IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 24, 2019, accepted June 23, 2019, date of publication June 28, 2019, date of current version July 22, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2925561*

# Real-Time Tiny Part Defect Detection System in Manufacturing Using Deep Learning

**JING YANG[1], (Student Member, IEEE), SHAOBO LI[1,2,3], ZHENG WANG[1], AND GUANCI YANG[3]**

[1]School of Mechanical Engineering, Guizhou University, Guiyang 550025, China
[2]Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, Guizhou University, Guiyang 550025, China
[3]Guizhou Provincial Key Laboratory of Public Big Data, Guizhou University, Guizhou 550025, China

Corresponding author: Shaobo Li (lishaobo@gzu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 91746116 Grant 61863005, in part by the Science and Technology Foundation of Guizhou Province under Grant 2015 4011, Grant 2016 5013, and Grant PTRC 2018 5702, and in part by the Collaborative Innovation of Guizhou Province under Grant YJSCXJH 2018 052.

**ABSTRACT** We adopted actual intelligent production requirements and proposed a tiny part defect detection method to obtain a stable and accurate real-time tiny part defect detection system and solve the problems of manually setting conveyor speed and industrial camera parameters in defect detection for factory products. First, we considered the important influences of the properties of tiny parts and the environmental parameters of a defect detection system on its stability. Second, we established a correlation model between the detection capability coefficient of the part system and the moving speed of the conveyor. Third, we proposed a defect detection algorithm for tiny parts that are based on a single short detector network (SSD) and deep learning. Finally, we combined an industrial real-time detection platform with the missed detection algorithm for mechanical parts based on intermediate variables to address the problem of missed detections. We used a 0.8 cm darning needle as the experimental object. The system defect detection accuracy was the highest when the speed of the conveyor belt was 7.67 m/min.

**INDEX TERMS** Defect detection, tiny parts, deep learning, SSD, missing detection rate.

## I. INTRODUCTION

During production and daily applications, machine tool production equipment can malfunction due to poor design and working conditions. Manufactured products are prone to defects, such as holes, sags, and abrasions. Corrosion and fatigue damage occur in daily applications, thus increasing production costs and causing considerable wasted resources and economic opportunities [1]. Defect detection technology has achieved favorable results in areas, such as pipelines [2.3], electronic components [4]–[6], parts [7]–[9], fault diagnosis [10.11] and others [12.13]. However, research on the defect detection of tiny parts, particularly the literature on real-time defect detection of tiny parts in conveyor belts, has remained scarce. Therefore, combined with defect detection sample attributes and the actual production environment, we study the defect detection technology based on deep learning, which will provide a reference for the manual setting of conveyor speed and industrial camera parameters in defect detection for factory products.

The associate editor coordinating the review of this manuscript and approving it for publication was Tomasz Trzcinski.

Deep learning, which has rapidly developed because of its efficient feature extraction ability [14.15], can be applied to the defect detection of tiny parts. The Faster R-CNN [16], [17], YOLO [18], and SSD[19] are currently the most popular methods for object detection. In 2015, Shao et al. proposed the Faster R-CNN deep learning object detection algorithm. However, the Faster R-CNN has a slow detection speed [17]. In 2016, Redmon et al. proposed the object detection algorithm YOLO [18] at an international conference on computer vision and pattern recognition (CVPR). In the same year, Wei Liu et al. proposed the object detection algorithm SSD [19] at a European conference on computer vision (ECCV). YOLO and SSD detect objects using regressions, and deep learning is used for real-time detection. The Faster R-CNN is a two-step object detection algorithm that detects objects through classification and regressions [20]. The SSD algorithm is a one-step object detection algorithm that directly detects objects using regressions. Therefore, we apply an SSD object detection method to improve the real-time performance of tiny part defect detection. A complete part defect detection system considers the feature extraction ability and real-time performance of the

detection algorithm. The stability and defect detection ability of the algorithm are indispensable key factors. For example, Taivedi *et al.* [21] described the significance of camera imaging for defect detection and proposed a single beam Fourier transform digital holographic interferometric technology for defect detection.

By combining a real-time defect detection system and the attributes of tiny parts, we propose a real-time tiny part defect detection method based on the abovementioned analysis. The experimental results demonstrate that the proposed method can achieve superior performance and strong adaptability. The main contributions of this work are presented as follows.

- The mainstream defect detection methods mainly focus on studying the defect degree of a detection sample. We fully consider the attributes of tiny parts and the environmental parameters of a defect detection system, including industrial camera parameters, illumination, and conveyor speed, and establish a relationship model between the detection capability coefficient of the tiny part system and the moving speed of parts. Thus, the robustness of the defect detection system is improved.
- With the integration of the improved SSD object detection algorithm and the correlation model between the detection capability coefficient and the moving speed of the parts, we analyze the optimal object recognition method SSD and propose a tiny part defect detection algorithm based on SSD and speed model. The proposed method has higher accuracy than YOLO V3, Faster-RCNN, and FPN.
- Missed detection is prone to occur when the dynamic defect detection is performed in the conveyor belt. Therefore, combined with the fiber sensor, conveyor speed, detection algorithm, and the attributes of tiny parts, we propose an algorithm for determining the missed detections of tiny parts based on intermediate variables, thereby increasing the stability and accuracy of the system.

The organization of this paper is presented as follows. In the first part, we review related research works on defect detection technology. In the second part, we describe the relationship model between the detection capability coefficient of a system of defective parts and the moving speed of parts. In the third part, we propose a tiny part defect recognition algorithm based on SSD. In the fourth part, we design an industrial real-time object detection platform and propose an algorithm to determine the missing detection of tiny parts based on intermediate variables. In the fifth part, we experimentally analyze the proposed method. Finally, we summarize the research work and discuss future research directions.

## II. RELATED WORK

Common defect detection methods include filtering, ultrasonic, and machine vision detection. For ultrasonic detection, Tan *et al.* [22] used computer simulations to study mobile thermal scanning to detect the defects detection bottom of mechanical products and adopted a second-order peak differential method to determine the defect depth, which has good accuracy for surface defect detection. However, the errors increased as the defect depth increased. Devivier *et al.* [23] proposed and applied a damage index that is based on the virtual field method to detect the defects of mechanical products. However, this index is sensitive to the changes of the properties, such as stiffness. To apply the current method to the defect detection of mechanical products, several tasks should be done to extend the current research scope to the defect detection of curved surfaces. Aiming at the challenge brought by the complexity of trailing pulses to ultrasonic detection, the self-focusing method of trailing pulses was proposed to improve the accuracy of defect location by theoretically deducing the characteristics of trailing pulses [24].

In filtering detection, Zou *et al.* [25] proposed a real-time X-ray flaw detection method for mechanical products based on Kalman filtering. Zhang *et al.* [26] used the zero-angle spatial filter and peak search to obtain the time center of the corresponding signal sources and proposed a new time-variant spatial filtering rearrangement scheme that is based on a microphone array. The proposed scheme overcame the Doppler distortion in acoustic bearing signals and the signal separation quality detection method of multi-bearing source mixing.

In machine vision defect detection, Boaretto and Centeno [27] proposed a double image exposure technique for mechanical product radiation image automatic detection and classification. The discontinuities of ''defects'' and ''no defects'' were taken as the indicators for the experiment, and the test data of the obtained classifier reached 88.6% accuracy with the use of semisupervised learning technology. Hajizadeh *et al.* [28] used high-frequency cameras to detect unmarked defect candidates and improve the imbalance of nondefective image data. Wakaf and Jalab [29] detected the defects of mechanical products using histogram matching from an image background. Martinez proposed the extraction of features from each region of the fused image and developed a machine vision system that detects the defects on machined metal parts. This research considered the illumination method in image processing to increase the defect detection accuracy, but it extracted the defective features in the fused image, thereby increasing the time overhead of defect detection and the difficulty of real-time defect detection [30].

