Crop Analysis Based on MATLAB Image Enhancement Technology

Xuehui Cui¹, *, Bo Ma², *
¹College of Electrical and information Engineering, Southwest MinZu University of China, Chengdu, Sichuan, 610225, P.R. China
²College of Electrical and information Engineering, Southwest MinZu University of China, Chengdu, Sichuan, 610225, P.R. China
*Corresponding author e-mail: 173822011@qq.com, 21466304633@qq.com,

Abstract. In response to the call of national precision agriculture, this paper proposes a method to easily detect the characteristics and quantities of agricultural products. On the platform of MATLAB software, we transform grayscale images, equalize histogram and enhance image with exponential low-pass filter. Next, we extract the binary features of the target in the image by using the regionprops function. Then, the binary image of grayscale image is filtered twice by median filter to obtain the number of targets in the image. By comparing and verifying the image effect, it can be seen that this method has obvious superiority in image recognition and extraction analysis, and can be applied in modern agricultural detection.

1. Introduction
With the advent of the era of high-speed information technology, the application of computer image processing is more extensive. Traditional data based on limited measurements can’t meet the requirements of precision agriculture in terms of accuracy and convenience. Besides, traditional manual management methods can’t obtain agricultural information timely and efficiently. Taking above conditions into consideration, this paper focuses on agricultural products, and uses computer image processing and analysis technology, aiming at automatically identifying and understanding images to detect the characteristics and quantities of agricultural products.

In 1991, Miller used near-infrared recognition technology, which made the accuracy rate of image recognition reach more than 60%. However, the noise problem in the image still exists, easily producing miscarriage of justice [1]. Casady used machine vision technology to measure rice canopy size [2]. Critten used Fourier analysis identify crops and weeds [3], but techniques by which shade is converted into shape are inherently unreliable. In view of the existing problems in agricultural products detection, this paper combines image enhancement algorithm with agricultural product feature recognition by MTALAB programming simulation, carrying out rapid and accurate agricultural products detection, which is more efficient than conventional agricultural products test methods. I believe that in the future, it will be more widely used in agricultural products detection.
2. Image Enhancement Technology

Image enhancement technology is one of the research objects of digital image processing, which plays a significant role in improving image quality [4]. But image enhancement is a paradoxical process. On the one hand, image enhancement technology hopes to enhance edges and remove noise. On the other hand, removing noise will blur the edges to a certain extent, while enhancing edges will increase noise meanwhile. Therefore, it is necessary to balance the two when considering image enhancement. The image enhancement method can be represented by the following figure.

![Image enhancement method](image)

**Fig. 1** Image enhancement method

We choose the combination of spatial domain method and frequency domain method. The former processes the pixels directly, which facilitates the recognition of the target area and the shape extraction while the latter filters out unnecessary components to make the image smooth.

2.1. Gray Level Transformation and Histogram

2.1.1. Dispersed Image The discrete function of the histogram of a digital image with a gray level of \([0, L-1]\) is:

\[
h(r_k) = n_k
\]  

(1)

Where \(r_k\) is the k-level grayscale, \(n_k\) is the number of pixels of the grayscale \(r_k\) in the image. The probabilistic estimations are obtained by dividing each of its values in the image by the total number of pixels \(n\):

\[
P(r_k) = \frac{n_k}{n} \quad (0 \leq r \leq 1, k = 0, 1, 2...L - 1)
\]  

(2)

The gray level is normalized to the interval \([0, 1]\), \(r = 0\) represents black, and \(r = 1\) represents white. The gray level in the interval \([0, 1]\) obtained by each pixel is random. Then the gray level of the image can be regarded as a random variable of the interval \([0, 1]\), and the probability density function \(p_r(r)\) can be used to represent the gray distribution of the original image. Let \(s\) be the enhanced image gray level pixel value, and the probability density function \(p_s(s)\) can be used to
represent the gray distribution of the enhanced image. For any gray level $s$ of the original image in the interval $[0, 1]$, we utilize the following conversion to obtain the output gray level $s$.

$$s = T(r)$$  \hspace{1cm} (3)

The inverse transform function from $s$ to $r$ can be expressed as follows.

$$r = T^{-1}(s) \quad (0 \leq s \leq 1)$$  \hspace{1cm} (4)

It is known from the basic probability theory that if $p_r(r)$ and $T(r)$ are known, there are:

$$p_s(s) = p_r(r) \cdot \frac{d}{ds}[T^{-1}(s)] = \left[ p_r(r) \cdot \frac{dr}{ds} \right]_{r=r^{-1}(s)}$$  \hspace{1cm} (5)

Therefore, the basis of the histogram technique is to control the probability density function of the gray level of the input image by the transformation function $T(r)$, and change the gray level of the output image, so as to enhance the image.

