Recognition and Location of Fruit Objects Based on Machine Vision

Hui Gu, Yaya Lu, Jilin Lou, and Weitong Zhang

Information Engineering College, Zhejiang University of Technology,
310014, Hangzhou, China
gh@zjut.edu.cn, {oo327, phonixlou, seasonzwt}@163.com

Abstract. This paper discussed the low level machine vision on fruit and vegetable harvesting robot, introduced the recognition and location of fruit and vegetable objects under nature scenes, put forward a new segmentation method combined with several color models. What's more, it presented a novel conception for the determination of the abscission point, successfully resolved the location of center and abscission point when the fruit were partially occluded. Meanwhile, by the technique of geometry, it settled the locations of the abscission point when the fruit grew askew. It proved good effect under the nature scene.

Keywords: Machine vision, fruit object, recognition, location.

1 Introduction

During the process of human conquering the Nature, rebuilding the Nature and promoting the society, humans are facing the problem of ability limitation. As a result, humans have been seeking for the robots to substitute the man to complete complicated tasks, and the intelligent robot is the best choice.

As we all know, vision is the main way of humans apperceiving the world. About *80%* information is got through vision. So, it is vital to grant vision function for intelligent robots. Here, we can define the machine vision as follows: it is able to produce some description about the content of the image after processing the input image [1].

There are many fields related with machine vision. So, it also has a wide application in various aspects, from medical image to remote sensed image, from industrial inspection to agricultural areas, etc.

The fruit and vegetable harvesting robot which we are going to discuss is one kind of automatic mechanical harvesting systems possessing the perceptive ability, can be programmed to harvest, transfer and pack the crops [2]. During the process of harvesting, the chief problem of the vision system is to recognize and locate the fruit object [3]. Here, recognition means segmentation of the fruit objects from the complicated background [4]. And location includes two aspects: location of the fruit center and abscission point.

Z. Pan et al. (Eds.): ICAT 2006, LNCS 4282, pp. 785 – 795, 2006.

Recently, there're many researches about fruit and vegetable harvesting robot based on machine vision [5][6]. Cai Jian-rong presented the machine vision recognition methods under the nature scene. Using the Otsu algorithm, it got the segmentation threshold automatically and extracted the target [7]. Miyanaga introduced the seeding grafting technique based on machine vision and the robot invented by them has been put into production [8]. Slaughter D.C set up one orange classier model by using the color feature in the chromatic digital image [9].

Among these researches, there have been many methods of extracting the fruits from complicated nature scene. But the basic conception is extracting the fruit object by converting one color model to another one which is easier to process or much more suitable for the case. However, still, there are two problems remain unsettled: 1) How to determine the abscission point when the fruits grow askew; 2) How to determine the center and abscission point when there are so many fruit overlapped each other that it is impossible to detect the whole edge. If both of the problems remain unsettled, it means the harvesting may be a failure.

And, what is more important, there is only about *40%* of the fruit and vegetable is visible in the orchard[10], which means about *60%* objects are partially occluded or completely occluded . Generally, the agricultural robots are fit with fans so as to blow the leaves covering the fruit. So, for the fruit occluded completely, it may be partially resolved in this way.

So, in the paper, we only discussed the problem of the fruit partially occluded, in particular, the case that one fruit overlap another one. As a whole, the problem we are to discuss belongs to the low level machine vision, and is one of the key steps in the machine vision.

2 Methodology Used in the Paper

2.1 Main Idea

From the analysis above, we knew, in order to segment the fruit from leaves and branches, we should use color model suits certain situations. The RGB color model commonly used is not suitable for the orchard images. Because in RGB color space, the tricolor (RGB) not only represent the hue value, but also represent the brightness. So, the change of the outward illumination may add the difficulty of the recognition, so RGB is undependable in the process of the segmentation. In order to make use of the fruit's clustering feature in hue space, we need to separate the hue and brightness information. We can achieve this goal by transferring the RGB to the models which separate hue and brightness.

