

Intelligent Bamboo Part Sorting System Design via Machine Vision

Tian-Hu Liu Zi-Di Wu Qin-Ling Chen Xiang-Ning Nie
Gui-Qi Li Hong-Jun Wang Di Zhang Wei Liu
Jin-Meng Wu

Abstract

The defect rate of initially produced block bamboo (Bambusoideae) parts is >20 percent. Sorting out these defective parts manually is a highly time-consuming and tedious process. An intelligent sorting system was developed based on machine vision using a Radial Basis Function (RBF) neural network learning algorithm in this study. First, a high-speed charge-coupled device camera was used to obtain a series of images of perfect and defective block bamboo parts. Next, the RBF neural-network learning algorithm was applied to obtain defect characteristics and to locate defective parts moving forward on a conveyor belt. An array of air jets was designed to force defective parts off the belt. Experimental results showed that the average defective part removal rate of the proposed system was 91.7 percent.

Bamboo (Bambusoideae) is commonly distributed throughout the Asia-Pacific region, Americas, and Africa, as shown in Figure 1. In Asia, bamboo plants are mainly distributed in China, India, and other developing countries. Bamboo mats are very popular consumer items during the summer months. Rectangle bamboo parts are the primary component of the bamboo mat. The defect rate of initially produced rectangle bamboo parts is >20 percent. Defective parts are sorted out manually in existing production systems, which is time-consuming and tedious. Additionally, as labor costs increase, it is also increasingly necessary to develop a rapid and accurate sorting method to maintain profitability in today's highly competitive manufacturing market.

There have been many previous studies on machine-vision-based sorting methods. For example, Sofu et al. (2016) designed an automatic machine vision system to sort 183 apple samples at three different conveyor belt speeds with 73 percent to 96 percent accuracy. Ohali (2011) created a neural network, computer-vision-based system to sort date fruit into three grades with accuracy of 80 percent. Moallem et al. (2017) proposed a computer-vision-based support vector machine (SVM) classifier to grade golden delicious apples at a recognition rate of 92.5 percent and 89.2 percent for two categories (healthy and defected) and three quality categories (first rank, second rank, and rejected), among 120 apple images, respectively. Arakeri and Lakshmana (2016) built a 96.47 percent accurate automatic tomato-grading system based on computer vision. Zhang et al. (2014) proposed a hybrid fruit classification method based on the fitness scale chaos artificial bee algorithm (FSCABC) and feed-forward neural network (FNN); the system showed

classification accuracy of 89.1 percent. Liu et al. (2019) identified immature and mature pomelo fruits on trees by fitting an elliptic model in the Cr-Cb color space; the recognition accuracy was 93.5 percent. Baltazar et al. (2008) applied data fusion to classify fresh, intact tomatoes based on their respective levels of ripeness.

There have been many other valuable contributions to the literature. Baigvand et al. (2015) developed a fig classification system based on machine vision, which demonstrated

The authors are, respectively, Associate Professor, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (liuparalake@126.com); Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (huanongdidi@163.com); Researcher, Guangdong Academy of Agric. Sci., Guangzhou, Guangdong, China (chenqinling@126.com [corresponding author]); Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (2297734355@qq.com); Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (liguiqi3@163.com); Professor, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (xtwhj@scau.edu.cn); Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (zd15189821268@126.com); Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (kkkliuwei@126.com); and Graduate student, College of Engineering, South China Agric. Univ., Guangzhou, Guangdong, China (891851949@qq.com). This paper was received for publication in May 2020. Article no. FPJ-D-20-00030.

©Forest Products Society 2021.

Forest Prod. J. 71(1):27–38.

doi:10.13073/FPJ-D-20-00030

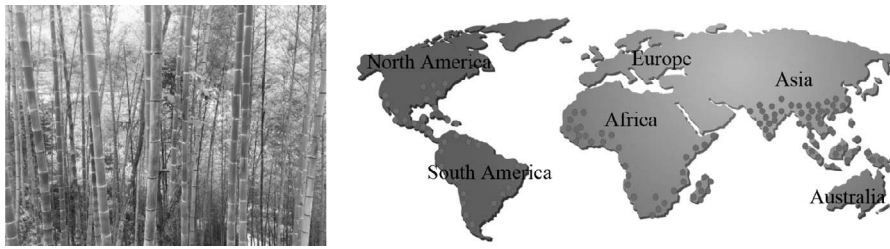


Figure 1.—Bamboo and distribution map of bamboo.

95.2 percent accuracy and 90 kg/h speed. Chen et al. (2019) developed a machine vision system to detect broken, chalky, damaged, or spotted defective rice seeds. Xu and Zhao (2010) classified the size, shape, and color of strawberries via k-means algorithm with size error <5 percent and color and shape accuracy of 88.8 percent and 90 percent, respectively. Yu et al. (2009) designed a machine-vision-based capacitor appearance defect detection system, which was found to be highly efficient and accurate. Gu et al. (2019) used a back propagation neural network to classify mesh defects. Li et al. (2016) introduced a fast-normalized cross correlation (FNCC) –based machine vision algorithm for detecting and counting immature green citrus fruits on outdoor color images; 84.4 percent of the fruits were successfully detected in 59 validation images. Subramanian et al. (2006) built an autonomous guidance system using machine vision and laser radar, which achieved successful guidance within 2.5–2.8 cm while travelling at 3.1 m/second. Hannan et al. (2009) introduced a machine vision algorithm to identify oranges with 90 percent detection accuracy and a 4 percent false-positive rate. Jabo (2011) uses ‘Adaboost’ kernel methodology in wood defect detection and classification. Choi et al. (2015) developed a machine vision system to detect fruit that had been dropped on the ground, which demonstrated 83 percent to 88 percent accuracy. Shin et al. (2012 a, b) developed a computer vision system for inspecting the quantity and size distribution of fruits on conveyor belts.

The production process of bamboo mat includes the following steps: (1) Material quality inspection and cutting; (2) Sifting; (3) Shaping and drilling hole; (4) Heating; (5)

Polish; (6) Sorting; and (7) Assembling (Fig. 2). Band scratches and dents on the upper surface, which will affect the final appearance of bamboo cushions and hurt users, are the main naturally caused defects on the outer surface of bamboo materials. Other defects, such as cracks and irregular shape in the lower surface and the sides, will be checked in the material quality inspection and cutting step. In the sorting step, those bamboo parts that have band scratches and dents on the upper surface must be removed before assembly. In order to achieve automatic sorting of bamboo parts, a machine-vision-based sorting system was established in this study. The proposed system works based on an image segmentation algorithm, which is an entirely novel approach to the best of the authors’ knowledge.

