2019 International Conference on Computer Intelligent Systems and Network Remote Control (CISNRC 2019)
ISBN: 978-1-60595-651-0

Egg Image Feature Value Extraction Method Based on OpenCV+Python

Shubin Yang, Zonglin Jiang and Wei Pan

ABSTRACT

Aiming at the high efficiency development of embedded products with egg size classification and the poor ability of developing code transplantation, this paper proposes an image feature value extraction method based on OpenCV+Python to measure the egg shape index(aspect ratio of the minimum circumscribed rectanglee) and other characteristic values of egg images. Firstly, the image of the egg is preprocessed by OpenCV+Python, then the image is edge-detected with the Canny operator. Finally, the contour region is extracted and the feature value is extracted based on the edge detection and the commonly used API in OpenCV. Experiments show that the method can accurately and effectively extract the feature values of the egg image, and the code amount is small and the transplantation ability is strong.

KEYWORDS

OpenCV+Python, Egg image, Edge Detection, Feature Extraction.

PREFACE

At present, experts and scholars and enterprise developers at home and abroad have spent a lot of time developing machine vision systems for quality inspection and automatic grading of agricultural and sideline products. Such as rice disease detection and classification according to the size and shape of eggs, walnuts, apples, etc. According to statistics, the annual per capita consumption demand of eggs in the world in 2019 reached 11.9 kg, and will continue to grow to 2040 at an annual growth rate of 0.1 kg. Therefore, it is extremely urgent to fundamentally solve the problems of quality inspection and grading of fresh eggs in China.

In the development of feature extraction methods for egg images, MATLAB software is mostly used as a development platform in China. For example, when Zhizhong

Shubin Yang, Zonglin Jiang, Wei Pan
School of Electrical and Information Engineering, Wuhan Institute of Technology, Wuhan, 430205, China
Hubei Provincial Key Laboratory of Intelligent Robot, Wuhan, 430205, China

Xing studied egg freshness, MATLAB was used as the development platform for egg image edge detection technology. Although the development was simple, the algorithm migration ability was poor, and it could only be run on a PC, which greatly limited the extraction conditions of egg image feature values[1]. Jingyi Ouyang also uses MATLAB as a development platform when studying the external crack detection of eggs. Although the code is small, it is only suitable for ordinary learning research. It is difficult to realize the industrial embedded products based on computer vision egg detection series, and MATLAB software is expensive[2]. In this paper, an image extraction method based on OpenCV+Python is proposed, which can effectively mark the outline of the egg image, and can obtain the contour length, the area of the feature area, and the egg shape index[3]these key feature values.

Python is an object-oriented design language, the third largest language after JAVA and C++. OpenCV also has a large number of optimized computer vision libraries, and can be easily ported to OpenCL (Open Computing Language) compatible devices (such as GPU, Graphics Processing Unit) through the conventional API interface. This method not only has the same development efficiency as MATLAB for image processing, but also has low cost.

INTRODUCTION TO EXPERIMENTAL PLATFORM CONSTRUCTION AND IMAGE PREPROCESSING METHOD

OPENCV+PYTHON ENVIRONMENT SETUP

The OpenCV+Python environment is the premise for intelligent image processing. Select https://www.python.org/downloads/ to select the corresponding version of Python for download. It is recommended to select Python3.X (such as Python 3.60) to download during the download process, which is more practical and expandable than Python 2.X. Python 3.X is the mainstream trend in Python development in the new era. In addition, Open CV installation in Windows CMD mode, if the Python environment has been configured globally in the computer, you can directly use the pip package management tool to install, open cmd direct input: pip3 install OpenCV + Python, through Python integration.

PRETREATMENT OF EGG IMAGES

Grayscale can improve the processing speed of egg image feature recognition. Because the RGB original image can be reduced to the occupied memory of the original image after being converted into a grayscale image. Binary image means that the image value matrix only retains 0 and 1 values to represent black and white. In the actual egg image processing experiment, it is the key to choose the appropriate threshold. Generally, the empirical method is used. Binarization can divide the egg and the background and highlight the egg target.

Among them, the most commonly used techniques for threshold processing are the TRIANGLE method and the OSTU method. The principle of the OTSU method is to use the threshold to divide the original image into foreground and background. When the optimal threshold is taken, the difference between the background and the foreground is the largest. At this time, a criterion for measuring the difference is needed. The standard for measuring the difference is the maximum inter-class variance[4-5].

