

## Research Article

# Automatic Defect Detection in Spring Clamp Production via Machine Vision

Xia Zhu,<sup>1,2</sup> Renwen Chen,<sup>1</sup> and Yulin Zhang<sup>2</sup>

<sup>1</sup> State Key Laboratory of Mechanics and Control of Mechanical Structures, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

<sup>2</sup> Faculty of Electronic and Electrical Engineering, Huaiyin Institute of Technology, Huaian 223003, China

Correspondence should be addressed to Renwen Chen; [rwchen@nuaa.edu.cn](mailto:rwchen@nuaa.edu.cn)

Received 25 April 2014; Accepted 20 May 2014; Published 9 July 2014

Academic Editor: Shen Yin

Copyright © 2014 Xia Zhu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There is an increasing demand for automatic online detection system and computer vision plays a prominent role in this growing field. In this paper, the automatic real-time detection system of the clamps based on machine vision is designed. Its hardware is composed of a specific light source, a laser sensor, an industrial camera, a computer, and a rejecting mechanism. The camera starts to capture an image of the clamp once triggered by the laser sensor. The image is then sent to the computer for defective judgment and location through gigabit Ethernet (GigE), after which the result will be sent to rejecting mechanism through RS485 and the unqualified ones will be removed. Experiments on real-world images demonstrate that the pulse coupled neural network can extract the defect region and judge defect. It can recognize any defect greater than 10 pixels under the speed of 2.8 clamps per second. Segmentations of various clamp images are implemented with the proposed approach and the experimental results demonstrate its reliability and validity.

## 1. Introduction

With the increasing demands on production quality, system performance, and economic requirement, modern industrial processes are more complicated both in structure and automation degrees. The reliability and safety issues on these complicated industrial processes become the most critical aspects for system design and are receiving considerably increasing attention nowadays [1, 2]. The spring clamp detection system in this paper talks about also facing the problems mentioned above.

A spring clamp is rounded by pressing spring steel, with two ears left around the circle. When to be clamped, tight ears need to hold down firmly, making larger set into the inner tube, and then you can loosen the grip of the hand, which is very simple to use, and these are the simple steps for using a spring clamp. The adopted material has high flexibility, good physical properties, and strong firmness, which makes the clamp suit the connection of pipe systems of vehicle cooling, heating, and ventilating. The high elasticity of

the clamp compensates the pipe shrinks caused by the change of temperature or the degradation of the pipe. In natural state, the clamp does not have the clamping force. It happens only when exerted to a larger pipe. The clamp can guarantee the reliability of the connection by generating a permanent distortion through the uniform pressure around the circle, which protects the fastness and time within a reasonable range.

The clamp detected in this paper is widely used. It is one of the essential accessory substances applied in fixing the vehicle pipes. Its quality, security, and service life are of utmost importance to the vehicle performance. As a result, the clamp detection is a well-concerned problem of the manufacturers. The detection is still relied on manual detecting at present, which causes problems such as labor intensive, low speed, and misdetection. Some foreign institutes started the relevant research to solve the problems above more than 10 years ago and have now made some progress, such as bottle defect detection, fabrication defect detection, vehicle headlamp lenses defect detection, and web defect detection.

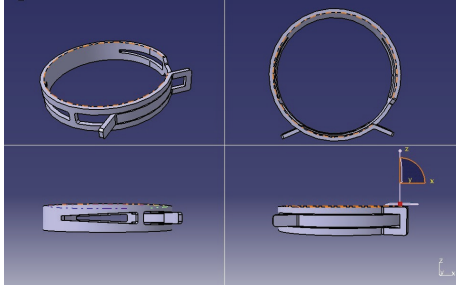


FIGURE 1: Images with various lighting illumination pattern.

Our country began to research the automatic detecting technology at the 1990s, such as the detection of presswork quality and detection of float glass fabrication.

The pipeline system introduced in this paper is in a nonclosed space, which makes the spot light infected by the natural light and the house illumination; in the meantime, serious background noise will be produced. According to the above peculiarities, we designed a spring clamp real-time detection system based on machine vision. Built on both sides of the production line, this system consists of a special light source to weaken the light interference, a camera with high resolution, an external trigger device which will trigger the camera to acquire an image, a host computer which has program to extract an image, recognize it, give a YES/NO judgment, and display the final results on the screen, and a rejecting mechanism which is used to reject nonconforming products. This system could make it possible to automatically detect the spring clamps and greatly improve the rate of defect identification and defect removal, thereby achieving the goals of increasing the quality of the products and avoiding the actual loss.

## 2. Problem Descriptions

According to the detection characteristics of the pipeline, the defect detection system is built on both sides of the production line. Online real-time detection device depends on the external trigger device to trigger the camera to acquire an image, which has no influence on the real production line. There are four parts in this system: optical illumination, image acquisition, image processing, and rejecting mechanism.

For an irregular circle clamp with a smooth curved surface, the image quality will be affected by the illumination mode. Back lighting makes the clamps contour clear, but it also makes it hard to distinguish details. Front parallel lighting will submerge the defects by forming highlights on the other side of the hollow when the light shines through the hollow in the clamp, as shown in Figure 1.