2.2 Color Models

We use three types of color models in the paper. The first one is *LCD (luminance and color difference)* model. There are four color attributes in this model, including brightness information Y, color difference of red, C_r, color difference of green C_g, color difference of blue C_b. The transform formula is as follows:

$$\begin{cases} Y = 0.299R + 0.587G + 0.114B \\ C_r = R - Y \\ C_g = G - Y \\ C_b = B - Y \end{cases} . \quad (1)$$

During the process of experiment, we found that the color difference of red of fruit is much higher than that of leaves or branches, even the unripe fruit, such as unripe tomato that would be referred later. So we only have to consider about the color difference of red C_r.

The second model we used is Normalized RGB. The diagram was used to represent the color properties of the three portions. The transform formula is defined as follows:

$$\begin{cases} r = R/(R+G+B) \\ g = G/(R+G+B) \\ b = B/(R+G+B) \end{cases} . \quad (2)$$

it is obvious it satisfies: $r + g + b = 1$ ·

Combined the advantages of the above two models, we can conclude the third color model called *LHM* in this paper. Choosing Y and C_r from the first color model, r and g from the second model; we can construct the formula as follow:

$$\begin{cases} Y = 0.299R + 0.587G + 0.114B \\ C_r = R - Y \\ r = R/(R+G+B) \\ g = G/(R+G+B) \end{cases} . \quad (3)$$

3 Segmentation

Under the nature scene of the orchard, the factors containing the non-uniform illumination, the occlusion of the leaf and branch all make it more difficult to segment. At present, we can classify the chromatic image segmentation into three classes: (1) Segmentation based on threshold; (2) Segmentation based on edge inspecting and area growing; (3) Segmentation based on color clustering [11].

3.1 Clustering and Classifier

The primary conception of clustering is to distinguish the different objects which include different classes of objects and different parts of the same object [12]. All classification algorithms are based on the assumption that the image in question depicts one or more features and that each of these features belongs to one of several distinct and exclusive classes.

The traditional way of classifier comprises two phases of process: training and testing. In the initial training phase, characteristic properties of typical image features are isolated and, based on these, a unique description of each classification category, *i.e.* training class, is created. In the subsequent testing phase, these feature-space partitions are used to classify image features.

In the experiment, we sampled *60* pixels of leaf, branch, and fruit respectively and constructed a classifier. Adopting two feature patterns m and n, we formed the decision functions: $f(m,n) = am + bn + c$, where a, b, and c are arbitrary constants as long as the points on the line satisfies the condition $f(m,n) = 0$. Here, feature pattern may be color, shape, size, or any properties of the objects. According to the decision functions $f(m,n) > 0$ or $f(m,n) < 0$, we can divide the image into two parts as shown in Fig 1:

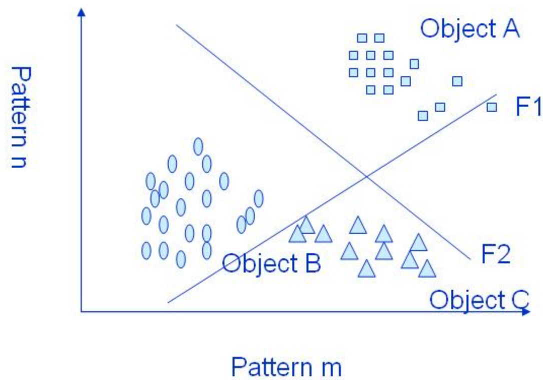


Fig. 1. Model of classifier

3.2 Segmentation of the Fruit Objects

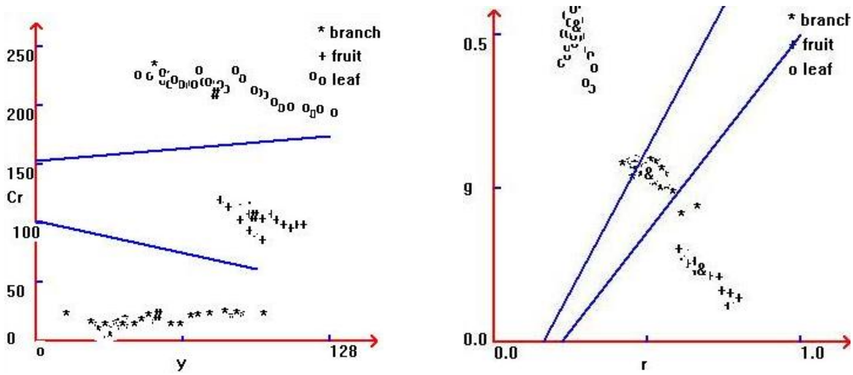
In this study, we adopted the segmentation method of several thresholds. The thresholds are derived from the above three models of the image using the decision functions. According to the above paragraphs, we could get three decision functions: the first function, *F1*, separated the fruit portion and the leaf portion, the second function, *F2*, separated the fruit portion from the branch portion, and the third function *F3*, separated the leaf portion from the branch portion. But, on the basis of the request of the experiment, we only have to segment the fruit from the background, and the leaf and branch portions were regarded as background. So, there was no need to consider *F3*.

3.3 Analyzing the Image Using the LCD Model

It is obvious that the fruit, leaf and branch had the different brightness and color difference of the red. So, sampled *60* pixels of the fruit, leaf and branch to train, from Fig2(a), we knew the distance between the mean values of the fruit object and that of the branch and leaf was rather great, so it was appropriate to use the minimum distance classifier.

From the training set, we could get the decision functions according to the minimum distance classifier as follows:

$$\begin{bmatrix} F_{leaf\,/\,fruit} \\ F_{fruit\,/\,branch} \\ F_{leaf\,/\,branch} \end{bmatrix} = \begin{bmatrix} -17.12 & 104.19 & -15867.4 \\ 41.16 & 80.59 & -8993.86 \\ 24.14 & 184.78 & -24861.26 \end{bmatrix} \begin{bmatrix} Y \\ C_r \\ 1 \end{bmatrix}. \tag{4}$$



(a) The training result of LCD **(b) The training result of NRGB**

Fig. 2. The experimental training result of different color models

3.4 Analyzing the Image Using the Normalized RGB Model

In the same way, we got the training set shown in Fig2 (b) and constructed the decision functions as follows:

$$\begin{bmatrix} F_{leaf\,/\,fruit} \\ F_{fruit\,/\,branch} \\ F_{leaf\,/\,branch} \end{bmatrix} = \begin{bmatrix} -0.358 & 0.33 & 0.101 \\ -0.11 & 0.106 & 0.037 \\ -0.148 & 0.162 & 0.054 \end{bmatrix} \begin{bmatrix} r \\ g \\ 1 \end{bmatrix}. \tag{5}$$

3.5 Analyzing the Image Using the LHM Model

Combined the advantages of two models, we could get better result from the intersection of two models. Additionally, the area of blemish points produced in the process of segmentation was rather small and was easy to erase, so the results of intersection did not affect the recognition at all. Results of segmentation by using the three models are as follows:

4 Connected Component Labeling

From Fig3, we can find that the objects obtained from the above steps still have some blemish points, which would disturb the normal recognition. Adopting 8-connected component to label the area, we can get several label values after labeling. So calculated the areas of each label value, we reserved the biggest area, and removed the others. For the further processing, we extracted the edge of the connected component; the result is shown in Fig 4.

