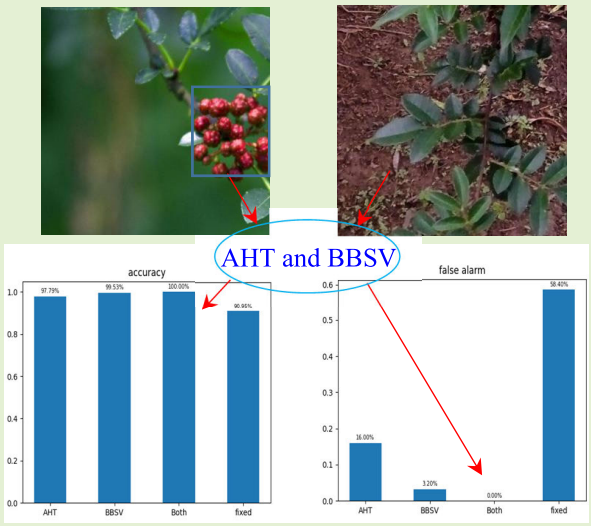
Zanthoxylum bungeanum Fruit Detection by Adaptive Thresholds in HSV Space for an Automatic Picking System

Longke He, *Member, IEEE*, Xiao Cheng, Aying Jiwa, Dan Li, Jing Fang, and Zhencong Du

***Abstract*—Zanthoxylum bungeanum is widely cultivated, and some automatic picking systems or robots are researched to pick the pepper fruit instead of human hands. The detection algorithm is very important to an automatic picking system. Hue, saturation, and value (HSV) color space is widely used in color image segmentation. There is a fixed threshold based on HSV color space, or there is a dynamic threshold based on the Otsu method after the color images are converted into the gray images, because the Otsu method is very popular for the gray pictures. After the evaluation of the fixed threshold and the Otsu method for the pepper fruit detection, we propose an adaptive threshold method directly based on the HSV color space, called AHT and BBSV, which means adaptive hue threshold and balance between saturation and value. There is an adaptive threshold of separate hue, saturation, and luminance component. The hue threshold is obtained according to the conditions of whether there is soil, rock, and the pepper fruit in the image. The saturation and luminance thresholds are obtained by keeping the sum of saturation and luminance unchangeable. There are many hundreds of pictures for the test dataset. Our proposed method works very well, and the recall rate, accuracy, and false alarm can separately achieve 100%, 100%, and 0%, and to evaluate the location precision, we address two metrics: the ratio of the overlap area between the detection region and the ideal region and the location error. The ratio of the overlap area of our method is above 64%, and the location error is about 8%. All performance has been greatly improved compared with the fixed threshold.**



***Index Terms*— Adaptive hue threshold (AHT), balance between saturation and luminance, color image segmentation, hue, saturation, and value (HSV) space, *Zanthoxylum bugeanum* fruit detection.**

I. Introduction

ZANTHOXYLUM bungeanum is widely cultivated in China, especially in Sichuan Province, so it is also called Chinese pepper or Sichuan pepper. Regarding the planted area or the production of Chinese pepper, China ranks first in the world [1]. So far, the pepper fruits are picked by hand at most places, and the workers are prone to be hurt because of the thorns in the tree branches. Moreover, the pepper tree height is generally 2–5 m, so the workers are difficult to pick the pepper fruit on the so high branches. It is always short of workers although the wages grow rapidly, since there are few and few farmers in the countryside with the economic development in recent years. Therefore, some researchers proposed new picking tools to improve picking efficiency [2], [3], and other researchers proposed automatic picking systems or intelligent picking robots without manual control to reduce the worker's number and improve efficiency greatly [4], [5], [6].

The machine vision device is the core part of the automatic picking system of *zanthoxylum bungeanum* fruits. It takes a

Manuscript received 16 April 2023; accepted 12 May 2023. Date of publication 25 May 2023; date of current version 29 June 2023. This work was supported by the Key Research and Development Projects in Liangshan Prefecture of Sichuan Province of China under Grant 21ZDYF0209. The associate editor coordinating the review of this article and approving it for publication was Prof. Yu-Dong Zhang. *(Corresponding author: Longke He.)*

Longke He, Xiao Cheng, Aying Jiwa, Dan Li, and Jing Fang are with the School of Information Technology, Xichang University, Xichang 615000, China (e-mail: helongke@xcc.edu.cn; 2114903426@qq.com; jiwaayingxp@163.com; 125433047@qq.com; fangjing1018@sina.cn).

Zhencong Du was with the School of Information Technology, Xichang University, Xichang 615000, China. He is now with the Department of Science, Yibin University, Yibin 644000, China (e-mail: 38625686@qq.com).

Digital Object Identifier 10.1109/JSEN.2023.3277042

photograph and detects the pepper fruit from the photograph and gets the location of the pepper fruit; then, the pepper fruit is picked and collected automatically by the system. So, the detection method of the pepper fruit is very important. The detection of the pepper fruit is the method of assigning a label to the entire pixel in an image, such that pixels with the identical label contribute to the pepper fruit, which is meant as image segmentation.

Thresholding is one of the most used methods for image segmentation. The method of thresholding is to segment an image into two or more object regions. Many methods are proposed to find out the optimal threshold, which maximizes or minimizes an objective function. The goal of thresholding is to select a set of thresholds, which can discriminate between object and background pixels. Bi-level thresholding selects only one threshold, which separates the pixels into two classes, while multilevel thresholding determines multiple thresholds, which divide the pixels into several groups. The Otsu method is one of the traditional threshold selection methods. The optimal threshold is calculated according to the distribution of the gray-level histogram based on variance and intensity. It is very popular for the gray pictures [7], [8], [9].

Since there are three color components in color pictures, thresholding of color image segmentation is complex and always different according to the different implementations based on the choice of the color space. A color space is used to discriminate different colors. All the sets of colors are specified by a 3-D coordinate system and a subspace with the color component and specific models, which can reconstruct all the colors of the pictures. The frequently used color spaces are RGB, hue, saturation, and value (HSV), CIELAB, YCbCr, and so on.

Since the Otsu method is used for the gray images, color images can be converted into gray images for the application of the Otsu method. Color image conversion can be based on the Karhunen–Loeve transform, which is one representation of a stochastic process. The process is to map the multidimensional data with correlation into a new coordinate in the region of the data distribution in order to compress the data information, while the orientation of the new coordinate should keep the maximum amount of the information. This method is helpful to achieve dimensionality reduction of high-dimensional data. Then, the multithreshold Otsu method is used and three pictures show good segmentation result [10].

RBG color space can be used for color images, and the multilevel Otsu algorithm is used for separate R, G, and B channels [11]. Beyond RGB color space and the multilevel Otsu method, group search optimization (GSO) is used to get the dynamic threshold. The method is able to outperform the comparison approaches for the tests considering three, four, and five fixed thresholds for each RGB channel, and its performance is at least as good as the best comparison approach considering two thresholds for each RGB channel [12]. RBG color space is also used to detect the pepper fruit, and the value of the red channel minus the green channel of the pepper fruit is not the same as the background and leaves, and the fixed threshold is used to split the pepper fruit region and other regions. There are 200 pictures with the pepper fruit photographed in the natural scenes including sunny and cloudy days. The location error of the center of the fruit bunch is about 10 mm [13].

CIELAB color space is used to detect tumors. Color pictures are converted to CIELAB color space from RGB. The A and B layers consist of all the color information that exists in the image. K-means divides the objects into several clusters using the Euclidean distance metric. The detection of the tumor uses support vector machine (SVM) classifier. Three pictures are tested and the tumor is found [14]. LAB color space is also used to detect the pepper fruit. According to the obvious red tone of the mature pepper fruit, the image is converted from RGB to CIELAB color space, and the A component is used with the K-means clustering algorithm by $K = 3$. The algorithm can work with the pepper fruit, but it fails when there are many branches and trunks [15].

YCbCr color space is used to detect navel orange. The Otsu threshold segmentation algorithm is used to calculate the threshold value of the Cr component in the YCbCr color model, remove the background of navel orange, and obtain the binary images of the segmented navel orange area. A total of 98 images including positive, normal, and reverse light images are randomly selected and the recognition rate is about 87% [16].

Only the hue component in HSV color space and threshold ranged between 110° and 130° is used to classify good leaf surfaces and backgrounds; 60 experimental results show that the algorithm can identify the leaf surface accurately and represent a 100% accuracy rate [17]. Only the hue component of HSV color space is also used to detect the pepper fruit. The histogram of the hue component is calculated, and the threshold in the histogram is limited to a small range. The pepper fruit region is obtained from the threshold by the Otsu algorithm, which is called the improved Otsu algorithm. The threshold is fixed by three values according to the three conditions: direct sunlight, backlight, and shade. There are 180 pictures with the pepper fruit to test, and the correct recognition rate is about 90% [18]. The pepper fruit is detected by a fixed threshold of the hue component in HSV color space after the homomorphic filter is used to compensate for the light at the edge of the pepper fruit, and according to the round shape, the degree of circularity of the region of the pepper fruit is calculated. There are 235 pictures with the pepper fruit from the pepper planting base to test, and the average recall rate is up to 94% [19].

HSV color space and all of the HSV components are used to detect human skin. The hue (H) component, which is scaled from 0° to 360°, was divided into 36 primary intervals. The saturation (S) and value (V) components, which are scaled from 0% to 100%, were separated into ten primary intervals. Each was applied to the 951 skin samples extracted from the pictures of the individuals comprised of the human skin samples and nonskin samples, and we observed which intervals contained at least one tone which could be regarded as a human skin tone. The HSV filter selected a well-defined band in the geometric representation of the color space reducing the spectrum to 94 030 tones (2.5352% of the total spectrum), thus rejecting 97.4648% of colors as probable human skin

tones [20]. The histogram of each component in the HSV color space is calculated to detect the tactile paving and the threshold is dynamic according to the average and standard deviation of each histogram. There are 870 images taken in several indoor and outdoor environments in Europe and Asia to test, and the accuracy is about 91% [21]. The fixed range of the hue and value components is used to detect the green pixels; 150 green fluorescence images are tested and the acceptance rate was 90% [22]. In the image of tea bud, there is no obvious boundaries change in the histogram of separate R, G, and B channels in the RGB color space, and the image presents a relatively obvious boundary after HSI/HSV transformation, and it is easier to set the threshold value to separate buds [23]. The image is color-converted by HSV; then, the region of interest (ROI) is selected by the fixed threshold to detect and track the moving target [24].

HSV color space is also used to detect human faces [25], forest fires [26], koi fish [27], and so on.

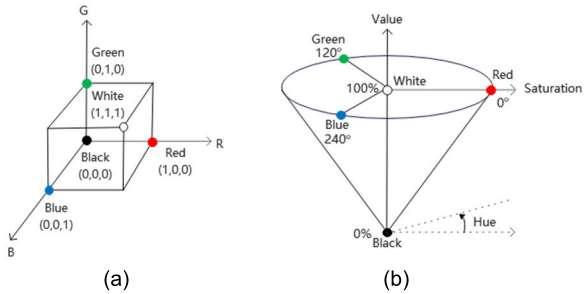
Some researchers used hybrid color spaces. The HSV and YCbCr color spaces are used to detect mango leaves. Each component including H, S, V, Cb, and Cr is compared with the threshold of the histogram by the Otsu method, and 62 images with mango leaves are tested; Cr component has a better performance, where the average recall is about 97% [28]. Both RBG and HSV color spaces are used to detect the pepper fruit. The R component is split with a fixed threshold of 0.6, and the H component is split with a fixed threshold of 0.9. Then, the result of R and H is multiplexed to get the final detection result [29].

