Optik 127 (2016) 1354–1362

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.de/ijleo

Recognition of apple fruit in natural environment

Lv Jidong[a,*], Zhao De-An[b,*], Ji Wei[b], Ding Shihong[b]

[a] School of Information Science and Engineering, Changzhou University, Gehu Road, Changzhou 213164, Jiangsu Province, China
[b] School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China

ARTICLE INFO

Article history:
Received 28 January 2015
Accepted 27 October 2015

Keywords:
Apple
Overlapping fruits
Occluded fruits
Image recognition
Improved RHT

ABSTRACT

The recognition method for apple fruits in natural environment was developed. Firstly, the acquired apple images were executed characteristic analysis and classified by difference of light acceptance and growth state. Secondly, the OTSU dynamic threshold segmentation method with I_2 color characteristic in the $I_1I_2I_3$ color space was chosen through comparing the apple image segmentation methods based on the color feature in different color spaces. Nextly, the image perfection and noise removal were carried out for the above segmentation image. Thirdly, the apple fruit contour model was established. The apple fruits were recognized by edge detection and the improved RHT transformation method, the overlapped apples and severely occluded apples by the branches and leaves were respectively done the separation and restoration operations before they were recognized. Finally, some fruits recognition experiments were done for apples in three different states with the non-occluded, overlapped and severely occluded by branches and leaves. The results showed that this proposed method was feasible and can basically meet the requirements of robot picking.

1. Introduction

It is a principal problem to develop algorithms that allow the apple harvesting robot to directly, quickly and accurately recognize fruits in a real-time [1]. In the natural environment, apple fruit recognition is usually more difficult due to the influence of lights, branches and leaf masking and so on. Consequently, it is significant to recognize apple fruits in the natural environment. Apple visual characteristics in the natural environment may be divided into two categories – non-occluded fruits and occluded fruits. Occluded fruits could also be classified into overlapping fruits and fruits occluded by branches and leaves. The recognition method will be different for the different visual characteristics of apple fruit [2].

Existing research literature on apple recognition methods [3–7,27] reported an ideal condition that apple fruits weren't occluded. Compared with the recognition methods of apple fruits under the non-occluded case, the recognition problems of the occluded apple fruits have been rarely considered. Ref. [8] detailed a procedure for detecting apples in tree images using shape analysis. The cor of the procedure consists of a so-termed convexity test that identifies edges that could correspond to three-dimensional convex objects of a given size range from a much larger set of edges. Analysis based on the edges led to correct detection of 94% of the apples visible in the images. Ref. [9] recognized apple fruits using a genetic algorithm, running the genetic algorithm multiple times for images with overlapping apple fruits. The overall fruit recognition rate was 97%. Ref. [10] segmented apple images by a three layer BP neural networks model, fitting circles to fruits and obtained their positions in the image according to their contour curvature variances. Fruit (include slightly occluded fruit by branches and leaves) recognition rate was 93%. However, these methods still had some disadvantages; for example, they didn't solve for the recognition of severely occluded apple fruits by branches and leaves, and the recognition rate was influenced by edge fractures on apple fruit outlines. Moreover, they didn't determine total recognition times, and whether their designed methods could meet the recognition requirement. In previous work, we adopted a composite method based on region growing and colour properties to segment the apple image after performing image enhancement, and using (Support Vector Machine) SVM based on colour and shape characteristics to classify and recognize the fruit [11]. The overall recognition rate was 89% with an average processing time of 352ms. However, these apple images only included non-occluded fruits. It is necessary to introduce a better, new occluded fruit recognition method with higher recognition rate and lower processing times. The objective is to realize at least 90% of recognition rate and a processing time of less than 1 s for occluded fruit images.

* Corresponding author. Tel.: +86 13656124208.
E-mail addresses: vveaglevv@163.com (L. Jidong), Zhaodean228@126.com (Z. De-An).

http://dx.doi.org/10.1016/j.ijleo.2015.10.177
0030-4026/

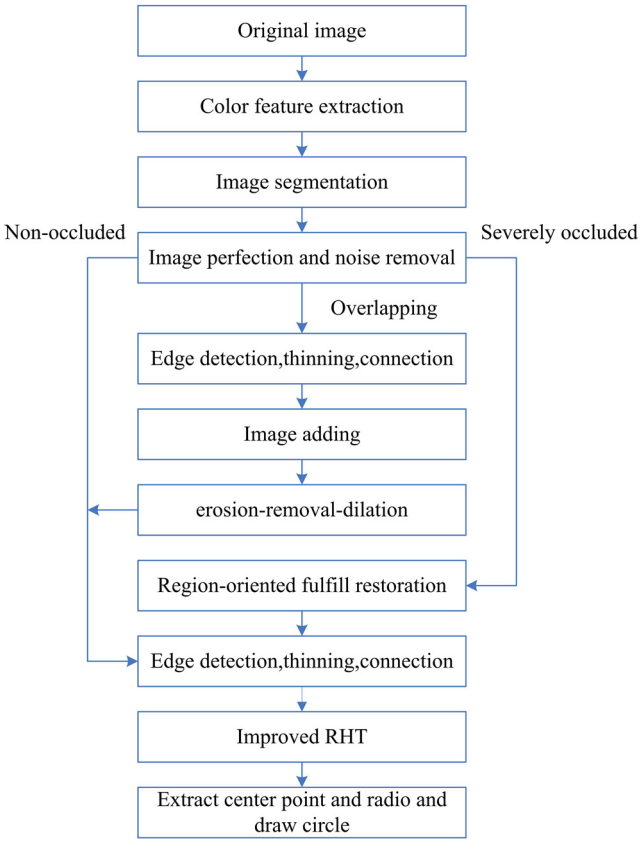


Fig. 1. The flow chart of apple fruits recognition.

This work proposes an apple fruit recognition method for overlapping apples and apples occluded by branches and leaves in the natural environment. The outline is organized as follows: in Sections 2.1–2.3, apple image acquisition, segmentation and perfection are presented; apple image recognition is presented in Section 2.4, mainly including fruits edge detection, fruits edge thinning and fruits edge connection; the test results are discussed to show the validation of the proposed method in Section 3; finally, the conclusions and suggestions for future research are drawn in Section 4.