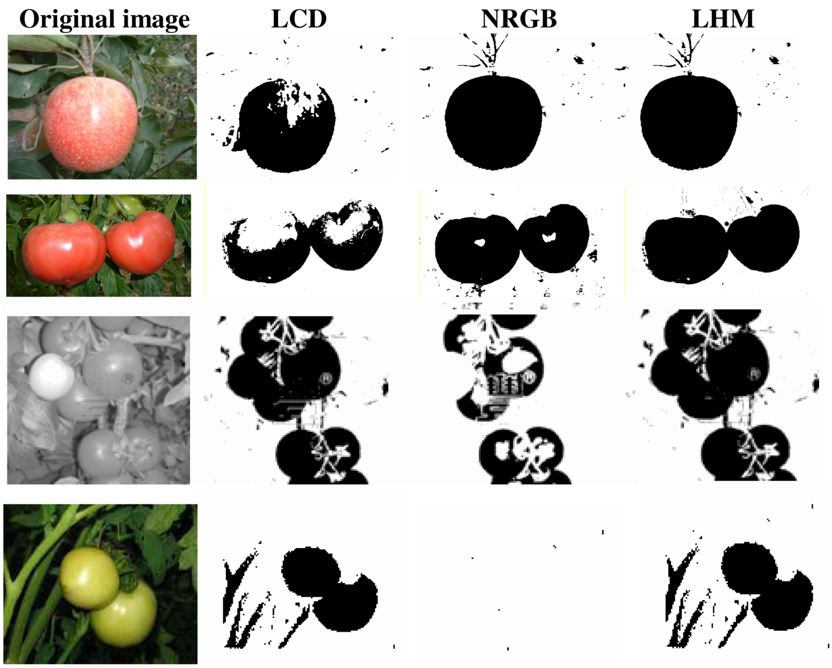


Fig. 3. Results of segmentation using three color models

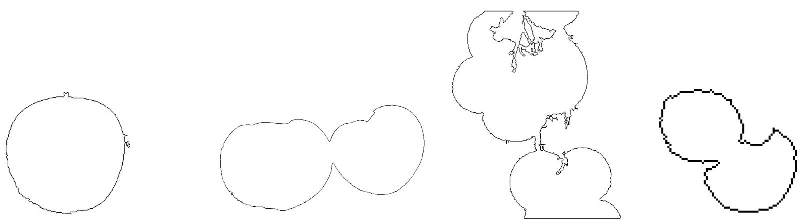


Fig. 4. Edge of the connected component

5 Location of Center and Abscission Point

For the spherical fruit and vegetable, such as apple, orange and tomato, the two dimension- graphics of these images seem to have a shape of circle because the similarity between them is high to *98%* [13]. So, we can simplify the spherical fruits into the problems of the circles.

Unfortunately, most of the paper did not cover the situation when several fruit objects overlapped each other, because the location and picking in such case is difficult. But out of common sense, we could understand the problem as follows: the robot pick the fruit one by one, so it is not necessary to locate the centers of the fruit set obtained from the image acquiring system at the same time. After one fruit was picked, the positions of left fruit objects would be changed accordingly due to the affection of the position and gravity. Under such understanding, we can assurance that locating one center of the fruit in the set is enough. In this way, we can shorten the processing time and simplify the processing step. So, we could claim that the problem lies in which fruit is to pick first. We can give a constraint that the robot always picking the highest fruit at first, which can be easily done by drawing horizontal tangent. And if there are several fruit objects intersect with the horizontal tangent, we can take the first point as the valid point called A from left to right as a rule. Then drew the vertical lines from the left and right at the same time and then got the points of intersection B and C and calculated $|AC|$ and $|AB|$. In the same way, drew a horizontal line from the bottom and got the intersection D, so the exterior rectangle is shown in Fig 5.