2.1.2. Continuous Image
Continuous image satisfies the following formula.

$$s = T(r) = \int_0^r p_r(r) dr$$  \hspace{1cm} (6)

Differentiate on $r$ on both sides of the above formula yields

$$\frac{ds}{dr} = p_r(r).$$  \hspace{1cm} (7)

Bring the result into (5) to get equation below.

$$p_s(s) = \left[ p_r(r) \cdot \frac{1}{p_r(r)} \right]_{r=r^{-1}(s)} = 1$$  \hspace{1cm} (8)

It can be seen that after the transform processing, an image with uniform probability density function of gray scale distribution can be generated. The gray level of the image is relatively well-distributed and covers the whole range $[0, 1]$.

2.1.3. MATLAB Implementation
Use the imadjust function provided in the MATLAB image processing toolbox to perform grayscale transformation to enhance image contrast. The grayscale image reflects the grayscale distribution and reflects the statistical relationship between each gray level and its pixel frequency.
As can be seen in the figure, the original image has a small dynamic range and is dark overall. It concentrates in a certain gray scale range. After the gray scale processing, the image becomes bright and the details are presented clearly.

The gray level elements between 0.1 and 0.9 of the original image are highlighted. We use the imhist and histeq functions provided in the MATLAB image processing toolbox to convert the histogram of the original image into a form of uniform distribution, which makes the grayscale spacing of the image open and increases the contrast.
It can be seen that the detail effect is more prominent, the amount of information is larger, the contrast is higher, and the dynamic range of the gray level of the image is enhanced.

2.2. Low-pass Filter
Since the energy of the image is mostly concentrated in the low and mid bands of the amplitude spectrum, the edges and noise of the image correspond to the high frequency portion. Therefore, a filter that reduces the amplitude of the high-frequency component can reduce the influence of noise. Consequently, we choose low-pass filter there. The following is the working flow chart of frequency domain processing.

![Flow chart of frequency domain processing.](image)

The general formula for an exponential low-pass filter is as follows.

\[
H(u, v) = e^{-\left[\frac{D(u, v)}{D_0}\right]^n}
\]  

(9)

Where, \( D_0 \) is cut-off frequency, equal to the value of \( D(u, v) \) when \( H(u, v) \) falls to 1/2 of its maximum value usually, \( n \) is attenuation coefficient. We think that \( D_0 \) is equal to the value of \( D(u, v) \) when \( H(u, v) \) falls to \( \frac{1}{\sqrt{2}} \) of its maximum value on the basis of the experimental results. The above formula can be changed as follows.

\[
H(u, v) = e^{-0.65\left[\frac{D(u, v)}{D_0}\right]^n}
\]  

(10)
First we add salt and pepper noise to get the following figure.

![Image with salt & pepper noise](image)

**Fig. 6** Image with salt & pepper noise

After Fourier transform, the following figure is obtained.

![Image after Fourier transform](image)

**Fig. 7** After Fourier transform

![Image after shifting to the middle](image)

**Fig. 8** After shifting to the middle

It looks black. In fact, there is a white spot in the upper left corner of the whole picture. That is, the upper left corner is missing something. We shift the frequency to the middle. We enhance the above image for the purpose of seeing more clearly, getting the image below. We can get the spectrum at the same time.

![Image after enhancing the white spot](image)

**Fig. 9** After enhancing the white spot

![Frequency spectrum](image)

**Fig. 10** Frequency spectrum

![Comparison before and after low-pass filter](image)

**Fig. 11** Comparison before and after low-pass filter

From the above picture, we can see that using low-pass filter to process the image with salt and pepper noise can make the image visually sound, and look a lot smoother. But the details are affected at the same time, leading to some ambiguity in the whole.
3. Image extraction
We directly use `im2bw` function to obtain image segmentation. Then use the `bwboundaries` function to display the boundaries of the targets in the image.

