Article

Comparison of Different Classifiers and the Majority Voting Rule for the Detection of Plum Fruits in Garden Conditions

Razieh Pourdarbani [1,*], Sajad Sabzi [1], Mario Hernández-Hernández [2], José Luis Hernández-Hernández [3], Ginés García-Mateos [4], Davood Kalantari [5] and José Miguel Molina-Martínez [6]

[1] Department of Biosystems Engineering, College of Agriculture, University of Mohaghegh Ardabili, Ardabil 56199-11367, Iran; sajadsabzi2@gmail.com

[2] Faculty of Engineering, Autonomous University of Guerrero, Chilpancingo, Guerrero 39087, Mexico; mhernandezh@uagro.mx

[3] Division of Research and Graduate Studies, TecNM/Technological Institute of Chilpancingo, Chilpancingo, Guerrero 39070, Mexico; joseluis.hernandez@itchilpancingo.edu.mx

[4] Computer Science and Systems Department, University of Murcia, 30100 Murcia, Spain; ginesgm@um.es

[5] Department of Mechanics of Biosystems Engineering, Faculty of Agricultural Engineering, Sari Agricultural Sciences and Natural Resources University, Sari 48181 68984, Iran; d.kalantari@sanru.ac.ir

[6] Agromotic and Marine Engineering Research Group, Technical University of Cartagena, 30203 Cartagena, Murcia, Spain; josem.molina@upct.es

* Correspondence: r_pourdarbani@uma.ac.ir; Tel.: +98-914-3162785

Abstract: Color segmentation is one of the most thoroughly studied problems in agricultural applications of remote image capture systems, since it is the key step in several different tasks, such as crop harvesting, site specific spraying, and targeted disease control under natural light. This paper studies and compares five methods to segment plum fruit images under ambient conditions at 12 different light intensities, and an ensemble method combining them. In these methods, several color features in different color spaces are first extracted for each pixel, and then the most effective features are selected using a hybrid approach of artificial neural networks and the cultural algorithm (ANN-CA). The features selected among the 38 defined channels were the b* channel of L*a*b*, and the color purity index, C*, from L*C*h. Next, fruit/background segmentation is performed using five classifiers: artificial neural network-imperialist competitive algorithm (ANN-ICA); hybrid artificial neural network-harmony search (ANN-HS); support vector machines (SVM); *k* nearest neighbors (kNN); and linear discriminant analysis (LDA). In the ensemble method, the final class for each pixel is determined using the majority voting method. The experiments showed that the correct classification rate for the majority voting method excluding LDA was 98.59%, outperforming the results of the constituent methods.

Keywords: remote sensing in agriculture; artificial neural network hybridization; environmental conditions; majority voting; plum segmentation

1. Introduction

It is a well-known fact that agricultural engineering is one of the main areas of application for remote sensing technologies [1]. These can include spaceborne, airborne and terrestrial platforms, with applications in crop monitoring, water management, fertigation, harvesting, and decision making in general. In particular, using automated or semi-automated harvesting methods is not only economically cost effective, but also more efficient than traditional methods, and significantly reduces the harvest

time, as observed in some of the early works on automated harvesting [2]. There are currently several methods of automated harvesting using different remote image capture techniques. The most common are based on the use of harvesting robots using image-processing modules [3–7]. Since farming robots need to work in uncontrolled environments, both natural lighting conditions and complex background have to be considered, making segmentation particularly sensitive and complicated. Direct sunlight causes images and frames to be extremely intense in the sunny areas and poorly lit in the shaded areas. This could make the segmentation system to be confused [8]. Moreover, other factors, such as complex background, partial occlusion, and poor capture quality [9], must also be considered.

Regarding these applications, Miao et al. [10] captured 380 RGB images in three morning, noon and evening lighting conditions for farmland recognition and classification, with a resolution of 2576×1932 pixels. Then, from each image, eight features were extracted that include the three components of the HSL color space, the second and third components of HSV, and the three components of HSI. The dimensionality of the data was reduced using 5 eigenvectors extracted with principal components analysis. Finally, images were classified using a radial basis function neural network. The results showed that this classification system was able to classify images in sunny and cloudy days with an accuracy of 84.58% and 68.11%, respectively. However, this classification accuracy is not enough for a practical use in machine vision systems, as it would result in many errors.

Zhao et al. [11] proposed a method for detecting immature green citrus fruits in orchards. This method was a combination of color features and the sum of the absolute conversion differences. To develop the detection algorithm on natural conditions, 126 images were taken in Florida, USA, at several different times of the day. The resulting images had a very high resolution of 3636×2736 pixels. From these images, 58 were used for training and the other 68 for a support vector machine (SVM) classification test. A comparative chromatic red and blue map was created and combined with the hue image extracted after the histogram alignment. Next, to identify potential fruit pixels, the sum of the absolute conversion difference and a block matching method were used. Finally, the results showed that the proposed algorithm had a detection accuracy of 83%.

Another interesting method for litchi fruit segmentation in stereo images under different lighting conditions has been proposed by Wang et al. [12]. This method involved the use of advanced wavelet transform to normalize object surface lighting. The Retinex-based image enhancement algorithm [13] was then used to highlight the fruit object from the normal brightness image. Finally, the fruit image was analyzed using *k*-means clustering. The results obtained indicated that this algorithm can be efficient in dealing with different light effects. Also, Sabzi et al. [14] proposed an apple segmentation algorithm under direct sunlight in garden conditions. To find the appropriate color space to perform part of the segmentation, 17 color spaces containing 51 color channels were examined. The main goal was to find the color space with the least number of colors to apply the threshold. One of the most important parts of this research was the arrangement of different segmentation steps. By applying color segmentation, texture analysis, and intensity conversion, a large number of background pixels were removed and apples were segmented in the images. The accuracy achieved by the segmentation algorithm was 98.92%.

Since shadows are problematic for segmentation, Weiyue et al. [15] presented a new algorithm for identifying and removing shadows in a system for apple segmentation. They combined group pixels and edge maps to produce hyper-pixel blocks with precise boundaries. A correlation matrix was used to obtain the ultra-metric counter map for shadow detection; a relative method for shadow removal was then used. Results showed that the mean square error decreased from 7.9% to 6.4% when using the edge probability map. Using this shadow removal algorithm, the segmentation accuracy was improved by 10%. The average processing time was 0.59 s, so the proposed algorithm was efficient to detect apples in orchards under natural light conditions. Finally, it is also worth mentioning the work by Sengupta and Lee [16], who developed an algorithm for identifying green tangerines under natural conditions. The shape and texture features were two integral parts of the algorithm. The shape attribute was used to identify as many fruits as possible. Tissue classification was also used to remove false positives. The reported fruit recognition rate was 80.4%.

More recently, Wang et al. [6] proposed an improvement in the methods of fruit localization for harvesting robots using binocular stereo images. The proposed method uses the Faster region-based convolutional neural network (R-CNN), which is able to achieve a recognition rate of 96.33% in six different conditions. Then, the window zooming method was applied to match detections in the stereo par, obtaining a 93.44% of correct matches, allowing the computation of the 3D position of the fruits. This is a breakthrough compared to other previous works of the authors, such as [17] for the detection of grape clusters in vineyards, [18] for green grape, and [19,20] for the case of litchi fruits. An interesting description of other current research works in fruit detection and localization can be found in [6].

As can be seen, segmentation is a very important step in many machine vision systems of remote imaging in agriculture. This step must be performed with high precision because the subsequent steps of the system depend on the accuracy of segmentation. For this reason, the purpose of the present study is to compare different pixel-by-pixel segmentation methods and their combination in an ensemble with a majority voting rule. The ensemble consists of different heterogeneous classifiers: hybrid artificial neural network-imperialist competitive algorithm (ANN-ICA); hybrid artificial neural network-harmony search (ANN-HS); support vector machines (SVM); and *k* nearest neighbors (kNN); and linear discriminant analysis (LDA), which can also be considered a baseline method. By using different classification techniques, the individual effectivity of these methods can be analyzed and compared. The system is applied to the segmentation of plum fruits in orchards in different ripeness stages under natural conditions.

