Received: November 13, 2023. Revised: January 9, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Enhanced Support Vector Machine Based on Grey Wolf Optimizer for Fruits Image Classification using MPEG-7 Color and Texture Features Fusion

Joko Siswantoro[1]*

[1]*Department of Informatics Engineering, University of Surabaya, Jl. Kalirungkut, Surabaya, 60293, Indonesia*
* Corresponding author's Email: joko_siswantoro@staff.ubaya.ac.id

Abstract: Fruit classification from images plays a pivotal role in diverse domains. Despite numerous efforts to tackle this challenge, it remains complex due to the diversity of fruit and applications. This study presents an enhanced support vector machine (SVM) based on grey wolf optimizer (GWO) for fruit image classification. GWO is used to optimize the hyperparameters of SVM and low variance feature selection threshold. The utilization of MPEG-7 visual descriptors negates the need for segmentation. The results showcase exceptional classification accuracy across Ubaya-IFDS3000, Ubaya-IFDS5000, and Supermarket produce datasets, with standout features achieving up to 99.21%, 98,28%, and 99.85% accuracies, respectively. Notably, the proposed method consistently outperforms SVM optimized with the other optimization algorithms. Further, it excels in classification accuracy when compared to previous state-of-the-art methods. This study emphasizes the importance of hyperparameter optimization using GWO and its effectiveness in fruit image classification.

Keywords: Fruit classification, Support vector machine, Hyperparameter optimization, Grey wolf optimizer.

1. Introduction

Fruit classification from images holds significant importance in various fields, such as agriculture, food industry, retail, and dietary recommendations [1, 2]. The classification of fruits from images has been the subject of extensive research by scholars. However, it remains a significant challenge due to the sheer diversity of fruit types available in the market and its wide array of application domains. With many fruit varieties and varying visual characteristics, this field of study continues to evolve, aiming to develop robust and adaptable classification systems that cater to the diverse needs of different industries and contexts [3]. Fruit classification from images is an inexpensive method compared to other approaches [4] and is an alternative to the traditional classification method [5]. Fruit classification from images can be broadly categorized into two main methods: traditional machine learning and deep learning. In the traditional machine learning approach, some features are extracted from fruit images and then used to train the machine learning model [6].

Numerous studies have employed traditional machine learning methods to classify fruits from images [1, 6-17]. This approach usually involves segmentation and feature extraction steps in the classification pipeline. Segmentation is the process of separating objects from their backgrounds, enabling subsequent analysis and recognition [18]. Various segmentation methods have been employed for fruit classification from images in some studies, including background subtraction [8, 10, 12, 16], split-and-merge [13-15, 17], GrabCut [7], and automatics thresholding [1]. Color, texture, and shape were the most used features for fruit classification from images in the traditional machine learning approach. Several studies have implemented feature and classifier fusion strategies to enhance classification accuracy, as reported in [1, 6, 9-11, 16]. In previous research, support vector machines (SVM) and artificial neural networks (ANN) have been the predominant classifiers for fruit classification from images that

Received: November 13, 2023. Revised: January 9, 2024.

Nomenclature	
$\boldsymbol{F}$	Features fusion
CS	Color structure
SC	Scalable color
CL	Color layout
HT	Homogeneous texture
ET	Edge histogram
$\mathbf{x}, \mathbf{x}_i$	Feature vector
y_i	Expected output
$y(\mathbf{x})$	Predicted output
$\mathbf{w}, b$	SVM parameters
C	Regularization parameter
$K(\mathbf{x}_i, \mathbf{x}_j)$	Kernel function
$\mathbf{P}(i), \mathbf{P}_j(i)$	The position vector of grey wolf
$\widehat{\mathbf{P}}_j$	The estimation of $\mathbf{P}$
$\mathbf{P}_p(i)$	The position vector of prey
$\mathbf{A}, \mathbf{A}_j, \mathbf{C}, \mathbf{C}_j$	Coefficient vectors
a	Scalar between 0 and 2
$\mathbf{r}_1, \mathbf{r}_2$	Random vectors
$\mathbf{D}, \mathbf{D}_j$	The distance between the wolf and the prey
$f(\mathbf{P})$	Objective function
T	The threshold value of LVFS
acc_t	Classification accuracy
C_t	The number of correctly classified images
T_t	The number of images in the dataset
α_i	Dual coefficient
ξ_i	Slack variables
γ	Kernel coefficient

achieved the best performance, such as in [1, 12-17], [19-22]. Derivative-free optimization techniques, such as fitness-scaled chaotic artificial bee colony (FSCABC) [19], biogeography-based optimization (BBO) [20], and Kalman filter (KF) [21], have also been applied to train classifiers for fruit recognition from images to enhance performance, as reported in [1, 14, 15].

In the case of fruit classification from images using deep learning, previous studies can be categorized into two main approaches: building convolutional neural networks (CNN) models from scratch [22-25] and utilizing pre-trained CNN models with transfer learning strategies [24, 26-32]. Some previous studies employed several pre-trained CNN model architectures to classify fruits from images. These architectures included MobileNetV2 [27, 28], VGG-16 [24], AlexNet [26], DenseNet [29-31], ResNet [30, 31], NASNet [30], EfficientNet [30], Inception V3 [31], and MangoNet [32]. Many researchers have employed a technique known as image augmentation to achieve a well-performing CNN model during the training phase. Image augmentation involves creating variations of the training data by applying transformations like rotations, flips, and zooms to increase the diversity of training data [23, 24, 26].

