High Precision - Low-Cost Apple Feature Recognition Model

Yilin Lian [#], Xuexue Song [#], Wei Wei [*]

School of Science, Nanjing Forestry University, Nanjing, China, 210037

* Corresponding Author Email: ww2020@njfu.edu.cn

[#]These authors contributed equally.

Abstract. This paper addresses the image recognition problem in apple-picking robots and proposes a high-precision, low-cost apple feature recognition model to improve efficiency and accuracy. First, based on object detection algorithms, the study develops an apple contour recognition model using morphological methods, alongside a K-means clustering model for data analysis. These models are used to determine the approximate spatial distribution of apple positions and classify apple maturity levels. Second, by integrating apple contour recognition with numerical simulation optimization techniques, the distribution range of apple quality is estimated. Finally, simulation experiments conducted on a given dataset demonstrate that the identified apple positions, maturity, and calculated quality align closely with empirical data, validating the model's effectiveness.

Keywords: Deep learning, Target detection, Fast R-CNN, K-means.

1. Introduction

China is the global leader in fruit production and consumption, with apples being one of the most widely cultivated and consumed fruits. The production and planting area of apples rank first in the world, and apple production plays a key role in the agricultural economy[1]. However, traditional production methods are inefficient and costly. With advancements in technology, fruit-picking robots have gradually become an effective tool for improving production efficiency and reducing labor costs[2]. The integration of modern agriculture and computer vision technology has driven the development of smart agriculture, with significant progress in the application of computer vision in apple detection[3].

Traditional apple detection methods mainly rely on image processing techniques, using characteristics such as fruit color and shape for segmentation. In recent years, deep learning algorithms, which extract deep features through convolutional neural networks, have further improved detection accuracy. Deep learning models such as Faster R-CNN[4,5], YOLOv3[6,7], and Mask R-CNN[8] have been widely applied to real-time detection of apple fruits with good detection results. However, the complex natural environment still poses challenges for apple detection, with factors such as leaf occlusion, fruit overlap, and low lighting conditions increasing the difficulty of recognition.

This paper proposes an apple recognition method based on deep learning and object detection technology for apple images in natural environments. By combining morphological methods and the K-means clustering model, the distribution of apple positions and maturity classification are derived, and the distribution range of apple quality is obtained through numerical simulation optimization. Finally, simulation experiments are conducted to verify the model's accuracy in apple position, maturity recognition, and quality calculation.

2. The basic funamental of Target Detection

2.1. Image preprocessing

The images from the dataset (soured from https://www.saikr.com/vse/apmcm/2023) are read for processing. Since images captured in natural environments are easily affected by natural light, occlusions, and shadows, the three components of the BGR color space are closely related to brightness. In other words, any change in brightness will result in corresponding changes to these three components. Therefore, it is necessary to convert the input image to the HSV color space for processing, adjust the HUE channel values, and then convert the modified HSV image back to the BGR space.

The formula to convert input image to HSV image is as follows:

$$\begin{cases} b = h * 255 \\ g = s * 255 \\ r = v * 255 \end{cases} \tag{1}$$

Next, increase the brightness factor of the image by a factor of 1.2 to better identify apples.

2.2. The structure of Target Detection

Object detection is an important technology in the field of computer vision, aiming to identify the object categories and positions within images or videos[9-12]. With the development of deep learning, significant progress has been made in the field of object detection, enabling not only the recognition of objects present in images but also the marking of object positions with bounding boxes. Currently, this technology is widely applied in fields such as automated harvesting and facial recognition. This section primarily introduces the Fast R-CNN algorithm.

Fast R-CNN[13], proposed in 2015, introduced improvements that enhanced the speed and efficiency of object detection while maintaining high accuracy. Its main process is as follows:

(1) Fast R-CNN takes the preprocessed image and a set of proposal boxes as input, processes the entire image through several convolutional layers and max-pooling layers to obtain a convolutional feature map.

(2) For each proposal box, the Region of Interest Pooling operation extracts the proposed region from the feature map and pools these regions into fixed-size feature matrices.

(3) Each feature map is then passed through a series of fully connected layers (FCs) to integrate these features.

(4) The integrated features are passed through a probability classifier and a bounding box regressor for output, followed by applying Non-Maximum Suppression (NMS) to remove redundant regions.