Compared with other color spaces, HSV is closer to people's visual features, and HSV color space is dominant for color image segmentation, especially for the pepper fruit with the red color character, so we use HSV color space in our research, and the thresholds in HSV space are mostly fixed or dynamic by the Otsu method or k-means algorithm. The fixed thresholds meet the limited conditions, so we try to get the dynamic or adaptive thresholds. The object function of the Otsu method is equivalent to that of the k-means algorithm in multilevel thresholding [30]. We evaluate the Otsu algorithm for the detection of the pepper fruit before we propose our method to get the adaptive threshold. Our method is that the threshold of each component in the HSV color space is adaptive according to the conditions of every image.

The rest of this article is organized as follows. In Section II, RGB and HSV color model is introduced, and we give the general flowchart to detect the red pepper fruit. The typical scenarios of the detection are defined and the Otsu algorithm is evaluated in Section III. Section IV describes our proposed algorithm and flowchart. The simulation and experimental results are discussed in Section V. The conclusion is presented in Section VI.

II. Color Space and General Flowchart

We generally regard the basic colors as red, green, and blue and define other colors as a mix of these three. So, the RGB color space is basic, but what we perceive as color seems to depend on the characteristics of brightness, hue, and saturation, which is HSV color space. So, we introduce the relationship between HSV and GBR.



(a) (b)

Fig. 1. Color spaces. (a) RGB. (b) HSV.

A. RGB Color Space

In this color space, the primary colors or color components are red (R), green (G), and blue (B). It is an additive model, in which colors are produced by adding components, with white having all colors present and black being the absence of any color. This is the model used for active displays, such as television and computer screens. The RGB model is usually represented by a unit cube with one corner located at the origin of a 3-D color coordinate system, the axes being labeled R, G, and B, and having a range of values [0, 1], which is shown in Fig. 1(a). The origin (0, 0, 0) is considered black and the diagonally opposite corner (1, 1, 1) is called white.

Mixtures of light of these primary colors cover a large part of the human color space and thus produce a large part of human color experiences. This is why color television sets or color computer monitors need only produce mixtures of red, green, and blue light.

So, the ith pixel in the color image can be expressed by

$$f_i = \left[x, y, \left[r, g, b\right]\right] \tag{1}$$

where x and y are the indexes of the width and height of the image, and r, g, and b are the red, green, and blue value, which is in the range from 0 to 1.

B. HSV Color Space

The HSV color space is a description of the color change of different gray-level colors. Compared with the RGB color space, the HSV space is less sensitive to illumination changes and can better reflect the color distribution of the picture.

In the HSV color space, color light of the basic parameters can be measured with hue (H), saturation (S), and value (V). Value or luminance is defined as the brightness of the light, which is caused by human eyes. In general, the larger energy the color light contains, the brighter it presents. On the contrary, it is dark. Luminance range is from 0% (black) to 100% (white). Hue reflects the color categories. Saturation refers to the degree of color depth. For the same hue color, the higher the degree of saturation is, the deeper the color is, and the lower the saturation becomes, the lighter its color is.

The HSV color space is represented by a cone, as shown in Fig. 1(b). The three points on the top surface represent the three color positions of red, green, and blue, respectively. The

S direction indicates the saturation change, and the closer to the outer frame, the higher the saturation; the height of the cone indicates the luminance value, and the bottom of V is black and the top is white.

So, the ith pixel in the color image can be expressed by

$$f_i = \left[x, y, [H, S, V]\right] \tag{2}$$

where x and y are the indexes of the width and height of the image, and H, S, and V are hue, saturation, and luminance, which is transformed from RBG by

$$H = \begin{cases} 0, & \text{if } L == M \\ 60^\circ \times \dfrac{g-b}{L-S} + 0^\circ, & \text{if } L == r \text{ and } g \geq b \\ 60^\circ \times \dfrac{g-b}{L-S} + 360^\circ, & \text{if } L == r \text{ and } g < b \\ 60^\circ \times \dfrac{b-r}{L-S} + 120^\circ, & \text{if } L == g \\ 60^\circ \times \dfrac{r-g}{L-S} + 240^\circ, & \text{if } L == b \end{cases} \tag{3}$$

$$S = \frac{L-M}{L} \tag{4}$$

$$V = L \tag{5}$$

$$L = \max(r, g, b) \tag{6}$$

$$M = \min(r, g, b) \tag{7}$$

where max is the function to obtain the maximal value, and min is the function to obtain the minimal value.

From (3) to (7), H is in the range from 0° to 360°, S is in the range from 0 to 1, and V is in the range from 0 to 1.

In our research, Opencv is used because it is very popular to process the digital image with open source. In Opencv, H, S, and V are stored by the integers of 8 bits, which means the stored value of H is divided by 2 and round to the integer, and the stored value of S is amplified by 255 and round to the integer, and the stored value of V is amplified by 255 and round to the integer. So, the range of the stored value of H is from 0 to 180, the range of the stored value of S is from 0 to 255, and the range of the stored value of V is from 0 to 255. In the next, we express the value of H, S, and V according to these ranges.

C. General Flowchart

Since there is an obvious character which is the pepper fruit is red to distinguish the leaves, branches, and other backgrounds, the pepper fruit can be detected from the images based on the color space. In general, the red pepper fruit in the color picture is detected by the following steps:

1) read the picture with the original RBG color space;
2) convert to HSV color space;
3) get the mask within some small range of H, S, and V separated by a threshold;
4) morphological operations, such as dilating and eroding;
5) get the contours; and
6) get the rectangle of the max contour.

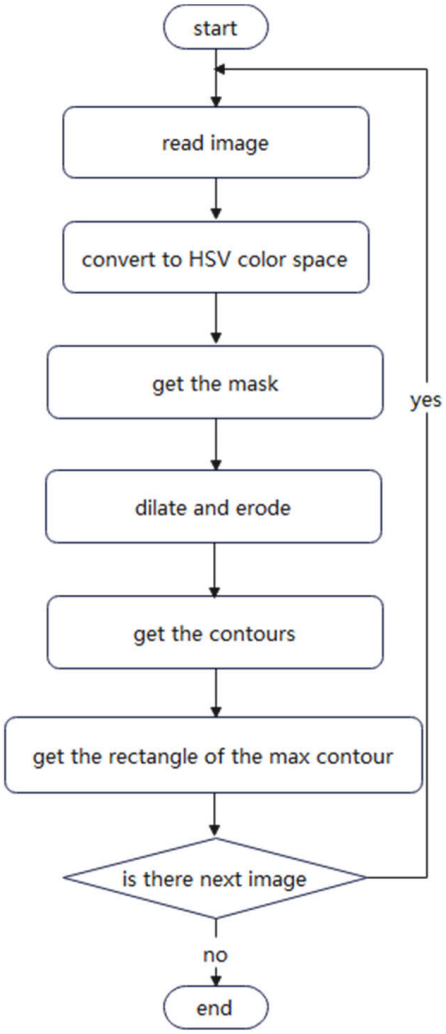


Fig. 2. General flowchart.

The detection algorithm is to improve the mask and better get the fruit region from the contour. Since the picture should be detected at one time before the picking system operates every time, so we get the max contour of each detection to calculate the location and make the picking system move to the fruit.

These steps can be shown in the general flowchart of Fig. 2.

III. Scenarios and Present Method

For investigating the detection problems of Sichuan Pepper fruit, we find out several typical scenarios and solve their problems from the features.

A. Typical Scenarios

Since the automatic picking system is located in the field, it can work well whenever the weather is cloudy or sunny, whether the background is soil, rock, road, or house. If the weather is cloudy, the pepper fruit is dim in the picture from the camera of the automatic picking system. The background of soil, rock, and others maybe interfere with the correct detection of the pepper fruit.

The typical scenarios are shown in the eight pictures in Fig. 3.

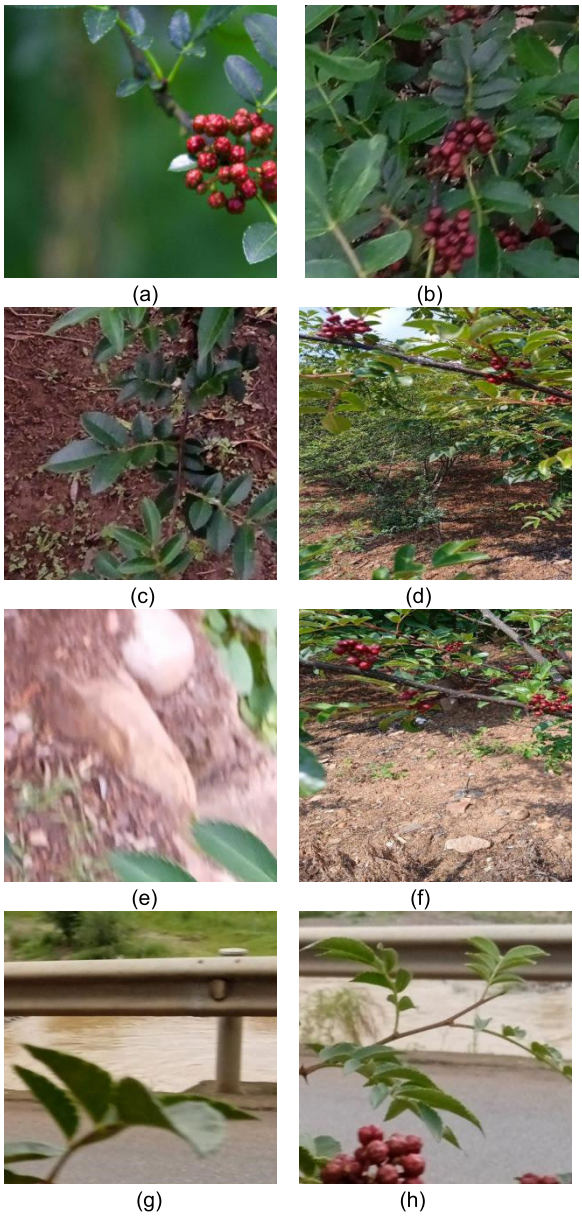


Fig. 3. Typical scenarios. (a) Sunny. (b) Cloudy. (c) Soil. (d) Pepper with soil. (e) Rock. (f) Pepper with rock. (g) Road. (h) Pepper with road.

B. Otsu Algorithm

The Otsu algorithm is used to maximize the interclass variance. The principle is as follows.

