

AUTOMATED TOMATO MATURITY GRADING USING ABC-TRAINED ARTIFICIAL NEURAL NETWORKS

Harvey Jake G. Opeña¹, John Paul T. Yusiong²

^{1,2}University of the Philippines Visayas Tacloban College, Magsaysay Boulevard, Tacloban City, Leyte, Philippines

Email: harveyjakeopena@hotmail.com¹, jpyusiong@gmail.com²

ABSTRACT

Tomato maturity grading is quite essential in commercial farms that produce large quantities of tomatoes, and human graders usually perform tomato maturity grading. This task is carried out by matching the surface color of tomatoes to the United States Department of Agriculture's (USDA) tomato color chart which shows six maturity stages: Green, Breakers, Turning, Pink, Light Red, and Red. However, due to some uncontrollable factors, manual classification involving human graders is prone to misclassification. Thus, this paper introduces an automated tomato classification system that uses Artificial Neural Network (ANN) classifier trained using the Artificial Bee Colony (ABC) algorithm. To effectively classify tomatoes, the researchers combined five color features (Red, Green, Red-Green, Hue, a^) from three color models (RGB, HSI, CIE $L^*a^*b^*$). These features are the inputs to the ANN classifier. Experiment results show that the ABC-trained ANN classifiers performed well in tomato classification and achieved high accuracy rate. Also, results show that combining the color features from different color models produce better accuracy rate than using color features from a single color model. With these results, an automated tomato classification system using an ABC-trained ANN classifier can be used to replace the manual classification procedure as it minimizes the chances of misclassification.*

Keywords: *Tomato Grading, Artificial Bee Colony Algorithm, ABC, Artificial Neural Networks, Image Processing*

1.0 INTRODUCTION

Tomato, *Lycopersicon esculentum*, is one of the most important and popular fruits in the world that can be consumed in several ways, for instance, as an ingredient in many dishes, sauces, and drinks. It has several varieties, which differ in shape, size, and color. Tomato maturity is commonly determined by its surface color because it is the most observable characteristic in estimating tomato maturity or ripeness, and most consumers buy tomatoes based only on the surface color. Hence, the color of the tomato is a major factor in determining the appropriate time to market the product since it influences consumers' preference [1, 2, 3].

Fruits such as tomatoes are sorted and classified into several classes before being transported to commercial establishments for public consumption. This task is manually done using human labor. Farmers (human graders) in most commercial farms classify tomato maturity by comparing its surface color to the USDA color chart. The United States Department of Agriculture designed a color chart as a standard for tomato classification, and according to this chart, tomato ripeness is divided into six maturity stages: Green stage, Breakers stage, Turning stage, Pink stage, Light Red stage, and Red stage [1, 4, 5, 7, 8, 17].

However, the traditional way of tomato ripeness classification is very tiring and prone to error. Human graders can suffer from visual fatigue and stress that may affect the quality of grading. Also, human grading may vary from one person to another [3]. Because of these factors, misclassification can always happen with this kind of system. Thus, an automated classification of tomatoes using computer vision and image processing techniques can increase the accuracy and improve the classification performance.

Automated tomato classification is the process of sorting tomatoes into different classes using machine vision and image processing techniques, which is said to be highly reliable and accurate [1]. Automated tomato classification involves two phases: the feature extraction phase, and the classification phase, that is, tomato images are taken as inputs and from these images certain features are extracted, analyzed and used in categorizing the images into different classes.

Several research studies dealt with the tomato classification problem, and these research studies can be classified into two groups; research studies that focus on classifying tomatoes in terms of their level of maturity only, and research studies that focus on classifying tomatoes in terms of quality and maturity. Researchers that give emphasis to classify tomatoes in terms of maturity consider color as the only crucial factor in estimating tomato ripeness. They select a color model from several color models such as RGB, HSI, HSV, CIE L*a*b*, and YCbCr to generate a feature vector for each tomato image. On the other hand, researchers that give emphasis to classify tomatoes in terms of quality and maturity use a multi-valued representation that includes several features such as color, texture, size, and shape. For instance, in assessing tomato ripeness, El-Bendary et al. [8], Ukirade [9] and Ceh-Varela et al. [10] chose the HSV color model for the feature extraction phase, but they differed in the machine learning techniques and the number of ripeness classes. Kassem et al. [11] used the CIE L*a*b* color model to generate the feature vectors for the tomato images and trained the artificial neural network classifier using the backpropagation algorithm to classify the tomato images into three ripeness classes. Goel et al. [7] proposed a Fuzzy Rule-Based Classification (FRBCS) method to classify tomato images into the six maturity stages using the RGB color model. Gejima et al. [1] compared the effectiveness of the two color models, RGB, and CIE L*a*b*, for tomato maturity grading. Baek et al. [4] introduced a rapid tomato quality sorting in real-time processing to estimate the quality of tomatoes with respect to their color or level of maturity. In evaluating the level of maturity, RGB images were taken and converted into CIE L*a*b* color model, and three maturity stages were defined: immature, half-ripe, and ripe. Also, Nagata et al. [6] used three color models, RGB, HSV, and CIE L*a*b*, to estimate the maturity level of tomatoes in five ripeness classes. Likewise, Wang et al. [5] applied a vision-based judgment in tomato maturity classification based on five maturity stages: Breakers, Turning, Pink, Light Red, and Red involving near-infrared and RGB/HSI tomato images. Zhang and McCarthy [12] investigated the potential of magnetic resonance imaging (MRI) in classifying tomatoes into different maturity stages. In contrast, aside from color, Pounroja et al. [2], Mhaski et al. [13], and Fang et al. [14] included other features such as texture, size, and shape in evaluating tomatoes in terms of quality and maturity. However, they differed in the machine learning techniques used to classify the tomato images into the different ripeness classes.