2. Materials and Methods

The system proposed in this paper is based on color analysis, pixel-by-pixel segmentation, and an ensemble of classifiers. Figure 1 presents a diagram of main stages of the system. The key aspects are the definition of the color features to be extracted, the selection of the most effective features, the individual classification with the five basic classifiers, and the combination using a voting method. All of these operations are applied at a pixel-level. Previously, a training/test set of pixels has been obtained by a human expert, dividing them into two groups: plum fruit pixels, and background pixels. Therefore, we have a binary classification problem.
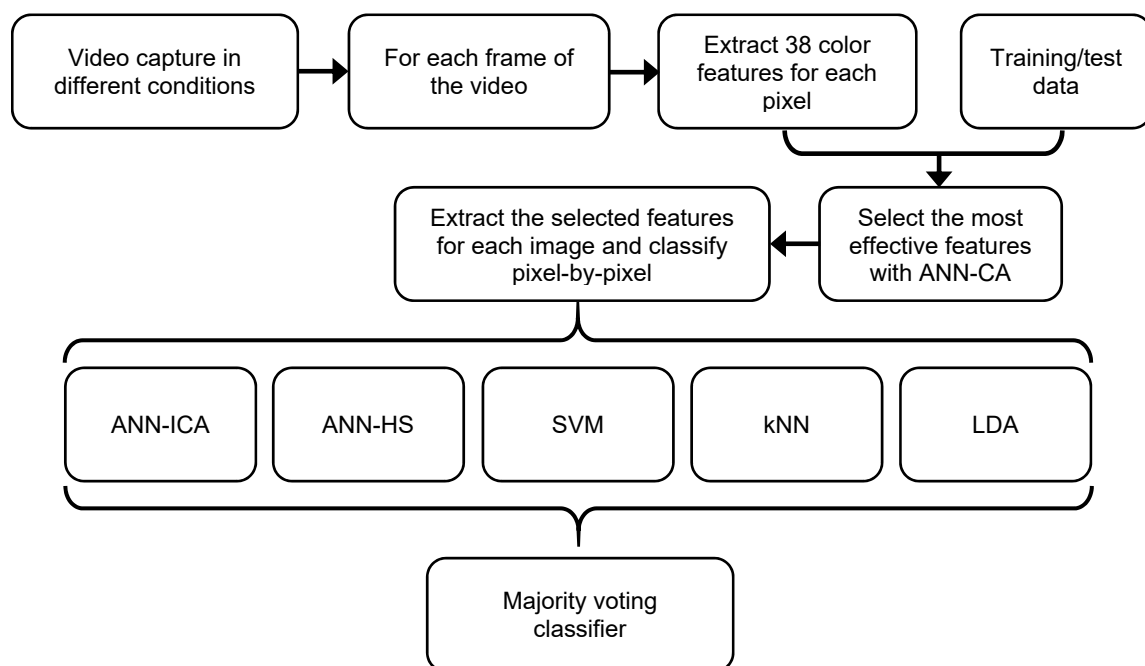

Figure 1. Diagram of the steps of the original approach for plum fruits segmentation.

2.1. Video Recording and Data Collection

The first step of the research process is to capture the videos in different environmental conditions. For this purpose, during the years 2018 and 2019, a total of 12 videos were recorded under different natural light intensities at different stages of plum fruit growth. The videos were produced from different fruit gardens in Kermanshah province, Iran (34°18′48.87″N, 47°4′6.92″E) using a color GigE industrial camera model DFK 23GM021 (Imaging Source Europe GmbH, Bremen, Germany), with a 1/3 inch Aptina CMOS MT9M021 sensor (ON Semiconductor, Aurora, CO, USA) and a spatial resolution of 1280 × 960 pixels. The mounted lens was model H0514-MP2 (Computar CBC Group, Tokyo, Japan), with f = 5 mm and F1.4. They were captured manually, simulating the route of a terrestrial harvesting robot.

Figure 2 shows eight image samples displaying different fruit growth conditions, as well as different light intensities. On the other hand, Table 1 shows the main characteristics of these videos. Finally, 49,347 pixels were manually extracted and classified from different frames by a human operator, classifying them in plum or background class. Of these, 34,543 pixels (70% of the data) were used to train the proposed algorithm, and the remaining 14,804 pixels (30% of the data) to test and validate the system.

(a)

(b)

(c)

(d)

(e)

(f)

Figure 2. *Cont.*
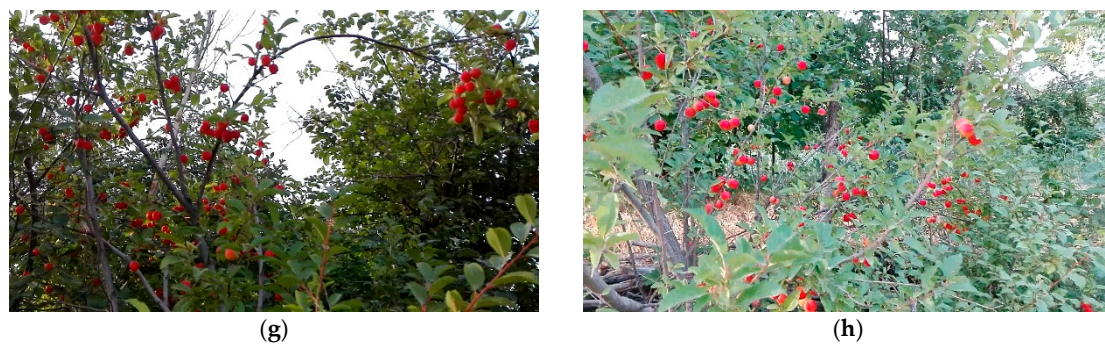
(**g**)　　　　　　　　　　　　　　　　　　　　　　　(**h**)

Figure 2. Different frames extracted from the recorded videos from different orchards. These frames correspond to the videos described in Table 1 with light intensities: (**a**) 826 lux; (**b**) 593 lux; (**c**) 342 lux; (**d**) 467 lux; (**e**) 1563 lux; (**f**) 1963 lux; (**g**) 639 lux; (**h**) 287 lux.

Table 1. Characteristics of the videos taken from the plum orchards under different conditions.

Number	Date	Time	Weather Condition	Light Intensity (lux)	Length (Minutes)	Number of Pixels Extracted
1	2018-7-9	10:25	Sunny	1769	01:02	3924
2	2018-7-16	18:45	Cloudy	593	00:50	3146
3	2018-7-20	20:15	Evening	826	01:10	4430
4	2018-7-27	15:45	Sunny	2078	00:50	3146
5	2018-8-1	6:45	Morning	287	00:58	3671
6	2018-8-6	17:35	Sunny	1563	01:58	7468
7	2019-7-6	17:30	Cloudy	342	00:45	2848
8	2019-7-11	11:45	Sunny	1689	00:54	3418
9	2019-7-14	13:45	Sunny	1963	01:15	4747
10	2019-7-18	7:15	Morning	467	00:58	3671
11	2019-7-25	20:30	Evening	639	01:15	4767
12	2019-7-29	15:15	Cloudy	738	01:05	4111

Since the purpose of the current research is to compare different segmentation algorithms under natural conditions, the appearance and complexity of the background is very important. Some of these backgrounds include leaves, branches, leaf tail, clear sky, cloudy sky, green vegetation, yellow plants, dense leaves and soil. Due to the variations in background objects, it is difficult to develop a comprehensive segmentation algorithm; but if the development of such an algorithm is achieved, many operations related to machine vision systems operating in different environmental conditions can be performed with high precision.