In order to solve the above problems, a new illumination mode, back lighting combined with lateral lighting, is brought out in this paper, of which the details are as follows: (1) front: a white frosted glass is set as the background and a quadrilateral light source is placed parallelly under the glass; (2) lateral: three cameras are placed parallelly above the glass

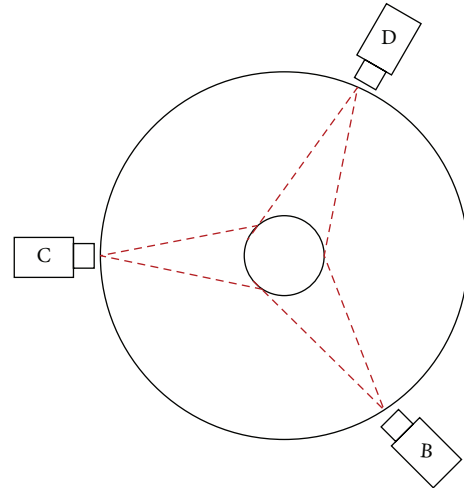


FIGURE 2: The illustrated diagram of the clamps lateral image acquisition.

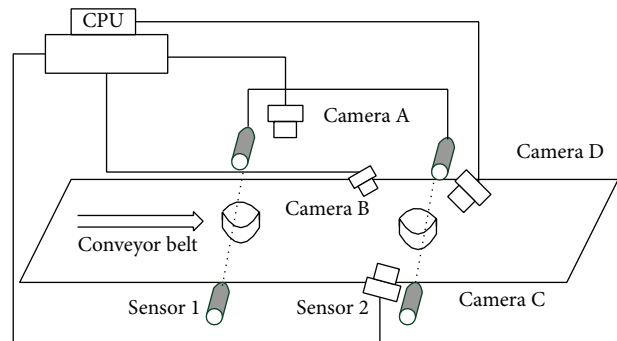


FIGURE 3: The model of the workbench.

with a  $120^\circ$  angle between each other. Each camera obtains an image of  $1/3$  of the clamp after fixing the relative position, as is illuminated in Figure 2. Images from these three cameras constitute the entire outside surface of the clamp, after which the capturing process will be finished.

The detection process is exposed in Figure 3: when the clamp passes through trigger 1, trigger 1 enables the camera A to capture a backlit image of the clamp. The shape and ears will be detected. When the clamp passes through trigger 2, trigger 2 enables the cameras B, C, and D to catch an image of the clamp, respectively. These three images which can constitute the whole external surface of the clamp are then sent to the computer to detect its color, scratch, height, and so on. Once there appears one defect, signal will be sent to the rejecting mechanism through RS485 to pick away the defective one.

## 3. Image Processing Algorithm

Image processing is the core technique in this system. With the development of a corresponding technique, a lot of methods are presented, such as neural network (NN) and wavelet, but the applicability of each algorithm is narrow and

just effective to their special scenes. The detection algorithm is the kernel of this system. The process is as follows: read images from those four cameras, respectively, and preprocess the images; detect the high, color, shape, crack, and so on; locate the defects, assess the defects, and display the results in the computer and pick away those with defects by rejecting mechanism.

**3.1. Image Preprocessing.** During the production of the spring clamps, there exists image noise in the original image captured in the limited environmental conditions; as a result, the original images should be preprocessed. The algorithm is utilized to reduce noise and delete the unwanted regions.

**3.1.1. Region of Interest (ROI).** The field of view of the captured pictures is definite because the relative position of the camera and the clamp is fixed in mechanical design of the system.

In order to increase the detection rate and real-time property of the system, it is necessary to locate ROI before image segmentation. We could judge the image based on abrupt changes in grayscale, which is able to determine the ROI effectively and quickly.

Summing the gray values of each column of an image in spatial domain (which is shown in Figure 4), we obtain  $C_j$ :

$$C_j = \sum_{i=0}^{N-1} f(x_i, y_j), \quad j = 0, 1, \dots, M - 1, \quad (1)$$

where  $f(x_i, y_j)$  is the gray value of the pixel in the coordinate position  $(x_i, y_j)$ . In a similar way, summing the gray values of each row  $R_i$  is obtained:

$$C_i = \sum_{j=0}^{M-1} f(x_i, y_j), \quad i = 0, 1, \dots, N - 1. \quad (2)$$

As we can see, abrupt changes in grayscale will cause correspondingly changes of  $C_j$  and  $R_i$ .

Then subtracting two adjacent values of  $C_j$ , we obtain

$$\Delta_j = C_j - C_{j-1}, \quad j = 1, 2, \dots, M - 1. \quad (3)$$

Considering the characteristics of the clamp, we set a maximum tolerance  $\Delta_{\max} = 10$  pixel. Set up an array flag  $[M - 1]$ :

$$\text{flag}[j] = \begin{cases} j, & \text{if } |\Delta_j| > \Delta_{\max}, \\ 0, & \text{else,} \end{cases} \quad (4)$$

$$j = 1, \dots, M - 1.$$

In Figure 5,  $X5$  is obtained, the first value that is not zero in the series flag  $[M - 1]$ , and  $X3$  is gotten, the last value that is not zero by applying the above algorithm, which could determine the up and down edges in Figure 5. In a similar way, we can also obtain the left and right edges:  $Y = Y2$  and  $Y = Y4$ . Thus, the upper-left corner  $(X5, Y2)$  and lower-right corner  $(X3, Y4)$  will be generated. Finally the ROI area will be fixed, which is shown in Figure 5.