Another interesting area of research is to employ the Artificial Neural Network (ANN) for image classification since its success greatly depends on the algorithm used in training ANNs [15]. ANNs are simple computational models that can handle different types of data. These computational models have been applied in the agricultural sector and the food industry as tools for food safety and quality analyses [16, 17, 18]. Goyal [18] provides a comprehensive review on the applications of the artificial neural networks in fruits, and this study shows the importance of ANNs in the agricultural sector and the food industry. Moreover, artificial bee colony (ABC) algorithm developed by Dervis Karaboga [19] has shown to be an effective trainer of ANNs because this algorithm performs both global and local search in each iteration [16, 20, 21].

The main objective of this study is to classify the tomatoes in terms of their level of maturity using ABC-trained ANN classifier. The ANN is selected as the classifier since it is extensively applied to pattern classification problems [16]. In the feature extraction phase, several color models will be used to generate the feature vectors of the tomato images while in the classification phase, an artificial neural network will be trained using the artificial bee colony algorithm to classify each tomato image into one of the six ripeness classes.

The remainder of the paper is organized as follows: Section 2 discusses image processing while Section 3 provides an overview of the Artificial Neural Network. The Artificial Bee Colony (ABC) algorithm is discussed in Section 4, the proposed approach, and the experiments and results are presented in Sections 5 and 6, respectively. Lastly, conclusions are presented in Section 7.

2.0 IMAGE PROCESSING

Color is one of the most commonly used visual features, and it provides vital information that can aid in estimating the maturity and quality of fruits. That is, a color feature is an important criterion in fruit identification and fruit quality, and is a good indicator of ripeness. A grading system for fruits is dependent on the color extracted from the image. Hence, color features extraction is a key component that affects the efficiency and effectiveness of this grading system [7, 22, 23].

A color space, also known as a color model, is a technique of specifying a color numerically or mathematically. Many color models have been proposed for different applications [23, 24, 25] because the use of color models in

image processing provides an interesting mechanism for object recognition and extraction. In tomato classification, RGB, HSV, and CIE $L^*a^*b^*$ are the most commonly used color models [1, 2, 4, 5, 6, 8, 9, 10, 11, 13, 14].

Image processing techniques are used to preprocess an image and transform it into a form appropriate for further examination. Hence, designing the image processing stages is crucial in achieving satisfactory results [26]. Various image processing techniques exist, however, only the following techniques are used in this research, namely, resizing, background removal, grayscaleing, binarization and cropping.

3.0 ARTIFICIAL NEURAL NETWORKS

An artificial neural network is an abstract representation of the biological nervous system which consists of a large set of interconnected neurons. It is a nature-inspired computational paradigm that has interesting attributes like the ability to adapt, learn and generalize. Moreover, because of its massively parallel processing, ANNs are effective in solving many complex problems and have been extensively used in many areas such as in image segmentation and object classification problems [15, 16, 17, 20, 21, 23, 27, 28, 29].

Also, artificial neural networks are computational models that have the ability to learn through adjustments of internal weight parameters as determined by the training algorithm in response to some training examples. The Multilayer Perceptron (MLP) is the most popular artificial neural network model used in many types of research. It is a feed-forward neural network that consists of several layers of neurons, and every neuron in each layer is connected to all neurons in the succeeding layers. A three-layer feedforward neural network is the most widely used MLP. For ANN to learn a certain task, it has to undergo training. The objective of the training is to search for the optimal set of connection weights that will cause the output of the artificial neural network to match the actual target values. Finding the optimal set of connection weights requires a training algorithm that will be able to adjust the connection weights of ANN so that the desired output from the network given a specific set of inputs can be obtained. The process of adjusting the weights is called learning [23, 28]. Hence, choosing an appropriate algorithm for a given problem greatly influences the success of the training process.

4.0 ARTIFICIAL BEE COLONY (ABC) ALGORITHM

The artificial bee colony (ABC) algorithm is popular because of its ability to perform both global and local search in each iteration, which significantly increases the probability of finding the optimal solution [16]. It is a swarm intelligence-based optimization algorithm proposed by Karaboga in 2005 [19]. This algorithm mimics the foraging behavior of the honey bee swarm. Bees communicate through waggle dance, and this dance enables bees to convey the distance and direction of the food source, and the quality of the nectar found. Hence, the formation of tuned collective knowledge is the result of this exchange of information among bees [20, 21, 30].

In ABC algorithm, a food source, x_i , corresponds to a possible solution to the problem. The nectar amount of a food source denotes the quality of the solution represented by the fitness value. The total number of food sources is equal to the number of employed bees, BN , which is also equal to the number of onlooker bees, NW . Also, the total number of employed bees is equal to the number of solutions in the population, SN . Typically, there is one employed bee for every food source. However, three groups of bees are used to find the optimal solution, and these are the employed bees, onlooker bees, and scout bees. These bees perform different tasks in the colony. Employed bees go to their food source, and then return to the colony and do the waggle dance. The employed bee whose food source has been abandoned becomes a scout bee and begins searching for a new food source. Onlooker bees observe the waggle dance and select food sources depending on the outcome of the waggle dance [20, 21].