2.2. Selection of the Most Effective Color Features

The proposed method is based on color, as a powerful and effective feature widely applied to computer vision in agriculture [21]. Different color spaces, including RGB, HSV, HSL, HSI, L*a*b*, L*u*v*, XYZ, YCbCr, YUV, YIQ, CMY and L*C*h, were used to extract the different color properties of the pixels. The first, second and third components were extracted for all these color spaces; so, since there are 12 color spaces, 36 color features were extracted from each pixel.

C* channel is also defined as the color purity index, and it was introduced by Clerici et al. in [22]. Observe that texture features were not included in the system, since the plums typically appear with a small size and there is no characteristic texture on them.

Since the purpose of the segmentation algorithm is to be used by a robot in real-time, the computational efficiency is a key parameter. It is not feasible to use all the predefined features, even if it does not interfere with the algorithm's performance, since computing these 38 properties for each pixel would be time consuming. To solve this problem, the most effective color features must be chosen. In this regard, a hybrid method of artificial neural networks (ANNs) and the cultural algorithm

(CA) was used to select the most effective color properties. Table 2 shows the parameters used in the multilayer perceptron ANN to select these properties.

Table 2. Definition of the color models used in the present research. The equations to obtain these channels from the original RGB values are shown.

Color Model	Channel	Transformation from RGB
HSV	V	$V = M$; with $M = \max\{R,G,B\}$; $m = \min\{R,G,B\}$; $p = 60{\cdot}m/M$
	S	$S = (M - m)/M$
	H	$H = \{\, p(G\text{-}B) \text{ if } M = R ; 120 + p(B\text{-}R) \text{ if } M{=}G ; 240 + p(R\text{-}G) \text{ if } M{=}B \,\}$
HLS	L	$L = (M+m)/2$
	S	$S = (M\text{-}m)/\min\{M+m,\, 2\text{-}M\text{-}m\}$
HSI	I	$I = (R+G+B)/3$
	S	$S = 255 - m/I$
L*a*b*	L*	$L^* = \{\, 116{\cdot}Y^{1/3} \text{ if } Y > k ; 903.3{\cdot}Y \text{ if } Y \leq k \,\}$ with $k = 0.008856$
	a*	$a^* = 500(f(X) - f(Y))$ with $f(t) = \{t^{1/3} \text{ if } t > k ; 7.787t + 0.1379 \text{ if } t \leq k \,\}$
	b*	$b^* = 200(f(Y) - f(Z))$
L*u*v*	u*	$u^* = 13{\cdot}L^*{\cdot}(4X/(X+15Y+3Z) - 0.197939)$
	v*	$v^* = 13{\cdot}L^*{\cdot}(9Y/(X+15Y+3Z) - 0.468311)$
XYZ	X	$X = 0.607{\cdot}R + 0.174{\cdot}G + 0.200{\cdot}B$
	Y	$Y = 0.299{\cdot}R + 0.587{\cdot}G + 0.114{\cdot}B$
	Z	$Z = 0.066{\cdot}G + 1.116{\cdot}B$
YCrCb	Cr	$Cr = 0.713{\cdot}(R - Y) + 128$
	Cb	$Cb = 0.564{\cdot}(B - Y) + 128$
YUV	U	$U = -0.14713{\cdot}R - 0.28886{\cdot}G + 0.436{\cdot}B$
	V	$V = 0.615{\cdot}R - 0.51499{\cdot}G - 0.10001{\cdot}B$
YIQ	I	$I = -0.595716{\cdot}R - 0.274453{\cdot}G - 0.321263{\cdot}B$
	Q	$Q = 0.211456{\cdot}R - 0.522581{\cdot}G - 0.311135{\cdot}B$
CMY	C	$C = 255 - R$
	M	$M = 255 - G$
	Y	$Y = 255 - B$
L*C*h	C*	$C^* = [\, (a^*)^2 + (b^*)^2 \,]^{1/2}$
	h	$h = \tan^{-1}(b^*/a^*)$

The CA metaheuristic, like the genetic algorithms (GA) [23–26], performs the control of an optimization process. GA considers natural and biological evolution, while CA considers cultural evolution and the impact of cultural and social conditions that ultimately lead to a model for solving a problem. In a society, everyone who is most considered will directly and indirectly have the most impact on cultural evolution. In fact, these people will influence the way society speaks, walks, dresses, and so on. Thus, the ultimate goal of this algorithm is to find and develop these elites for cultural evolution [27].

The procedure of the hybrid approach ANN-CA is as follows. All the color features extracted for each pixel are considered. The process selects small vectors of properties, for example, vectors with two, three, or four features, with the corresponding output class, fruit or background. For each selection of properties, the vectors are sent to a multilayer perceptron ANN with the hidden layer properties shown in Table 3. The method divides the inputs in 70% for training, 15% for validation, and 15% for testing. The mean squared error (MSE) of each input vector is recorded, obtaining the average MSE of each selection of features. Finally, the vector with the least MSE is selected as the optimal vector, and the corresponding properties are selected as the most effective color features. The task of the CA metaheuristic is to generate the different combinations of features in a controlled way.

Table 3. Parameters used in the multilayer perceptron artificial neural network (ANN) to select the most effective features.

Number of Layers	Number of Neurons	Transfer Function	Backpropagation Network Training Function	Backpropagation Weight/Bias Learning Function
1	First Layer: 10	First Layer: tangent sigmoid	Levenberg-Marquardt	LVQ1 weight learning function

2.3. Underlying Color Classifiers

As depicted in Figure 1, the proposed segmentation method is based on an ensemble of different classifiers, which are then combined with a majority voting step. That is, each pixel is first classified with the five underlying methods, and the final class is obtained with the majority rule. For example, if three of the classifiers consider a pixel as fruit and the other two consider it as background, the final class of the pixel is fruit. This procedure has the additional advantage that it allows us to compare the accuracy obtained by each basic classifier. These are explained in the following subsections.

2.3.1. Hybrid Artificial Neural Network-Imperialist Competitive Algorithm (ANN-ICA)

The imperialist competitive algorithm (ICA) is a metaheuristic algorithm based on cultural, social and political evolution. In this algorithm, all countries are looking for a general optimal point to solve the optimization problem [28]. In our case, the task of ICA is to determine the optimal values of the multilayer perceptron ANN parameters. This network has 5 customizable hyperparameters that define the structure of the network and influence its performance: the number of hidden layers; the number of neurons per layer; the transfer function of each layer; the back-propagation network training function; and the weight/bias-back-propagation learning function.

The range of neurons selected was between 0 and 25 for each hidden layer, meaning a 0 that there is no hidden layer. The number of hidden layers is between 1 and 3. For each layer, a transfer function is required. In this study, this transfer function will be selected from the 13 transfer functions available in MATLAB neural network toolbox (MathWorks) [29], such as tangent sigmoid, log-sigmoid or linear. The next hyperparameter for the ICA is the function of the back-propagation network training; this will be selected from the 19 functions available also in the cited toolbox. Finally, the weight/bias backpropagation learning function was selected from 15 different available functions.

The procedure is that, initially, the ICA algorithm considers a vector with the same number of parameters as a vector with a minimum of 4 and a maximum of 8 members. Each member represents a parameter. For example, a vector of the form x = {19, radbas, trainscg, learnsomb} indicates that the ANN under study has one hidden layer with 19 neurons, with radial basis transfer function, scaled conjugate gradient backpropagation training function, and self-organizing map weight learning function. The obtained average MSE determines the performance of the ANN for each of the vectors transmitted, so that the vector with the least average MSE is selected as the optimal vector to adjust the ANN parameters. As in the other methods, the input samples are divided in 70% for training, 15% for validation, and 15% for testing.