In addition, deep learning [31]–[34] is widely used in product defect detection. Yang et al. proposed three-point circle fitting and convolutional neural network (CNN) to achieve automatic aperture detection. The automatic defect detection system will save time and labor costs. Song *et al.* [32] considered the defect detection problem of surface damage, surface dirt, and stripped screws; proposed the screw surface defect detection technology based on the CNN; and proved that deep learning technology was better than the traditional template matching technology. Wei *et al.* [33] used the CNN to classify the defects of a printed circuit board (PCB) and achieved

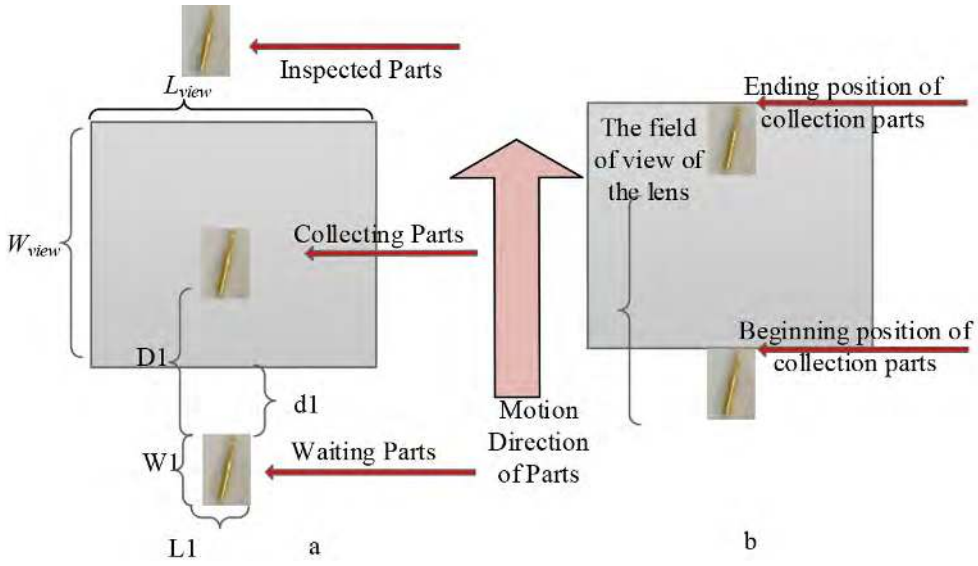


**FIGURE 1.** Schematic of the tiny part detection (a. Collection of tiny parts. b. Collection of a single sample).

better classification results for a data set containing 1818 collected images. Liu *et al.* [19], Krummenacher *et al.* [35] proposed a pipeline defect detection method based on deep learning, and showed that the size of the data set, the initialization of the network model, the training mode, and the network super-parameter impact the performance of the model. In conclusion, the application of deep learning to defect detection is highly significant.

## III. RELATION MODEL OF DETECTION CAPABILITY COEFFICIENT AND PART MOTION VELOCITY

The attributes of tiny parts and the environmental parameters of defect detection systems are the main factors that affect the stability of the system. By analyzing the performance and experimental parameters of each part of a system, we define the index that reflects the excellent defect detection of the system for tiny parts as the defect detection capability coefficient of tiny parts in unit time $\theta$.

We define the length and width of the camera field $d$ as $L_{view}$ and $W_{view}$, respectively. The length and width of a tiny part are $L_1$ and $W_1$, correspondingly. The distance of the tiny parts to be positioned from the bottom edge of the field of vision is $d_1$, and the conveyor belt speed is $v$. The direction of motion is the direction indicated by the arrow, as illustrated in Figure 1.

Figure 1(b) demonstrates that, in accordance with the size of the small part, when the camera' field of vision $d$ is larger than the size of the tiny parts, that is, $W_{view} > \mathrm{W}_1, L_{view} > L_1$. The image collection of the tiny parts begins when the upper edge of a tiny part coincides with the edge of the camera' field of vision, while the collection ends when the upper edges of the tiny parts coincide with the edge of the lens. Considering the processing image time of computers, $T$ is assumed to be the necessary time for the system to respond. If $T \ll t$, then the necessary time for collecting an image is expressed as follows:

$$t = {W_{view}}/{V}. \tag{1}$$

If the sample image is not collected after time $t$, then a missed detection will occur. The formula for calculating the capability coefficient of the system in detecting tiny parts in unit time is expressed as follows:

$$\theta = \frac{{W_{view}}/{(D_1 + W_1)}}{t}. \tag{2}$$

Formula (1) is substituted into Formula (2) to establish the relationship model between the detection coefficient and the movement speed of the tiny parts.

$$\theta = \frac{v}{D_1 + W_1}. \tag{3}$$

Formula (3) presents that, if only the size of tiny parts and the camera' field of vision are considered, then the defect detection capability coefficient of the system in unit time is directly proportional to the conveyor belt speed when the size of the tiny parts and the camera' field of vision remains unchanged. If $T > t$, then the processing time of the computer is long, the processing of the tiny parts will be incomplete, and the tiny parts to be processed will enter the camera' field of vision. We define the time of the missing tiny parts as follows:

$$\Delta t = T - t. \tag{4}$$

In the initial stage, the distance of a tiny part in $\Delta t$ time can be expressed as follows:

$$\Delta S = \Delta t.v. \tag{5}$$

When $\Delta S = W_{view}$, the detection of a tiny part is missed. The system loss rate $\varepsilon$ is defined as follows:

$$\varepsilon = 50\% \tag{6}$$

At this time, the defect detection capability coefficient of the system can be expressed as follows:

$$\theta = (1-\varepsilon)\frac{v}{D_1+W_1} = \frac{v}{2(D_1+W_1)}. \tag{7}$$

When $d_1 \le \Delta s \le W_{view}$ and $\Delta s = d_1$, the second sample of tiny parts can be identified, but the third sample image cannot be recognized. This case is called data loss. When $\Delta s > d_1$, the distance between the *nth* sample image and the standard position is presented as follows:

$$S_{dis} = n.\Delta S. \tag{8}$$

When $S_{dis} = W_{view}$, the situation can be considered an error cycle.

$$n = \frac{W_{view}}{\Delta S}. \tag{9}$$

Formulas (4) and (5) are integrated into Formula (9).

$$n = \frac{W_{view}}{v.(T-t)}. \tag{10}$$

After considering the number of samples to be tested, we assume that the number of samples in camera' field of vision is expressed as follows:

$$P = {}^{W_{view}}\!/_{(D_1+W_1)}. \tag{11}$$

The loss rate of tiny parts at this time is computed as follows:

$$\varepsilon = \frac{P.v.(T-t)}{W_{view}} \times 100\%. \tag{12}$$

The substitution of Formula (11) into Formula (12) yields

$$\varepsilon = v.\frac{T-t}{D_1+W_1} \times 100\%. \tag{13}$$

The detection capability coefficient of the system at this time for tiny parts is expressed as follows:

$$\theta = -v^2.\frac{T-t}{(D_1+W_1)^2} + v.\frac{1}{D_1+W_1}. \tag{14}$$

When the speed of the conveyor belt $v$, the camera' field of vision range $W_{view}$, and the size of tiny parts are determined, the defect detection coefficient of the system's tiny parts is inversely proportional to the detection speed $T$ of the program. In Formula (14), $v$ considerably influences the coefficients of the tiny parts in the system. When $W_{view} \le \Delta S \le 2W_{view}$,

$$\epsilon = \frac{2v.(T-t)}{D_1+W_1} \times 100\%. \tag{15}$$

By analogy, when $2W_{view} \le \Delta S \le 3W_{view}$,

$$\varepsilon = \frac{3v.(T-t)}{D_1+W_1} \times 100\%. \tag{16}$$

When $(m-1)W_{view} \le \Delta S \le mW_{view}$ ($m > 1$ and $m$ is a positive integer),

$$\varepsilon = \frac{mv.(T-t)}{D_1+W_1} \times 100\%. \tag{17}$$

With the integration of Formula (17) into Formula (7), the system defect detection capability coefficient can be calculated as follows:

$$\theta = -v^2.\frac{m\,(\text{T-t})}{(D_1+W_1)^2} + v.\frac{1}{D_1+W_1}. \tag{18}$$

According to the abovementioned analysis, when the lens field $W_{view}$, the system response time $t$, and the distance $D_1$ between two tiny parts are constant, the defect detection capability coefficient of the system is a quadratic function in relation to the conveyor belt speed. Moreover, the quadratic function $\theta$ is continuous and derivable. The maximum point of $\theta$ is $v = (W_3 \times m + D_1 + W_1)/2mT$. The speed of the system is the optimal conveyor belt speed, and its defect detection ability is the optimal rate.

## IV. TINY PART DEFECT RECOGNITION ALGORITHM BASED ON SSD

### A. SSD OBJECT DETECTION

An SSD network combines the anchor mechanism of the YOLO [18] regression and the Faster-RCNN [17]. The regression is adopted to simplify the computational complexity of the neural network and improve the real-time performance of the algorithm. The anchor mechanism is used to extract the features with different aspect ratios. This local feature extraction method is more reasonable and effective than the YOLO method for global feature extraction for a certain location in terms of the feature extraction ability. In addition, the SSD adopts the method of multiscale [36] object feature extraction for the feature expression of different scales. This design is helpful for improving the robustness of detecting objects with different scales. Given these advantages of SSD networks, this study uses an SSD network to detect defects in tiny parts.