The general structure of the ABC algorithm is as follows [19, 30]:

```

Begin
[01] Initialize Population
[02] Evaluate Population
[03] Set cycle=1
[04] Repeat
[05] Produce new solution  $v_i$  in the neighborhood of  $x_i$ 

```

```

[06] Evaluate Population
[07] Apply Greedy Selection between  $x_i$  and  $v_i$ 
[08] Calculate the probability values  $P_i$ 
[09] Normalize  $P_i$  values into [0,1]
[10] Produce new solutions  $v_i$  from  $x_i$  based on the value of  $P_i$ 
[11] Evaluate Population
[12] Apply Greedy Selection between  $x_i$  and  $v_i$ 
[13] If abandoned solution exists
[14]     Then replace abandoned solution with new solution
[15] Memorize best solution
[16] cycle=cycle+1
[17] Until (termination condition is met)
End

```

In the employed bee phase (lines 5-7), bees try to find new food sources using the equation (1):

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

The symbol ϕ_{ij} is a random number between [-1,1] and x_k is a randomly selected food source in the neighborhood of x_i . After producing the food source v_i , its fitness is compared with the fitness of the current food source x_i . A greedy selection is performed between the two food sources. The fitness function is shown in equation (2).

$$\text{fit}_i = \frac{1}{1+\text{f}_i} \quad (2)$$

In the onlooker bee phase (lines 8-12), the probability of a solution to be exploited by the onlooker bees depends on the probability function P_i , where:

$$P_i = \frac{\text{fit}_i(x_{ij})}{\sum_{i=1}^{SN} \text{fit}_i(x_{ij})} \quad (3)$$

The probability is proportional to the nectar amount of the food source. The higher the fitness value of a solution means the higher is its probability to be chosen by the onlooker bees. Once a food source is chosen, the number of onlooker bees, N_i , to be assigned to that specific food source is determined by:

$$N_i = P_i * NW \quad (4)$$

Lastly, in the scout bee phase (lines 13-14), scouting is performed, which is the process of replacing the worst solutions in the population with randomly generated new solutions. Worst solutions are those with the lowest fitness values and for each worst solution, a random solution is generated and then a greedy selection is performed between the two solutions. The number of worst solutions is equal to the number of scout bees which is 10% of the total number of employed bees. Replacement is necessary for the ABC algorithm because it allows solutions of poor quality to be replaced with better solutions.

5.0 AUTOMATED TOMATO MATURITY GRADING USING ABC-TRAINED ANN

Fig. 1 shows the proposed framework for the automated tomato maturity grading using an ABC-trained ANN.

In this study, the dataset consists of colored images of tomatoes with $M \times N$ resolution as shown in Fig. 2. Five color features will be extracted from each image in the dataset, and these feature vectors are the inputs to the artificial neural network classifier. Three of the five features are from the RGB color model, and these are the Red (R), Green (G), and Red minus Green (R-G) components. The two other features are Hue (H) from the HSI color model and a^* from the CIE $L^*a^*b^*$ color model.

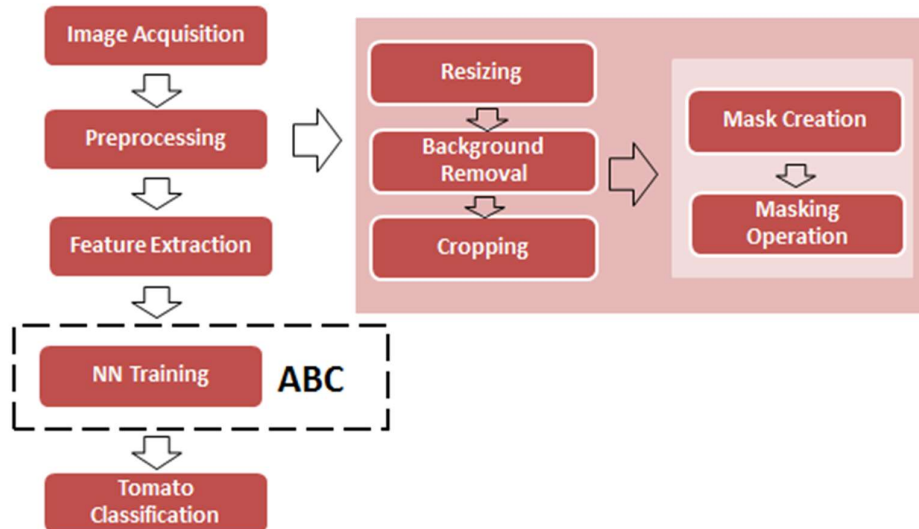


Fig. 1: The proposed framework for tomato maturity grading using ABC-trained ANNs



Fig. 2: Colored images of tomatoes

5.1 Image acquisition

The tomato images used in this study were acquired using a Sony digital camera. These tomatoes came from the local market. In the image acquisition step, the tomatoes were placed on top of a plain white paper, which served as its background. The tomato images acquired are of high resolution and contain the background area. Hence, there is a need to preprocess these images before proceeding to the feature extraction phase.

5.2 Image preprocessing

Preprocessing the tomato images is an important step to ensure that the extracted values of the color features are accurate. The tomato preprocessing techniques used in this study are the following: resizing, background removal, and cropping as shown in Fig. 1.

5.2.1 Image resizing

The first image preprocessing technique is resizing. In this study, tomato images undergo image resizing twice. The first instance happens before performing background removal wherein a tomato image with high resolution is resized to an image with a dimension of 200 x 200 pixels. Resizing the images makes the background removal process much faster. The second instance occurs after image cropping. After the second resizing, tomato images have a size of 64 x 64 pixels and these images are now ready for feature extraction.

5.2.2 Background Removal

There are four steps in background removal and these are:

1. Convert to blue channel
2. Convert to grayscale
3. Create binary mask
4. Remove background by applying the binary mask

The first step in background removal involves the conversion of an RGB image to a blue channel image as illustrated in Fig. 3. Since the blue channel is just a subset of the RGB model, converting an RGB image to a blue channel image is accomplished by setting the values of red and green components to zero. In effect, only the blue component retains its value as shown in Fig. 4.