2.3.2. Hybrid Artificial Neural Network-Harmony Search (ANN-HS)

This method is also a hybrid approach where a meta-heuristic algorithm is used to find the optimal hyperparameters of an ANN. In this case, the harmony search (HS) procedure imitates the natural process of music optimization. In the making of a song, the beauty of the song determines the step of each musical device; in other words, each device must be optimized. Therefore, the value of the objective function is determined by the values of the problem variables [30]. The process to optimize the ANN hyperparameters is very similar to the process described in the previous subsection, with the only difference in the order in which the vectors are selected by the algorithm. The maximum number of hidden layers, neurons per layer, and the available functions are the same as in ANN-ICA.

2.3.3. Support Vector Machines (SVM) Classifier

Support vector machines [31] are one of the most popular supervised learning methods used for regression and classification problems. This method performs classification operations using linear classification of data. This way, it attempts to find the hyperplane of maximum separability of the samples of the two classes. Using a kernel function, the input data is implicitly mapped into a high-dimensional space, thus allowing not linear decision boundaries.

In the experiments, the kernel function for the SVM classifier was a simple dot product. The soft margin parameter, or *C* value, was selected to 1. These values were empirically obtained through a series of experiments with the same dataset, where the kernel functions tested were: dot product, polynomial, radial basis function (RBF) and sigmoid. The values tested for C were: 0.01, 0.1, 1 and 10.

2.3.4. *k*-Nearest Neighbors (kNN) Classifier

k-nearest neighbors is another non-parametric method used for classification [32,33]. Similar to SVM, it is also a sample-based method. This algorithm classifies each new sample based on the closest properties of other samples in its neighborhood. In short, this method works as follows, given a parameter *k* indicating the number of neighbors to be considered:

1. Calculate the distance between the input sample and all the training samples. This distance is usually measured in terms of the Euclidean distance. If $X_1 = (X_{11}, X_{12} \ldots X_{1n})$ and $X_2 = (X_{21}, X_{22} \ldots X_{2n})$ are two given tuples, then the Euclidean distance between them is calculated as:

$$d(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (X_{1i} - X_{2i})^2} \tag{1}$$

2. Sort the training samples in ascending order of the distance, selecting the *k* nearest samples.

3. The class with the greatest number of neighbors in the previous set of *k* samples is assigned to the input sample.

Although, in theory, this method requires the distance to all the training samples to be computed, the process can be optimized to reduce the computational time required, such as using the condensed nearest neighbor rule [34]. In our case, considering the low dimensionality of the input vectors and the large number of training samples, the value of *k* was selected to 5 in the experiments. This value was empirically obtained in a trial and error process, testing different odd values from 1 to 11.

2.3.5. Linear Discriminant Analysis (LDA) Classifier

The last method used in the ensemble classifier is LDA, which is another well-known technique. LDA tries to find a hyperplane to separate the samples of both classes, considering the mean and variance of them. The method of LDA can be performed in three ways: direct, hierarchical and stepwise. The step-by-step approach is more widely used by researchers because it incorporates independent variables based on predictive power. Therefore, the stepwise method was used in this study [35].

In our research, LDA is used as a baseline method for the comparison of the other more advanced techniques. For this reason, the color features used in the LDA space are the same as in the other classifiers, instead of selecting different features for each classifier. It has been observed in some of our previous research that many non-linear decision boundaries appear in low dimensional color spaces, especially when a complex and varied background is considered [21]. Thus, LDA is not expected to achieve great results, but it is interesting to compare its accuracy.

2.4. Combination of the Ensemble Classifier

In an ensemble of classifiers, each input sample is processed by the constituent classifiers, and then their results are combined according to a predefined rule. In our case, the outputs of the basic classifiers are combined using a majority voting method. This means that a pixel is assigned to a certain class when a half or more classifiers produced that class. Other combination techniques have

been proposed in the literature, such as the variation of weighted voting [36], although they were not considered adequate for the present application.

2.5. Performance Evaluation Parameters of the Classifiers

2.5.1. Performance Parameters Based on the Confusion Matrix

The confusion matrix is a table that relates the real and predicted instances achieved by a classifier. It is a square matrix with a row and column for each existing class; in our case, it is 2×2. The matrix itself is useful for analyzing the performance of the classifier. In addition, using this matrix, different metrics can be defined and applied, as discussed below:

Sensitivity or Recall. Percent of the correct samples that have been correctly identified:

$$\text{Recall} = \frac{TP}{TP + FN} \times 100 \tag{2}$$

Accuracy or correct classification rate or overall accuracy. Total percentage of the correct system responses:

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} \times 100 \tag{3}$$

Specificity. Percentage of negative samples that are correctly identified:

$$\text{Specificity} = \frac{TN}{TN + FP} \times 100 \tag{4}$$

Precision. Percentage of correctly identified outputs that are actually true:

$$\text{Precision} = \frac{TP}{TP + FP} \times 100 \tag{5}$$

F_measure. Harmonic mean of the recall and the precision:

$$\text{F\_measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{6}$$

Here, the positive class is the fruit (the object of interest) and the negative class is the background. Therefore, TP is equal to the number of samples of plum fruit that are correctly classified; TN is the number of samples of the background class that are correctly classified; FN is the number of fruit pixels misclassified as background; and FP is the number of background pixels misclassified as fruit. It has to be noted that some measures should not be analyzed by themselves. For example, a naïve system that always says true would have a recall of 100%, while a system that always says false would have a specificity of 100%.

2.5.2. Receiver Operating Characteristic

Receiver operating characteristic (ROC) charts are used to evaluate the performance of classification systems. These charts are plotted in a coordinate system with two axes, the horizontal axis being the FP rate, and the vertical axis being the TP rate achieved by the classifier. The number of graphs is the number of classes available, although in the case of two classes both curves are equivalent. An ideal ROC curve would be a single point with $FP = 0$ and $TP = 1$, indicating perfect classification accuracy. Therefore, a diagram looking like orthogonal indicates a higher performance [37]. The points of the ROC represent the different configurations of the classifier, from the most restrictive (near the point with $TP = 0$, $FP = 0$) to the most permissive (point $TP = 1$, $FP = 1$). If the classification method cannot be configured, it would only have a working point, so the ROC would consist of two straight segments. This happens, for example, with SVM and kNN classifiers.

A useful criterion that summarizes the performance of segmentation based on the ROC chart is the area under the curve (AUC), which is the integral of the ROC curve. The minimum value for this criterion is 0.5 (a random classifier) and the maximum is 1 (a perfect classifier).

3. Results and Discussion

This section describes the experimental results obtained, and discusses the main findings from these experiments. First the partial results of the intermediate steps are presented, that is, the selection of the most effective color features and the configuration of the ANNs with ICA and HS metaheuristic algorithms. Then, the accuracy of the basic classifiers and the ensemble method are analyzed using the evaluation parameters previously described. Finally, the results obtained are compared with other state-of-the-art methods available in the literature.

3.1. Selection of the Color Features and Configuration of ANN-ICA and ANN-HS

As described in Section 2.2, the first step of the process is to select the most effective color features for the problem of interest, among the 38 available color channels. This is done with a hybrid approach ANN-CA, which tests different combinations of channels. Finally, the most effective features automatically selected were channel b* in the L*a*b* color space, and the color purity index, C*, from the L*C*h space, which is also derived from a* and b* channels (see Table 1). This predominance of the L*a*b* color space has also been reported in other applications of computer vision in agriculture [38]. However, it should be noted that these results may depend on the specific application domain.

Another interesting result to discuss is that the process ends with the selection of only 2 color features. Observe that this number is not fixed a priori. The ANN-CA process tests different combinations between one and six channels. However, only two channels are selected as the optimal configuration. This could indicate that a greater number of channels is prone to produce overfitting, thus leading to poor classification results. This conclusion is consistent with [38], where all the combinations of one, two, and three channels in nine standard color spaces for a problem of plant/soil segmentation are tested; the reported results indicate that the optimal selection consists of the channels L* and a* in L*a*b*. Therefore, a reduced number of features is enough for an effective use of color in segmentation problems. This limitation is also very useful to achieve good computational efficiency in the algorithms.