![Image segmentation](image1.png)  ![Display the boundaries](image2.png)

We use `bwlabel` function to mark successive areas in the image, pseudo-colorize them, and display them in turn with their digital markers.

![Mark successive areas](image3.png)

We choose the `regionprops` function to extract the following binary features for each target in the image.

| Area | Centroid | BoundingBox | SubarrayIdx | MajorAxisLength | MinorAxisLength | Eccentricity | Orientation | ConvexHull |
|------|----------|--------------|-------------|-----------------|-----------------|--------------|-------------|------------|
| 30119 | [178.5139, ...] | [0.5000, 0.2895, ...] | 1x2 cell | 326.4847 | 241.5854 | 0.8727 | 24.9929 | cell double |
| 1319 | [33.0364, ...] | [0.5000, 0.1975, ...] | 1x2 cell | 46.2179 | 39.4265 | 0.5218 | 4.0869 | cell double |
| 1 | [9.218] | [0.5000, 0.2175, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |
| 1 | [30.491] | [0.5000, 0.4905, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | cell double |
| 1 | [39.201] | [0.5000, 0.3085, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |
| 1 | [71.296] | [0.5000, 0.2955, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |
| 188 | [82.5213, ...] | [0.5000, 0.2965, ...] | 1x2 cell | 29.0441 | 11.0642 | 0.9246 | -68.2040 | double |
| 1 | [73.497] | [0.5000, 0.4965, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |
| 1970 | [101.7041, ...] | [0.5000, 0.2345, ...] | 1x2 cell | 55.2979 | 46.5006 | 0.4803 | 83.0512 | double |
| 5 | 79.2000, ...] | [0.5000, 0.5115, ...] | 1x2 cell | 3.3066 | 2.1292 | 0.7051 | -18.4349 | double |
| 13 | [80.5385, ...] | [0.5000, 0.3225, ...] | 1x2 cell | 7.6421 | 2.2881 | 0.9541 | 87.8554 | double |
| 4 | [86.5125, ...] | [0.5000, 0.5115, ...] | 1x2 cell | 3.0551 | 2.0817 | 0.7319 | 0 | double |
| 2 | [88.5080, ...] | [0.5000, 0.5075, ...] | 1x2 cell | 2.3094 | 1.1547 | 0.8660 | 90 | double |
| 2 | [118.5000, ...] | [0.5000, 0.3225, ...] | 1x2 cell | 2.3094 | 1.1547 | 0.8660 | 0 | cell double |
| 1 | [120.419] | [0.5000, 0.418, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |
| 1 | [129.366] | [0.5000, 0.359, ...] | 1x2 cell | 1.1547 | 1.1547 | 0 | 0 | double |

![Data from MATLAB](image4.png)

From here we can see a lot of data, such as area, eccentricity and so on. It can also express the feature vector map more clearly. Take the area and thinness ratio as an example.
Next, we calculate the number of flowers according to the flow chart below.

After calculation, it is concluded that “Number = 14”. Actually, there are 15 flowers. Thus, we can come to the conclusion the deviation is so small that we can ignore it.

4. Summary
This paper proposes a method to easily detect the characteristics and quantities of agricultural products. Through MATLAB simulation, the combination of image enhancement technology and image extraction technology is achieved, and the contrast and gray dynamic range of the image are improved to a certain extent. We have access to agricultural information more conveniently and faster.
Acknowledgements
This work was partially supported by Sichuan Youth Science and Technology Innovation Research Team (2017TD0028). Also was supported by the Fundamental Research Funds for Central University, Southwest Minzu University (2017NZYQN45).

References
[1] MILLERBK, DELWICHE M J. Peach defect detection with machine vision [J]. Transactions of the ASAE, 1991, 34 (6): 2588-2597.
[2] Casady W W, Singh N, Costello T A. Machine Vision for Measurement of Rice Canopy Dimensions [J]. Transactions of the ASAE, 1996, 39 (5): 1891-1898.
[3] Critten D L. Fourier based techniques for the identification of plants and weeds [J]. Journal of agricultural engineering research, 1996, 64 (2): 149-154.
[4] Rahman Z, Jobson D J, Woodell G A. Retinex processing for automatic image enhancement [J]. Journal of Electronic imaging, 2004, 13 (1): 100-111. Rahman Z, Jobson D J, Woodell G A. Retinex processing for automatic image enhancement [J]. Journal of Electronic imaging, 2004, 13 (1): 100-111.