

Article

A Method of Apple Image Segmentation Based on Color-Texture Fusion Feature and Machine Learning

Chunlong Zhang ², Kunlin Zou ²  and Yue Pan ^{1,*},[†]

¹ School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China

² College of Engineering, China Agricultural University, Beijing 100083, China; zcl1515@cau.edu.cn (C.Z.); zoukunlin@163.com (K.Z.)

* Correspondence: panyue19871014@163.com

[†] Current address: School of Mechatronical Engineering, Beijing Institute of Technology, 5 South Zhongguancun Street, Haidian District, Beijing 100081, China.

Received: 5 June 2020; Accepted: 28 June 2020; Published: 6 July 2020



Abstract: Apples are one of the most kind of important fruit in the world. China has been the largest apple producing country. Yield estimating, robot harvesting, precise spraying are important processes for precise planting apples. Image segmentation is an important step in machine vision systems for precision apple planting. In this paper, an apple fruit segmentation algorithm applied in the orchard was studied. The effect of many color features in classifying apple fruit pixels from other pixels was evaluated. Three color features were selected. This color features could effectively distinguish the apple fruit pixels from other pixels. The GLCM (Grey-Level Co-occurrence Matrix) was used to extract texture features. The best distance and orientation parameters for GLCM were found. Nine machine learning algorithms had been used to develop pixel classifiers. The classifier was trained with 100 pixels and tested with 100 pixels. The accuracy of the classifier based on Random Forest reached 0.94. One hundred images of an apple orchard were artificially labeled with apple fruit pixels and other pixels. At the same time, a classifier was used to segment these images. Regression analysis was performed on the results of artificial labeling and classifier classification. The average values of Af (segmentation error), FPR (false positive rate) and FNR (false negative rate) were 0.07, 0.13 and 0.15, respectively. This result showed that this algorithm could segment apple fruit in orchard images effectively. It could provide a reference for precise apple planting management.

Keywords: machine learning; apple fruit; image segmentation; color; texture

1. Introduction

Apples are one of the most important fruit in the world [1]. China is the largest apple producer country in the world. It takes a lot of labor to plant and harvest apples. With the continuous expansion of planting area and yield, precision planting and mechanized harvesting of apples are urgently needed [2,3]. Yield estimating, robot harvesting, precise spraying are important processes for precise planting [4–7]. Image segmentation is an important step for these processes [8].

In recent years, some researchers have developed many methods to segment apple fruit in the images. Researchers used color cameras, spectral cameras, and thermal cameras to obtain images of apple trees in orchards and segment apple fruit from others in these pictures [9,10]. Spectral cameras and thermal cameras can obtain heat and spectral information [11]. With this information, apple fruit can be recognized easily on the images [12]. However, the data obtained by these images are very big. This makes that the time used to process these images are long. The color camera is the most common, and it can provide color, geometric, and texture information. It was widely used in the fruit image segmentation [13]. Some researchers use threshold segmentation to segment fruit that are significantly

different in color from leaves [14]. These methods are simple and fast [1,15]. Sabzi et al. [15] proposed an apple segmentation method applying different thresholds in RGB color space. Combining with other processing the segmentation result was higher than 99.12%. However, the segmented apples in this study were all red apples at the mature stage, and the image segmentation of green apples was not involved. Lv et al. [1] designed an image segmentation method of bagged green apples. Images were performed with the OTSU segmentation algorithm and denoising to extract the normal light region of fruit. Under direct light, the average values of Af, FPR and FNR calculated by the designed method in the work were 12.15%, 8.79% and 11.83%, respectively. The researchers also pointed out that the extracted fruit region from the image of bagged green apple by using the designed method is not perfect. The reason is that the fruit bagging's the bottom edge is either leading to segmentation removal of the corresponding covered fruit region or retained as the fruit region. Image segmentation results are usually worse when the target and background colors are similar. Deep learning and machine learning were widely used in image segmentation in these years. Convolutional Neural Network (CNN) has been used to detect fruit in orchards [16–18]. For example, Halsted et al. [19] have presented a vision-only system that can accurately estimate the quantity and ripeness of sweet pepper. This system was based on a Parallel-RFCNN structure network. To improve the performance of machine vision in fruit detection for a strawberry harvesting robot, Yu et al. [16] adopted Resnet50 as a backbone network, combined with the Feature Pyramid Network (FPN) architecture for feature extraction. The result showed that the average detection precision rate was 95.78%, the recall rate was 95.41% and the mean intersection over union (MIoU) rate for instance segmentation was 89.85%. However, the training of CNNs needs a massive labeled image. Collecting and labeling so many images require a great amount of time and labor. In addition, the training of CNNs requires a large number of computing resources [7]. Three-dimensional information can be used to identify apple fruit and leaves effectively [18,20,21]. Ji et al. [20] applied morphological skeleton features and binocular stereo vision technology to apple branch positioning. However, only the depth information from binocular cameras was obtained, which was far more adequate to guide the harvesting robot picking apple fruit. Nguyen et al. [21] applied an RGB-D camera to apple fruit recognition and positioning. The distance and color features were used to remove leaves, branches, and tree trunks in the 3D point cloud [22]. In addition, then the euclidean clustering algorithm was used for apple fruit recognition. However, the proposed method had limited correct recognition rate for partially occluded apples. Gené-Mola et al. [18] used color (RGB), depth (D) and range-corrected intensity signal (S) five channels information obtained from Kinect v2 RGB-D camera and Faster R-CNN for apple fruit detection. However, the main limitation of the this methodology is that the working conditions are restricted to low illuminance levels. Acquiring 3D under outdoor conditions is limited by the sensor technology and the data processing speed [23].

Duo the drawback of deeplearning or 3D imaging, some researchers still focus on traditional image segmentation methods that make up of image features and machine learning classier [9,15,24–28]. Montalvo et al. [29] successfully segmented maize crops from weeds using combinations of RGB color components derived from Principal Component Analysis (PCA). However, when the color difference between plants and weeds is not significant, inevitably other processes are needed, such as color space transformation [30–32]. Huang et al. [33] reported that land cover classification accuracy improved for the GeoEye-1 satellite imagery with an increase in intensity levels of traditional GLCMs, while for the QuickBird satellite it reduced from 91.5% to 90.3% with an increase in intensity levels. However, the effect of texture observes depends on some parameters. When using texture feature these parameters need to be optimized [34]. Liu et al. [35] used color and shape features combine with support vector machine classifiers to segment apple fruit from images. They reported the average values of recall, precision, and F1 reach 89.80%, 95.12%, and 92.38% respectively. Ji et al. [36] proposed a new classification algorithm based on support combine with color features. The result showed a recognition success rate of approximately 89%. So it is possible to use the combination of color features, texture features, and machine learning classifiers to segment the apple image [37].

In this research, we developed an apple image segmentation algorithm for the robot in the field. We considered the combination of color and texture features with machine learning to provide a classifier. In addition, apple fruit are segmented from others from the images with this classifier. Such an apple fruit segmentation operation calls the integration of several image processing approaches. Therefore the objectives of this research were:

1. To assess and optimize the suitability of color features, texture features for apple fruit image segmentation.
2. To develop an apple fruit pixel classifier based on machine learning to segmentation images.