2. Materials and methods

The total implementation process of apple fruits recognition is shown in Fig. 1. Apple fruit images were acquired, then we chose the segmentation method after comparing apple segmentation images based on colour features in different colour spaces. Image perfection and noise removal were carried out for the segmentation image. Then apple fruits were recognized through edge detection and the improved random Hough transformation, during which the problems of non-perfect edge and edge fracture were solved by edge thinning and edge connection. Overlapping fruit and fruit badly occluded by branches and leaves were separated and restored before being recognized. Fruit recognition experiments were conducted for 60 images containing 113 apple fruits with three different visual characteristics, respectively, i.e., non-occluded, overlapping and severely occluded cases. The program software of fruit recognition experiments was Matlab R2010b based on the following computer configuration: Intel (R), Core (TM) 2, Duo CPU E7300@2.66 GHz, 2 G (Memory), 320 G (HDD).

2.1. Apple image acquisition

The variety of apple tested in this study was Fuji, which is the most popular variety in China. Colour images of Fuji apple fruits in the orchard were acquired using a digital camera (Sony Cybershot) in the different times during the third week of October, the harvesting season. The colour signals from the camera were transferred as a 24-b red, green, blue (RGB) colour image data (320 pixel by 240 pixel in each colour band). The acquired images included the separate and non-occluded fruits, the overlapping fruits and the occluded fruits under different lighting conditions ((a) full sun, front lighting; (b)full sun, back lighting; (c) full sun, fruits in the shade). The fruit trees were randomly selected from the apple demonstration orchard of Feng Country Government, Xuzhou City, Jiangsu Province (longitude 116.57°E, latitude 34.79°N).

2.2. Apple image segmentation

2.2.1. Selection for colour space and feature

It was necessary to choose a suitable colour space and colour feature for image segmentation. Images of fruits (including branches, leaves and sky) in RGB colour space were transformed to *XYZ*, Lab and $I_1I_2I_3$ colour space.

RGB colour space was converted to *XYZ* colour space by the following set of equations [12]:

$$\begin{cases} X = 0.607R + 0.174G + 0.201B \\ Y = 0.299R + 0.587G + 0.114B \\ Z = 0.066G + 1.117B \end{cases} \quad (1)$$

RGB colour space was converted to Lab colour space by the following set of equations[13]:

$$\begin{cases} L = 0.2126R + 0.7152G + 0.0722B \\ a = 1.4749\,(0.2213R - 0.339G + 0.1177B) + 128 \\ b = 0.6245\,(0.1949R + 0.6057G - 0.8006B) + 128 \end{cases} \quad (2)$$

RGB colour space was converted to $I_1I_2I_3$ colour space by the following set of equations [14]:

$$\begin{cases} I_1 = \dfrac{R+G+B}{3} \\ I_2 = R - G \\ I_3 = 2R - G - B \end{cases} \quad (3)$$

2.2.2. Segmentation Method

The adaptability of the fixed threshold segmentation method was not strong, especially for the apple images with complex background. Therefore, we adopted a dynamic threshold segmentation method – OTSU [15,16]. For a certain image, it is assumed *T* was the segmentation threshold between the foreground and background in image; w_0 was the ratio of its foreground points and its total points; u_0 was the average grey of its foreground; w_1 was the ratio of its background points and its total points; u_1 was the average grey of its background. The total average grey value was obtained from Formula (4). *T*, traversed from the minimum grey value to the maximum grey value, was the best segmentation threshold when making the variance value σ^2 of the Formula (5) maximum. Usually the calculation cost of applying OTSU method was larger, so the variance value σ^2 of this work used the modified equivalent Formula (6).

$$u_T = w_0 \times u_0 + w_1 \times u_1 \quad (4)$$

$$\sigma^2 = w_0 \times (u_0 - u_T)^2 + w_1 \times (u_1 - u_T)^2 \quad (5)$$

$$\sigma^2 = w_0 \times w_1 \times (u_0 - u_1)^2 \quad (6)$$

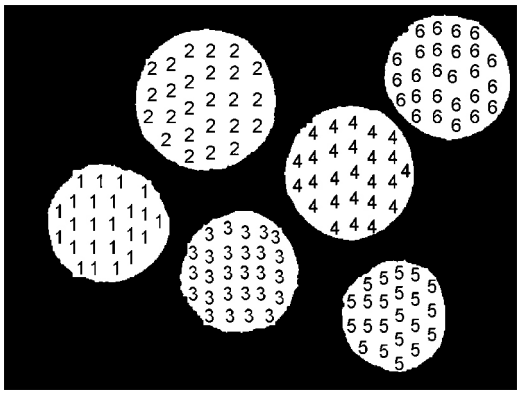


Fig. 2. The labelled connected regions.

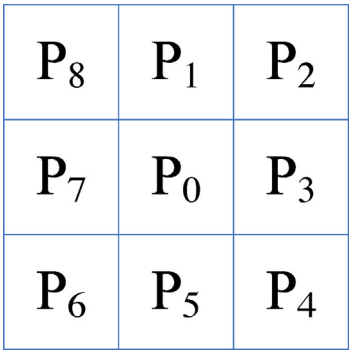


Fig. 3. Eight-neighbours label.

2.3. Apple image perfection and noise removal

There were many holes and even the large area loss in the tail of fruits due to the evident colour differences between the end calyx and other parts of apple fruits after image segmentation. In addition, there were still other small holes and split pieces in image, so image perfection and noise removal operations were made. The floodfill algorithm [17] was used to fill holes. For the large missing area of apple tail image, the eight-neighbours labelling method [18] was applied to label the connected image region in the image first. The different connected regions obained the only serial number from left to right. Fig. 2 showed the process. The number of pixels in each connected region was counted, and the regions with pixel number less than 100 were performed the morphological dilation operation with 2×2 square structure element. The holes were then filled using the floodfill algorithm again. There were other split pieces in images – such as the smaller fruit parts and the remaining branches and leaves – were expanded during the above expansion operation; therefore, the regions with less than 500 pixel were cleared, and the larger fruit parts in image reserved for the following recognition. The threshold 100 and 500 were the statistical results of the apple end calyx debris and the image noise after the expansion-filling, respectively, which should be adjusted according to the work distance of the different visual systems. In the end, the image was divided into the fruits and the background. The above combination method of image perfection and noise removal could be called the expansion-filling-removal method of the selective region.

