electronics

MDPI

Article

Surface Defects Recognition of Wheel Hub Based on Improved Faster R-CNN

Xiaohong Sun [1,2], Jinan Gu [1,*], Rui Huang [1], Rong Zou [1] and Benjamin Giron Palomares [3]

[1] School of Mechanical Engineering, Jiangsu University, Zhenjiang 212000, China; jxxsxh_1002@163.com (X.S.); easitmickly@163.com (R.H.); zr@ujs.cn (R.Z.)

[2] School of Mechanical Engineering, Anyang Institute of Technology, Anyang 455000, China

[3] Training Center, Anyang Institute of Technology, Anyang 455000, China; tiny_ikari@yahoo.com.mx

* Correspondence: gujinan@tsinghua.org.cn

Received: 2 April 2019; Accepted: 25 April 2019; Published: 29 April 2019

Abstract: Machine vision is one of the key technologies used to perform intelligent manufacturing. In order to improve the recognition rate of multi-class defects in wheel hubs, an improved Faster R-CNN method was proposed. A data set for wheel hub defects was built. This data set consisted of four types of defects in 2,412 1080 × 1440 pixels images. Faster R-CNN was modified, trained, verified and tested based on this database. The recognition rate for this proposed method was excellent. The proposed method was compared with the popular R-CNN and YOLOv3 methods showing simpler, faster, and more accurate defect detection, which demonstrates the superiority of the improved Faster R-CNN for wheel hub defects.

Keywords: defects recognition; deep learning; regional proposal network; Faster R-CNN

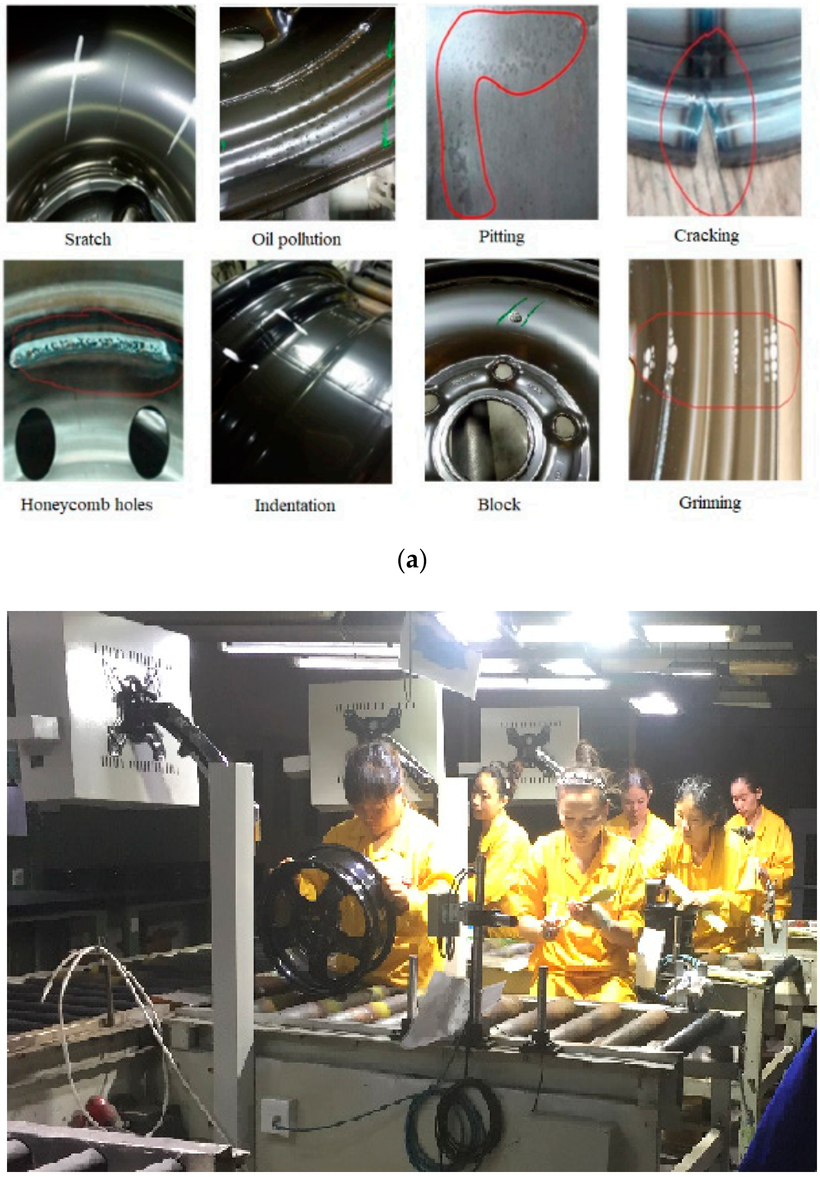
1. Introduction

The automotive industry is an important part of the national economy, and the automobile is an essential means of transportation in daily life. The wheel hub is an important part of the automobile. In recent years, due to the rapid growth of production and imperfect processing technology, more than 40 kinds of defects in the hub are generated (see some examples in Figure 1a). These defects will affect the good appearance of the product and brand image, and some defects will lead to serious traffic accidents. Therefore, the quality control is very important.

Due to the different definitions of hub defects at home and abroad, foreign testing equipment cannot meet the standards of the domestic enterprises, and hence many enterprises still employ manual inspection for complex surfaces. Nevertheless, the inter-class similarity and intra-class diversity of defects are the main difficulties for the detection. Traditional manual defect detection methods have great limitations, such as low efficiency and high labor costs. The most important problem of manual detection is its susceptibility to workers' engagement degree and the level of relevant knowledge.

Machine vision detection provides the following advantages: High production efficiency, high automation level, good detection rate, and adaptability to the special industrial environment. Therefore, visual-based defect detection has been widely used in various fields, such as ceramic tile detection [1], fabric detection [2], and plant disease detection [3]. Multiple studies have been performed on surface defect detection [4–7].

Gong at al. [8] presented a method for the rapid detection of surface defect areas of strip steel. Five statistical projection features were extracted from the detection area of the surface image, and were used by the extreme learning machine (ELM) and region of background (ROB) pre-detection classifiers. A coating damage/corrosion detection device based on a three-layer feedforward artificial neural network was introduced by Reference [9]. Krummenacher et al. [10] designed an artificial neural network with constant cyclic movement to detect wheel deviation and roundness error, and they simulated the relationship between the inherent measurement values of these defects. Cha et al. [11] used the deep convolutional neural network (CNN) to detect concrete cracks. The robustness and adaptability of their method were significantly improved compared with traditional edge detection methods (Canny and Sobel).



(**a**)

(**b**)

Figure 1. Wheel hub inspection: (**a**) common defects; (**b**) manual detection process.

However, most methods can only detect specific types of defects, but cannot achieve accurate detection of multiple defects. Similarities between classes and the diversity within classes of defects make vision inspection challenging. At present, for complex workpieces with multi-curved surfaces, manual repeated inspection is commonly used in order to improve the detection rate. According to Cong et al. [12], 57% of the enterprises follow similar procedures, and therefore intelligent and robust detection methods are urgently needed to replace manual detection.