Methods and Materials

Hardware

The hardware of the sorting system is mainly composed of a detection unit, transmission unit, sorting unit, air supply system, and control system (Fig. 3). The detection unit includes a light source, a light source control card, a lighting box, and a charge-coupled device (CCD) camera. The transmission unit is a black conveyer belt mounted on a stainless table. The control system includes an industrial computer, a computer power supply, and an output expansion module that controls the conveyor belt velocity. Figure 4 shows a schematic diagram of the sorting unit, which contains six air jets, six jet valves, an air compressor, a 24-V power supply, and a solenoid valve control card. The

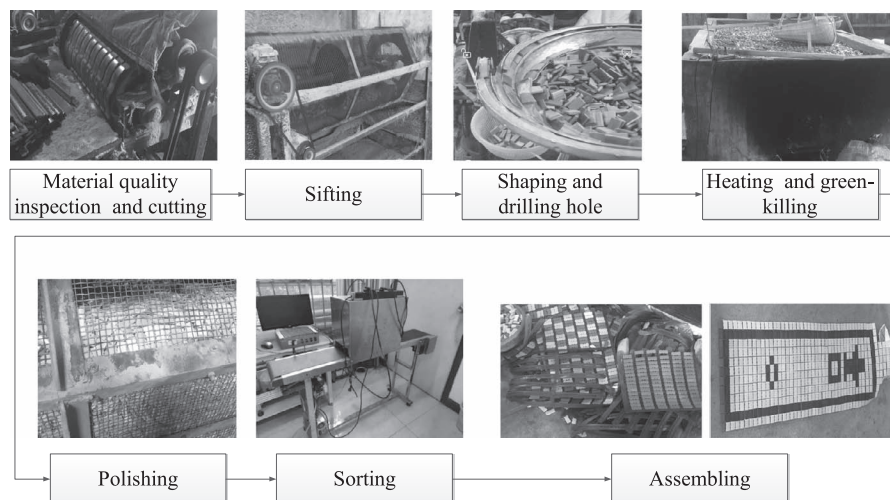


Figure 2.—The production process of the bamboo mat.

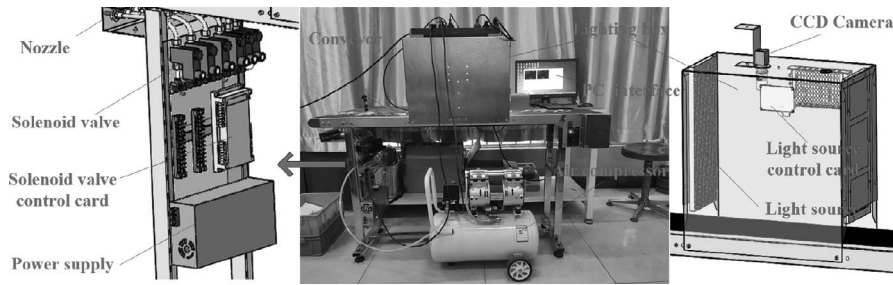


Figure 3.—Bamboo parts sorting platform based on machine vision.

air jet is connected to a solenoid valve, which is connected to the air compressor through a duct.

Workflow

The workflow of the proposed sorting procedure is shown in Figure 5. At the beginning of the sorting process, the electric source of the air compressor is switched on to provide 0.5 mpa of compressed air. The light source is switched on and the resolution of the CCD camera is adjusted to obtain clear images. The belt is then started, and bamboo parts are placed on the conveyor and moved forward. As the bamboo parts enter the CCD camera's capturing field, the camera captures and transmits images to the computer for processing. A software system analyzes the image to locate any defective parts. The defective parts are transported to the end of the conveyor belt, then the computer sends an open signal to the solenoid valve control card. Finally, the corresponding air jet is opened and blows away the defective parts.

Light source

An appropriate light source can present high-contrast images for optimal image analysis. Detection is usually performed in a box to control the lighting as necessary and prevent the interference of external light. A light source was installed in a box to obtain photos with sufficient brightness. Figure 6 showed a few comparisons among light emitting diode (LED), fluorescent lamps, and halogen light, which indicated that the endurance, performance cost ratio, response speed, thermal diffusion, and designability of LED is the highest. Therefore, LED was selected as light source. Using light of the same color as the test object will

brighten the image. Illumination experiment results of using white, red, yellow, blue, green, cyan, and purple LEDs (Fig. 7) indicated that red light source provided the worst contrast between defects and intact parts and yellow light source provided the best, so two yellow LED surface light sources were installed at both sides of the box for sufficient low-angle lighting. The structure of lighting system is shown in Figure 8.

Image acquisition

A total of 150 defective and 150 intact bamboo parts were randomly selected as training samples. A high-speed CCD camera (128 photos/s) was used to capture images. The training sample images were saved as $1,280 \times 1,024$ 24-bit JPG files. Microsoft Visual C++ 2013 and Computer Vision Library (Halcon 17.12, The MVTec, Inc.) softwares were used to realize the proposed segmentation and identification algorithm on an Intel(R) Core (TM) i7-4600 CPU @ 2.10 GHz 2.69 GHz, 4.00 GB RAM industrial computer.

Image preprocessing

Median filtering.—The noise in a digital image mainly originates from image acquisition and transmission processes. The illumination level and sensor stability are the main factors responsible for such noise. Median filtering, average filtering, and adaptive threshold filtering can be used to remove noise from images; median filtering was adopted here to remove salt-and-pepper noise, specifically:

$$y(i) = \text{Med}[x_{(i-N)}, \dots, x_{(i)}, \dots, x_{(i+N)}] \quad (1)$$

Threshold segmentation.—The segmentation process removes background and reveals the contour of bamboo

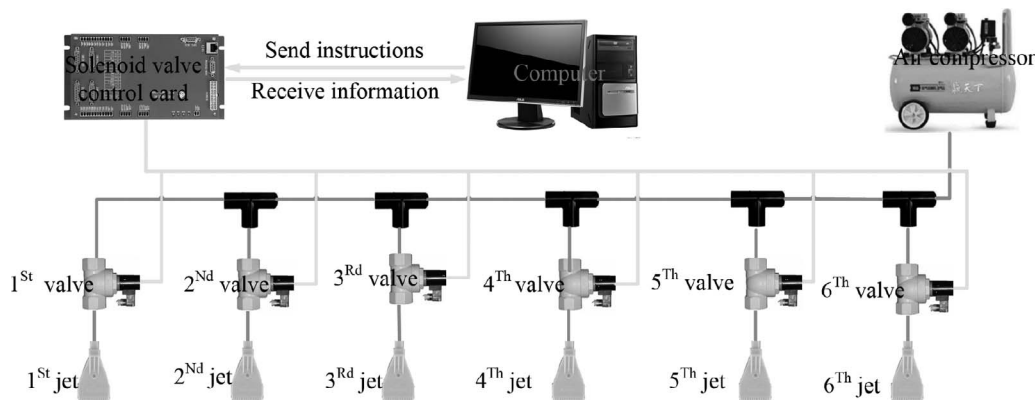


Figure 4.—Schematic diagram of the sorting unit.