#### 1) FEATURE LAYER MAPPING MODEL

In consideration of the different attributes of tiny parts and the idea that an SSD network adopts a multiscale method to obtain multiple feature graphs with different sizes, a m-layer feature graph is used for model detection. The formula for calculating the proportion of the default box of the *k-th* feature graph is as follows:

$$S_k = S_{min} + \frac{S_{max}-S_{min}}{m-1}(k-1), \quad k \in \{1,2,\ldots,m\}. \tag{19}$$

where $S_{min} = 0.2$ and $S_{max} = 0.95$, which represents the proportion of the default box of the feature layer in the input image. The SSD uses the anchor mechanism to determine the different aspect ratios for the default box on the same feature layer to enhance the robustness of the default box to the shape

of the object. In this work, the default box aspect ratio is $r = \{1, 2, 1/2, 3, 1/3\}$. For the class when the aspect ratio is equal to 1, $S_k^{'} = \sqrt{S_k S_{k+1}}$ is added. Then, we get the following.

$$W_k^n = S_k\sqrt{r}_n, h_k^n = \frac{S_k}{\sqrt{r}_n}, \quad n \in \{1, 2, 3, 4, 5\}. \tag{20}$$

$$W_k^6 = h_k^6 = \sqrt{S_k S_{k+1}}. \tag{21}$$

We set the default box of the center for $\left(\frac{a+0.5}{|f_k|}, \frac{b+0.5}{|f_k|}\right)$, whereby $|f_k|$ is the *k-th* feature image size, and $a, b \in \{0, 1, 2, \ldots, |f_k| - 1\}$. We set the coordinates of the default box so that it is within [0,1]. The mapping relationship between the default box coordinates on the feature image and the original image coordinates is as follows.

$$x_{min} = \frac{c_x + \frac{w_b}{2}}{w_{feature}} w_{ing} = (\frac{a + 0.5}{|f_k|} - \frac{w_k}{2}) w_{ing} \tag{22}$$

$$y_{min} = \frac{c_y + \frac{h_b}{2}}{h_{feature}} h_{ing} = (\frac{b + 0.5}{|f_k|} - \frac{h_k}{2}) h_{ing}. \tag{23}$$

$$x_{max} = \frac{c_x + \frac{w_b}{2}}{w_{feature}} w_{ing} = (\frac{a + 0.5}{|f_k|} + \frac{w_k}{2}) w_{ing}. \tag{24}$$

$$y_{max} = \frac{c_y + \frac{h_b}{2}}{h_{feature}} h_{ing} = (\frac{b + 0.5}{|f_k|} + \frac{h_k}{2}) h_{ing}. \tag{25}$$

Here, $(c_x, c_y)$ are the coordinates of the default box center on the feature layer; $w_b$ and $h_b$ are the width and height of the default box, respectively; $w_{feature}$ and $h_{feature}$ are the width and height of the feature layer, respectively; and $w_{img}$ and $h_{img}$ are the width and height of the original image, respectively. After ($x_{min}$, $y_{min}$, $x_{max}$, and $y_{max}$) are obtained, the object frame coordinates of the *k-th* layer are mapped to the original image.

#### 2) LOSS FUNCTION

The SSD's training simultaneously conducts the regressions of the position and the object type. The object loss function is the sum of the confidence and the position loss, and its expression is as follows:

$$L(z, c, l, g) = \frac{1}{n}(L_{conf}(z, c) + L_{loc}(z, l, g)). \tag{26}$$

where $n$ is the number of default frames matching the ground truth object frame; $L_{conf}(z,c)$ is the confidence loss; $L_{loc}(z,l,g)$ is the position loss [37]; $z$ is the matching result of the default box and the ground truth object boxes of different categories; $c$ is the confidence of the prediction object frame; $l$ is the position information of the prediction object box; $g$ is the position information of the object frame of the ground truth; and $\alpha$ is a parameter that weighs the confidence loss against the position loss, whereby we set $\alpha$ to 1. The object loss function and position contain incredible losses. In the training process, the SSD algorithm improves the confidence of the prediction box object by reducing the loss function value and enhancing the position reliability of the prediction box. The object detection performance of the model is continuously improved through several optimizations of the object detection results. Hence, an enhanced prediction model is trained.

### B. TINY-PART DEFECT DETECTION ALGORITHM BASED ON SSD AND SPEED MODEL

The original color image is converted into a binary image by preprocessing to improve the accuracy of the defect detection of tiny parts. The areas of tiny parts in the conveyor belt are determined by locating these areas to simplify the calculation, and the detected simple image is cut to remove useless background noise. A boundary detection algorithm is used to determine the four boundaries of tiny parts and realize an accurate location. the grid is divided to extract the data information of the tiny parts, and the extracted data are transmitted to the subsequent SSD defect identification program to obtain the class of defects. The specific algorithm is described as follows.

**Algorithm 1** Tiny Part Defect Detection Algorithm Based on SSD and Speed Model (TP-SSDM)

**Input:** Image data $X_{pic}$
**Output:** Image data $X_{\text{pic}}$; predicted class estimates for defect types $p_{\text{pic}}$; part class $P_{class}$
**Step 1:** Set the image pixels of the sample tiny parts.
**Step 2:** When the optical fiber sensor transmits the frequency of the optical signal to the industrial camera, the sensor begins the collection of the sample image of tiny parts.
**Step 3:** Set the background information of the tiny part sample. The industrial camera is set to the bright part area, and the background color is dark.
**Step 4:** Use the subpixel edge extraction function *select_obj()* in Halcon to extract the contours of the tiny parts.
**Step 5:** Fit the shapes of circles and lines in the detection sample to obtain the position and size information of the detection object, which is the bounding box of the first feature map.
**Step 6:** The results obtained in Step 3 are integrated into Formulas (19)– (25), the feature information of the *k-th* feature graph is obtained as the output vector of the first layer of the SSD network, and the output vector is used as the input of the second layer.
**Step 7:** The VGGNet [38] network is used as the training network of the SSD, and convolution and pooling operations are used to obtain the feature vector $V$ of the samples.
**Step 8:** Formula (26) is used to reduce the value of the loss function and thus improve the confidence of the type of prediction box while enhancing the position reliability of the prediction box.
**Step 9:** The eigenvector $V$ is used as the input of the *softmax()* classification function, and the predicted class estimates of defect type $p_{\text{pic}}$ and part class $P_{class}$ are obtained.
**Step 10:** Output the predictive probability estimate $p_{\text{pic}}$ and part class $P_{class}$.

## V. INDUSTRIAL REAL-TIME TINY PART DETECTION EXPERIMENTAL PLATFORM

Figure. 2 illustrates the experimental testing platform that was designed by the research group [39]. It includes a conveyor belt, data processor, data acquisition sensor, light source, and parts for mechanical support. The touch screen is used for the data input and the display is a 32-inch industrial touch screen. The vision sensor device uses a MindVision high-speed industrial camera with an electronic rolling shutter, which can collect high-speed samples that can be tested in real time. The data processor is a Raspberry Pi B3. To ensure the sufficiency of the light in the system box, a strip environmental light source LED with adjustable brightness is installed on the box. A biological LED ring light is used for sample imprinting. The workstation that is used to reduce the computing load of the data processor is configured with an Intel Xeon e5-1620 V3 3.5 GHz CPU, 16 GB of memory, a Nvidia GeForce GTX1080, and Ubuntu16.04 and is mainly used for data analysis. The Raspberry Pi and the workstation are equipped with OpenCV 3.1, TensorFlow 1.4, YOLO V3, and the SSD. The Raspberry Pi B3 has a wireless communication module that can realize end-to-end communication between the experimental test platform and the workstation.

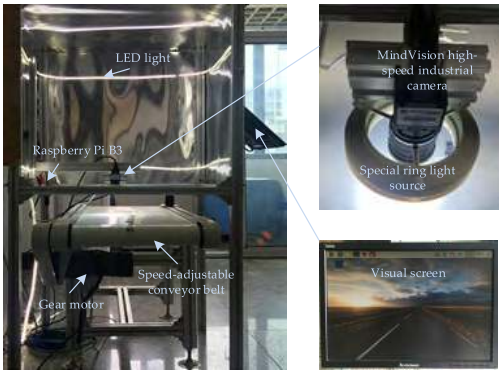


**FIGURE 2.** Industrial real-time part detection experimental platform.

To enable the algorithm to detect the defects of different tiny parts quickly and accurately, we combine it with the real-time detection equipment. Then, we design a real-time defect detection process for tiny parts based on Algorithms 1 and 2 and the speed model. The flowchart is shown in Figure. 3. When the optical fiber sensor detects the mechanical parts on the conveyor belt, the control module judges the movement of the mechanical parts into the visual field close to the camera system. In the center, trigger pulses are sent to the image acquisition part, and the trigger pulses are then sent to the industrial camera and the lighting system according to preset procedures and delays. The industrial camera begins capturing images, and the microscope LED ring light is illuminated by a special ring light source. The lighting opening time matches the exposure time of the industrial camera. After the camera captures the image, the digital image is stored in the memory of the processor. The defect detection algorithm based on the SSD is used to process, analyze, and identify the defects of the tiny parts, and the result of the detection is obtained. The processing result is sent to the control unit for sorting and selecting the defective parts.