Suppose $|AB|>|AC|$, consider all the points in B->A->C->D, give up considering the points in B—>D. Because the points in set B—>D have the least probability located in the circles passing through point A, so give up considering these points can decrease the computational complexity too. According to the experience, for one kind of the fruit or vegetable, we often can form a model for them. For instance, the tomatoes we experimented can be modeled as Fig6 (a), and apples can be modeled as Fig6 (b). Point O is the center, Point A is the intersection of a horizontal line passing through the center as the leftmost outline of the fruit, and point B is the abscission point. The angle OAB is defined as α. measured the angle of 100 tomatoes and apples; we can get the mean angel value, for example, the tomatoes' α≈48°, and apples' α≈51°. So, as long as the fruit nutate naturally, we can get the center and abscission point by the model.

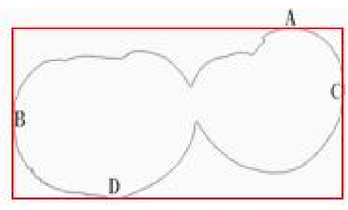
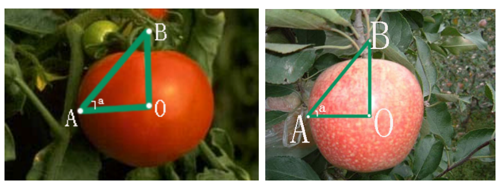


Fig. 5. Exterior rectangle



(a) **(b)**

Fig. 6. Models of fruit

5.1 Center

We have mentioned above that these fruit and vegetable can be simplified as circles, so the determination of the center of the fruit equals to the determination of the center of the circle. Recently, most researches on center are based on the improvement of the Hough transform [14]; others are based on the geometry methods [15]. Yet, the computational complexity of these methods is rather huge, which result in slow speed of process. As a consequence, it doesn't fit the harvesting machine vision that demands real-time performance.

Consider the points in B->A->C->D of Fig 5, let point A as the starting point, select four points v_1, v_2, v_3, v_4 at random, you can choose A as the constant point that the circle must pass through. According to the theorem that three noncollinear points can determine one circle, we know that four edge points can generally determine four circles: C_{123}, C_{234}, C_{124}, C_{134}. And the circle can be written as:

$$2ax + aby + d = x^2 + y^2 . \tag{6}$$

And here it satisfies:

$$d = r^2 - a^2 - b^2 . \tag{7}$$

So, on the condition that three points are noncollinear, we can get one circle. Note this circle as $C_{123,}$ center as (a_{123}, b_{123}) , radius as r_{123} , it can be calculated the center and radius as follows [16]:

$$a_{123} = \frac{\begin{vmatrix} x_2^2 + y_2^2 - (x_1^2 + y_1^2) & 2(y_2 - y_1) \\ x_3^2 + y_3^2 - (x_1^2 + y_1^2) & 2(y_3 - y_1) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))} . \tag{8}$$

$$b_{123} = \frac{\begin{vmatrix} 2(x_2 - x_1) & x_2^2 + y_2^2 - (x_1^2 + y_1^2) \\ 2(x_3 - x_1) & x_3^2 + y_3^2 - (x_1^2 + y_1^2) \end{vmatrix}}{4((x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1))} . \tag{9}$$

$$r_{123} = \sqrt{(x_i - a_{123})^2 + (y_i - b_{123})^2} \cdot \tag{10}$$

Let v_4 $(x_4; y_4)$ be the fourth edge pixel; then the distance between v_4 and the boundary of the circle C_{123} is denoted by:

$$d_{4->123} = \left| \sqrt{(x_4 - a_{123})^2 + (y_4 - b_{123})^2} - r_{123} \right| \cdot \tag{11}$$

If v_4 exactly lies on the circle$C_{123},$ the equation above equals 0. But due to the images acquired are digital, so it is hard to assure the point lie on the circle exactly. Therefore, the goal of circle detection is to detect a set of edge pixels which lie not exactly but roughly on a digital circle. For convenience, we denote the circle which passes through v_i, v_j, v_k by C_{ijk} and its center and radius are denoted by $(a_{ijk}, b_{ij\mathrm{k}})$ and r_{ijk}, respectively. Let the distance between v_m and C_{ijk} be denoted by:

$$d_{m->ijk} = \left| \sqrt{(x_m - a_{1ijk})^2 + (y_m - b_{ijk})^2} - r_{ijk} \right| \cdot \tag{12}$$

