Neurocomputing 422 (2021) 34–50

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Automatic zipper tape defect detection using two-stage multi-scale convolutional networks

Houzhang Fang [a], Mingjiang Xia [a], Hehui Liu [b], Yi Chang [c], Liming Wang [a,*], Xiyang Liu [a]

[a] Software Engineering Institute, School of Computer Science and Technology, Xidian University, Xi'an 710071, China
[b] Nanjing Cognitive Internet of Things Research Institute, Nanjing 210001, China
[c] Artificial Intelligence Research Center, Pengcheng Laboratory, Shenzhen 518055, China

ARTICLE INFO

Article history:
Received 17 June 2020
Revised 16 August 2020
Accepted 27 September 2020
Available online 6 October 2020
Communicated by Steven Hoi

Keywords:
Automatic defect detection
Zipper tape inspection
Fully convolutional neural network
Feature fusion
Multi-scale detection

ABSTRACT

Defects inevitably occur during the manufacturing process of the zipper, significantly affecting its value. Zipper inspection is of significant importance in ensuring the quality of the zipper products. Traditional zipper inspection requires skilled inspectors and is labor-intensive, inefficient, and inaccurate. Currently, automated zipper defects inspection with high precision and high efficiency is still very challenging. In this paper, we propose a novel zipper tape defect detection framework based on fully convolutional networks in a two-stage coarse-to-fine cascade manner. For our special application, the zipper tape defects have multi-scale characteristics. Most of the existing deep learning methods have great advantages in detecting the large-scale defects with prominent features, but are prone to fail in detecting the small-scale ones due to their less remarkable features as well as their general location in a large background area. Thus, we propose to detect first the large local context regions containing the small-scale defects using a multi-scale detection architecture with high efficiency, which integrates a new detection branch by fusing the features in the shallow layer into the high-level layer to boost the detection performance of the context regions. Then we finely detect the small-scale defects from the local context regions detected in the first stage, which can be regarded as large-scale objects that are more easily detected. Extensive comparative experiments demonstrate that the proposed method offers a high detection accuracy while still having high detection efficiency compared with the state-of-the-art methods, coupled with good robustness in some complex cases.

1. Introduction

Automatic zipper defect detection plays an important role in product quality inspection of industrial automation production lines and is an urgent need for current zipper intelligent manufacturing. Its goal is to recognize and localize the defects on the surface of zipper during quality inspection. Due to the strict quality standards of the zipper products and technical challenges of the quality inspection, the current zipper quality inspection still relied mostly on subjective visual inspection by experienced inspectors, subject to poor inspection efficiency, high labor intensity and high missing and false positive rates resulting from human inexperience and fatigue. Besides, the traditional manual inspection fails to yield accurate quantitative statistical results for different types of defects, which are conducive to subsequent defects source analysis. Automated visual zipper inspection with high precision, high efficiency, and low cost is imperative for better productivity of the zipper manufacturing industry.

In this paper, we focus on plastic steel zipper tape defect detection. Typical zipper tape defects are divided into tape indentation, large-scale broken tape defects, small-scale broken tape defects, and other small-scale defects that are not included in the above defects, as shown in Fig. 1. The zipper images are acquired by a black box acquisition device that includes two industrial cameras and two auxiliary light sources. To acquire all the defects of the zipper, the zipper images are taken from both sides when the zipper goes through the middle of the two cameras. The cameras are placed on both sides inside the black box and the auxiliary light sources are installed on both sides of the transmission path.

Due to the complexity of the zipper defects and the imperfect zipper image acquisition environment, current zipper tape defect detection is faced with the following challenges. First, the tape defects have multi-scale characteristics (as shown in Fig. 1). The features of large-scale defects are obvious and thus relatively easy

* Corresponding author.
E-mail addresses: houzhangfang@xidian.edu.cn (H. Fang), liuhehui@niciot.com.cn (H. Liu), wanglm@mail.xidian.edu.cn (L. Wang), xyliu@xidian.edu.cn (X. Liu).

https://doi.org/10.1016/j.neucom.2020.09.046
0925-2312/

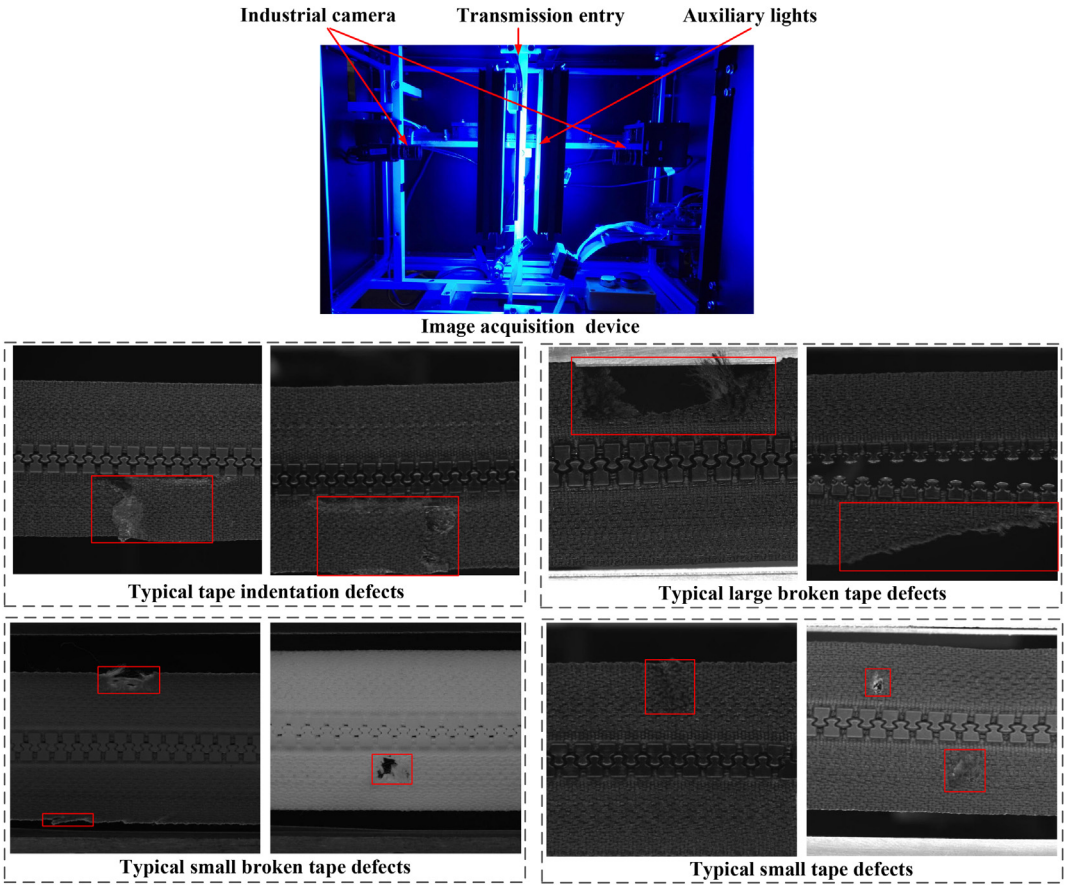


Fig. 1. Image acquisition device and several typical plastic steel zipper tape defects.

to detect. The small-scale defects, however, are very difficult to detect because the defects have no remarkable change in the features. Second, the defects are of diverse categories. There are various defect types and even a single class of defects can differ in size and position, which brings a great challenge to defect detection. Third, it is not easy to collect a large number of high quality training samples in industry, especially for some atypical defect types. Insufficient training datasets may lead to a poor generalization ability of the model. Besides, the zipper tape images are often twisted with a low contrast. Hence, how to develop a detection method with high detection accuracy and real-time performance is extremely challenging.

As far as we know, there are few works on the zipper tape defect detection up to now. Zipper defect detection can be regarded as a typical application of object detection in industry [1,2]. The most related work to our research should be the fabric defect detection [3–6], because both of the zipper tape and the fabric have the textures structures. However, these methods are hard to handle some complex situations, such as nonuniform distribution of brightness in an image, image blurring, and image twist. Hence, the above traditional methods can hardly meet the stringent requirements of zipper quality inspection. Although several deep learning based approaches [7–11] have recently been adopted to detect the fabric defects, none of them are utilized for the zipper defect detection. In this paper, we propose to adopt the state-of-the-art deep learning-based method to detect the zipper tape defects.

Due to the stringent requirements for time-efficiency and precision of the zipper production lines, an appropriate detector plays an important role in our zipper defect detection system. We adopt the state-of-the-art single-stage detector YOLOv3-SPP [12] as a basic detector for our real-time zipper quality inspection applications. However, it does not work so well when it is directly applied to our zipper tape data. The detected results for our zipper tape data using a single-stage detector YOLOv3-SPP show that the detection accuracy of the original small-scale defects is not satisfactory. To improve the detection accuracy, we introduce a new detection network branch by fusing the features from the high and shallow layer into the YOLOv3-SPP framework. Furthermore, the improved YOLOv3-SPP network performs prediction in the multi-scale feature maps from the different layers, thus capable of detecting the different scale defects. In addition, the pooling operation will weaken the features of small-scale defects in the detection network. Thus, we propose to replace the pooling operation with the dilated convolution in order to further reduce the loss of small-scale features.

Generally, the zipper defects cause local changes of intensity in an image, leading to corresponding local discontinuities in the gray values of the obtained image. Hence, in this study, we propose to first detect the larger local regions that contain the small-scale defects. The introduction of information in the contextual local regions enclosing the small-scale defects highlights the discontinuous changes between the small-scale defects and surrounding normal textures. The features of the discontinuous changes are relatively more discriminative, which are more easily captured by the CNN network. Thus, an integration of local contextual information around the small-scale defects is expected to enhance the detection accuracy of the small-scale defects.

The contributions of the proposed method are listed as follows:

- We propose a novel two-stage coarse-to-fine cascade framework to solve the zipper tape defect detection problem. In the first stage, we detect the large-scale defects and local context

regions containing small-scale defects. In the second stage, we detect small-scale defects from the local context regions detected in the first stage, which can be seen as a refining process. In this case, small-scale defects can be regarded as large-scale objects, which are more easily detected.

- We propose a new improved multi-scale network architecture in the first stage by adding a new detection branch in the high layer of the multi-scale detection network, where the shallow layer features with more information about edges and contours are fused into the high layer to further boost the detection accuracy of the local context region containing small-scale defects.
- We integrate the spatial pyramid convolution (SPC) module obtained by concatenating the spatial pyramid feature maps with different receptive fields into the medium layer detection branch in the network architectures of the first stage to further improve the detection accuracy of the medium-scale defects.
- The proposed method has a high detection speed compared with most of the state-of-the-art methods, promising in real applications. The detection accuracy in two stages is over 99.0% for all zipper tape defects on our testing dataset, comparable to those of the state-of-the-art methods.

To the best of our knowledge, this is the first study to apply the deep learning based object detection techniques to the zipper tape defect detection problem. We call the improved method in the first stage as YOLOv3++-SPC.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 introduces the system review. Section 4 details the defect detection system. Section 5 introduces the dataset and data augmentation. Section 6 describes the experiments and results. Finally, some conclusions are drawn in Section 7.

2. Related work

In this section, we elaborate the most related work to our research. We first review the conventional approaches of detecting the fabric defects. Then, we introduce the deep learning based methods that have been applied to the fabric defects detection. Finally, we introduce the general deep learning based object detection techniques.