**Algorithm 2** Missing Detection Algorithm for Mechanical Parts Based on Intermediate Variables (**AMP-IV**)

**Input:** Image data $X_{\text{pic}}$
**Output:** Loss rate of tiny parts $\varepsilon$
**Step 1:** Initialize the intermediate variables $\mu_i = 0$, and set the attributes of the tiny parts to be detected: the number of the tiny parts and the size of the pixels of the captured image.
**Step 2:** According to the transmission signal of the optical fiber sensor, the industrial camera captures the sample image $X_{\text{pic}}$.
**Step 3:** Use Algorithm 1 for defect detection to obtain the predicted class estimation value $p_{\text{pic}}$ of the defect type of the detection samples.
**Step 4: If** $P_{class} \neq 0$ or $P_{pic} \neq 0$, **then**
The AMP-IV algorithm does not detect the missing tiny parts. The predicted class estimation value $p_{\text{pic}=}\mu_0$ and $\varepsilon \neq 0$.
**else**
The loss rate of tiny parts is $\varepsilon = 0$. The AMP-IV algorithm detects the missing tiny parts and proceeds to Step 3.
**end if**
**Step 5: if** $\mu_1 - \mu_0 \neq 0$, (that is, no missed detection is observed), **then**
Continue to Step 3, and output $\varepsilon = 0$.
**else**
$\mu_1 - \mu_0 = 0$ indicates that AMP-IV has missing parts between the two tiny parts. Output $\varepsilon = 1$.
**end if**

We propose an algorithm based on intermediate variables to determine the missed detections of tiny parts. The specific algorithm is described as follows.

## VI. EXPERIMENTAL RESULT AND ANALYSIS

### A. DATASET

#### 1) EXPERIMENTAL DATA COLLECTION

The four kinds of defects in 0.8 cm darning needles, which are frequently used in actual production, are crooked shapes, length size errors, endpoint size errors, and wringing. The 0.8 cm darning needle dataset is collected in motion with the high-speed industrial camera on the self-built industrial real-time object detection platform. The distance between the camera and the belt is 10 cm. We collected the 0.8 cm darning needle dataset with 3000 images using constant changes in the locations of the detection objects, including 2140 training and 860 testing images. The training and testing images contained 6306 and 2000 0.8 cm darning needle labels, correspondingly. The collected samples are depicted in Figure 4.

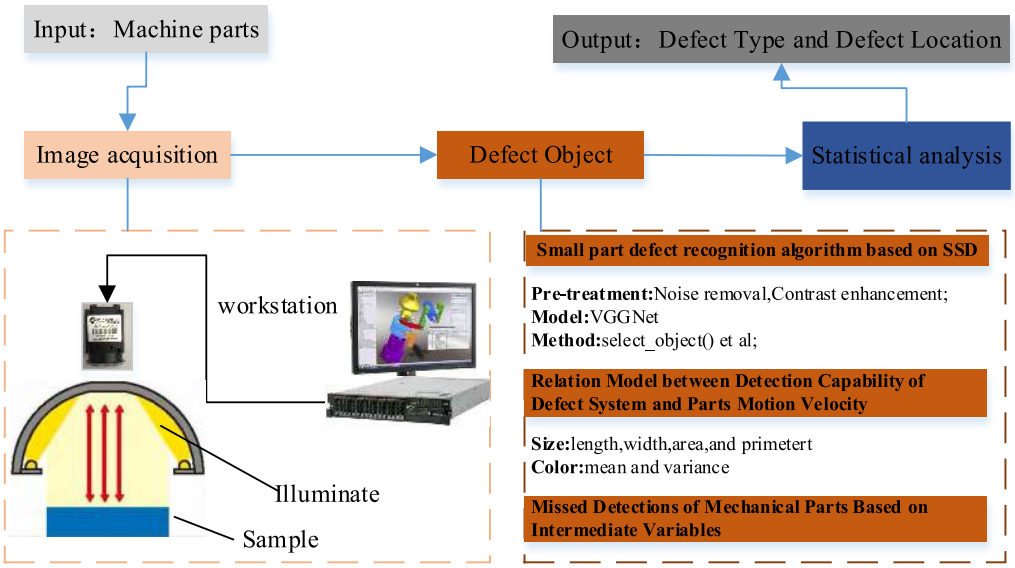


**FIGURE 3.** Defect detection process for tiny parts.

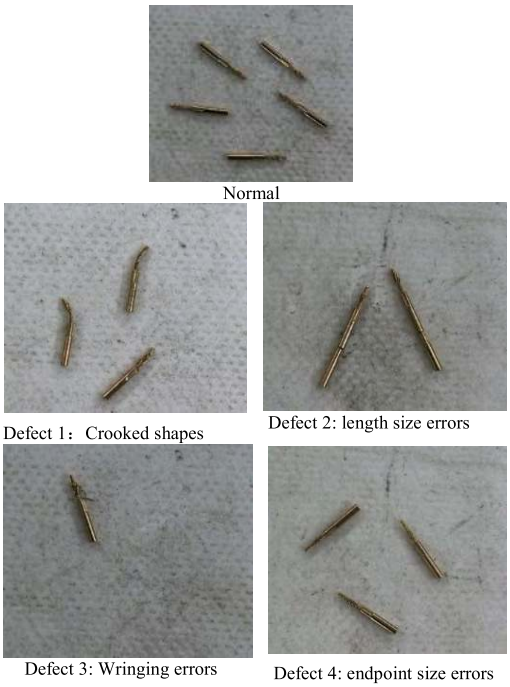


**FIGURE 4.** Example of autonomous collection of experimental data.

### 2) EXPERIMENTAL DATA AUGMENTATION

Data augmentation is a common method that can improve the robustness of an algorithm without degrading the detection accuracy. The collection of data that satisfy various conditions for training and testing can be difficult. Therefore, many object detection algorithms use data augmentation to evaluate the generalizability of the designed algorithm in specific scenarios and practical applications. Data augmentation methods include horizontal flipping, translation, cutting, and color dithering [40]. We use data augmentation operations on the image data with rotation angles of 0°, 22.5°, 45°, 67.5°, and 90° to improve the generalizability of the model in different directions. Table 1 summarizes the detailed information on the label number of classes in each category of the original data, data augmentation, and new data.

**TABLE 1.** 0.8 cm darning needle data label information.

| Data / Defect | Original Train | Augmentation Train | New data Train | New data Test |
|---|---|---|---|---|
| Normal | 1261 | 600 | 1861 | 400 |
| Defect 1 | 1258 | 600 | 1858 | 400 |
| Defect 2 | 1239 | 600 | 1839 | 400 |
| Defect 3 | 1246 | 600 | 1846 | 400 |
| Defect 4 | 1302 | 600 | 1902 | 400 |
| Total | 6306 | 3000 | 9306 | 2000 |

## B. RELATIONAL MODEL OF DETECTION CAPABILITY COEFFICIENT OF DEFECT SYSTEM AND PART MOTION VELOCITY

### 1) EXPERIMENTAL PARAMETER SETTING

The proposed method is validated on the experimental platform. In accordance with the relationship model between the defect detection system of the tiny parts and the moving speed of the conveyor belt, combined with the structures of the experimental equipment and the tiny parts, we set the parameters, as listed in Table 2. Moreover, we use Formula (18) to calculate the optimal speed of the conveyor belt under the experimental parameters and detect the defects. The following definition is used: The conveyor belt speed per second is approximately $v = [0\,\text{m}, 10\,\text{m}]$. We use MATLAB to simulate