If we find one distance is smaller than given threshold T_d, we claim that this point lie on the circle. Here, we also have to guarantee the distance between any two points of the three points selected randomly should be greater than the given threshold T_a. If two points are too close, the circle may be not a real one. For the example shown in Fig 7, point v_1, v_2,v_3 lie on the boundary of true circle, but undesirable case occurs when v_2 and v_3 are too close, the circle determined by v_1, v_2 and v_3 differs from the true circle. To avoid such case, we should restrict the distances between any two points be greater than a certain threshold T_a.

The following step is to find which circle possesses the most valid points among all true circles. We set a counter $C=0$ for this possible circle how many edge pixels lie on the possible circle. For any pixel in the set V, we can calculate $d_{m->ijk}$ If $d_{m->ijk} <= T_d$, we increment the C by one and remove the pixel v_m from V. Otherwise, we proceed to the next pixel. We continue the above process until all the edge pixels in V have been examined. Note n_p is the number of the pixels lies on the possible circle. If n_p is greater than the global threshold T_g, call this circle a true one, otherwise it is a false one and return all the points to the set V. Then traverse the set n_p, we get the biggest one, and let the corresponding circle as the closest circle.

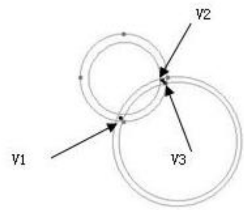


Fig. 7. Example of undesirable Circle

By the way, we should normalize the mentioned global threshold T_g. Because the circles with different radium have different circumferences. Thus, employing some large global threshold T_g is unfair to those circles with small radium. Since any circle in a digital image has a finite radius, the number of pixels on the boundary of a circle is estimated to be $2\prod r$. Hence, when there are n_p edge pixels lying on the possible circle C_{ijk} and the ratio of n_p over the theoretical value $2\prod r_{ijk}$ is larger than the given ratio threshold T_r , we claim that the possible circle is a true circle. Otherwise, the possible circle is a false circle and we return those n_p edge pixels into the set V.

For the original image in Fig 3, the effects are shown in Fig 8.

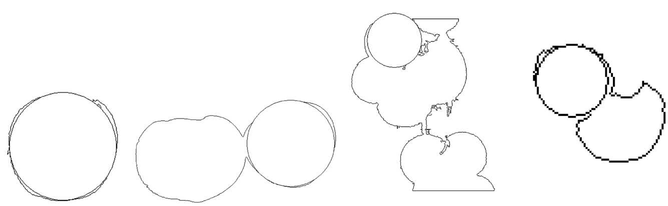


Fig. 8. The closest circle

5.2 Abscission Point

From the analysis, we found the center played an important role in recognition. But due to the randomicity of the development of the fruit, especially when there are several fruit overlap each other. It is common to find that the abscission point deviates from the vertical line passing through center. Take the right tomato in *Fig 3* for example, it is evident the abscission deviates a lot. So, we have to revise the model.

Meanwhile, we could observe the exterior rectangle of the fruit nutate naturally in Fig 9: there is one intersection on each edge, but the top intersection, bottom intersection and the center are collinear nearly. So it is easy to get the abscission point due to the fact that the abscission is on the vertical line passing through the center, and the deviation is too small to ignore.

In addition, we can observe that when the abscission point has a deflection angle, the four intersections in exterior rectangle will be changed correspondingly as shown in Fig 10. Assume point O as center, F as the abscission point, E is the intersection between the vertical line passing through O and the exterior. In true life, we know the length of |OF| almost equals to the radius of the fruit and we have the knowledge that $|OA|=r$. So, we could easily get $\angle AOE \approx \angle EOF$. Hence, if we can get the coordinates of O and A, we can easily get F. So the correctness of abscission relies on the location of center O. The experimental results are shown in Fig 11 denoted in single lines.