Over the years, the commonly used traditional methods for fabric defect detection can be roughly classified into four categories: statistical [13], spectral [14,6], model-based [15,16] and learning approaches [17,18]. For the statistical methods, the fabric features are characterized by the grayscale distributions of image regions. Kuo et al. [13] proposed a statistical approach by employing the co-occurrence matrix to extract features and the gray relational analysis to study the correlation of the analyzed factors from a randomized factor sequence. The idea of the spectral analysis methods is to convert the fabric image into the spectral domain using an appropriate transform, e.g, wavelet transform [14] or contourlet transform [6]. The major drawback of these methods is that the transform basis is fixed. The representative model-based method is to decompose a fabric image into the repeated patterns (e.g. texture) and defective objects. The repeated patterns can be represented by the low-rank component [16] or p-norm [15] and the defective parts are characterized by the total variation norm [15] or sparse constraint [16]. The model-based methods enjoy good performance for the periodic patterned fabric data. However, they may still not be able to handle some complex situations effectively, such as when defects of smaller scale are involved. For the machine learning methods [17,18], the key is to choose appropriate handcrafted feature descriptors for the fabric samples. The feature descriptors may be not robust to some unexpected changes, such as nonuniform distribution of brightness in an image and image blurring, which will impair the performance of the method.

Because of the successes of deep learning in computer vision, deep learningbased object detection methods have been employed in recent years for the detection of the fabric defects [7–11]. Mei et al. [7,8] constructed a Gaussian pyramid-based convolutional denoising autoencoder networks architecture to distinguish defective and defect-free regions, which can achieve a satisfactory performance for the fabric with homogeneous and nonregular textured surface in an unsupervised manner. The detection method in [7,8] consists of image preprocessing, patch extraction, model application, and threshold determination. Some important parameters are very sensitive in the above sub-procedures. Hence, the methods developed here will be challenging in automatically dealing with the complex zipper defects. Ouyang et al. [9] presented a convolutional neural network (CNN)-based hybrid method for the fabric defect detection. This method first adopted a traditional autocorrelation statistical rule to generate a candidate defect probability map, which is introduced as prior knowledge of defects in the CNN defect detection. The method is effective in detecting the fabric with regular textured structures. Jing et al. in [10] proposed a CNN based fabric defect detection method, which is composed of fabric image local patches decomposition, transfer learning, and defects detection. Hu et al. in [11] presented a convolutional generative adversarial network for the fabric defects detection. Both the methods presented in [10,11] need to process the local patch image obtained in a sliding way over the whole image, which thus reduce the efficiency of the algorithm.

Recently, other deep learning based generic object detection approaches also achieves great progress in detection accuracy and efficiency [1,2]. These methods can be mainly divided to two categories: region proposal based two-stage detectors, e.g., Faster R-CNN [19] and Cascade R-CNN [20]; and regression based single-stage detectors, e.g., YOLO [21] and SSD [22]. The two-stage method first utilized a region proposal network (RPN) to generate regions of interests and then fed the region proposals into object classification and bounding-box regression. The single-stage method directly predicts class probabilities and bounding box offsets from the entire feature maps. In general, RPN based two-stage approaches have a high detection accuracy, but are computationally expensive. The single-stage approaches have a high detection speed but the accuracy trails that of two-stage methods. The later work, such as YOLOv2 [23], YOLOv3-SPP [12],[1] and RetinaNet [24], has made efforts to improve the detection accuracy of the single-stage method.

3. System overview

The motivation of the proposed detection system is to automatically detect the zipper tape defects and finally output the class and position of each defect. The whole defect detection system involves two main modules: (1) local context regions and large-scale defects detection, (2) small-scale defects detection and location mapping. Fig. 2 presents the pipeline of the detection system. Overview of the two stages is as follows.

3.1. Detection of local context regions and large-scale defects (Stage 1)

The aim of this stage is to detect the local context regions with the small-scale defects and the large-scale defects in the zipper tape images captured by the industrial cameras. The large-scale defects are relatively easy to detect. The features of the small-

[1] The version YOLOv3-SPP employed in this paper is an improved version of YOLOv3. The SPP denotes the spatial pyramid pooling.

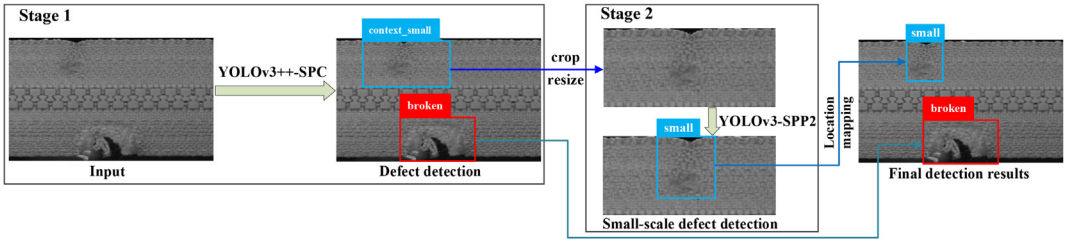


Fig. 2. The overview of the proposed zipper defect detection system. We first detect the local context regions and the large-scale defects in Stage 1, and then detect the small-scale defects from the local context regions in Stage 2. Finally, we map the location information of the small-scale defects and the large-scale tape defects into the original zipper image.

scale zipper tape defects are less discriminative, for which it is very difficult to train a robust detector. Thus, in Stage 1, the local context regions with the small-scale defects are first detected using an improved multiple-scale detection network (YOLOv3++-SPC) where the speed and accuracy are performed well. The position of the regions of the small-scale defects in this stage is coarsely localized.

3.2. Detection of small-scale defects (Stage 2)

The background being greatly reduced in the local context regions containing the small-scale defects detected from Stage 1, small-scale defects can be seen as the large-scale objects compared with the local context regions. The defects detection at this stage is a relatively easy task, thus an improved YOLOv3-SPP2 network is employed.

Finally, after the small-scale defects is detected in the local context regions, the real positions of the small-scale defects in the original zipper tape image can be determined by applying the location information of the context region in the original image.

4. The defect detection system

The proposed detection system contains two stages, which is composed mainly of two defects detection networks. In this section, we will introduce the two detection networks in detail.

4.1. Detection of the local context regions and large-scale defects using YOLOv3++-SPC

The zipper tape contains the large-scale and small-scale defects due to the randomness of the defects. While the large-scale defects can be detected well because of their prominent features, it is a challenging task to directly detect the small-scale defects in the original large tape images. In this study, we propose to detect the small-scale defects using a two-stage cascading way. We first coarsely detect the larger context regions around the small-scale defects in the original large tape images and then finely detect the small-scale defects from the context regions that contain the small-scale defects.

To detect the large-scale defects and the context regions containing the small-scale defects in the original large tape images, an improved YOLOv3 framework that performs well in terms of both speed and accuracy is adopted. Fig. 3 presents the improved architectures of the YOLOv3++-SPC framework. We also give all the detailed specifications of the YOLOv3++-SPC framework in Table A.7. YOLO series algorithms (e.g., YOLOv1 [21], YOLOv2 [23], YOLOv3-SPP [12]) have been employed in many fields, such as sewer pipes defect detection [25], real-time behavior detection and judgment of egg breeders [26], and wind turbine blades defects inspection [27]. However, it has not previously been applied to the zipper tape defect detection problem. The YOLOv3-SPP framework has the capability to detect the multi-scale defects and can achieve very high detection speed compared with the region-based objects detection algorithms (such as, R-CNN, Fast R-CNN, Faster R-CNN [1,2]). One outstanding characteristic of YOLO series that is different from R-CNN series is that it utilizes the pre-defined grid cell to execute the prediction, which significantly boosts the detection speed, making real-time object detection possible by this algorithm.

The original YOLOv3-SPP framework has three detection branches in three corresponding three scales, extracting features from these scales using a hybrid approach by invoking a similar concept to feature pyramid networks. Specifically, the features maps of 19×19 are suitable for large-scale defects detection (Scale 1 in Fig. 3), the 38×38 for medium-scale size (Scale 2 in Fig. 3), and the 76×76 for small-scale size (Scale 3 in Fig. 3). To further improve the localization performance of the context regions containing the small-scale defects, we add a new detection branch in the 152×152 Scale 4 of the high layer to fuse low-level cues and high-level semantic information for detection (Scale 4 in Fig. 3).

The improved YOLOv3++-SPC framework has 132 layers in total including 90 convolutional layers together with residual blocks, detection layers, and upsampling layers. The improved framework predicts boxes at four different scales (i.e., $19 \times 19, 38 \times 38, 76 \times 76, 152 \times 152$) in order to efficiently detect the defects of different scales. For each predicted bounding box in the feature map, the number of the predicted attributes is equal to $(5 + C) \times B = 27$, where 5 denotes the objectness score and the four box center coordinations (x, y, w, h). C is the number of class of the predicted defects. We need to predict four types of defects in this stage, so $C = 4$. B is the number of the predicted bounding boxes for each grid cell and $B = 3$ in the YOLOv3++-SPC model. For each grid cell, the total number of the predicted bounding boxes for each image is $(19 \times 19 + 38 \times 38 + 76 \times 76 + 152 \times 152) \times 3 = 528$ million. However, only one bounding box needs to be output for a defect in each image and the final results may be multiple bounding boxes of 528 million bounding boxes corresponding to the same defect. Two steps are utilized to remove the redundant predicted bounding boxes [12,25]. First the bounding boxes with the objectness confidence under a predefined threshold are ignored and then a non-maximum suppression [28] is used to choose the final bounding boxes for the multiple detections. In the detection layer, YOLOv3++-SPC uses the independent logistic regression classifier instead of the softmax function to perform the classification of the detected defects.