In addition to selecting the appropriate classification model, the choice and tuning of hyperparameters play a crucial role in determining the model's performance. Hyperparameters are settings or configurations not learned from the data. Still, they are essential for the model's behavior, such as learning rates, batch sizes, or the number of layers in a neural network. Optimizing these hyperparameters is crucial to obtain the best model performance [33, 34]. However, in the existing literature, hyperparameter tuning has predominantly been explored in studies focusing on fruit classification from images through deep learning approach, as reported in [29-32]. Several optimization algorithms have been employed for hyperparameter tuning in deep learning models for fruit image classification. Noteworthy among them are the Aquila Optimization Algorithm (AOA) [29], Tunicate Swarm Algorithm (TSA) [31], Harris Hawks Optimization (HHO) [32], and Bayesian Optimization [30]. On the other hand, in machine learning approach, researchers have primarily focused on selecting models, feature engineering, data preprocessing, and sometimes neglecting the crucial step of fine-tuning hyperparameters.

This study proposes an enhanced SVM based on Gray Wolf Optimizer (GWO) to classify fruit from images. GWO is a nature-inspired metaheuristic algorithm based on grey wolves' hunting and social behavior. It mimics the pack's hierarchy and cooperation to find optimal solutions [35]. GWO has been effectively applied in various hyperparameter optimization tasks for traditional machine learning [36-38] and deep learning [39-41]. In this study, GWO is employed to optimize the hyperparameters of SVM and the threshold value of low variance feature selection (LVFS) to improve the accuracy of SVM in classifying fruit from images. Moving Picture Experts Group-7 (MPEG-7) visual description [42] is used as input features to SVM. Using the MPEG-7 visual descriptor, the fruit image classification process does not require preprocessing and segmentation [43].

The strong points of the proposed method are as follows:

- Using GWO for hyperparameter optimization in SVM brings an innovative dimension to fruit image classification. To the best of the Author's knowledge, GWO has never been employed to optimize the hyperparameter of SVM for fruit image classification in the literature.
- By fusing MPEG-7 color and texture features, the proposed approach revolutionizes fruit image classification, eliminating the need for preprocessing and segmentation as well as

 DOI: 10.22266/ijies2024.0630.01

significantly enhancing the performance of SVM in classifying fruit images.

- Acknowledging the diverse range of fruit types, the proposed method stands out for its ability to provide accurate and adaptable classification across various fruit datasets.
- A new Indonesian fruit image dataset, called Ubaya-IFDS5000 dataset, is also proposed in this paper.

The rest of the paper is organized as follows. In Section 2, the focus is on the materials used and the proposed method. Section 3 unfolds the outcomes of experiments, presenting a meticulous analysis of the results obtained. Finally, Section 4 encapsulates the essence of the study, synthesizing the key takeaways and implications drawn from the study.

2. Materials and methods

2.1 Ubaya-IFDS3000 dataset

The Ubaya-IFDS3000 dataset [6] is the first image dataset used in this study. The dataset has 15 classes of Indonesian fruits, namely ambarella, avocado, dragon fruit, duku, durian, guava, mangosteen, pacitan orange, persimmon, pineapple, salak, sapodilla, siam lime, soursop, and star fruit. The dataset contains a total of 3000 images, with 200 images per class. All images were captured using a Canon EOS Kiss X6i camera in RGB (Red, Green, Blue) color space, with a dimension of 2592×1456 pixels and a resolution of 72 dpi and saved as a JPEG file. The dataset incorporated five background colors (pink, white, light blue, light green, and light yellow) and two illumination levels (160 and 1050 lumens). Images were taken with the camera tilted at 0° or 45° to introduce variance. Deliberate choices such as varying object counts and shadows enhanced dataset complexity. Fig. 1 displays some fruit images from the Ubaya-IFDS3000 dataset.

Figure. 1 Some fruit images in Ubaya-IFDS3000 dataset

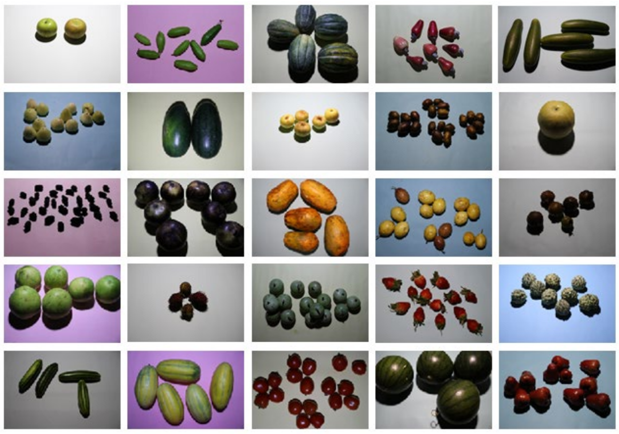

Figure. 2 Some fruit images in Ubaya-IFDS5000 dataset

2.2 Ubaya-IFDS5000 dataset

This study introduces Ubaya-IFDS5000, a novel Indonesian fruit image dataset, serving as the second dataset. Comprising 25 diverse Indonesian fruits, including ana apple, bilimbi, cantaloupe, cucumber, green water apple, cashew, star apple, long watermelon, manalagi apple, matoa, melon, mulberry, palmyra palm fruit, papaya, passion fruit, pomegranate, pomelo, rambutan, strawberry, sugar apple, timun krai, timun suri, tomato, watermelon, and watery rose apple. the dataset was sourced from traditional markets in Surabaya and Sidoarjo, East Java, Indonesia. All images were acquired using the same setup as in Ubaya-IFDS3000 dataset. The dataset contained a total of 5000 images, with 200 images per class. A Canon EOS 80D camera was used to capture all images in RGB color space, with a dimension of 2976×1984 pixels and a resolution of 72 dpi and saved as a JPEG file. Some images from the Ubaya-IFDS5000 can be seen in Fig. 2.

2.3 Supermarket produce dataset

Supermarket produce dataset [10] is the third image dataset used in this study. The dataset consists of 15 classes of fruits and vegetables, including agata potato, asterix potato, cashew, diamond peach, fuji apple, granny smith apple, honeydew melon, kiwi, nectarine, onion, orange, plum, spanish pear, taiti lime, and watermelon, as shown in Fig. 3. This dataset has a total of 2633 images with 75 to 264 images per class. Each image was captured on a white or clear background using a Canon PowerShot P1 camera with the dimension of 1024×768 pixels in RGB color space. The illumination of each image in the dataset was different when it is recorded. Each image in the dataset contains a different number of objects. The dataset contained images in diverse

Figure. 3 Some fruit and vegetable images from Supermarket produce dataset

poses, some with objects enclosed in plastic bags, intensifying specular reflection. Shadows and partially obscured objects, adding realism, were deliberately included in the dataset.

