active Contour Models," *Sci. J. Informatics*, vol. 7, no. 1, pp. 66–74, 2020, doi: 10.15294/sji.v7i1.23625.
- [3] D. Aydin and A. Uğur, "Extraction of flower regions in color images using ant colony optimization," *Procedia Comput. Sci.*, vol. 3, pp. 530–536, 2011, doi: 10.1016/j.procs.2010.12.088.
- [4] M. Zhang, H. Su, and J. Wen, "Classification of flower image based on attention mechanism and multi-loss attention network," *Comput. Commun.*, vol. 179, no. August, pp. 307–317, 2021, doi: 10.1016/j.comcom.2021.09.001.
- [5] M. Yu, Y. Sun, R. Ma, H. Mu, D. Liu, and Z. Li, "YNU Flower: A market-oriented benchmark dataset for flower classification," *Sci. Hortic. (Amsterdam)*, vol. 338, no. July, p. 113510, 2024, doi: 10.1016/j.scienta.2024.113510.
- [6] S. S. Sitanggang, Y. Yuhandri, and Adil Setiawan, "Image Transformation With Lung Image Thresholding and Segmentation Method," *J. RESTI (Rekayasa Sist. dan Teknol. Informatika)*, vol. 7, no. 2, pp. 278–285, Mar. 2023, doi: 10.29207/resti.v7i2.4321.
- [7] H. Biabi, S. Abdanan Mehdizadeh, and M. Salehi Salmi, "Design and implementation of a smart system for water management of liliium flower using image processing," *Comput. Electron. Agric.*, vol. 160, no. December 2018, pp. 131–143, 2019, doi: 10.1016/j.compag.2019.03.019.
- [8] Y. Shang, M. Geng, H. Fang, M. Cai, J. Wang, and H. Song, "Using unmanned aerial vehicle acquired RGB images and Density-Cluster-Count model for tree-level apple flower quantification," *Comput. Electron. Agric.*, vol. 226, no. January, p. 109389, 2024, doi: 10.1016/j.compag.2024.109389.
- [9] N. Huda, A. Y. Dewi, and A. Mahiruna, "Plasmodium falciparum Identification Using Otsu Thresholding Segmentation Method Based on Microscopic Blood Image," *Sci. J. Informatics*, vol. 10, no. 4, 2023, doi: https://doi.org/10.15294/sji.v10i4.47924.
- [10] B. Ambrus et al., "Field-grown tomato yield estimation using point cloud segmentation with 3D shaping and RGB pictures from a field robot and digital single lens reflex cameras," *Heliyon*, vol. 10, no. 20, 2024, doi: 10.1016/j.heliyon.2024.e37997.
- [11] T. Mustaqim, K. Umam, and M. A. Muslim, "Twitter text mining for sentiment analysis on government's response to forest fires

- with vader lexicon polarity detection and k-nearest neighbor algorithm,” *J. Phys. Conf. Ser.*, vol. 1567, no. 3, 2020, doi: 10.1088/1742-6596/1567/3/032024.
- [12] M. AlRifae, S. Almanasra, A. Hnaif, A. Althunibat, M. Abdallah, and T. Alrawashdeh, “Adaptive Segmentation for Unconstrained Iris Recognition,” *Comput. Mater. Contin.*, vol. 78, no. 2, pp. 1591–1609, 2024, doi: 10.32604/cmc.2023.043520.
- [13] X. Yao, Q. Yi, F. Wang, T. Xu, J. Zheng, and Z. Shi, “Estimating rice flower intensity using flower spectral information from unmanned aerial vehicle (UAV) hyperspectral images,” *Int. J. Appl. Earth Obs. Geoinf.*, vol. 122, no. June, p. 103415, 2023, doi: 10.1016/j.jag.2023.103415.
- [14] A. Abdullah, Agus Harjoko, and Othman Mahmud, “Classification of Fruits Based on Shape and Color using Combined Nearest Mean Classifiers,” *J. RESTI (Rekayasa Sist. dan Teknol. Informatika)*, vol. 7, no. 1, pp. 51–57, 2023, doi: 10.29207/resti.v7i1.4693.
- [15] F. Ismawan, “288089012,” vol. 5, no. 1, pp. 26–30, 2015.
- [16] S. Inthiyaz, B. T. P. Madhav, and P. V. V. Kishore, “Flower image segmentation with PCA fused colored covariance and gabor texture features based level sets,” *Ain Shams Eng. J.*, vol. 9, no. 4, pp. 3277–3291, 2018, doi: 10.1016/j.asej.2017.12.007.
- [17] K. Anwar, “Image Convolution to Obtain Color ROI after Segmentation Process with Fuzzy Cmeans,” *J. RESTI (Rekayasa Sist. dan Teknol. Informatika)*, vol. 7, no. 2, pp. 376–380, 2023, doi: 10.29207/resti.v7i2.4874.
- [18] O. Somantri et al., “Performance Comparison and Optimization of Text Document Classification using k-NN and Naïve Bayes Classification Techniques,” *Procedia Comput. Sci.*, vol. 4, no. 1, pp. 107–112, 2023, doi: 10.1016/j.procs.2017.10.017.
- [19] K. K. Teo, N. F. B. M. Noor, S. Palaiahnakote, and M. N. Bin Ayub, “An efficientnet-based model for classification of oil palm, coconut and banana trees in drone images,” *Smart Agric. Technol.*, vol. 10, no. December 2024, p. 100748, 2025, doi: 10.1016/j.atech.2024.100748.
- [20] M. Alom, M. Y. Ali, M. T. Islam, A. H. Uddin, and W. Rahman, “Species classification of brassica napus based on flowers, leaves, and packets using deep neural networks,” *J. Agric. Food Res.*, vol. 14, p. 100658, Dec. 2023, doi: 10.1016/j.jafr.2023.100658.