The YOLOv3++-SPC framework needs to obtain the feature maps at different scales so as to detect the multi-scale defects in these feature maps. Thus, downsampling processes are adopted to resize the images. The downsampling operations in the backbone network Darknet53 are conducted using the convolution implementation. Specifically, the last convolution operations in the layers 1–2, 3–6, 7–13, 14–38, and 39–63 of the Table A.7

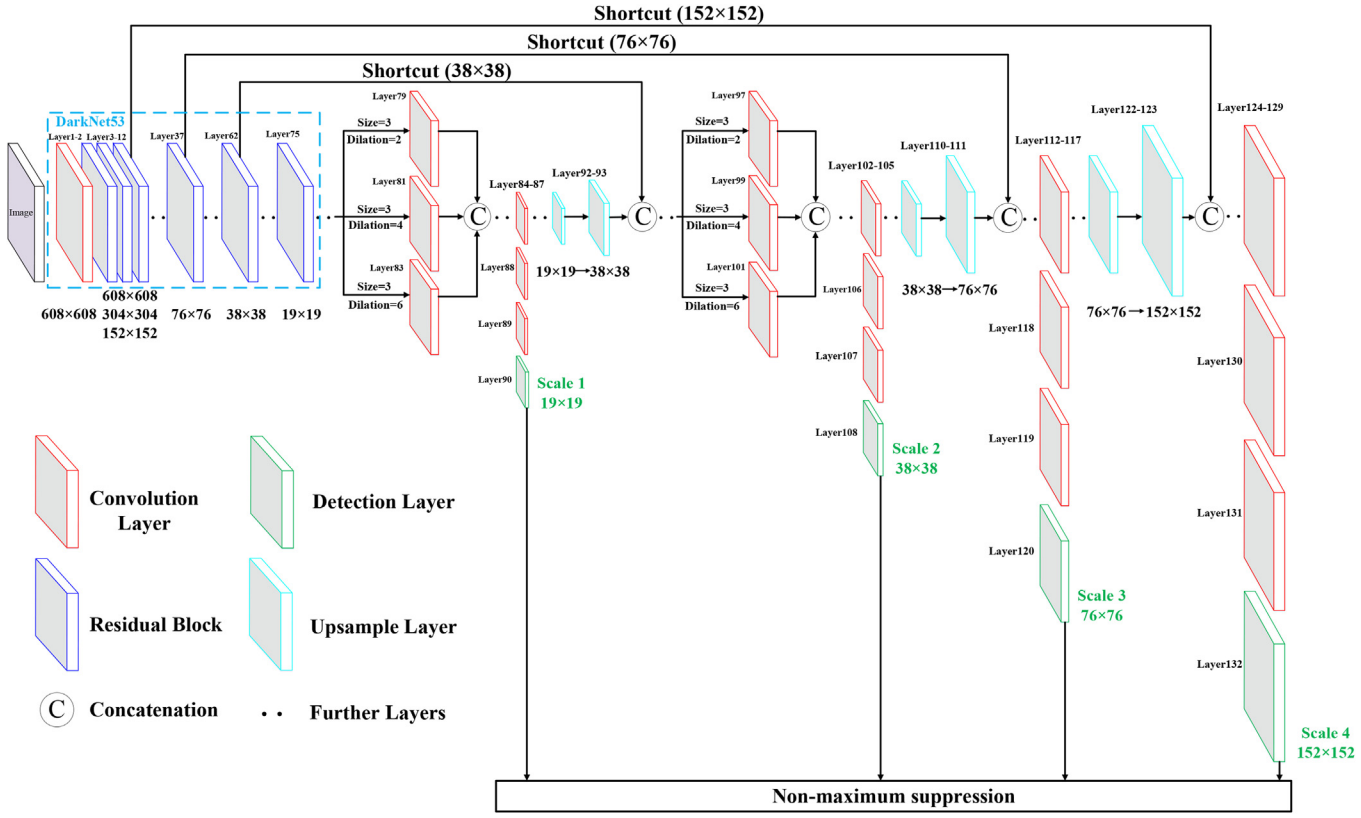


Fig. 3. The architecture of the proposed multi-scale YOLOv3++-SPC network. To better detect the small-scale defect, we add a new detection branch in the high layer (Scale 4) and utilize the dilated convolution to replace the pooling operation in the SPP feature fusion module.

perform the downsampling by changing the stride of the filters. It should be avoided to reduce too many features of the small-scale defects by carrying out the downsampling operations using the convolution instead of the pooling, which is conductive to detecting the small-scale defects.

4.1.1. SPC module

A network that detects small-scale defects requires that the fine structure features be preserved as much as possible, hence the pooling operations should be avoided in the CNN network. Generally, the spatial pyramid pooling model [29] with different receptive fields is employed to detect the objects of different scales. The SPP module in the original YOLOv3-SPP is adopted to extract multiscale features and separate out the most important context features with different receptive fields and fuses these deep features by concatenating them in the depth direction of the feature maps. The introduction of the SPP module is expected to boost the detection accuracy of the multi-scale objects. To reduce the loss of small-scale features, the pooling operation in the SPP module is replaced with the dilated convolution.

The original YOLOv3-SPP framework has a SPP module (Fig. 4 (a)) in the 19×19 detection branch. We propose an improved SPC module (Fig. 4(b)) in the YOLOv3++-SPC framework and add the module to the 38×38 detection branch for enhancing the detection performance of the medium scale defects. The SPP module in the original YOLOv3-SPP concatenate four feature maps with receptive fields of four different sizes to improve the detection performance of the medium and large scale objects. In this study, we adopt the dilated convolution to replace the pooling operation for better preserving the fine structure information. Specifically, we achieve three effective receptive field sizes of $5 \times 5, 9 \times 9$, and 13×13 with dilation factors 2, 4, and 6, respectively, which

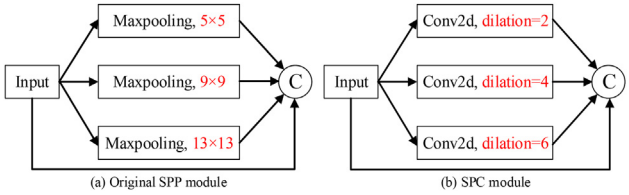


Fig. 4. Two concatenation modules. (a) Original SPP module. (b) Proposed SPC module.

correspond to three pooling kernels $5 \times 5, 9 \times 9$, and 13×13 in the original YOLOv3-SPP. Then, we concatenate the input feature map and three feature maps obtained by performing three dilation convolutions with the input feature map.

4.1.2. Training procedure

The training of the YOLOv3++-SPC generally consists of two steps: supervised pre-training and domain-specific fine-tuning. The supervised pre-training from scratch is carried out on the ImageNet data. During the domain-specific fine-tuning, the weights of the proposed network are initialized by the pre-trained model and then trained with our zipper domain-specific data. The zipper tape domain-specific data in this stage are the labelled original zipper data, where the labelled information contains the large-scale tape defects and local context regions. All training images for the proposed network are scaled to the required resolution (i.e. 608×608 in the paper). Both the positive and negative examples are sampled from the bounding boxes according to certain overlap thresholds.

To improve the generalization ability of the proposed method, several data augmentation methods (including horizontal, vertical,

diagonal flips, random brightness, and random skew) are employed to expand the number of the training samples. We need to ensure that the expanded data with data augmentation is very consistent with the original data; otherwise data augmentation can degrade the performance of the models. The data augmentation technique as an implicit regularization has been verified to increases the robustness of the CNN networks [30].

4.2. Detection of the small-scale defects using YOLOv3-SPP2

After the large-scale defects are detected in Stage 1, the classes and locations of the defects are output. The small-scale defects are finely detected from the local context regions obtained from Stage 1. In this stage, small-scale defects can be seen as the medium-scale and large-scale objects in the local context regions. To recognize and localize the medium-scale defects in this stage more accurately, an SPP module is also incorporated into the medium-level layer of YOLOv3-SPP. The SPP module can fuse the features with different receptive field sizes without the need for additional model parameters while hardly slowing down the network operation. Thus, the YOLOv3-SPP2 framework can perform well in terms of both speed and accuracy.

The training of the YOLO-SPP2 also consists of two steps: supervised pre-training and domain-specific fine-tuning. The supervised pre-training is executed on the ImageNet data. In the domain-specific fine-tuning, the network weights are initialized with the pre-trained model and then trained with our small-scale zipper domain-specific data. The zipper tape domain-specific data in this stage are the labelled local context region images, where the labelled information contains the small-scale defects.

5. Dataset

5.1. Preparation of the dataset

In this section, we describe the dataset in detail. All of the zipper data is obtained from the real zipper production line. The size of each zipper image acquired is 560×1760. We annotate each image based on the format of the PASCAL VOC dataset [31] using a specialized label tool. We first manually annotate each defect with a possibly small box. As shown in Table 1, the number of defective samples is 918 images in total. We find that more than 45% of the defects have an area smaller than 120×200. This implies that the proportion of the defective samples with a small scale is relatively large. Hence, in our experiments, the zipper tape defects with a height less than 120 and a width less than 200 are deemed small-scale defects, which will be reannotated with the larger local context regions. The other tape defects are classified as large-scale defects. The data partition for two stages are presented in Table 1. In Table 1, the categories of the defects are as follows: the "large broken tape" and "tape indentation" are the large-scale defects in the first stage. The "context_broken" denotes the smaller scale broken tape defects and "context_small" represents the small-scale tape defects that do not fall into the above categories in the first stage. The "broken tape" and "small tape" are the small-scale defects in the first stage and regarded as the large-scale defects in the second stage.

5.2. Data augmentation

It is usually harsh and impracticable to collect a large number of the defective samples for training a CNN network. It is much easier to collect the normal zipper images in our case compared to acquiring zippers with the defects. The models trained on the small datasets can cause overfitting in the network. Some data augmentation techniques are able to increase the number of the samples, which can achieve a significant boost to the model performance especially on small datasets and help prevent models from overfitting the training data. It is necessary for the data with augmentation techniques to follow the same distribution as the original training data. In our case, the horizontal, vertical, diagonal flips, random brightness, and random skew are the effective data augmentation techniques. It can be observed from Table 1 that the total number of the expanded image dataset after data augmentation in Stage 1 is 11201, which is divided into three parts: the training set with 9567 images, the validation set with 798 images, and the testing set with 836 images. The total number of the expanded dataset in Stage 2 is 7150 and is also split into three parts: the training set, the validation set, and the testing set, with 5930, 800, and 420 images, respectively.

6. Experimental results and discussion

To evaluate the capability of the proposed method, we firstly introduce the evaluation metrics. And then, we give the training process, convergence evaluation, and effectiveness of the context information for detection. Next, a series of experiments are performed for comparison state-of-the-art methods to evaluate the method in terms of the average precision (AP) and the processing time costs (in seconds). Finally, we also discuss the robustness of the proposed method.

6.1. Experimental configuration

The experimental configuration of all the deep CNNs in this paper is as follows: Deep learning framework PyTorch, Linux Ubuntu 16.04 operating system, Intel Xeon Silver 4110 CPU, two NVIDIA RTX-2080 Ti with GPU memory of 10 GB, NVIDIA CUDA10, and 157G RAM.

6.2. Evaluation metrics

Precision (P) and recall (R) are the two indicators commonly used for most deep learning models. The average precision (AP) for each class is used as the quantitative measure. The mean average precision (mAP) for all classes and F_1 score are used to measure the performance of the proposed algorithm, which are calculated based on the relationship P(R) of precision (P) and recall (R). They are computed as follows:

$$\text{Precision(P)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\%, \tag{1}$$

$$\text{Recall(R)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\%, \tag{2}$$

$$\text{AP} = \int_0^1 \text{P(R)dR}, \tag{3}$$

$$\text{mAP} = \frac{1}{N_{\text{class}}} \int_0^1 \text{P(R)dR}, \tag{4}$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{5}$$

where true positive (TP) is the number of ground-truth defects that are exactly predicted as defects; false positives (FP) indicates the number of non-defect that are predicted as defects; false negatives (FN) denotes the number of ground-truth defects that are predicted as non-defect, which implies the defects are not detected by the model. High precision means more defects are detected correctly

Table 1
Total number of the defects images before and after data augmentation for each stage.

Stage	Defects type	Training		Validation		Testing	
		Before	After	Before	After	Before	After
Stage 1	Large broken tape	46	2371	32	231	32	239
	Tape indentation	325	3000	33	250	33	250
	Context_broken	98	1380	31	79	19	117
	Context_small	205	2816	32	238	32	230
Stage 2	Broken tape	98	2965	31	400	19	127
	Small tape	316	2965	43	400	42	293

while high recall indicates fewer defects are missed during the detection. The AP is adopted to compute the average precision for different levels of recall (0 to 1) for a specific class of object, while the mAP is the average of all the classes. F_1 is utilized to evaluate the performance of the classifier by computing the harmonic mean of precision and recall, which has a range of values between 0 and 1.