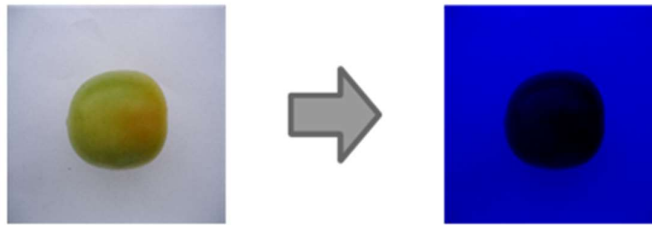


Fig. 3: An RGB image converted to blue channel image



Fig. 4: Blue channel image conversion

The second step in background removal is grayscaling. After converting an RGB image to a blue channel image, the blue channel image will be converted to a grayscale image using the averaging method as shown in equation (5).

$$\text{Gray} = (\text{Red} + \text{Green} + \text{Blue}) / 3 \quad (5)$$

As seen in Fig. 5, a pixel in a grayscale image contains identical values for the three color components. In effect, a pixel in a grayscale image can be represented as a single value. Fig. 6 illustrates a blue channel image converted to grayscale image.



Fig. 5: Grayscale a blue channel image

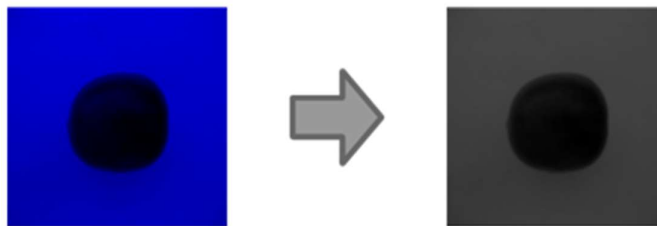


Fig. 6: A blue channel image converted to grayscale image

The third step in background removal, which is after grayscaling, is the creation of a binary mask. In this study, the Otsu's method [31] is used to create the binary mask. The Otsu's method developed by Nobuyuki Otsu is a binarization technique based on histograms. Fig. 7 shows the binary mask that was created after performing the Otsu's method.

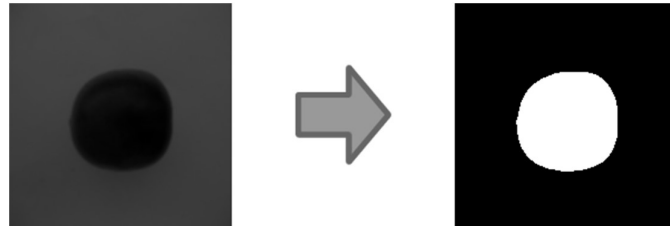


Fig. 7: A binary mask creation

The final step in background removal is the masking operation or applying the binary mask to the colored image. By multiplying the colored image and the binary mask, the region of interest is extracted from the colored image, as shown in Fig. 8(c). If the pixel value of the binary mask is 1, then the product is the same as the original pixel Fig. 8(a). However, if the pixel value of the binary mask is 0, then the product is the same as the binary mask pixel Fig. 8(b).

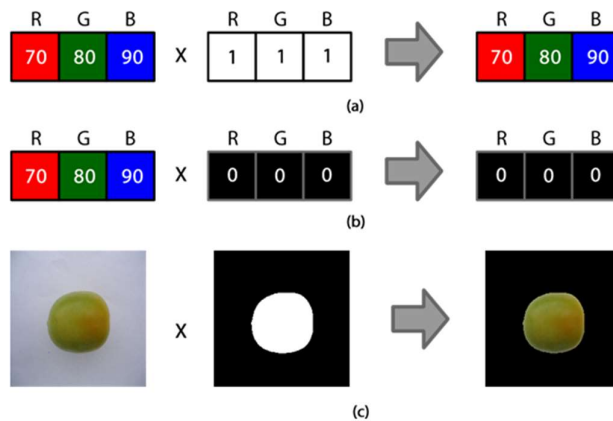


Fig. 8: The masking operation

5.2.3 Image Cropping

Background pixels have no use in the feature extraction phase and if these pixels are not removed, the processing time increases. Each tomato image is cropped to reduce these background pixels. In Fig. 9(a), these background pixels are represented by black color. Fig. 9(b) shows the cropped image.

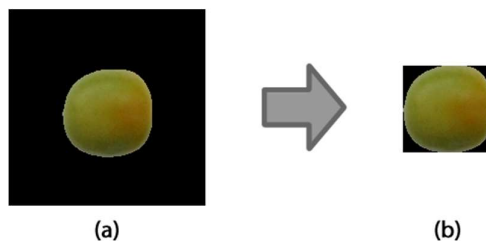


Fig. 9: Image cropping

5.3 Feature Extraction

The selection of the features is one of the major factors that can affect the success of an artificial neural network. In this paper, color features will be used to determine tomato maturity. In this study, three color models will be

utilized, and these are the RGB, HSI and CIE L*a*b* color models. Five color features will be extracted from the 64 x 64 cropped image and these are the average Red (RGB), average Green (RGB), average of Red-Green difference (RGB), average Hue (HSI), and average a* (L*a*b*). Extracting the RGB features is fairly straightforward since the tomato image is in RGB format. However, to extract the hue and a* features there is a need to transform the RGB format into HSI and CIE L*a*b*, respectively [23, 24, 25]. Fig. 10 shows the color features vector while Fig. 11 shows the corresponding color features vector of a tomato image. Each tomato image in the dataset will have its equivalent color features vector.