Suppose there are L gray levels $[1, 2, \ldots, L]$ in a gray image. The number of pixels at level i is denoted by n_i and the total number of pixels by $N = n_1 + n_2 + \cdots + n_L$. The gray-level histogram is normalized and its probability distribution is

$$p_i = p_i/N, \quad p_i \geq 0, \quad \sum_{i=1}^{L} p_i = 1. \tag{8}$$

The pixels are divided into two classes C_0, and C_1 by a threshold at level k; C_0 denotes pixels with levels $[1, \ldots, k]$, and C_1 denotes pixels with levels $[k+1, \ldots, L]$. Then, the probabilities and averages of C_0, and C_1, respectively, are given by

$$\omega_0 = \Pr(\omega_0) = \sum_{i=1}^{k} p_i = \omega(k) \tag{9}$$

$$\omega_1 = \Pr(\omega_1) = \sum_{i=k+1}^{L} p_i = 1 - \omega(k) \tag{10}$$

and

$$\mu_0 = \sum_{i=1}^{k} i\Pr(i|C_0) = \sum_{i=1}^{k} ip_i/\omega_0 = \mu(k)/\omega(k) \tag{11}$$

$$\mu_1 = \sum_{i=k+1}^{L} i\Pr(i|C_0) = \sum_{i=K+1}^{L} ip_i/\omega_1 = \frac{\mu_T - \mu(k)}{1 - \omega(k)} \tag{12}$$

where

$$\mu(k) = \sum_{i=1}^{k} ip_i \tag{13}$$

$$\mu_T = \mu(L) = \sum_{i=1}^{L} ip_i. \tag{14}$$

So, we can get the relation

$$\omega_0\mu_0 + \omega_1\mu_1 = \mu_T, \quad \omega_0 + \omega_1 = 1. \tag{15}$$

The class variances are

$$\sigma_0^2 = \sum_{i=1}^{k} (i - \mu_0)^2 \Pr(i|C_0) = \sum_{i=1}^{k} (i - \mu_0)^2 p_i/\omega_0 \tag{16}$$

$$\sigma_1^2 = \sum_{i=k+1}^{L} (i - \mu_1)^2 \Pr(i|C_1) = \sum_{i=k+1}^{k} (i - \mu_1)^2 p_i/\omega_1. \tag{17}$$

The optimal threshold k that maximizes interclass variance η or equivalently maximizes σ_B^2

$$\eta = \sigma_B^2/\sigma_T^2 \tag{18}$$

where

$$\sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 = \omega_0\omega_1(\mu_1 - \mu_0)^2 \tag{19}$$

$$\sigma_T^2 = \sum_{i=1}^{L} (i - \mu_T)^2 p_i. \tag{20}$$

The Otsu algorithm is used for the gray picture, and we can convert the color picture into the gray one to use the Otsu algorithm instead of converting to HSV color space in the general chart of Fig. 2. The pixel number of every gray level from 0 to 255 is calculated, and the gray histogram is obtained, and the threshold is found by the Otsu algorithm. Then, we can get the thresholding image as the mask, and the thresholding image means the pixel value is set to 255 if the original pixel value is bigger than the threshold and set to 0 otherwise.

From the gray histogram of Fig. 4(b), we can get the threshold of 84; then, we can get the thresholding image of

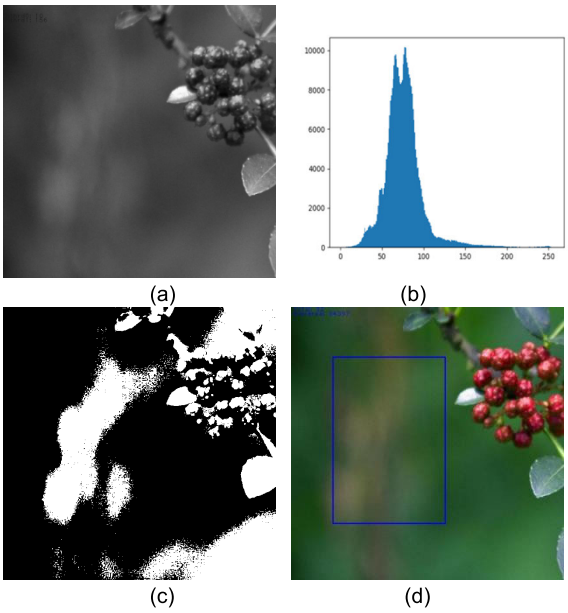


Fig. 4. Detection of Fig. 3(a) with the Otsu algorithm. (a) Gray image of Fig. 3(a). (b) Gray histogram. (c) Thresholding. (d) Detection result.

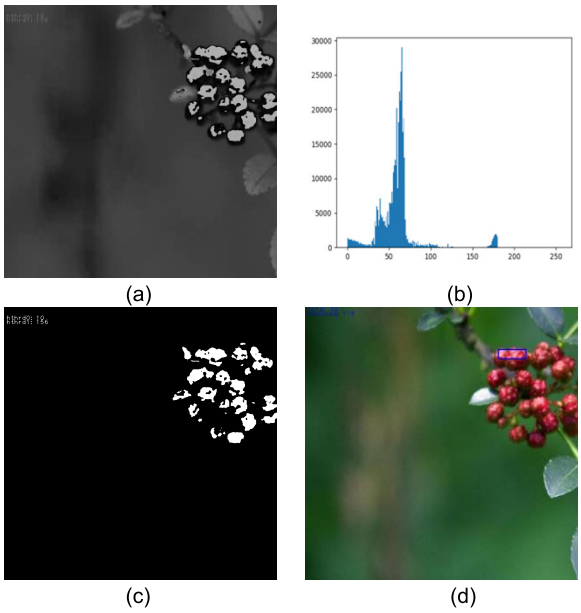


Fig. 5. Detection of Fig. 3(a) with the Otsu algorithm. (a) Hue image of Fig. 3(a). (b) Hue histogram. (c) Thresholding hue image. (d) Detection result.

Fig. 4(c) and get the detection result of Fig. 4(d) with the blue rectangle.

In general, we give the detection of the maxim area because our detection is for the location of the pepper fruit to control the picking system, which means once detection once to pick.

On the other hand, we can use the image of the hue component instead of the gray one.

From the hue histogram of Fig. 5(b), we can get the threshold of 113; then, we can get the thresholding image of Fig. 5(c) and get the detection result of Fig. 5(d).

The Otsu method can work by using the image of the hue component instead of the gray one if the fruit exists, but it cannot work if there is no fruit, as an example in Fig. 6; from the hue histogram of Fig. 6(b), we can get the threshold of 84; then, we can get the thresholding image of Fig. 6(c) and get the detection result of Fig. 6(d), where the blue rectangle is the false detection of the pepper fruit.

C. Fixed Threshold

Since the Otsu algorithm is used for the gray picture, the color ROI is segmented through the color space, such as GBR, HSV, and so on.

When HSV space is used for segmentation, the red ROI is extracted by

$$H \in [h_{0L}, h_{0H}] \cup [h_{1L}, h_{1H}] \tag{21}$$

$$S \in [s_L, s_H] \tag{22}$$

$$V \in [v_L, v_H] \tag{23}$$

where $\in$ means to belong to some set, $\cup$ means the union, and [] means a set.

According to the color space of HSV, h_{0L} is fixed to 0 and h_{1H} is fixed to 180. Empirically, h_{0H} can be 10 and h_{1L} can be 156, s_L can be 43 and s_H can be 255, and v_L can be 46 and v_H can be 255.

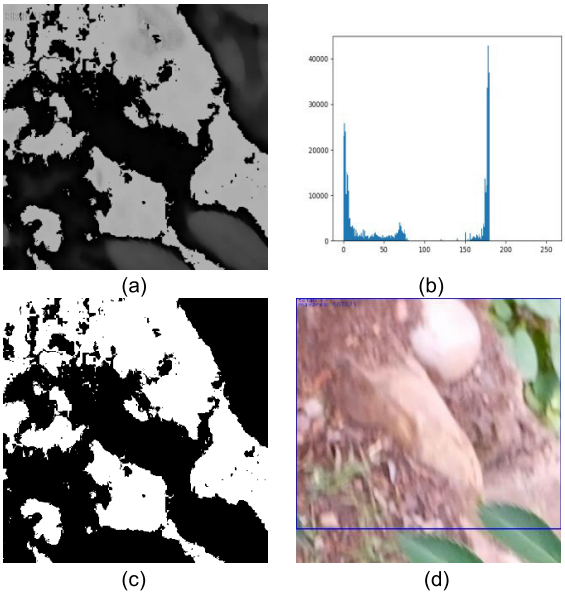


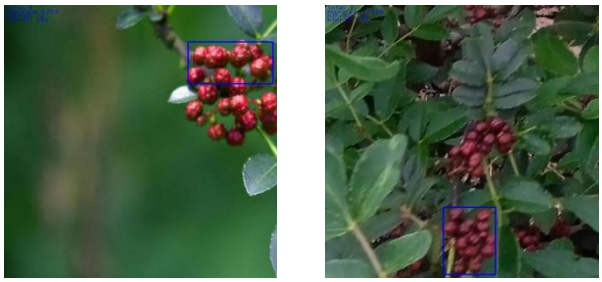
Fig. 6. Detection of Fig. 3(e) with the Otsu algorithm. (a) Hue image of Fig. 3(e). (b) Hue histogram. (c) Thresholding hue image. (d) Detection result.

When the empirical fixed values of H, S, and V are used, the detection results of Fig. 3(a) and (b) are good, as shown in Fig. 7.

In Fig. 7, the blue rectangle means the ROI, which means the pepper fruit is detected successfully.

But when the same empirical fixed values of H, S, and V are used, the soil or rock is regarded falsely as the pepper fruit, as Fig. 8, where almost the whole of the picture is falsely regarded as the ROI.

From Figs. 7 and 8, although the empirical fixed values can work well if the real pepper fruit exits, they cannot work if there is no pepper fruit and just soil or rock.



(a) (b)

Fig. 7. Successful detection with empirical fixed values. (a) Sunny detection. (b) Cloudy detection.

(a) (b)

Fig. 8. False detection with empirical fixed values. (a) Soil detection. (b) Rock detection.

Next, we propose our method to get adaptive thresholds of H, S, and V components.

IV. Proposed Method

The empirical fixed values cannot meet the condition of soil or rock, what is the reason? We found out some interesting features.

A. Interesting Features

First of all, we found that the saturation value of rock is not beyond 50. For example, we choose two regions to analyze, as shown in Fig. 9. The red, blue, and green points of Fig. 9(b) and (d) are separate hue, saturation, and luminance values.

From Fig. 9, we can watch that the values of H, S, and V in Regions 1 and 2 are very stable. In Region 1, the average value of hue, saturation, and luminance is separately 160, 10, and 255, and in Region 2, the average value of hue, saturation, and luminance is separately 6, 44, and 243.