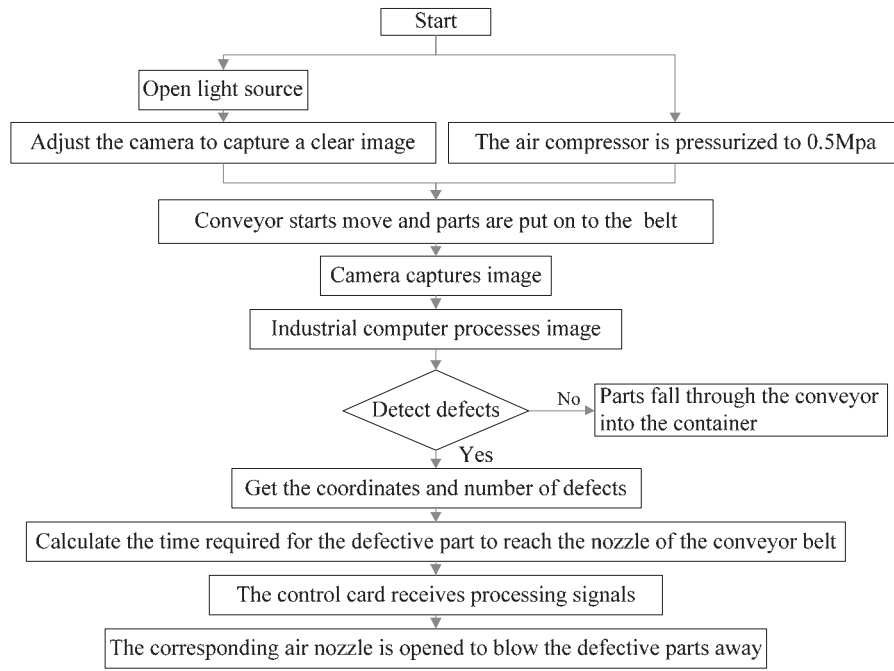


Figure 5.—The workflow of the sorting procedure.

parts. Image backgrounds can be removed by watershed segmentation, color-based segmentation, threshold-based segmentation, and edge-based segmentation. There is a significant difference in color between the bamboo parts and the conveyor belt, so threshold segmentation was selected here to separate the parts from the belt in the image. The histograms of belt background and bamboo part pixels showed in Figure 9 indicated that the grey level of background pixels is lower than 80 and that of bamboo part pixels is higher than 150. According to the results of a series segmentation experiments, the threshold was set to 100 to convert the color image into a gray-scale image and remove most belt background pixels, and the algorithm was operated as

$$B(i,j) = \begin{cases} Y(i,j) & Y(i,j) > 100 \\ 0 & Y(i,j) \leq 100 \end{cases} \quad (2)$$

Area filtering.—After filtering by gray value threshold, there was a great deal of noise in the segmented area yet to be removed. The segmented binary image was labeled using 8-connected components to evaluate the areas of the objects; then two area threshold values, $T_1 = 1,000$ and $T_2 = 100,000$

(Eq. 3), were used to classify the bamboo part and block the remaining noise. Figure 10 shows an original image, its median filtered result, and its filtered defect area.

$$B(i,j) = \begin{cases} Y(i,j) & T_1 \leq \text{Area} \leq T_2 \\ 0 & \text{Area} < T_1 \text{ or } \text{Area} > T_2 \end{cases} \quad (3)$$

Feature extraction.—Two processing methods can be used to extract the characteristics of bamboo parts: (1) extracting all characteristics of all parts simultaneously, or (2) extracting each part in turn. The advantage of the first method is that it can get the shape of defect more precisely. The disadvantage is that it is considerably more time-consuming. The first method was selected after weighing its advantages and disadvantages.

The main purpose of the proposed technique is to separate defective parts from intact parts. Different features such as shape, color, gray value, perimeter, and area are usually used for classification in cases such as this, but the defective area on the bamboo part has no fixed shape, area, or perimeter. The color of the defective area does, however, differ from other areas. Sample images of defect regions, intact parts, and backgrounds along with regions of color

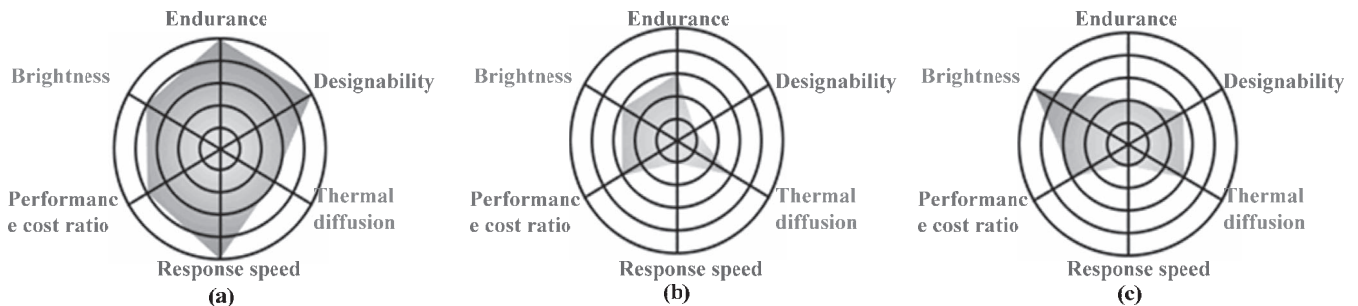


Figure 6.—Comparisons among (a) light-emitting diode (LED), (b) fluorescent lamps, and (c) halogen light.

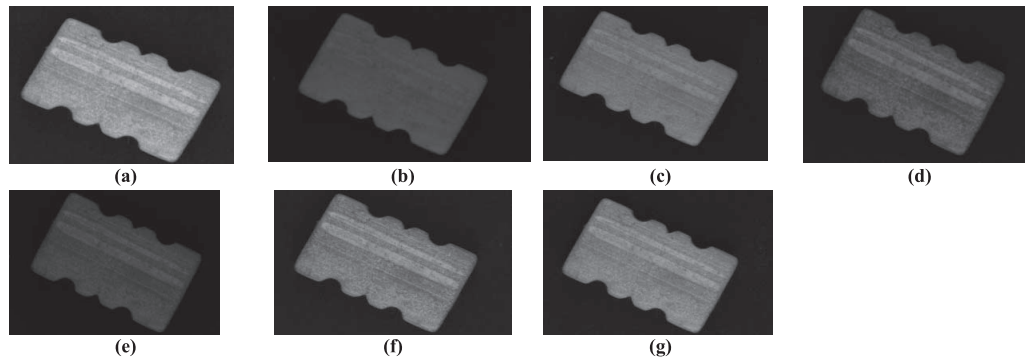


Figure 7.—Pictures of bamboo parts under various light sources, (a) white light source, (b) red light source, (c) yellow light source, (d) blue light source, (e) green light source, (f) cyan light source, and (g) purple light source.

clustered in red–green–blue (RGB) coordinates are shown in Figure 11, which indicated that they are clustered in different areas and can be divided into different groups in RGB coordinates. Therefore, color is feasible as the recognition feature, so it was used here as the feature for defect extraction.