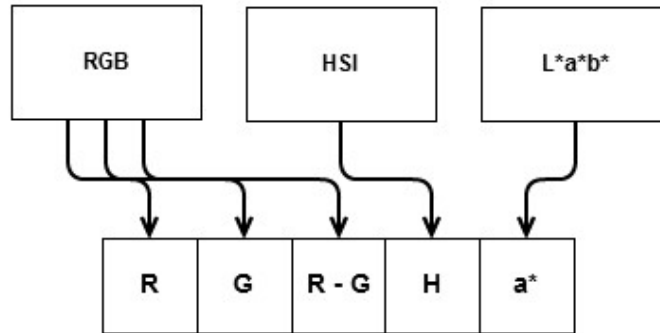


Fig. 10: Color features vector

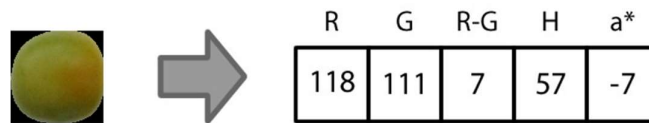


Fig. 11: A sample tomato color features vector

5.4 ABC-based Neural Network Training

The ABC algorithm finds the optimal solution (food source) by searching through the solution space. The ABC algorithm will train artificial neural networks (see Fig. 12) using the following parameters:

- Number of employed bees (BN) – corresponds to the initial number of solutions.
- Number of onlooker bees (NW) – corresponds to the maximum number of onlooker bees that will exploit a particular solution.
- Number of scout bees – 10% of the number of employed bees.
- Maximum cycle number (MCN) – maximum number of iterations.

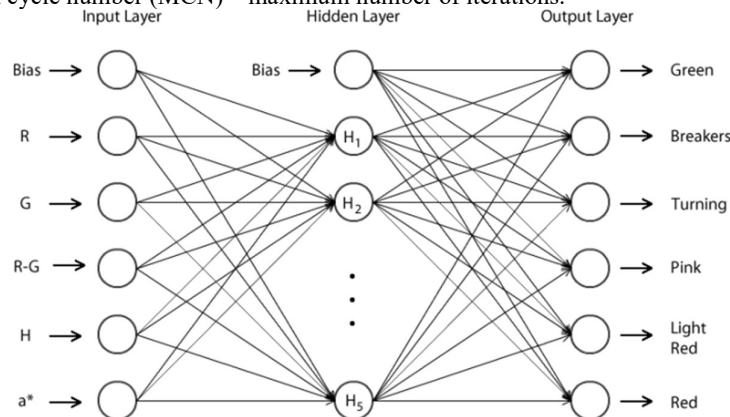


Fig. 12: The artificial neural network structure

A food source X_i is defined as a weights vector associated with an artificial neural network.

$$X_i = x_1 x_2 x_3 x_4 \dots x_D \quad (6)$$

In equation (6), x_i represents a connection weight and D is the dimension of the artificial neural network. The food sources are randomly initialized, and the number of food sources is equal to the number of employed bees. Fig. 13 shows two randomly generated food sources and the equivalent artificial neural network structure is shown in Fig. 14.

X_i	j	X_{i+1}	j
-0.1027805310361889	1	-0.00101851109402528	1
-0.49568075301100467	2	-0.9700464063418458	2
0.064730132432812	3	0.05544388141473133	3
-0.9242055378391367	4	-0.09297810349074154	4
⋮		⋮	
0.7713050059231521	66	-0.2875945501114362	66

(a) (b)

Fig. 13: Sample food sources

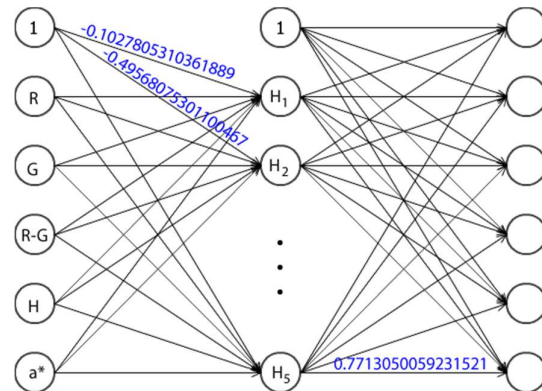


Fig. 14: ANN structure of the food source in Fig. 13(a)

Each food source in the initial population is evaluated using equation (2). Since each food source represents the connection weights of an artificial neural network, the mean squared error (MSE) as shown in equation (7) is used to assess the accuracy of the solution, which corresponds to the function f_i in equation (2).

$$MSE = \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M (T_{ij} - A_{ij})^2 \quad (7)$$

where N is the number of training patterns/examples, M is the number of output nodes, T_{ij} is the target value of pattern i in node j , and A_{ij} is the actual value of pattern i in node j .

Both the employed and onlooker bees perform the neighborhood search while the greedy selection is performed by the three groups of bees. In neighborhood search, the objective is to improve the current solution by looking for a neighboring solution using equation (1). Right after the neighborhood search, the greedy selection is performed between the current solution X_i and the newly generated solution V_i . If the fitness value of V_i is equal or higher than X_i , X_i is replaced with V_i in the memory. Otherwise, the solution X_i is retained in the memory. Fig. 15 shows X_i and V_i , after performing neighborhood search on X_i .

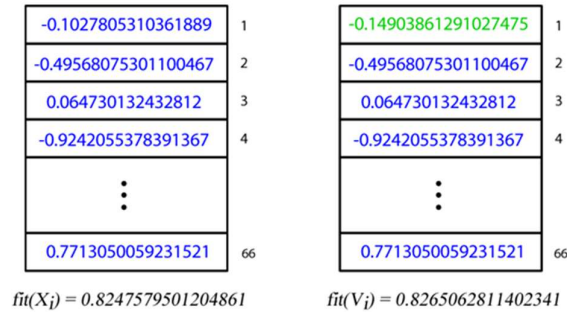


Fig 15: (a) X_i , (b) V_i , where $j = 1$

In the scout bee phase, a randomly generated food source is compared to the worst food source and then the greedy selection is performed between the two food sources. The number of worst solutions is equal to the number of scout bees which is 10% of the total number of employed bees. Fig. 16 shows a sample worst solution and the randomly generated solution.