2.3.1. Fruits edge detection

Fruit edges include valuable target boundary information. To detect the edges, the Canny operator was used [19]. An important advantage of the Canny operator is that the detected edge line is very thin. We obtained better edge connection using the Canny operator compared to other edge detection operators that we tested (Robert, Sobel, Prewitt, Kirsch operators).

2.3.2. Fruits edge thinning

The fruits edge obtained by Canny operator were not the edge with a single pixel level, especially in the corner points of image. To obtain the single-pixel precision edge, an extension of the method described by Zhang and Suen [20] was applied, as follows.

Step1: observing the 8 neighbourhood of the edge pixels (pixel value 1), let the selected pixel point as P_0, then the remaining points are numbered with the clockwise as P_i ($i = 1, 2, \ldots, 8$) (Fig. 3).

Step2: Traverse all edge pixel points according to the following two rules, respectively, by Fig. 3.

Rule 1:	(a) $2 \le A(P_0) \le 6$	Rule 2:	(a) $2 \le A(P_0) \le 6$
	(b) $1 \le B(P_0) \le 2$		(b) $1 \le B(P_0) \le 2$
	(c) $P_1 * P_3 * P_5 = 0$		(c) $P_1 * P_5 * P_7 = 0$
	(d) $P_3 * P_5 * P_7 = 0$		(d) $P_1 * P_3 * P_7 = 0$

(a)–(d) in Rule are logical-and relative. $A(P_0)$ is the numbers of nonzero point in 8 neighbourhood around P_0, $B(P_0)$ is change numbers from 0 to 1 in the sequence $P_1P_2P_3P_4P_5P_6P_7P_8$. $P_i * P_j * P_k$ is used as the logical product of these pixels. $2 \le A(P_0) \le 6$ shows that P_0 can be deleted if there are two to six adjacent pixels in the 3×3 area centering on it. If P_0 only has one adjacent pixel, it will be known as the target endpoint, which cannot be deleted. If P_0 has more than six adjacent pixels, then it isn't the target frontier point, which cannot be deleted either. In the traversal process, the points meeting the requirements are marked. They are removed after the second traversal process, i.e., they are set as 0.

2.3.3. Fruits edge connection

The edge outline with single pixel width could be obtained after edge detection and edge thinning. However, there were still some fracture edges in edge outline. Considering there are relations on the edge spread direction among the fracture edges, we achieved the connection of fracture according to these relations.

The edge connection was similar to the way of expansion operation. In the operation, the edge endpoints needed to be found first, and then the growth operation was performed towards the direction of endpoints. The detection and growth of edge endpoints were operated by the structural elements. The corresponding structural elements in Fig. 4, Fig. 4(a) and (c) were used for endpoint detection. Fig. 4(a) could be rotated to other directions by 45°. Fig. 4(c) could be rotated to other directions by 90°. Fig. 4(b) and (d) were used for growth operation, like Fig. 4(a) and (c), and could also be rotated to other directions by 45° and 90° rotation, respectively.

The main processing steps of edge connection were described as following:

Step1: The minimum enclosing rectangle frame method [21] is used to calculate the dominate area of apple fruits in the binary edge image, on basis of which the edge of the dominate area is expanded by a pixel to ensure that the endpoint detection will not exceed the frame range. The apple fruits are framed to decrease the following calculation cost of endpoint detection.

Step2: In the frame range, starting from the top left corner, the points are detected whether they are edge endpoints judging by Fig. 4(a) and (c) and their variant shapes after rotation. The variant shapes of Fig. 4(a) and (c) in horizontal and vertical directions should be used together. If matching the rotated Fig. 4(a) in the horizontal or vertical direction, the detected points will grow according to the rotated Fig. 4(b) in the corresponding direction. If matching the rotated Fig. 4(c) in the horizontal or vertical direction, the detected points will grow according to the rotated Fig. 4(d) in

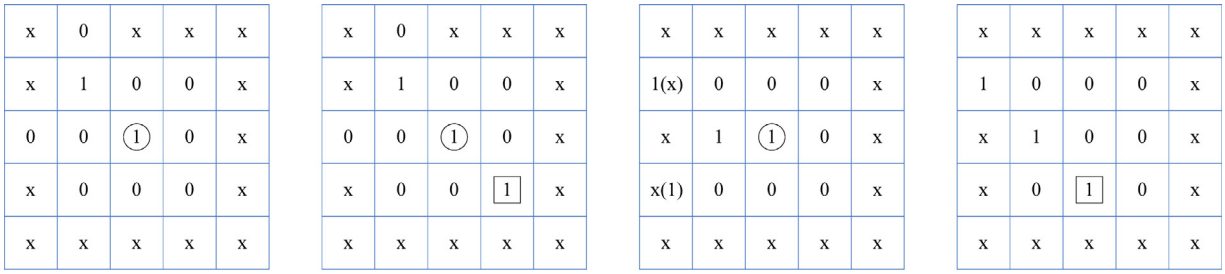


Fig. 4. Edge connection structure element models. (1), [1] and *x* represent the edge endpoint, the growing point and an arbitrary pixel value, respectively.

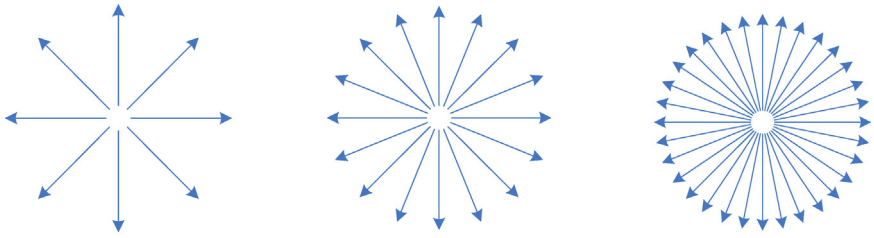


Fig. 5. Filling directions used in the region-oriented filling method.

the corresponding direction. Only Fig. 4(a) and the variant shapes of Fig. 4(a) are used in the other four directions. If matching, the detected points will grow according to the rotated Fig. 4(b) in the corresponding direction.