2.4 Feature extraction

In this study, five MPEG-7 color and texture features, namely color layout (CL), color structure (CS), scalable color (SC), edge histogram (EH), and homogeneous texture (HT), were directly acquired from the whole pixels in each fruit image. No preprocessing or segmentation steps were used in the extraction of these features. The color feature is selected as the one to use since it is a visual feature frequently utilized in object recognition. Additionally, color is resistant to being translated, rotated, and viewed from different angles. On the other hand, texture not only contains the structural information of the surface but also conveys the visual pattern of the surface [42].

Color layout refers to how colors are distributed over an image in the spatial domain. It is extracted in the YCbCr color space. Firstly, the image is divided into 64 equal blocks to ensure resolution invariance. The averages of pixel intensities in each channel are calculated as representative colors from each block to produce three 8×8 tiny images. The tiny images were transformed into frequency domain using discrete cosine Fourier transform (DCT). A zigzag scanning is carried out to select the first few DCT coefficients. The chosen coefficients are then subjected to a nonlinear quantization process to produce color layout features having a length of 120, 64 from the Y channel, 28 from the Cb channel, and 28 from the Cr channel.

The color structure of an image is an expression of the spatial color structure present in a particular location as well as the overall color distribution of the image. The spatial color structure information makes the feature sensitive to the specific image characteristics that are not visible by employing an ordinary color histogram. This feature is obtained in HMMD color space. The entire image is scanned with an 8×8 structuring element to produce a 256-bin histogram. The value of each bin is updated at each place in the image by counting the number of occurrences of a particular color within the structuring element.

Scalable color is extracted in HSV color space by constructing a 256 bins color histogram. The color space is uniformly quantized to 256 colors consisting of 16 levels H channels, four S channels, and four V channels. The histogram is normalized and then mapped into a 4-bit integer representation to give great weight to small values that occurred with a higher probability. After that, a Haar transform is applied to encode the histogram. This process is used to facilitate the scalability of the descriptor.

The spatial edge distribution of an image can be described using an edge histogram. Before the edge histogram can be extracted, the image is initially partitioned into 4×4 nonoverlapping big blocks. The edge information on each block is computed and categorized into five groups: vertical, horizontal, 45° diagonal, 135° diagonal, and isotropic. This is accomplished with the assistance of four directional selective edge detectors and one isotropic edge detector. As a result, the edge histogram feature contains five bins on each block. Therefore, there are 80 bins across the image.

The direction, hardness, and frequency of the pattern in the image can be characterized by a homogeneous texture feature. This feature is suitable for quantifying the texture of the image with a homogeneous characteristic. The 2D frequency space of the image is firstly segmented into 30 channels by five octave segments in the radial direction and six equal segments in the angle direction at the interval of 30 degrees before the homogeneous texture extraction. A Gabor-filtered Fourier transform is employed in each frequency channel. The mean energy and the deviation of energy are calculated from the filter output in each channel to produce a 60-bin histogram. The histogram is then concatenated with one bin histogram from the mean of pixel intensities and one bin histogram from the standard deviation of pixel intensities to obtain a 62-bin histogram. This study selects some fusion of MPEG-7 color and texture features according to the features used in fruit classification proposed in [9]. All fusions are tabulated in Table 1.

Table 1. The fusion of MPEG-7 color and texture features used for classification

i	Features fusion (F_i)	i	Features fusion (F_i)
1	CS	13	CS+CL+HT
2	SC	14	CS+CL+EH
3	CS+SC	15	CS+HT+EH
4	CS+CL	16	SC+CL+HT
5	CS+HT	17	SC+CL+EH
6	CS+EH	18	SC+HT+EH
7	SC+CL	19	CS+SC+CL+HT
8	SC+HT	20	CS+SC+CL+EH
9	SC+EH	21	CS+SC+HT+EH
10	CS+SC+CL	22	CS+CL+HT+EH
11	CS+SC+HT	23	SC+CL+HT+EH
12	CS+SC+EH	24	CS+SC+CL+HT+EH

2.5 Low variance features selection

The essential goals of feature selection are data cleaning, developing models that are less complicated and easier to understand, as well as improving classification accuracy. Several feature selection methods have been proposed and can be divided into four groups: statistical based, similarity based, sparse learning based, and information theoretical based. The technique applied during the feature selection process serves as the basis for this categorization [44].

Statistical based feature selection employs some statistical measures to extract the characteristics of features while selecting the important features. This method works independently from the learning algorithm. Therefore, it is more efficient compared to other methods. This study employs a simple statistical based feature selection that relies on variance, called low variance feature selection (LVFS) [44]. The importance of each feature is ranked based on its variance. A feature with a larger variance is considered more important than a feature with a smaller variance. A threshold value T needs to be determined first before feature selection. A feature with a variance less than T is considered unimportant and will be removed from the feature set. In this study, the value of T is determined using Grey Wolf Optimizer in the range [0,10], such that the best classification performance is achieved. Furthermore, this study uses the implementation of LVFS in Scikit-learn library [45] to perform feature selection.