2. Materials and Methods

2.1. Apple Orchard Image Capture

Images of apple orchard were taken from 20:00 to 22:00 on 23 September 2018, at the Beijing International Urban Agricultural Science and Technology Park, Beijing, China (116 47'57" E, 39 52'7" N). The camera was NIKON D300S, with the original resolution of 4032×3016 . This study aimed to design an algorithm for a mobile work platform. It was needed to limit the computational resources that the algorithm consumes. Image resolution was reduced to 400×300 to reduce the computational resources. Apple fruit were aimed to be segmented from leaves and sky. In total, 105 images were provided. They were randomly divided into 2 groups. One group contained 5 images that were applied for algorithm development. The other group contained 100 images that were used for algorithm assessment. Some of the original images are shown in Figure 1.

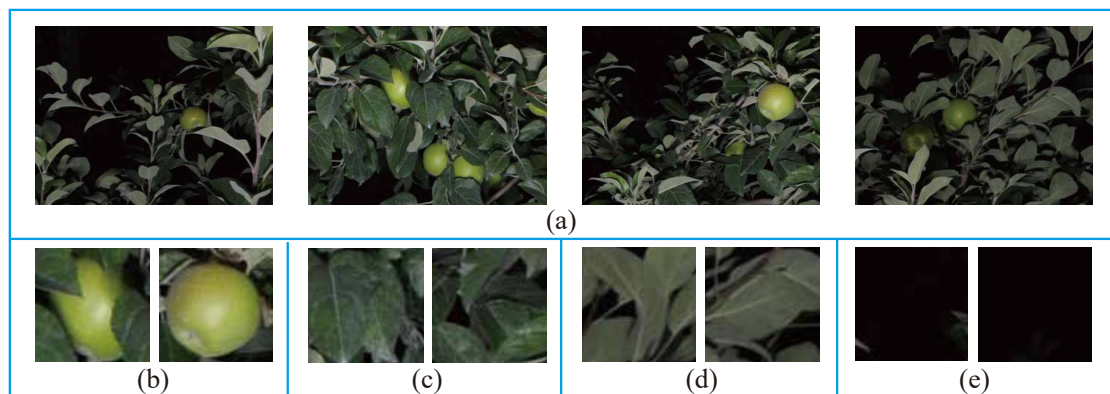


Figure 1. Some of the original images: (a) original images; (b) apple fruit; (c) up side of Leaves; (d) down side of leaves; (e) sky.

Among the five images for algorithm development, 500 pixels were randomly selected on each image. These pixels were then manually labeled as apple fruit pixels and other pixels. Figure 2 showed some of the labeled pixels on the image. In the random sampling results, there were fewer apple pixels and more other pixels, so it was necessary to downsample. One hundred apple fruit pixels and 100 other pixels were randomly selected from the labeled pixels. Among these 200 pixels, 100 pixels are randomly selected as the training set and 100 pixels as the test set.

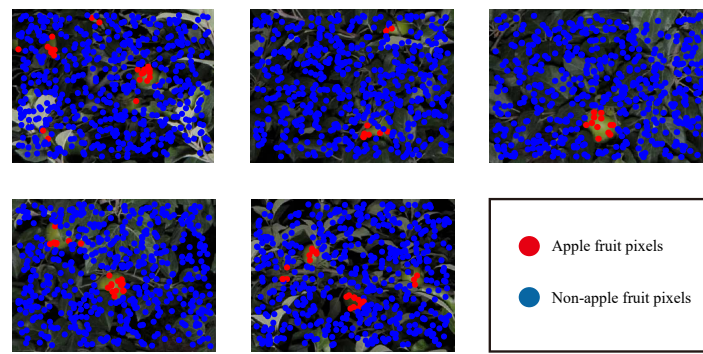


Figure 2. Sample points on images.

2.2. General Steps of the Apple Fruit Segmentation Algorithm

The segmentation algorithm was composed of two general steps including: (1) color features selection and texture observers optimization; (2) pixels classifier based on machine learning development, segmentation by pixel classification with classifier. Figure 3 shows the summarised flowchart of image segmentation strategy using the proposed algorithm. OpenCVpyhon(3.4.2), numpy(1.13.3), scikitlearn(0.20.0), scikit image(0.13.0) were used to analyse the data.

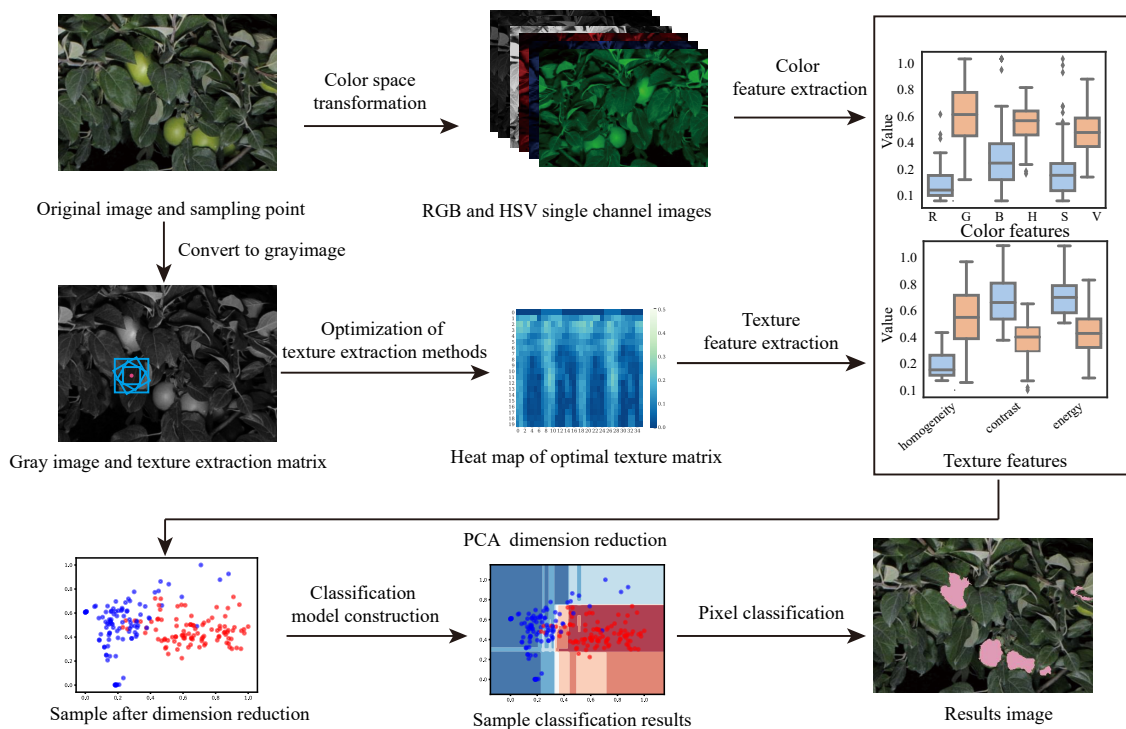


Figure 3. General steps of apple fruit segmentation algorithm.

2.3. Apple Fruit Color Features Extraction

In this paper, systematic experimentation was used to identify the acceptable color feature that best fits this apple fruit segmentation algorithm. Several frequently used color spaces were employed in this research, including RGB, HSV, XYZ, LAB, HED, YUV, YIQ color spaces. The gray-value of each raw channel of each color space was extracted as a color feature. In total, there were 21 color features obtained. They were R, G, B, H, S, V, X, Y, Z, L, A, B.1 (the B channel of LAB color space), H.1 (the H channel of HED color space), E, D, Y.1 (the Y channel of YUV color space), U, V.1 (the V channel of YUV

color space), Y.2 (the Y channel of YIQ color space), I, Q. After that, correlation analysis and chi-square tests were used to select effective features and redundant features. Pearson Correlation Coefficient (PCC) of each color feature group combination was calculated. A chi-square test was used to analyze the correlation between features and the classification target. When the PCC between two features was higher than or equal to 0.8, the feature of high p -value from the chi-square test was deleted. The rest of the features was used for classifier development.