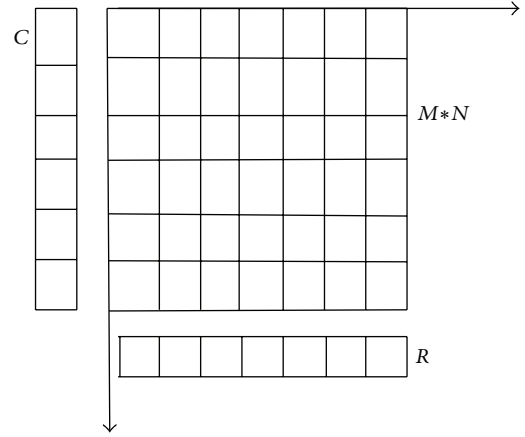


FIGURE 4: A spatial domain of the image.

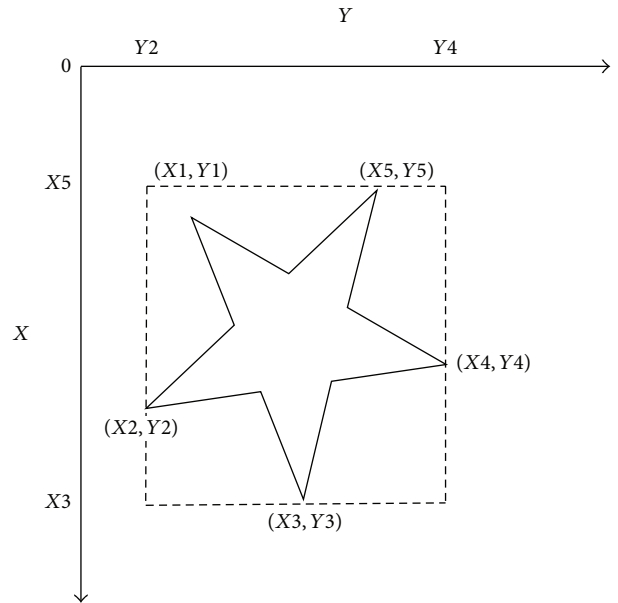


FIGURE 5: The structure display of an image in spatial domain.

**3.1.2. Median Filtering Based on Weight Value.** The complicated environment of the factory causes noise in our captured images. Meanwhile, in industrial processes uneven tiny scratches and spots can be also caused. The image filtering will solve the above problems effectively. In order to eliminate noise as well as protect the details of an image, median filter is implemented to change the values of those pixels which have biggish gray difference with the surrounding pixels and then eliminate isolated noise.

In order to conquer the drawbacks of simple local median filter, different weight values are given to the pixels involved in the operation, which is weighted median filter. Normally the weight values are determined according to the following principles: (1) assign biggish weight value to the pixel to be processed and smaller values to the rest; (2) determine the weight value of the rest of the pixels according to the distance to the pixel to be processed and the closer, the bigger;

(3) determine the weight value of the rest pixels according to the degree of closeness between the pixel to be taken up and the one of the rest.

**3.2. Height Detection.** The height of clamp can be calculated from three lateral images. The detection method is as follows. Firstly, detect the edges with ROI and obtain the straight edge lines  $L1$  and  $L2$  by using straight line fitting method at edge points. Secondly, draw a perpendicular bisector ( $H$ ) in ROI and calculate the points of intersection between  $H$  and  $L1$  and  $H$  and  $L2$  ( $(X1, Y)$ ,  $(X2, Y)$ ). Finally, the height can be calculated by the absolute value of  $X1$  and  $X2$ . The site test pattern is displayed in Figure 6. As we can see, this method can calculate not only the height of the clamp, but also the height of the hollow part in the middle of the clamp, which can provide a basis for detailed height detection.

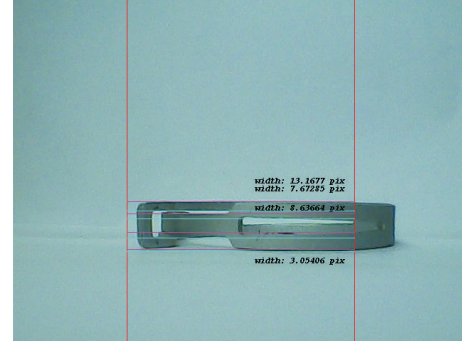
**3.3. Defect Detection.** The first step is always location of defect, of which the theory is the same as ROI location. After locating the defects comes defect segmentation and recognition. At present, there are some segmentation methods, such as threshold method, the maximum entropy method, and histogram threshold method [3–5]. Due to the inherent characteristics of the product, defects in the images are of small proportion in the whole image. What kind of algorithm to segment defects and clamps greatly is one of the difficulties in this research.

After positioning, we can see that, as in the right image in Figure 7, some burrs are removed and the defect area is enlarged, which make it easier to improve the processing speed of the software. In order to get the perfect segmentation effect and improve the defects proportion in a histogram of an image to be segmented, segmentation algorithms based on threshold, region segmentation method, morphological segmentation method, and so forth have been tested in the tests. At the end, the PCNN method is adopted, and some improvement has been put forward based on the original. PCNN (pulse coupled neural network) is derived from Eckhorn et al.'s [6] research on nerve cells in the cat visual cortex. In the PCNN model, neurons with similar input generate impulses at the same time, which is able to reduce local gray level difference as well as making up local minor disconnection in an image. PCNN method is unrivalled by other segmentation methods. This property has also been used in fields such as shadow removal, image denoising, and edge extraction.