6.3. Training process and convergence evaluation

The setting of the training hyperparameters in the proposed method is shown in Table 2. In Stage 1, the training epoch is set to 140 and the momentum is 0.8. The initial learning rate is set to 0.002324 for speeding up convergence, and then is multiplied by 0.1 between epochs 112 and 126. Finally, the learning rate stays at 0.00002324 between epochs 127 and 140. In Stage 2, the epoch and momentum are 180 and 0.9, respectively. The learning rate is initialized to 0.0001, and then turns to 0.00001 after 144 iterations. From the 162th iteration to the end, the learning rate is fixed at 0.000001. The batch size in Stage 1 is smaller than in Stage 2, as the image size in Stage 1 is larger. In our method, we also generate the anchor frames with different sizes by k-means clustering and apply them to the detection layers at different scales. Table 3 shows the details. Our experiments show that these anchors can be adopted to detect the defects with different scales well.

We present the convergence of the training processes of the proposed method for the Stage 1 and Stage 2 in Fig. 5. It can be seen that the change of the IoU, confidence, and classification loss functions with the number of epochs finally tends to become stable, which means that the proposed method in the first and second stages is showing good convergence during the training.

Fig. 6 illustrates the evolution of the four measurement indexes of precision, recall, mAP, and F_1 along with the first 140 and 180 epochs in Stage 1 and Stage 2 during the training processes, respectively. It can be observed clearly that the proposed method achieves good convergence during the training and the fluctuations eventually tend to be relatively stable after about 130 epochs for Stage 1 and about 170 epochs for Stage 2.

6.4. Effectiveness of the introduction of the context information for detection

In this subsection, we present a comparative experiment to verify the effectiveness of the introduction of the context information for the zipper tape defect detection, as shown in Table 4. The YOLOv3-SPP is capable of detecting multi-scale tape defects and achieves high detection accuracy and speed compared with some commonly used baseline methods. It is seen from Table 4 that the YOLOv3-SPP obtain high AP for the large broken tape defects and indentation defects, however, AP of small scale tape defects is relatively low. The main reason is that features of small-scale defects are not obvious compared with those of large-scale defects. We re-label the small-scale zipper tape defects by adding the surrounding region of small-scale defects into the label boxes to highlight the discontinuous changes between the small-scale defects and surrounding normal textures. We can see from the right-hand side of Table 4 that the YOLOv3-SPP has a remarkable improvement in AP for small-scale defects. This implies that the introduction of the context information improves the detection accuracy of the small-scale defects. We also note that AP of the indentation defects detection using YOLOv3-SPP shows a slight reduction. This is due to the fact that the scales of the current indentation defects label boxes are relatively smaller than those of small-scale defects after relabeling, which affects the detection accuracy of the indentation defects. This problem is overcome by the YOLOv3++-SPC. It is noted from Table 4 that the proposed YOLOv3++-SPC has a consistent obvious improvement for all defects.

The above conclusion is also observed in the detected images shown in Fig. 7. The accuracy for small-scale defects without context is typically low. Specially, a small-scale defect is not detected in the last row of the first column. The YOLOv3-SPP and YOLOv3++-SPC consistently obtain high detection accuracy in the small-scale defects labelled with context information.

Table 2
The setting of the training hyperparameters on Stage 1 and 2.

Hyperparameter name	Value	
	Stage 1	Stage 2
Epoch	140	180
Batch size	8	16
Initial learning rate	0.002324	0.0001
Momentum value	0.8	0.9
Weight decay	0.0009	0.0004569
IoU threshold	0.5	0.5

Table 3
The scales of the feature maps and the corresponding sizes of the anchors for two stages.

Method	Scale	Anchors
YOLOv3++-SPC	19×19	(145.5, 56.5), (170.0, 51.0), (233.8, 53.8)
	38×38	(108.9, 56.4), (120.9, 55.7), (133.3, 56.5)
	76×76	(68.9, 55.7), (81.8, 51.6), (96.0, 55.3)
	152×152	(37.1, 57.4), (48.6, 55.1), (57.4, 61.1)
YOLOv3-SPP2	13×13	(227.4, 160.7), (300.2, 154.1), (224.4, 237.7)
	26×26	(182.8, 114.6), (220.8, 115.8), (182.8, 179.3)
	52×52	(121.8, 97.0), (155.1, 99.0), (151.9, 134.7)

6.5. Comparison with state-of-the-art methods

6.5.1. Comparison with state-of-the-art methods in each stage

To show the advantages of our method, we compare the performance of the proposed method with the state-of-the-art methods, i.e., Faster R-CNN1 [32] (Faster R-CNN with the feature pyramid networks (FPN) and backbone ResNet-50 is a 50 layers residual network), Faster R-CNN2 [32] (Faster R-CNN with the FPN and

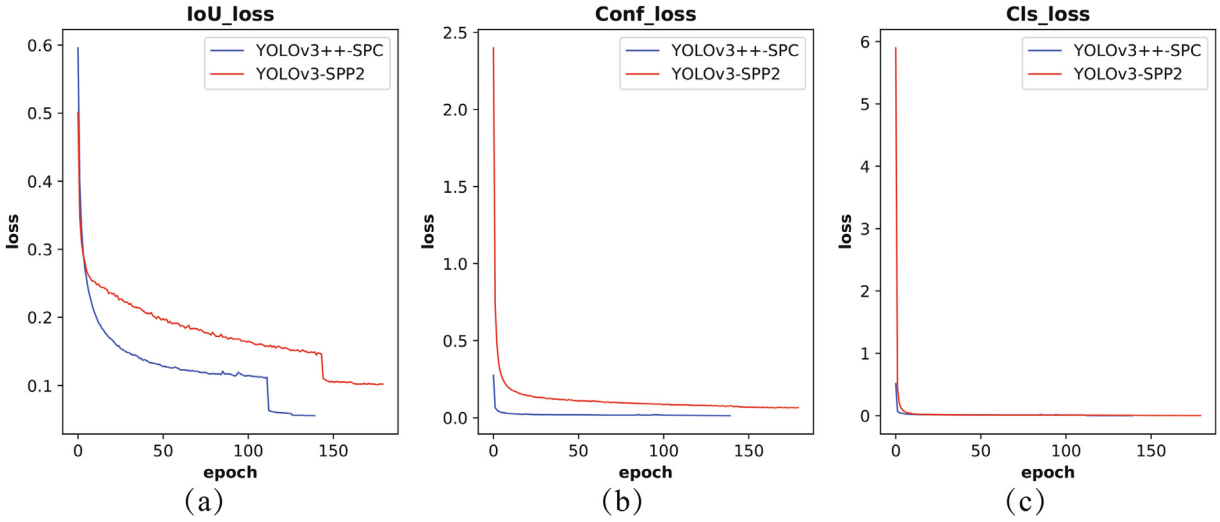


Fig. 5. Evolution curves of the loss function versus the number of epochs in the model training of the Stage 1 and Stage 2. (a) IoU loss, (b) confidence loss, and (c) classification loss.

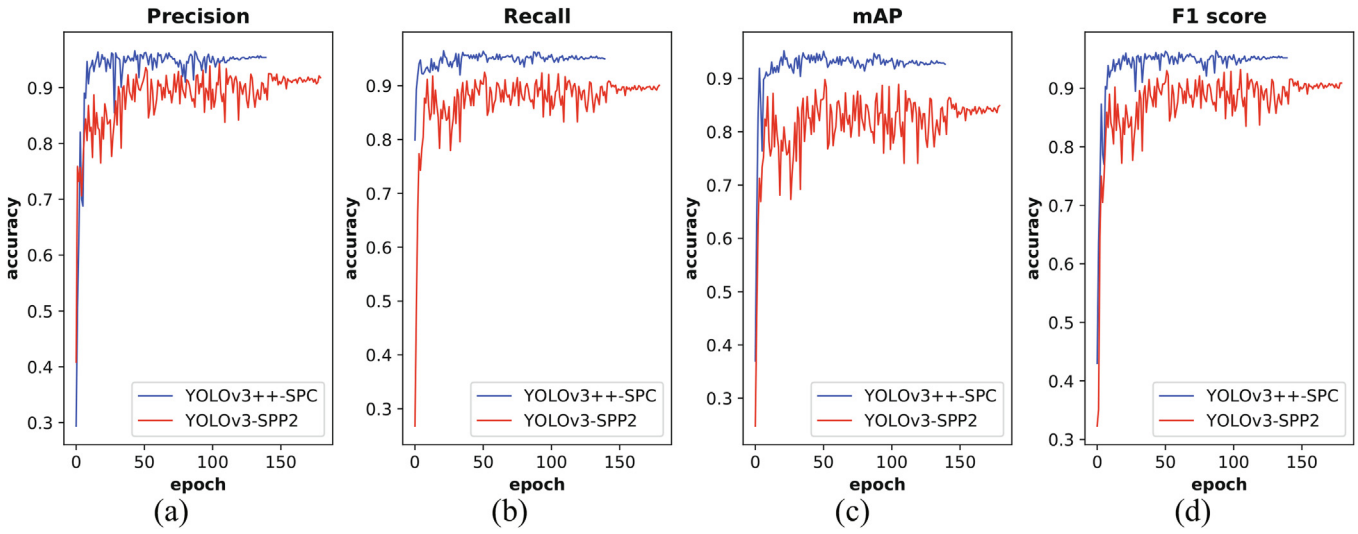


Fig. 6. Evolution curves of four measurement indexes versus the number of epochs in Stage 1 and Stage 2. (a) precision, (b) recall, (c) mAP, and (d) F_1 score.

Table 4
Evaluation results of two methods for the defects labels without and with context using the indexes of AP and mAP.

Method	Without context				With context				
	Large broken	Indentation	Small	mAP	Large broken	Indentation	Context_ broken	Context_ small	mAP
YOLOv3-SPP	0.9595	0.9850	0.9042	0.9482	0.9861	0.9761	0.9982	0.9900	0.9876
YOLOv3++-SPC	–	–	–	–	0.9947	0.9969	0.9993	0.9968	0.9969

backbone ResNet-101 is a 101 layers residual network), Cascade R-CNN1 [20] (Faster R-CNN with the cascade architecture and backbone ResNet-50), Cascade R-CNN2 [20] (Faster R-CNN with the cascade architecture and backbone ResNet-101), RetinaNet1 [24] (with the FPN and backbone ResNet-50), RetinaNet2 [24] (with the FPN and backbone ResNet-101), SSD [22], TridentNet [33], YOLOv3-SPP [12]. These methods are trained and tested using the same images. The parameters in each detection method are tuned for the best performance.