2.4. Apple Fruit Texture Features Extraction

The grey-level co-occurrence matrix (GLCM) is one of the most popular statistical approaches used in texture discrimination [38]. A unique co-occurrence matrix exists for each spatial relationship. The calculation of textures is dependent upon the direction (D) and the orientation (O) [39]. Therefore, D and O parameters of texture observer were optimized by the grid search method. The search range of D was $[0,19]$ and O was $[0,360]^\circ$ (resolution is 10°). The GLCM was quite complex, and some characteristic values of texture features are usually used as texture features. In this paper, characteristic values used include contrast, dissimilarity, homogeneity, ASM, energy, correlation. The calculation formulas are shown in Equations (1)–(6). After texture features extraction, Correlation Coefficient analyses and the chi-square test were also used to select effective features.

$$\text{Contrast} = \sum_{i,j=0}^{\text{levels}-1} P_{i,j}(i-j)^2 \quad (1)$$

$$\text{Dissimilarity} = \sum_{i,j=0}^{\text{levels}-1} P_{i,j}|i-j| \quad (2)$$

$$\text{Homogeneity} = \sum_{i,j=0}^{\text{levels}-1} \frac{P_{i,j}}{1+(i-j)^2} \quad (3)$$

$$\text{ASM} = \sum_{i,j=0}^{\text{levels}-1} P_{i,j}^2 \quad (4)$$

$$\text{Energy} = \sqrt{\text{ASM}} \quad (5)$$

$$\text{Correlation} = \sum_{i,j=0}^{\text{levels}-1} P_{i,j} \left[\frac{(i-u_i)(j-u_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} \right] \quad (6)$$

where i is the row number; j is the column number; $P_{i,j}$ is the normalized value in the cell i, j ; N is the number of rows or columns, μ is the mean value, σ is the variance.

2.5. Data Normalization and Dimension Reduction

The data after obtaining should be scaled to a reasonable range, and transferred to a non-dimensional data, for the features extracted in this paper contained multiple dimensions and were not steady. Normalization formula was used to ensure that each dimension of the data ranges from zero to one [40]. Data normalization was calculated by Equation (7).

$$Y = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (7)$$

where Y is the normalized value, X is the original value, X_{\min} is the minimum value and X_{\max} is the maximum value.

Selected features contained both useful and irrelevant information for the apple fruit pixels identification. Also, it was necessary to do dimensionality reduction to reduce computer

resource-consuming [39]. Principal Components Analysis (PCA) was used to decompose a multivariate dataset in a set of successive orthogonal components that explain a maximum amount of the variance. In scikit-learn, PCA is implemented as a transformer object that learns components in its fit method and can be used on new data to project it on this components [41]. To retain more information while dimension reduction, two were selected as the number of components [42].

2.6. Classifier Development and Pixels Classification

The classification was one of the main components of a segmentation algorithm. For this reason, the classifier should be selected carefully [43]. In this research, nine machine learning algorithms had been used to develop pixel classifiers. They were Nearest Neighbors, Linear Support Vector Machine (Linear SVM), Radial Basis Function Support Vector Machine (RBF SVM), Gaussian Process, Decision Tree, Random Forest, Neural Net, AdaBoost, Naive Bayes, Quadratic Discriminant Analysis (QDA) [41].

One hundred samples were used to train these classifiers, and one hundred samples were used to test these classifiers. The accuracy of the training set and testing set was calculated respectively to test the effect of classifier and avoid over-fitting. The accuracy calculation method is shown in (8). True Positive Rate (TPR) was used to evaluate classifier performance. The calculation formulas are shown in (9).

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative}} \quad (8)$$

$$\text{True Positive Rate} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (9)$$

where *TruePositive* (TP) is the number of pixels detected as apple fruit pixels correctly. *TrueNegative* (TN) is the number of pixels detected as others correctly. *FalsePositive* (FP) is the number of other pixels detected as apple fruit pixels, *FalseNegative* (FN) is the number of apple fruit pixels detected as others.

2.7. Apple Fruit Segmentation Result Test

To test the developed algorithm, pixels in 100 images were manually labeled as either apple fruit pixels or other. The manually labeled pixels were seen as ground-truth. and the algorithm evaluation result was seen as predict result. The image segmentation effect of the algorithm was evaluated by the total number of manually labeled apple fruit pixels and the total number of apple fruit pixels predicted by the algorithm. Finally, the experimental results were evaluated by the segmentation error A_f , false positive rate FPR and false negative rate FNR. These three criteria were calculated by Equations (10)–(12).

$$A_f = \frac{|A_1 - A_2|}{A_1} \times 100\% \quad (10)$$

$$FPR = \frac{|A_2 - (A_2 \cap A_1)|}{\bar{A}_1} \times 100\% \quad (11)$$

$$FNR = \frac{|A_1 - (A_1 \cap A_2)|}{A_1} \times 100\% \quad (12)$$

where A_1 represents the real area of fruit target; A_2 the fruit area acquired after segmentation; \bar{A}_1 the complementary set of A_1 ; FPR the percentage of the pixel belonging to the background that is mistakenly segmented as fruit pixel in algorithm segmentation; FNR the percentage of the pixel belonging to fruit that being mistakenly segmented as background pixel in algorithm segmentation. The smaller the values of A_f , FPR and FNR are, the better of segmentation effectiveness and higher accuracy will be.

3. Results and Discussion

3.1. Color Features Selection Result

The chi-square test p -value of color features is shown in Table 1. There were some p -value of the color channel was lower than 0.05. This meant that these channels could significantly distinguish apple fruit pixels from non-apple fruit pixels. These channels were R, G, S, V, X, Y, L, A, B.1, H.1, Y.1, U, Y.2, I, Q. There was also some p -value of the color channel was higher than 0.05. This meant that these channels could not significantly distinguish apple fruit pixels from non-apple fruit pixels. These channels were B, H, Z, E, D, V.1.

Table 1. Color features chi - square test result.

Color Features	p -Value
R	6.09×10^{-4}
G	5.66×10^{-4}
B	2.54×10^{-1}
H	4.04×10^{-1}
S	1.75×10^{-2}
V	5.64×10^{-4}
X	8.90×10^{-3}
Y	4.36×10^{-2}
Z	4.10×10^{-1}
L	3.12×10^{-4}
A	3.01×10^{-4}
B.1	8.57×10^{-9}
H.1	7.46×10^{-4}
E	4.35×10^{-1}
D	4.16×10^{-1}
Y.1	1.10×10^{-3}
U	1.06×10^{-4}
V.1	4.06×10^{-1}
Y.2	1.10×10^{-3}
I	5.21×10^{-6}
Q	2.10×10^{-4}

By observing the correlation matrix heat map (Figure 4), the features with a correlation coefficient greater than 0.8 were divided into one group. In total, there were 6 groups. The first group was R, G, B, V, X, Y, L, A, B.1, H.1, Y.1, U, Y.2, I, Q. The second group was the H channel. The third group was the S channel. The 4th group was the E channel. The 5th group was the D channel. The 5th group was the Y.1 channel. According to the chi-square test, the p -value of H, E, D channel were higher than 0.05. They were unable to distinguish the pixels of the apple fruit from the pixels of others. They have not been used in the segmentation algorithm. There were 15 channels in the first group. However, they were significantly correlated. They contained much of the same information. It was not necessary to use all of them in the algorithm. The channel with the lowest p -value, the B.1 channel (8.57×10^{-09}), was selected to develop the segmentation algorithm. Finally, the B.1, S, and Y.1 channel was selected as the color features to build the segmentation algorithm.