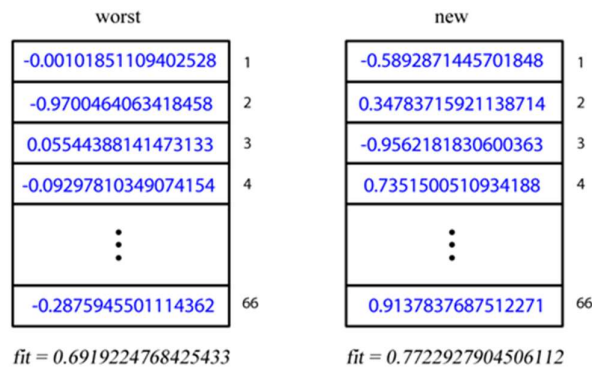


Fig. 16: Worst solution and a newly generated solution

After a certain number of cycles, a solution with the least mean squared error (MSE) is produced as a result of the training phase. This solution, which is the best artificial neural network obtained is used as a tomato classifier in the classification phase.

5.5 Tomato Classification

In the tomato classification phase, the features extracted from a tomato image will be used as inputs to the artificial neural network classifier. The actual values of the output neurons are in decimal formats between 0 and 1. However, it is necessary to set the output neuron with the highest value to 1 while the other output neurons are initialized to 0 in order to determine the maturity stage of a tomato. Fig. 17 shows an ANN classifier with a resulting output vector of [0, 1, 0, 0, 0, 0], which means that the tomato is in the Breakers stage.

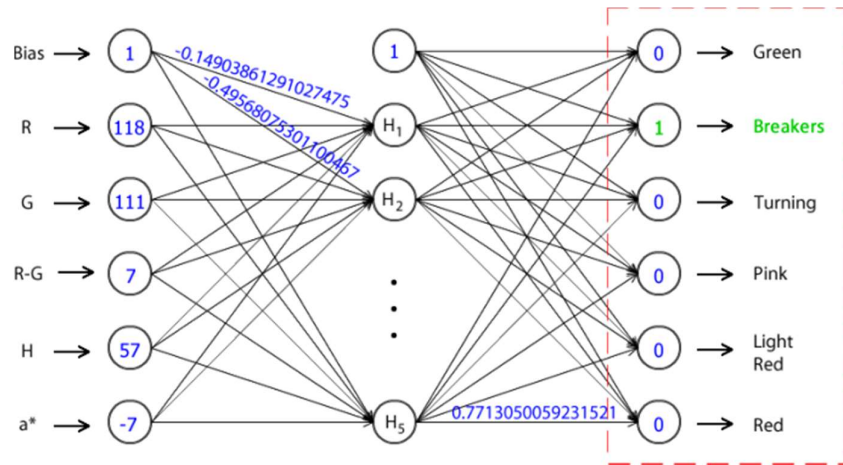


Fig. 17: An ANN classifier for tomato classification

6.0 EXPERIMENTS AND RESULTS

The dataset used in the experiments is composed of six hundred (600) images of tomatoes, one hundred (100) images per maturity stage. These images were acquired using a digital camera and were manually classified according to their maturity stages by experts in the local agriculturist office. The effectiveness and efficiency of the ABC-trained ANN classifiers will be evaluated using this dataset. In assessing the performance of the ABC-ANN classifiers for tomato maturity classification, the dataset was randomly divided into two sets. Seventy percent (70%) of it was used for training, and the remaining thirty percent (30%) was used for testing.

To assess the effectiveness of the ABC-trained ANN classifiers, the parameter values of the ABC algorithm and artificial neural networks are shown in Table 1 and Table 2, respectively. Each experimental setup was performed thirty (30) times to ensure the statistical acceptability of the results.

Table 1: ABC parameter values

Parameters	Default Values
Number of Employed Bees (BN)	15
Number of Onlooker Bees (NW)	15
Max Cycle (MCN)	750

Table 2: ANN parameter values

Parameters	Default Values
Number of Hidden Layer	1
Number of Hidden Nodes	5 (excluding the bias)
Dimension	66

For the first experiment, two setups were used to evaluate the ABC-trained ANN classifiers. The first setup referred to as the fixed setup, uses a fixed training set and a fixed test set for the entire thirty (30) runs. This setup means that the images in the training and test sets are the same in each run. For the second setup referred to as the random setup, the training set, and test set are re-arranged in each run. This setup means that the images in the training and test sets are different for each run. Moreover, in each run, out of the one hundred images per maturity stage, 70% is assigned to the training set while the remaining 30% is assigned to the test set, and this is done randomly.

It can be observed from Table 3 that the ABC-trained ANN classifiers performed well in both the fixed and random setups. Table 3 shows that regardless of how the datasets are arranged, the ABC-trained ANN classifiers can accurately classify the tomato images, which is a good indicator that the ABC-trained ANN classifiers have

good generalization capability. The performance of an ANN classifier is computed using equation (8), which is the accuracy rate of ANN classifier.

$$\text{Accuracy Rate} = \frac{\text{Number of correctly classified tomato images}}{\text{Total number of tomato images in a set}} \quad (8)$$

Table 3: Fixed vs. random setups

Statistical Parameters	Fixed		Random	
	Test Set	Training Set	Test Set	Training Set
Mean	98.19 %	98.73 %	97.81 %	98.75 %
Median	98.34 %	98.81 %	97.50 %	98.81 %
Maximum	100 %	99.52 %	99.44 %	99.52 %
Minimum	96.11 %	96.90 %	95.56 %	97.14 %
SD	1.04	0.62	1.08	0.52
Variance	1.08	0.38	1.17	0.27

For the fixed setup, another experiment was conducted where the number of color features used to classify the tomato images was varied. Table 4 shows the experimental setup using different color features.