These show the application value of this study. The fundamental reason why this application cannot be realized is the technical difficulty of generalization recognition using deep learning, such as the method of quickly generating the region proposal, the identification robustness of complex objects, and the balance between accuracy and time consuming, which are all aspects of the scientific value of this paper.

In this paper, a faster R-CNN method was developed to detect several common types of defects in fabricated wheel hubs. The developed method was arduously tested and compared to commonly used methodologies. The structure of this paper is as follows: Various solutions to defect or damages recognition are described in Section 2. The generation of the image database for the wheel hub defects is described in Section 3. The Faster R-CNN model and modified Faster R-CNN for multi-class defects of the wheel hub are explained in Section 4. In Section 5, the experimental procedure is depicted, the results of training, validation, and testing are discussed, and a comparison of the improved method with the state-of-the-art methods is presented. Section 6 summarizes this research and future efforts.

2. Related Work

At present, several non-contact detection methods based on the traditional computer vision have been successfully applied. For example, an improved hub defect peak localization algorithm was proposed by Li et al. [13]. They used a trend peak algorithm to extract the hub defect area and then a BP neural network to classify and identify the hub defect. In order to complete surface defect detection of printed circuit boards (PCBs), an effective similarity measurement method has been proposed [14]. This method uses the adjoint matrix of two comparative images to calculate the symmetric matrix. The rank of the symmetric matrix is used as the similarity index of defect detection. The rank value of a defect-free image is zero, and the rank value of a defect image is obviously larger. However, this method cannot be adapted to multi-curved surface hub defect detection. A method based on hybrid chromosome genetic algorithms was conducted to classify metal surface defects [15]. Similarly, aiming at metal surface defects, a method based on digital image singular value decomposition was developed by [16]. Although these methods have improved somewhat, they still need some preprocessing and postprocessing techniques, and hence they are time-consuming. Additionally, the types of defects that they can detect are limited.

In order to solve the problems of image processing technology mentioned above, deep learning has been used. Deep learning combines low-level features to form more abstract high-level attribute representation and to discover distributed feature representation of data. Therefore, its excellent performance has been gradually employed by researchers since 2006. For example, Yi et al. [17] adopted the end-to-end method based on a convolutional neural network to realize the identification and classification of seven defects of a particular steel product. A region-based convolutional neural network method has been adopted to detect ships [18]. Aiming at the surface detection of solar panels with uneven structure and complex background, a visual defect detection method based on multi-spectral deep convolutional neural network (CNN) was designed by adjusting the depth and width of the network [19]. A method based on a deep convolution neural network (DCNNS) for defect detection of parts and components was proposed by Reference [20]. This method combines three serial detection stages based on DCNN and includes two detectors to locate the cantilever joint and its fasteners in turn, and incorporates a classifier to diagnose the fastener defects. Although all these methods can use sliding windows to locate defects, it is difficult to determine the size of sliding windows due to the different scales of defects in the test set.

Breakthroughs in object detection methods have always been driven by the success of regional proposed methods. For example, Girshick [21] proposed a scale adjustable detection algorithm based on the combination of regional proposals and CNN in order to achieve multi-objective detection. This method had two key points: First, the convolution of the high-performance network was adopted to realize the bottom-up regional proposal localizing and segmenting the defects; second, pre-training supervision was conducted when training data was insufficient, and fine-tuning of the specified region was carried out to significantly improve performance. Compared with traditional CNN method with sliding windows, region-CNN (R-CNN) [22] can significantly improve the accuracy of target detection. However, the method is time-consuming because it is not an end-to-end network, but three processes

(CNN, regression, SVM) at the same time. Failure to implement computation sharing is a major cause of time consumption.

Time consumed in object proposal is the major bottleneck for detection technology. Aiming at this problem, Ross Girshick [23] proposed the FAST R-CNN for object detection by training deep learning network VGG19. The operational speed of this method was nine times faster than that of R-CNN, and three times faster than that of SPP net. Moreover, it achieved the highest average precision (66%) and a detection time of 300 ms per image for PASCAL VOC 2012. However, the detection speed and precision of FAST R-CNN can be improved. Time-consuming and deficient training is generated, because the object proposal is implemented using external methods, such as selective search. To solve this problem, Ren et al. [24] achieved a detection accuracy of 73.2% by combining region proposal network (RPN) and Fast R-CNN as one network through sharing features. The detection time of a single image was only 198 ms. This methodology greatly reduced the calculation cost and improved the detection accuracy by better training of the data. Several detection techniques have implemented the combination of RPN and Fast R-CNN. For example, Liu et al. [25] used this combination in order to effectively detect the defects of complex texture fabrics. They adopted the non-maximum suppression and data enhancement strategies to improve the detection accuracy.

Inspired by the previous researches mentioned above, a new method was proposed by this research work to detect multi-class defects in wheel hubs. A faster R-CNN framework was modified to complete training, validation, and testing. Four defects (scratch, oil pollution, block and grinning,) of a wheel hub were used as representatives to achieve recognition and classification. More importantly, this flexible method can allow the easy addition of other defect types to the dataset in order to achieve universality.

3. Dataset

3.1. Image Data Collection

The four defects (scratch, oil pollution, block, and grinning) with the highest frequency occurrence are presented in Figure 2. Four hundred and two images (1440 × 1080 pixels) of wheel hubs were collected. These images were taken under different lighting conditions and at a distance of 0.3–0.5 m. The images were taken in a wheel hub manufacturing company located in Hangzhou, Zhejiang Province, China.

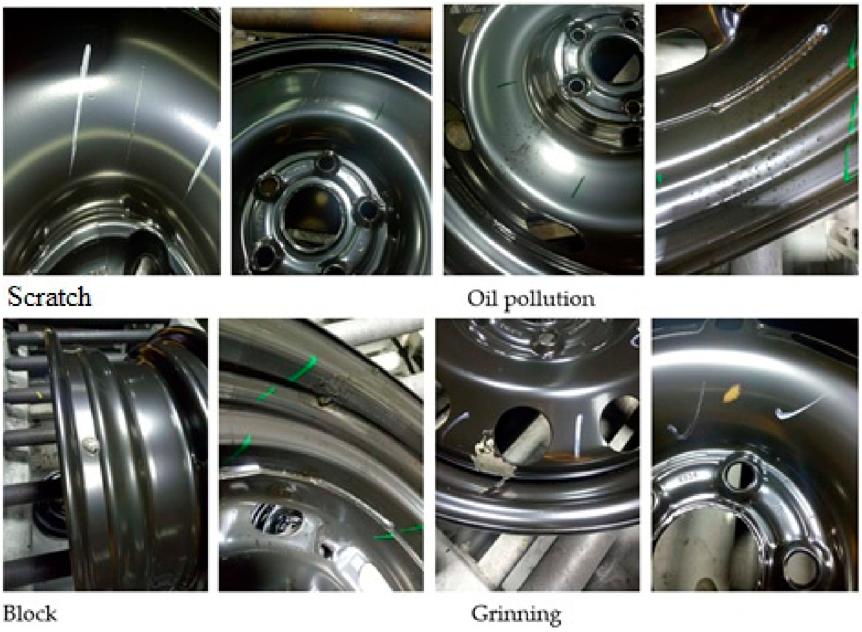


Figure 2. Types of wheel hub defects.