The detection flowchart of Fast R-CNN is shown in Figure 1:

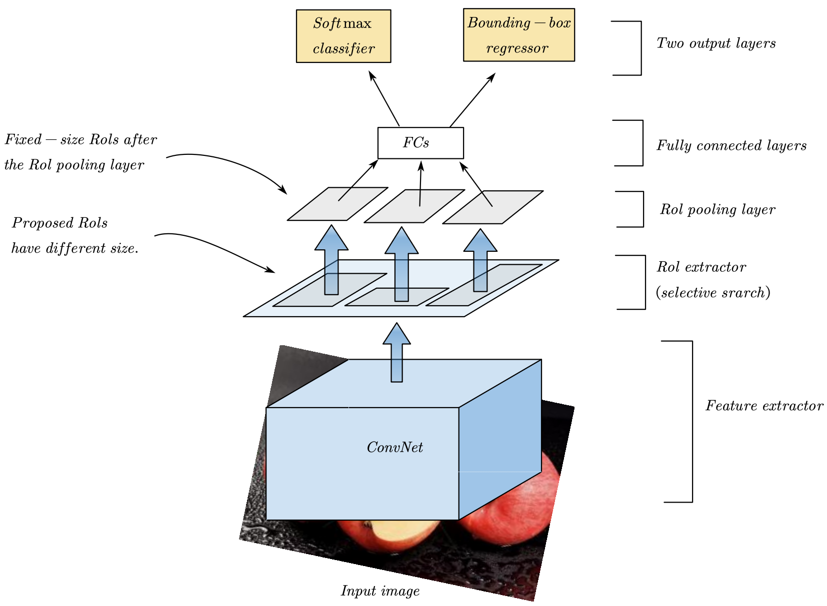

Figure 1: Fast R-CNN Detection Flowchart

For example, using image 161.jpg, Fast R-CNN extracts image similarities such as color, texture, size, and spatial filling factors, and outputs the feature map after integrating the above metrics. The results are shown in Figures 2 and 3:

Figure 2 161.jpg Part of the apple label

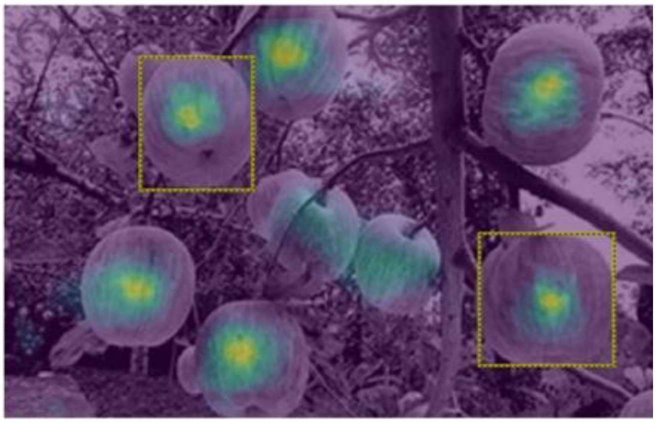

Figure 3 Density map generated by 161.jpg

2.3. Apple Contour Detection Based on Morphological

In the morphological apple contour detection, morphological operations are used to highlight and refine the apple contours to improve positioning accuracy. First, the image is converted into a grayscale image and simplified, followed by thresholding to convert it into a binary image. Morphological operations are then applied to enhance the contours. The specific operations include:

(1) Dilation: Expands the target area, filling small holes or connecting adjacent targets.

(2) Erosion: Shrinks the target area, removing noise or separating targets.

(3)Opening Operation: Performs dilation followed by erosion, removing noise while preserving the shape of the target.

(4)Closing Operation: Performs erosion followed by dilation, filling small holes and connecting targets.

In object detection, the model outputs a bounding box to represent the location of the detected object. The bounding box is typically defined by four coordinates: the top-left corner (x_1, y_1) and the bottom-right corner (x_2, y_2). These four coordinates define a rectangular region that encompasses the object.

In this section, the origin is defined at the bottom-left corner of each image. The x-coordinate increases from left to right, and the y-coordinate decreases from top to bottom. If (x_1, y_1) represents the top-left corner of the bounding box and (x_2, y_2) represents the bottom-right corner, the center position of the object can be obtained by calculating the geometric center of the bounding box. This can be achieved by taking the average of the x and y coordinates, as follows:

$$\begin{cases} x = \dfrac{x_1 + x_2}{2} \\ y = \dfrac{y_1 + y_2}{2} \end{cases} \tag{2}$$

2.4. Apple Ripeness Identification Based on K-means Clustering Method

The maturity of apples can be identified by calculating and comparing the color features of detected objects using object detection algorithms. According to research, the maturity stages of apples can be roughly divided into three categories: immature, semi-mature, and fully mature, corresponding to the colors green, yellow-green, and red, with RGB reference values of (0,128,0), (128,128,0), and (255,0,0), respectively.