Step3: Testing will not be stopped until all pixel tests in the framed range are finished, and the same operations as step 2 are done from the bottom right corner in all framed range. The complete edge will be obtained when the operation is complete.

2.3.4. Occluded fruits restoration

Generally, the occluded fruits included two cases: the fruit was occluded by branches and leaves slightly; he fruit was occluded severely – i.e., an apple was split into several parts by branches and leave after edge detection. The method to recognize apple fruits occluded slightly was the same as that of apple fruits without being occluded. However, for apple fruits occluded severely, if the same recognition method was applied, the error of one apple with several positioning circles may occur after recognizing by the following improved RHT. So a region-oriented filling approach was proposed, making the several parts split by branches and leaves as a whole for facilitating the following recognition.

The steps of the region-oriented filling method were described as follows:

Step1: Framing the apple fruits in the dilated binary image (the fruit pixel is 1, and the background pixel 0) by the minimum enclosing rectangle method.

Step2: Start searching from a background pixel towards *n* directions in the framed region. If the pixel within the apple fruits area is found in any directions, then set this direction flag as 1 and stop searching. Otherwise, search till the boundary of the framed area.

Step3: Counting the direction flags of the background pixel in *n* directions. If the direction flags of more than *n*/2 directions are set, then fill the pixel value as 1. Otherwise keep the pixel value unchanged.

Step4: Searching other background pixels as Step2 and Step3 till all the background pixels are searched, and then continue to search other regions.

The value of *n* in the region-oriented filling method, depending on the actual implementation results, generally took 8, 16, and 32, as shown in Fig. 5.

2.3.5. Apple recognition based on the improved RHT

When the more complex images, with several circles and a certain amount of noise, were processed by RHT [22,23], a large number of invalid accumulations emerged due to the introduction of lots of invalid units in random sampling. To overcome the above drawbacks of RHT, an improved RHT method was applied to the recognition of apple images based on the literatures.

At the beginning, the mark of edge image of apple fruits traversed the pixels within their region to obtain the horizontal minimum enclosing rectangle as an effective image area. Through the above operation, the image was decomposed into several effective image areas, and apple fruits distributed in each effective image area. Because the range of parameters was greatly reduced, a large number of invalid units would not be introduced in random sampling to some extent. At the same time, the computational cost reduced greatly, and the processing speed would be improved too, with unaffected recognition results. And then three points on the edge of apple fruit were selected simultaneously to determine a circle by applying the geometric feature of circle, i.e., the perpendicular bisectors of any two non-parallel strings on the circumference intersect at the centre point. In comparison with the parameters obtained by solving the circle standard equations in the traditional RHT method, the parameters calculation process of the improved RHT method was simplified. The specific steps were as follows:

Fig. 6 shows p_1, p_2 and p_3 are not on the same line, the perpendicular bisector p_0p_0' and p_0p_0'' of the line p_1p_2 and p_2p_3

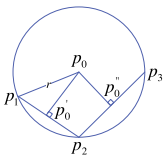


Fig. 6. Circle geometrical property.

cross at the circle centre, and the radius r is the length of the line p_0p_1 (or p_0p_2 or p_0p_3). Assuming the coordinates of p_1p_2, p_3 are (x_1, y_1), (x_2, y_2), (x_3, y_3), the midpoint coordinates of the strings p_1p_2 and p_2p_3 are $\left((x_1+x_2)/2,\ (y_1+y_2)/2\right)$ and $\left((x_2+x_3)/2,\ (y_2+y_3)/2\right)$, their slope are $k_1 = (y_2-y_1)/(x_2-x_1)$ and $k_2 = (y_3-y_2)/(x_3-x_2)$, respectively. So the following equations can be obtained.

$$\begin{cases} y - \dfrac{y_1+y_2}{2} = k_1\left(x - \dfrac{x_1+x_2}{2}\right) \\ y - \dfrac{y_2+y_3}{2} = k_2\left(x - \dfrac{x_2+x_3}{2}\right) \end{cases} \tag{7}$$

The keys to the equations are the coordinates (x_0, y_0) of the center point p_0:

$$x_0 = \frac{x_1^2 + y_1^2 - x_2^2 - y_2^2}{2g_f}(y_1 - y_3) - \frac{x_1^2 + y_1^2 - x_3^2 - y_3^2}{2g_f}(y_1 - y_2) \tag{8}$$

$$y_0 = \frac{x_1^2 + y_1^2 - x_3^2 - y_3^2}{2g_f}(x_1 - x_2) - \frac{x_1^2 + y_1^2 - x_2^2 - y_2^2}{2g_f}(x_1 - x_3) \tag{9}$$

$$g_f = (x_1 - x_2)(y_2 - y_3) - (x_2 - x_3)(y_1 - y_2) \tag{10}$$

The radius of circle is:

$$r = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \tag{11}$$

where g_f is the detection sign of three collinear points. If $g_f = 0$, the three chosen points are collinear, then they shall be given up.

In the practical application, in order to avoid the effect of the digital quantization error, we took d as the threshold. If $g_f < d$, the three chosen points were taken as collinear points, else the circle parameters were calculated according to Formulas (8), (9) and (11). In the parameter space, the above circle parameters were mapped and then accumulated. When the mapping ended, the maximum value would appear in corresponding point in the parameter space if there was a circle in image. The maximum value was detected. If the maximum value was greater than a certain threshold, the eventual circle parameters were determined, and the recognition of apple fruits completed by taking a circle.