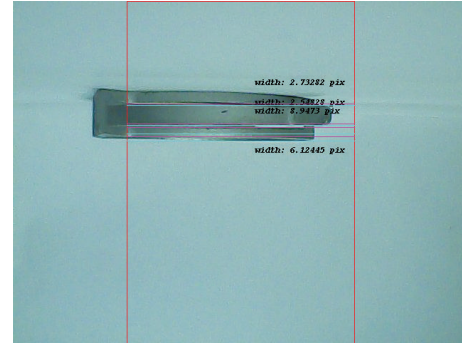
On the basis of Eckhorn's research, Johnson explained PCNN model with circuit theory, which is illustrated in Figure 8. Parameter definition is the same as Johnson and Padgett's [7].

The formula derivation from the circuit in Figure 8 is listed below. In the actual derivation, the same mistakes were found in documents from Johnson and Padgett's [7] and Yide et al's [8]. Different part between my formulas and theirs is roughly listed below. For detailed derivation of the formulas, see the Appendix

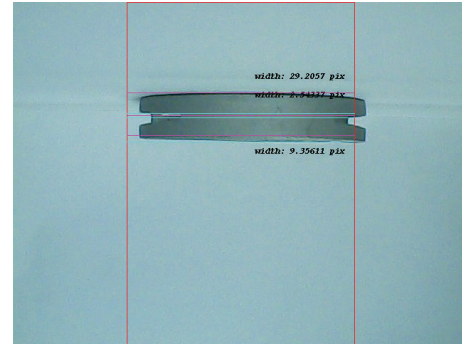
$$\dot{V} = \tilde{M}V + R, \quad (5)$$



(a)



(b)



(c)

FIGURE 6: The height calculation diagrams of the three lateral images.

where

$$V = \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$$

$$\tilde{M} = \begin{pmatrix} -\frac{1}{C_1} (g_{m1} + g_{12} + F) & \frac{g_{12}}{C_1} \\ \frac{g_{12}}{C_1} & -\frac{1}{C_2} (g_{m1} + g_{12} + L) \end{pmatrix} \quad (6)$$

$$R = \begin{pmatrix} \frac{1}{C_1} (g_{m1} E_m + F E_s - \Delta V Y(t)) \\ \frac{1}{C_2} (g_{m2} E_m + E_s L) \end{pmatrix},$$

where  $V_1$ ,  $V_2$  are the electrical potentials of their neuron membranes. Equation (5) indicates that pulse signal is not





FIGURE 7: Schematic defect location.

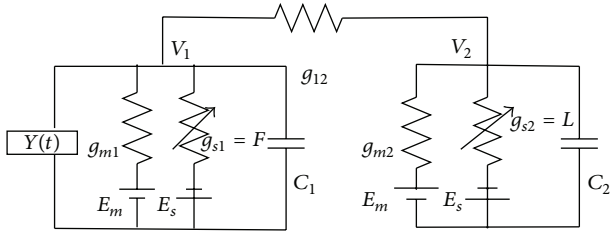


FIGURE 8: The equivalent circuit of neuron.

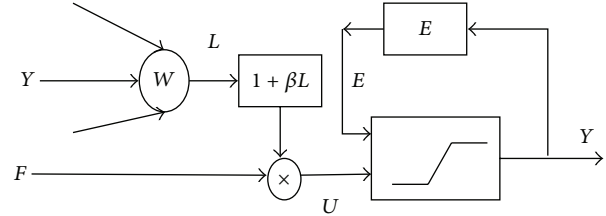


FIGURE 9: The model of PCNN neuron.

only related to the transfer conductance between synapses  $g_{s1}$ ,  $g_{s2}$ , but also related to the equivalent leakage conductance, the equivalent capacitance, and the intrinsic capacitance, which indicates that whether a neuron can be fired depends on the exterior input of its neighboring neuron, as much as its own internal activity.

Pulse function  $Y(t)$  will be generated when the neuron potential  $V_1$  goes larger than the threshold potential; and there are no signals to be generated when the inner activities achieve balance; that is to say,  $V$  is zero.

Then the following equation can be derived from (5):

$$V_1 = \frac{a + bF + cL + dFL}{e + fF + gL + hFL}, \quad (7)$$

where

$$\begin{aligned} a &= g_{12}g_{m2}E_m + g_{12}g_{m1}E_m + g_{m1}g_{m2}E_m, \\ b &= g_{12}E_s + g_{m2}E_s, \\ c &= g_{12}E_s + g_{m1}E_m, \\ d &= E_s, \\ e &= g_{12}g_{m2} + g_{12}g_{m1} + g_{m1}g_{m2}, \\ f &= g_{12} + g_{m2}, \\ g &= g_{12} + g_{m1}, \\ h &= 1. \end{aligned} \quad (8)$$

Equation (7) is actually the functional relation of the neuron membrane potentials in equilibrium. It indicates that the nonlinear multiplying modulating coupling property of PCNN is caused by the conductance between the axon of a neuron (presynaptic) and the axon of a neighboring neuron (postsynaptic). While neurons that input conductance are

controlled by the pulse voltage, the characteristics of neuronal synapses with this character are transferred to the adjacent conductance neurons.