The average values of hue, saturation, and luminance are separately obtained by these steps: 1) choose the hue value of all pixels of the region;

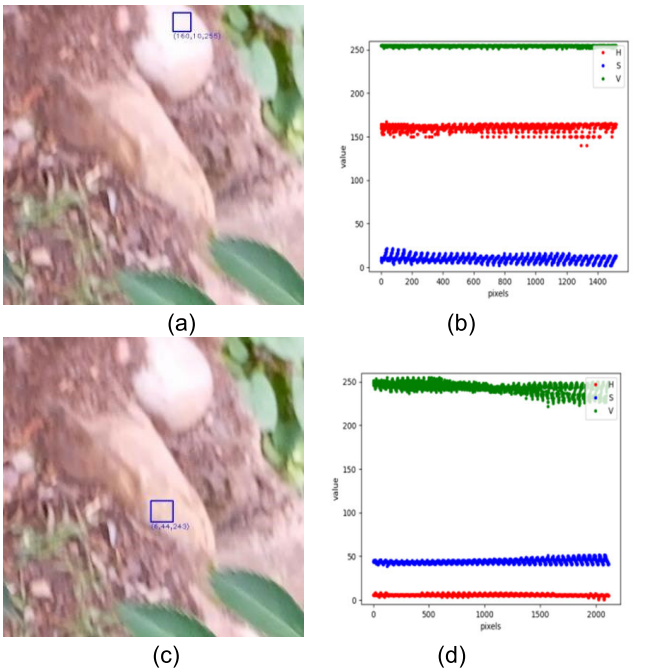
2) convert the original picture with the format of jpg or png into HSV space, according to (3)–(7); and

3) calculate the average values.

Second, we found that the saturation value of soil increases and the luminance value decreases compared with rock. For example, we choose two regions to analyze, as shown in Fig. 10.

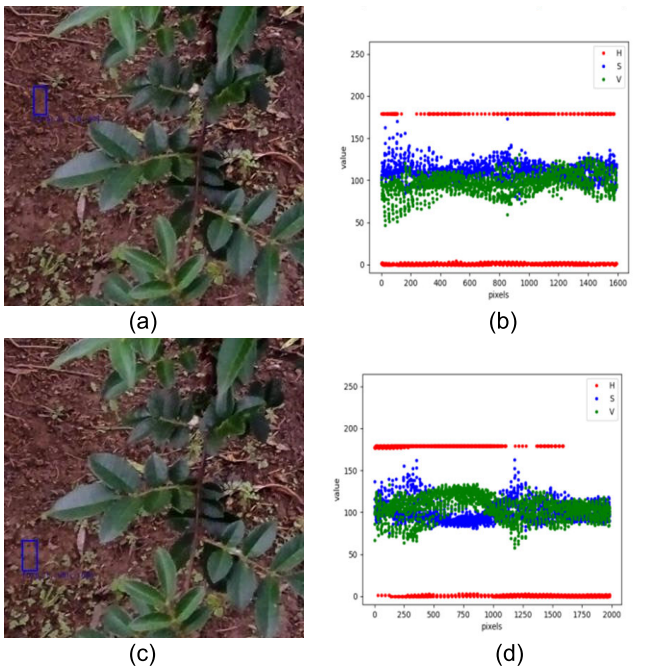
From Fig. 10, we can watch that the values of H, S, and V in Regions 1 and 2 are very stable. In Region 1, the average value of H is 179 and 1 for the two ranges of [156, 180] and [0, 10], and the average value of saturation and luminance separately is 110 and 97. In Region 2, the average value of H is also 179 and 1, and the average value of saturation and luminance is separately 101 and 105.

Third, we found that the saturation and luminance values of the pepper fruit change alternately, which means when



(a) (b) (c) (d)

Fig. 9. Rock region and its HSV distribution. (a) Rock region 1. (b) HSV distribution of rock region 1. (c) Rock region 2. (d) HSV distribution of rock region 2.



(a) (b) (c) (d)

Fig. 10. Soil region and its HSV distribution. (a) Soil region 1. (b) HSV distribution of soil region 1. (c) Soil region 2. (d) HSV distribution of soil region 2.

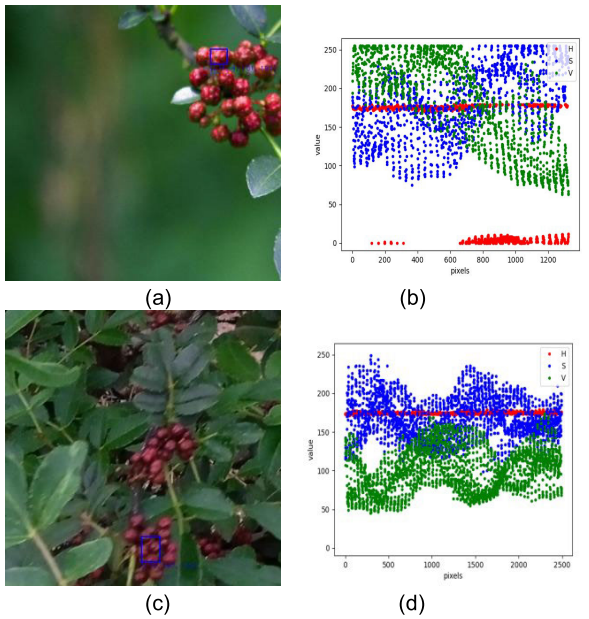


Fig. 11. Pepper fruit region and its HSV distribution. (a) Fruit region 1. (b) HSV distribution of fruit region 1. (c) Fruit region 2. (d) HSV distribution of fruit region 2.

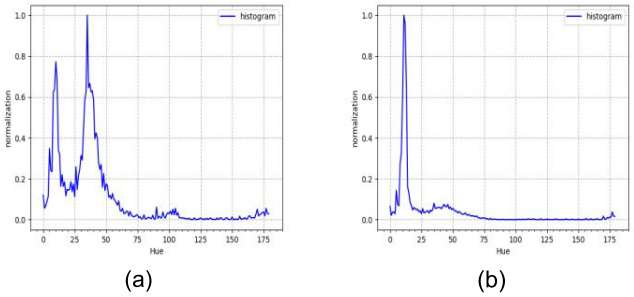


Fig. 12. Hue histogram. (a) Histogram of Fig. 3(d). (b) Histogram of Fig. 3(f).

saturation increases a little, luminance decreases a little at the same time and vice versa. For example, we get the statistical result, as shown in Fig. 11.

From Fig. 11, we can watch that the value of H is very stable, and S and V change alternately both in Regions 1 and 2. In Region 1, the average value of H is 176 and 4, and the average value of S and V is separately 181 and 180. In Region 2, the average value of H, S, and V is separately 175, 167, and 100.

Now, we give the analysis of S and V changing alternately. The RBG distribution of fruit region 1 is shown in Fig. 12(a). The value of the r component is always bigger than the values of the g or b components, and the value of the g component is very close to the value of the b component. The values are evenly divided into five parts according to the horizontal axis pixel. The values of each RGB component of each part are averaged to calculate the values of the saturation and value components by (4)–(7). The difference in $S + V$ values between the five parts is small, especially between part 1 and part 2, as shown in Table I. The calculation results in Table I are in good agreement with Fig. 12(b).

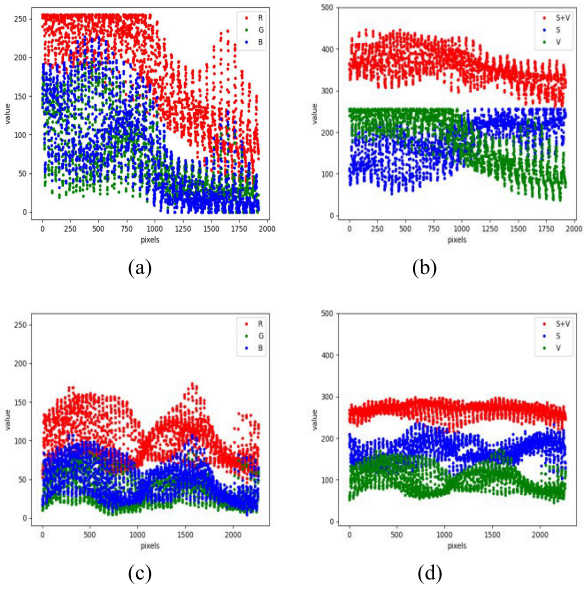


Fig. 13. RGB distribution and $S + V$ of each region. (a) RGB distribution of region 1. (b) $S + V$ of region 1. (c) RGB distribution of region 2. (d) $S + V$ of region 2.

TABLE I
S AND V VALUES CALCULATED IN FIVE PARTS OF REGION 1

	r	g	b	S	V	S+V
Part 1	233	107	126	255-107/233*255	233	371
Part 2	232	97	118	255-97/232*255	232	381
Part 3	194	71	77	255-71/194*255	194	356
Part 4	129	34	23	255-23/129*255	129	339
Part 5	101	27	22	255-22/101*255	101	302

TABLE II
S AND V VALUES CALCULATED IN FIVE PARTS OF REGION 2

	r	g	b	S	V	S+V
Part 1	116	49	62	255-49/116*255	116	263
Part 2	102	36	48	255-36/102*255	102	267
Part 3	96	32	42	255-32/96*255	96	266
Part 4	110	41	53	255-41/110*255	110	270
Part 5	78	23	28	255-23/78*255	78	258

Similarly, the RBG distribution of fruit region 2 is shown in Fig. 12(c). The values are also evenly divided into five parts according to the horizontal axis pixel. The values of each RGB component of each part are averaged to calculate the values of the saturation and value components by (4)–(7). There is almost no difference in $S + V$ values between the five parts. The calculation results in Table II are in good agreement with Fig. 12(d).

Fourthly, if we calculate the hue histogram of the pepper fruit image with soil, rock, or road, there is a peak around 11, which is the start value of the orange color, namely the neighbor color of the red one. For example, we get the histogram result of soil and rock, as shown in Fig. 13.

The histogram of hue is obtained by the following steps:

1) convert the original picture with the format of jpg or png into HSV space, according to (3)–(7);

2) choose the hue value of all pixels;
3) get the sum number of pixels for every hue value;
4) search the maxim of the sum numbers;
5) normalize the sum numbers with maxim;
6) plot the histogram with the x-axis as the hue value from 0 to 180 and the y-axis as the normalization of the pixel numbers.

B. Balance Between Saturation and Value

We can get the balance between saturation and luminance of the pepper fruit, but there is no balance in the images of rock and soil, so we can use that to extract the pepper fruit.

Our proposed algorithm of balance between saturation and luminance (BBSV) can be expressed by:

1) initialize ROI to be empty;
2) set $s_i = s_{\min}$;
3) according to $B = s_i + v_i$ and $v_i \geq v_{\min}$, get v_i;
4) use (21)–(23) with $s_L = s_i$, $s_H = 255$, $v_L = v_i$, and $v_H = 255$ to extract the objective region and get ROI_i;
5) $\mathrm{ROI} = \mathrm{ROI} \cup \mathrm{ROI}_i$;
6) $s_{i+1} = s_i + s_{\text{step}}$; and
7) repeat 3)–6) until $s_{i+1} > 255$ or $v_{i+1} < v_{\min}$

where B, $s_{\min}$, $v_{\min}$, and s_{step} are the parameters and will be analyzed in Section V; h_{0L} is fixed to 0 and h_{1H} is fixed to 180, and h_{0H} and h_{1L} can be adaptive value described in the next.

C. Adaptive Hue Threshold

Since there is a peak around 11 in the hue histogram, which is due to the dry grass or soil, and at the same time, there is a peak of 0, which is generated by the red color of the pepper fruit, so there is a minimum between the two peaks, and we should get the minimum in the range from 1 to 10 as the value of h_{0H}.

The another red range is similar. If there is a peak around 155 due to the dry grass or soil, and at the same time, there is a peak around 180 due to the pepper fruit, so there is a minimum in the range from 156 to 179 between the two peaks, and we should get the minimum as the value of h_{1L}.