3. Experimental results

Based on the above methods, Fruit recognition experiments were conducted in Jiangsu University. We analysed first the colour features of the fruits and background (including branches, leaves and sky) in the collected apple images based on *XYZ*, Lab and $I_1I_2I_3$ colour space. Fig. 7 shows apple fruits and their background, taking X–Y, a, I_2, I_3 as the segmentation colour feature, have the observably difference, propitious to segment out apple fruits from image.

Fig. 8 shows examples of segmentation using the fixed threshold method and the OTSU dynamic threshold method based on X–Y, a, I_2 and I_3. The chosen thresholds using the fixed threshold method from Fig. 7 were 10, 140, 35, and 80, respectively. There was over-segmentation using the fixed threshold segmentation method based on X–Y, a, I_2 and I_3 as seen for example in Fig. 8(b, d, f and h) where without any contextual information. The apple fruits were normally segmented out by using the OTSU dynamic threshold segmentation method based on X–Y, a, I_2 and I_3 (Fig. 8(c, e, g and i)); the OTSU dynamic threshold segmentation image based on I_3 included

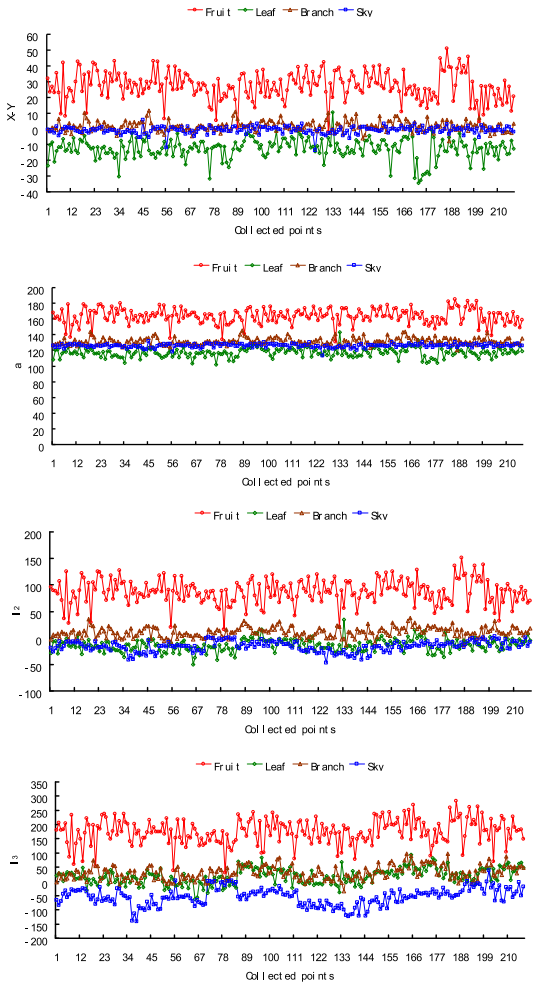


Fig. 7. X–Y, a, I_2, I_3 statistical analysis result.

more branches and leaves and was not as clean as the OTSU dynamic threshold segmentation image based on X–Y, a, I_2. Considering as well that the calculation cost of converting RGB to $I_1I_2I_3$ colour space was simpler than that from RGB to *XYZ* or Lab colour space, and the computing time of I_2 colour feature was less 16% at least than that of X–Y or a colour feature, we used the OTSU dynamic threshold segmentation method based on I_2 in $I_1I_2I_3$ colour space.

There were many holes and even the large area loss in the tail of fruits after image segmentation. In addition, there were still other small holes and split pieces in image, so image perfection and noise removal operations were made. Fig. 9 shows an achieved result of the dilation-filling-removal method of the selective region.

To detect the edges, the Canny operator was used as in Fig. 10. The edges thinning result is shown in Fig. 11. The dashed circles in Fig. 11(a) are the edges required to be refined. The dashed circles in Fig. 11(b) are the refined edges. Fig. 12 shows the effect of edge-connected graph. The dashed circles in Fig. 12(a) and (b) are the enlarged display of the fracture edge and the connected fracture edge, respectively. Here the contour information of apple fruits were obtained completely.

As for the separation of the overlapping apple fruits, first, the R channel grey image and the binary image were extracted from the segmented apple image. Then, the complete edge image could be obtained through edge detection, edge thinning and edge connection operations for the R channel grey image. Next, the edge image and the binary image were added to obtain a new binary image.

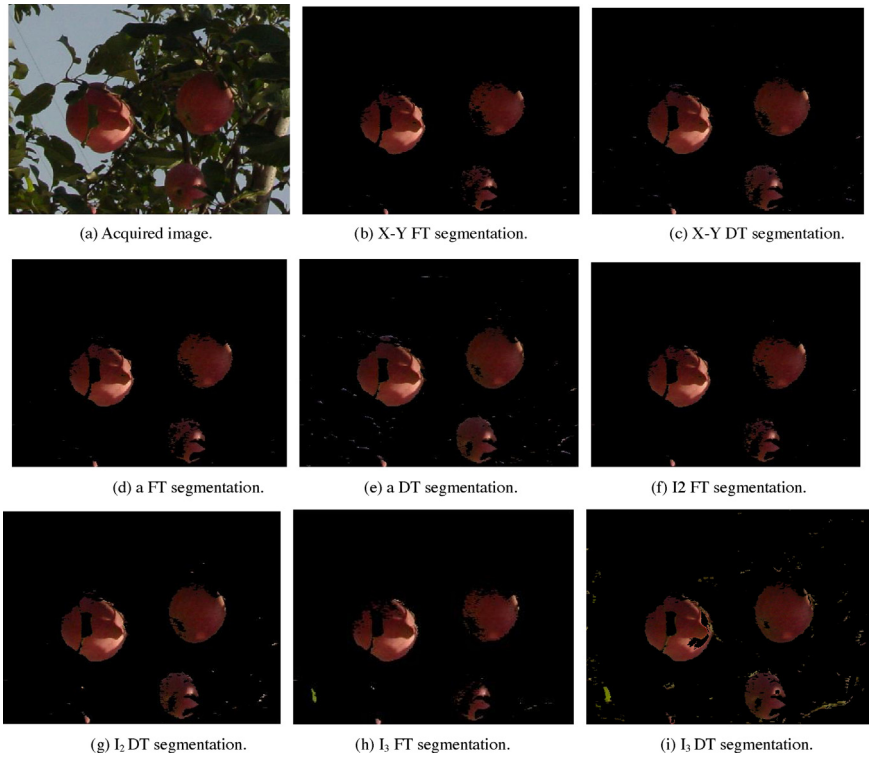


Fig. 8. The segmentation comparison chart of fixed threshold (FT) and OTSU dynamic threshold (DT) for apple images.