Each neuron includes 3 parts: dendrite, nonlinear connection modulation, and pulse generation element. The model of a PCNN neuron is illustrated in Figure 9 [7]. The dendrite is utilized to receive information from the neighboring neurons through the linear connection input channel and feedback input channel. Nonlinear connection modulation, namely, the internal activities of neurons ( $U$ ), is obtained by multiplying the linear part which has an offset and the feedback input part. The generation of the pulse depends on whether the inner activities can stimulate the dynamic threshold, and the threshold value ( $E$ ) will be modified with the output state of the neuron. When  $E$  is lower than  $U$ , the neuron will be stimulated ( $Y = 1$ ), which is called fired. Next  $E$  suddenly increases because of the feedback of the output, and thus the neuron will be suppressed immediately ( $Y = 0$ ). The output will yield a pulse signal which connects to the input of the neighboring neuron with weight coefficient, so as to influence the stimulus state of these neurons.

The model can be described with discrete functions as follows:

$$F_i = \sum_j (M_{ki}Y_k(t) \otimes \Phi_{ki}(t)) + I_i, \quad (9)$$

$$L_i = \sum_k (M_{ki}Y_k(t) \otimes \Phi_{ki}(t)) + J_i, \quad (10)$$

$$U_i = F_i(1 + \beta L_i), \quad (11)$$

$$Y_i(t) = \text{step}(U_i - \Theta_i), \quad (12)$$

$$\Theta_i = -\alpha_T \Theta_i + V_T Y_i(t), \quad (13)$$

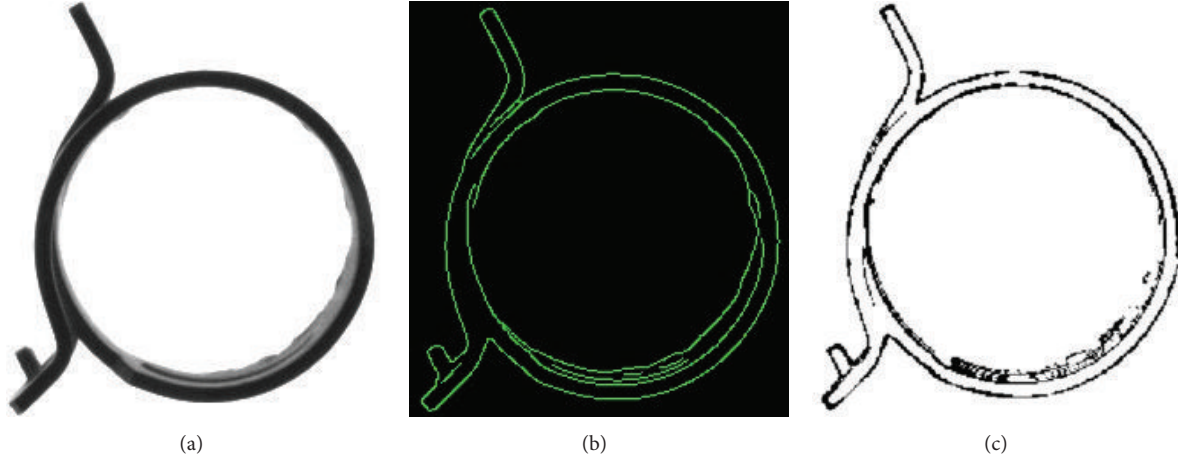


FIGURE 10: The original image (a), the output image after threshold segmentation (b), and the output image after PCNN segmentation (c).

where  $\otimes$  is convolution,  $\Theta_{ij}(t) = A_{ij}e^{-\alpha_{ij}t}$  is the temporal responses kernel of the  $ij$ th neuron,  $M_{ij}$  and  $W_{ij}$  are the weight coefficient matrices of the linear connection input channel and the feedback input channel,  $I_i$  and  $J_i$  are the constant inputs,  $F_i$  is the feeding input,  $L_i$  is the linking input,  $U_i$  is the total internal activity,  $Y_i$  is the output pulse,  $\Theta_i$  is the neuron threshold,  $i, j, k$  represent different neurons,  $A_{ij}$  and  $\alpha_{ij}$  are the dendrite state parameters,  $\alpha_T$  and  $V_T$  are the time constant and magnification coefficient of threshold value,  $V_T$  is usually set to a larger value which is larger than  $U_i$ , and  $\beta_i$  is the linking strength coefficient. Under this discrete mode, (9)–(13) must be calculated in sequence in the listed order; and  $\Theta_i$  and  $V_T$  use their values as evaluated at the previous time step.

In the light of the inherent characteristics of the clamps, weighted median filter is adopted in image processing in the system. At the same time, PCNN algorithm is used to enhance the filtering effect before segmentation. From (11), the value of  $\beta$  influences the filtering effect. A low-pass filter can be used to constrain internal activities so as to reduce noise. Equation (11) can be slightly modified as follows:

$$U_i = F_i(1 + \beta L_i) - r \left\langle \sum_{\text{cells}} Y_i(t) \right\rangle_{\text{lowpass}}, \quad r \in [0, 1]. \quad (14)$$

By allowing  $r$  changing from 0 to 1, (14) gives a significant decrease in both the temporal and the spatial noise when it is turned on the equilibrium operation of the PCNN. The above idea with PCNN is used to realize the image segmentation on the case in this paper, in which further analysis and modification have also been done.