But sometimes, there is no peak around 11 or 155 in the hue histogram if there is no dry grass or soil in the image, and the value of h_{0H} and h_{1L} should be separately 10 and 156.

So, we should estimate whether there is a peak around 11 and 155 in the hue histogram. If there is a peak around 11, there is a minimum in the range from 0 to 10, which is called the left minimum, and there is a minimum in the range from 11 to 25 (the end value of the orange color or the start value of the yellow color minus 1), which is called the right minimum. The slope of fitting the line from the left minimum to the peak (left line) is plus, and the slope of fitting the line from the peak to the right minimum (right line) is minus. We can use the slopes to estimate the presence of the peak. If the slope of the left line is beyond a threshold and the slope of the right line is below a threshold, the peak shows up.

In some cases, there are multiplex minimums in the range from 1 to 10 or from 156 to 179 in the hue histogram, or there are multiple values that are almost equal to the minimum. Then, we should get the biggest hue value as h_{0H} and the smallest hue value as h_{1L}.

In some cases, although the trend from 0 to minimum between 1 and 10 or from 180 to minimum between 156 and 179 is declining, it maybe fluctuant, and the minimum is not as good as the value of h_{0H} or h_{1L}. If the value in the range from 0 to minimum between 1 and 10 or from 180 to minimum is smaller than the front and back neighbor values or the difference between the contiguous values alternates with positive and negative, the fluctuation is present, and we should get the first value whose next value toward the minimum is bigger and beyond a threshold, which is the fluctuation elimination.

Overall, the steps of our proposed algorithm of an adaptive hue threshold (AHT) are as follows:

1) calculate the histogram of hue and get the pixels number of every hue value: $R(h_{\min}, \ldots, h_{\max})$;
2) assume h_a, h_b, and $h_c (h_c > h_b > h_a)$ as the first, second and third color start point, for example, $h_a = 0$ is the red color start point, $h_b = 11$ is the orange color start point, and $h_c = 26$ is the yellow color start point;
3) initialize $h_{0H} = h_b - 1$;
4) search the left minimum $R_{h_l,\min} = \min_{h_a < h_i < \mathrm{int}((h_a+h_b)/2)} R(h_i)$ and get the index of h_l;
5) if there is multiplex minimums, then choose the bigger index: from $h_l + 1$ to $\mathrm{int}((h_a + h_b)/2)$ to search h_i, if $(R_{h_i} - R_{h_l,\min}) / \max_{h_{\min} < h_i < h_{\max}} R(h_i) < T_0$, then $h_l = h_i$, $R_{h_l,\min} = R_{h_i}$;
6) search the peak $R_{h_p,\max} = \max_{h_b < h_i < ((h_a+h_b)/2)\mathrm{int}} R(h_i)$ and get the index of h_p;
7) search the right minimum $R_{h_r,\min} = \min_{\mathrm{int}((h_b+h_c)/2) < h_i < h_c} R(h_i)$ and get the index of h_r;
8) fit a straight line with the x-axis of $h_i / h_p (h_p \geq h_i \geq h_l)$ and the y-axis of $R_{h_i} / R_{h_p,\max}$, then get the slope of the line $g_{l,l}$;
9) fit another straight line with the x-axis of $h_i / h_r (h_r \geq h_i \geq h_p)$ and the y-axis of $R_{h_i} / R_{h_p,\max}$, then get the slope of the line $g_{r,l}$;
10) if $g_{l,l} \geq T_1$ and $g_{r,l} \leq T_2$, then $h_{0H} = h_l$;
11) get the normalized histogram of hue $R'(h_{\min}, \ldots, h_{\max}) = R(h_{\min}, \ldots, h_{\max}) / \max_{h_{\min} \leq h_i \leq h_{\max}} R(h_i)$;
12) calculate the difference between the neighbors as: $D(h_i) = R'(h_i + 1) - R'(h_i)$, $h_{\max} > h_i \geq h_{\min}$;
13) initialize set $U_{\max}$ and $U_{\min}$ to be empty;
14) if $D(h_i) \geq 0$ and $D(h_i - 1) \leq 0$, then $U_{\min} = U_{\min} \cup [h_i]$;
15) if $D(h_i) \leq 0$ and $D(h_i - 1) \geq 0$, then $U_{\max} = U_{\max} \cup [h_i]$;
16) repeat 14)–15) with $h_b - 1 > h_i > h_a$ from $h_a + 1$ to $h_b - 2$;
17) from 0 to the maxim index of $U_{\max}$ and $U_{\min}$, search the first matched result: if $U_{\min}(i) < U_{\max}(j) < U_{\min}(i + 1)$ and $R'(U_{\max}(j)) - R'(U_{\min}(i)) > T_3$, then $h_{0H} = \min(h_{0H}, U_{\min}(i))$. Note: the steps of 11)–17) are the fluctuation elimination.

18) assume h_d, h_e, and $h_f (h_d > h_e > h_f)$ as the first, second, and third color endpoint, for example, $h_d = 180$ is the red color endpoint, $h_e = 155$ is the purple color endpoint, and $h_f = 124$ is the blue color endpoint; 19) initialize the $h_{1L} = h_e + 1$;

20) search the right minimum $R_{h_r,\min} = \min_{h_d<h_i<\text{int}((h_d+h_e)/2)} R(h_i)$ and get the index of h_r;

21) if there are multiplex minimums, then choose the smaller index: from $h_r - 1$ to $\text{int}((h_d + h_e)/2)$ search h_i, if $(R_{h_i} - R_{h_r,\min})/\max_{h_{\min}<h_i<h_{\max}} R(h_i) < T_4$, then $h_r = h_i$, $R_{h_r,\min} = R_{h_i}$;

22) search the peak $R_{h_p,\max} = \max_{h_e<h_i<\text{int}((h_e+h_f)/2)} R(h_i)$ and get the index of h_p;

23) search the left minimum $R_{h_l,\min} = \min_{\text{int}((h_e+h_f)/2)h_i>h_f} R(h_i)$ and get the index of h_l;

24) fit a straight line with the x-axis of $h_i/h_r (h_r \geq h_i \geq h_p)$ and the y-axis of $R_{h_i}/R_{h_p,\max}$, then get the slope of the line $g_{r,r}$;

25) fit another straight line with the x-axis of $h_i/h_p (h_p \geq h_i \geq h_l)$ and the y-axis of $R_{h_i}/\max_{h_{\min}<h_i<h_{\max}} R_{h_p,\max}$, then get the slope of the line $g_{l,r}$;

26) if $g_{l,r} \geq T_5$ and $g_{r,r} \leq T_6$, then $h_{1L} = h_r$;

27) get the normalized histogram of hue $R'(h_{\min}, \ldots, h_{\max}) = R(h_{\min}, \ldots, h_{\max})/R(h_i)$;

28) calculate the difference between the neighbor as: $D(h_i) = R'(h_i + 1) - R'(h_i)$, $h_{\max} > h_i \geq h_{\min}$;

29) initialize set $U_{\max}$ and $U_{\min}$ to be empty;

30) if $D(h_i) \geq 0$ and $D(h_i - 1) \leq 0$, then $U_{\min} = U_{\min} \cup [h_i]$;

31) if $D(h_i) \leq 0$ and $D(h_i - 1) \geq 0$, then $U_{\max} = U_{\max} \cup [h_i]$;

32) repeat 13)–14) with $h_d - 1 > h_i > h_e$ from $h_d - 2$ to $h_e + 1$; and

33) from 0 to the maxim index of $U_{\max}$ and $U_{\min}$ search the first matched result: if $U_{\min}(i) < U_{\max}(j) < U_{\min}(i + 1)$ and $R'(U_{\max}(j)) - R'(U_{\min}(i)) > T_7$, then $h_{1L} = \min(h_{1L}, U_{\min}(i))$.

Note: the steps of 27)–33) are the fluctuation elimination. In the above steps, $T_0 T_7$ are the parameters, and int is the function to get the integer, and in our research, $h_{\min} = 0$ and $h_{\max} = 180$.

D. Flowchart

After the processing of AHT and BBSV, there is edge detection to remove the indistinct region, which is induced by the pepper fruit far away. The gray level is very low after the indistinct region is processed by the edge detection, and it can be removed when it is below a threshold.

The Sobel algorithm is widely used because its process is fast and the edge detection result is effective. The Sobel edge processor uses a convolution kernel to create a series of gradient magnitudes. Applying convolution K to pixel group p, the output N is

$$N(x, y) = \sum_{j=-1}^{1} \sum_{i=-1}^{1} K(i, j)\, p(x - i, y - j). \tag{24}$$

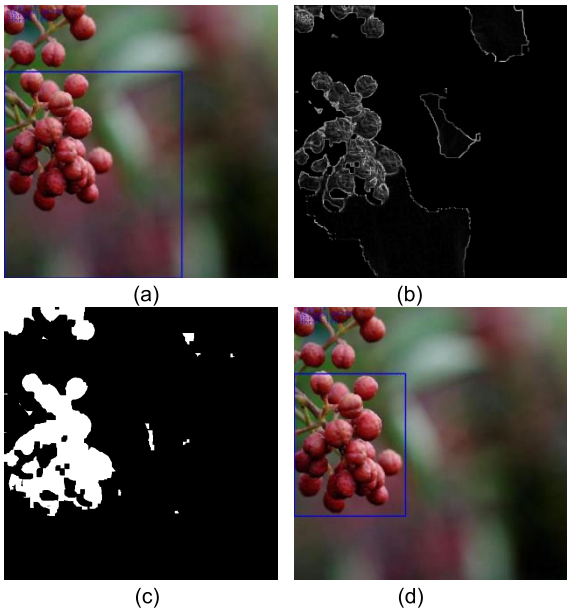


Fig. 14. Detection with edge. (a) Detection without edge. (b) Sobel detection. (c) Dilate. (d) Detection with edge.

In general, the convolution K is divided into two convolution kernels, one to detect changes in horizontal contrast (K_x) and another to detect vertical contrast (K_y)

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{25}$$

$$K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}. \tag{26}$$

After the horizontal and vertical convolution, the horizontal output is $N_x(x, y)$ and the vertical output is $N_y(x, y)$, so the Sobel processor output is

$$N(x, y) = \sqrt{N_x^2(x, y) + N_y^2(x, y)}. \tag{27}$$

From Fig. 14, we can watch out that the indistinct region is removed very well after Sobel detection and dilating and removing the lower gray region.

Our proposed algorithm is used before getting the mask and after converting to HSV color space in the general flowchart in Fig. 2, so the general flowchart is modified and the flowchart of the whole detection of our proposed method is shown in Fig. 15.

V. Experience Results and Analysis

We evaluate the parameters of our proposed algorithm first and then introduce the dataset and performance metrics to get the final experience result.

A. Parameters of AHT and BBSV

For AHT, $T_0 - T_7$ are the parameters. T_0 is 0.08, T_1 is 0.45, T_2 is -0.45, T_3 is 0.15, T_4 is 0.05, T_5 is 0.45, T_6 is -0.45, and T_7 is 0.15.