2.6 Support vector machine

Support vector machine (SVM) is a classifier used initially for binary classification problems. Suppose the training data for the binary classification problem consists of the input feature vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$ and the corresponding expected output $y_1, y_2, \ldots, y_N$, where $\mathbf{x}_i \in \mathbb{R}^m$, $y_i \in \{-1,1\}$ for $i = 1,2,..,N$ and N is the number of training data. SVM aims to find a hyperplane as in Eq. (1),

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b \tag{1}$$

that can be used to classify an unknown input feature vector $\mathbf{x}$ by $\mathrm{sign}(y(\mathbf{x}))$, where $\mathbf{w} \in \mathbb{R}^m$ and $b \in \mathbb{R}$ are SVM parameters, and ϕ is a feature space mapping. The values $\mathbf{w}$ and b are determined by maximizing margin, which is the distance between $\mathbf{w}^T \phi(\mathbf{x}) + b = 0$ and the closest of the input feature vectors in training data [46].

The problem finding the optimum values of $\mathbf{w}$ and b can be formulated using the optimization problem in Eq. (2) and (3),

$$\min_{\mathbf{w},b,\xi_1,\xi_2,\ldots\xi_N} \frac{1}{2}\|\mathbf{w}\| + C\sum_{i=1}^{N} \xi_i \tag{2}$$

subject to

$$y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \;\; \xi_i \geq 0 \tag{3}$$

where $\xi_i, i = 1,2, \ldots, N$ are slack variables defined as distance between expected output y_i and predicted output $y(\mathbf{x}_i)$, and $C > 0$ is regularization parameter to control trade of between the slack variables and the margin. The above optimization problem can be formulated as dual problem as in Eq. (5) and (6),

$$\max_{\alpha_1,\alpha_2,\ldots\alpha_N} \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j\, y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{5}$$

subject to

$$0 \leq \alpha_i \leq C, \sum_{i=1}^{N} \alpha_i\, y_i = 0 \tag{6}$$

where $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ is the kernel function and $\alpha_1, \alpha_2, \ldots \alpha_N$ are the dual coefficients.

SVM can be extended to the multiclass problem by combining some binary SVM classifiers using either one versus one or one versus rest approaches. This study uses a one versus one approach by contracting $k(k+1)/2$ binary SVM classifiers on all combinations of two classes. The kernel function used in this study is the radial basis function as in

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma\|\mathbf{x} - \mathbf{x}'\|) \tag{7}$$

Eq. (7), where γ is a kernel coefficient. The accuracy of SVM depends on the values of hyper-parameters C and γ provided by the user. Therefore, the values of C and γ need to be optimized to obtain the best classification performance. In this study, Grey Wolf Optimizer was also employed to obtain the best values of C and γ from the range $[1,1000]$ and $[0.01,1]$, respectively. This study also used the implementation of multiclass SVM in Scikit-learn library to train the SVM model.

2.7 Grey wolf optimizer

Grey wolf optimizer (GWO) is a metaheuristic optimization algorithm that takes inspiration from grey wolves (*Canis lupus*) [35]. It works by imitating the social hierarchy and hunting strategy of grey wolves found in their natural environment. The social hierarchy is simulated using four types of grey: alpha, beta, delta, and omega. The alpha wolf is the group leader who decides on prey hunting. The beta wolf is on the second level that helps the alpha wolf make decisions or do other activities. The third level is the delta wolf with the job of caretakers, hunters, elders, sentinels, and scouts. The omega wolf consisted of wolves, not in alpha, beta, and delta. In each iteration of the GWO algorithm, the best solution is modelled as the alpha wolf (α). The second and third best solutions are the beta wolf (β) and the delta wolf (δ), respectively. The remaining possible solutions are all considered to be the omega wolf (ω). α, β, and δ wolves act as guides during optimization process and their movement will be followed by ω wolves.

The GWO algorithm employs a three-step hunting strategy: searching, encircling, and attacking prey to find the best solution. Suppose $\mathbf{P}(i)$ and $\mathbf{P}_p(i)$ are the position vector of a grey wolf and the prey at i^{th} iteration. The encircling behavior of a grey wolf can be modelled using Eq. (8) - (11),

$$\mathbf{A} = 2a\mathbf{r}_1 - a \tag{8}$$

$$\mathbf{C} = 2\mathbf{r}_2 \tag{9}$$

$$\mathbf{D} = \left|\mathbf{C}.\mathbf{P}_p(i) - \mathbf{P}(i)\right| \tag{10}$$

$$\mathbf{P}(i+1) = \mathbf{P}_p(i) - \mathbf{A}.\mathbf{D} \tag{11}$$

where $\mathbf{A}$ and $\mathbf{C}$ are coefficient vectors, a is a scalar with its value linearly decreased from 2 to 0 during the iteration process, $\mathbf{r}_1, \mathbf{r}_2$ are random vectors with elements falling in the range $[0,1]$, $|\ \ |$ is the element-wise absolute value of the vector, and the dot (.) operator is the element-wise vector multiplication.

The position of the best solution is unknown in the actual case. Therefore, during the hunting process, the movement of a grey wolf will be guided by α, β, and δ wolves. This condition assumes that α, β, and δ wolves have more information about the prospective prey location. Eq. (12) - (14) are used to describe the movement of the grey wolf in each iteration based on the position of α, β, and δ wolves,

$$\mathbf{D}_j = \left|\mathbf{C}_j.\mathbf{P}_j(i) - \mathbf{P}(i)\right| \tag{12}$$

$$\widehat{\mathbf{P}}_j = \mathbf{P}_j(i) - \mathbf{A}_j.\mathbf{D}_j \tag{13}$$

$$\mathbf{P}(i+1) = \frac{1}{3}\left(\widehat{\mathbf{P}}_\alpha + \widehat{\mathbf{P}}_\beta + \widehat{\mathbf{P}}_\delta\right) \tag{14}$$

where $\mathbf{A}_j$ and $\mathbf{C}_j$, for $j = \alpha, \beta, \delta$, are coefficient vectors as defined in Eq. (10) and Eq. (11), respectively, and $\mathbf{P}_\alpha(i), \mathbf{P}_\beta(i), \mathbf{P}_\delta(i)$ are the position of α, β, and δ wolves at i^{th} iteration, respectively.