For each detected apple, the average RGB color within its contour is calculated. This involves summing the color values of all pixels within the contour and dividing by the total number of pixels to obtain the average color value, which reflects the overall color tendency of each apple. Therefore, in the third task, calculating the average color value for each apple can be used to differentiate between maturity levels.

To distinguish the maturity of apples, the K-means clustering method can be employed[14]. The workflow is as follows:

(1) Find a point P_{11} in the space, and use the KD-Tree to find the nearest n points to it. Judge the distance between these n points and P_{11}. Place the points with distances less than the threshold r, such as $P_{12}, P_{13}, P_{14}, \cdots$ in class Q.

(2)Find a point P_{12} in $Q(P_{11})$ and repeat step 1.

(3)In $Q(P_{11}, P_{12})$,find a point and repeat step 1 to find $P_{22}, P_{23}, P_{24}, \cdots$ and place them all in Q.

(4)When Q cannot have any new points added, the search is completed.

The reference colors for immature, semi-mature, and mature fruits are defined. The average color within the detected apple contours is calculated, and its Euclidean distance is computed using the reference values defined for the three maturity levels. Clustering analysis is then performed to determine the maturity of all apples.

Define the reference colors for unripe, semi-ripe, and ripe fruits. Calculate the color mean within the detected apple contour and calculate its Euclidean distance with the defined reference values of the three maturity levels. Cluster analysis determines the maturity of all apples. First, based on the RGB values of apples at different maturity levels, select k initial cluster centers C. Then calculate the Euclidean distance between other values and the data center:

$$\mathrm{dist}(s, C) = \sqrt{\sum_{i=1}^{k} (s - C_i)^2} \tag{3}$$

2.5. Apple Mass Calculation

Based on the object detection model established in Section 2.2, the apple contours in the image are obtained, and the contour area is calculated by counting the number of pixels within the contour. According to the reference data, the benchmark mass and average cross-sectional area for apples at three different maturity levels (immature, semi-mature, and mature) are known. Based on this data, the following steps can be carried out:

First, the average area of apples with the same maturity level in the image is divided by the corresponding actual average area of apples at that maturity level, resulting in the magnification factor for image a:

$$a = \frac{S_a}{S_b} \tag{4}$$

In this context, S_a represents the average area of all apples in the image with the specified maturity level, while S_b represents the average area of apples at the corresponding maturity level in reality.

Next, based on the obtained cross-sectional area of each apple, the volume of each apple in the image is calculated. The resulting volume is then divided by the magnification factor to obtain the real volume of the apples in the image V_t:

$$V_t = \frac{V}{a} \tag{5}$$

Here, V represents the volume of a specific apple in the image.

By dividing the obtained actual volume by the average density of apples at the corresponding maturity level and then multiplying by the benchmark weight at that maturity level, the mass of each apple in the image, denoted as m, can be determined:

$$m = m_x \cdot \frac{V_t}{V_b} \tag{6}$$

In the formula, m_x represents the benchmark mass corresponding to the maturity level (the benchmark masses for immature, semi-mature, and mature apples are 150g, 180g, and 225g, respectively), while V_b represents the average volume of apples at the corresponding maturity level.

Using the formula for the area of a circle and the volume of a sphere, the final expression for the apple mass is derived as follows:

$$m = m_x \cdot \frac{S^{\frac{3}{2}} S_b^{\frac{1}{2}}}{S_a^{2}} \tag{7}$$

3. Results and Analysis

In this section, Fast R-CNN object detection technique (as shown in Figure 1) will be used to perform feature extraction on the images depicted in Figures 2 and 3. The distribution of the apple center positions, the ripeness and quality of each apple are obtained, and the results are analyzed and discussed separately.

3.1. The result of the apple position distribution

The resolution of all the images is 270×185. Then use a 3×3 convolution kernel and the common apple colors : red, green, and pink to create a mask to search for the contours of the apples throughout the entire image. During this process, it is necessary to determine a base number to filter out targets with contours smaller than the base number, ensuring that the detected contours are all apples. However, the relative size of the apples in each image varies. After multiple attempts, we use 25% of the maximum contour in the image as the base number, discarding contours smaller than this base number. Then, we calculate the center coordinates of these contours. The final scatter plot of all the apple coordinates in Attachment is as follows:

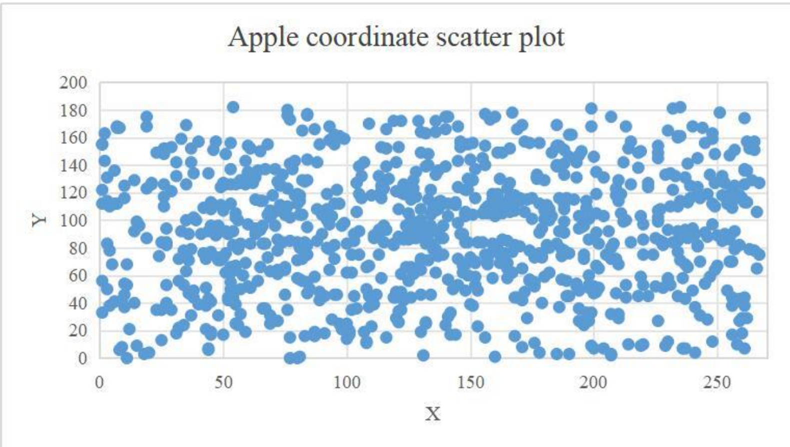

Figure 4 Apple coordinate scatter plot

The image provides information about the location of each apple in the image. The apples are distributed evenly in all directions in the coordinate graph. This indicates that the model has good adaptability, greatly reducing the probability of inability to recognize apples in the boundary region of the image. The results obtained are reliable.

3.2. The result of the apple maturity

To solve the question of apple’s ripeness, each apple’s contour has been obtained. Calculate the average RGB color of each contour and calculate the distance between it and the standard color for classification. Finally, we obtained the classification of three levels of ripeness for apples:

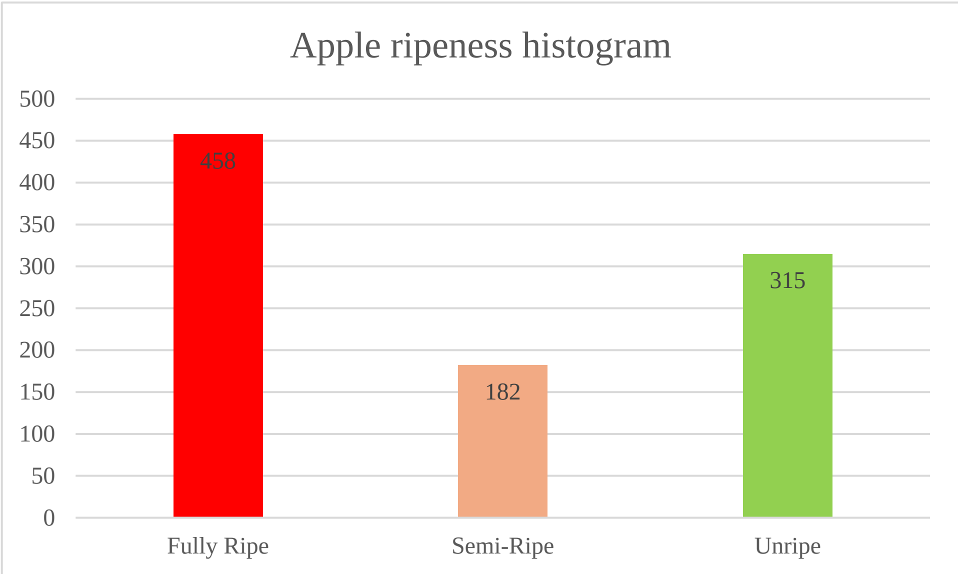

Figure 5 Apple ripeness histogram

To assess the accuracy of the maturity judgment model, we randomly selected 3 images and compared their judged results with the actual results, as shown in the table 1 below:

Table 1 Comparison of apple ripeness judgment results with real results

Picture	Act Full Ripe	Act Semi-ripe	Act Unripe	Pre Full Ripe	Pre Unripe
60.jpg	2	0	0	2	0
96.jpg	0	1	0	0	0
152.jpg	14	1	10	9	9

From Table 1, it can be seen that the judgment of model of apple ripeness is almost the same as the actual situation. It is particularly accurate in identifying semi-ripe and unripe fruits, and the recognition error for ripe fruits is also relatively small. Therefore, this model can accurately identify the ripeness of apples.