For the feature value extraction requirements of the egg image, the original color image of the collected egg should be grayed out. The experimental results are shown in Figure.1(a) and Figure.1 (b). The core code is written as follows:

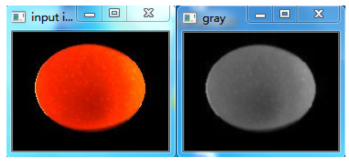


Figure.1(a) Original image Figure.2(b) OSTU thresholding.

```
src=cv.imread("D:/vcprojects/images/egg.bmp")        #input RGB eggs image
gray = cv.cvtColor(image, cv.COLOR_BGR2GRAY) #gray
cv.imshow("gray",gray)                                #show the image
```

Considering the uneven illumination, illumination angle and other issues, there is some noise in the egg image. Based on Python combined with OpenCV's powerful open source library, the above two threshold processing methods are used respectively, and the experimental results are compared. The comparison found that the image of the OTSU threshold processing method has no salt and pepper noise, and the effect is better. The results of the two threshold processing experiments are shown in Figure.2(a) and Figure.2(b). The core code is as follows:

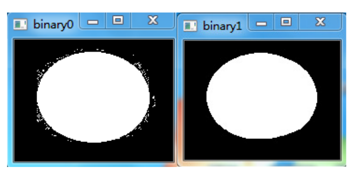


Figure 2. (a)TRIANGLE thresholding Figure 2. (b) OSTU thresholding.

```
def threshold_demo(image):
gray = cv.cvtColor(image, cv.COLOR_RGB2GRAY)
#Threshold processing using the TRIANGLE method
ret,binary=cv.threshold(gray,0,255)
cv.THRESH_BINARY | cv.THRESH_TRIANGLE) TRIANGLE
cv.namedWindow("binary0", cv.WINDOW_AUTOSIZE)
cv.imshow("binary0", binary)
```

```
def local_threshold(image):
gray = cv.cvtColor(image, cv.COLOR_RGB2GRAY)
    #Threshold processing using the OTSU method
ret,binary=cv.threshold(gray,0,255)
    cv.THRESH_BINARY | cv.THRESH_OTSU)
cv.namedWindow("binary1", cv.WINDOW_AUTOSIZE)
cv.imshow("binary1", binary)
```

SEGMENTATION OF EGG IMAGES - CANNY EDGE OPERATOR

The Canny edge operator is an optimal filter that uses a first-order directional derivative in any direction of a two-dimensional Gaussian function as a filter, which is filtered by convolving with the image; then finding the local maximum of the image gradient to determine the image. edge. According to the calculation of the signal-to-noise ratio and the positioning product, the optimal approximation operator is obtained.

The Canny edge detection algorithm is generally divided into the following four steps:

1) smoothing the image using a Gaussian filter;

2) Converting the original RGB multi-channel image into a single-channel grayscale image using grayscale conversion;

3) using the difference of the first-order partial derivative to calculate the magnitude and direction of the gradient;

4) Perform non-maximum suppression on the amplitude of 3) by finding local maxima in the image gradient and then making the other non-local maxima zero to obtain the refined edge;

5) Use the high and low threshold algorithms to detect edges.

The mathematical description of the key steps is as follows:

1) The two-dimensional Gaussian function is:

$$G(x, y) = \frac{1}{2\pi\sigma^2}\exp[-\frac{(x^2+y^2)}{2\sigma^2}] \qquad (1)$$

The first-order directional derivative of the Gaussian function $G(x, y)$ in a certain direction n is：

$$G_n = \frac{\partial G(x, y)}{\partial n} = n\nabla G(x, y) \qquad (2)$$

$$n = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \qquad (3)$$

$$\nabla G(x,y) = \begin{bmatrix} \dfrac{\partial G}{\partial x} \\ \dfrac{\partial G}{\partial y} \end{bmatrix} \quad (4)$$

In the formula, n is direction vector, $\nabla G(x,y)$ is gradient vector.

Image $f(x,y)$ and G_n are convoluted while changing the direction of n, where $G_n * f(x,y)$ taking the maximum value is the direction orthogonal to the detected edge.

2) Calculation of the magnitude and direction of the gradient:

$$E_x = \frac{\partial G(x,y)}{\partial x} * f(x,y) \quad (5)$$

$$E_y = \frac{\partial G(x,y)}{\partial y} * f(x,y) \quad (6)$$

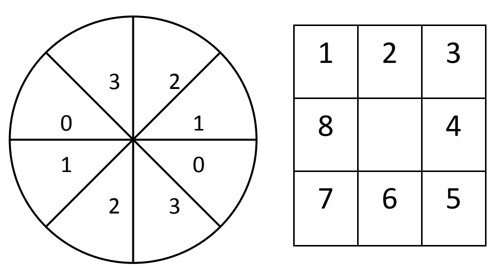
Order $A(x,y) = \sqrt{E_x{}^2 + E_y{}^2}$

$$\theta = \arctan(\frac{E_y}{E_x}) \quad (7)$$

In the formula, $A(x,y)$ reflects the edge intensity at (x,y) points of the image, and θ is the normal vector at (x,y) points of the image.