3.2. Image Data Augmentation

Due to insufficient image data, image amplification was performed. This amplification can improve the performance of CNN and reduce the probability of overfitting. Adding noise (Gaussian noise, gaussian blur, salt and pepper noise, Poisson noise, or motion blur) to images are commonly used methods of amplification (see Figure 3). Finally, the number of images after amplification is 2412.

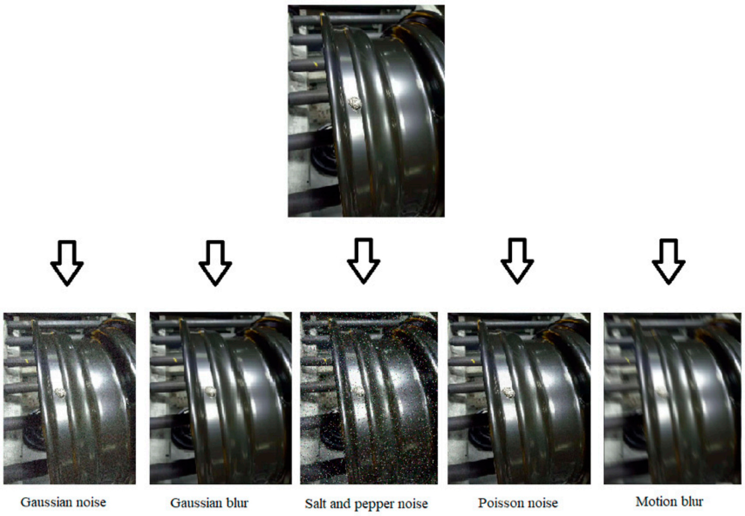


Figure 3. Augmentation of the images.

3.3. Annotation of the Images

In order to annotate the images (defect types and coordinates of the bounding box), a Python code was used to manually label the image. During the labeling process, 4554 targets were marked out from 2,412 images (examples can be seen in Figure 4). It is worth noting that the scratch defect was carried by the raw material itself or produced by scratching with a sharp object during the processing. Oil pollution defects are formed when the oil falls on the surface of the wheel hub. The coarse agglomeration on the surface of paint was called a block defect. Due to the imperfection of coating technology, some areas on the surface of the hub can be unpainted, forming a defect called grinning.

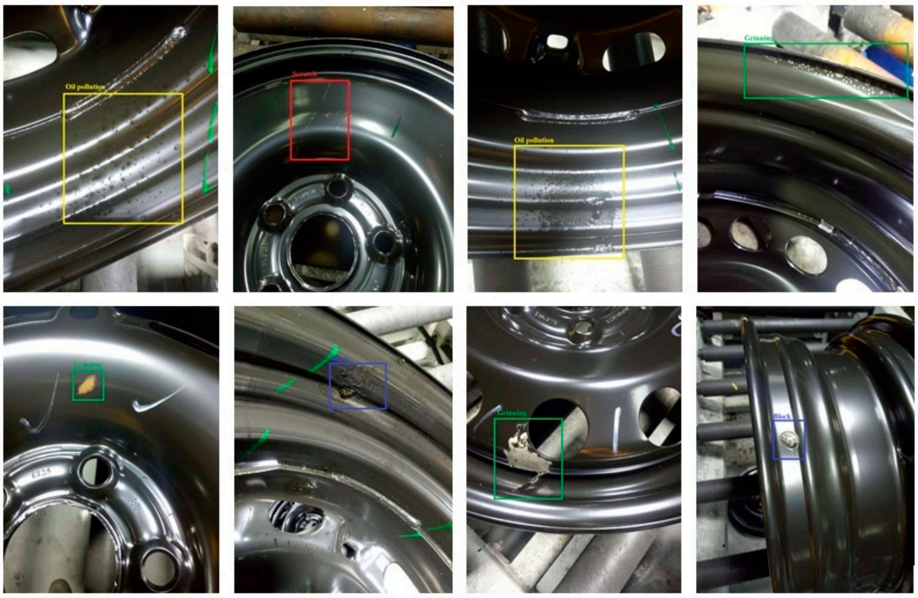


Figure 4. Images with bounding boxes and labels.

The testing set was randomly selected from annotated images, which contains 30% of the defects. In addition to the test set, the remaining images were used as the training set and validation set. The proportions of training, validation and testing sets are shown in Table 1.

Table 1. The proportion of training, validation, and testing sets.

Defect Class	Training (50%) and Validation (20%)		Testing (30%)	
	Defect	Number of Images	Defect	Number of Images
Scratch	395	200	169	100
Oil pollution	1683	828	831	414
Block	466	238	182	119
Grinning	597	198	231	99

4. Methods

Faster R-CNN has been applied very well in the field of multi-target detection [26], because RPN networks can generate object proposals with a high recall rate. As shown in Figure 5, the original Faster R-CNN is composed of two networks. RPN and FAST R-CNN share the same convolution results. RPN is used to generate the proposals, and FAST R-CNN is used to accurately locate the object [27]. However, due to the small number of available training samples with labels, the weight of the model cannot be initialized randomly, otherwise, it will easily lead to overfitting or non-convergence of the algorithm. Fortunately, transfer learning [28] is a good way to solve this kind of problem. Accordingly, the mature classification model was adopted, and then the network structure was adjusted according to the specific object.

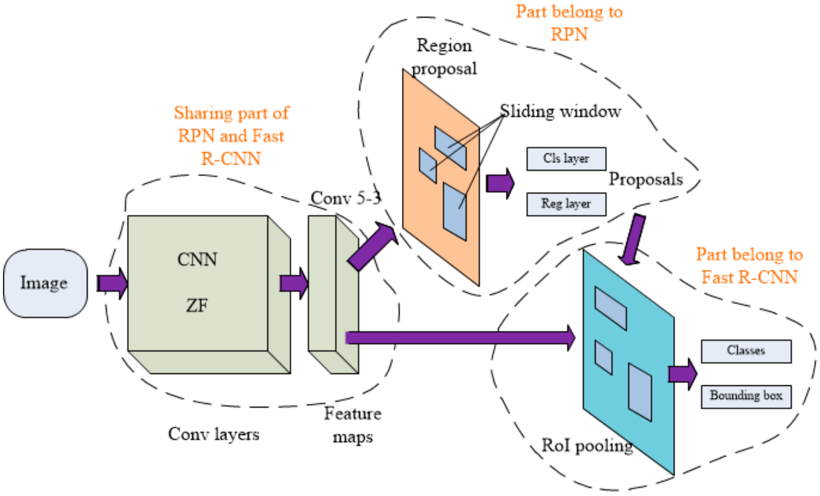


Figure 5. The Faster R-CNN model.