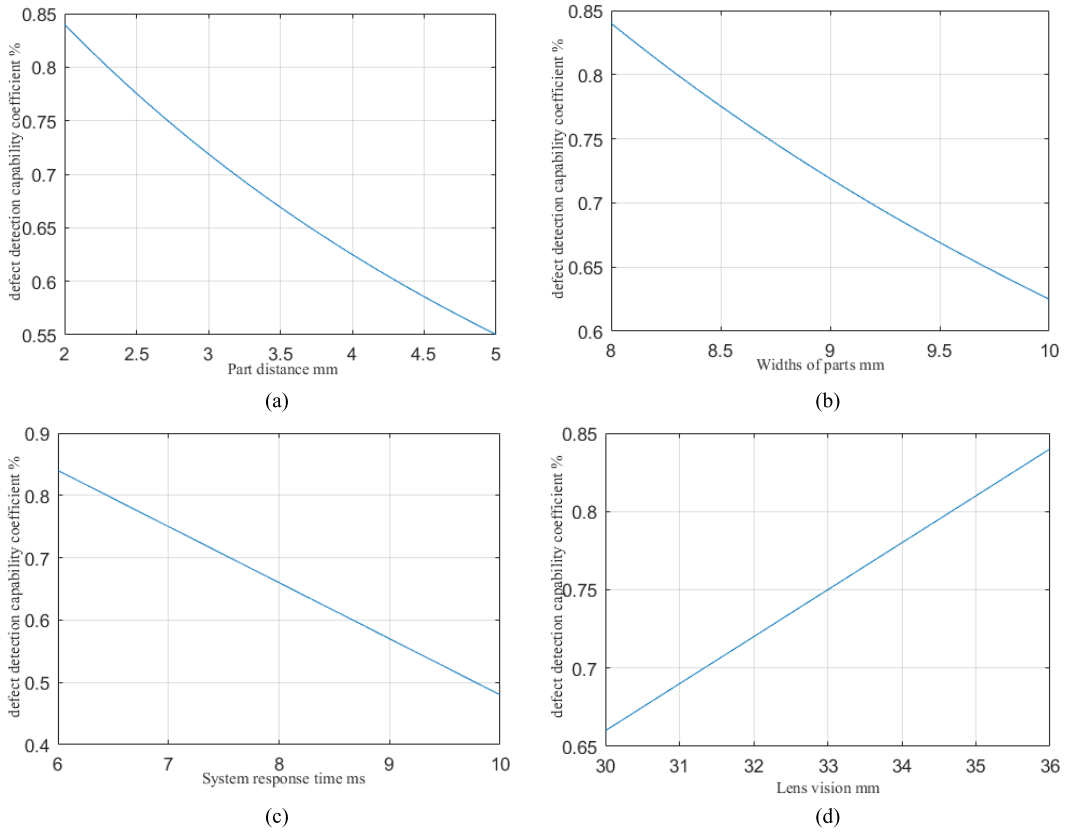


**FIGURE 5.** Influence of other system parameters on the system defect detection capability coefficient: (a) distance between parts and the system defect detection capability coefficient, (b) width of parts and the system defect detection capability coefficient, (c) system response time and the defect detection capability coefficient, and (d) the camera' field of vision and the system defect detection capability coefficient.

**TABLE 2.** Parameter setting of experimental system.

| Parameter Name | $D1(/mm)$ | $W_1(/mm)$ | $W_{view}(/mm)$ |
|---|---|---|---|
| Parameter Value | 2 | 8 | 36 |
| Parameter Name | $t(/ms)$ | $T(/ms)$ | $m$ |
| Parameter Value | $W_{view}$/v | 6 | 1 |

and analyze the defect detection capability coefficient of the system and the speed of the conveyor belt. The simulation results show that the speed of the conveyor belt is directly proportional to the defect detection capability of the system.

$$\begin{aligned}\theta &= -v^2 \times \frac{m\,(\text{T-t})}{(D_1+W_1)^2} + v \times \frac{1}{D_1+W_1} \\ &= -v^2 \times \frac{6-{}^{36}\!/_{v}}{(2+8)^2} + v \times \frac{1}{2+8} \\ &= -v^2 \times \frac{3}{50} + \frac{9}{25} \times v + \frac{v}{10} \\ &= -v^2 \times \frac{3}{50} + \frac{23}{50} \times v.\end{aligned}$$

This function is quadratic, and the parabolic opening is downward. The point whose derivative is 0. its unique maximum point.

$$\theta' = -v \times \frac{3}{50} + \frac{23}{50}.$$

When $\theta' = 0$, the following formulas are obtained:

$$-v \times \frac{3}{50} + \frac{23}{50} = 0$$
$$v = \frac{23}{3} \approx 7.67.$$

### 2) DEFECT DETECTION SYSTEM FOR IDENTIFYING THE RELATION BETWEEN PART COEFFICIENT AND MOTION VELOCITY

The high-speed recognition algorithm of tiny parts based on the SSD realizes the high-speed defect detection of tiny parts. In the experiment, $520 \times 520\ pix$ are the pixels of the sampled image. Considering the reflection and complex background interference on the actual production line, the defect detection ability of the miniature parts with the conveyor belt speed of 7.67 m/min is the optimum in the sampling process.

Figure 6. Influence of other system parameters on the system defect detection capability coefficient: (a) distance between the parts and the system defect detection capability coefficient, (b) width of the parts and the system defect

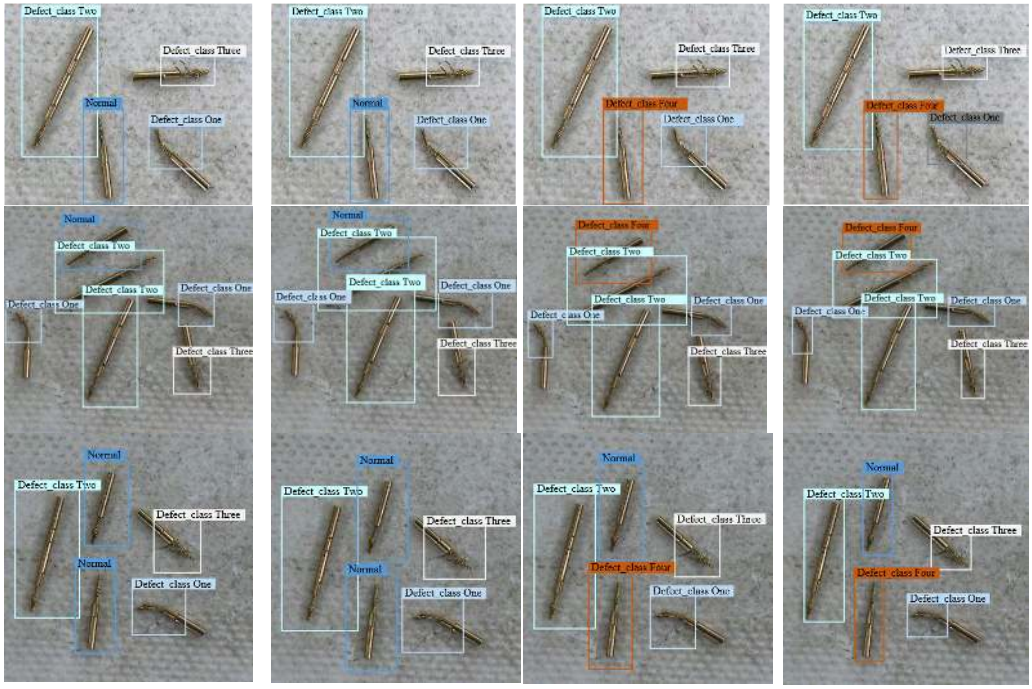

**FIGURE 6.** Shows the defect detection sample (It compares the pedestrian detection results with those of state-of-the-art methods. The first column shows the input images with the ground truths annotated with red rectangles. The other columns show the detection results (green rectangles) of YOLO V3, the FPN, the Faster-R-CNN, and our method. Our method can successfully detect most small-size instances that the two state-of-the-art methods miss.).

**TABLE 3.** Detection results of Defect 1 under different algorithms.

| Algorithm | Predictive probability estimates (%) | Accuracy rate (%) | Location time (/s) | Missing detection rate (%) | Training time (/h) |
|---|---|---|---|---|---|
| YOLO V3 | 87.35% | 92.80% | **0.51** | 1.40% | **32** |
| Faster-RCNN | 88.65% | 91.20% | 1.39 | 3.80% | |
| FPN | 92.27% | 95.00% | 0.68 | 1.40% | |
| Our Method | **90.67%** | **98.00%** | 0.56 | **1.40%** | |

detection capability coefficient, (c) system response time and defect detection capability coefficient, and (d) the camera' field of vision and system defect detection capability coefficient.

A comparison of the influence of each parameter on the system defect detection capability coefficient in Figure 6 indicates that the small distance between two tiny parts leads to a narrow width of the tiny parts. When the system processing time is short, the system defect detection capability coefficient is high; when the camera' field of vision is large, and the movement speed of the tiny parts is 7.67 m/min, the system defect detection capability coefficient is low. The experiments show that, when the distance between two defect detections is 2.00 mm, the width of the tiny parts is 8.00 mm, the camera' field of vision is 36.00 mm, and the defect detection capability coefficient of the tiny parts on the experimental platform is the highest.