#### 4. System Implementation

According to the design idea, we build this detection system. In the present machine vision, mostly, CCD camera is the main module to capture images. Java Advanced Imaging (JAI) company concentrates on providing customs with abundant products from line scanning to area scanning, from analog to digital technology, and from Camera Link to gigabit Ethernet

(Gig E) vision. This research applied a digital camera with the type JAI CM-030GE/CB-030GE, which adopts a Sony IC424 CCD sensor with a size of  $1/3''$ , effective pixels  $656 * 494$ , the max frame rate 90fps under the max resolution and continuous acquisition mode, and a Gig E interface to communicate. The load of a Gig E camera is  $656 * 494 * 1 * 90 = 28$  MByte/s, while the maximum load of Gig E link is 120 MByte/s. Therefore, 4 cameras which are applied in this research meet the load requirement.

For the purpose of testing the accuracy of this algorithm, one thousand clamps are detected randomly, among which 100 are unqualified and the other 900 are qualified. Four images of each clamp were captured by the image acquisition system which was triggered by an external trigger and then conveyed to the host computer for defect detection. The process time of each image is 0.087 s, which means the time to detect one clamp is 0.348 s. The results show that there are no misjudgments in one year's debugging.

A two-dimensional image with size of  $N * M$  can be considered as a PCNN network whose number of neuron is  $N * M$ , and each pixel corresponds to a unique neuron input. In the first step, the internal activity is equal to the external stimulus. If the output of the neuron is 1, it is naturally fired. At the same time, the threshold value will increase sharply and decay exponentially with time. It is obvious that if the corresponding pixels of the neighboring neuron and the fired neuron at the last iteration are of the same intensity, the neighboring neuron is easier to be captured and fired. This shows that the natural fire of one neuron can lead to a collective fire to the similar neurons around. Image segmentation could be realized based on the property that the neuron groups formed by the naturally fired neuron and a small area in the image have similar characteristics. The above idea with PCNN is used to realize the image segmentation on the case in this paper, in which further analysis and modification have also been done.

In the right image of Figure 10, it is obvious that even small detailed information like the wrinkle on the inner side of the clamp in the lower right corner of the image and the trail at the overlap region in the center-left of the image

are segmented clearly and completely. Without doubt, the segmentation results of the middle image in Figure 10 using the threshold segmentation method are also good except that some details are missed in this case. The reason why PCNN method can realize the ideal edge detection is that it includes two significant characteristics: firstly, the output of PCNN is binary; secondly, the output is an area with single gray level, which ensures the successful detection of the real image edges and provides basis for segmentation.

Figure 11 shows several segmentation images with different  $r$  in (14), where the edges are similar. The value of the middle image is 0.3. The value of the right image is 0.8. We can find that the middle image is already good enough. More interesting results could be obtained by decreasing of  $r$ . The image will be segmented further based on the former coarse segmentation with the fire in different iteration of PCNN. However, it is also found that excessive segmentation will increase the difficulty in target recognition (the middle image of Figure 11). Thus, the number of iteration determines the capacity of PCNN to recognize edges with different gray levels.

Derived from the research on the target recognition mechanism of the mammalian visual neurons, the PCNN model can extract the edge and region information only after a few iterations instead of plenty of image training processes. Figure 12 shows several segmentation images with different iteration numbers, where the edges are similar. The number of Figures 12(b)~12(f) is 4, 6, 7, 9, and 10. We can find that Figure 12(d) is already good. More interesting results could be obtained by increasing the number of iteration. The image will be segmented further based on the former coarse segmentation with the fire in different iteration of PCNN. However, it is also found that excessive segmentation will increase the difficulty in target recognition (Figures 12(e), and 12(f)). Thus, the number of iteration determines the capacity of PCNN to recognize edges with different gray levels.

### 5. Conclusions

A spring clamp detection system based on machine vision is built in this paper, which is built on both sides of the production line. This system could detect accurately under the speed 2.8 clamps per second and detect any defect over 10 pixels. The rates of defect recognition and defective removal have reached a high level, which can achieve the goals of improving the overall quality of the products manufactured and avoid actual loss. The major work of this paper is as below: (1) an automatic location algorithm ROI is designed, which can fast locate the ROI in an image; (2) the most concise geometric principle is used in height detection algorithm design, which can achieve fast measurement on the premise that the angle between camera and detecting object is  $90^\circ$  or  $0^\circ$ ; (3) some mistakes of the equivalent circuit in Johnson and MA Yi-des documents are corrected; (4) it helps us to improve the original PCNN algorithm and apply it in this detection. This system is the application of machine vision in online detection of the spring clamp, which

is capable of automatically detecting the surface of the clamp. Of course, the detection system also needs to be optimized in a lot of aspects, for example, the distortion-free source coding problems such as how to remove unimportant information in an image during transmission in order to save cost and improve compression effect, which will be the next key point in further research.