Defect detection and location

The support vector machine (SVM), k-nearest neighbor (KNN), multilayer perceptron (MLP), and radial basis function (RBF) neural network were compared in separate nondefective and defective bamboo parts to determine which best suits the proposed system. The SVM is a generalized linear classifier that classifies binary data under supervised learning; it uses a kernel function $w \times x + b = 1$ to map the nonlinear data to the high-dimensional space, then seeks the hyperplane that decompresses the data in the new space with the maximum margin. The KNN does not use any learning process. The data set has classifying eigenvalues in advance, then is classified directly after receiving new samples. The MLP is a forward artificial neural network that maps a set of input vectors to a set of

output vectors. It consists of multiple node layers, each of which is connected to the next layer. In addition to input nodes, each node is a neuron with a nonlinear activation function. The MLP overcomes the disadvantage of single-layer perceptron, which cannot recognize nonlinear data. The RBF is also a forward network; its hidden layer adopts nonlinear function as the basis function and its output layer is a linear function. The input space is converted to hidden space by the RBF, which makes the original linear indivisible problem separable.

The R, G, and B values of intact, defective, and background areas were selected as three feature quantities for comparison and the identified three areas were taken as an output. The classification time of each of the four classifiers is presented in Table 1. The RBF takes the least time to operate while the SVM classifier takes the longest. The advantages and disadvantages of each method were weighed to ultimately select the RBF as the classifier for this study.

Detection via RBF network.—The RBF network is composed of an input layer, a middle layer, and an output layer. The number of nodes in the input layer is equal to the dimension of the input data, and the number of nodes in the output layer is equal to the dimension of the output data. The number of nodes in the hidden layer is determined by the complexity of the problem. The input layer does not process the input data but rather passes it directly to the hidden layer. The basis function of the hidden layer is formed by a function similar to a Gaussian kernel function, which generates a partial response to the input signal. The output layer combines the outputs of the hidden layer linearly. The R, G, and B channel pixel values were taken here as the input characteristics while the output data were the pixel points of the intact area, defect area, and background area. The Gaussian function was selected as the hidden diameter basis function of hidden layer as follows:

$$R_i(x) = \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right) \quad i = 1, 2, \dots, m \quad (4)$$

where x is the input vector, c_i is the center of the basis function, σ_i is the perception variable, and m is the number of perception units.

In the RBF network, input vectors in the input layer are mapped to the hidden layer in a nonlinear mode. Vectors in the hidden layer are mapped to the output layer in a linear

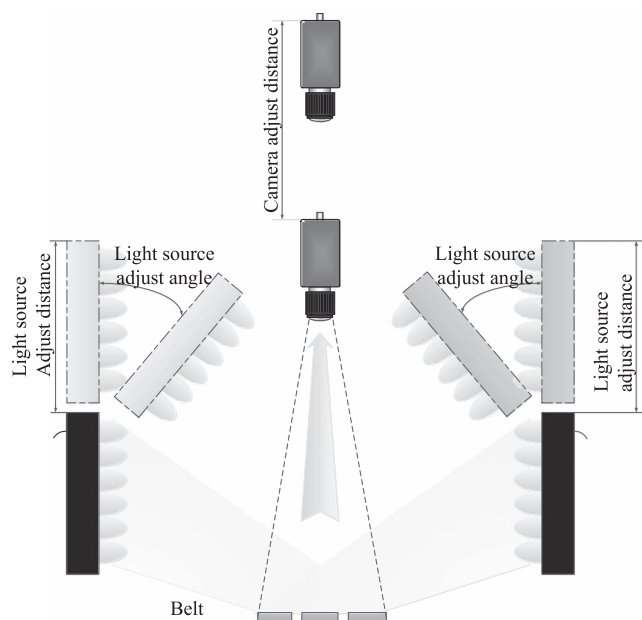


Figure 8.—The structure of the lighting system.

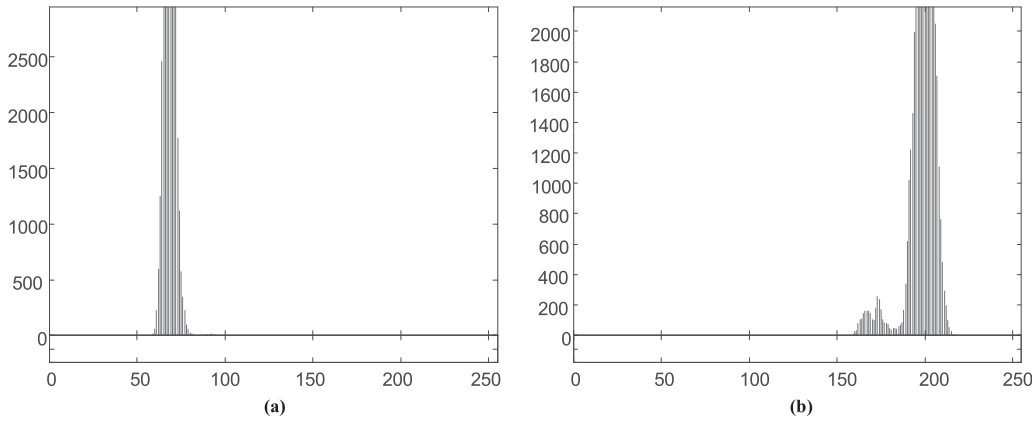


Figure 9.—Grey histogram. (a) Belt background pixels, and (b) bamboo part pixels.

mode. The linear mapping formula is as follows:

$$y_k = \sum_{i=1}^m w_{ik} R_i(x) \quad k = 1, 2, \dots, p \quad (5)$$

where p is the number of output nodes and w_{ik} the weight between hidden layer node i and output layer node k . A schematic diagram of the RBF network is shown in Figure 12.

Appropriate center function and weight parameters are necessary to secure effective training results. Here, they were optimized according to the feedback mechanism. The gradient descent method was used to optimize the network. A series of 200 images of bamboo parts were taken as training samples and another 100 images as test samples. The diffusion factor was set to 22 and the error tolerance to 0.001. The training results indicated that the mean square error decreased as the number of neurons in the hidden layer increased (Fig. 13). Table 2 lists the mean square error with neuron numbers of 40, 41, 42, and 43. Table 3 lists the expected output vectors of intact pixels, defect pixels, and background pixels. Four typical sample pixel recognition results are listed in Table 4, and three examples of detection results are shown in Figure 14.

Defect location.—Once the system recognizes that defective parts exist, they must be located properly so as to calculate the time required for the parts to reach the air jet and be removed from the conveyor belt. The relationship between the three-dimensional geometric position of a point on the surface of the object and the corresponding point in the image was associated on a geometric model of the camera through calibration.

The imaging principle of camera is shown in Figure 15. There are four coordinate sets: world, image pixel, camera,

and imaging plane coordinates (from right to left; Fig. 15). The world coordinates express the positions of certain objects in reality. The position of a point P is expressed as (X_w, Y_w, Z_w) in this coordinate. The origin of the camera coordinates O_i is located at the focal length center of the camera; its Z_c axis extends along camera's optical direction while its X_c and Y_c axes are parallel to the virtual imaging coordinates. There are two kinds of image coordinates: imaging plane and image pixel coordinates.