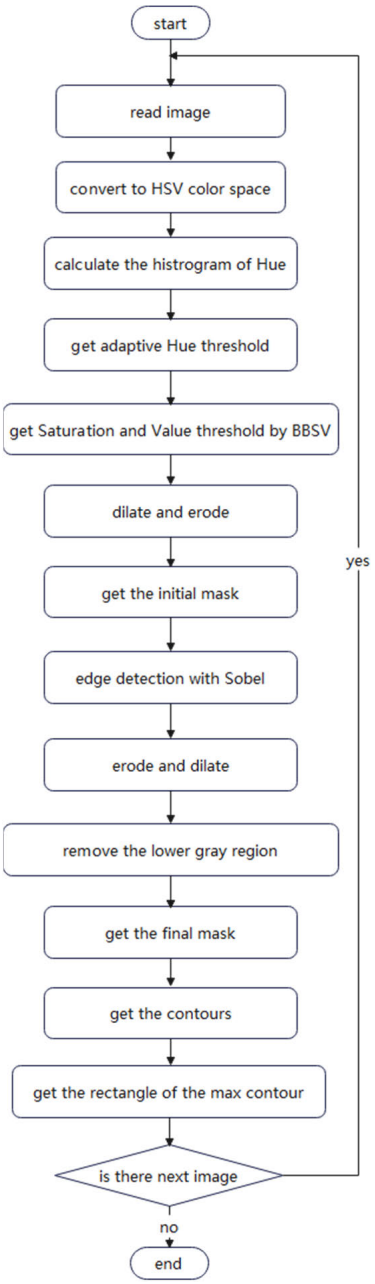


Fig. 15. Flowchart of our proposed method.

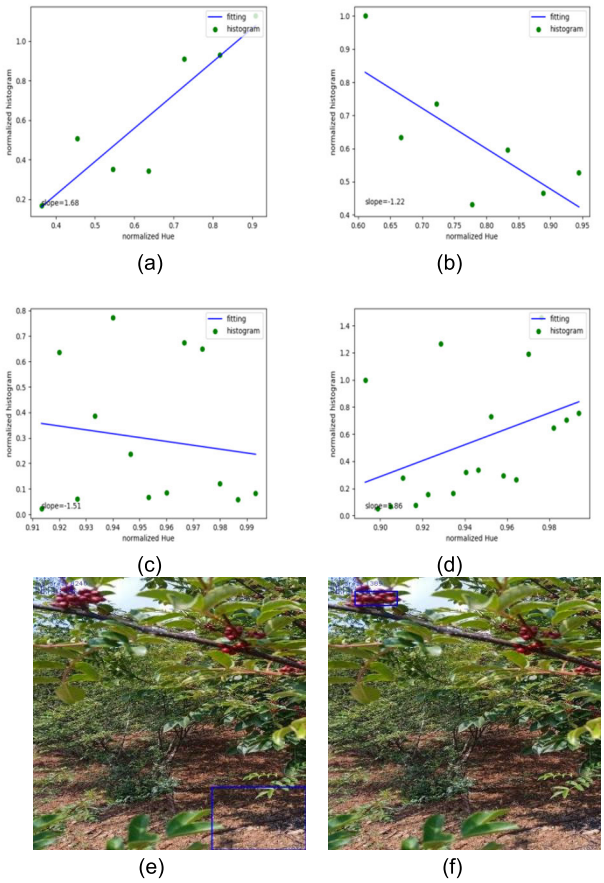


Fig. 16. AHT detection with its parameters. (a) Left of peak around 10. (b) Right of peak around 10. (c) Left of peak around 156. (d) Right of peak around 156. (e) Fixed threshold. (f) Adaptive Hue threshold.

Fig. 17. Correct detection with BBSV. (a) Soil detection. (b) Rock detection.

Fig. 16(a)–(d) shows the fitting line result of Fig. 13(a), and the blue rectangle of Fig. 16(e) and (f) is the detection of the pepper fruit, and the output result is only the maxim of the detection contour. The hue threshold is 4 and 156 according to Fig. 16(a)–(d) and the threshold is used in Fig. 16(f), and the fixed threshold of 10 and 156 is used in Fig. 16(e).

From Fig. 16, we can watch that the parameters with the typical value are effective and get the correct detection result; in contrast, the fixed hue threshold works invalidly.

For BBSV, the typical parameter values are $B = 230$, $s_{\min} = 110$, $v_{\min} = 40$, and $s_{\text{step}} = 10$, and Fig. 17 is the detection result by BBSV with the typical parameters. There is no blue rectangle of detection, and it is obvious that BBSV works very well and there is no false detection anymore.

Now, we turn on and off the fluctuation elimination, as shown in Fig. 18, where FE means fluctuation elimination. We can watch that the false detection can be eliminated further.

Finally, we turn on the AHT and BTSV at the same time to see how our proposed algorithm works in the typical scenarios of Fig. 3. Fig. 19 is the detection results, and we can watch that our proposed algorithm works very well.

B. Dataset

To evaluate the performance of our algorithm, we build the dataset to test.

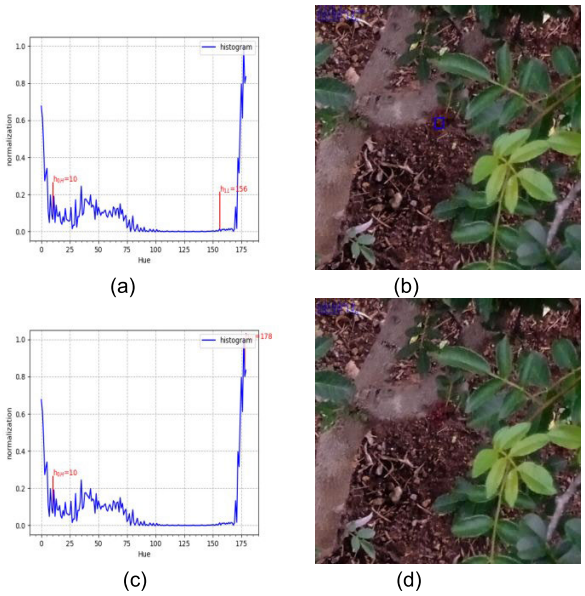


Fig. 18. Detection with edge. (a) Threshold without FE. (b) False detection without FE. (c) Threshold with FE. (d) Correct detection with FE.

The original pictures with the pepper fruit are more than 300, which is at most 1706 × 1280 (expressed by *origin* in the following equation), and some of them are got from the internet, and others are photographed in the *zanthoxylum bungeanum* garden. We use the following equations to generate many pictures of 640 × 640 (expressed by *img*) from every original picture:

$$\text{img}\,(x, y) = \text{origin}\,(\tilde{x}, \tilde{y}) \tag{28}$$

$$\tilde{x} = \begin{cases} 640 * i + x, & W > 640 * (i+1) \\ W - 640 + x, & W \le 640 * (i+1) \end{cases} \tag{29}$$

$$\tilde{y} = \begin{cases} 640 * j + y, & H > 640 * (j+1) \\ H - 640 + y, & H \le 640 * (j+1) \end{cases} \tag{30}$$

where x and y are the pixel indexes, W is the width of the original picture, H is the height of the original picture, and

$$0 \le i \le \text{int}\,(W/640) \tag{31}$$

$$0 \le j \le \text{int}\,(H/640)\,. \tag{32}$$

Then, we can get more than 1900 pictures for the test dataset, and there are close to 1800 pictures with the pepper fruit and more than 120 pictures without the pepper fruit.

C. Performance Metrics

To evaluate the performance of our algorithm, we build the dataset to test.

True positive (TP) and true negative (TN) correspond to images, in which the presence or absence of the pepper fruit was correctly recognized. False positive (FP) and false negative (FN) correspond to images, in which the presence or absence of the pepper fruit was not correctly recognized.

The recall rate is

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{33}$$

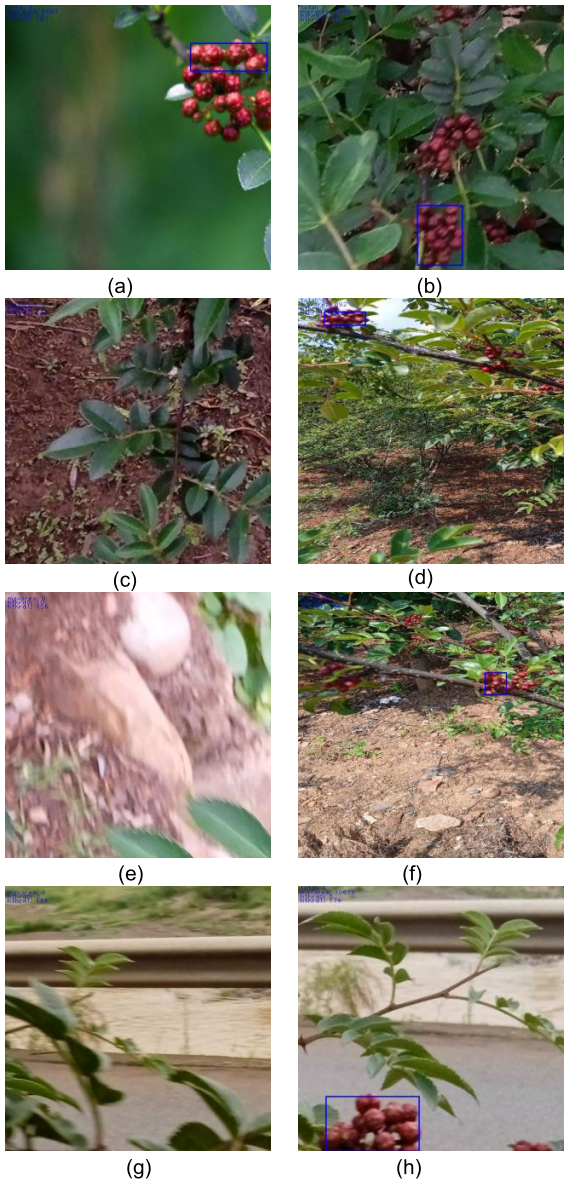


Fig. 19. Detection result of our proposed algorithm. (a) Sunny. (b) Cloudy. (c) Soil. (d) Pepper with soil. (e) Rock. (f) Pepper with rock. (g) Road. (h) Pepper with road.

The accuracy is

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{34}$$

The false alarm is

$$\text{FA} = \frac{\text{FN}}{\text{TN} + \text{FN}}. \tag{35}$$

Beyond the above classic measurement, we add another two measurements to evaluate the location precision of the detection fruit.

The first is the ratio of the overlap area between the real rectangle and the ideal rectangle of the fruit region

$$\text{Roa} = \frac{A_o}{A_{\text{ideal}}} \tag{36}$$

TABLE III
PERFORMANCE WITH DIFFERENT VALUES OF B

B	recall	accuracy	false alarm	Roa	coordinate error
230	99.72%	99.47%	4.00%	70.63%	6.68%
240	99.77%	99.57%	3.20%	68.54%	6.99%
250	99.88%	99.78%	1.60%	66.44%	7.27%
260	100.00%	100.00%	0.00%	64.20%	7.71%
270	100.00%	100.00%	0.00%	61.41%	8.34%
fixed	95.09%	90.95%	58.40%	86.22%	13.35%

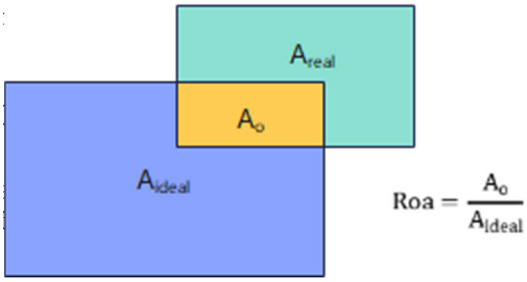


Fig. 20. Overlap area ratio definition.

where A_o is the overlap area, and A_{ideal} is the ideal rectangle area. The ideal region is manually labeled according to the fruit cluster width and height.