### Appendix

From Figure 8, the following equation can be derived from nodal equation:

$$(V_2 - V_1) g_{12} = \Delta VY(t) + (V_1 - E_m) g_{m1} + (V_1 - E_s) F + C_1 \frac{dV_1}{dt}. \tag{A.1}$$

We obtain (A.2) from (A.1):

$$C_1 \frac{dV_1}{dt} = (V_2 - V_1) g_{12} - \Delta VY(t) - (V_1 - E_m) g_{m1} - (V_1 - E_s) F \tag{A.2}$$

$$= -(g_{m1} + g_{12} + F) V_1 + g_{12} V_2 + E_m g_{m1} + E_s F - \Delta VF(t). \tag{A.3}$$

Similarly,

$$(V_1 - V_2) g_{12} = (V_2 - E_m) g_{m2} + (V_2 - E_s) L + C_2 \frac{dV_2}{dt}. \tag{A.4}$$

From (A.3), we obtain

$$C_2 \frac{dV_2}{dt} = (V_1 - V_2) g_{12} - (V_2 - E_m) g_{m2} - (V_2 - E_s) L - (g_{m2} + g_{12} + L) V_2 + g_{12} V_1 + E_m g_{m2} + E_s L. \tag{A.5}$$

From (A.3) and (A.6), we obtain

$$\dot{V} = \bar{M}V + R, \tag{A.7}$$

where

$$V = \begin{pmatrix} V_1 \\ V_2 \end{pmatrix}$$

$$\bar{M} = \begin{pmatrix} -\frac{1}{C_1} (g_{m1} + g_{12} + F) & \frac{g_{12}}{C_1} \\ \frac{g_{12}}{C_1} & -\frac{1}{C_2} (g_{m1} + g_{12} + L) \end{pmatrix},$$

$$R = \begin{pmatrix} \frac{1}{C_1} (g_{m1} E_m + F E_s - \Delta VY(t)) \\ \frac{1}{C_2} (g_{m2} E_m + E_s L) \end{pmatrix}. \tag{A.8}$$

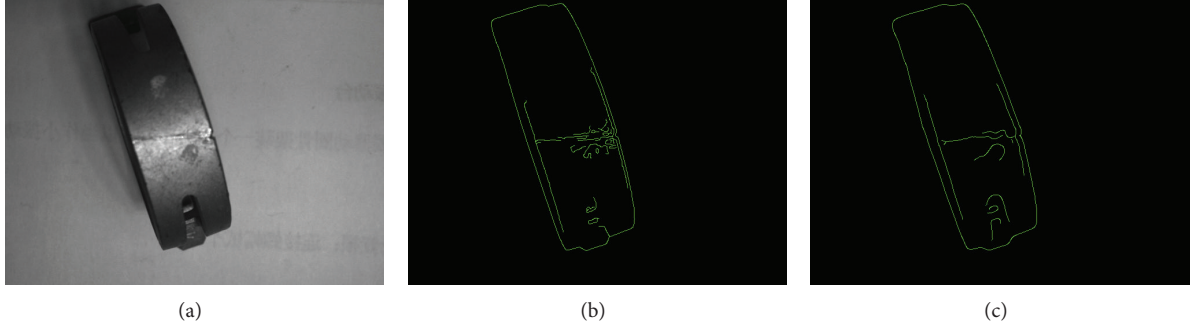


FIGURE 11: The original image (a), the output image based on original PCNN segmentation (b), and the output image based on our PCNN segmentation (c).