In this study, GWO was used to determine the value of threshold T in LVFS and the value of hyperparameters C and γ in SVM such that the best classification accuracy of SVM can be achieved in classifying fruit images. Therefore, the position vector of a grey wolf will consist of T, C, and γ, as in Eq. (15).

$$\mathbf{P} = (T, C, \gamma) \tag{15}$$

The number of grey wolf population (N) used to search for the best solution was five wolves with the maximum iteration of 10. GWO will search for the best solution $\mathbf{P}$ by maximizing classification accuracy with the objective function as in Eq. (16),

$$f(\mathbf{P}) = 1 - acc(\mathbf{P}) \tag{16}$$

where $acc(\mathbf{P})$ is the classification accuracy of SVM with $C = \mathbf{P}[2]$ and $\gamma = \mathbf{P}[3]$ and LVFS is performed with $T = \mathbf{P}[1]$.

2.8 Evaluation

Every dataset was divided into two nonoverlapping subsets with a proportion of 1:1, one for the training dataset and the remaining for the testing dataset. Stratified random sampling without replacement is employed to construct five pairs of training and testing datasets to ensure that all classes have the same proportion in both subsets.

The SVM classifier was trained using five training datasets, and the performance was evaluated using the corresponding testing data set. Five accuracies $acc_t,\ t = 1,2,3,4,5$ were calculated from five testing datasets using Eq. (16),

$$acc_t = \frac{C_t}{T_t} \times 100\% \quad (17)$$

where C_t and T_t are the number of correctly classified images and the total images in the t^{th} testing dataset, respectively. Finally, all acc_t were summarized using average to represent the performance of the SVM classifier.

3. Results and Discussion

The summary in Table 2 outlines the classification accuracy of enhanced SVM based on GWO on three datasets. The data printed in bold indicate the top three classification accuracies for each dataset. Among the feature fusions analyzed, the top three performing feature fusions in terms of classification accuracy on the Ubaya-IFDS3000 dataset are as follows. The feature fusion with the highest accuracy, securing the first and second ranks, were $\boldsymbol{F}_3$ and $\boldsymbol{F}_{10}$ with an average accuracy of 99.21%. Following closely, $\boldsymbol{F}_{11}$ claimed the third rank with average accuracies of 99.16%. These features showcase exceptional discriminative capabilities, playing a crucial role in achieving the outstanding performance of the fruit image classification model on the Ubaya-IFDS3000 dataset.

The exploration of classification accuracy within the Ubaya-IFDS5000 dataset revealed a marginally lower performance than Ubaya-IFDS3000 dataset, ranging from 94.83% to 98.29%. This disparity can be attributed to the inherent complexity and diversity present in the Ubaya-IFDS5000 dataset, potentially posing challenges for precise fruit image classification. The top three performing feature fusions regarding classification accuracy using the proposed method in Ubaya-IFDS5000 dataset are as follows. Feature fusion $\boldsymbol{F}_{19}$ stood out with an impressive average accuracy of 98.29, closely followed by $\boldsymbol{F}_{11}$ and $\boldsymbol{F}_{10}$, which exhibited notable accuracy scores of 98.18 and 97.71% respectively. These feature fusions demonstrated a remarkable ability to differentiate between the dataset's diverse fruit classes. Interestingly, as can be observed from Table 2, the feature fusions that attained the highest classification accuracy in the Ubaya-IFDS5000 dataset differ from those in the Ubaya-IFDS3000 dataset. This discrepancy shows each dataset's unique challenges and intricacies, necessitating distinct features for optimal classification performance.

Table 2. Classification accuracy of optimized SVM using GWO

$\boldsymbol{F}_i$	Average accuracy (%)		
	Ubaya-IFDS3000	**Ubaya-IFDS5000**	**Supermarket produce**
	GWO	GWO	GWO
$\boldsymbol{F}_1$	98.20	95.45	99.67
$\boldsymbol{F}_2$	98.61	96.30	99.48
$\boldsymbol{F}_3$	**99.21**	97.43	**99.82**
$\boldsymbol{F}_4$	98.19	95.64	99.65
$\boldsymbol{F}_5$	97.89	96.30	99.61
$\boldsymbol{F}_6$	97.67	95.40	99.56
$\boldsymbol{F}_7$	98.85	96.50	99.67
$\boldsymbol{F}_8$	98.60	97.47	99.68
$\boldsymbol{F}_9$	98.37	95.04	99.57
$\boldsymbol{F}_{10}$	**99.21**	**97.71**	**99.85**
$\boldsymbol{F}_{11}$	**99.16**	**98.18**	**99.77**
$\boldsymbol{F}_{12}$	98.85	96.26	99.70
$\boldsymbol{F}_{13}$	98.09	96.66	99.59
$\boldsymbol{F}_{14}$	98.08	95.45	99.53
$\boldsymbol{F}_{15}$	96.88	96.09	99.45
$\boldsymbol{F}_{16}$	99.03	97.69	99.64
$\boldsymbol{F}_{17}$	98.55	94.83	99.64
$\boldsymbol{F}_{18}$	98.45	96.87	99.61
$\boldsymbol{F}_{19}$	99.15	**98.29**	99.76
$\boldsymbol{F}_{20}$	98.77	96.90	99.70
$\boldsymbol{F}_{21}$	98.89	97.51	99.71
$\boldsymbol{F}_{22}$	97.21	96.05	99.53
$\boldsymbol{F}_{23}$	98.59	97.11	99.61
$\boldsymbol{F}_{24}$	98.80	97.60	99.68

The classification accuracy results on the Supermarket produce dataset exhibited a remarkable level of performance, surpassing the outcomes observed in the two preceding datasets. The achieved classification accuracy showed a consistent trend of superiority across the features, ranging from 99.45% to 99.85%. Particularly noteworthy was the top-tier accuracy attained by the proposed method by employing feature fusions $\boldsymbol{F}_{10}$, $\boldsymbol{F}_3$, and $\boldsymbol{F}_{11}$, with remarkable accuracies of 99.85%, 99.82%, and 99.77%, respectively. Interestingly, the feature fusions that produced the highest accuracy on Supermarket produce dataset is the same as Ubaya-IFDS3000 dataset.