4.1. *Regional Proposal Network (RPN)*

The role of RPN [24] is to generate proposals, including a rectangular box for proposals and probability of each proposal. The implementation of RPN is adopting a sliding window of n * n (in the paper, chose n = 3) to the convolution feature map of convolution layer 5–3, (conv 5–3) (in the paper, chose n = 3), then a length of 256 (Zeilerand Fergusmodel, ZF) [29] or 512 (the Simonyan and Zisserman model, VGG16) [30] fully connected network is generated. Two fully connected layers of the same level regression layer (reg layer), classification layer (cls layer) [24] are following the 256-dimensional or 512-dimensional features. The reg layer is used to predict center coordinates and wide high value of the anchor, and the cls layer can be used to judge whether the proposal is an object or background as shown in Figure 5. Sliding window ensures that the two layers are related to all the feature spaces of conv 5-3. In the RPN network, we need to focus on understanding the conception of anchors and loss functions.

4.1.1. Anchors

Multiple regional proposals are predicted simultaneously during the process of window sliding. For the proposals, there are k possible shapes of the prediction box, therefore the cls layer has 2k outputs (0/1), and the reg layer has 4k outputs (x, y, w, h). The k proposals for the same localization are called anchors. Anchor point is located in the center of the sliding window and related to the scale and aspect ratio. By default, we use 3 scales and 3 aspect ratios to generate $k = 9$ anchors. So, for a convolutional feature map of size $W * H$ (about 2400), $W * H * k$ anchors are produced. The method of producing anchor with k-mean methods does not have translation invariance [31], on the contrary, the anchor generated by this method has translation invariance. It is worth mentioning that translational invariance also reduces the model size, and then the number of parameters in the output layer is two orders of magnitude less than that in the multi- box method. Even considering the feature prediction layer, our method is still one order of magnitude less than the multi-box approach, which reduces the risk of overfitting on small data sets.

4.1.2. Loss Function

For training RPN, a binary class tag (object or no object) is assigned to each anchor point. To classify an anchor content as an object or no object, the following rules should be applied:

(1) The anchor point with the highest intersection over union (IoU) should be defined as a positive sample (see Figure 6 and Equation (1)).

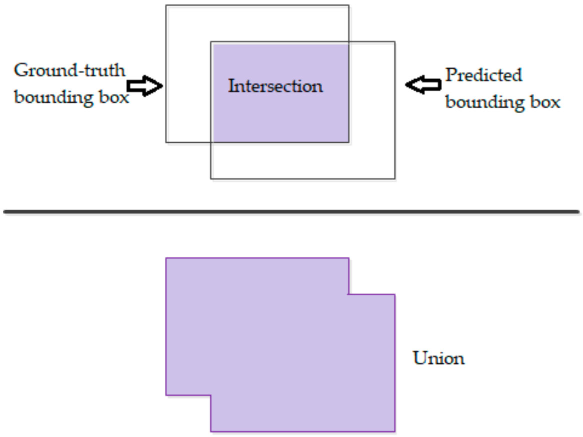


Figure 6. The implication of IoU.

(2) The IoU with any truth box over 0.7 should be defined as positive samples.

(3) If the IoU between an anchor and any target area were less than 0.3, it should be judged as a negative sample.

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{1}$$

Note that one ground-truth box can assign positive labels to multiple anchor points. Although the second condition is sufficient to determine the positive sample, the first rule is often used because sometimes the positive sample cannot be found in the second condition. Anchors that are neither positive nor negative contribute nothing to the training. With these definitions, an objective function is minimized after the multitasking loss of Fast R-CNN. The loss function of an image can be defined as [24]:

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \tag{2}$$

where i is the serial number of the anchor points in each mini- batch, p_i and $p_i{}^*$ are ground truth label (0/1) and the probability of anchor i being object, t_i and $t_i{}^*$ are prediction box parameters and calibration box parameters, and L_{cls} and L_{reg} are classification loss function and regression loss function. $p_i{}^* L_{reg}$ means that the regression is only applied to the positive sample (negative sample $p_i{}^* = 0$). Cls-layer and reg-layer outputs are p_i and t_i.

The loss function was determined by normalizing N_{cls} (depend on mini batch size, here 256) and N_{reg} (depends on the number of anchor points, here 2400), and then weighting them by an equilibrium parameter λ (by default, $\lambda = 10$).

For the bounding box regression, the following four coordinates were parameterized: x, y, w, and h. x and y represent the central coordinates of a $w \times h$ box. Variables x, x_a, and x^* represent the prediction box, anchor box, and truth box, respectively. A similar convention is followed for derivate variables of y, w, and h. This can be considered as the anchor box regressing to the nearby ground truth box. The variables t_i and $t_i{}^*$ in Equation (2) are used to define the geometrical differences between the predicted bounding box and anchor, as well as the ground truth box and the anchor. These geometrical differences are calculated as:

$$\begin{aligned}
&t_x = (x - x_a)/w_a, t_y = (y - y_a)/h_a,\\
&t_w = \log(w/w_a), t_h = \log(h/h_a),\\
&t_x{}^* = (x^* - x_a)/w_a, t_y{}^* = (y^* - y_a)/h_a,\\
&t_w{}^* = \log(w^*/w_a), t_h{}^* = \log(h^*/h_a),
\end{aligned} \tag{3}$$

4.2. Faster R-CNN Model and Training

For the input of the region proposal in the Fast R-CNN [32] network, a selective research method is adopted, which takes more time and limited optimization space for the whole system. However, Faster R-CNN used the RPN network to generate region proposal, which making efficiency jump again. Since Faster R-CNN is implemented by sharing the convolution layer of the RPN and the Fast R-CNN network, the RPN and Fast R-CNN cannot be trained independently, otherwise the parameters of the convolution layer will be changed. Therefore, training of the Faster R-CNN is more complex, and a four-step training strategy is adopted. The steps are as follows:

(1) The RPN network is trained separately, and the training model is initialized with ImageNet, and parameters are adjusted end to end.

(2) The detection network, Fast R-CNN, is trained independently. Object proposals for training are from RPN net in step 1, and the ImageNet model is adopted for model initialization.

(3) The parameters of step 2 are used to initialize the RPN model, but the convolution layer is fixed during the training, while the parameters belonging to the RPN in Figure 5 are adjusted.

(4) Keep the shared convolutional layer fixed and use the RPN output proposals (step 3) as the input to fine-tune the parameters belonging to Fast R-CNN in Figure 5.

4.3. The Improved Faster R-CNN

The ZF network [29] and VGG [30] are two commonly used networks in sharing convolution between RPN and Fast R-CNN. However, ZF net is known for its speed, which has been confirmed in the literature [24,33], and therefore this paper adopts the ZF net. In order to make the method adapt to multi-class defects detection for wheel hubs, we made the following improvements for ZF net. Firstly, we improved the original ZF net for the RPN. The last maximum pooling layer and full connection layer of ZF net were replaced by a sliding convolution layer, then a full connection layer with a depth of 256 was connected, and its softmax layer was replaced by a softmax layer and regression layer, which was Figure 7.