Detailed in Table 5 are the detection results of the proposed method in terms of the AP, mAP, and time consumption for the large-scale defects and local context regions containing the small-scale defects in Stage 1 and small-scale defects in Stage 2 in comparison with several state-of-the-art methods. We compare the proposed methods with several widely used detectors such as two-stage detectors and one-stage detectors. The time consumed for each method is computed based on the whole testing set. As shown in Table 5, for the large-scale broken and indentation defects in Stage 1, the AP values of our method are 0.9947 and 0.9969, respectively, and the best AP values are 0.9990 and 0.9992 obtained by the RetinaNet1 (RetinaNet2) and Cascade R-CNN2, respectively. For the context_broken defects, the AP of our method is higher than those of several other methods. The main reason is that our method YOLOv3++-SPC integrates a new

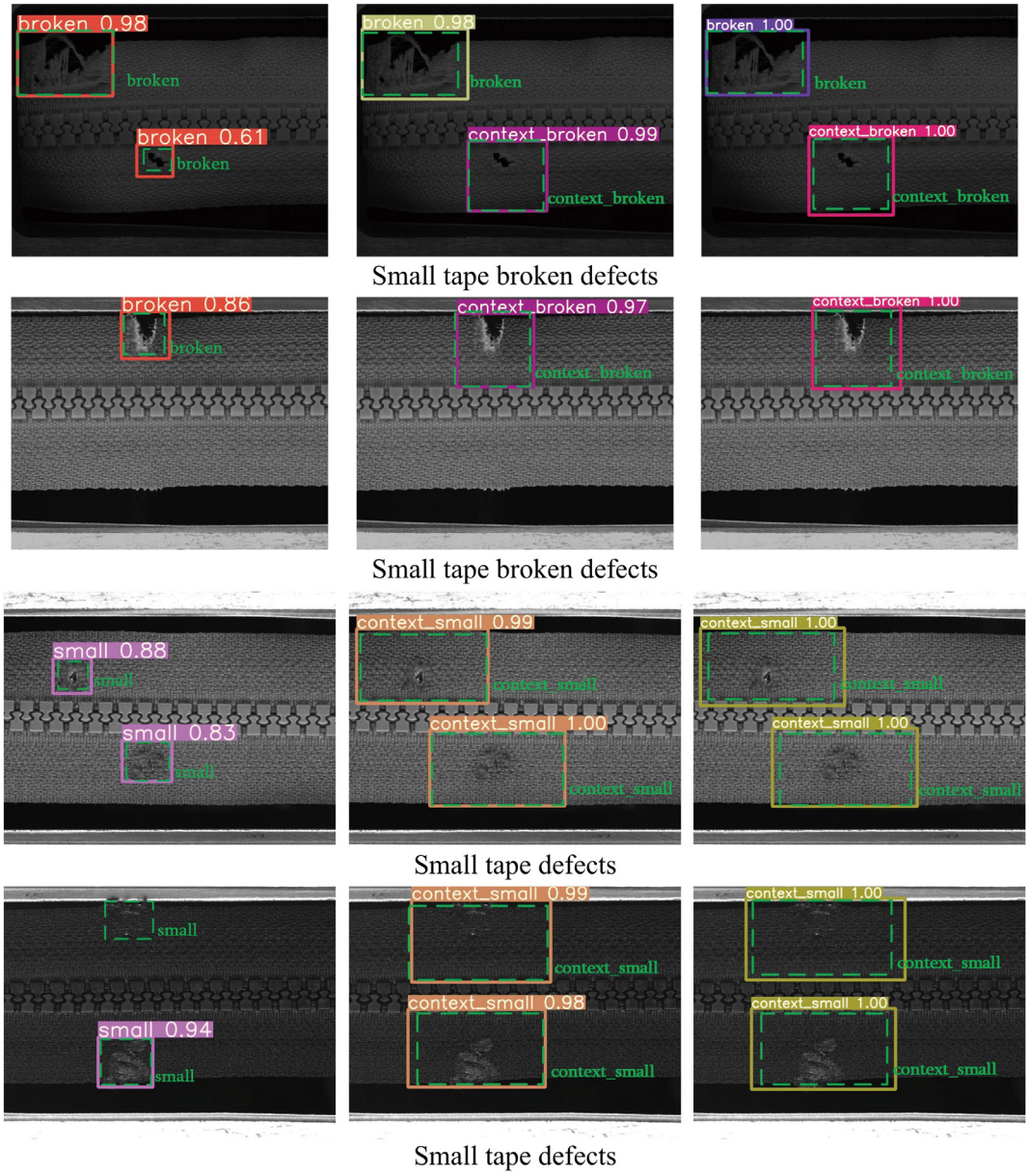


Fig. 7. The detected results of the YOLOv3-SPP and YOLOv3++-SPC for small-scale defects labels without and with context information. First column: detection using YOLOv3-SPP for the defects labels without context, second column: detection using YOLOv3-SPP for the defects labels with context, third column: detection using YOLOv3++-SPC for the defects labels with context.

Table 5
Evaluation results of four defects in Stage 1 and two defects in Stage 2 in comparison with state-of-the-art methods using the indexes of AP, mAP, and time.

Method	Stage 1						Stage 2			
	Large broken	Indentation	Context_ broken	Context_ small	mAP	Time (s)	Small broken	Small	mAP	Time (s)
Faster R-CNN1	0.9942	0.9674	0.9975	0.9955	0.9887	133.832	0.9984	0.9761	0.9873	62.375
Faster R-CNN2	0.9982	0.9688	0.9938	0.9953	0.9865	144.405	0.9906	0.9442	0.9674	73.612
Cascade R-CNN1	0.9954	0.9719	0.9984	**0.9988**	0.9911	142.701	0.9557	0.9120	0.9339	75.085
Cascade R-CNN2	0.9952	**0.9992**	0.9976	0.9982	**0.9975**	162.142	**1.0**	0.9556	0.9778	75.821
RetinaNet1	**0.9990**	0.9948	0.9819	0.9899	0.9914	176.202	0.9991	0.9539	0.9765	68.409
RetinaNet2	**0.9990**	0.9834	0.9856	0.9967	0.9912	137.972	0.9925	0.9777	0.9851	62.283
SSD	0.9948	0.9991	0.9057	0.9365	0.9590	76.539	0.9821	0.9006	0.9414	17.669
TridentNet	0.9952	0.9820	0.9970	0.9862	0.9901	258.84	0.9950	0.9606	0.9778	129.423
YOLOv3-SPP	0.9861	0.9761	0.9982	0.9900	0.9876	**14.310**	0.9734	0.9793	0.9763	**7.542**
YOLOv3++-SPC	0.9947	0.9969	**0.9993**	0.9968	0.9969	15.035	–	–	–	–
YOLOv3-SPP2	–	–	–	–	–	–	0.9998	**0.9968**	**0.9983**	7.652

Table 6
The total mAP and total time consumption on the cascaded network.

Method	mAP	time (s)
Faster R-CNN1	0.9761	196.207
Faster R-CNN2	0.9640	218.017
Cascade R-CNN1	0.9256	217.786
Cascade R-CNN2	0.9754	237.954
RetinaNet1	0.9681	244.611
RetinaNet2	0.9764	200.255
SSD	0.9028	94.208
TridentNet	0.9681	388.26
YOLOv3-SPP	0.9643	**21.852**
Our framework	**0.9952**	22.687

detection branch in the low-level layer, conducive to detecting small-scale defects. For the Context_small defects, the AP of our method is 0.9968 and the highest AP is 0.9988 by Cascade R-CNN1. The mAP of our method for four kinds of defects in Stage 1 is 0.9969 and the highest mAP is 0.9975 achieved by Cascade R-CNN2. These results in Stage 1 show that the proposed method has a slight difference with other state-of-the-art methods in terms of AP and mAP. On the other hand, our method consistently achieves a higher mAP score than 0.99 when measured against four defects and obtains a competitive advantage in time consumption over other methods. The Faster R-CNN1, Faster R-CNN2, Cascade R-CNN1, Cascade R-CNN2, and TridentNet are two-stage detectors. In these methods, the region proposal network (RPN) will take the time to generate the region proposals, which further increase the computational time. The RetinaNet1, RetinaNet2, SSD, and YOLOv3-SPP are one-stage detectors, which take less time than two-stage detectors. In the consumption of time, our method is about 15.035 s, which is slightly higher than YOLOv3-SPP and is significantly lower than other methods. This implies that the proposed method will take 17.98 ms for each test image with the size of 608 × 608 pixels, this highlights the advantage of the proposed method for the requirements of real-time applications.

The detection results in Stage 2 for several methods are shown on the right-hand side of Table 5. For the small-scale broken defects, the AP of our method is 0.9998 and the highest AP is 1 achieved by Cascade R-CNN2. For other small-scale defects, they

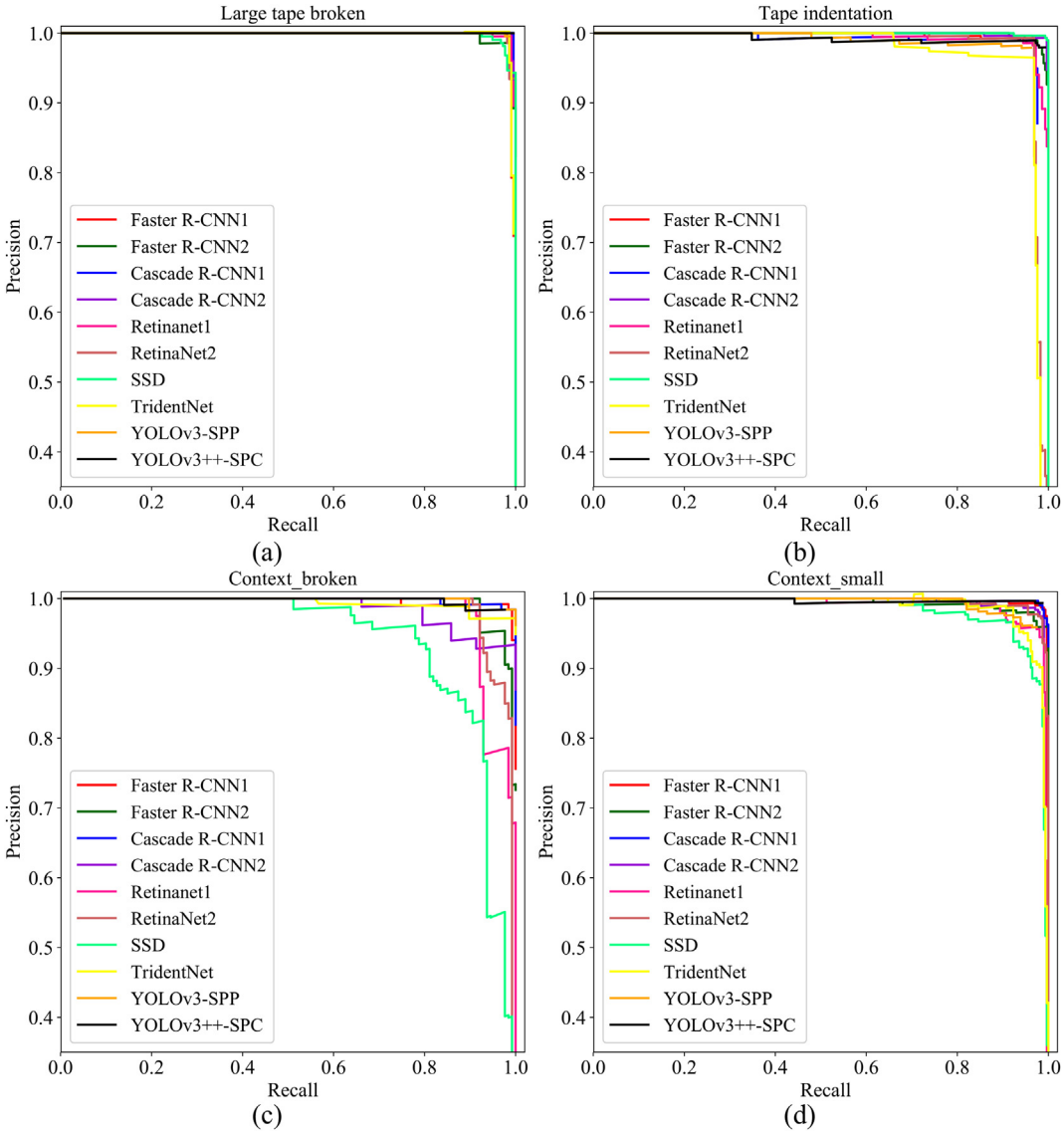


Fig. 8. Precision-recall curves for each class using the different detection models in Stage 1. (a), (b), (c), and (d) are the results of the large-scale broken tape defects, tape indentation defects, context region context_broken defects, and context region context_small defects, respectively.