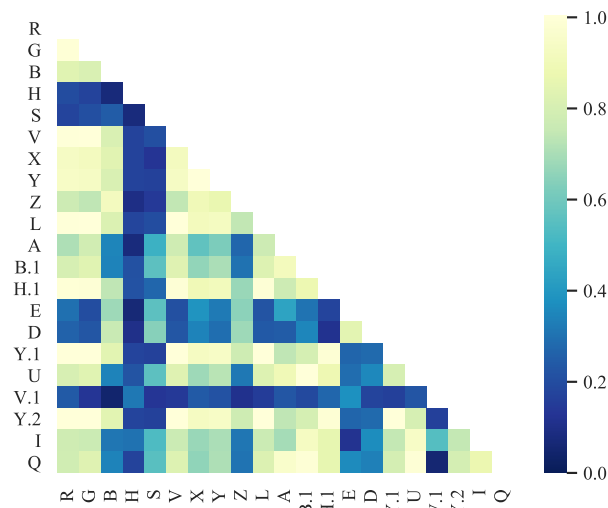


Figure 4. Correlation matrix heat map of color features.

The boxplot of selected color features is shown in Figure 5. In the three channels, the mean value of Apple fruit pixels was lower and that of non-apple fruit pixels was higher. The quartiles of the two kind of samples had no intersection. Most of the Apple fruit pixels and Non-apple fruit pixels samples had different values in these channels. However, there was overlap between the lower edge of the non-apple fruit pixels sample set and the upper edge of apple fruit pixels, which indicates that the values of a small number of samples in this channel were the same. This might be due to the fact that the colors of Apple fruit and leaves were relatively similar, so the difference between the two types of samples in color characteristics was not large. At the same time, in the three channels, the dispersion degree of Apple fruit pixels samples was relatively small, and that of non-apple fruit pixels samples was relatively large. This was because there was only one class of Apple fruit pixels samples, but non-apple fruit pixels was composed of two different types of samples, sky and leaves, so it was relatively dispersed. In the three channels, there were some samples that cannot be effectively distinguished, so it was impossible to distinguish the two types of samples by color features alone, and they need to be combined with other features.

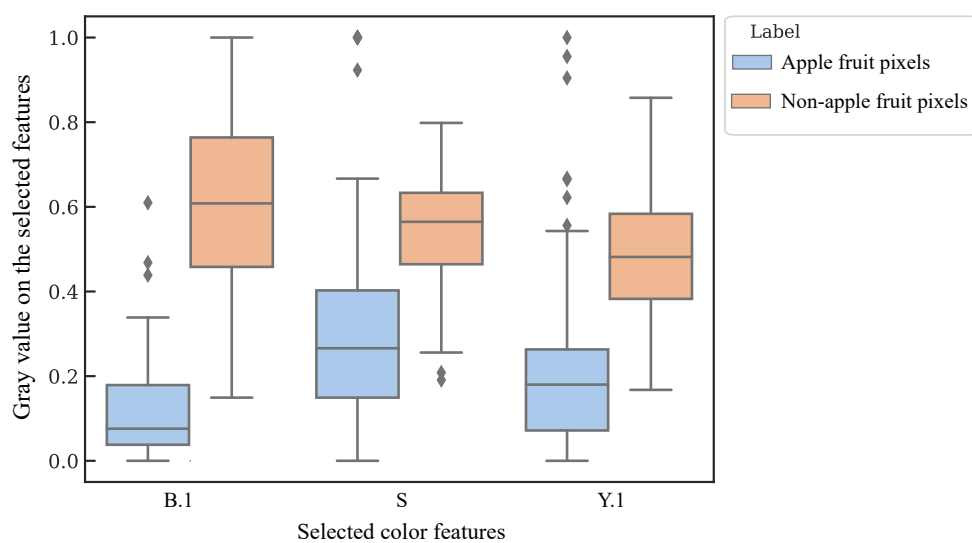


Figure 5. Boxplot of selected color features.

3.2. Texture Features Selection Result

Chi-square test results of texture features extracted from GLCM of different distance and orientation are shown in Figure 6. It could be seen from the picture that the minimum p -value of Contrast appeared at (3,12). The minimum p -value of dissimilarity appeared at 7 positions, and the first one was (0,3). The minimum p -value of dissimilarity appeared at 7 positions, and the first one was (0,3). The minimum p -value of homogeneity appeared at 2 positions, and the first one was (19,3). The minimum p -value of ASM appeared at 14 positions, and the first one was (2,3). The minimum p -value of energy appeared at 14 positions, and the first one was (2,3). The minimum p -value of correlation appeared at 36 positions, and the first one was (0,0). ASM and energy minimum p -value was lower than 0.05, which can significantly distinguish the target.

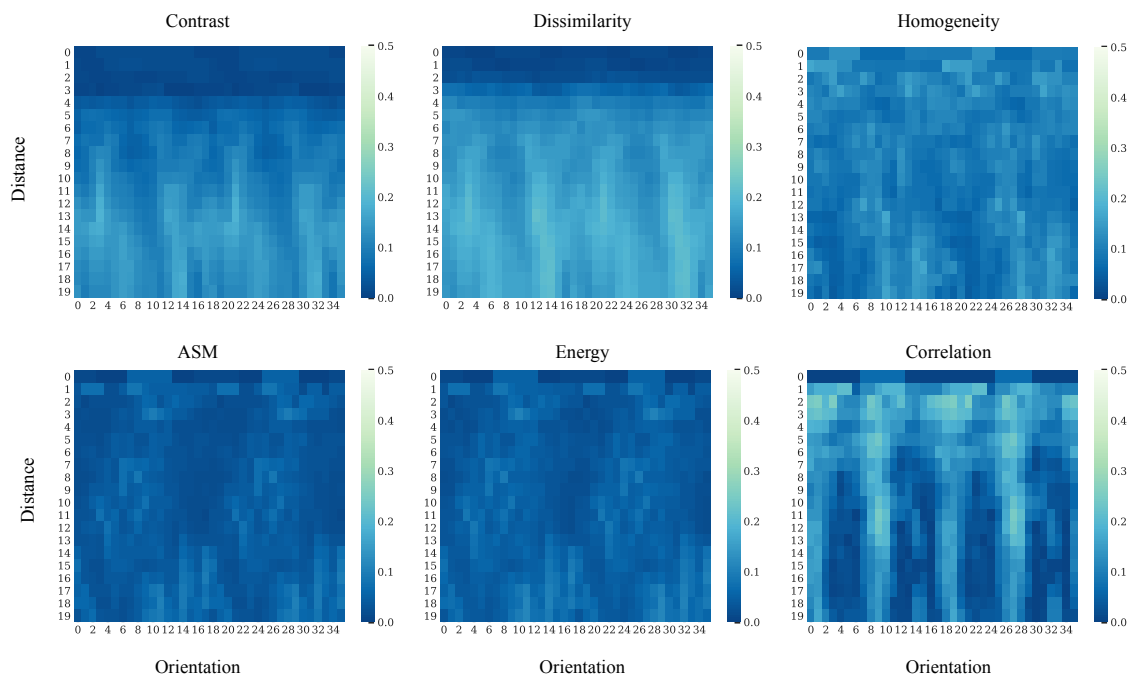


Figure 6. Heat map of optimization results of Distance and Orientation with different texture features.