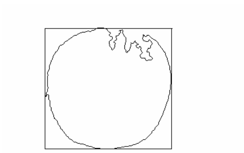


Fig. 9. The exterior rectangle of fruit nutate naturally

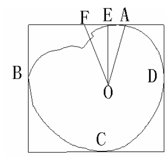


Fig. 10. The exterior rectangle of fruit grow askew

Fig. 11. The abscission point of fruit

6 Conclusions

This paper discussed the low level machine vision on fruit and vegetable harvesting robot, introduced the recognition and location of fruit and vegetable objects under nature scenes, put forward a new segmentation method combined with several color models. What's more, it brought forward a novel conception for the determination of the abscission point, resolved the location of center and abscission point when fruit was partially occluded successfully. Meanwhile, by the technique of geometry, it settled the locations of the abscission point when the fruit grew askew. It proved good effect under the nature scene. The accuracy of recognition was high to *94.38%*, the location of center was high to *92.6%* and the abscission point was high to *95.65%*.

References

1. Milan Sonka, Vaclav Hlavac, Roger Boyle. Image Processing, Analysis and Machine Vision. 2ndVersion. Posts and Telecom press. 2003.
2. Edan Y, Gaines E. Systems engineering of agricultural robot design IEEE Transactions on Systems ,Man , and Cybernetics ,1994 ,24 (8) :1259- 1265.
3. Tang Xiu-ying, Zhang Tie-zhong. Robotics for fruit and vegetable harvesting.Robot, 2005,1(27):90—95.
4. Gu Hui, Cheng Guangyi, Lu Yaya. A curve fitting method based on the direction tracing. Acta Electronica sinica(English Version). 2006 special issue 4.
5. A.R. Jiménez, R. Ceres and J.L. Pons. A survey of computer vision methods for locating fruit on trees. Transaction of ASAE, 2000,43(6):1911-1920.
6. K.F. Sanders. Orange harvesting systems review Biosystem Engineering. 2005,90(2),115-125.
7. Wang Yaqin, Gao Hua. Study on the segmentation and orientation of fruit image under natural environment. Computer Engineering. 2004,30(13):128.
8. Miyanaga T , Fukumo i I, Susaw a K, et al. Technical report of the institute of agricultural machinery. Omiya, Saitama, Japan. 1998.
9. Slaughter D, Harrel R. Discriminating fruit for robotic harvest using color in natural outdoor scenes. Transactions of the ASAE, 1989,32(2):757-763.
10. F. Juste and F. Sevilla, Citrus: A European project to study the robotic harvesting of oranges, in Proceedings, 3rd Int. Symp. Fruit, Nut and Vegetable Harvesting Mechanization, Denmark-Sweden-Norway, 331-338 (1991).
11. Nikhil R ,Pal Sankar K. A Review on image segmentation techniques. Pattern Recognition, 1993,26(9) :1277～1294.
12. REN Jing. Improved minimum distance classifier-weighted minimum distance classifier. Computer application.2005,25(5):992—994.
13. Zhao Jing. Stem recognition and fruit determination in fruit shape recognition. Journal of Shangdong University of Technology(Sci &Tech).2004,18(5):28.
14. Dimitrios Ioannou, Walter Huda. Circle recognition through a 2D Hough Transform and radius histogramming. Image and vision computing. 1999,17:15-26.
15. Heung-Soo Kim, Jong-Hwan Kim. A two-step circle detection algorithm from the intersecting chords. Pattern recognition letters. 2001,22: 787-798.
16. Teh-Chuan Chen, Kuo-Liang Chung. An Efficient Randomized Algorithm for Detecting Circle. Computer vision and image understanding. 2001,83:172-191.