### C. COMPARISON OF PROPOSED ALGORITHM WITH CLASSICAL ALGORITHM

We use an 0.8 cm darning needle as the experimental object to verify the validity of the algorithm further and compare the proposed algorithm with YOLO V3 [18], Faster-RCNN [17], FPN [41], and the proposed method. The settings of comparison algorithms are similar to those of the SSD experiment [19]. The four kinds of defect data that are observed in the test dataset are analyzed from subjective and objective perspectives. The test set has 860 and 2000 images in the 0.8 cm darning needle label data. The detailed data information is summarized in Table 1. The statistical results of the recognition accuracy and predictive probability estimates of the algorithm in the system are presented in Tables 3 –6, and the subjective detection results are illustrated in Figure 6.

#### 1) ANALYSIS OF SUBJECTIVE TEST RESULTS

Figure 6 depicts the experimental results of the subjective defect detections of the algorithm in this work and the compared algorithms. The figure also demonstrates that YOLO V3 and Faster-RCNN cannot reliably detect the size error (fat point size) defects. This result occurs mainly because the sample in YOLO V3 must identify the image segmentation for a $7 \times 7$ grid. The inside of the cell is used to detect the object neurons with information loss problems, thereby resulting in a model with a strong spatial constraint in the

**TABLE 4.** Detection results of Defect 2 under different algorithms.

| Algorithm | Predictive probability estimates (%) | Accuracy rate (%) | location time (/s) | Missing detection rate (%) | Training time (/h) |
|---|---|---|---|---|---|
| YOLO V3 | 90.46% | 95.00% | **0.47** | 0.60% | 162 |
| Faster-RCNN | 92.37% | 94.80% | 1.27 | 1.60% | |
| FPN | 92.89% | 96.80% | 0.55 | 1.00% | |
| Our Method | **95.36%** | **99.00%** | 0.52 | **0.60%** | |

**TABLE 5.** Detection results of Defect 3 under different algorithms.

| Algorithm | Predictive probability estimates (%) | Accuracy rate (%) | location time (/s) | Missing detection rate (%) | Training time (/h) |
|---|---|---|---|---|---|
| YOLO V3 | 86.59% | 88.20% | **0.46** | 1.60% | 123 |
| Faster-RCNN | 89.76% | 87.60% | 1.27 | 3.20% | |
| FPN | 91.52% | 93.40% | 0.57 | 2.00% | |
| Our Method | **93.89%** | **97.80%** | 0.56 | **1.00%** | |

**TABLE 6.** Detection results of Defect 4 under different algorithms.

| Algorithm | Predictive probability estimates (%) | Accuracy rate (%) | location time (/s) | Missing detection rate (%) | Training time (/h) |
|---|---|---|---|---|---|
| YOLO V3 | 49.38% | 53.80% | **0.47** | 1.60% | 107 |
| Faster-RCNN | 53.49% | 56.40% | 1.33 | 3.00% | |
| FPN | 66.79% | 72.60% | 0.63 | 1.80% | |
| Our Method | **71.53%** | **79.40%** | 0.57 | **1.00%** | |

process of feature extraction. If the grid contains small, similar detection tasks, then the system cannot simultaneously detect all the objects. The Faster-RCNN detects an object based on information, such as the color and image edges, thus denoting its insufficient ability to detect weak objects. The bounding box is remarkably larger in the FPN than in the method proposed in the present work. The bounding box regression of the FPN does not converge. This work is based on the SSD for defect detection. The SSD network is found in the multi-feature layer detection frame extraction and improves the accuracy of tiny object bounding box regression. The FPN can also obtain the defect type and provide the approximate locations of the defects accurately. However, accurate and appropriate object location is crucial to calculating the grasping position of the manipulator in an actual robotic grabbing task.

### 2) OBJECTIVE ANALYSIS OF TEST RESULTS

Tables 3–6 correspond to the detection results of the proposed method and the comparison algorithms under the four defect type datasets. Table 3 reflects that, under Defect Type 1, the predictive probability estimates are 3.32%, 5.20%, and −1.60% higher in the proposed method than in YOLO V3, Faster-RCNN, and FPN, respectively. The accuracy rates are 5.20%, 6.80%, and 3.00% higher than those of the correspondingly compared algorithms. The location times are −0.05, 0.83, and 0.12 less than those of the respectively compared algorithms. The missing detection rate is 2.40% lower than that of the Faster-RCNN.

In Table 4, under Defect Type 2, the predictive probability estimates, and accuracy of the proposed method are 95.36% and 99.00%, correspondingly. The FPN has the highest predictive probability estimation and accuracy among the compared algorithms. The estimated probability and accuracy of the proposed method are 2.47% and 1.20% higher than those of the FPN, respectively. In Table 5, in the case of Defect Type 3, the prediction probability estimation and accuracy of the proposed method are 2.37% and 3.40% higher, correspondingly, than those of the FPN with the highest comparison algorithm. The location time is −0.10 less than that of the lowest YOLO V3 algorithm. The missed detection rate is the same as that of the lowest YOLO V3 algorithm.

In the case of Defect Type 4, the prediction probability estimation and accuracy of the proposed method are 4.74% and 6.80% higher, respectively, than those of the FPN. The location time is −0.10 less than that of the lowest YOLO V3 algorithm. The missed detection rate is 0.60% less than that of the lowest YOLO V3 algorithm.

Further observation of Figure 7 indicates that the accuracy is higher in the proposed method than in the compared algorithms under the four defect types. The location time of YOLO V3 is the shortest, and the location time is shorter in the proposed method than in the Faster-RCNN and FPN. In terms of the relevant datasets and experimental parameters, the missed detection rate is lower in the proposed method than in the comparison algorithms. The training time is 32 h shorter in the proposed method than in YOLO V3 and 55 h and 16 h lower than in the Faster-RCNN and FPN, respectively.

In summary, the accuracy is higher in the proposed method than in the comparison algorithms because an anchor mechanism is used to extract the features with different aspect ratios. The proposed local feature extraction method is more reasonable and effective than the global feature extraction

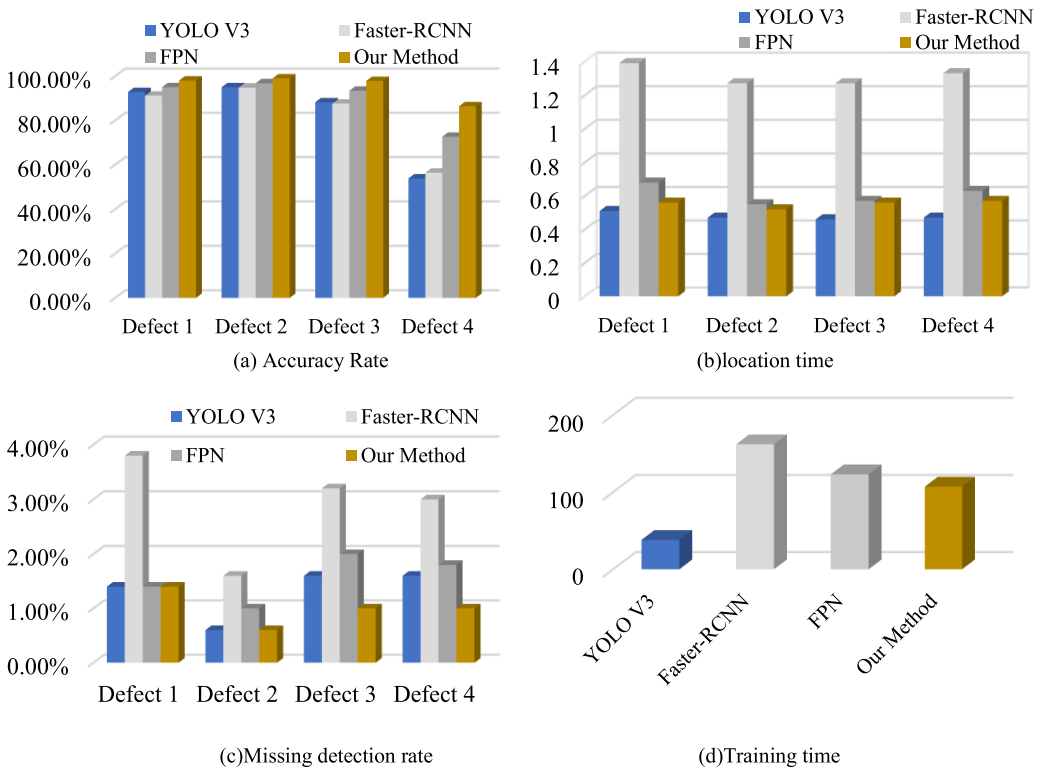


**FIGURE 7.** Test results of the system under different indicators.

method of YOLO V3 for a certain location in terms of the feature extraction ability. However, detecting YOLO V3 samples as a single regression problem directly uses image pixel optimization and can detect the sample bounding box location and conduct classification. In addition, the proposed method has a higher location time than that of YOLO V3. The average missed detection rate of the proposed method is approximately 1.00%. We propose a missed detection algorithm for tiny parts based on intermediate variables, which can effectively identify missed detections in industrial processes. The proposed regions extracted by the Faster-RCNN overlap with one another, and the overlapping parts become repeatedly extracted features. The feature extraction process of the Faster-RCNN increases the positioning time of the model, thereby increasing the missed detection rate. The longer model training time of the proposed method than that of YOLO V3 is caused by the numerous parameters of the SSD-based method. This problem must be solved in future studies. Defect Type 4 has a large similarity to the standard 0.8 cm darning needle; thus, the accuracy is lower.