Results of texture feature correlation analysis are shown in Figure 7. In addition, the result of the texture feature chi-square test is shown in Table 2. The correlation between the ASM and energy was more than 0.8. The correlation coefficient between other features was no more than 0.8, indicating that the texture features contained less repeated information. This suggested that ASM and energy contained much of the same information. It was necessary to remove one of them. The energy had a lower P-value than ASM, so the ASM was removed. The energy was selected to build the segmentation algorithm.

Table 2. Minimum value of texture feature chi - square test.

Texture Features	Minimum p -Value
Contrast	9.10×10^{-1}
Dissimilarity	8.51×10^{-1}
Homogeneity	2.24×10^{-1}
ASM	1.59×10^{-2}
Energy	1.23×10^{-2}
Correlation	1.23×10^{-1}

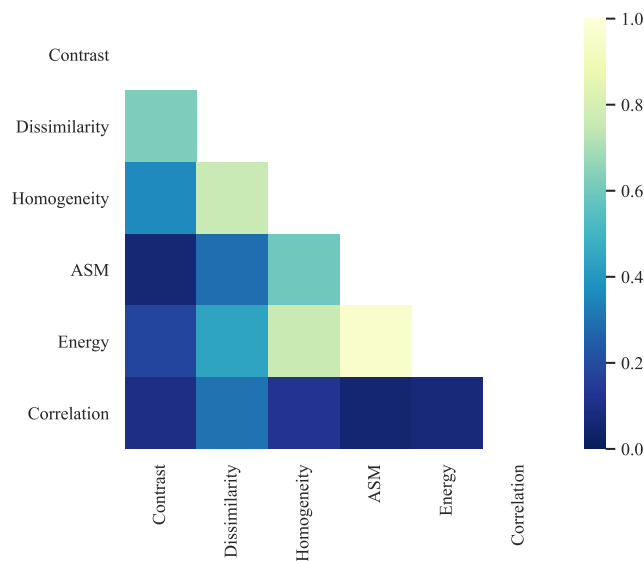


Figure 7. Correlation matrix heat map of texture features.

Texture features were very poor in differentiating the expressions of apple fruit pixels. In the optimization of distance and orientation, the relationship of the distance and performance was that the larger the distance, the worse the performance. The information about the orientation of the performance was not significant. The minimum p -value of many texture features appears in many orientation points. Even all the p -values of orientation were the same. At the same time, the lowest p -value mostly appears in the position with a small distance. Even the lowest p -value appeared at (0,0). This suggested that the larger the GLCM, the worse the effect. It showed that the influence of adjacent pixels was very weak, even causing interference to the classification. This might be due to the smooth surface of the apple fruit. There was also no special texture. Therefore, it was difficult for GLCM to extract the features that distinguish apple fruit pixels effectively.

3.3. Apple Fruit Pixels Classification Result

Sample distribution after dimension reduction analysis of PCA is shown in Figure 8. The apple fruit pixels samples were relatively concentrated in the principal component space. This was because apple fruit pixels were all derived from apple fruit. The non-apple fruit pixels samples were dispersed in the principal component space. This was due to the complex origin of non-apple fruit pixels, including sky and leaves. In addition, leaves contained two kinds of, front and back. The two types of samples were interlaced in the principal component space. This would cause a classification error. The two types of samples could not perfectly linearly be separated in the principal component space. Therefore, more complex classifiers were needed to be designed.

Different apple fruit pixels classifiers development result is shown in Figure 9. The Nearest Neighbors algorithm could better classify apple fruit pixels and non-apple fruit pixels. The classification boundaries were also clear. The classifier obtained by the Linear SVM algorithm would perform Linear segmentation of the two types of samples in the solution space. However, because these two types of samples were not completely linearly separable in the solution space. The effect of this classifier was poor. The classifier based on the RBF SVM algorithm used RBF kernel function, so it has a good performance on linearly indivisible samples. The classifier obtained by the Gaussian Process could solve the nonlinear separable problem well. It was sensitive to the density of sample points. In the region of solution space with less sample distribution, it had no classification bias. The Decision Tree algorithm adopted multiple straight lines in the solution space. It made up for the problem of non-linear sample classification by multiple linear classifications. It had a very distinct classification boundary. The Random Forest algorithm improved the accuracy by integrating multiple

Decision Tree algorithms. However, it could not be classified with curves either. It was also classified by multiple lines. Neural Net algorithm can only carry out linear classification due to the small number of neurons in the hidden layer. The AdaBoost algorithm also classified by composite lines. Naive Bayes was also a very good curve classifier and has achieved a very good classification effect.

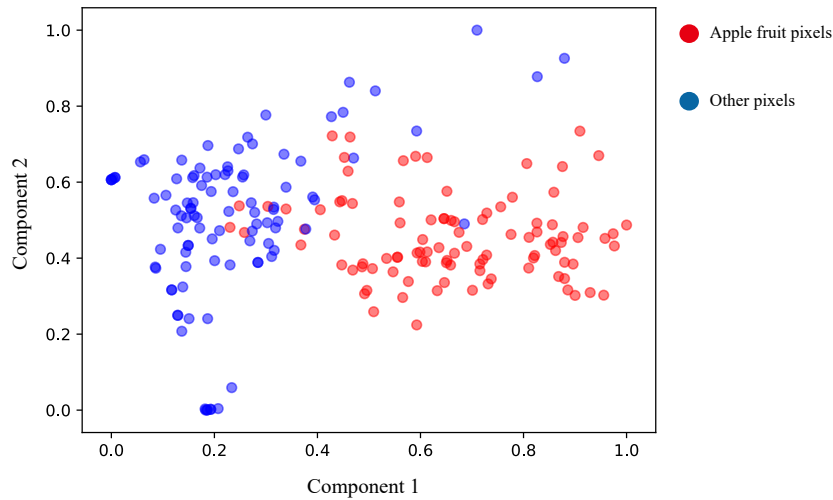


Figure 8. The scatter diagram of apple fruit pixels and other pixels in the dimension reducing principal component space.