As the camera shoots objects in space, the object is converted from the world to the camera coordinates, from the camera to the imaging plane coordinates, and then from the imaging plane to the image pixel coordinates. The camera lens usually presents radial distortion over the course of this process. The world coordinates can be converted to the camera coordinates by rotating and shifting the matrix while rotating θ around the Z axis:

$$x = x' \times \cos\theta - y' \sin\theta \quad (6)$$

$$y = x' \times \sin\theta + y' \cos\theta \quad (7)$$

$$z = z' \quad (8)$$

Equations (6), (7), and (8) can also be expressed in matrix form as follows:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_1 \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} \quad (9)$$

The rotations of α and β angles around X and Y can be expressed similarly as

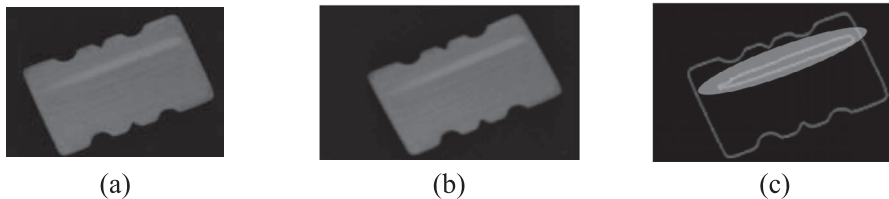


Figure 10.—(a) Original image, (b) median filter, and (c) defect area.

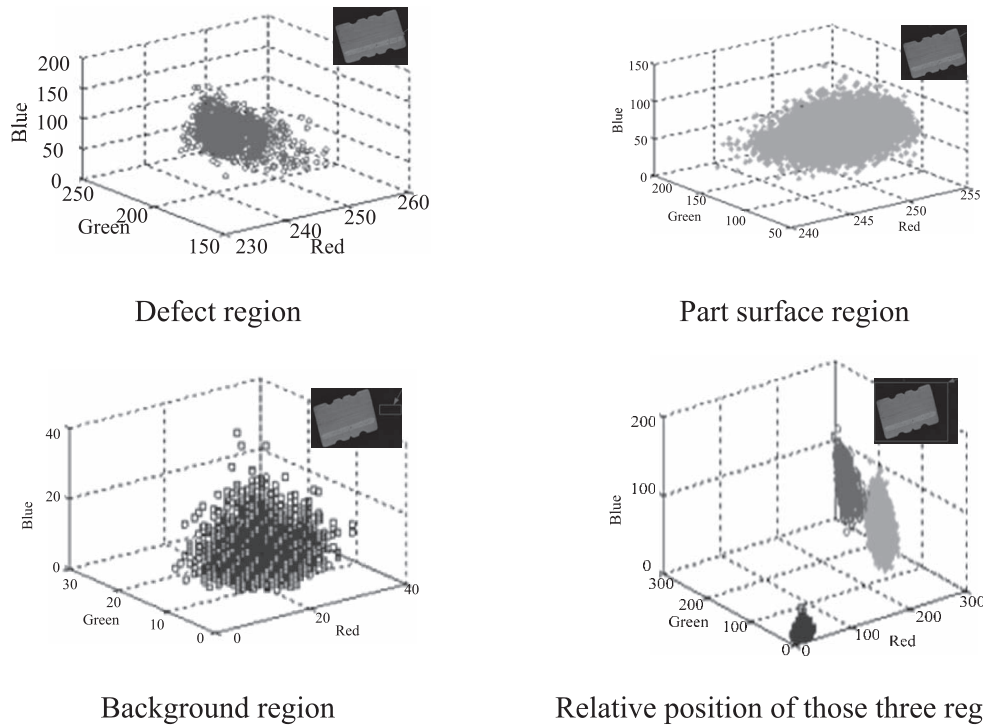


Figure 11.—Sample image and its spatial distribution of pixels in different regions in the red–green–blue (RGB) co-ordinates.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R_1 \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad (10)$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R_1 \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \quad (11)$$

The conversion from world coordinates to camera coordinates for points P_w can be realized as follows:

$$P_c = RP_w + T \quad (12)$$

where T is a translation vector, (t_x, t_y, t_z) ; R is a rotation matrix that can be expressed as

$$R = R_1 \times R_2 \times R_3 \quad (13)$$

In an ideal state, the transformation from camera coordinates to imaging plane coordinates is expressed as follows:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{f}{Z_c} \begin{pmatrix} x_c \\ y_c \end{pmatrix} \quad (14)$$

where $\begin{pmatrix} u \\ v \end{pmatrix}$ is the position of point P in the imaging plane coordinates.

However, the ideal image differs from the actual image as a result of distortion of the optical lens. The camera distortion must be corrected using radial distortion coefficient k . The ideal image plane coordinates are converted to the real image plane coordinates with k as follows:

$$\begin{pmatrix} \tilde{u} \\ \tilde{v} \end{pmatrix} = \frac{2}{1 + \sqrt{1 - 4k(u^2 + v^2)}} \begin{pmatrix} u \\ v \end{pmatrix} \quad (15)$$

In the last transformation step, imaging plane coordinates are converted into image coordinates by

$$\begin{pmatrix} r \\ c \end{pmatrix} = \begin{pmatrix} \frac{\tilde{v}}{S_y} + C_y \\ \frac{\tilde{u}}{S_x} + C_x \end{pmatrix} \quad (16)$$

where $\begin{pmatrix} r \\ c \end{pmatrix}$ is the position of point P in the image pixel coordinate, C_x and C_y are the vertical projection of coordinate centers on the imaging plane, and S_x and S_y are

Table 1.—Schedule of various methods (unit: millisecond). Support vector machine (SVM), k -nearest neighbor (KNN), multilayer perceptron (MLP), and radial basis function (RBF) neural networks.

Sorting method	Image size	Training time	Recognition time	Total time
SVM	1,280 × 1,024	105,705	4,443	110,148
KNN	1,280 × 1,024	9.7	545.5	555.2
MLP	1,280 × 1,024	192.2	32.4	224.6
RBF	1,280 × 1,024	205.3	15.6	220.9

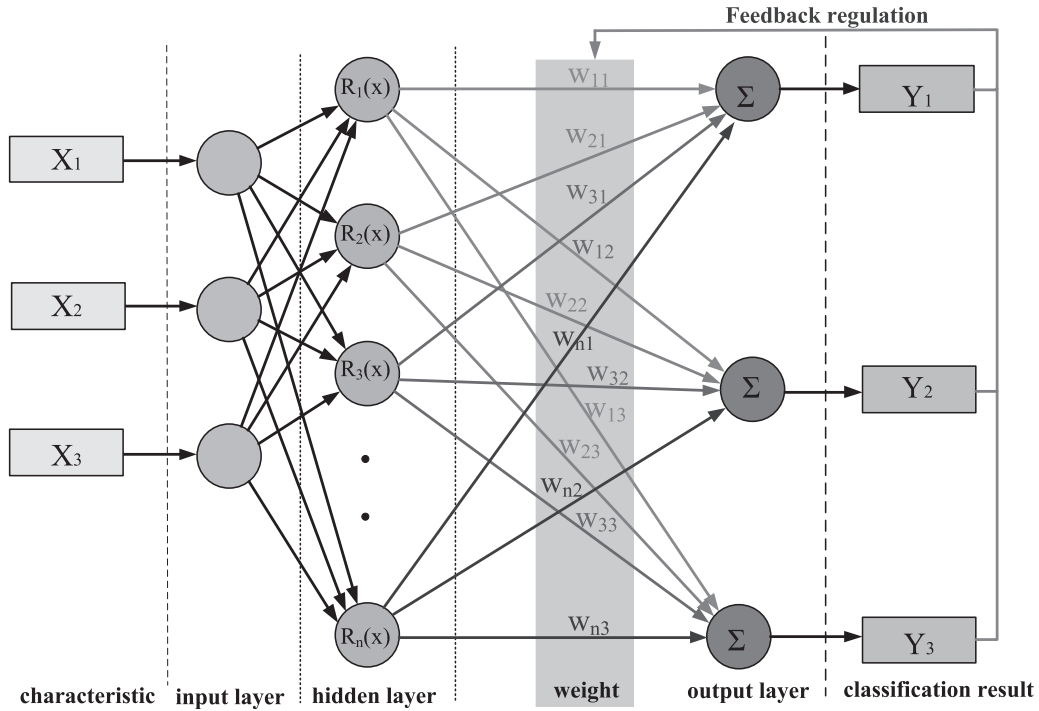


Figure 12.—Schematic diagram of radial basis neural network.