The other applications of hybrid approaches are in the configuration of the hyperparameters of the ANNs for classification, with ANN-ICA and ANN-HS, using the two features mentioned. The optimal configurations for ANN-ICA and ANN-HS are shown in Tables 4 and 5, respectively.

Table 4. Optimal values of ANN parameters determined by imperialist competitive algorithm (ICA).

Number of Layers	Number of Neurons	Transfer Function	Backpropagation Network Training Function	Backpropagation Weight/Bias Learning Function
2	First layer: 10 Second layer: 20	First layer: tangent sigmoid Second layer: tangent sigmoid	Levenberg-Marquardt	Gradient descent with momentum

Table 5. Optimal values of ANN parameters determined by harmony search (HS) algorithm.

Number of Layers	Number of Neurons	Transfer Function	Backpropagation Network Training Function	Backpropagation Weight/Bias Learning Function
2	First layer: 13 Second layer: 24	First layer: tangent sigmoid Second layer: triangular basis	Conscience bias	BFGS quasi-Newton backpropagation

In both cases, the optimal configuration of the ANNs consists of only two hidden layers with a similar number of neurons, although the process tests bigger sizes. This result may be related to the fact that the input tuples consist of only two values, so the necessary decision boundaries can be created with only 2 layers between 10 and 24 neurons per layer.

3.2. Classification Results of the Ensemble Method and the Basic Classifiers

In order to evaluate the reliability of the classifiers, 275 repetitions were performed for each method, that is, 275 independent executions of the training/testing process. The proposed ensemble method originally consisted of the 5 classifiers presented: ANN-ICA; ANN-HS; SVM; kNN; and LDA. However, it was observed that the poor results of LDA (as presented below) seriously affected the global performance of the ensemble. Therefore, we decided to include a new majority voting method removing LDA from the ensemble.

Figure 3 shows boxplots of the correct classification rates (CCR), or overall accuracy, achieved by all the classifiers in the fruit/background segmentation for the 275 executions. The red crosses indicate exceptionally low or high execution results.
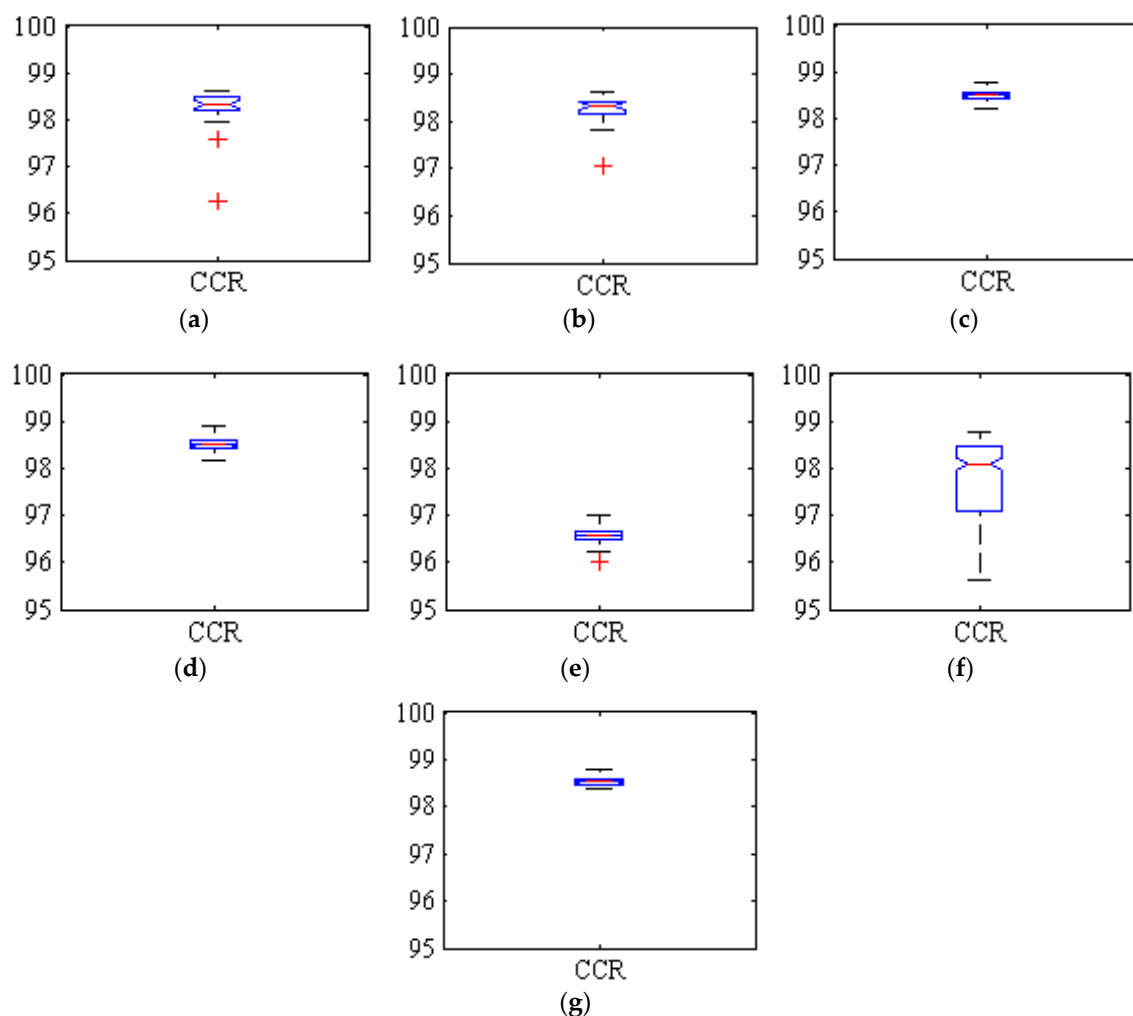

Figure 3. Boxplots of the correct classification rate (CCR) obtained by the classifiers for the 275 executions. (**a**) Hybrid ANN-ICA; (**b**) Hybrid ANN-HS; (**c**) SVM; (**d**) kNN; (**e**) LDA; (**f**) Voting method with LDA; (**g**) Voting method without LDA.

In all cases, the accuracy achieved was always above 96%, with best average results above 98%. Some methods were more consistent in these good results (i.e., the variance between executions is very

small), such as SVM, kNN and specially the ensemble method, while the results of LDA were also consistent but significantly lower, below 97%. This compactness of the boxplots indicates the close proximity of the values in different executions, and consequently the high reliability of the classification. The two hybrid methods based on ANN (i.e., ANN-ICA and ANN-HS) also produced good results, but with a bigger variance between executions. On the other hand, the original ensemble classifier including LDA was clearly affected by the errors of the poorest methods; it had an average accuracy of only 97.68%, and the variance was larger than that of all the basic classifiers. However, removing LDA from the ensemble, the method is able to improve the results of the constituent classifiers, both achieving a good average accuracy of 98.59%, and a very low variance.

Table 6 presents the confusion matrices and the error rates per class for the test data in the 275 repetitions for all the classifiers. Since there are 14,804 test samples and 275 iterations performed, the total accumulated is equivalent to 4,071,100 samples (32.6% of the fruit class, and 67.4% background).

Table 6. Confusion matrices, rates of error by class, and CCR for all the classifiers and the 275 repetitions.