This study also compared classification accuracy between the proposed method and optimized SVM based on RSO, AOA, TSA, HHO, BO for hyperparameter tuning. Table 3 presents a comparative analysis of the proposed method against other optimization algorithms in terms of accuracy across different datasets. In Ubaya-IFDS3000 dataset with feature fusion CS+SC, optimized SVM

Table 3. Comparison of the proposed method with the other optimization algorithms

Optimization algorithm	Accuracy (%)
Ubaya- IFDS3000 dataset with CS+SC	
RSO	98.32
AOA	98.91
TSA	99.13
HHO	98.91
BO	98.84
GWO (this study)	**99.21**
Ubaya- IFDS3000 dataset with CS+SC+CL+HT	
RSO	97.27
AOA	97.97
TSA	97.82
HHO	98.25
BO	96.90
GWO (this study)	**98.29**
Supermarket produce dataset with CS+SC+CL	
RSO	99.79
AOA	99/79
TSA	99.82
HHO	99.83
BO	99.79
GWO (this study)	**99.85**

based on GWO outperforms alternative algorithms with an accuracy of 99.21%, surpassing RSO, AOA, TSA, HHO, and BO, which scored 98.32%, 98.91%, 99.13%, 98.91%, and 98.84% respectively. Similarly, on the Ubaya-IFDS5000 dataset with feature fusion CS+SC+CL+HT, GWO achieves an accuracy of 98.29%, surpassing RSO, AOA, TSA, HHO and BO, which scored 97.27%, 97.97%, 97.82%, 98.25% and 96.90% respectively. The comparison extends to the Supermarket produce dataset with feature fusion CS+SC+CL, where GWO demonstrates the highest accuracy at 99.85%, surpassing RSO, AOA, TSA, HHO and BO, which achieved 99.79%, 99.79%, 99.82%, 99.83%, and 99.79% accuracy respectively. These results underscore the superior performance of the proposed GWO method across various datasets in fruit image classification.

The classification performance of the proposed method was also compared to other classification methods proposed in previous studies in classifying fruit images on Ubaya-IFDS3000 and Supermarket produce datasets, as shown in Table 4 and Table 5, respectively. Previously, two studies have utilized the Ubaya-IFDS3000 dataset. The first study employed an ensemble of k-nearest neighbors (k-NN) and Linear Discriminant Analysis (LDA) with features CS, SC, CL+EH [6]. The second study used an ensemble of optimized Extreme Learning Machines (ELMs) with features SC+HT, CS+SC+CL, CS+SC+HT [9]. As shown in Table 4, the proposed method with features CS+SC,

Table 4. Comparison of the proposed method with the previous studies on Ubaya-IFDS3000 dataset

Method	Accuracy (%)
Ensemble of *k*-NN and LDA [6]	97.80
Ensemble of optimized ELMs [9]	98.03
Enhanced SVM based on GWO with CS+SC (this study)	**99.21**

demonstrates superior performance. These results indicate that the proposed method outperforms the previous studies regarding classification accuracy on the Ubaya-IFDS3000 dataset.

For Supermarket Produce dataset, several methods have been proposed to classify fruit images in this dataset both for traditional machine learning and deep learning approaches. For traditional machine learning approach, there were SVM-fusion with input features global color descriptor (GCH), Unser descriptor, and color coherent vector (CCV) [10], automatic classifier fusion [11], SVM with an improved sum and difference histogram (ISADH) as input features [12], SVM with input features fusion of color and texture features [16], and SVM with input features census transform histogram (CENTRIST) and hue saturation histogram [8].

For deep learning approach, there were six layers of convolutional neural network (CNN) with data augmentation and pretrained visual geometry group-16 (VGG-16) model [24], attention-based MobileNetV2 [28], and optimized RNN with DenseNet169 as feature extraction [29]. As shown in Table 5, the proposed method outperforms the results of previous studies in classifying fruit images from Supermarket Produce dataset. Even though the VGG-16 [24] and optimized RNN [29] produced

Table 5. Comparison of the proposed method with the previous studies on Supermarket produce dataset

Method	Accuracy (%)
SVM-fusion with GCH+Unser+ CCV [10]	97.00
Automatic classifier fusion [11]	98.80
SVM with ISADH [12]	99.00
SVM with fusion of color and texture features [16]	93.84
SVM with CENTIRIST+ Hue Saturation Histogram [8]	97.23
AlexNet with data augmentation [26]	99.46
Six layers of CNN with data augmentation [24]	99.49
VGG-16 [24]	99.75
Attention-based MobileNetV2 [28]	95.75
DenseNet169+Optimized RNN [29]	99.84
Enhanced SVM based on GWO with CS+SC+CL (this study)	**99.85**

almost the same accuracy as the proposed method, these methods required more training data (85% and 70%, respectively) than the proposed method in this study (50%). Furthermore, optimized RNN [29] only used 75 samples per class in Supermarket produce dataset in the experiment.

4. Conclusion

This study proposes an enhanced support vector machine (SVM) based on grey wolf optimizer (GWO) to classify fruit from images. GWO, inspired by the social behavior of grey wolves, has been employed successfully for optimizing the hyperparameters of SVM and the threshold of low variance feature selection. This approach demonstrates its potential in improving SVM accuracy, using MPEG-7 visual descriptors fusion as input features, avoiding the need for preprocessing and segmentation steps. The experimental results show that the proposed method produced remarkable accuracies across three diverse datasets. The top-performing feature fusions, particularly CS+SC, CS+SC+CL+HT, and CS+SC+CL, consistently outshine others, yielding impressive average accuracies of 99.21%, 98.29%, and 99.85% on the Ubaya-IFDS3000, Ubaya-IFDS5000, and Supermarket produce datasets, respectively.