the distance between adjacent pixels in the horizontal and vertical directions of the image sensor, respectively.

Calibration serves to minimize the distance between the central coordinate points $m_{i,j}$ obtained by extracting the edge contour and the calculated coordinate point $T_i(M_i, c)$ by projection, which can be expressed as follows:

$$\text{mind}(c) = \sum_{j=1}^l \sum_{i=1}^k \|m_{i,j} - T_i(M_i, c)\|^2 \quad (17)$$

The calibration is obtained by $f, k, S_x, S_y, C_x, C_y, t_x, t_y, t_z, \alpha, \beta, \gamma$, and other parameters. A calibration card was used to calibrate the camera for a size of $60 \text{ mm} \times 60 \text{ mm}$, which is about one-third of the field-of-view. The card was placed in different positions rotating around the X-axis and Y-axis appropriately within the view field of the camera, and then

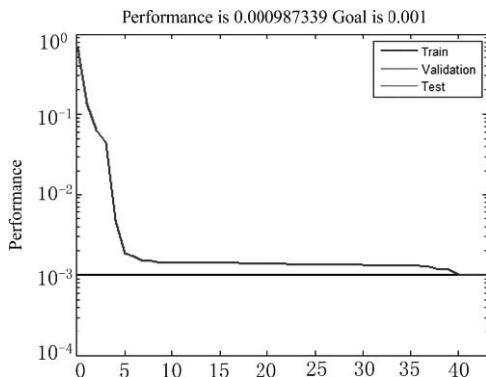


Figure 13.—Curve of the mean square error.

20 photos of different poses were taken to obtain an accurate distortion coefficient.

Method evaluation

In order to evaluate the performance of the proposed method in detecting the defect of bamboo part, the area of defect was selected as the evaluation index. The data set was divided into training set (two-thirds of the total data) and test set (one-third of the total data). The coefficient of determination (R^2), root mean square error (RMSEP) and relative root mean square error (rRMSE) as shown in Equations (18) ~ (19) were used to predict the accuracy.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y'_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (18)$$

Where y_i is the measured area of defect, y'_i is the detected area of defect, \bar{y}_i is the average of measured area of defect, and n is the number of samples.

$$\text{RMSEP} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2} \quad (19)$$

Table 2.—Partial mean square error (MSE).

Neurons	MSE
40	0.00102795
41	0.00100302
42	0.00100302
43	0.000987339

Table 3.—Expected output.

Pixel type	Expected output
Intact pixels	(1, 0, 0)
Defect pixels	(0, 1, 0)
Background pixels	(0, 0, 1)

$$rRMSE = \frac{RMSEP}{\bar{y}_i} \times 100\% \quad (20)$$

10-fold cross-validation divides the data set into 10, and takes turns using 9 of the 10 data sets as training data and 1 as test data, and then finds the average as the estimate of accuracy; this was used to evaluate the stability and accuracy of the method. Accuracy rate, precision rate, and recall rate, which are three commonly used performance metrics in evaluating a binary classification problem, were selected as indicators for evaluating the quality of the proposed method in classifying rectangle bamboo parts. The larger their value, the better the classification effect of the model.

The precision rate (*PR*) represents the proportion of the true positive samples in the samples predicted to be positive, which is expressed in Equation (21).

$$PR = \frac{TP}{TP + FP} \quad (21)$$

where *TP* represents positives that are correctly identified, *FP* is negatives that are wrongly identified.

The recall rate (*RR*) indicates how many positive samples are predicted correctly, which is calculated by Equation (22).

$$RR = \frac{FP}{TN + FP} \quad (22)$$

where *TN* is negatives that are correctly identified.

The accuracy rate (*AR*) is the ratio of the number of samples predicted correctly to the total number of samples, which is obtained by Equation (23).

$$AR = \frac{TP + TN}{TP + TN + FN + FP} \quad (23)$$

where *FN* is positives that are wrongly identified.

Software Development

The main interface of the software consists of an image display area, log display area, parameter setting area, jet test area, and status display area (Fig. 16). The jet test area was used to debug the opening and closing of the air jets. The parameter setting area was given parameters for each serial

Table 4.—Four typical sample pixel recognition results.

Test sample	Actual output	Expected output
1	(1.1320, 0.1121, -0.0013)	(1, 0, 0)
2	(0.1701, 0.1223, 0.9724)	(0, 0, 1)
3	(-0.1423, 1.015, 0.1251)	(0, 1, 0)
4	(0.0237, -0.1921, 1.0191)	(0, 0, 1)

Table 5.—Statistical index values for modeling bamboo parts.

Items	Value
R^2	0.64
RMSEP	0.042
rRMSE	5.54%

port to ensure accurate communication among them. The image display area is composed of a real-time image display and detection results display, wherein each defect region is marked in white color. The log area was used to record the instructions sent to each air jet and the sending time. The status display area covers the resolution of the image acquired by the camera, the total frame number of the image displayed in the window, and the total frame number of the image acquired by the camera.

Results and Discussion

A manual sort was compared as a basis with this RBF neural network-based automatic method. The method can predict the area of defect effectively ($R^2 = 0.64$, RMSEP = 0.042, and rRMSE = 5.54%; Table 5).