### 3) TINY PART DEFECT DETECTION RESULTS OF THE DIFFERENT FIELDS OF VISION OF THE CAMERA AND VARIOUS SPEEDS OF THE CONVEYOR BELT

In Section relational model of detection capability coefficient of defect system and part motion velocity, we have theoretically confirmed that the defect detection capability of small parts is optimal when $W_{view}$ is 36.00 mm and the conveyor speed $v$ is 7.67 m/min. We compare our method defect detection algorithm in the test set to obtain better performance than that of the comparison algorithm in Section Comparison of proposed algorithm with classical algorithm. In this part, we combine the speed model and the TP-SSDM defect detection method to verify the actual application on the defect detection simulation platform. We set the simulation platform conveyor motor range of speed per minute $v = [0$ m/min, 10 m/min] and consider the performance of the proposed method when the actual camera' field of vision of the industrial $W_{view}$ are 32.00, 36.00, and 40.00 mm. We deploy the algorithm on a workstation with one GTX1080. The workstation connects the industrial camera on the simulation platform through the USB serial port to test the defect detection accuracy, the missed detection rate, and the average number of detections within 10 s at different conveyor speeds with 1000 defect type parts. Table 7 displays the result of an average test of 10 times.

Table 7 presents that, when the velocity is $v = 1$ m/min, the recognition accuracy of defect detection is the highest. The accuracy rates of $W_{view} = 32.00, 36.00,$ and 40.00mm are 86.87%, 93.26%, and 87.23, respectively. The average number of defect detections is the least, and the detection numbers of $W_{view} = 32.00, 36.00,$ and 40.00mm are 10, 13, and 15, correspondingly. Furthermore, the missed detection rate is the lowest, and the missing detection rates of $W_{view} = 32.00, 36.00,$ and 40.00 mm are 2.35%, 1.20%, and 2.11%, respectively. This finding is due to the slow movement of

**TABLE 7.** Tiny part defect detection results in different field of the camera' field of vision and different speeds of the conveyor belt.

| m/min | $W_{view}$=32.00mm | | | $W_{view}$=36.00mm | | | $W_{view}$=40.00mm | | |
|---|---|---|---|---|---|---|---|---|---|
| | Missing detection rate (%) | Accuracy rate (%) | detection number | Missing detection rate (%) | Accuracy rate (%) | detection number | Missing detection rate (%) | Accuracy rate (%) | detection number |
| 1 | 2.35 | 86.87 | 10 | 1.20 | 93.26 | 13 | 2.11 | 87.23 | 15 |
| 2 | 2.69 | 86.35 | 17 | 1.27 | 92.39 | 22 | 2.47 | 86.96 | 26 |
| 3 | 3.42 | 85.54 | 26 | 1.76 | 92.18 | 35 | 3.02 | 86.15 | 46 |
| 4 | 4.67 | 83.81 | 38 | 2.33 | 90.25 | 47 | 4.13 | 83.65 | 59 |
| 5 | 5.89 | 82.17 | 49 | 2.89 | 89.54 | 59 | 5.39 | 83.24 | 70 |
| 6 | 6.24 | 80.72 | 62 | 3.24 | 87.63 | 75 | 6.05 | 82.57 | 83 |
| 7 | 6.82 | 80.26 | 70 | 3.45 | 86.82 | 80 | 6.57 | 82.05 | 93 |
| **7. 67** | **7.65** | **79.64** | **78** | **3.60** | **85.50** | **91** | **7.18** | **81.72** | **104** |
| 8 | 9.16 | 76.35 | 93 | 5.32 | 83.67 | 105 | 8.79 | 79.88 | 119 |
| 9 | 11.39 | 72.57 | 107 | 7.89 | 78.93 | 118 | 10.14 | 75.32 | 137 |
| 10 | 14.18 | 66.39 | 120 | 10.53 | 72.47 | 135 | 13.32 | 69.61 | 152 |

the conveyor belt, the clearly captured sample image of the camera, the missing detection rate, the increased detection number, and the decreased accuracy rate of the algorithm. When the speed is $v = 10$ m/min, the defect detection has the lowest recognition accuracy, and the defect detection sample number and the missed detection rate are the highest. This finding is due to the number of cameras captured increases with the speed of the conveyor belt. Therefore, the number of detections rises. However, the sharpness of the picture is degraded, the missing detection rate of the proposed algorithm is increased, and the recognition accuracy is decreased.

Using the lowest conveyor belt speed not only obtains the highest defect detection accuracy but also the effect of product life cycle in the actual defect detection. Although the fastest conveyor speed increases the number of tiny parts tested, it reduces the defect detection accuracy of tiny parts. Therefore, when the conveyor speed is between $v =$ [5 and 8 m/min], the missing detection rate, accuracy rate, and detection number are acceptable ranges for the enterprise. This finding further confirms our previous theoretical derivation. When $W_{view} = 36.00$ mm and the conveyor speed $v = 7.67$ m/min, the system has the optimal defect detection capability for small parts. The missing detection rate is 3.60%. The accuracy rate is 85.50%, and the detection number is 91.

Based on the abovementioned analysis, the proposed method obtains the optimal defect detection system capability coefficient considering the factors, such as conveyor speed, part size attribute, camera' field of vision, missing detection rate, and detection number. Given the complexity of the actual working conditions and the jitter, the recognition accuracy on the simulation platform is lower in the proposed method in the test dataset in the PC.

## VII. CONCLUSION

We proposed a real-time tiny part defect detection system for manufacturing using an end-to-end CNN algorithm. This system is based on the defects of a 0.8 cm darning needle, such as crooked shapes, length and endpoint size errors, and wringing. Subsequently, we studied the correlation relationship model between the part coefficients identified by the defect detection system and the moving speed of the parts. We obtained HD detected samples and a stable tiny part data acquisition system using the optimal speed and appropriate camera position, respectively. Our study used the advanced object detection algorithm SSD, which can detect the defects of tiny parts accurately and efficiently. The proposed method is applicable to the defect detection of 0.8 cm darning needles and other tiny parts.

1) We proposed a relationship model between the detection capability coefficient of defective part systems and the movement speed of parts and obtained the best transmission belt speed of the defective part detection system. This method improves the system stability for tiny part defect detection.

2) The end-to-end defect detection model was trained based on an SSD algorithm using the defect data of a 0.8 cm darning needle. The accuracy rates of the model in Defect Types 1, 2, 3, and 4 were 98.00%, 99.00%, 97.80%, and 79.40%, respectively. Compared with the compared object detection algorithms, our proposed SSD-based tiny part defect detection algorithm is more suitable for the defect detection of 0.8 cm darning needles.

3) We built a real-time industrial detection platform and propose an algorithm for mechanical part omission detection that is based on intermediate variables. The platform can effectively detect missing tiny parts and provide a theoretical reference for the practical application of defect detection technology.

In addition, for occluded objects, the increased attention mechanism method will provide more contextual information that will provide new ideas for the automated detection of tiny part defects.

## REFERENCES

[1] M. Rezaei-Malek, M. Mohammadi, J.-Y. Dantan, A. Siadat, and R. Tavakkoli-Moghaddam, "A review on optimisation of part quality inspection planning in a multi-stage manufacturing system," *Int. J. Prod. Res.*, pp. 1–18, Apr. 2018. doi: 10.1080/00207543.2018.1464231.

[2] Q. Song, W. Ding, H. Peng, J. Gu, and J. Shuai, "Pipe defect detection with remote magnetic inspection and wavelet analysis," *Wireless Pers. Commun.*, vol. 95, no. 3, pp. 2299–2313, Aug. 2299-2313.