Classification Method	Real/ Obtained Class	Fruit	Background	Total Data	Classification Error per Class (%)	Correct Classification Rate (%)
ANN-ICA	Fruit	1,283,547	44,638	1,328,185	3.36	98.28
	Backgr.	25,189	2,717,726	2,742,915	0.918	
ANN-HS	Fruit	1,281,556	46,629	1,328,185	3.51	98.24
	Backgr.	22,287	2,720,628	2,742,915	0.813	
SVM	Fruit	1,301,672	26,513	1,328,185	1.99	98.49
	Backgr.	34,989	2,707,926	2,742,915	1.28	
kNN	Fruit	1,297,335	30,850	1,328,185	2.32	98.50
	Backgr.	30,342	2,712,573	2,742,915	1.11	
LDA	Fruit	1,197,201	130,984	1,328,185	9.86	96.57
	Backgr.	8,581	2,734,334	2,742,915	0.313	
Voting with LDA	Fruit	1,247,066	81,119	1,328,185	6.12	97.68
	Backgr.	13,163	2,729,752	2,742,915	0.480	
Voting without LDA	Fruit	1,298,650	29,535	1,328,185	2.22	98.59
	Backgr.	27,819	2,715,096	2,742,915	1.01	

These matrices allow a deeper insight into the results. Since the number of background samples is bigger than fruit samples, all the methods tend to over-classify the new inputs into the background class, producing higher error rates for this class, i.e., the number of FP and FN samples are not balanced. This is especially prominent in the LDA method, where the error in the fruit class is 31 times bigger. Only the SVM classifier exhibits a balanced accuracy, with 1.99% error in the fruit class and 1.28% in the background. The CCR of SVM (98.49%) is lower than that kNN (98.50%), but the difference is not significant. The imbalance in the results of the ensemble without LDA is relatively small (as compared, for example, with the two ANN methods), and it also outperforms the CCR of the constituent methods.

The imbalance observed in the accuracies for the different classes is most probably due to the imbalance in the dataset between the fruit and background samples. In our case of study, the plums only represent about 2% of the whole images. Thus, the plum class has been oversampled, with 33% and 67% samples of the fruit and background classes, respectively. But although the imbalance of the samples has been reduced, there are still twice as many samples of the background class. It would be interesting to perform an additional subsampling of the background class, using for example half of the samples. However, while this could reduce the classification bias, the overall classification accuracy of the images would be smaller, since they contain about 98% of background.

The ROC curves obtained for all the methods are depicted in Figure 4. In these curves, the closer the curve is to the vertical, the higher the performance. In general, all the ROCs exhibit good results,

but the accuracy of SVM is again the best of all the basic methods. As mentioned above, the curves of SVM, kNN and the ensemble classifiers are piecewise linear, since these techniques cannot be adjusted to be more or less restrictive. This fact makes that, even that these methods achieve a good accuracy, the AUC parameter is degenerated since it loses its original meaning of measuring different configurations of the classifier.
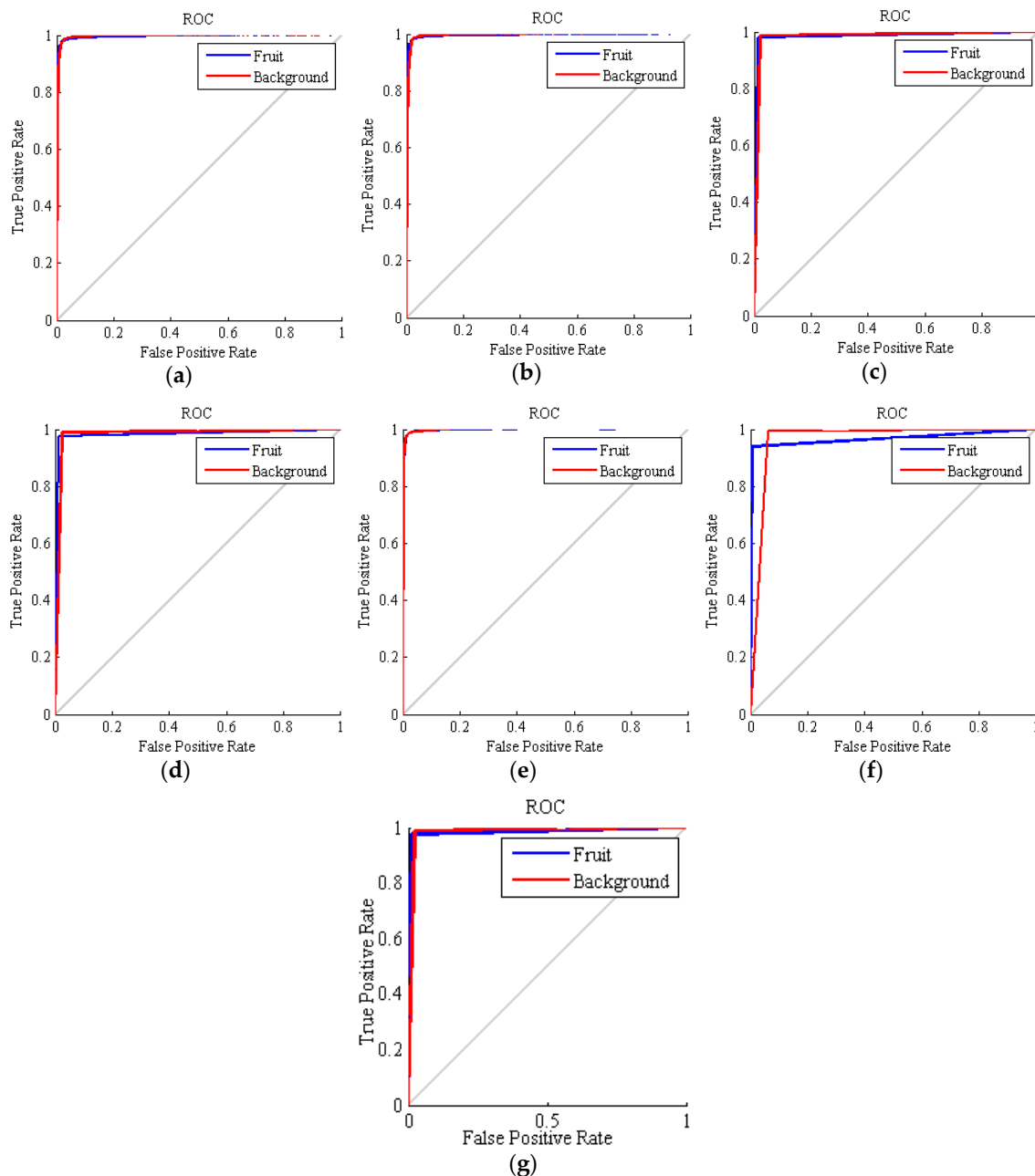

Figure 4. Receiver operating characteristic (ROC) curves obtained by the classifiers for the 275 executions. (**a**) hybrid ANN-ICA; (**b**) hybrid ANN-HS; (**c**) support vector machines (SVM); (**d**) *k* nearest neighbors (kNN); (**e**) linear discriminant analysis (LDA); (**f**) voting method with LDA; (**g**) voting method without LDA.

Finally, Table 7 contains the six performance evaluation criteria obtained in the experiments. As can be seen, except for the LDA classifier and the original voting method, all the values were above 96%, indicating high performance in the fruit/background classification in general. The high recall value of LDA is due to the fact that it tends to overclassify in the background class; so this cannot be

considered a good result, when considering its low specificity. Since LDA performs a simple linear separation of the samples, its poor results indicate that the samples cannot be classified with a linear decision boundary. This fact seriously affects the results of the voting method with LDA. The two ANN methods also present a certain tendency to overclassify in the background class, which was evident from their low precision. However, the majority voting method without LDA is able to reduce this bias; it achieves high results for recall and precision, which are transformed in an F_measure of 98.95%.

Table 7. Performance evaluation criteria of the classifiers and the 275 executions.