The Roa definition can be shown in Fig. 20. The second is the coordinate error, which is defined by the ratio of the central coordinate difference between the real and the ideal contrast with the ideal coordination width or height

$$x_{\text{ce}} = \frac{|x_{\text{ideal}} - x_{\text{real}}|}{w_{\text{ideal}}} \tag{37}$$

$$y_{\text{ce}} = \frac{|y_{\text{ideal}} - y_{\text{real}}|}{h_{\text{ideal}}} \tag{38}$$

where x_{ideal} and x_{real} are separately the ideal and real central coordinate of the x-axis, y_{ideal} and y_{real} are separately the ideal and real central coordinate of the y-axis, and w_{ideal} and h_{ideal} are separately the ideal width and height.

The coordinate error definition can be shown in Fig. 21.

D. Result Analysis

The performance result with the different parameters is shown in Fig. 22 and Table III. The coordinate error in Table III is the average of x_{ce} and y_{ce}. From Fig. 22 or Table III, we can watch out that the false alarm of the fixed parameter is very bad, and since the picture number without the pepper fruit is a small ratio, the recall and accuracy are still good.

While the parameter value of B increases from 230 to 270, the recall, accuracy, and false alarm become better. When the parameter is above 250, the false alarm is zero, and the location error is below 10%. If false detection happens, the picking system will operate wrong, and the picking system default is at high risk when the picking system recognizes the rock or soil as the pepper fruit. So, we usually choose the parameter to reduce the false alarm as possible, and 260 is the best parameter of B.

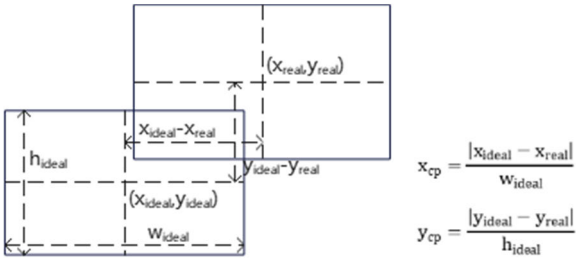


Fig. 21. Coordinate error definition.

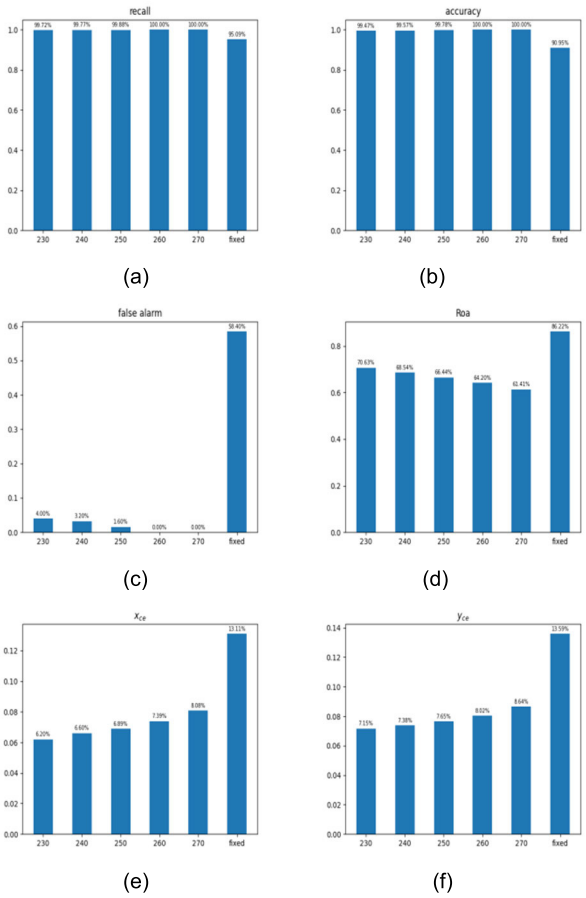


Fig. 22. Detection performance. (a) Recall. (b) Accuracy. (c) False alarm. (d) Roa. (e) x_{ce}. (f) y_{ce}.

With the best parameter, the recall rate, accuracy, and false alarm of our proposed method can separately achieve 100%, 100%, and 0%, and the ratio of the overlap area can get above 64%, and the coordination error is about 8%. All performance is greatly improved compared with the fixed threshold.

While the parameter value of B increases from 230 to 270, the ratio of the overlap area becomes smaller, and the location error becomes higher. The overlap area becomes smaller, which can be watched out through Fig. 24, which is the mask of different parameters. When the parameter value of B increases to 250, there is no connection between the left pepper fruit region and the right one as Fig. 24(c) and (d). Since we only get the maximal contour of the region, the maximal region is left, and the detection result is left as Fig. 23(b). When the parameter value of B increases to 260, there is no connection

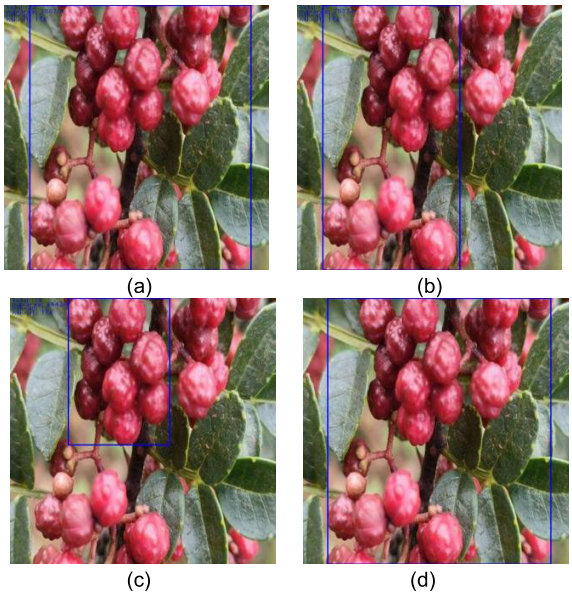


Fig. 23. Detection with different parameters. (a) 230. (b) 250. (c) 260. (d) Ideal.

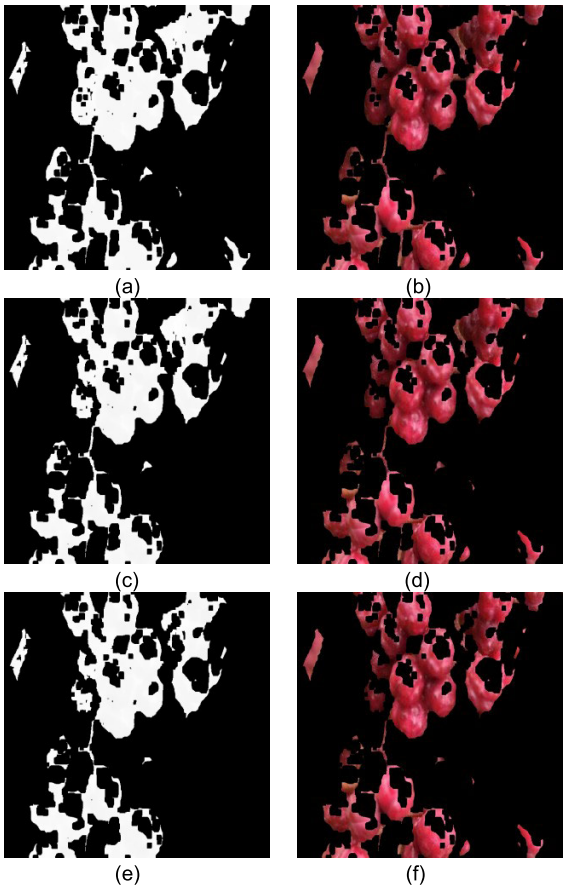


Fig. 24. Mask and region with different parameters. (a) 230 mask. (b) 230 region. (c) 250 mask. (d) 250 region. (e) 260 mask. (f) 260 region.

between the top pepper fruit region and the bottom one as Fig. 24(e) and (f), and the detection result is top as Fig. 23(c).

In addition, the overlap area ratio with the fixed parameter increases and the location error increases at the same time, that is because the detection region is much bigger due to the false alarm, so the region includes the ideal detection part, which makes the overlap area ratio bigger and the region is out of range of the ideal detection part, which makes the location error bigger. It can be watched out from Fig. 25,

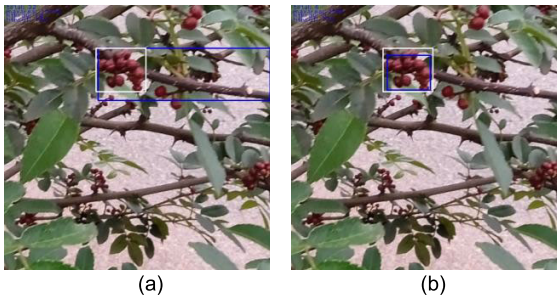


Fig. 25. Detection with fixed and adaptive parameter. (a) Fixed. (b) 230.

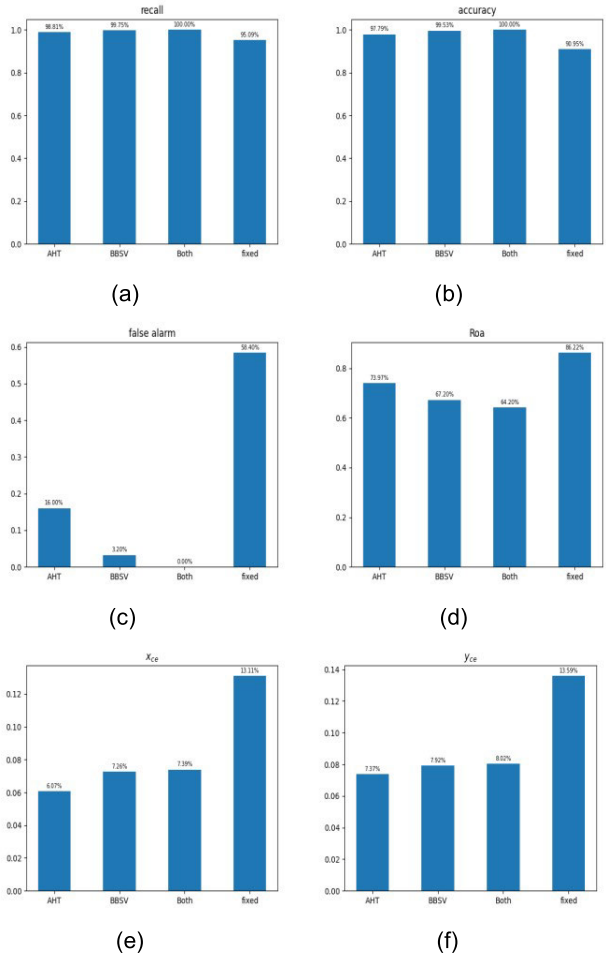


Fig. 26. BBSV and AHT comparison. (a) Recall. (b) Accuracy. (c) False alarm. (d) Roa. (e) x_{ce}. (f) y_{ce}.