[3] S. W. Oh, D.-B. Yoon, G. J. Kim, J.-H. Bae, and H. S. Kim, "Acoustic data condensation to enhance pipeline leak detection," *Nucl. Eng. Des.*, vol. 327, pp. 198–211, Feb. 2018.
[4] C. Wu, S. Yao, and B. Corinne, "Leakage current study and relevant defect localization in integrated circuit failure analysis," *Microelectron. Rel.*, vol. 55, nos. 3–4, pp. 463–469, Feb. 2015.
[5] M. A. J. Bouwens, D. J. Maas, J. C. J. van der Donck, P. F. A. Alkemade, and P. van der Walle, "Enhancing re-detection efficacy of defects on blank wafers using stealth fiducial markers," *Microelectron. Eng.*, vol. 153, pp. 48–54, Mar. 2016.
[6] J.-P. Wang, Y. Wu, and T.-W. Zhao, "Short critical area model and extraction algorithm based on defect characteristics in integrated circuits," *Analog Integr. Circuits Signal Process.*, vol. 91, no. 1, pp. 83–91, Apr. 2017.
[7] T. Li, L. Gao, P. Li, and Q. Pan, "An ensemble fruit fly optimization algorithm for solving range image registration to improve quality inspection of free-form surface parts," *Inf. Sci.*, vols. 367–368, pp. 953–974, Nov. 2016.
[8] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, Jul. 2018.
[9] W. Zhou, M. Fei, H. Zhou, and K. Li, "A sparse representation based fast detection method for surface defect detection of bottle caps," *Neurocomputing*, vol. 123, pp. 406–414, Jan. 2014.
[10] F. R. López-Estrada, D. Theilliol, C. M. Astorga-Zaragoza, J. C. Ponsart, G. Valencia-Palomo, and J. Camas-Anzueto, "Fault diagnosis observer for descriptor Takagi–Sugeno systems," *Neurocomputing*, vol. 331, pp. 10–17, Feb. 2018.
[11] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief network," *IEEE Trans. Ind. Electron*, vol. 65, no. 3, pp. 2727–2736, Mar. 2018.
[12] Y. Li, D. Zhang, and D.-J. Lee, "Automatic fabric defect detection with a wide-and-compact network," *Neurocomputing*, vol. 329, pp. 329–338, Feb. 2019.
[13] J. Lei, X. Gao, Z. Feng, H. Qiu, and M. Song, "Scale insensitive and focus driven mobile screen defect detection in industry," *Neurocomputing*, vol. 294, pp. 72–81, Jun. 2018.
[14] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. Liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11–26, Apr. 2017.
[15] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.
[16] J. Yang and G. Yang, "Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer," *Algorithms*, vol. 11, no. 3, pp. 28–43, Mar. 2018.
[17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.
[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Sep. 2016, pp. 21–37.
[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2980–2988.
[21] V. Trivedi, M. Joglekar, S. Mahajan, N. Patel, V. Chhaniwal, B. Javidi, and A. Anand, "Digital holographic imaging of refractive index distributions for defect detection," *Opt. Laser Technol.*, vol. 111, pp. 439–446, Apr. 2019.
[22] Y. C. Tan, W. K. Chiu, and N. Rajic, "Quantitative defect detection on the underside of a flat plate using mobile thermal scanning," *Procedia Eng.*, vol. 188, pp. 493–498, May 2017.
[23] C. Devivier, F. Pierron, and M. R. Wisnom, "Impact damage detection in composite plates using deflectometry and the virtual fields method," *Composites A, Appl. Sci. Manuf.*, vol. 48, pp. 201–218, May 2013.
[24] Y. Shao, L. Zeng, J. Lin, W. Wu, and H. Zhang, "Trailing pulses self-focusing for ultrasonic-based damage detection in thick plates," *Mech. Syst. Signal Process.*, vol. 119, pp. 420–431, Mar. 2019.
[25] Y. Zou, D. Du, B. Chang, L. Ji, and J. Pan, "Automatic weld defect detection method based on Kalman filtering for real-time radiographic inspection of spiral pipe," *NDT E Int.*, vol. 72, pp. 1–9, Jun. 2015.
[26] S. Zhang, Q. He, K. Ouyang, and W. Xiong, "Multi-bearing weak defect detection for wayside acoustic diagnosis based on a time-varying spatial filtering rearrangement," *Mech. Syst. Signal Process.*, vol. 100, pp. 224–241, Jul. 2017.
[27] N. Boaretto and T. M. Centeno, "Automated detection of welding defects in pipelines from radiographic images DWDI," *NDT E Int.*, vol. 86, pp. 7–13, Mar. 2017.
[28] S. Hajizadeh, A. Núñez, and D. M. J. Tax, "Semi-supervised rail defect detection from imbalanced image data," *IFAC-PapersOnLine*, vol. 49, no. 3, pp. 78–83, 2016.
[29] Z. Wakaf and H. A. Jalab, "Defect detection based on extreme edge of defective region histogram," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 30, no. 1, pp. 33–40, Jan. 2018.
[30] S. S. Martínez, C. O. Vázquez, J. G. García, and J. G. Ortega, "Quality inspection of machined metal parts using an image fusion technique," *Measurement*, vol. 111, pp. 374–383, Dec. 2017.
[31] Y. Yang, Y. Lou, M. Gao, and G. Ma, "An automatic aperture detection system for LED cup based on machine vision," *Multimedia Tools Appl.*, vol. 77, no. 18, pp. 23227–23244, Sep. 2018.
[32] L. Song, X. Li, Y. Yang, X. Zhu, Q. Guo, and H. Yang, "Detection of micro-defects on metal screw surfaces based on deep convolutional neural networks," *Sensors*, vol. 18, no. 11, pp. 3709–3723, Oct. 2018.
[33] P. Wei, C. Liu, M. Liu, Y. Gao, and H. Liu, "CNN-based reference comparison method for classifying bare PCB defects," *J. Eng.*, vol. 2018, no. 16, pp. 1528–1533, Nov. 2018.
[34] J. C. P. Cheng and M. Wang, "Automated detection of sewer pipe defects in closed-circuit television images using deep learning techniques," *Automat. Construct.*, vol. 95, pp. 155–171, Nov. 2018.
[35] G. Krummenacher, C. S. Ong, S. Koller, S. Kobayashi, and J. M. Buhmann, "Wheel defect detection with machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1176–1187, Apr. 2018.
[36] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis. Eur. Conf. Comput. Vis. (ECCV)*, Cham, Switzerland, Sep. 2016, pp. 354–370.
[37] O. Oktay, E. Ferrante, K. Kamnitsas, M. Heinrich, W. Bai, J. Caballero, S. A. Cook, A. de Marvao, T. Dawes, and D. P. O Regan, "Anatomically constrained neural networks (ACNNs): Application to cardiac image enhancement and segmentation," *IEEE Trans. Med. Imag.*, vol. 37, no. 2, pp. 384–395, Feb. 2018.
[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556
[39] J. Yang, S. Li, Z. Gao, Z. Wang, and W. Liu, "Real-time recognition method for 0.8 cm darning needles and KR22 bearings based on convolution neural networks and data increase," *Appl. Sci.*, vol. 8, no. 10, pp. 1857–1875, Oct. 2018.
[40] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
[41] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Nov. 2017, pp. 2117–2125.

**JING YANG** received the B.Sc. degree from Anyang Normal University, in 2015. He is currently pursuing the Ph.D. degree with Guizhou University, China. He has received a Scholarship from the China Scholarship Council (CSC), from 2018 to 2019, under the State Scholarship Fund to pursue the degree with Oklahoma State University, as a joint Ph.D. Student with the Institute for mechatronic engineering and joined Prof. G. Fan's Group in U.S. His main research interest includes machine vision and deep learning in indoor robots and smart manufacturing applications.

**SHAOBO LI** was a Professor with the School of Mechanical Engineering, Guizhou University (GZU), China. He has been the Dean of the School of Mechanical Engineering, GZU, since 2015. He was the Vice Director of the Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, GZU, from 2007 to 2015. His research has been supported by the National Science Foundation of China (NSFC) and the National High-Tech R&D Program (863 Program). His main research interests include intelligence manufacturing and big data.

**ZHENG WANG** received the B.Sc. degree from Guizhou University, China, in 2017, where he is currently pursuing the master's degree. His main research interests include the on-line nondestructive quality inspection of mechanical products and machine learning.

**GUANCI YANG** received the Ph.D. degree in computer software and theory from the University of Chinese Academy of Sciences, in 2012. He is currently a Professor with the Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, Guizhou University, China. His current areas of interest mainly include computational intelligence, human–robot interaction, and intelligent control systems.