A total of 3,000 bamboo parts were used for the final sorting testing; 1,500 parts were intact and the other 1,500 were defective. The test samples were divided into 100 groups and placed randomly onto the conveyor belt, which moves forward at a speed of 2 m/second. The test results are summarized in Table 6. There were 1,360 pieces of defective parts correctly sorted out and 140 pieces missed. The correct identification rate and missed rate are 90.6 percent and 9.4 percent, respectively. This correct identification rate is greater than that of manual sorting. For those 1,500 intact parts, 110 were false-positively sorted out at a rate of 7.3 percent, which is slightly lower than the same

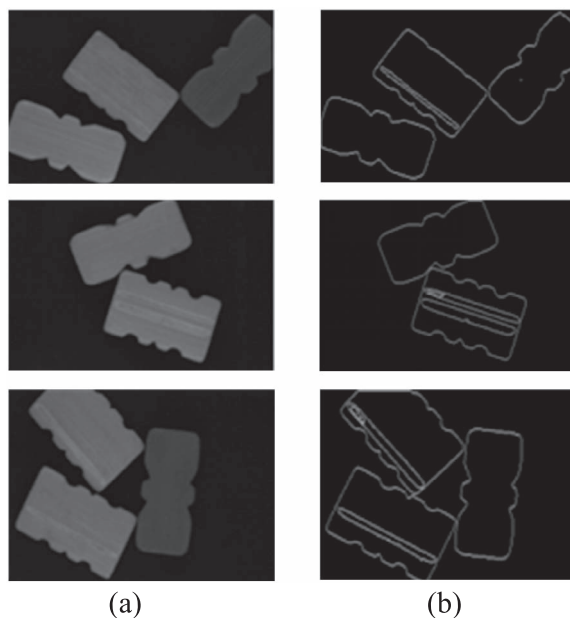


Figure 14.—Three detection examples. (a) Original images and (b) detection results.

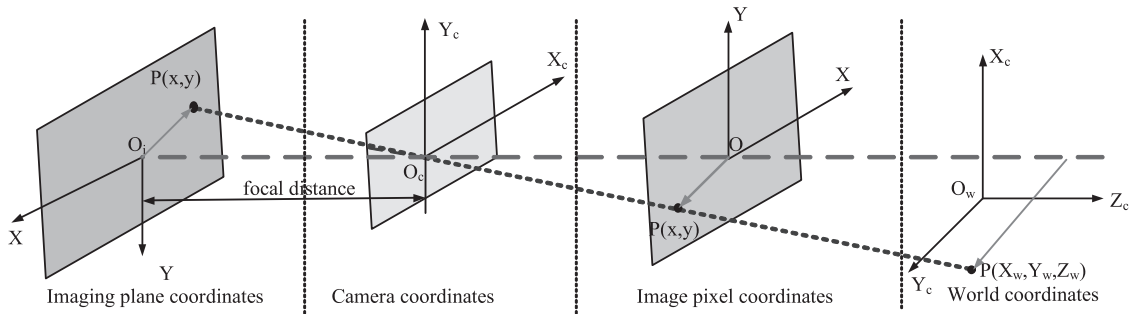


Figure 15.—Schematic diagram of world coordinate, image pixel coordinate, camera coordinate, and imaging plane coordinate.

rate by manual sorting. The statistical results of automatic sorting and manual sorting, which includes average, maximum, minimum, standard deviations, and coefficient of variation (CV) are summarized in Table 7. Here, the standard deviations indicated that volatilities of the identified defective parts and intact parts by automotive sorting and manual sorting among different groups were small. It is also indicated that volatilities of the identified defective parts by automotive sorting is lower than by manual sorting, but that of identified intact parts by automotive sorting is greater than by manual sorting. The missed sorting can mainly be attributed to color similarity between intact and defective regions in some samples. The response speed of air jet is relatively low because of the characteristics of compressed air, so if defective and intact parts happened to be folded together, the intact component was blown away with the defective one resulting in a false positive. The sorting time of the 3,000 samples was about 10

minutes. By comparison, manual sorting of the same quantity of parts takes about 48 minutes.

The results of 10-fold cross-validation experiment listed in Table 8 indicated that the average accuracy rate, recall rate, and precision rate of this method were 91.1 percent, 92.5 percent, and 91.7 percent, respectively. And their standard deviations were 0.0055, 0.0182, and 0.0100, respectively. Those test result indicated that this sorting method is robust.

The traditional threshold segmentation method did not reveal the defective bamboo parts accurately. However, the intact, defective, and background areas were clustered effectively in the RGB space of images. The RGB-space identification is also simpler and more direct than other methods. An RBF is composed of only three layers with simple structure and fast convergence; its training and identification speeds are faster than those of SVM, KNN, or MLP. The RBF also optimizes the weights of the network

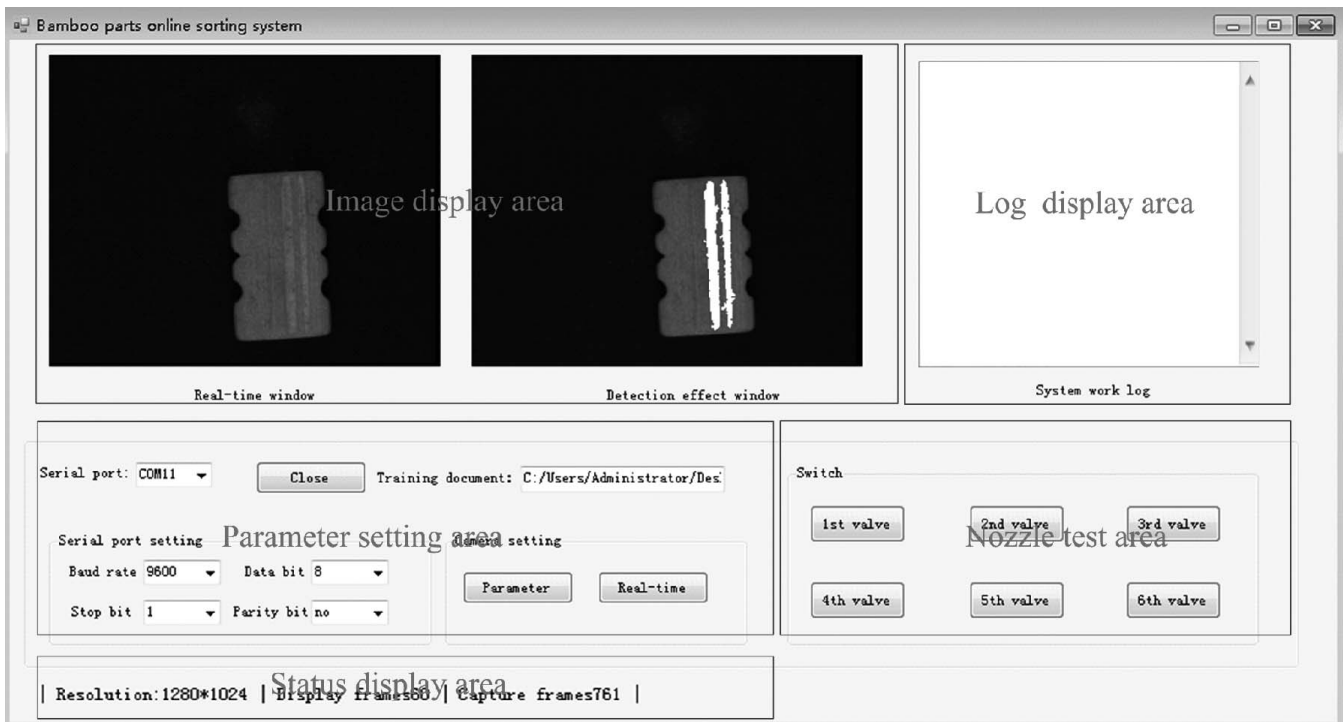


Figure 16.—Main interface of the software.

Table 6.—Identification results of bamboo parts.