Classification Method	Recall (%)	Specificity (%)	Precision (%)	F_Measure (%)	AUC (Mean ± Std. dev.)	Accuracy (Mean % ± Std. dev.)
ANN-ICA	98.94	98.38	96.63	97.28	0.9976 ± 0.0027	98.28 ± 0.4719
ANN-HS	98.09	98.31	96.50	97.27	0.9974 ± 0.0036	98.24 ± 0.3266
SVM	97.38	99.03	98.00	97.69	0.9826 ± 0.0013	98.49 ± 0.0962
kNN	97.71	98.87	97.68	97.70	0.9829 ± 0.0014	98.50 ± 0.1237
LDA	99.29	95.43	90.14	94.49	0.9983 ± 0.0002	96.57 ± 0.1386
Voting w. LDA	98.95	97.11	93.89	96.36	0.9670 ± 0.0313	97.68 ± 1.9820
Voting wo. LDA	98.92	97.90	98.98	98.95	0.9830 ± 0.0013	98.59 ± 0.0940

As a concluding remark, it is evident that the ensemble method without LDA is able to improve significantly the accuracy of the constituent methods. It exhibits a better overall accuracy and greatly reduces the over-classification tendency of the ANN methods. The standard deviation of the accuracy also indicates a great consistency in these good results, as previously mentioned. The LDA is the worst of all the methods, and the accuracy obtained was only 96.57%. This fact justifies that LDA should be removed from the ensemble, since it only worsens the final result, producing also a standard deviation nearly ±2.

Another interesting aspect to consider is that the application of the proposed ensemble method without LDA requires the execution of all four classifiers that compose it. It is still unclear whether the computational cost of the ensemble and the majority voting rule is or is not beneficial, taking into account the small increment in the accuracy of only 0.1%. Alternatively, the SVM classifier also achieves good results, with low standard deviation, and is computationally more efficient. This is another advantage with respect to the kNN, which requires storing and processing all the training samples in the classification, while the SVM only stores the selected support vectors. In any case, the bottleneck of the process is most probably the color space conversion; since all the methods rely on L*a*b*, the application of the ensemble method could be justified. More experiments concerning the computational efficiency, which are outside the scope of this paper, would be needed.

3.3. Final Proposed Method and Comparison with the State-of-the-Art

After examining the experimental results, the final algorithm for the segmentation of plum fruits from the background in the videos of plum orchards is illustrated in Figure 5. As previously explained, this method is based on the ensemble of the four classifiers excluding LDA, which is the most accurate and reliable method, and with a good balance in the accuracy per class. This scheme assumes that the basic classifiers have been trained, so it only contains the steps for the classification of a new frame.
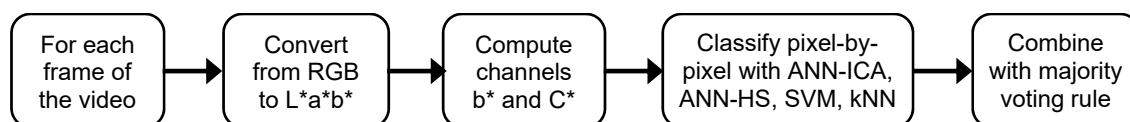

Figure 5. Diagram of the proposed algorithm for plum fruits segmentation. A sample video of the results can be accessed in: https://youtu.be/PoSrkz5PSJI.

Table 8 contains a comparison of the correct detection rate of the proposed method with other methods presented by different researchers in the literature. According to this table, the proposed

method is more accurate than those proposed by the other works. However, this conclusion should be taken with caution. A direct comparison of the results is not possible, because the application domain is different, the input images are also different, and in some cases the problem is not exactly the same.

Table 8. Comparison of the accuracy of the segmentation algorithms reported in different works.

Method	Number of Samples	Correct Classification Rate (%)
Proposed in this study	14,804	98.59
Sabzi et al. (2018) [9]	210,752	96.80
Aquino et al. (2017) [39]	152	95.72
Tang et al. (2016) [40]	100	92.5
Zhao et al. (2016) [11]	68	83

4. Conclusions

This study has addressed the segmentation of plum fruits in videos recorded under natural lighting conditions, for application in remote sensing problems. The method works at a pixel level, using color features. Five basic classifiers have been analyzed, and the combination of four of them using a majority voting rule. Two methods are hybrids of ANN and metaheuristics (ANN-ICA and ANN-HS), two methods are non-parametric sample-based methods (SVM and kNN), and the last method is based on linear discriminant analysis (LDA), which was not included in the ensemble due to its poor results.

Pixel-by-pixel analysis is highly efficient, because it measures the color properties of each pixel. The selection of the most effective color features has shown that a reduced number of parameters can be useful to obtain an accurate result. In our case, channel b* of L*a*b* and the color purity, C*, from L*C*h were found to be the optimal color properties.

The single methods were able to achieve very accurate results. SVM and kNN are the optimal selections, being able to produce 98.49% and 98.50% correct classification. The methods based on ANN also obtained good results, above 98.2%, but suffered from a certain imbalance in the accuracies per class. All of these methods were outperformed by the ensemble method without LDA, which removes this imbalance and achieves an overall accuracy of 98.59%.

An interesting aspect to consider in future research is the trade-off between accuracy and computational efficiency. While the proposed ensemble technique is able to improve the results of the constituent methods, it also accumulates the cost of all of them. The particular characteristics of the hardware, the minimum image resolution, and the required speed should be taken into account. Another promising future line of work is the comparison with other ensemble techniques, such as random forests, that has been successfully applied to similar problems, and the inclusion of other basic methods, such as logistic regression and feature weighting. Post-processing techniques could also be applied to improve the results of the pixel-by-pixel segmentation.