3) Non-maximal suppression of gradient amplitude

The gradient angle is discretized into one of four sectors, and a 3x3 window is used for the suppression operation. The four sectors are numbered from 0 to 3. At each point, the center pixel $M(x,y)$ of the neighborhood is compared to two pixels along the gradient line, as shown in Figure.3.



b) 3x3 area

Figure 3. Non-maximum suppression.

4) Double threshold algorithm for edge discrimination and join edges

①Edge discrimination: the edge intensity is greater than the high threshold is the edge point. It is not the edge point that the edge intensity is less than the threshold; If the edge intensity is greater than the low threshold and less than the high threshold, it is determined whether the adjacent image of the pixel exceeds the edge point of the high threshold. If it is exceeded, it is the edge point, and vice versa.

②Connection edge: The double threshold algorithm applies two values $T1$ and $T2$ to the non-maximum suppression image, and $2T1 = T2$ so that two threshold edge images $G_1(x, y)$ and $G_2(x, y)$ can be obtained[6-8].

In OpenCV+Python, each step of the edge detection of the egg image using the Canny edge detector has a corresponding API, and the operation is very simple. The experimental results are shown in Figure.4(a) and Figure.4(b). The shown and complete code is as follows:

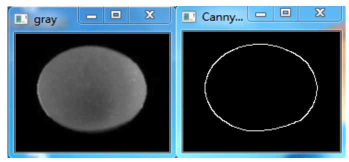


Figure 4. (a) Grayed image. Figure 4. (b) Canny Edge detection.

```
blurred = cv.GaussianBlur(image, (3, 3), 0)          #Gaussian filtering to smooth the noise
gray = cv.cvtColor(blurred, cv.COLOR_BGR2GRAY)#grayscale conversion
cv.imshow("gray", gray)#display grayscale image
xgrad = cv.Sobel(gray, cv.CV_16SC1, 1, 0)          # X Gradient
ygrad = cv.Sobel(gray, cv.CV_16SC1, 0, 1)          # Y Gradient
edge_output = cv.Canny(xgrad, ygrad, 35, 165)      #edge，35,165 are two thresholds T
```

EGG IMAGE CONTOUR DISCOVERY AND GEOMETRIC FEATURE EXTRACTION

CONTOUR DISCOVERY

A contour is a series of pixels that form an ordered set of points. Contour discovery is a method for finding the outline of an object based on the basis of image edge extraction.n OpenCV, an output point set is provided. The findContours() in the common API calculates the outline from the binary image. It can use the Canny() function to process the image because such an image contains edges pixel. According to the description in Section 2, the edge of the image is extracted by the Canny operator, and the output result can be analyzed by the corresponding contour. In addition, the binary image processed by the threshold is simultaneously used as an input of the contour finding, thereby finally obtaining the found contour. The declaration of this function is as follows[9]:

```
findContours(canny_output,contours,hierarchy,CV_RETR_EXTERNAL,CV_CHAIN_APPROX_SIMPLE,Point(0,0))
```

canny_output: Use the canny operator to extract the result graph of the edge, this image must be a binary image;

Contours: Extract contour map;

Hierarchy: Topological information about the output image;

CV_RETR_EXTERNAL: Contour retrieval mode;

CV_CHAIN_APPROX_SIMPLE: Contour approximation method;

Point(): Offset.

Similarly, OpenCV also provides a function drawContours to draw the contours found. Its function is declared as follows:

drawContours(drawing, contours, i, color, CV_FILLED,8, hierarchy,0,Point())

Drawing: the target image;

Contours: the contours of all inputs;

i:specify that the outline is drawn;

Color: the color of the outline;

CV_FILLED: the width of the line;

8: the degree of connection of the line;

hierarchy :This parameter is required only when you only need to draw some outlines:

0: The maximum level of the drawn outline, when 0, means that only the specified outline is drawn;

Point(): Offset.

GEOMETRIC FEATURE EXTRACTION

The geometric features of an image refer to a series of features such as the position, perimeter, and area of an object in an image. The image is typically pre-processed before the image geometry is extracted. The steps have been given above and the desired experimental results have been obtained.