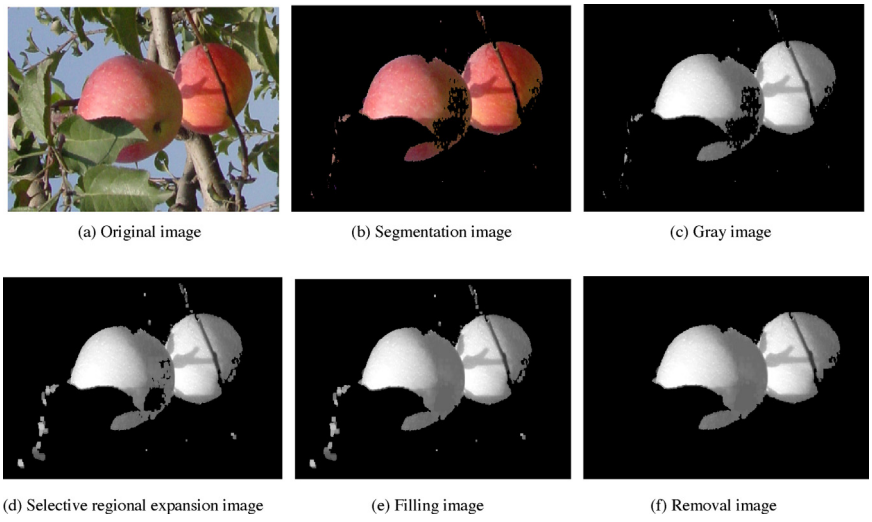


Fig. 9. The selective regional expansion-filling-removal images.

In the new binary image, the appeared separation was small, so the erosion-removal-dilation operation was performed to enlarge the separation gap in order to recognize fruits easier. The above method did not change fruits shape in comparison with Watershed algorithm and CFS algorithm [29], unconcerned with the occluded region size of the overlapping fruits in comparison with the separation method based on the edge curvature analysis [24]. Fig. 13 shows the separated apple fruits.

The region-oriented filling approach was used, making the several parts split by branches and leaves as a whole. Fig. 14 shows the effect of the region-oriented filling method. It could be seen from Fig. 14 that the actual experiment effect with $n = 16$ was better than that of $n = 8$. Although the testing result $n = 16$ looked worse than $n = 32$, the contour information had already meet the accurate recognition requirements of the follow-up RHT method. In this study, n was selected as 16 taking the reduction of computational cost into consideration.

The test results of apple fruits recognition are shown in Fig. 15, where the centre and centre coordinates are marked as red colour, and apple fruits in three images are recognized well. However, errors were introduced by using circles to fit apple fruits since the projected form of an apple is not a perfect circle. Similarly, there were errors when restoring apples occluded by branches and leaves. Table 1 shows that the above errors were not large. The maximum image coordinate error between the apple centre obtained by the RHT method and the real apple centre is 5 pixel.

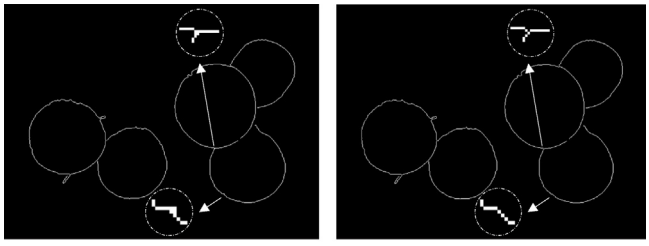
Table 1
Apple fruits recognition error.

Fruits situation	Fruits centre (pixel)	RHT centre (pixel)	Error (pixel)
Non-occluded	(149, 61)	(146, 60)	(−3, −1)
	(96, 170)	(93, 173)	(−3, 3)
	(265, 162)	(265, 161)	(0, −1)
Overlapping	(65, 136)	(65, 136)	(0, 0)
	(131, 161)	(131, 163)	(0, 2)
	(266, 68)	(263, 73)	(−3, 5)
Badly occluded	(216, 104)	(216, 104)	(0, 0)
	(249, 166)	(247, 165)	(−2, −1)
	(203, 183)	(203, 184)	(0, 1)

(a) Acquired image. (b) Segmentation image.

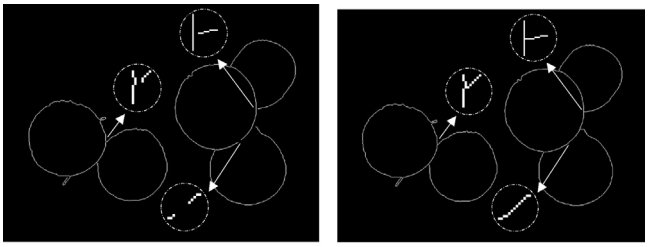
(c) Gray image. (d) Edge image.

Fig. 10. Edge detection image.



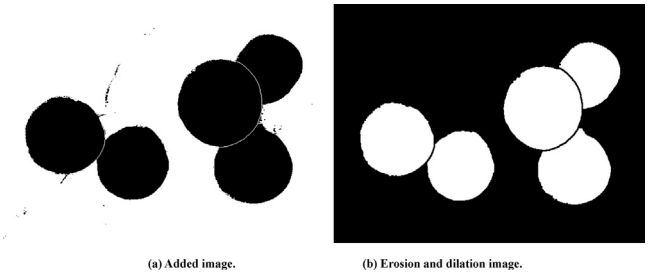
(a) Edge image. (b) Edge thinning image.

Fig. 11. Edge thinning image.



(a) Edge thinning image. (b) Edge connection image.