Author Contributions: Conceptualization, R.P. and S.S.; Methodology, R.P., S.S., M.H.-H., J.L.H.-H. and G.G.-M.; Software, S.S.; Validation, R.P., S.S., M.H.-H., D.K. and J.M.M.-M.; Formal Analysis, R.P., S.S. and G.G.-M.; Investigation, R.P., S.S., M.H.-H., J.L.H.-H., D.K., G.G.-M. and J.M.M.-M.; Resources, R.P. and S.S.; Writing—Original Draft Preparation, R.P. and S.S.; Writing—Review & Editing, M.H.-H., J.L.H.-H., D.K., G.G.-M. and J.M.M.-M.; Supervision, R.P.; Project Administration, R.P., G.G.-M. and J.M.M.-M.; Funding Acquisition, M.H.-H., J.L.H.-H., G.G.-M., D.K. and J.M.M.-M.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Atzberger, C. Advances in remote sensing of agriculture: Context description, existing operational monitoring systems and major information needs. *Remote Sens.* **2013**, *5*, 949–981. [CrossRef]
2. Jiménez, A.R.; Ceres, R.; Pons, J.L. A survey of computer vision methods for locating fruit on trees. *Trans. Am. Soc. Agric. Eng.* **2000**, *43*, 1911–1920. [CrossRef]
3. Yu, Y.; Zhang, K.; Yang, L.; Zhang, D. Fruit detection for strawberry harvesting robot in non-structural environment based on Mask-RCNN. *Comput. Electron. Agric.* **2019**, *163*, 104846. [CrossRef]
4. Fu, L.; Tola, E.; Al-Mallahi, A.; Li, R.; Cui, Y. A novel image processing algorithm to separate linearly clustered kiwifruits. *Biosyst. Eng.* **2019**, *183*, 184–195. [CrossRef]
5. Li, Z.; Miao, F.; Yang, Z.; Chai, P.; Yang, S. Factors affecting human hand grasp type in tomato fruit-picking: A statistical investigation for ergonomic development of harvesting robot. *Comput. Electron. Agric.* **2019**, *157*, 90–97. [CrossRef]
6. Wang, C.; Luo, T.; Zhao, L.; Tang, Y.; Zou, X. Window Zooming–Based Localization Algorithm of Fruit and Vegetable for Harvesting Robot. *IEEE Access* **2019**, *7*, 103639–103649. [CrossRef]
7. Wang, Y.; Zhu, X.; Ji, C. Machine Vision Based Cotton Recognition for Cotton Harvesting Robot. In *Computer And Computing Technologies in Agriculture, Volume II*; Li, D., Ed.; CCTA 2007. The International Federation for Information Processing; Springer: Boston, MA, USA, 2008; Volumn 259.
8. Sun, L.; Meng, X.; Xu, J.; Tian, Y. An image segmentation method using an active contour model based on improved SPF and LIF. *Appl. Sci.* **2018**, *8*, 2576. [CrossRef]
9. Sabzi, S.; Abbaspour-Gilandeh, Y.; García-Mateos, G.; Ruiz-Canales, A.; Molina-Martínez, J.M. Segmentation of apples in aerial images under sixteen different lighting conditions using color and texture for optimal irrigation. *Water* **2018**, *10*, 1634. [CrossRef]
10. Miao, R.H.; Tang, J.L.; Chen, X.Q. Classification of farmland images based on color features. *J. Vis. Commun. Image Represent.* **2015**, *29*, 138–146. [CrossRef]
11. Zhao, C.; Lee, W.S.; He, D. Immature green citrus detection based on colour feature and sum of absolute transformed difference (SATD) using colour images in the citrus grove. *Comput. Electron. Agric.* **2016**, *124*, 243–253. [CrossRef]
12. Wang, C.; Tang, Y.; Zou, X.; SiTu, W.; Feng, W. A robust fruit image segmentation algorithm against varying illumination for vision system of fruit harvesting robot. *Optik* **2017**, *131*, 626–631. [CrossRef]
13. Meylan, L.; Süsstrunk, S. Color image enhancement using a Retinex-based adaptive filter. In Proceedings of the CGIV 2004-Second European Conference on Color in Graphics, Imaging, and Vision and Sixth International Symposium on Multispectral Color Science, Aachen, Germany, 5–6 April 2004.
14. Sabzi, S.; Abbaspour-Gilandeh, Y.; Hernandez-Hernandez, J.L.; Azadshahraki, F.; Karimzadeh, R. The use of the combination of texture, color and intensity transformation features for segmentation in the outdoors with emphasis on video processing. *Agriculture* **2019**, *9*, 104. [CrossRef]
15. Xu, W.; Chen, H.; Su, Q.; Ji, C.; Xu, W.; Memon, M.S.; Zhou, J. Shadow detection and removal in apple image segmentation under natural light conditions using an ultrametric contour map. *Biosyst. Eng.* **2019**, *184*, 142–154. [CrossRef]
16. Sengupta, S.; Lee, W.S. Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions. *Biosyst. Eng.* **2014**, *117*, 51–61. [CrossRef]
17. Luo, L.; Tang, Y.; Zou, X.; Wang, C.; Zhang, P.; Feng, W. Robust Grape Cluster Detection in a Vineyard by Combining the AdaBoost Framework and Multiple Color Components. *Sensors* **2016**, *16*, 2098. [CrossRef] [PubMed]
18. Xiong, J.; Liu, Z.; Lin, R.; Bu, R.; He, Z.; Yang, Z.; Liang, C. Green grape detection and picking-point calculation in a night-time natural environment using a charge-coupled device (CCD) vision sensor with artificial illumination. *Sensors* **2018**, *18*, 969. [CrossRef] [PubMed]
19. Wang, C.; Tang, Y.; Zou, X.; Luo, L.; Chen, X. Recognition and matching of clustered mature litchi fruits using binocular charge-coupled device (CCD) color cameras. *Sensors* **2017**, *17*, 2564. [CrossRef]
20. Wang, C.; Zou, X.; Tang, Y.; Luo, L.; Feng, W. Localisation of litchi in an unstructured environment using binocular stereo vision. *Biosyst. Eng.* **2016**, *145*, 39–51. [CrossRef]

21. Hernández-Hernández, J.L.; García-Mateos, G.; González-Esquiva, J.M.; Escarabajal-Henarejos, D.; Ruiz-Canales, A.; Molina-Martínez, J.M. Optimal color space selection method for plant/soil segmentation in agriculture. *Comput. Electron. Agric.* **2016**, *122*, 124–132. [CrossRef]

22. Clerici, M.T.P.S.; Kallmann, C.; Gaspi, F.O.G.; Morgano, M.A.; Martinez-Bustos, F.; Chang, Y.K. Physical, chemical and technological characteristics of Solanum lycocarpum A. St.-HILL (Solanaceae) fruit flour and starch. *Food Res. Int.* **2011**, *44*, 2143–2150. [CrossRef]

23. Goldberg, D.E.; Holland, J.H. Genetic Algorithms and Machine Learning. *Mach. Learn.* **1988**, *3*, 95–99. [CrossRef]

24. Goldberg, D. Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex. Syst.* **1991**, *5*, 139–167.

25. Mitchell, M. *An Introduction to Genetic Algorithms (Complex. Adaptive Systems)*; MIT Press: Cambridge, MA, USA, 1998.

26. Montana, D.J.; Davis, L. Training Feedforward Neural Networks Using Genetic Algorithms. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, Detroit, MI, USA, 20–25 August 1989.

27. Ali, M.Z.; Awad, N.H.; Suganthan, P.N.; Duwairi, R.M.; Reynolds, R.G. A novel hybrid Cultural Algorithms framework with trajectory-based search for global numerical optimization. *Inf. Sci.* **2016**, *334*, 219–249. [CrossRef]

28. Atashpaz-Gargari, E.; Lucas, C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, CEC, Singapore, 25–28 September 2007.

29. Sabzi, S.; Abbaspour-Gilandeh, Y.; García-Mateos, G. A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms. *Comput. Ind.* **2018**, *98*, 80–89. [CrossRef]

30. Lee, K.S.; Geem, Z.W. A new meta-heuristic algorithm for continuous engineering optimization: Harmony search theory and practice. *Comput. Methods Appl. Mech. Eng.* **2005**, *194*, 3902–3933. [CrossRef]

31. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]

32. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **1992**, *43*, 175–185.

33. Fix, E.; Hodges, J.L., Jr. *Discriminatory Analysis-Nonparametric Discrimination: Consistency Properties*; University of California: Berkeley, CA, USA, 1951.

34. Hart, P. The condensed nearest neighbor rule. *IEEE Trans. Inf. Theory* **1968**, *18*, 515–516. [CrossRef]

35. Kemsley, E.K. Discriminant analysis of high-dimensional data: A comparison of principal components analysis and partial least squares data reduction methods. *Chemom. Intell. Lab. Syst.* **1996**, *33*, 47–61. [CrossRef]

36. Carrillo-De-Gea, J.M.; García-Mateos, G. Detection of normality/pathology on chest radiographs using LBP. In Proceedings of the BIOINFORMATICS 2010-Proceedings of the 1st International Conference on Bioinformatics, Valencia, Spain, 20–23 January 2010.

37. Guijarro, M.; Riomoros, I.; Pajares, G.; Zitinski, P. Discrete wavelets transform for improving greenness image segmentation in agricultural images. *Comput. Electron. Agric.* **2015**, *118*, 396–407. [CrossRef]

38. Hernández-Hernández, J.L.; Ruiz-Hernández, J.; García-Mateos, G.; González-Esquiva, J.M.; Ruiz-Canales, A.; Molina-Martínez, J.M. A new portable application for automatic segmentation of plants in agriculture. *Agric. Water Manag.* **2017**, *183*, 146–157. [CrossRef]

39. Aquino, A.; Diago, M.P.; Millán, B.; Tardáguila, J. A new methodology for estimating the grapevine-berry number per cluster using image analysis. *Biosyst. Eng.* **2017**, *156*, 80–95. [CrossRef]

40. Tang, J.L.; Chen, X.Q.; Miao, R.H.; Wang, D. Weed detection using image processing under different illumination for site-specific areas spraying. *Comput. Electron. Agric.* **2016**, *122*, 103–111. [CrossRef]