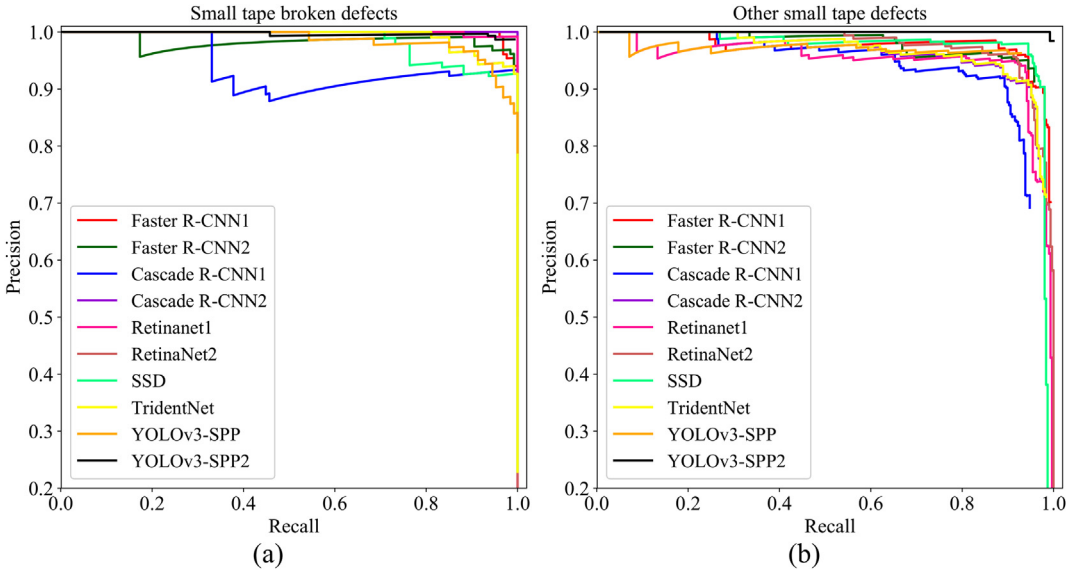


Fig. 9. Precision-recall curves for each class using the different detection models in Stage 2. (a) and (b) are the results of the small tape broken defects and other small tape defects, respectively.

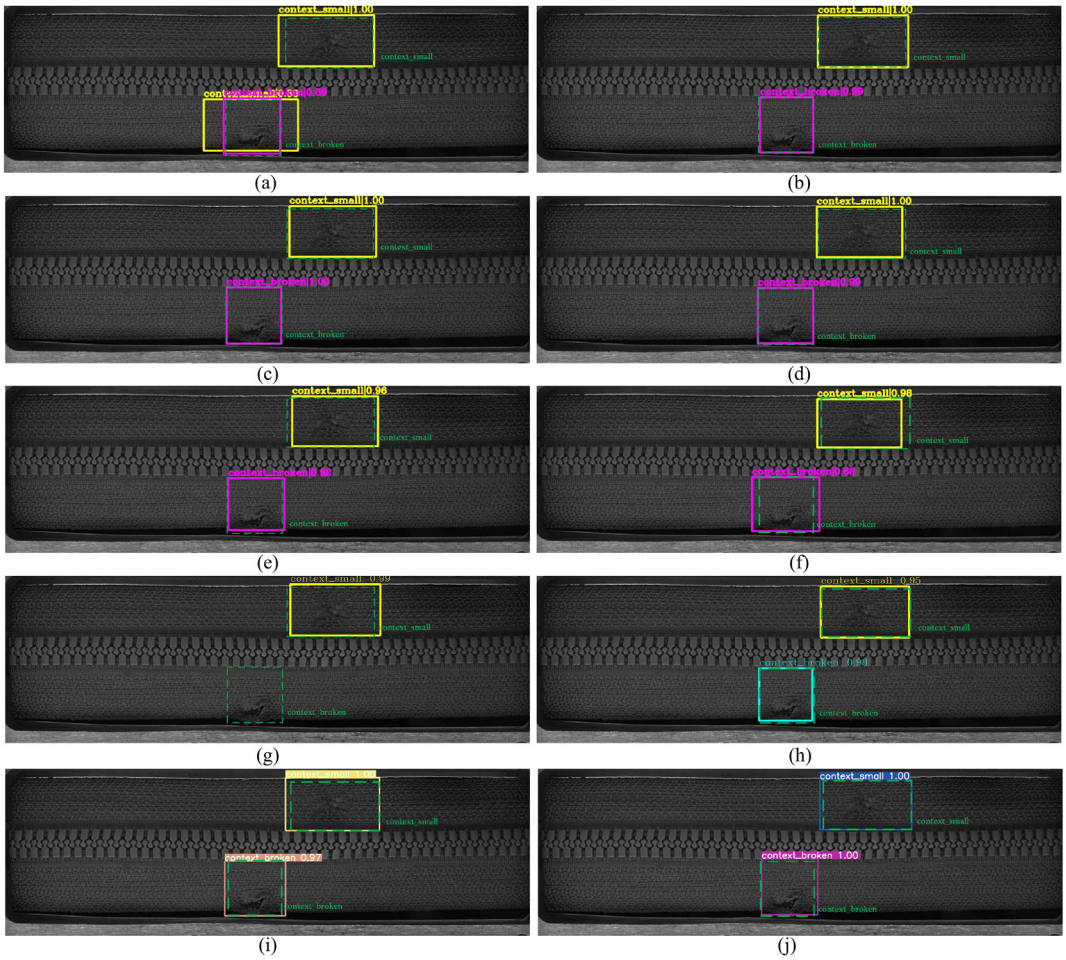


Fig. 10. The detected results of ten methods for two defects. The green dotted line box represents the ground truth of the defects and the solid line boxes denote the detection results by (a) Faster R-CNN1, (b) Faster R-CNN2, (c) Cascade R-CNN1, (d) Cascade R-CNN2, (e) RetinaNet1, (f) RetinaNet2, (g) SSD, (h) TridentNet, (i) YOLOv3-SPP, and (j) YOLOv3++-SPC.

can be seen as the large-scale or medium-scale objects in this stage, our method has the highest AP compared with other methods. The main reason is that the improved YOLOv3-SPP2 incorporates a SPP into the medium-level detection layer, boosting the detection performance of the medium-scale defects. The execution time of our method in this stage is about 7.652 s, which means that our method will take 18.22 ms for each test image with the size of 416×416 pixels. In a word, our method yields a larger mAP than all other methods and a significantly lower time consumption than other methods except YOLOv3-SPP.

6.5.2. Effectiveness of the whole two-stage cascade architecture

To investigate the overall effectiveness of the two-stage cascade framework, we present a comparison in terms of the total mAP and the total time consumption as shown in Table 6. We can observe from Table 6 that the proposed cascaded framework achieves a comparable accuracy to those by several state-of-the-art methods. Furthermore, the time consumption of our method is slightly higher than that of YOLOv3-SPP but obviously lower than those of other methods. The complete detection for an image using our two-stage framework costs about 36.20 ms, which meets the real-time requirements of industrial production lines. Overall, the cascaded two-stage framework is necessary to efficiently and accurately detect the defects of the zipper tape from the acquired images.

6.5.3. Comparison of the testing results with P–R curves

The P–R curves of ten test methods in the two stages are shown in Figs. 8 and 9. In Stage 1, the horizontal and vertical coordinate ranges are adjusted to [0.001, 1.02] and [0.35, 1.01] respectively for clear presentation. In Stage 2, the horizontal and vertical coordinate ranges are set to [0.001, 1.02] and [0.2, 1.01], respectively. The AP of each defect class can be computed using Eq. (3), which corresponds to the area under the P–R curve of the corresponding class. The mAP of all the classes can be computed by Eq. (4).The P–R curves are adopted to visualize the

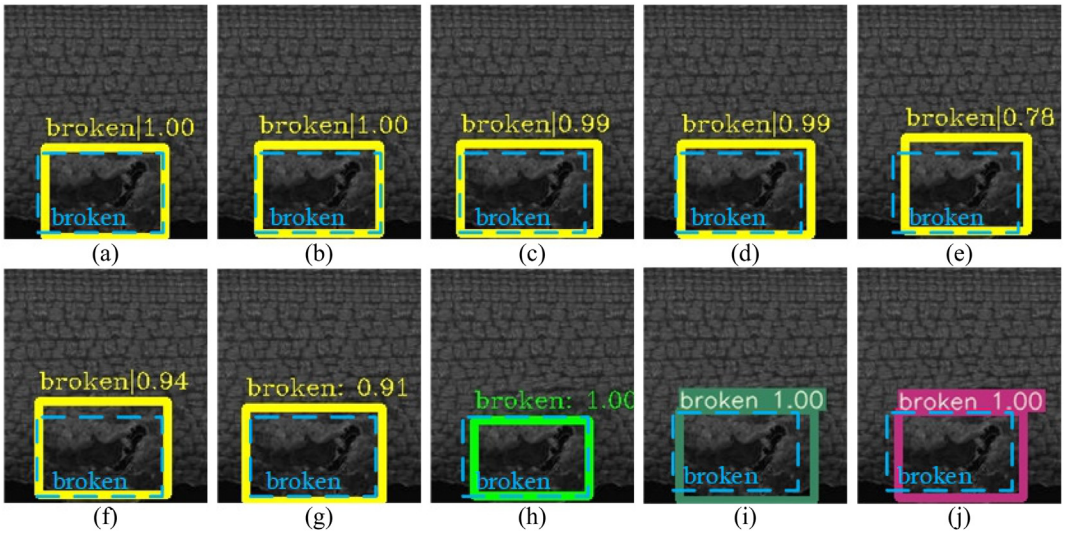


Fig. 11. The detected results of ten methods for small-scale broken defects in Stage 2. The light blue dotted line box denotes the ground truth and the solid line boxes denote the detection results by (a) Faster R-CNN1, (b) Faster R-CNN2, (c) Cascade R-CNN1, (d) Cascade R-CNN2, (e) RetinaNet1, (f) RetinaNet2, (g) SSD, (h) TridentNet, (i) YOLOv3-SPP, and (j) YOLOv3-SPP2.