Second, the ZF net was improved for the Fast R-CNN. The last maximum pooling layer was replaced by a region of interest (RoI) pooling layer. To prevent over-fitting during training, drop-out layers with a threshold of 0.5 were added in-between fully connected layers. The depth value of the final fully connection layers was changed to five (four types of defects and a background) to ensure compatibility. Finally, the softmax layer was replaced by the softmax layer and the regress regression layer (see Figure 8).

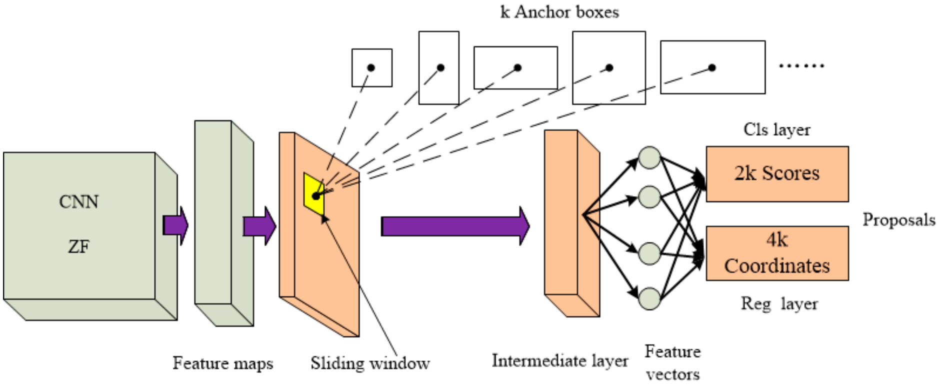


Figure 7. The structure of the improved region proposal network (RPN).

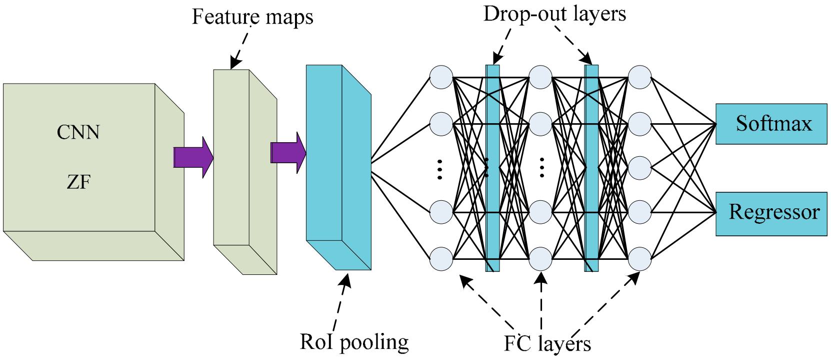


Figure 8. The structure of the improved Fast R-CNN.

As mentioned above, because the first nine layers of RPN and Fast R-CNN have the same structure in Faster R-CNN, CNN computing sharing was achieved. Figure 9 shows the whole structure of the improved Faster R-CNN.

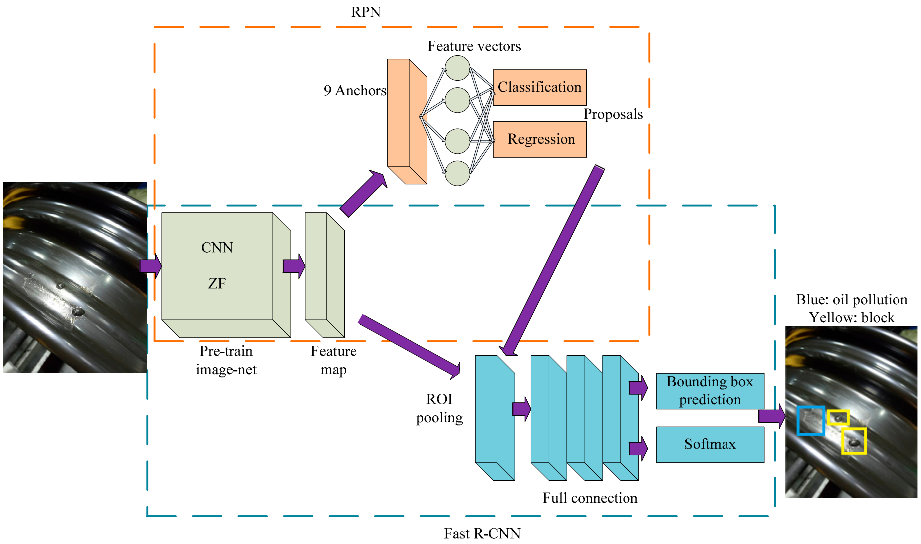


Figure 9. The structure of the improved Faster R-CNN.

For one image, RPN may generate more than 2000 object proposals, which will lead to expensive calculation and may reduce the accuracy of detection. Therefore, the output of RPN was sorted according to the score of softmax layers. It is known that under the premise of not reducing the recognition accuracy, the number of proposals can be appropriately reduced to improve the detection speed. Accordingly, a maximum of 300 proposals was adopted by this investigation.

5. Experimental Results and Discussion

5.1. Experiment Implementation

The open source Faster R-CNN library was adopted to complete the experimental investigation. Faster R-CNN was implemented by means of MATLAB 2014a, CUDA6.5 and CUDNN5.1 on a computer with a Core Xeon E5-2650 v3@2.3 GHz CPU, 64 GB DDR4 memory, and 8 GB memory NVIDIA Quadro K5200 graphics processing unit (GPU). At present, there is no good solution for the setting of initial parameters, and therefore a trial-error method is a good choice. In order to find the optimum anchor scale and ratio, 11 kinds of combinations were used by selecting 3 scales from 96, 128, 192, 256, 384, and 512, and by choosing 3 ratios from 0.2, 0.35, 0.5, 0.85, 1, 1.15, 1.7, 1.85, and 2.

5.2. Results of Training, Validation, and Testing

For 11 cases, the four-step training strategy described in Section 4.2 was applied, and its detection accuracy was evaluated by a test set. The training time per case was nearly 16 hours, and the test time for each image (1080 × 1440 pixels) was 0.3 s. As shown in Figure 10, the performance of this method was measured by two parameters: Average precision (AP) and mean average precision (mAP). AP is an indicator to measure the performance of the detection algorithm, and mAP is the average value of the APs for the different types of defect detection.