Comparative analyses against alternative optimization algorithms and previous studies highlight the superiority of the proposed GWO-based method. Across various datasets, the optimized SVM using GWO consistently outperforms RSO, AOA, TSA, HHO, and BO, with accuracy differences ranging from 0.02% to 1.39%. Moreover, when compared to previous studies, the proposed method demonstrates superior performance on both Ubaya-IFDS3000 and Supermarket produce datasets, showcasing its efficacy in fruit image classification. In addition to the GWO employed in this study for hyperparameter optimization, future research can explore alternative metaheuristic optimization techniques further to enhance the performance of fruit image classification models in traditional machine learning and deep learning models.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This study was supported by Kemendikbud Ristek Republic of Indonesia and University of Surabaya under PDUPT Grant No. 018/SP-Lit/LPPM-01/RistekBRIN/Multi/FT/III/2020 and PPB Grant No. 151/SP-Lit/LPPM-01/FT/XI/2023.

References

[1] J. Siswantoro, A. S. Prabuwono, A. Abdullah, and B. Indrus, "Hybrid neural network and linear model for natural produce recognition using computer vision", *J. ICT Res. Appl.*, Vol. 11, No. 2, 2017.

[2] M. K. Tripathi and D. D. Maktedar, "A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields: A survey", *Inf. Process. Agric.*, Vol. 7, No. 2, pp. 183–203, 2020.

[3] K. Hameed, D. Chai, and A. Rassau, "A comprehensive review of fruit and vegetable classification techniques", *Image Vis. Comput.*, Vol. 80, pp. 24–44, 2018.

[4] Y.-D. Zhang, Z. Dong, X. Chen, W. Jia, S. Du, K. Muhammad, and S.-H. Wang, "Image based fruit category classification by 13-layer deep convolutional neural network and data augmentation", *Multimed. Tools Appl.*, Vol. 78, No. 3, pp. 3613–3632, 2019.

[5] S. M. M. Roomi, R. J. Priya, S. Bhumesh, and P. Monisha, "Classification of mangoes by object features and contour modeling", In: *Proc. of International Conf. on Machine Vision and Image Processing (MVIP)*, pp. 165–168, 2012.

[6] J. Siswantoro, H. Arwoko, and M. Widiasri, "Indonesian fruits classification from image using MPEG-7 descriptors and ensemble of simple classifiers", *J. Food Process Eng.*, Vol. 43, No. 7, pp. 1–13, 2020.

[7] S. Jana, S. Basak, and R. Parekh, "Automatic fruit recognition from natural images using color and texture features", In: *Proc. of Devices for Integrated Circuit (DevIC)*, pp. 620–624, 2017.

[8] J. Y. R. Cornejo and H. Pedrini, "Automatic fruit and vegetable recognition based on CENTRIST and color representation", In: *Proc. of Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 21st Iberoamerican Congress, CIARP 2016, Lima, Peru*, pp. 76–83, 2017.

[9] J. Siswantoro, H. Arwoko, and M. Z. F. N. Siswantoro, "Fruits Classification from Image using MPEG-7 Visual Descriptors and Extreme Learning Machine", In: *Proc. of 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, pp. 682-687, 2020.

[10] A. Rocha, D. C. Hauagge, J. Wainer, and S. Goldenstein, "Automatic fruit and vegetable classification from images", *Comput. Electron. Agric.*, Vol. 70, No. 1, pp. 96–104, 2010.

[11] F. A. Faria, J. A. dos Santos, A. Rocha, and R. S. Torres, "Automatic Classifier Fusion for Produce Recognition", In: *Proc. of Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on*, pp. 252–259, 2012.

[12] S. R. Dubey and A. S. Jalal, "Robust Approach for Fruit and Vegetable Classification", *Procedia Eng.*, Vol. 38, pp. 3449–3453, 2012.

[13] Y. Zhang and L. Wu, "Classification of fruits using computer vision and a multiclass support vector machine", *Sensors (Basel)*, Vol. 12, No. 9, pp. 12489–12505, 2012.

[14] Y. Zhang, S. Wang, G. Ji, and P. Phillips, "Fruit classification using computer vision and feedforward neural network", *J. Food Eng.*, Vol. 143, pp. 167–177, 2014.

[15] S. Wang, Y. Zhang, G. Ji, J. Yang, J. Wu, and L. Wei, "Fruit classification by wavelet-entropy and feedforward neural network trained by fitness-scaled chaotic ABC and biogeography-based optimization", *Entropy*, Vol. 17, No. 8, pp. 5711–5728, 2015.

[16] S. R. Dubey and A. S. Jalal, "Fruit and vegetable recognition by fusing colour and texture features of the image using machine learning", *Int. J. Appl. pattern Recognit.*, Vol. 2, No. 2, pp. 160–181, 2015.

[17] Y. Zhang, P. Phillips, S. Wang, G. Ji, J. Yang, and J. Wu, "Fruit classification by biogeography-based optimization and feedforward neural network", *Expert Syst.*, Vol. 33, No. 3, pp. 239–253, 2016.

[18] J. Siswantoro, A. S. Prabuwono, A. Abdullah, and B. Idrus, "Automatic image segmentation using sobel operator and k-means clustering: A case study in volume measurement system for food products", In: *Proc. of 2015 International Conference on Science in Information Technology (ICSITech)*, pp. 13-18, 2016.

[19] Y.-D. Zhang, L. Wu, and S. Wang, "Magnetic resonance brain image classification by an improved artificial bee colony algorithm", *Prog. Electromagn. Res.*, Vol. 116, pp. 65–79, 2011.

[20] A. A. Christy and P. A. D. V. Raj, "Adaptive biogeography based predator–prey optimization technique for optimal power flow", *Int. J. Electr. Power Energy Syst.*, Vol. 62, pp. 344–352, 2014.

[21] J. Siswantoro, A. S. Prabuwono, A. Abdullah, and B. Idrus, "A linear model based on Kalman filter for improving neural network classification performance", *Expert Syst. Appl.*, Vol. 49, 2016.