TABLE IV
PERFORMANCE COMPARISON

method	recall	accuracy	false alarm	Roa	coordinate error
AHT	98.81%	97.79%	16.00%	73.91%	6.72%
BBSV	99.75%	99.53%	3.20%	67.20%	7.59%
BOTH	100.00%	100.00%	0.00%	64.20%	7.71%
fixed	95.09%	90.95%	58.40%	86.22%	13.35%

where the white rectangle is the ideal detection part and the blue rectangle is the real detection part.

Now, we can compare the function of the BBSV and AHT algorithms. Fig. 26 and Table IV are the result. The coordinate error in Table IV is the average of x_{ce} and y_{ce}. From Fig. 26 or Table IV, we can see BBSV is a bigger function, but both BBSV and AHT take effect.

VI. Conclusion

We address a method including BBSV and AHT to detect the *zanthoxylum bungeanum* fruit of the photograph from an automatic picking system. The recall rate, accuracy, and false alarm of our method can separately achieve 100%, 100%, and 0%, and to evaluate the location precision, we address two metrics: the ratio of the overlap area between the real detection rectangle region and the ideal detection rectangle region and the coordination error between the real x-axis/y-axis and ideal x-axis/y-axis. The ratio of the overlap area of our method can get above 64%, and the coordination error is about 8%. All performance has been greatly improved compared with the empirical fixed threshold.

References

[1] Z. Zhihui, Z. Yong, L. Xueyin, G. Heng, and Y. Yonglin, "Visual recognition of pepper picker based on deep learning," *Machinery*, vol. 48, no. 11, pp. 17–24, Nov. 2021.

[2] W. Yunpeng et al., "Chinese pepper picking tool designs and evaluations based on the TRIZ theory and the triangular fuzzy number," in *Proc. 10th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Oct. 2017, pp. 14–16.

[3] Z. Wenbin, "Design of a new combing teeth pepper picking machine," *J. Agricult. Mechanization Res.*, vol. 37, no. 9, pp. 147–150, Sep. 2015.

[4] G. Ya, Z. Qinling, and J. Cunshu, "Automatic visual recognition picking device of *zanthoxylum bungeanum* and its control method," Chinese Patent 111 480 457 A, Aug. 4, 2020.

[5] L. Xueyin et al., "An intelligent picking device of *zanthoxylum bungeanum* based on visual recognition," Chinese Patent 111 758 397 A, Oct. 13, 2020.

[6] L. Xueyin, Z. Mengwei, F. Yu, L. Bolin, L. Jun, and L. Xiang, "A self propelled pepper picking robot based on visual collaboration system," Chinese Patent 111 758 396 A, Oct. 13, 2020.

[7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[8] J. Liu and W. Li, "The automatic thresholding of gray-level pictures via two-dimensional Otsu method," *Acta Automatica Sinica*, vol. 19, no. 1, pp. 101–105, Jan. 1993.

[9] X. Jing, J. Li, and Y. Liu, "Image segmentation based on 3-D maximum between-cluster variance," *Acta Automatica Sinica*, vol. 31, no. 9, pp. 1281–1285, Sep. 2003.

[10] C. Wang and J. Watada, "Robust color image segmentation by Karhunen–Loeve transform based Otsu multi-thresholding and K-means clustering," in *Proc. 5th Int. Conf. Genetic Evol. Comput.*, Aug. 2011, pp. 377–380.

[11] N. M. Nimbarte and M. M. Mushrif, "Multi-level thresholding algorithm for color image segmentation," in *Proc. 2nd Int. Conf. Comput. Eng. Appl.*, 2010, pp. 231–233.

[12] L. D. S. Pacifico, T. B. Ludermir, and L. F. S. Britto, "A hybrid improved group search optimization and Otsu method for color image segmentation," in *Proc. 7th Brazilian Conf. Intell. Syst. (BRACIS)*, Oct. 2018, pp. 296–301.

[13] W. Fangxin, B. Mingchang, H. Zhiyang, and H. Xiaopeng, "Identification of Chinese prickly ash under the natural scenes," *J. Chin. Agricult. Mechanization*, vol. 37, no. 10, pp. 115–119, Oct. 2016.

[14] H. Yadav, P. Bansal, and R. K. Sunkaria, "Color dependent K-means clustering for color image segmentation of colored medical images," in *Proc. 1st Int. Conf. Next Gener. Comput. Technol. (NGCT)*, Sep. 2015, pp. 858–862.

[15] Y. Ping and G. Zhicheng, "Vision recognition and location solution of *Zanthoxylum bungeanum* picking robot," *J. Heibei Agricult. Univ.*, vol. 43, no. 3, pp. 121–129, May 2020.

[16] K. Liu, C. Yu, and L. Lai, "Navel orange recognition method based on improved Otsu algorithm," in *Proc. Int. Conf. Intell. Comput., Autom. Syst. (ICICAS)*, Dec. 2019, pp. 113–117.

[17] N. Chumuang, S. Thaiparnit, and M. Ketcham, "Algorithm design in leaf surface separation by degree in HSV color model and estimation of leaf area by linear regression," in *Proc. 12th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Nov. 2016, pp. 628–631.

[18] Q. Ruili, C. Manlong, Y. Zonghao, and D. Min, "Image segmentation of Sichuan pepper based on HSV model and improved Otsu algorithm," *J. Chin. Agricult. Mechanization*, vol. 40, no. 11, pp. 155–160, Nov. 2019.

[19] W. Jie, C. Manlong, L. Kui, D. Min, and W. Kun, "Prickly ash image recognition based on HSV and shape feature fusion," *J. Chin. Agricult. Mechanization*, vol. 42, no. 10, pp. 180–185, Oct. 2021.

[20] R. D. F. Feitosa, A. D. S. Soares, and L. C. Pereyra, "A new clustering-based thresholding method for human skin segmentation using HSV color space," in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Jun. 2018, pp. 1177–1180.

[21] Y. Ito, C. Premachandra, S. Sumathipala, H. W. H. Premachandra, and B. S. Sudantha, "Tactile paving detection by dynamic thresholding based on HSV space analysis for developing a walking support system," *IEEE Access*, vol. 9, pp. 20358–20367, 2021.

[22] C. Yang and X. Li, "The green fluorescence image segmentation with the combination of HSV model and watershed algorithm," in *Proc. 6th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Sep. 2015, pp. 607–610.

[23] Z. Bojie, W. Dong, S. Weizhong, L. Yu, and W. Ke, "Research on tea bud identification technology based on HSI/HSV color transformation," in *Proc. 6th Int. Conf. Inf. Sci. Control Eng. (ICISCE)*, Dec. 2019, pp. 511–515.

[24] C. Xiang and J. Mao, "Research on target detection method based on HSV fusion Gaussian mixture model," in *Proc. 3rd Int. Conf. Electron. Inf. Technol. Comput. Eng. (EITCE)*, Oct. 2019, pp. 327–331.

[25] J. Das and H. Roy, "Human face detection in color images using HSV color histogram and WLD," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Nov. 2014, pp. 198–202.

[26] F. Yan, X. Xu, and N. Han, "Identification method of forest fire based on color space," in *Proc. 2nd Int. Conf. Ind. Mechatronics Autom.*, May 2010, pp. 448–451.

[27] D. S. Y. Kartika and D. Herumurti, "Koi fish classification based on HSV color space," in *Proc. Int. Conf. Inf. Commun. Technol. Syst. (ICTS)*, Oct. 2016, pp. 96–100.

[28] E. Prasetyo, R. D. Adityo, N. Suciati, and C. Fatichah, "Mango leaf image segmentation on HSV and YCbCr color spaces using Otsu thresholding," in *Proc. 3rd Int. Conf. Sci. Technol.-Comput. (ICST)*, Jul. 2017, pp. 99–103.

[29] Z. Yongmei and L. Juxia, "Study on automatic recognition technology of ripe pepper fruit," *Agricult. Technol. Equip.*, vol. 349, no. 1, pp. 4–6, Jan. 2019.

[30] D. Liu and J. Yu, "Otsu method and K-means," in *Proc. 9th Int. Conf. Hybrid Intell. Syst.*, Aug. 2009, pp. 344–349.

Longke He (Member, IEEE) received the B.S. degree in microelectronics and solid state electronics from Sichuan University, Chengdu, China, in 1999, and the Ph.D. degree in microelectronics and solid state electronics from the Shanghai Institute of Microsystems and Information Technology, Chinese Academy of Sciences, Shanghai, China, in 2005.

From 2006 to 2020, he was an Algorithm Design Engineer with the Shanghai Institute of Huawei Technologies Company Ltd., Shanghai, and Chengdu Institute of Huawei Technologies Company Ltd., Chengdu. Since 2021, he has been an Associate Professor with the School of Information Technology, Xichang University, Xichang, China. He is the author of more than 20 inventions, including six U.S. Patents and five European Patents. His research interests include 6G, wireless communication, wireless positioning, machine learning, and machine vision.

Dr. He is the member of China Association of Artificial Intelligence.

Xiao Cheng received the Ph.D. degree in physics from Chongqing University, Chongqing, China, in 2021.

Since 2021, he has been a Lecturer with the School of Information Technology, Xichang University, Xichang, China. He has published four articles in various academic journals. His research direction is quantum transport in research field of the theoretical condensed matter physics. The research results during the doctoral period have important reference significance for the design of new organic semiconductor electronic devices in the future. At present, he is also engaged in the research on the application of machine learning in physics.

Aying Jiwa received the B.S. degree in computer science and technology and the M.S. degree in computer technology from Sichuan University, Chengdu, China, in 2015 and 2018, respectively.

From 2019 to 2020, he was a Data Engineer with Chengdu Kuancheng Technologies Company Ltd., Chengdu, China. Since 2021, he has been a Teaching Assistant with the School of Information Technology, Xichang University, Xichang, China. His research interests include machine learning and natural language processing (NLP).

Dan Li received the B.S. degree in electronic information engineering from Xijing University, Xi'an, China, in 2014, and the M.S. degree in computer science from Guizhou Normal University, Guiyang, China, in 2017.

From 2017 to 2020, she was an Algorithm Engineer at Chengdu Techman Software Company Ltd., Chengdu, China. Since 2020, she has been a Teacher of Electronic Engineering with the School of Xichang University, Xichang, China. She is the author of more than ten articles, including SCI and EI. Her research interests include signal processing, machine learning, deep learning, and machine vision.

Jing Fang received the B.S. and M.S. degrees in software engineering from the Beijing University of Technology, Beijing, China, in 2014 and 2017, respectively.

She was a Software Development Engineer at the Bank of China Software Development Center, Shanghai, China, from 2017 to 2018. She then was a Teacher of Software Engineering at the Sichuan College of Applied Sciences, Chengdu, China, from 2018 to 2019. Since 2019, she has been an Assistant in computer science and technology with Xichang University, Xichang, China. The major research was information security.

Zhencong Du received the B.S. degree in applied electronic technology from the Chengdu University of Technology, Chengdu, China, in 1997, and the Ph.D. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, in 2008.

His research interests include biomedical information processing and signal processing.