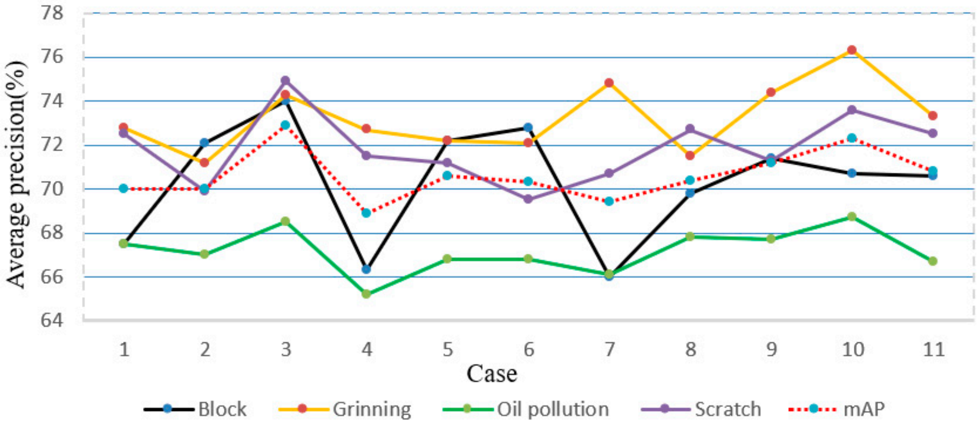
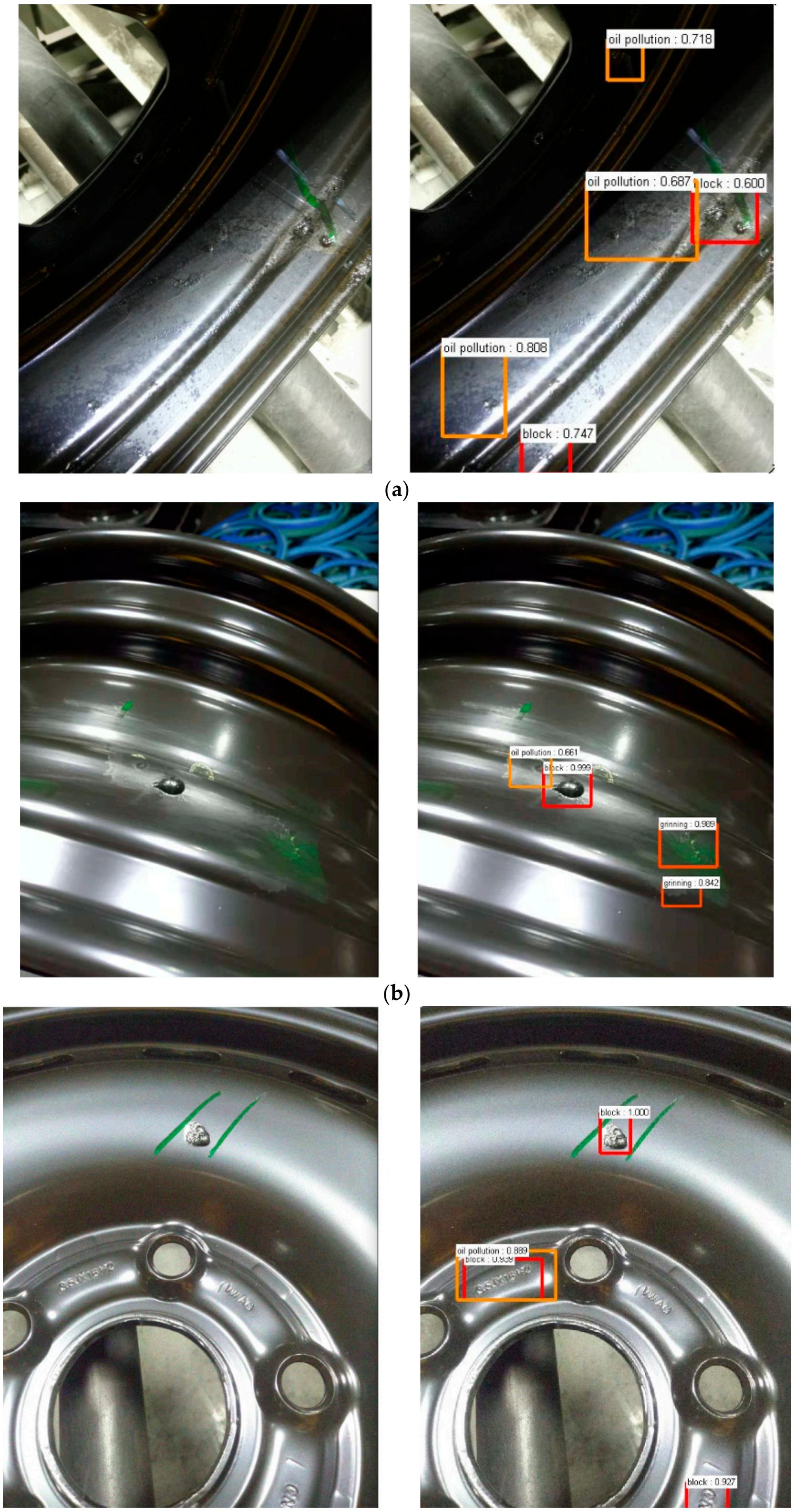


Figure 10. The performance of the network for the testing set.

As can be seen from Figure 10, the overall recognition rate of the four types of defects is unsatisfactory. The reasons for the low detection accuracy might be: Poor lighting, insufficient image data, and the intra-class diversity of these four types of defects. In the future, these issues can be solved by improving lighting methods and adding more training images. The highest detection accuracy was obtained for grinning (76.3%). In order to ensure reasonable average detection accuracies, case 3 (mAP = 72.9%) was chosen as the test model. The detection accuracies for this model were 75.0%, 68.5%, 73.9% and 74.3% for scratch, oil pollution, block, and grinning defects, respectively. The case 3 anchor parameters were 0.2, 1.15, and 1.8 for the ratio, and 96, 256, and 384 for scale.

5.3. Testing New Images

In order to better understand how the improved Faster R-CNN implemented the defect detection of the wheel hub, six additional test images were investigated. These images were taken in the same shooting environment. To ensure similar detection accuracies as those in case 3, each image size was 1080 × 1440 pixels. The images and their detection accuracies are shown in Figure 11. The grinning defect had a high detection rate, while the oil pollution defect showed a relatively low detection rate.