The characteristic values of the eggs extracted mainly in this paper are arc length, area and egg shape index. Among them, the arc length and the area are based on one pixel unit. In engineering practice, it is necessary to obtain the measurement units such as mm^2 and cm^2 after conversion. Technically speaking, based on the edge detection in Section 2 above, the edge is a series of points, each of which has a length, and the closed curve can obtain the arc length and area. Similarly, use the cv.arcLength and cv.contourArea APIs respectively to calculate the arc length and area. In addition, you can use the cv.boundingRect API to get the circumscribing rectangle of the feature area, and then get the egg shape index. Eggs of different sizes have different arc lengths, cross-sectional areas and (egg shape index). These three eigenvalues can be used as a measure of the size of an egg, making good preparations for subsequent egg size grading [10].

In the contour discovery and feature extraction, the experimental results are shown in Figure.5 (a) and Figure.5(b), part of the core code is as follows:

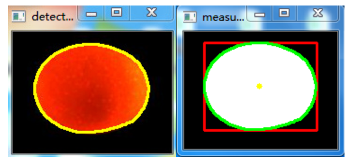


Figure 5. (a) Findcontours image. Figure 5. (b) Contours and boundingrect.

```
    contours, hireachy = cv.findContours(binary, cv.RETR_EXTERNAL,
cv.CHAIN_APPROX_SIMPLE)
for i, contour in enumerate(contours):
area = cv.contourArea(contour)                     #contourArea
length = cv.arcLength(contour,True)                #contourlength
    x, y, w, h = cv.boundingRect(contour)          #boundingRect
    rate = min(w, h)/max(w, h)                     #1/egg shape index
    if approxCurve.shape[0] > 6:
cv.drawContours(dst, contours, i, (0, 255,255), 2)     #The outline of the
outline polygon fitting number >6 is yellow
```

Using the same method, select an egg with a larger arc length, area, and mass than the first experimental sample, numbered 2, and the experimental results obtained twice are shown in Table I below:

TABLE I. CHARACTERISTIC VALUE OF AN EGG SAMPLE IMAGE.

Geometric Features / Sample	Arc length pixel number	Area pixel number	1/Egg shape index
Egg No. 1 image	369.705	9485.0	0.776
Egg No. 2 image	386.534	10047.0	0.784

The above experiments show that the relationship between the egg shape index and the quality in the reference[3] is consistent, that is, the larger the egg shape index and the smaller the mass, the eigenvalues extracted by the method are reliable, which provides support for the size grading of the eggs on the assembly line.

CONCLUSION

This paper proposes a method for extracting eigenvalues of egg images based on OpenCV+Python. The preprocessing processes such as image graying and OTSU method for global optimal binarization are briefly described. The principle of Canny operator edge detection is described in detail, and the core code of each function of OpenCV is implemented in the Python development platform. The outline of the egg image is contoured by the Python language and the OpenCV common API, and the arc length, area, and egg shape index are calculated. The method is fast in development, powerful in algorithmic portability, and easy to deploy to embedded products based on machine vision and automated egg grading. At the same time, the method fits well with the grading of egg size grading, laying a foundation for machine vision-based automation.

ACKNOWLEDGMENTS

Shubin Yang Corresponding author: yshubin@wit.edu.cn

This paper was funded by the Hubei Provincial Key Laboratory Open Fund for Intelligent Robots (No. HBIR201406).

REFERENCES

1. Zhizhong Xing, Haidong Zhang. Egg freshness detection based on computer vision and neural network. Jiangsu Agricultural Sciences, 2017(11).
2. Jingyi Ouyang, Muhua Liu. Computer Vision Based Egg Crack Detection Methods. Journal of Agricultural Mechanization Research, 2012, 34(3).
3. Yi-lin Ren, Jianying Liu, Lirong Xiong . Research on the Basic Physics Properties of Eggs. Journal of Agricultural Mechanization Research, 2005(3):191-192.
4. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins. Digital Image Processing Using Matlab (sencond edition). Beijing:Electronic Industry Press, 2014, 1.
5. Han Xiaodong, Wang Haosen, Wang Shuo. The application of Python in image processing. Beijing Surveying and Mapping, 2018.
6. Qi Li, Yuwei Ji, Zhipeng Wang, Xuan Dou. Design of road icing detection system based on OpenCV+Python. Journal of Shaanxi University of Science and Technology (Natural Science Edition), 2017, 35(2).
7. Zhiyao Chen. Design and Application Research of Image Segmentation Technology Based on OpenCV+Python. China New Communication, 2018, 20(19):94.
8. Yueguan Zhang, Zeng Qinghua, Jianye Liu. An Improved Image Edge Detection Algorithm Based on Canny Algorithm. Navigation and control, 2019, 18(01):89-95.
9. Nixon et al. Feature Extraction and Image Processing. Beijing: Publishing House of Electronics Industry, 2010, 10.
10. Duan Yufei. Study on multi-quality high-throughput online rapid non-destructive testing of eggs. 2017.