Fig. 12. Edge connection image.



(a) Added image. (b) Erosion and dilation image.

Fig. 13. Separation image of overlapped apples.

In order to test the recognition rate and recognition time of apple fruits in three different visual characteristics, 60 images with 113 apple fruits were selected for experiments. The relative errors of abscissa and ordinate between the apple centre obtained by the proposed method and the real apple centre are all less than 5%. This error is acceptable for us, because our end-effector of the apple harvesting robot is of spoon shape with the larger space when it is opened, and can grip fruit. In addition, the recognition time is achieved by carrying out the flow chart in Fig. 1, but the test time of the improved RHT method is unstable due to random sampling, each image was processed five times to get their average value. The results are shown in Tables 2 and 3.

Table 2 shows 100% of the recognition rate of the proposed method in this study for apple fruits in non-occluded and overlapping occluded states meets our requirement – more than 90%. For apple fruits occluded by branches and leaves, the recognition rate is 86%, which does not meet our requirement (more than 90%), but the proposed method in this study can partly solve the recognition problem on severely occluded apple fruits. The main cases of incorrect recognition were apples located close to each other and those badly occluded by branches and leaves. It needs to be further researched.

From the data analysis of recognition time in Table 3, the maximum recognition time of non-occluded apples is 0.65 s; the

Table 2
Apple fruits recognition rates.

Fruits situation	Number of images	Total number of fruits	Number of correctly identified fruit	Number of fruit missed	Recognition rate (%)
Non-occluded	20	35	35	0	100
Overlapping	20	56	56	0	100
Badly occluded	20	22	19	3	86

Table 3
Apple fruits recognition time.

Fruits situation	Maximum recognition time (s)	Minimum recognition time (s)	Average Recognition time (s)	Standard deviation of recognition time (s)
non-occluded	0.65	0.18	0.42	0.13
Overlapping	0.96	0.48	0.72	0.15
Badly occluded	0.90	0.61	0.77	0.09

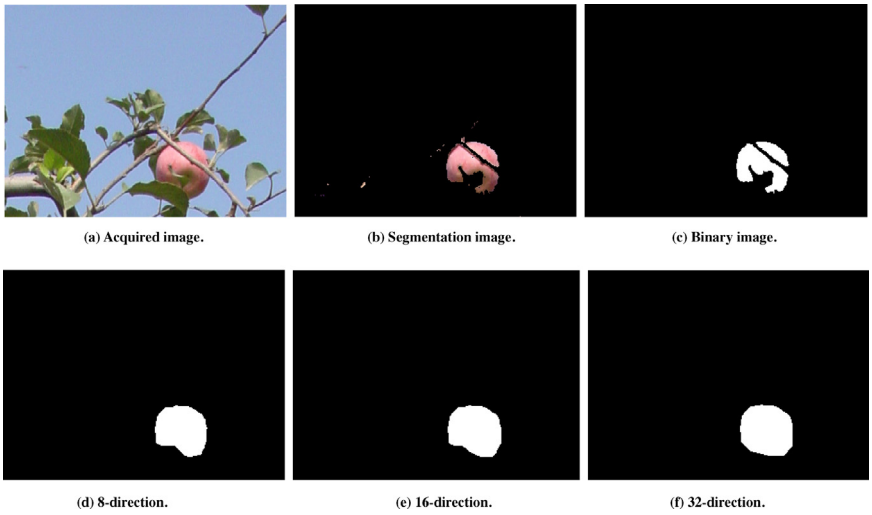


Fig. 14. Test image of the region-oriented filling method.

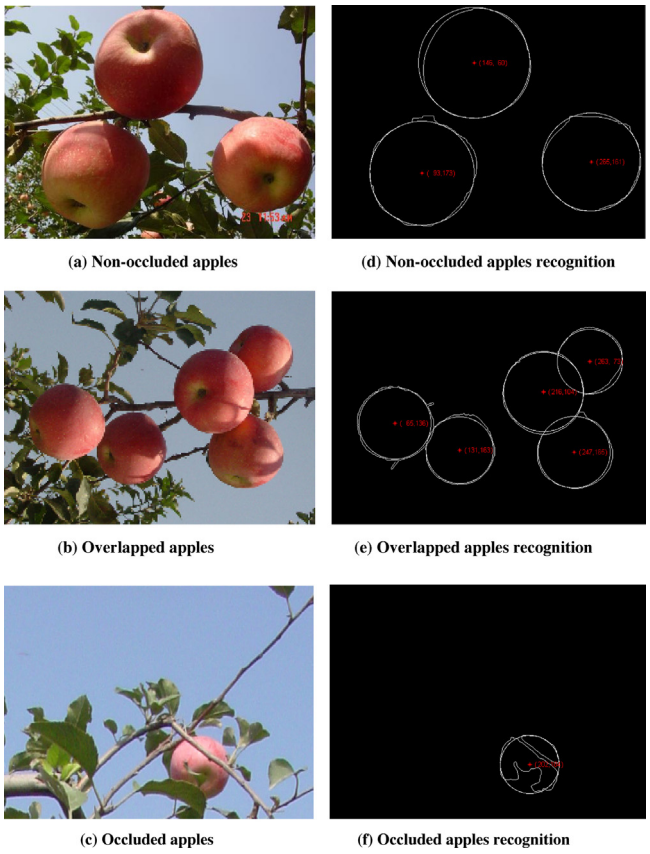


Fig. 15. Apple fruits recognition of the different visual characteristics.

minimum recognition time of that is 0.18 s; the average recognition time of that is 0.42 s (the standard deviation is 0.13 s). The maximum recognition time of overlapping apples is 0.96 s; the minimum recognition time of that is 0.48 s; the average recognition time of overlapping apples is 0.72 s (the standard deviation is 0.15 s). The maximum recognition time of badly occluded apples is 0.90 s; the minimum recognition time of that is 0.61 s; the average recognition time of that is 0.77 s (the standard deviation is 0.09 s). Their recognition time are all less than 1 s, which meets our requirement (less than 1 s).