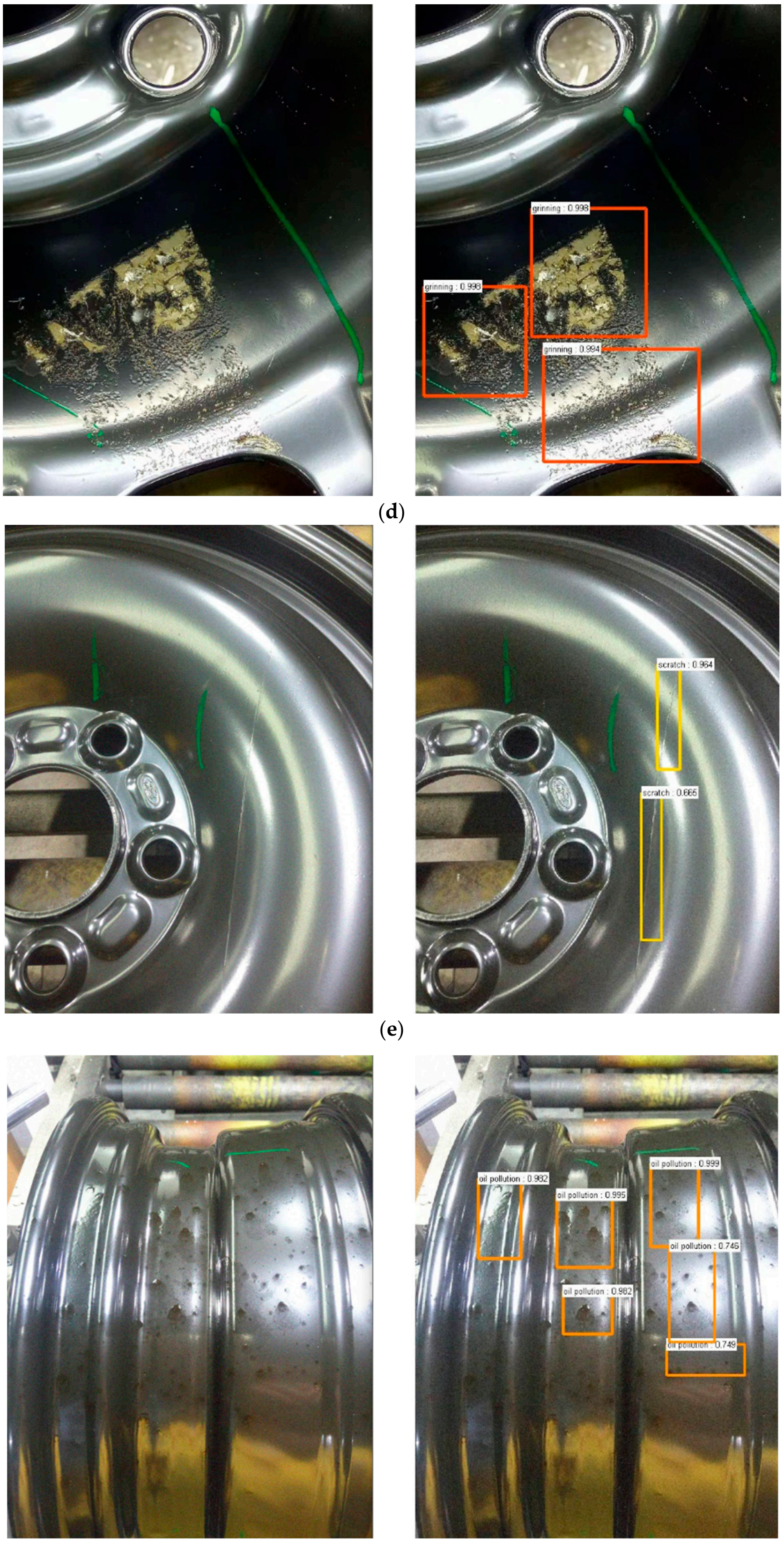


(**a**)

(**b**)

(**c**)

Figure 11. *Cont.*



(**d**)

(**e**)

(**f**)

Figure 11. Testing results for six additional images. (**a**) oil pollution and block; (**b**) block and oil pollution; (**c**) block; (**d**) grinning; (**e**) scratch; (**f**) oil pollution.

The printed characters showed in Figure 11c were misjudged as oil pollution or block. As seen in Figure 11f, some oil pollution regions were not detected. All these problems are mainly related to uneven illumination and inadequate training image data. Such problems can be addressed through multi-angle lighting, the addition of polarizers to eliminate the surface reflection phenomenon, and a larger training data set. Therefore, in future research, larger data sets and wider shooting distances should be used to improve the recognition performance and generalization of the method. The small errors of this method can be considered negligible for the overall performance, especially for complex surface defects in industrial products.

5.4. Comparative Study

In order to assess the performance of the proposed method, it was compared with the popular methods of R-CNN and YOLOv3 using the same image dataset. In relation to the R-CNN method, the regional proposals obtained by selective searching in the original image during the first step of the training procedure were as many as 2000. Moreover, CNN feature extraction and SVM classification [34] should be performed for each image. Therefore, these complex calculation procedures slowed down the detection of defects. As a method based on regression, YOLOv3 [35] has no region proposal mechanism but grid regression. This regression methodology provoked an imprecise positioning of the object. As a result, the detection accuracy of YOLOv3 was not very high. The proposed method equipped with the RPN employed an anchor with nine different bounding boxes to locate the defects. RPN can find many more defects of different lengths and shapes.

According to the information reflected in Tables 2 and 3, the method proposed in this research can be efficiently used for multi-class defect detection on the surface of wheel hubs ensuring optimal detection rates and fast detection speeds.

Table 2. Comparison of average precision.

	R-CNN	**YOLOv3**	**Ours**
Scratch	66.9%	70.03%	74.9%
Oil pollution	64.5%	67.67%	68.5%
Block	68.0%	70.52%	74.0%
Grinning	71.3%	72.33%	74.3%
mAP	67.7%	70.39%	72.9%

Table 3. Comparison of detection speed.

	R-CNN	**YOLOv3**	**Ours**
Test time per image	78 s	0.033 s	0.3 s

6. Conclusions

In the traditional CNN method, when a fixed sliding window is used to locate defects, it is difficult to determine the size of a window. Therefore, a method based on Faster R-CNN was proposed for detecting four kinds of defects (block, grinning, oil pollution, and scratches) on wheel hubs. Four hundred and two images (1440 × 1080 pixels) were collected. Data augmentation was accomplished by adding noise (Gaussian noise, gaussian blur, salt and pepper noise, and motion blur) to the original set of images. The resultant set of images was manually labeled. The training set, validation set, and testing set were generated by randomly selecting from these annotated images. In order to obtain the optimal detection accuracy, a trial and error method was adopted to set the initial parameters. In addition, the robustness of the network was verified by using 6 additional images. Furthermore, a comparative study was conducted with the popular methods R-CNN and YOLOv3.

For detecting and locating different kinds of defects, it is difficult to determine the advantages of each detection method because of the different training sets. However, it can be concluded that the structure of the proposed method based on network optimization has better computing efficiency, because RPNs can provide more flexible bounding boxes for different sizes of input images, and RPNs can efficiently and accurately generate regional proposals. Through sharing convolution features with downstream detection networks, the detection accuracy of the overall network can be improved.

Future detection methods based on this proposed method should improve the detection accuracy and robustness by using better quality images and wider shooting distances when building the image set. Finally, it is important to mention that Faster R-CNN can be certainly used to completely automate the detection of surface defects similar to those of the wheel hubs.

Author Contributions: Conceptualization, X.S.; data curation, X.S., R.H.; formal analysis, X.S., R.Z.; funding acquisition, J.G., X.S.; investigation, X.S., R.H.; methodology, X.S.; resources, X.S., R.Z.; software, X.S., R.H.; supervision, J.G., B.G.P.; writing—original draft, X.S., B.G.P.; writing—review & editing, X.S., R.H., J.G., B.G.P.