Types of parts	Total	Automatic sorting		Manual sorting	
		Correct sorting	Missing	Correct sorting	Missing
Defective	1,500	1,360	140	1,346	154
Percentage	100.0	90.6	9.4	89.7	10.3
Intact	1,500	1,390	110	1,448	52
Percentage	100.0	92.6	7.3	96.5	3.5

via feedback mechanism, so its training effect is continuously improved.

The results of this test show that the RBF effectively reveals defects in bamboo parts, but the proposed method does merit further improvement. For example, the luminance of some parts is uneven under low-angle lighting; a more suitable lighting design and light sources are necessary to ensure every bamboo part within the camera’s field-of-view receives even lighting. The weight adjustment of the RBF also depends on the batch gradient descent method, which is increasingly time-consuming to update as the sample size increases. The small-batch gradient descent method (mini-batch GD) and random gradient descent method (SGD) may be suitable to optimize the network. Other neural networks, such as random neural networks and self-organizing neural networks or combinations thereof, may be a valuable future research direction.

Conclusion

A bamboo part sorting method was established in this study and supported by corresponding hardware and software platforms. A series of comparisons showed that the RBF trains and identifies samples faster than SVM, KNN, or MLP methods. An RBF neural network with 43 neurons was used to classify defective, intact, and background areas from images of bamboo parts. The correct identification rate (90.6%) was found to be greater than that of manual sorting (89.7%) while the speed is up to four times greater; however, the false-positive rate (7.3%) is also greater than that of manual sorting (3.5%) and thus merits further improvement.

Although the proposed method is much more correct and much faster than manual sorting, it is a pre-study and does merit further improvement. A more suitable lighting design and light sources are necessary to ensure every bamboo part within the camera’s field-of-view receives even lighting. Small-batch gradient descent method (mini-batch GD) and random gradient descent method (SGD) can be considered to optimize the network.

Table 7.—Statistical summary of identification results.

Items		Automatic sorting					Manual sorting				
		\bar{x}	Max.	Min.	SD	CV	\bar{x}	Max.	Min.	SD	CV
Defective parts	Number	13.6	15	12	0.35	0.026	13.46	15	11	0.47	0.035
	Percentage	90.6	100	86.7	2.34	0.026	89.7	100	73.3	3.15	0.035
Intact parts	Number	13.9	15	13	0.28	0.020	14.48	15	13	0.21	0.015
	Percentage	92.6	100	80	1.87	0.020	96.5	100	86.7	1.40	0.015

Table 8.—10-fold cross validation results.

K-fold	Precision rate	Recall rate	Accuracy rate
1-fold	0.911	0.927	0.918
2-fold	0.901	0.927	0.922
3-fold	0.910	0.928	0.916
4-fold	0.915	0.936	0.912
5-fold	0.906	0.924	0.925
6-fold	0.905	0.932	0.912
7-fold	0.912	0.922	0.911
8-fold	0.917	0.925	0.920
9-fold	0.901	0.917	0.919
10-fold	0.902	0.912	0.915
Mean	0.911	0.925	0.917
SD	0.0055	0.0182	0.0100

Acknowledgments

This research is co-supported by National Key Research and Development Program of China [Project approval No.2018YFD0101001], and by Guangdong Science and Technology Plan Projects with Research Grant 2017A010102024 and 2017A020208052.

Literature Cited

- Arakeri, M. P. and Lakshmana. 2016. Computer vision based fruit grading system for quality evaluation of tomato in agriculture industry. *Procedia Comput. Sci.* 79:426–433.
- Baigvand, M., A. Banakar, S. Minaei, J. Khodaei, and N. Behrooz-Khazaei. 2015. Machine vision system for grading of dried figs. *Comput. Electron. Agric.* 119:158–165.
- Baltazar, A., J. I. Aranda, and G. González-Aguilar. 2008. Bayesian classification of ripening stages of tomato fruit using acoustic impact and colorimeter sensor data. *Comput. Electron. Agric.* 60(2):113–121.
- Chen, S. M., J. T. Xiong, W. T. Guo, R. B. Bu, Z. H. Zheng, Y. Q. Chen, Z. G. Yang, and R. Lin. 2019. Colored rice quality inspection system using machine vision. *J. Cereal Sci.* 88:87–95.
- Choi, D., W. S. Lee, R. Ehsani, and F. M. Roka. 2015. A machine vision system for quantification of citrus fruit dropped on the ground under the canopy. *Trans. ASABE* 58(4):933–946.
- Gu, J. N., L. Y. Tang, Y. Xu, and S. X. Tang. 2019. Application of machine vision in detection and classification of mesh defects. *Machin. Design Manuf.* s1:47–49, 53.
- Hannan, M., T. F. Burks, and D. M. Bulanon. 2009. A machine vision algorithm for orange fruit detection. *CIGR Ej.* XI(December): Manuscript 1281.
- Jabo, S. 2011. Machine vision for wood defect detection and classification. Master of Science thesis. Chalmers University of Technology, Sweden.
- Li, H., W. S. Lee, and K. Wang. 2016. Immature green citrus fruit detection and counting based on fast normalized cross correlation (FNCC) using natural outdoor color images. *Precis. Agric.* 17(6):678–697.
- Liu, T. H., R. Ehsani, A. Toudeshki, X. J. Zou, and H. J. Wang. 2019. Identifying immature and mature pomelo fruits in trees by elliptical model fitting in the Cr–Cb color space. *Precis. Agric.* 20(1):138–156.

- Moallem, P., A. Serajoddin, and H. Pourghassem. 2017. Computer vision-based apple grading for golden delicious apples based on surface features. *Inf. Process. Agric.* 4(1):33–40.
- Ohali, Y. A. 2011. Computer vision based date fruit grading system: Design and implementation. *J. King Saud Univ.-Comput. Inf. Sci.* 23(1):29–36.
- Shin, J. S. W. S. Lee, and R. Ehsani. 2012a. Postharvest citrus mass and size estimation using logistic classification model and watershed algorithm. *Biosyst. Eng.* 113(1):42–53.
- Shin, J. S., W. S. Lee, and R. J. Ehsani. 2012b. Machine vision based citrus mass estimation during post harvesting using supervised machine learning algorithms. *Acta Hortic.* 965:209–216.
- Sofu, M. M., O. Er, M. C. Kayacan, and B. Cetislib. 2016. Design of an automatic apple sorting system using machine vision. *Comput. Electron. Agric.* 127(C):395–405.
- Subramanian, V., T. F. Burks, A. A. Arroyo. 2006. Machine vision and laser radar-based vehicle guidance systems for citrus grove navigation. *Comput. Electron. Agric.* 53: 130–143.
- Xu, L. M. and Y. C. Zhao. 2010. Automated strawberry grading system based on image processing. *Comput. Electron. Agric.* 71(1):32–39.
- Yu, Y., Z. Z. Chen, Z. Y. Chen, and W. J. Shen. 2009. Application of machine vision in capacitor appearance defect detection. *Comput. Technol. Appl.* 45(9):97–100, 105.
- Zhang, Y. D., S. H. Wang, G. L. Ji, and P. Phillips. 2014. Fruit classification using computer vision and feedforward neural network. *J. Food Eng.* 143:167–177.