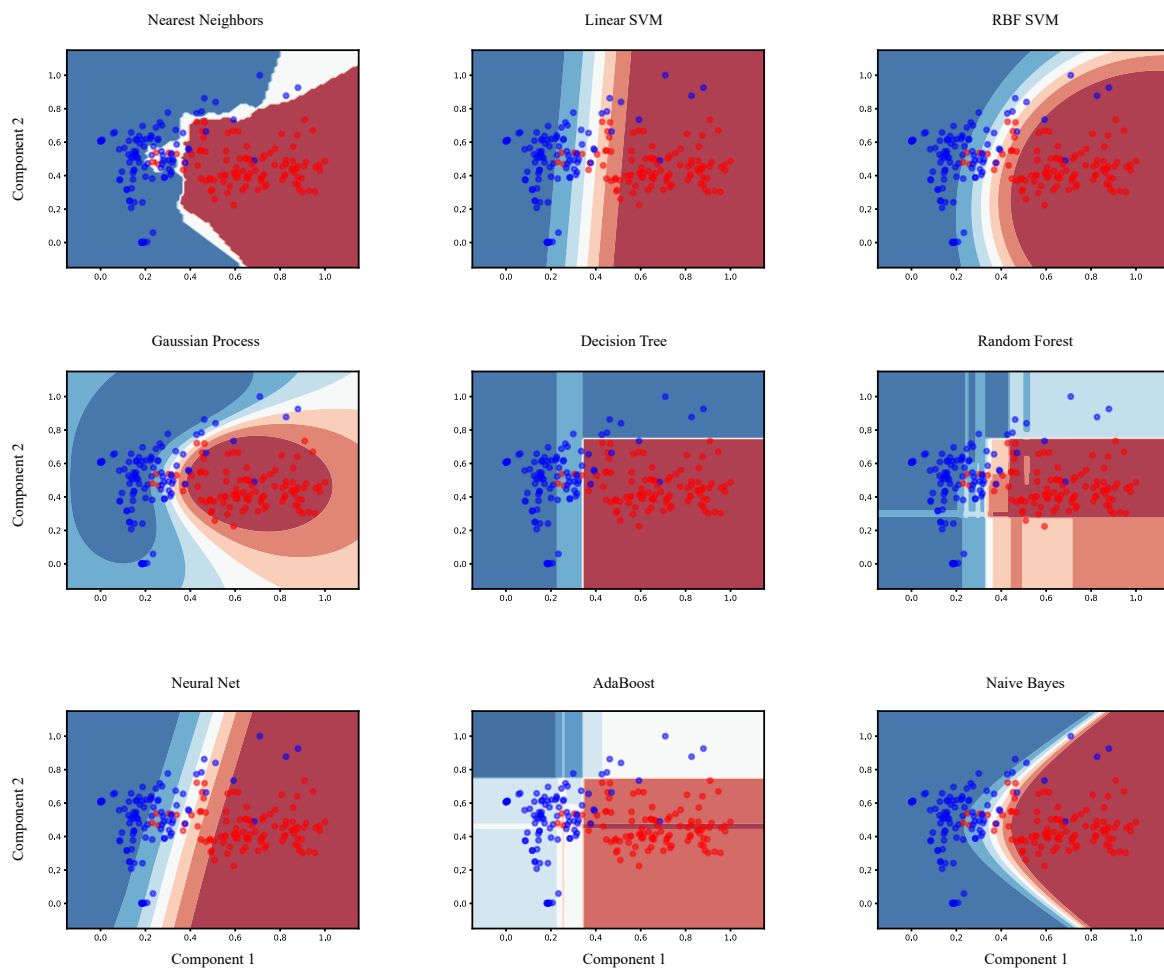


Figure 9. Different apple fruit pixels classifiers development result.

The results of the classifier classification in the training set and test set are shown in Table 3. The classifiers based on Nearest Neighbors, Decision Tree, and Random Forest, the classifier had achieved high classification accuracy. Their accuracy was 0.94, 0.95, and 0.94, respectively. However, the classification accuracy of the classifiers based on the Decision Tree and Nearest Neighbors algorithm in the testing set was significantly lower than that in the training set. These classifiers were overfitting. The accuracy of the classifier training set and testing set based on the Random Forest algorithm is 0.94. It was not overfitting, and its classification accuracy was relatively high. Meanwhile, it can be seen that the TPR of the Random Forest algorithm was 0.90 in the test set. This indicated that the algorithm classified some other pixels as apple pixels. This was because some of the leaf pixels were similar in color and texture to apples. The TPR value of the Random Forest algorithm in the test set was also higher than that of other algorithms. Therefore, the classifier of the Random Forest was selected to classify the pixels in this study.

Table 3. Accuracy of apple fruit pixels classifier.

Classifier Name	Train Set Accuracy	Test Set Accuracy	Train Set TPR	Test Set TPR
Nearest Neighbors	0.94	0.85	0.89	0.86
Linear SVM	0.87	0.88	0.88	0.79
RBF SVM	0.90	0.91	0.85	0.83
Gaussian Process	0.91	0.92	0.91	0.85
Decision Tree	0.95	0.91	0.95	0.86
Random Forest	0.94	0.94	0.94	0.90
Neural Net	0.88	0.90	0.85	0.85
AdaBoost	0.92	0.91	1.0	0.86
Naive Bayes	0.90	0.90	0.86	0.83
QDA	0.92	0.91	0.86	0.87

3.4. Apple Fruit Image Segmentation Result

We compared the segmentation results of the proposed method with three other segmentation algorithms. These were Otsu based on R-B and boundary object removal [1], K-means cluster segmentation method based on R-B, and adaptive threshold segmentation method based on R-B. The results are shown in Table 4. Calculated by the designed segmentation method in the work, the average values of Af, FPR and FNR were 0.07, 0.13 and 0.15, respectively. Calculated by the Otsu based on R-B and boundary object removal, the average values of Af, FPR and FNR were 0.26, 0.09 and 0.34, respectively. When calculated by the K-means cluster segmentation method based on R-B, the average values of Af, FPR and FNR were 0.29, 0.28 and 0.18, respectively. When calculated by the auto-adaptive threshold segmentation algorithm based on R-B, the average values of Af, FPR and FNR were 0.35, 0.39 and 0.14, respectively. It can be seen that the algorithm proposed in this paper has a good effect on Af, FPR and FNR.

Table 4. Evaluation index statistics of different segmentation method.

Method	Af	FPR	FNR
This designed segmentation method	0.07	0.13	0.15
Otsu based on R-B and boundary object removal	0.26	0.09	0.34
K-means cluster segmentation method based on R-B	0.29	0.28	0.18
Adaptive threshold segmentation method based on R-B	0.35	0.39	0.14

It could be seen from Figure 10 that the method based on Otsu and boundary object removal had good results. However, some apple pixels were classified as others. Therefore, the FPR value of the segmentation results obtained by this algorithm was low, but the FNR value was high. The Adaptive Threshold segmentation method based on the R-B segmentation method had poor segmentation results. The error mainly comes from the fact that some other pixels were classified as Apple pixels. Therefore,

the FPR value of the segmentation results of this algorithm was higher, while the FNR value was lower. The R-B segmentation method based on the K-means Cluster segmentation method also misdivided some other pixel points into Apple pixel points. The results of the segmentation algorithm presented in this paper had a good performance in each evaluation index.