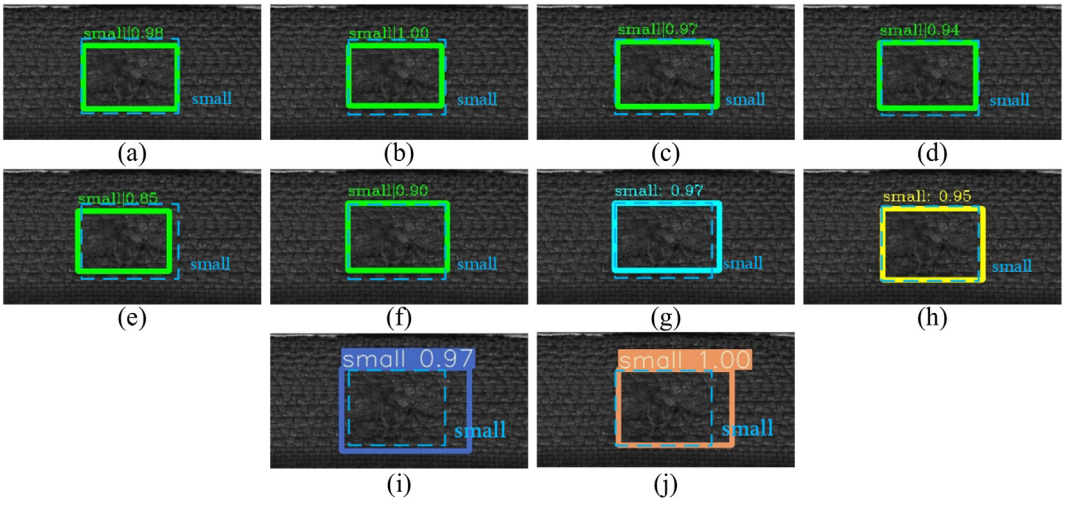


Fig. 12. The detected results of ten methods for small tape defects in Stage 2. The light blue dotted line box denotes the ground truth and the solid line boxes denote the detection results by (a) Faster R-CNN1, (b) Faster R-CNN2, (c) Cascade R-CNN1, (d) Cascade R-CNN2, (e) RetinaNet1, (f) RetinaNet2, (g) SSD, (h) TridentNet, (i) YOLOv3-SPP, and (j) YOLOv3-SPP2.

detection performance of different detection methods. In general, a larger area under the P–R curve means that the method has a higher detection accuracy.

In Stage 1, Fig. 8 shows the P-R curves of ten methods for four defects. From Fig. 8(a), we can observe that all the compared methods have a similar good performance for the broken tape defect. For tape indentation defects shown in Fig. 8(b), the precision of our method shows a slight decrease when the recall is greater than 0.3 and the precisions of other methods begin to decrease rapidly when the recall is larger than 0.9. The change in precision versus recall is more obvious for the latter two tape defects. From Fig. 8 (c) and (d), after the recall is greater than 0.6, our method stays relatively stable while the precision of other methods drops obviously. Overall, the proposed method is comparable in accuracy to the best state-of-the-art methods.

In Stage 2, Fig. 9 shows the P–R curves of the small broken tape defects and other small tape defects. In Fig. 9(a), after the recall is larger than 0.2, the precisions of other methods decrease one after another. Our method and Cascade R-CNN2 achieve relatively stable curves around 1. In Fig. 9(b), the proposed method consistently maintains the highest accuracy when the recall is larger than 0.1. This indicates that the SPP module we add in Stage 2 significantly improves the detection accuracy.

6.5.4. Comparison of the visualized detection results

To better visualize the detection results of different methods, we present the detection results of ten methods for the context_broken and context_small defects in Stage 1 as shown in Fig. 10. Meanwhile, we also show the detection results for the small-scale broken tape defects in the context_broken and small tape defects in the context_small in Stage 2 as shown in Figs. 11 and 12, respectively. In Figs. 10–12, the green dotted line box and light blue dotted line boxes represent the ground truth of the defects and the solid line boxes denote the detection results with the different methods. The confidence score of the detection box for each defect is shown on the top left of each box. It can be observed from Fig. 10 that both our method and Cascade R-CNN1 obtain a 100% accuracy for two defects. The SSD has a missing detection for the context_broken defect. The Faster R-CNN1 and RetinaNet2 have a low accuracy for the context_broken defect. Other methods except SSD, Faster R-CNN1, and RetinaNet2 achieve good accuracy.

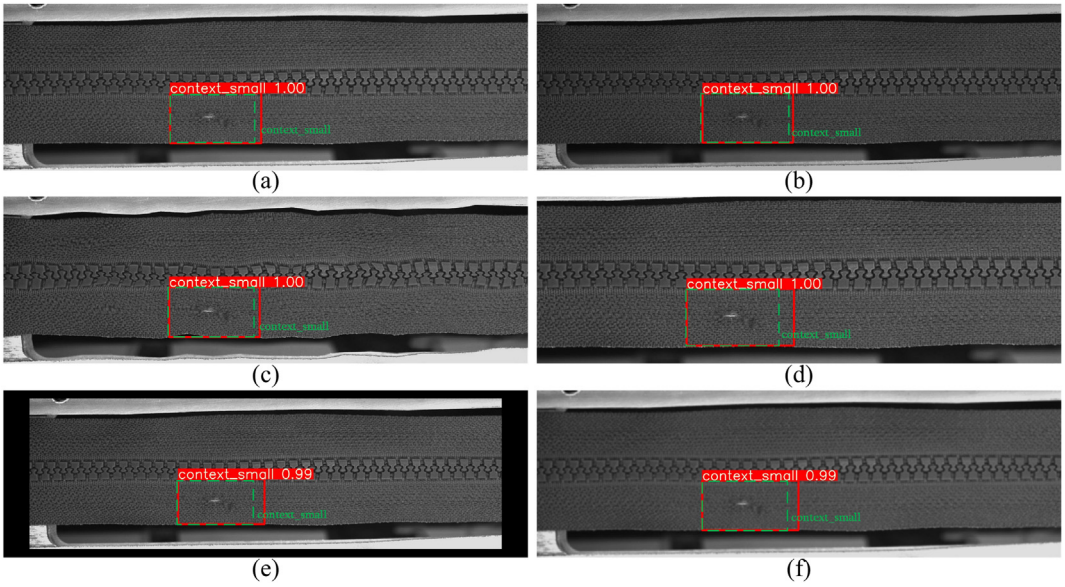


Fig. 13. Robustness of our method with respect to different image variations in Stage 1. (a) Original image. Robustness with respect to: (b) brightness dark, (c) image twist, (d) larger scale defect variation, (e) smaller scale defect variation, and (f) image blurring.

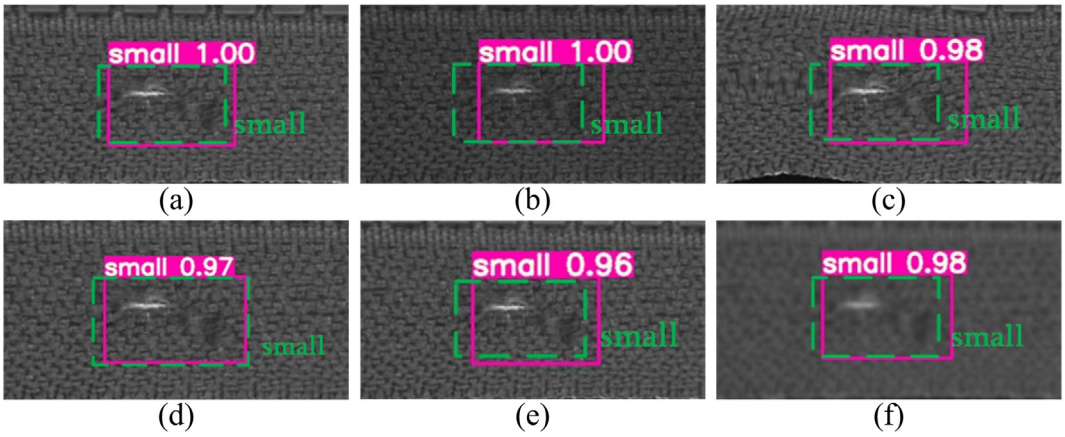


Fig. 14. Robustness of our method with respect to different image variations in Stage 2. (a) Original image containing small-scale defect. Robustness with respect to: (b) brightness dark, (c) image twist, (d) larger scale defect variation, (e) smaller scale defect variation, and (f) image blurring.

Fig. 11 shows the detection results of ten methods for small-scale defect with low illumination. We can see from Fig. 11 that Faster R-CNN1, Faster R-CNN2, TridentNet, YOLOv3-SPP, and our method obtain a 100% accuracy for small-scale broken defects. Other methods demonstrate more or less poorer detection performance. As shown in Fig. 12, for small-scale tape defect, the Faster R-CNN2 and our method achieve a 100% accuracy while the other eight methods obtain a lower detection accuracy, which again verifies the effectiveness of the proposed method.

6.6. Robustness of the proposed method to various image variations

The robustness of the proposed method is evaluated with respect to different image variations, i.e., the brightness of images, deformation of images, variation in the image scale, and image blurring. The acquired zipper images can be blurry due to the quick movement of the zipper during transmission. These variations can bring a great challenge to the detection model. Fig. 13 shows the detection results of our method in Stage 1 for the original image and five images with different variations on the original image. The detection accuracy of our method for the original image and the images obtained by the brightness decrement, deformation, and scale enlargement is 100%. For Fig. 13(e) and (f), the detection accuracy decreases a little bit but still reaches 99%. The possible reason for the accuracy reduction is that the scale reduction of image in Fig. 13(e) and (f), and image blurring in Fig. 13(f) weaken the features of the context_small defect. Fig. 14 presents the context region containing small-scale tape defects cropped from Fig. 13 in Stage 1. It can be noted from Fig. 14 that our method suffers from the image variations but still has very good detection capability. These results indicate that the proposed network in Stage 2 has a good robustness. In certain scenarios, such as image blurring in Fig. 14, the defect that is barely noticeable visually can be well detected by our method.

7. Conclusion

The vision insight based detection system for zipper tape defects can significantly improve the inspection efficiency and reduce the labor costs, especially with the development of machine vision technology in the intelligent manufacturing field. In the paper, we propose a two-stage detection framework in a cascading manner, each stage with a deep fully CNN based defects detection architecture. The proposed YOLOv3++-SPC in the first stage has four detection branches on four different scale feature maps, thus its ability to detect multi-scale defects with high detection accuracy and efficiency. The small-scale defects in the second stage are considered as the large-scale objects, which are detected using an improved fast YOLOv3-SPP2. The proposed two-stage framework is superior in both accuracy and efficiency to several latest two-stage and one-stage detectors.

CRediT authorship contribution statement

Houzhang Fang: Methodology, Writing - original draft. **Mingjiang Xia:** Software. **Hehui Liu:** Writing - review & editing. **Yi Chang:** Writing - review & editing. **Liming Wang:** Writing - review & editing. **Xiyang Liu:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors are grateful to the editors and the reviewers for their valuable comments and suggestions. The authors would like to thank Dr. Ming Zhang for correcting the grammatical errors. This work was supported by the National Natural Science Foundation of China under Grant 41501371, the Natural Science Foundation of Shaanxi Province under Grant 2018JM4036, the Fundamental Research Funds for the Central Universities under Grants XJS190302, the Open Research Fund of the National Key Laboratory of Science and Technology on Multispectral Information Processing under Grants 6142113190103, and the Opening Project of Shanghai Trusted Industrial Control Platform under grant TICPSH202003018-ZC.

Appendix A

The detailed specifications of the proposed improved architecture YOLOv3++-SPC is shown in Table A.7.

Table A.7
The detailed specifications of the proposed improved architecture YOLOv3++-SPC.