3.3. The result of the apple mass calculation

Based on the size of each apple's contour, the mass can be calculated using equation (7). The histogram of the distribution of all apple masses is as follows:

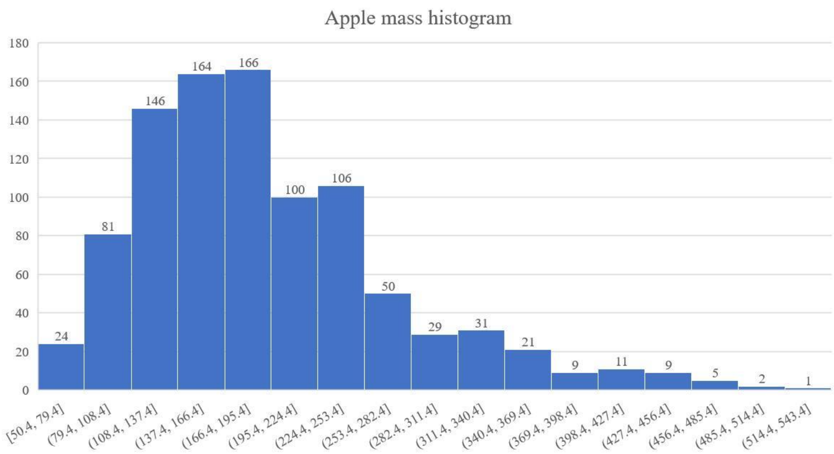

Figure 6 Apple mass histogram

Obtain the outline of the apple through morphological operations, calculate the area of the apple by summing pixel points, derive the formula for calculating the apple's weight through mathematical reasoning, and calculate the result as shown in Figure 6. The recognition effect is perfect, indicating that the establishment of this model is relatively reasonable.

4. Conclusions

This paper addresses the image recognition problem in apple-picking robots by combining deep learning and object detection technologies. First, based on object detection algorithms, the study develops an apple contour recognition model using morphological methods, alongside a K-means clustering model for approximate the apple position distribution and classify its maturity. Next, by integrating numerical simulation optimization, the distribution range of apple quality is further derived, enhancing the practical value of the model. Finally, simulation experiments are conducted on a given dataset. The results show that the proposed model can accurately identify the positions and maturity of apples, and the calculated quality closely matches the actual data, in line with the expected empirical laws. The experimental results validate the effectiveness and reliability of the model, providing theoretical support for precise recognition in apple-picking robots for real-world production applications.

Acknowledgements

The authors gratefully acknowledge the financial support from the Jiangsu Provincial General Innovation Program (Project No.202210298179H) under the Industry-University Collaboration Fund.

References

[1] Wang L, Liu J, Gao J M. Spatial distribution pattern and interannual dynamic changes of apples in China [J]. China Agricultural Information, 2019, 31(04): 84-93.

[2] Chen Q, Yin C K, Guo Z L, et al. Research status and development trends of key technologies in apple picking robots [J]. Transactions of the Chinese Society of Agricultural Engineering, 2023, 39(04): 1-15

[3] Qi J W. Research on apple target recognition and positioning algorithm based on deep learning [J]. Electronic Technology & Software Engineering, 2022, (08): 189-92.

[4] SA I, GE Z, DAYOUB F, et al. DeepFruits: A Fruit Detection System Using Deep Neural Networks [J]. Sensors (Basel, Switzerland), 2016, 16.

[5] BARGOTI S, UNDERWOOD J P. Deep fruit detection in orchards [J]. 2017 IEEE International Conference on Robotics and Automation (ICRA), 2016: 3626-33.

[6] TIAN Y, YANG G, WANG Z, et al. Apple detection during different growth stages in orchards using the improved YOLO-V3 model [J]. Comput Electron Agric, 2019, 157: 417-26.

[7] Wu X, Qi Z Y, Wang L J, et al. Apple detection method based on lightweight YOLOv3 convolutional neural network [J]. Journal of Agricultural Mechanization Research, 2020, 51(08): 17-25.

[8] Tian B K. Research on apple detection, classification, and localization technology under complex environments based on deep learning [D], 2020.

[9] Zhu Z, Liang D, Zhang S, et al. Traffic-sign detection and classification in the wild[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 2110-2118.

[10] Russakovsky O, Deng J, Su H, et al. Imagenet large scale visual recognition challenge[J]. International journal of computer vision, 2015, 115: 211-252.

[11] Selvaraju R R, Cogswell M, Das A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization[C]//Proceedings of the IEEE international conference on computer vision. 2017: 618-626.

[12] Chen H, Chen Z, Chai Z, et al. A single-side neural network-aided canonical correlation analysis with applications to fault diagnosis[J]. IEEE Transactions on Cybernetics, 2021, 52(9): 9454-9466.

[13] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

[14] Mou Q S. Research on a universal fruit picking detection model based on deep learning [D], 2021.