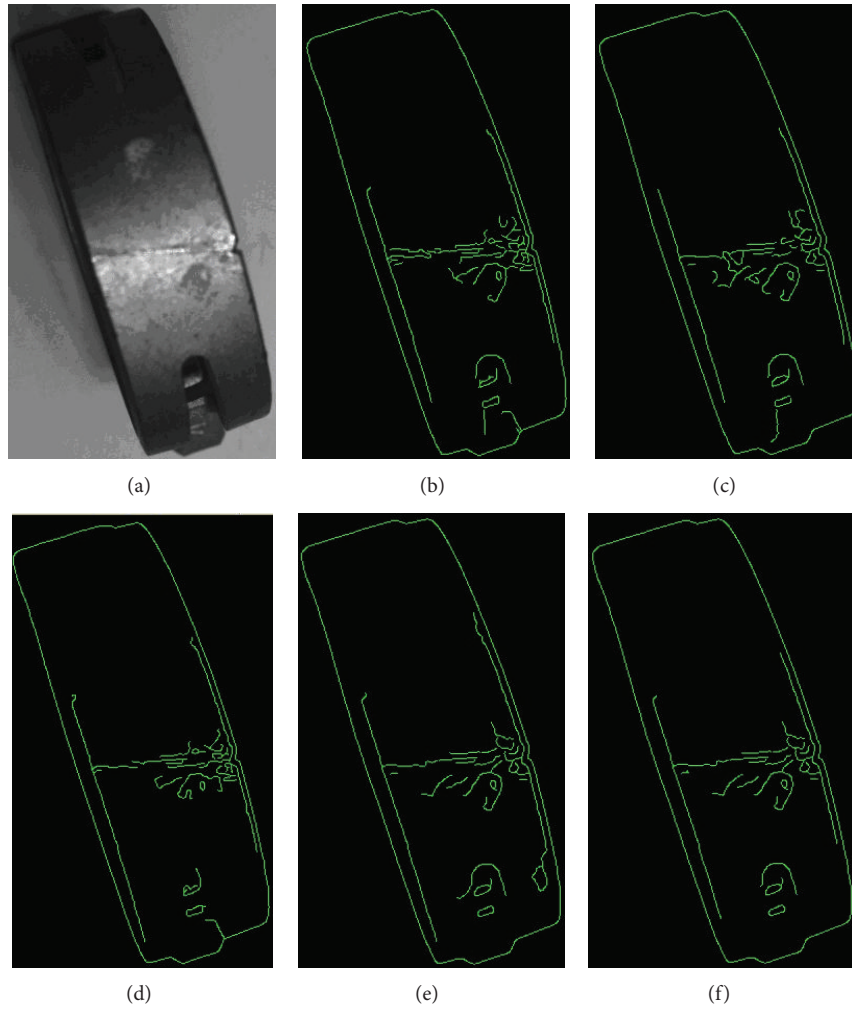


FIGURE 12: The original image (a) and segmentation images with different iteration numbers (b–f).

When the neuron achieves a stable stage, namely, no stimulus,  
 $\dot{V} = 0$  and  $Y(t) = 0$

$$\begin{aligned} -(g_{m1} + g_{12} + F)V_1 + g_{12}V_2 + E_m g_{m1} + E_s F &= 0, \\ g_{12}V_1 - (g_{m2} + g_{12} + L)V_2 + E_m g_{m2} + E_s L &= 0. \end{aligned} \quad (\text{A.9})$$

$V_1$  is solved:

$$\begin{aligned} &-(g_{m1} + g_{12} + F)(g_{12} + g_{m2} + L)V_1 \\ &+ E_m g_{m1}(g_{12} + g_{m2} + L) + E_s F(g_{12} + g_{m2} + L) \\ &+ g_{12}^2 V_1 + E_m g_{m2} g_{12} + g_{12} E_s L = 0. \end{aligned} \quad (\text{A.10})$$



Extracting  $V_1$ , we obtain

$$V_1 = \frac{w1}{w2}, \quad (\text{A.11})$$

where

$$\begin{aligned} w1 &= E_m g_{m2} g_{12} + E_m g_{m1} g_{12} + E_m g_{m2} g_{m1} \\ &\quad + (E_s g_{12} + E_m g_{m1}) L \\ &\quad + (E_s g_{12} + g_{m2} E_s) F + E_s FL, \end{aligned} \quad (\text{A.12})$$

$$\begin{aligned} w2 &= g_{m2} g_{12} + g_{m1} g_{m2} + g_{m1} g_{12} \\ &\quad + (g_{m2} + g_{12}) F + (g_{m1} + g_{12}) L + FL. \end{aligned} \quad (\text{A.13})$$

Simplify the expression above, we obtain,

$$V_1 = \frac{a + bF + cL + dFL}{e + fF + gL + hFL}, \quad (\text{A.14})$$

where,

$$\begin{aligned} a &= g_{12} g_{m2} E_m + g_{12} g_{m1} E_m + g_{m1} g_{m2} E_m, \\ b &= g_{12} E_s + g_{m2} E_s, \\ c &= g_{12} E_s + g_{m1} E_m, \\ d &= E_s, \\ e &= g_{12} g_{m2} + g_{12} g_{m1} + g_{m1} g_{m2}, \\ f &= g_{12} + g_{m2}, \\ g &= g_{12} + g_{m1}, \\ h &= 1. \end{aligned} \quad (\text{A.15})$$

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

The authors thank the reviewers and editors for their corresponding contributions in making the paper more presentable. This research is supported by National Natural Science Foundation of China (10972102) and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

## References

- [1] S. Yin, H. Luo, and S. Ding, "Real-time implementation of fault-tolerant control systems with performance optimization," *IEEE Transactions on Industrial I: Electronics*, vol. 64, no. 5, pp. 2402–2411, 2014.
- [2] S. Yin, S. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 11, pp. 6418–6428, 2014.
- [3] S. Yin, G. Wang, and H. Karimi, "Data-driven design of robust fault detection system for wind turbines," *Mechatronics*, vol. 24, no. 3, pp. 298–306, 2013.
- [4] S. Yin, S. X. Ding, A. H. A. Sari, and H. Hao, "Data-driven monitoring for stochastic systems and its application on batch process," *International Journal of Systems Science. Principles and Applications of Systems and Integration*, vol. 44, no. 7, pp. 1366–1376, 2013.
- [5] S. Yin, S. X. Ding, A. Haghani, H. Hao, and P. Zhang, "A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process," *Journal of Process Control*, vol. 22, no. 9, pp. 1567–1581, 2012.
- [6] R. Eckhorn, H. J. Reitboeck, M. Arndt, and P. Dicke, "Feature linking via synchronization among distributed assemblies: simulations of results from cat visual cortex," *Neural Computation*, vol. 2, no. 3, pp. 293–307, 1990.
- [7] J. L. Johnson and M. L. Padgett, "PCNN models and applications," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 480–498, 1999.
- [8] M. Yide, D. Rolan, and L. Lian, "Image segmentation of embryonic plant cell using pulse-coupled neural networks," *Journal on Communications*, vol. 23, no. 1, pp. 46–51, 2002.