Table 4: Varying the color features

Set	Color Feature	Color Model
A	a*	CIE L*a*b*
B	R-G, Hue	RGB, HSI
C	Red, Green, Blue	RGB
D	Red, Green, Red-Green, Hue, a*	RGB, HSI, CIE L*a*b*

Since each color feature corresponds to an input neuron, the number of input neurons and the dimension of the artificial neural networks vary in each setup. Table 5 shows that increasing the number of color features for tomato classification increases the training time. This experimental setup reveals that the number of color features used in classification affects the artificial neural network dimension, making its structure more complex. Essentially, increasing the number of color features adds to the complexity of the ANN structure thereby requiring a longer time to optimize an ANN classifier.

Table 5: Training time (in seconds) when varying the color features

	A	B	C	D
Mean	133.04	139.69	144.14	153.46
Median	132.34	139.97	142.46	151.77
Maximum	145.44	144.00	156.30	173.32
Minimum	128.31	134.69	137.94	146.10
SD	3.84	2.51	5.71	5.87
Variance	14.78	6.29	32.60	34.50

Table 6: Test set results when varying the color features

	A	B	C	D
Mean	83.20 %	95.39 %	92.93 %	98.19 %
Median	82.22 %	95.56 %	93.33 %	98.34 %
Maximum	90.55 %	97.22 %	97.78 %	100.0 %
Minimum	82.22 %	92.22 %	77.22 %	96.11 %
SD	2.29	1.43	3.92	1.04
Variance	5.25	2.05	15.34	1.08

In terms of accuracy rate, Table 6 and Table 7 show that set A and set C, which used a single color model, produced poor results. It can be observed that increasing the number of color features does not necessarily

improve the performance of the classifier as set B (2 color features) got better results than set C (3 color features). These observations are true for both the test set and training set.

Table 7: Training set results when varying the color features

	A	B	C	D
Mean	83.24 %	96.33 %	94.25 %	98.73 %
Median	82.38 %	96.67 %	94.88 %	98.81 %
Maximum	89.76 %	98.10 %	98.57 %	99.52 %
Minimum	81.90 %	91.90 %	80.24 %	96.90 %
SD	2.14	1.41	3.74	0.62
Variance	4.58	1.98	14.00	0.38

Also, the experimental results reveal that combining color features from different color models (sets B and D) produce better performance than selecting color features from a single color model (sets A and C). That is, using color features from a single color model is not very effective in classifying tomato images accurately. Among the different setups, set D which has five color features from three color models has the highest accuracy rate. Essentially, selecting the right combination of color features from the different color models affects the generalization capability of an ABC-trained ANN classifier.

Table 8 compares the classification accuracy of the proposed approach against the existing methods found in the literature that considered six (6) ripeness classes or maturity stages.

Table 8: Classification accuracy comparison

Algorithm	Color Model	Classification Accuracy
ANN-based MVS [17]	RGB	97.00%
FRBCS [7]	RGB	94.29%
OAO multi-class SVM [8]	HSV	90.80%
OAA multi-class SVM [8]	HSV	84.80%
LDA algorithm [8]	HSV	84.00%
ABC-trained ANN Tomato Classifier (our method)	RGB, HSI, CIE L*a*b*	98.19 %

As shown in Table 8, the proposed ABC-trained ANN tomato classifier achieved the highest classification accuracy of 98.19%, and this demonstrates the effectiveness of the ABC algorithm in training artificial neural networks for tomato classification.

7.0 CONCLUSIONS

In this paper, an ABC-trained ANN classifier was implemented to solve the tomato maturity classification problem. In order to evaluate its robustness, an experiment was conducted that varied the test set and training set in each run. This experiment produced very satisfactory results. Also, experimental results show that the ANN classifiers obtained very satisfactory performance both in the test set and training set.

Moreover, the generalization capability of the ABC-trained ANN classifier is greatly influenced by the selected color features. Hence, choosing the right color features from different the color models as inputs to the artificial neural network should be taken into account. In this paper, the combination of five color features (R, G, R-G, Hue, and a*) from three color models (RGB, HSI, CIE L*a*b*) enabled the ABC-trained ANN classifiers to classify the tomato images accurately. Thus, based on the experimental results, an ABC-trained ANN classifier can be used in developing an automated tomato classification system.

REFERENCES

- [1] Y. Gejima, H. Zhang and M. Nagata, "Judgment on Level of Maturity for Tomato Quality Using L*a*b* Color Image Processing", *2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 2, 2003, pp. 1355-1359.