Layers	Type	Number of operations	Stride	Padding	Dilation	Filters
1–2	3 × 3 Conv + BN + ReLU	1	1	1	1	32
	3 × 3 Conv + BN + ReLU	1	2	1	1	64
3–6	1 × 1 Conv + BN + ReLU	1	1	0	1	32
	3 × 3 Conv + BN + ReLU	1	1	1	1	64
	shortcut					
	3 × 3 Conv + BN + ReLU	1	2	1	1	128
7–13	1 × 1 Conv + BN + ReLU	2	1	0	1	64
	3 × 3 Conv + BN + ReLU	2	1	1	1	128
	shortcut	2				
	3 × 3 Conv + BN + ReLU	1	2	1	1	256
14–38	1 × 1 Conv + BN + ReLU	8	1	0	1	128
	3 × 3 Conv + BN + ReLU	8	1	1	1	256
	shortcut	8				
	3 × 3 Conv + BN + ReLU	1	2	1	1	512
39–63	1 × 1 Conv + BN + ReLU	8	1	0	1	256
	3 × 3 Conv + BN + ReLU	8	1	1	1	512
	shortcut	8				
	3 × 3 Conv + BN + ReLU	1	2	1	1	1024
64–75	1 × 1 Conv + BN + ReLU	4	1	0	1	512
	3 × 3 Conv + BN + ReLU	4	1	1	1	1024
	shortcut	4				
76–78	1 × 1 Conv + BN + ReLU	2	1	0	1	512
	3 × 3 Conv + BN + ReLU	1	1	1	1	1024
79–84	3 × 3 Conv + BN + ReLU	1	1	2	2	512
	route					
	3 × 3 Conv + BN + ReLU	1	1	4	4	512
	route					
	3 × 3 Conv + BN + ReLU	1	1	6	6	512
	route					
85–89	1 × 1 Conv + BN + ReLU	2	1	0	1	512
	3 × 3 Conv + BN + ReLU	2	1	1	1	1024
	1 × 1 Conv + BN + Linear	1	1	0	1	27
90	Detection					
91–94	route					
	1 × 1 Conv + BN + ReLU	1	1	0	1	256
	2× Upsample	1				
	shortcut					
95–96	1 × 1 Conv + BN + ReLU	1	1	0	1	256
	3 × 3 Conv + BN + ReLU	1	1	1	1	512
97–102	3 × 3 Conv + BN + ReLU	1	1	2	2	512
	route					
	3 × 3 Conv + BN + ReLU	1	1	4	4	512
	route					
	3 × 3 Conv + BN + ReLU	1	1	6	6	512
	route					
103–107	1 × 1 Conv + BN + ReLU	2	1	0	1	256
	3 × 3 Conv + BN + ReLU	2	1	1	1	512
	1 × 1 Conv + BN + Linear	1	1	0	1	27
108	Detection					
109–112	route					
	1 × 1 Conv + BN + ReLU	1	1	0	1	128
	2× Upsample	1				
	shortcut					
113–119	1 × 1 Conv + BN + ReLU	3	1	0	1	128
	3 × 3 Conv + BN + ReLU	3	1	1	1	256
	1 × 1 Conv + BN + Linear	1	1	0	1	27
120	Detection					
121–124	route					
	1 × 1 Conv + BN + ReLU	1	1	0	1	64
	2× Upsample	1				
	shortcut					
125–131	1 × 1 Conv + BN + ReLU	3	1	0	1	64
	3 × 3 Conv + BN + ReLU	3	1	1	1	128
	1 × 1 Conv + BN + Linear	1	1	0	1	27
132	Detection					

References

[1] Z. Zou, Z. Shi, Y. Guo, J. Ye, Object detection in 20 years: A survey, arXiv: 1905.05055v2.

[2] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, M. Pietikäinen, Deep learning for generic object detection: a survey, International Journal of Computer Vision (2019) 1–58, https://doi.org/10.1007/s11263-019-01247-4.

[3] A. Kumar, Computer-vision-based fabric defect detection: A survey, IEEE Transactions on Industrial Electronics 55 (1) (2008) 348–363.

[4] H.Y. Ngan, G.K. Pang, N.H. Yung, Automated fabric defect detection-A review, Image and Vision Computing 29 (7) (2011) 442–458.

[5] K. Hanbay, M.F. Talu, Ö.F. Özgüven, Fabric defect detection systems and methods-A systematic literature review, OPTIK 127 (24) (2016) 11960–11973.

[6] D. Yapi, M.S. Allili, N. Baaziz, Automatic fabric defect detection using learning-based local textural distributions in the contourlet domain, IEEE Transactions on Automation Science and Engineering 15 (3) (2018) 1014–1026.

[7] S. Mei, Y. Wang, G. Wen, Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model, Sensors 18 (4) (2018) 1–18.

[8] S. Mei, H. Yang, Z. Yin, An unsupervised-learning-based approach for automated defect inspection on textured surfaces, IEEE Transactions on Instrumentation and Measurement 67 (6) (2018) 1266–1277.

[9] W. Ouyang, B. Xu, J. Hou, X. Yuan, Fabric defect detection using activation layer embedded convolutional neural network, IEEE Access 7 (2019) 70130–70140.

[10] J. Jing, H. Ma, H. Zhang, Automatic fabric defect detection using a deep convolutional neural network, Coloration Technology 135 (3) (2019) 213–223.

[11] G. Hu, J. Huang, Q. Wang, J. Li, Z. Xu, X. Huang, Unsupervised fabric defect detection based on a deep convolutional generative adversarial network, Textile Research Journal 90 (3–4) (2020) 213–223.

[12] J. Redmon, A. Farhadi, YOLOv3: An incremental improvement, arXiv: 1804.02767v1.

[13] C.J. Kuo, T. Su, Gray relational analysis for recognizing fabric defects, Textile Research Journal 73 (5) (2003) 461–465.

[14] H.Y.T. Ngan, G.K.H. Pang, S.P. Yung, M.K. Ng, Wavelet based methods on patterned fabric defect detection, Pattern Recognition 38 (4) (2005) 559–576.

[15] M.K. Ng, H.Y.T. Ngan, X. Yuan, W. Zhang, Patterned fabric inspection and visualization by the method of image decomposition, IEEE Transactions on Automation Science and Engineering 11 (3) (2014) 943–947.

[16] C. Li, G. Gao, Z. Liu, D. Huang, J. Xi, Defect detection for patterned fabric images based on ghog and low-rank decomposition, IEEE Access 7 (2019) 83962–83973.

[17] A. Kumar, H.C. Shen, Texture inspection for defects using neural networks and support vector machines, in: IEEE Proceedings of International Conference on Image Processing, vol. 3, 2002, pp. III-353–III-356.

[18] H. Bu, J. Wang, X. Huang, Fabric defect detection based on multiple fractal features and support vector data description, Engineering Applications of Artificial Intelligence 22 (2) (2009) 224–235.

[19] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems 28, Curran Associates Inc, 2015, pp. 91–99.

[20] Z. Cai, N. Vasconcelos, Cascade R-CNN: High quality object detection and instance segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence (2019) 1–16, https://doi.org/10.1109/TPAMI.2019.2956516.

[21] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, real-time object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, A.C. Berg, SSD: Single shot multibox detector, in: L. B., M. J., S. N., W. M. (Eds.), European Conference on Computer Vision, vol. 9905, 2016, pp. 21–37.

[23] J. Redmon, A. Farhadi, YOLO9000: Better, faster, stronger, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6517–6525.

[24] T. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2) (2020) 318–327.

[25] X. Yin, Y. Chen, A. Bouferguene, H. Zaman, M. Al-Hussein, L. Kurach, A deep learning-based framework for an automated defect detection system for sewer pipes, Automation in Construction 109 109 (1–17) (2020) 102967.

[26] J. Wang, N. Wang, L. Li, Z. Ren, Real-time behavior detection and judgment of egg breeders based on YOLOv3, Neural Computing and Applications (2019) 1–11.

[27] Z. Qiu, S. Wang, Z. Zeng, D. Yu, Automatic visual defects inspection of wind turbine blades via YOLO-based small object detection approach, Journal of Electronic Imaging 28 (4) (2019) 1–11.

[28] A. Neubeck, L. Van Gool, Efficient non-maximum suppression, in: 18th International Conference on Pattern Recognition (ICPR'06), vol. 3, 2006, pp. 850–855.

[29] K. He, X. Zhang, S. Ren, J. Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, in: European Conference on Computer Vision 2014, Springer International Publishing, Cham, 2014, pp. 346–361.

[30] A. Hernández-García, P. König, Further advantages of data augmentation on convolutional neural networks, in: Artificial Neural Networks and Machine Learning – ICANN 2018, Springer International Publishing, Cham, 2018, pp. 95–103.

[31] M. Everingham, S.M.A. Eslami, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: A retrospective, International Journal of Computer Vision 111 (2015) 98–136.

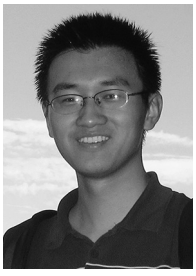
[32] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944.

[33] Y. Li, Y. Chen, N. Wang, Z. Zhang, Scale-aware trident networks for object detection, in: 2019 IEEE International Conference on Computer Vision(ICCV), 2019, pp. 1–10.

Houzhang Fang received the Ph.D. degree in control science and engineering in 2014 from the Huazhong University of Science and Technology. He is currently a lecturer with Xidian University. His research interests include image processing, computer vision, and deep learning.

Mingjiang Xia received the B.S. degree in electronic commerce from the NorthWest Agriculture and Forestry University, Yangling, China, in 2019. He is currently working towards the M.S. degree in software engineering at Xidian University, Xi'an, China. His research interests include computer vision and deep learning.

Hehui Liu received the B.S. degree in software engineering in 2003 and the M.S. degree in computer science in 2006, respectively, from Xidian University, Xi'an, China. He is currently the CTO of Nanjing Cognitive Internet of Things Research Institute, Nanjing, China. His research interests include artificial intelligence, internet of things, and development and application of software engineering technology.



Yi Chang received the B.S. degree in automation from the University of Electronic Science and Technology of China, Chengdu, China, in 2011, and the M.S. degree in pattern recognition and intelligent systems in 2014 and the Ph.D. degree in control science and engineering in 2019, both from the Huazhong University of Science and Technology, China. From 2014 to 2015, he was a Research Assistant with Peking University, Beijing, China. He is with the Pengcheng Laboratory. His research interests include deep learning, image processing, and computer vision.

Liming Wang received the B.S. degree in software engineering in 2004, the M.S. degree and the Ph.D. degree in computer science in 2007 and 2013, respectively, from Xidian University, Xi'an, China. He is currently an Associate Professor in the School of Computer Science and Technology, Xidian University. His research interests include artificial intelligence in medicine and industry intelligence.

Xiyang Liu received the B.S. degree in software engineering from the Xidian University, in 1992, and the M. S. degree in software engineering in 1995, and the Ph.D. degree in software engineering in 2007, both from the Xidian University, Xi'an, China. He is currently a Professor with the School of Computer Science and Technology, Xidian University. He is also the director of the Software Engineering Institute, Xidian University. He is the Member of Software Engineering Professional Committee of China Computer Society and the Member of the Medical Artificial Intelligence Branch of the Chinese Society of Biomedical Engineering. His research interests include clinically applicable machine learning for high performance medical image interpretation (3D breast ultrasound and PET volumes), computational pathology(hepatocellular and gastric carcinoma) and clinical decision support in cases of diagnostic uncertainty and complexity (women's pain related diseases), industrial intelligent visual inspection, and high trustworthy software technology for aerospace systems.