4. Conclusions

This work studied the recognition of occluded apple under natural environment. For apple fruits in overlapping state, the processing was carried out through the method of adding the segmented binary image to the edge image, and the overlapping and adhering parts between apple fruits were separated. For apple fruits severely occluded by branches and leaves, the region-oriented filling method was used to restore apple fruits split into several parts by branches and leaves, and then the feature extraction and recognition for them were conducted.

The developed method resulted in recognition percentages of 100%, 100% and 86%, respectively, for non occluded fruit, overlapping fruit and fruit occluded by branches and leaves in 60 images containing 113 apple fruits. Recognition times for fruit images (groups of fruit) were, respectively, 0.42 ± 0.13 s, $0.72 \pm$ s (st.dev 0.15 s) and 0.77 ± 0.09 s. These results meet the goal set at the start of the project of less than 1 s for occluded fruit images.

The proposed method can provide a basis for the recognition of other spherical fruits with the significant colour differences. However, the recognition of clusters of touching fruit and improvements in recognition rates for fruit occluded by branches and leaves require further research. Algorithms and software programs need to be optimized to reduce the recognition time as much as possible.

Acknowledgments

This work was partly supported by Natural Science Foundation of Jiangsu Province under Grant BK20140266, Natural Science Research Program for Higher Education in Jiangsu Province under Grant 14KJB210001, Scientific Research Foundation for Changzhou University under Grant ZMF13020019.

References

[1] D.A. Zhao, J.W. Lv, W. Ji, Design and control of an apple harvesting robot, Biosyst. Eng. 110 (2) (2011) 112–122.
[2] A.R. Jimenez, A.K. Jain, R. Ceres, Automatic fruit recognition: a survey and new results using range/attenuation images, Pattern Recognit. 32 (10) (1999) 1719–1736.
[3] V. Leemans, H. Magein, M. Destain, Defect segmentation of Jonagold apples using colour vision and Bayesian classification method, Comput. Electron. Agric. 23 (1) (1999) 43–53.
[4] S.H. Zhang, T. Takahashi, Studies on automation of work in orchards, J. Jpn. Soc. Agric. Mach. 58 (1) (1996) 9–16.
[5] A. Plebe, G. Grasso, Localization of spherical fruits for robotic harvesting, Mach. Vision Appl. 13 (2) (2001) 70–79.

[6] D.M. Bulanon, T. Kataoka, Y. Ota, Segmentation algorithm for the automatic recognition of Fuji apples at harvest, Biosyst. Eng. 83 (4) (2002) 405–412.
[7] M. Huang, Apple fruit recognition based on template matching, Comput. Appl. Softw. 27 (5) (2010) 240–242.
[8] E. Kelman, R. Linker, Vision-based localisation of mature apples in tree images using convexity, Biosyst. Eng. 118 (2) (2014) 174–185.
[9] Y.S. Si, J. Qiao, G. Liu, Recognition and shape features extraction of apples based on machine vision, Trans. Chin. Soc. Agric. Mach. 40 (8) (2009) 161–165.
[10] Y. Xun, X. Chen, W. Li, Automatic recognition of on tree apples based on contour curvature, J. Jiansu Univ. (Nat. Sci. Ed.) 28 (6) (2007) 461–464.
[11] W. Ji, D.A. Zhao, F.Y. Cheng, Automatic recognition vision system guided for apple harvesting robot, Comput. Electr. Eng. 38 (5) (2012) 1186–1195.
[12] D.C. Tseng, C.H. Chang, Colour segmentation using perceptual attributes, in: IEEE International Conference on Pattern Recognition, 1992, pp. 228–231.
[13] P.J. Baldevbhai, R.S. Anand, Color image segmentation for medical images using L^*a^*b colour space, J. Electron. Commun. Eng. 1 (2) (2012) 24–45.
[14] A. Koschan, Chromatic block matching for dense stereo correspondence, in: Proceedings of the 7th Intemational Conference on Image Analysis and Processing, 1993, pp. 641–648.
[15] N. Otsu, A threshold selection method from grey-level histograms, IEEE Trans. Syst. Man Cybern SMC-9 (1) (1979) 62–66.
[16] D.F. Zhang, Detailed Description of Matlab Digital Image Processing, Publishing House of Electronics Industry, Beijing, China, 2010.
[17] G. Bradski, A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, Tsinghua University Press, Beijing, China, 2009.
[18] S.N. Mark, S.A. Alberto, Feature Extraction and Image Processing, second ed., Publishing House of Electronics Industry, Beijing, China, 2010.
[19] C.G. Rafael, E.W. Richard, Digital Image Processing, second ed., Publishing House of Electronics Industry, Beijing, China, 2005.
[20] T.Y. Zhang, C.Y. Suen, A fast parallel algorithm for thinning digital patterns, Commun. ACM 27 (3) (1984) 236–239.
[21] J.D. Lv, D.A. Zhao, W. Ji, Dynamic recognition of oscillating fruit for harvesting robot, Trans. Chin. Soc. Agric. Mach. 43 (5) (2012) 173–178, 196.
[22] L. Xu, E. Oja, P. Kultanen, Randomized Hough transform (RHT): basic mechanisms, algorithms, and computational complexities, Comput. Vision Graphics Image Process: Image Understanding 57 (2) (1993) 131–154.
[23] M. Silveira, An Algorithm for the Detection of Multiple Concentric Circles, Pattern Recognition and Image Analysis, Lecture Notes in Computer Science, vol. 3523, Springer Berlin, Heidelberg, 2005.
[24] Y. Xiang, Y.B. Ying, H.Y. Jiang, Recognition of overlapping tomatoes based on edge curvature analysis, Trans. Chin. Soc. Agric. Mach. 43 (3) (2012) 157–162.
[27] E.A. Parrish Jr., A.K. Goksel, A camera model for natural scene processing, Pattern Recognit. 9 (3) (1977) 131–136.
[29] T.J. Zhang, T.Z. Zhang, L. Yang, Comparison of two algorithms based on mathematical morphology for segmentation of touching strawberry fruits, Trans. Chin. Soc. Agric. Eng. 23 (9) (2007) 164–168.