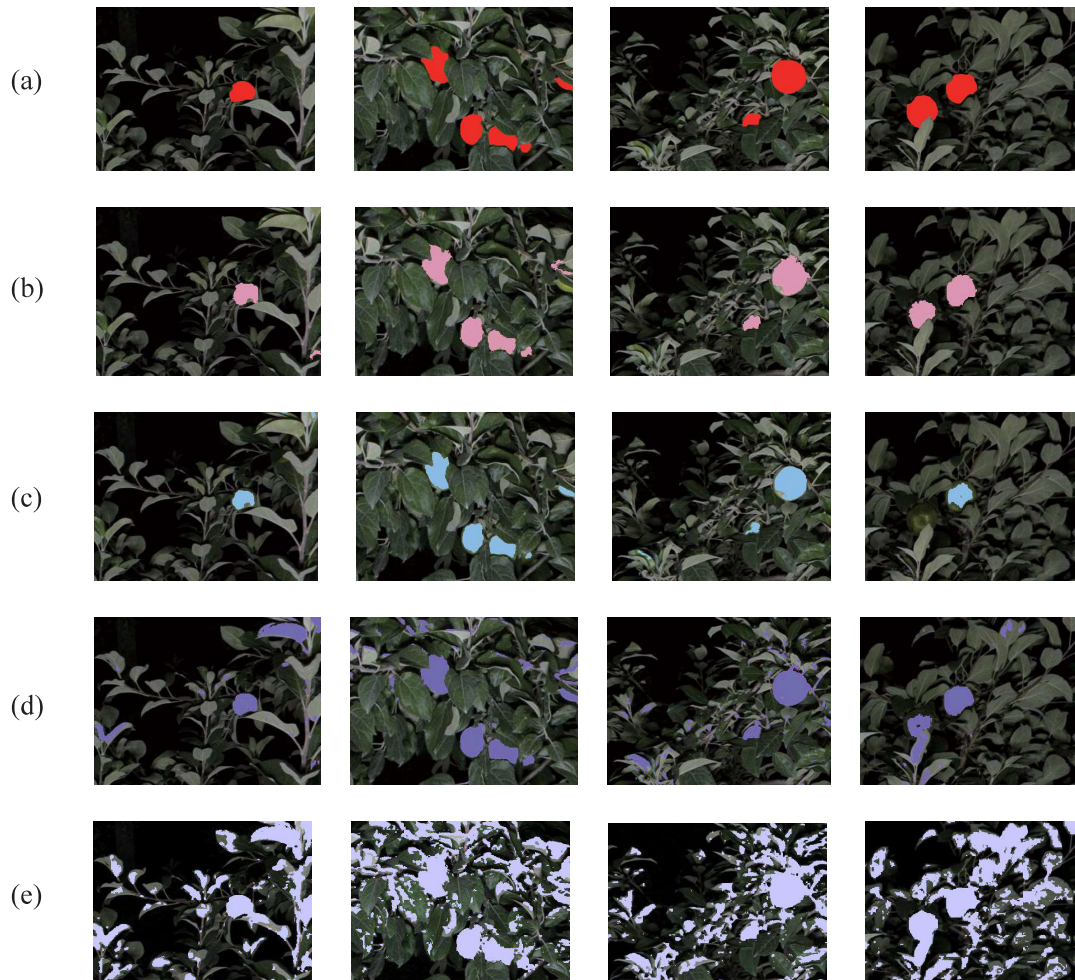


Figure 10. Manually labeling results and algorithm segmentation results: (a) Manually labeling results, (b) This designed segmentation method results, (c) Otsu based on R-B and boundary object removal, (d) K-means cluster segmentation method based on R-B, (e) Adaptive threshold segmentation method based on R-B.

3.5. Discussion

As can be seen from Figure 10, the method of Otsu based on R-B and boundary object removal could effectively segment part of apple and background. However, this algorithm segment some apple pixels into others when the light was dark. At the same time, this algorithm removed some small areas of apple. So part of the apple was not segmented correct. Both the k-means algorithm and the adaptive threshold algorithm had errors in the segmenting part of leaf pixels as apple pixels. This was mainly because the color of the underside of some leaves was similar to apples in a dark environment, and it was difficult to segment them simply by color. The algorithm proposed in this paper combined color and texture features, which could complement each other. Texture features played an important role in separating leaves and apples. Therefore, the algorithm proposed in this paper could segment apples better.

The image segmentation method based on deep learning has achieved good results in fruit recognition in the orchard environment [44,45]. Because the deep learning algorithm can have strong

adaptability. However, these algorithms are very complex. Lots of labeled pictures are needed during training. The image label task for the semantic segmentation task is the pixel by pixel level. Such tasks require a lot of human labor. The structure of a deep learning network is complex, so it needs to run on the platform with rich computing resources. Due to the limitations of space and energy consumption, these mobile platforms are equipped with very limited computing resources, and the current deep learning algorithms are not suitable for deployment on the robot. The algorithm studied in this paper, based on meeting the needs of orchard robot work, reduced the computational resource burden to the greatest extent. It is an algorithm suitable for orchard robots.

4. Conclusions

In this paper, apple images from orchards were gathered by a camera. Color and texture features were used to build pixels classifiers. In addition, nine classifiers based on different machine learning algorithms were built to classify apple fruit pixels from other pixels. In addition, the apple segmentation method was obtained. Through the analysis of the experimental results, it was found that:

- (1) Color features could effectively distinguish apple fruit pixels from others, while texture features had a poor performance in this;
- (2) The classification algorithm based on Random Forest could effectively classify the apple fruit pixels, and the accuracy was 0.94
- (3) Image segmentation can be done through pixel classification. The average values of Af, FPR and FNR were 0.07, 0.13 and 0.15, respectively.
- (4) The image segmentation model established by pixel classification could effectively segment apple fruit from photos.

Author Contributions: Data curation, C.Z.; Methodology, K.Z.; Project administration, Y.P.; Software, K.Z.; Writing - original draft, C.Z. All authors have read and agree to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (31601217) and National Science and Technology Infrastructure Program (2015BAF20B02).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lv, J.; Wang, Y.; Xu, L.; Gu, Y.; Zou, L.; Yang, B.; Ma, Z. A method to obtain the near-large fruit from apple image in orchard for single-arm apple harvesting robot. *Sci. Hortic.* **2019**, *257*, 108758. [[CrossRef](#)]
2. Bednarz, C.W.; Shurley, W.D.; Anthony, W.S. Losses in Yield, Quality, and Profitability of Cotton from Improper Harvest Timing. *Agron. J.* **2002**, *94*, 1004–1011. [[CrossRef](#)]
3. Pothula, A.K.; Zhang, Z.; Lu, R. Design features and bruise evaluation of an apple harvest and in-field presorting machine. *Trans. ASABE* **2018**, *61*, 1135–1144.
4. Linker, R.; Cohen, O.; Naor, A. Determination of the number of green apples in RGB images recorded in orchards. *Comput. Electron. Agric.* **2012**, *81*, 45–57.
5. Dong, C.; Tian, F.; Dong, X.; Zhao, X.; Li, F. The Structure and Control Analysis of AMR Automatic Harvesting Robot. In Proceedings of the International Conference on Mechatronics and Intelligent Robotics, Kunming, China, 20–21 May 2017; Springer: Cham, Switzerland, 2017; pp. 457–463.
6. Qian, L.; Wang, X.; Luo, W.; Qi, Z.; Sun, H.; Luo, Y. An improved CROPR model for estimating cotton yield under soil aeration stress. *Crop Pasture Sci.* **2017**, *68*, 366–377. [[CrossRef](#)]
7. Koirala, A.; Walsh, K.B.; Wang, Z.; McCarthy, C. Deep learning—Method overview and review of use for fruit detection and yield estimation. *Comput. Electron. Agric.* **2019**, *162*, 219–234. [[CrossRef](#)]
8. Stein, M.; Bargoti, S.; Underwood, J. Image Based Mango Fruit Detection, Localisation and Yield Estimation Using Multiple View Geometry. *Sensors* **2016**, *16*, 1915. [[CrossRef](#)] [[PubMed](#)]
9. Gongal, A.; Amatya, S.; Karkee, M.; Zhang, Q.; Lewis, K. Sensors and systems for fruit detection and localization: A review. *Comput. Electron. Agric.* **2015**, *116*, 8–19. [[CrossRef](#)]
10. Kang, H.; Chen, C. Fruit detection and segmentation for apple harvesting using visual sensor in orchards. *Sensors* **2019**, *19*, 4599. [[CrossRef](#)]