Funding: This research was funded by the National Natural Science Foundation of China (No. 51875266), Jiangsu Province Graduate Research and Innovation Program (No. KYCX18-2227).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hanzaei, S.H.; Afshara, A.; Barazandehb, F. Automatic detection and classification of the ceramic tiles'surface defects. *Pattern Recogn.* **2017**, *66*, 174–189. [CrossRef]
2. Bissi, L.; Baruffa, G.; Placidi, P.; Ricci, E.; Scorzoni, A.; Valigi, P. Automated defect detection in uniform and structured fabrics using Gabor filters and PCA. *J. Visual Commun. Image Represent.* **2013**, *24*, 838–845. [CrossRef]
3. Mohanty, S.P.; Hughes, D.P.; Marcel, S. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419–1430. [CrossRef]
4. Cha, Y.J.; Choi, W.; Suh, G.; Mahmoudkhani, S.; Buyukozturk, O. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Comput.-Aided Civ. Infrastruct. Eng.* **2017**, *33*, 731–747. [CrossRef]
5. Li, W.B.; Lu, C.H.; Zhang, J.C. A local annular contrast based real-time inspection algorithm for steel bar surface defects. *Appl. Surf. Sci.* **2012**, *258*, 6080–6086. [CrossRef]
6. Sun, X.H.; Gu, J.N.; Tang, S.X.; Li, J. Research Progress of Visual Inspection Technology of Steel Products–A Review. *Appl. Sci.* **2018**, *8*, 2195. [CrossRef]
7. Park, J.K.; Kwon, B.K.; Park, J.H.; Kang, D.J. Machine learning-based imaging system for surface defect inspection. *Int. J. Precis. Eng. Manuf-Green. Technol.* **2016**, *3*, 303–310. [CrossRef]
8. Gong, R.F.; Chu, M.X.; Wang, A.N.; Yang, Y.H. A fast detection method for region of defect on strip steel surface. *ISIJ Int.* **2015**, *55*, 207–212. [CrossRef]
9. Ortiz, A.; Bonnin-Pascual, F.; Garcia-Fidalgo, E.; Company-Corcoles, J.P. Vision-based corrosion detection assisted by a micro-aerial vehicle in a vessel inspection application. *Sensors* **2016**, *16*, 2118. [CrossRef] [PubMed]
10. Krummenacher, G.; Ong, C.S.; Koller, S.; Kobayashi, S.; Buhmann, J.M. Wheel defect detection with machine learning. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 1176–1187. [CrossRef]
11. Cha, Y.J.; Choi, W.; Buyukozturk, O. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Comput.-Aided Civ. Infrastruct. Eng.* **2017**, *32*, 361–378. [CrossRef]
12. Cong, R.; Lei, J.; Fu, H.; Cheng, M.M.; Lin, W.; Huang, Q. Review of visual saliency detection with comprehensive information. *IEEE Trans. Circuits Syst. Video Technol.* **2018**. [CrossRef]
13. Li, W.; Li, K.; Huang, Y.; Deng, X. A New Trend Peak Algorithm with X-ray Image for Wheel Hubs Detection and Recognition. In Proceedings of the 7th International Symposium, ISICA 2015, Guangzhou, China, 21–22 November 2015; pp. 23–31.
14. Gaidhane, V.H.; Hote, Y.V.; Singh, V. An efficient similarity measure approach for PCB surface defect detection. *Pattern Anal. Appl.* **2017**, *21*, 277–289. [CrossRef]

15. Hu, H.; Liu, Y.; Liu, M.; Nie, L. Surface defect classification in large-scale strip steel image collection via hybrid chromosome genetic algorithm. *Neurocomputing* **2015**, *181*, 86–95. [CrossRef]
16. Sun, Q.L.; Cai, J.H.; Sun, Z.Y. Detection of Surface Defects on Steel Strips Based on Singular Value Decomposition of Digital Image. *Math. Prob. Eng.* **2016**, *8*, 1–12. [CrossRef]
17. Yi, L.; Li, G.; Jiang, M. An end-to-end steel strip surface defects recognition system based on convolutional neural networks. *Steel Res. Int.* **2016**, *87*, 176–187. [CrossRef]
18. Liu, Z.; Hu, J.; Weng, L.; Yang, Y. Rotated region-based CNN for ship detection. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 900–904. [CrossRef]
19. Chen, H.; Pang, Y.; Hu, Q.; Liu, K. Solar cell surface defect inspection based on multispectral convolutional neural network. *J. Intell. Manuf.* **2018**, *4*, 1–16. [CrossRef]
20. Chen, J.; Liu, Z.; Wang, H.; Nunez, A.; Han, Z. Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 257–269. [CrossRef]
21. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 455–460.
22. Li, G.; Liu, J.; Jiang, C.; Zhang, L.; Lin, M.; Tang, K. Relief R-CNN: Utilizing convolutional features for fast object detection. In Proceedings of the Advances in Neural Networks–ISNN 2017, Sapporo, Hakodate, and Muroran, Hokkaido, Japan, 21–26 June 2017.
23. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
24. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef]
25. Liu, X.; Liu, Z.; Li, C.; Li, B.; Wang, B. Fabric defect detection based on faster R-CNN. In Proceedings of the Ninth International Conference on Graphic and Image Processing (ICGIP 2017), Qingdao, China, 14–16 October 2017. [CrossRef]
26. Zhang, J.; Xing, W.; Xing, M.; Sun, G. Terahertz image detection with the improved faster region-based convolutional neural network. *Sensors* **2018**, *18*, 2327. [CrossRef]
27. Xu, X.; Lei, Y.; Yang, F. Railway subgrade defect automatic recognition method based on improved Faster R-CNN. *Sci. Program.* **2018**, 4832972. [CrossRef]
28. Li, J.; Qu, C.; Shao, J. Ship detection in SAR images based on an improved faster R-CNN. In Proceedings of the 2017 SAR in Big Data Era: Models, Methods and Applications (BIGSARDATA), Beijing, China, 13–14 November 2017.
29. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional neural networks. In Proceedings of the 13th European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
30. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the International Conference on Learning Representations-ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
31. Eggert, C.; Brehm, S.; Winschel, A.; Zecha, D.; Lienhart, R. A closer look: Small object detection in faster R-CNN. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, China, 10–14 July 2017.
32. Shi, K.; Bao, H.; Ma, N. Forward Vehicle Detection Based on Incremental Learning and Fast R-CNN. In Proceedings of the 13th International Conference on Computational Intelligence and Security, Hong Kong, China, 15–18 December 2018.
33. Yang, T.; Long, X.; Sangaiah, A.K.; Zheng, Z.; Tong, C. Deep detection network for real-life traffic sign in vehicular networks. *Comput. Netw.* **2018**, *136*, 95–104. [CrossRef]

34. Chen, C.; Liu, M.Y.; Tuzel, O.; Xiao, J. R-CNN for small object detection. In Proceedings of the Computer Vision– ACCV 2016, Taipei, Taiwan, 20–24 November 2016.
35. Redmon, J.; Farhadi, A. Yolov3: An Incremental Improvement. Available online: https://pjreddie.com/media/files/papers/YOLOv3.pdf (accessed on 8 April 2018).