[22] H. Muresan and M. Oltean, "Fruit recognition from images using deep learning", *Acta Univ. Sapientiae*, Vol. 10, No. 1, pp. 26–42, 2018.

[23] S.-H. Wang and Y. Chen, "Fruit category classification via an eight-layer convolutional neural network with parametric rectified linear unit and dropout technique", *Multimed. Tools Appl.*, pp. 1–17, 2018.

[24] M. S. Hossain, M. Al-Hammadi, and G. Muhammad, "Automatic Fruit Classification Using Deep Learning for Industrial Applications,", *IEEE Trans. Ind. Informatics*, Vol. 15, No. 2, pp. 1027–1034, 2019.

[25] J. L. Joseph, V. A. Kumar, and S. P. Mathew, "Fruit Classification Using Deep Learning", In: *Proc. of Innovations in Electrical and Electronic Engineering*, pp. 807-817, 2021.

[26] A. Patino-Saucedo, H. Rostro-Gonzalez, and J. Conradt, "Tropical fruits classification using an alexnet-type convolutional neural network and image augmentation", In: *Proc. of Neural Information Processing: 25th International Conference, ICONIP 2018, Siem Reap, Cambodia*, pp. 371–379, 2018.

[27] J. L. Rojas-Aranda, J. I. Nunez-Varela, J. C. Cuevas-Tello, and G. Rangel-Ramirez, "Fruit classification for retail stores using deep learning", In: *Proc. of Pattern Recognition: 12th Mexican Conference, MCPR 2020, Morelia, Mexico*, pp. 3–13, 2020.

[28] T. B. Shahi, C. Sitaula, A. Neupane, and W. Guo, "Fruit classification using attention-based MobileNetV2 for industrial applications", *PLoS One*, Vol. 17, No. 2, p. e0264586, 2022.

[29] K. Shankar *et al.*, "An automated hyperparameter tuning recurrent neural network model for fruit classification", *Mathematics*, Vol. 10, No. 13, p. 2358, 2022.

[30] N. Ismail and O. A. Malik, "Real-time visual inspection system for grading fruits using computer vision and deep learning techniques", *Inf. Process. Agric.*, Vol. 9, No. 1, pp. 24–37, 2022.

[31] A. H. Alharbi, S. Alkhalaf, Y. Asiri, S. Abdel-Khalek, and R. F. Mansour, "Automated Fruit Classification using Enhanced Tunicate Swarm Algorithm with Fusion based Deep Learning", *Comput. Electr. Eng.*, Vol. 108, p. 108657, 2023.

[32] R. R. Rajalaxmi, M. Saradha, S. K. Fathima, V. E. Sathish Kumar, M. Sandeep Kumar, and J. Prabhu, "An Improved MangoNet Architecture Using Harris Hawks Optimization for Fruit Classification with Uncertainty Estimation", *J. Uncertain Syst.*, Vol. 16, No. 1, p. 2242006, 2023.

[33] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization", *J. Mach. Learn. Res.*, Vol. 13, No. 1, pp. 281–305, 2012.

[34] B. Guo, J. Hu, W. Wu, Q. Peng, and F. Wu, "The Tabu_Genetic algorithm: A novel method for hyper-parameter optimization of learning algorithms", *Electronics*, Vol. 8, No. 5, p. 579, 2019.

[35] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer", *Adv. Eng. Softw.*, Vol. 69, pp. 46–61, 2014.

[36] A. Al Shorman, H. Faris, and I. Aljarah, "Unsupervised intelligent system based on one class support vector machine and Grey Wolf optimization for IoT botnet detection", *J. Ambient Intell. Humaniz. Comput.*, Vol. 11, pp. 2809–2825, 2020.

[37] S. K. Parhi and S. K. Patro, "Prediction of compressive strength of geopolymer concrete using a hybrid ensemble of grey wolf optimized machine learning estimators", *J. Build. Eng.*, Vol. 71, p. 106521, 2023.

[38] K. Chen, S. Laghrouche, and A. Djerdir, "Remaining useful life prediction for fuel cell based on support vector regression and grey wolf optimizer algorithm", *IEEE Trans. Energy Convers.*, Vol. 37, No. 2, pp. 778–787, 2021.

[39] R. Mohakud and R. Dash, "Designing a grey wolf optimization based hyper-parameter optimized convolutional neural network classifier for skin cancer detection", *J. King Saud Univ. Inf. Sci.*, Vol. 34, No. 8, pp. 6280–6291, 2022.

[40] B. Z. Aufa, S. Suyanto, and A. Arifianto, "Hyperparameter setting of LSTM-based language model using grey wolf optimizer", In: *Proc. of International Conf. On Data Science and Its applications (ICoDSA)*, pp. 1–5, 2020.

[41] M. R. Falahzadeh, F. Farokhi, A. Harimi, and R. Sabbaghi-Nadooshan, "Deep convolutional neural network and gray wolf optimization algorithm for speech emotion recognition", *Circuits, Syst. Signal Process.*, Vol. 42, No. 1, pp. 449–492, 2023.

[42] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: multimedia content description interface*. John Wiley & Sons, 2002.

[43] J. Siswantoro, H. Arwoko, and M. Widiasri, Image Based Indonesian Fruit Recognition using MPEG-7 Color Structure Descriptor and k-Nearest Neighbor", In: *Proc. of International Conf. on Informatics, Technology, and Engineering 2019 (InCITE 2019)*, p. D-84-D-90, 2019.

[44] J. Li *et al.*, "Feature Selection: A Data Perspective", *ACM Comput. Surv.*, Vol. 50, No. 6, pp. 1–45, 2017.

[45] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python", *J. Mach. Learn. Res.*, Vol. 12, pp. 2825–2830, Oct. 2011.

[46] C. M. Bishop, *Pattern recognition and machine learning*, springer, 2006.