- [2] V. Pavithra, R. Pounroja and B. SathyaBama, "Machine Vision Based Automatic Sorting Cherry Tomatoes", *Proceedings of the 2nd International Conference on Electronics and Communication System (ICECS 2015)*, 2015, pp. 271-275.
- [3] W. M. Syahrir, A. Suryanti and C. Connsynn, "Color Grading in Tomato Maturity Estimator using Image Processing Technique", *2nd IEEE International Conference on Computer Science and Information Technology (ICCSIT 2009)*, 2009, pp. 276-280.
- [4] I-S Baek, B-K Cho and Y-S Kim, "Development of a Compact Quality Sorting Machine for Cherry Tomatoes Based on Real-Time Color Image Processing", *4th International Workshop on Computer Image Analysis in Agriculture*, 2012.
- [5] X. Wang, H. Mao, X. Han and J. Yin, "Vision-based judgment of tomato maturity under growth conditions" *African Journal of Biotechnology*, vol. 10, no. 18, 2011, pp. 3616-3623.
- [6] M. Nagata, J. Tallada, F. Ishino, Y. Gejima and S. Kai, "Estimation of Tomato Ripening Stages Using Three Color Models", *Bulletin of the Faculty of Agriculture, Miyazaki University*, vol. 50(1-2), 2004, pp. 65-71.
- [7] N. Goel and P. Sehgal. "Fuzzy classification of pre-harvest tomatoes for ripeness estimation-An approach based on automatic rule learning using decision tree", *Applied Soft Computing*, vol. 36, 2015, pp. 45-56.
- [8] N. El-Bendary, E. E. Hariri, A. E. Hassanien and A. Badr, "Using machine learning techniques for evaluating tomato ripeness", *Expert Systems with Applications*, vol. 42, 2015, pp. 1892-1905.
- [9] N. S. Ukirade, "Color Grading System for Evaluating Tomato Maturity", *International Journal of Research in Management, Science & Technology*, vol. 2, no. 1, 2014, pp. 41-45.
- [10] E. Ceh-Varela and G. Hernandez-Chan, "Tomatoes Classifier using Color Histograms", *Proceedings of the International Conference on Advances in Engineering Science and Management*, 2015, pp. 97-102.
- [11] S. Kassem, M. A. Sabbah, A. M. Aboukarima and R. M. Kamel, "A study on color sorting of tomatoes maturity using machine vision and artificial neural networks", *Egyptian Journal of Agricultural Research*, vol. 93, no. 1, 2015, pp. 147-161.
- [12] M. J. McCarthy and L. Zhang, "Measurement and evaluation of tomato maturity using magnetic resonance imaging", *Postharvest Biology and Technology* vol. 67, 2011, pp. 37-43.
- [13] R. R. Mhaski, P. B. Chopade and M. P. Dale, "Determination of Ripeness and Grading of Tomato using Image Analysis on Raspberry Pi", *Proceedings of the 2015 International Conference on Communication, Control and Intelligent Systems (CCIS)*, 2015, pp. 214-220.
- [14] J. Fang, C. Zhang and S. Wang, "Application of Genetic Algorithm (GA) Trained Artificial Neural Network to Identify Tomatoes with Physiological Diseases", *International Federation for Information Processing*, vol. 259; *Computer and Computing Technologies in Agriculture*, vol. 2, 2008, pp. 1103-1111.
- [15] W. Kim, H-K Lee, S. J. Yoo and S. W. Baik, "Neural Network Based Adult Image Classification", *ICANN 2005, LNCS 3696*, 2005, pp. 481-486.
- [16] Y. Zhang, S. Wang, G. Ji and P. Phillips, "Fruit classification using computer vision and feedforward neural network", *Journal of Food Engineering*, vol. 143, 2014, pp. 167-177.
- [17] J. P. Pabico, A. V. De Grano and A. L. Zarsuela, "Neural Network Classifiers for Natural Food Products", *12th Philippine Computing Science Congress (PCSC 2012)*, 2012.
- [18] S. Goyal, "Artificial Neural Networks in Fruits: A Comprehensive Review", *I.J. Image, Graphics and Signal Processing*, vol. 5, 2014, pp. 53-63.

- [19] D. Karaboga, "An Idea Based on Honey Bee Swarm for Numerical Optimization", *Technical Report-TR06*, Erciyes University, Engineering Faculty, Computer Engineering Department, Turkey, 2005.
- [20] H. Dhari, A. M. Alimi and A. Abraham, "Designing Beta Basis Function Neural Network for Optimization Using Artificial Bee Colony (ABC)", *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1-7.
- [21] P. Y. Kumbhar and S. Krishnan, "Use Of Artificial Bee Colony (ABC) Algorithm in Artificial Neural Network Synthesis", *International Journal of Advanced Engineering Sciences and Technologies*, vol. 11, no. 1, 2011, pp. 162-171.
- [22] Z. May and M. H. Amaran, "Automated Oil Palm Fruit Grading System using Artificial Intelligence", *International Journal of Video & Image Processing and Network Security*, vol. 11, no. 3, 2011, pp. 30-35.
- [23] T. Acharya and A. K. Ray, *Image Processing: Principles and Applications*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2005.
- [24] C-L Chien and D-C Tseng, "Color Image Enhancement with Exact HSI Color Model", *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 12, 2011, pp. 6691-6710.
- [25] N. A. Ibraheem, M. M. Hasan, R. Z. Khan and U.P. Mishra, "Understanding Color Models: A Review", *ARPN Journal of Science and Technology*, vol. 2, no. 3, 2012, pp. 265-275.
- [26] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2010.
- [27] H. Saad and A. Hussain, "Classification for the Ripeness of Papayas Using Artificial Neural Network (ANN) and Threshold Rule". *4th Student Conference on Research and Development (SCORED 2006)*, 2006, pp. 132-136.
- [28] J. P. Yusiong, "Optimizing Artificial Neural Networks using Cat Swarm Optimization Algorithm" *International Journal of Intelligent Systems and Applications*, vol. 5, no.1, 2012, pp. 69-80.
- [29] X. Yao, "Evolving Artificial Neural Networks", *Proceedings of the IEEE*, vol. 87, no. 9, 1999, pp.1423-1447.
- [30] J. P. Yusiong, G. M. Seno and J. Pacurib, "SudokuBee: An Artificial Bee Colony-based Approach in Solving Sudoku Puzzles", *International Journal of Advanced Research in Computer Science*, vol. 1, no. 3, 2010, pp. 95-101.
- [31] N. Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, 1979, pp. 62-66.