11. Zemmour, E.; Kurtser, P.; Edan, Y. Automatic Parameter Tuning for Adaptive Thresholding in Fruit Detection. *Sensors* **2019**, *19*, 2130. [[CrossRef](#)] [[PubMed](#)]
12. Feng, J.; Zeng, L.; He, L. Apple Fruit Recognition Algorithm Based on Multi-Spectral Dynamic Image Analysis. *Sensors* **2019**, *19*, 949. [[CrossRef](#)]
13. Chaivivatrakul, S.; Dailey, M.N. Texture-based fruit detection. *Precis. Agric.* **2014**, *15*, 662–683. [[CrossRef](#)]
14. Huang, L.; He, D.; Yang, S.X. Segmentation on Ripe Fuji Apple with Fuzzy 2D Entropy based on 2D histogram and GA Optimization. *Intell. Autom. Soft Comput.* **2013**, *19*, 239–251. [[CrossRef](#)]
15. Sabzi, S.; Abbaspourgilandeh, Y.; Garciamateos, G.; Ruizcanales, A.; Molinamartinez, 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](#)]
16. 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](#)]
17. Liu, B.; Zhang, Y.; He, D.; Li, Y. Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks. *Symmetry* **2017**, *10*, 11. [[CrossRef](#)]
18. Gené-Mola, J.; Vilaplana, V.; Rosell-Polo, J.R.; Morros, J.R.; Ruiz-Hidalgo, J.; Gregorio, E. Multi-modal deep learning for Fuji apple detection using RGB-D cameras and their radiometric capabilities. *Comput. Electron. Agric.* **2019**, *162*, 689–698. [[CrossRef](#)]
19. Bargoti, S.; Underwood, J. Image Segmentation for Fruit Detection and Yield Estimation in Apple Orchards. *J. Field Robot.* **2017**, *34*, 1039–1060. [[CrossRef](#)]
20. Ji, W.; Meng, X.; Qian, Z.; Xu, B.; Zhao, D. Branch localization method based on the skeleton feature extraction and stereo matching for apple harvesting robot. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 172988141770527. [[CrossRef](#)]
21. Nguyen, T.T.; Vandevoorde, K.; Wouters, N.; Kayacan, E.; De Baerdemaeker, J.; Saeys, W. Detection of red and bicoloured apples on tree with an RGB-D camera. *Biosyst. Eng.* **2016**, *146*, 33–44. [[CrossRef](#)]
22. Mendez, V.; Perezromero, A.; Solaguirado, R.; Mirandafuentes, A.; Manzanoagugliaro, F.; Zapatasierra, A.; Rodriguezlizana, A. In-Field Estimation of Orange Number and Size by 3D Laser Scanning. *Agronomy* **2019**, *9*, 885. [[CrossRef](#)]
23. Kazmi, W.; Foix, S.; Alenya, G.; Andersen, H.J. Indoor and outdoor depth imaging of leaves with time-of-flight and stereo vision sensors: Analysis and comparison. *ISPRS J. Photogramm. Remote Sens.* **2014**, *88*, 128–146. [[CrossRef](#)]
24. Linker, R. Machine learning based analysis of night-time images for yield prediction in apple orchard. *Biosyst. Eng.* **2018**, *167*, 114–125. [[CrossRef](#)]
25. Nyarko, E.K.; Vidovic, I.; Radocaj, K.; Cupec, R. A nearest neighbor approach for fruit recognition in RGB-D images based on detection of convex surfaces. *Expert Syst. Appl.* **2018**, *114*, 454–466. [[CrossRef](#)]
26. Sabzi, S.; Abbaspourgilandeh, Y.; Hernandezhernandez, 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](#)]
27. Yamamoto, K.; Guo, W.; Yoshioka, Y.; Ninomiya, S. On Plant Detection of Intact Tomato Fruits Using Image Analysis and Machine Learning Methods. *Sensors* **2014**, *14*, 12191–12206. [[CrossRef](#)]
28. Ismail, A.; Idris, M.Y.L.; Ayub, M.N.; Por, L.Y. Investigation of Fusion Features for Apple Classification in Smart Manufacturing. *Symmetry* **2019**, *11*, 1194. [[CrossRef](#)]
29. Montalvo, M.; Guijarro, M.; Guerrero, J.M.; Ribeiro, A. Identification of Plant Textures in Agricultural Images by Principal Component Analysis. In Proceedings of the International Conference on Hybrid Artificial Intelligence Systems, Seville, Spain, 18–20 April 2016; pp. 391–401.
30. Tufail, Z.; Khurshid, K.; Salman, A.; Nizami, I.F.; Khurshid, K.; Jeon, B. Improved dark channel prior for image defogging using RGB and YCbCr color space. *IEEE Access* **2018**, *6*, 32576–32587. [[CrossRef](#)]
31. Zhang, T.; Hu, H.M.; Li, B. A naturalness preserved fast dehazing algorithm using HSV color space. *IEEE Access* **2018**, *6*, 10644–10649. [[CrossRef](#)]
32. Kahu, S.Y.; Bhurchandi, K.M. A Low-Complexity, Sequential Video Compression Scheme Using Frame Differential Directional Filter Bank Decomposition in CIELa*b* Color Space. *IEEE Access* **2017**, *5*, 14914–14929. [[CrossRef](#)]
33. Huang, X.; Liu, X.; Zhang, L. A Multichannel Gray Level Co-Occurrence Matrix for Multi/Hyperspectral Image Texture Representation. *Remote Sens.* **2014**, *6*, 8424–8445. [[CrossRef](#)]

34. Rehman, T.U.; Zaman, Q.U.; Chang, Y.K.; Schumann, A.W.; Corscadden, K.; Esau, T. Optimising the parameters influencing performance and weed (goldenrod) identification accuracy of colour co-occurrence matrices. *Biosyst. Eng.* **2018**, *170*, 85–95. [[CrossRef](#)]
35. Liu, X.; Zhao, D.; Jia, W.; Ji, W.; Sun, Y. A Detection Method for Apple Fruits Based on Color and Shape Features. *IEEE Access* **2019**, *7*, 67923–67933. [[CrossRef](#)]
36. Ji, W.; Zhao, D.; Cheng, F.; Xu, B.; Zhang, Y.; Wang, J. Automatic recognition vision system guided for apple harvesting robot. *Comput. Electr. Eng.* **2012**, *38*, 1186–1195. [[CrossRef](#)]
37. Zhao, Y.; Gong, L.; Huang, Y.; Liu, C. Robust Tomato Recognition for Robotic Harvesting Using Feature Images Fusion. *Sensors* **2016**, *16*, 173–173. [[CrossRef](#)] [[PubMed](#)]
38. Haralick, R.M. Statistical and structural approaches to texture. *Proc. IEEE* **1979**, *67*, 786–804. [[CrossRef](#)]
39. Lin, F.; Zhang, D.; Huang, Y.; Wang, X.; Chen, X. Detection of Corn and Weed Species by the Combination of Spectral, Shape and Textural Features. *Sustainability* **2017**, *9*, 1335. [[CrossRef](#)]
40. Nakanishi, N. Normalization condition and normal and abnormal solutions of the bethe-salpeter equation. *Phys. Rev.* **1965**, *138*, B1182. [[CrossRef](#)]
41. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
42. Wen, J.; Fang, X.; Cui, J.; Fei, L.; Yan, K.; Chen, Y.; Xu, Y. Robust Sparse Linear Discriminant Analysis. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *29*, 390–403. [[CrossRef](#)]
43. Sabzi, S.; Abbaspourgilandeh, Y.; Garciamateos, G. A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms. *Comput. Ind.* **2018**, *98*, 80–89. [[CrossRef](#)]
44. Sa, I.; Ge, Z.; Dayoub, F.; Upcroft, B.; Perez, T.; Mccool, C. DeepFruits: A Fruit Detection System Using Deep Neural Networks. *Sensors* **2016**, *16*, 1222. [[CrossRef](#)] [[PubMed](#)]
45. Zhang, Q.; Liu, Y.; Gong, C.; Chen, Y.; Yu, H. Applications of Deep Learning for Dense Scenes Analysis in Agriculture: A Review. *Sensors* **2020**, *20*, 1520. [[CrossRef](#)] [[PubMed](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).