

Article

# An Intelligent Method for Detecting Surface Defects in Aluminium Profiles Based on the Improved YOLOv5 Algorithm

Teng Wang<sup>1</sup>, Jianhuan Su<sup>1,2,\*</sup>, Chuan Xu<sup>1</sup> and Yinguang Zhang<sup>1</sup>

<sup>1</sup> School of Automation, Guangxi University of Science and Technology, Liuzhou 545000, China; 221055206@stdmail.gxust.edu.cn (T.W.); 221068413@stdmail.gxust.edu.cn (C.X.); 221068426@stdmail.gxust.edu.cn (Y.Z.)

<sup>2</sup> School of Artificial Intelligence and Manufacturing, Hechi University, Yizhou 546300, China

\* Correspondence: 05003@hcnu.edu.cn

**Abstract:** In response to problems such as low recognition rate, random distribution of defects and large-scale differences in the detection of surface defects of aluminum profiles by other state-of-the-art algorithms, this paper proposes an improved MS-YOLOv5 model based on the YOLOv5 algorithm. First, a PE-Neck structure is proposed to replace the neck part of the original algorithm in order to enhance the model's ability to extract and locate defects at different scales. Secondly, a multi-streamnet is proposed as the first detection head of the algorithm to increase the model's ability to identify distributed random defects. Meanwhile, to overcome the problem of inadequate industrial defect samples, the training set is enhanced by geometric variations and image-processing techniques. Experiments show that the proposed MS-YOLOv5 model has the best mean average precision (*mAP*) compared to the mainstream target-detection algorithm for detecting surface defects in aluminium profiles, whereas the average single image recognition time is within 19.1FPS, meeting the real-time requirements of industrial inspection.



check for updates

**Citation:** Wang, T.; Su, J.; Xu, C.; Zhang, Y. An Intelligent Method for Detecting Surface Defects in Aluminium Profiles Based on the Improved YOLOv5 Algorithm. *Electronics* **2022**, *11*, 2304. <https://doi.org/10.3390/electronics11152304>

Academic Editors: Muhammad Salman Haleem, Liangxiu Han, Ernesto Iadanza and Baihua Li

Received: 14 June 2022  
Accepted: 19 July 2022  
Published: 23 July 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** surface defects of aluminium profiles; YOLOv5; MS-YOLOv5; PE-Neck; multi-streamnet

## 1. Introduction

As one of the most valuable materials in the industrial arena, aluminium profiles are inseparable from the aerospace and high-speed railway industries. However, due to external factors such as uneven production equipment and different standards of production processes, different types of defects can occur on the surface of aluminium profiles during the actual production process, affecting the service life of aluminium profiles. With the continuous development of artificial intelligence technology, the design of an accurate and fast detection method to address the needs of automated detection in industry is a key area for many researchers.

Metal defect detection methods have gone through three stages: manual based detection methods, single-mechanism-based detection methods, and machine vision-based classification methods. Manual inspection methods are influenced by subjective factors, resulting in low accuracy rates and the frequent occurrence of missed detections. The single-mechanism detection method mainly uses photoelectric technology, ultrasonic technology, and related devices to detect surface defects on aluminium profiles by using the acquired optical, electrical, and magnetic signals and ultrasonic waves. H. Khatun et al. [1] used piezoelectric transducers to inspect aluminium plates, proposing to use the finite element method (FEM) to obtain the intrinsic frequency response of aluminium plates and then use piezoelectric actuators to generate periodic excitation of aluminium plates, followed by using piezoelectric transducers to detect the mechanical impact response. Finally, the sensor response data is used to classify the defects by using the KNN classification algorithm. Leslie Bustamante et al. [2] proposed a non-contact method for the non-destructive detection of defects in aluminium plates, by using air-coupled ultrasound for identifying the size and location of defects. The defects are detected by the variation of their demonstrated

wave values. E. Ramírez-Pacheco et al. [3] proposed an eddy current inspection technique for the detection of defects in aluminium. By using a giant magneto resistance (GMR) sensor to detect surface defects in aluminium, a flat coil is used to generate the magnetic field, and it is proposed that the GMR output voltage depends on the width of the defect with a simple linear relationship to the depth. However, detection methods based on a single mechanism have limited accuracy and are costly and cumbersome to detect. The machine recognition-based classification method classifies aluminium profile defects in three main parts: image pre-processing, feature extraction, and defect classification. Features can be made visible through image pre-processing. For feature extraction, features such as texture, structure, colour, and shape can be extracted. The defect detection part is to discriminate and classify the extracted defect features by means of machine learning algorithms. Commonly used classification algorithms are support vector machines (SVMs) [4], Adaboost [5], and decision trees [6]. Kai Yan et al. [7], studied how to solve the problem of inaccurate classification due to the small size of metal weld defects. A completed local ternary pattern (CLTP) was proposed for weld-defect detection and an SVM classifier was used to classify the weld defects. S. R. Aghdam et al. [8] proposed the use of principal component analysis (PCA) and bagging (bootstrap aggregating) on the local binary pattern of operator extraction features in order to solve the problem of excessive time in the identification of steel surface defects, and used Decision Tree as a classifier for the classification of steel surface defects. F. Duan et al. [9] proposed an automated weld-defect detection system which first identifies potential defects by using the adaptive thresholding method of background subtraction. Next, the greyscale features and geometric properties of the defects are extracted. Finally, the defects are classified by using Adaboost. Although the problem of classifying defects in aluminium profiles can be solved by using machine vision, it is difficult to identify different types of defects at the same time when there are several different defects in a single picture and the classifier cannot accurately detect the exact location of the defects. In addition, these methods require the design of different feature extractors for different types of defects, and the accuracy of the recognition of defects depends entirely on the goodness of the designed feature extractor, and the actual production process is diverse, making it difficult to design a feature extractor that meets the reality, thus causing the disadvantage of poor generalisation ability.

Convolutional neural network (CNN)-based approaches are currently making great progress. Karen Simonyan et al. [10] proposed the VGG model, which performs well on all types of datasets through its regular design profile of stackable convolutional blocks. Alex Krizhevsky et al. [11] proposed AlexNet, proposed by using ReLU instead of Sigmoid as the activation function of CNN, and successfully solved the gradient dispersion problem of Sigmoid that occurs when the network is deeper; they used Dropout to randomly ignore some neurons to avoid model overfitting and proposed an LRN layer to suppress neurons with smaller feedback that enhances the generalization ability of the model. The method received extremely good results in the ImageNet 2012 competition. Kaiming He et al. [12] proposed ResNet, which solved the problem of gradient disappearance by using the residual module, and it effectively deepened the depth of the network, and many scholars have applied its ideas since then. Gao Huang et al. [13] proposed DenseNet, which uses dense block to reuse the feature maps of each layer to enhance the transfer of features in the network and improve the network performance while reducing the number of parameters in the network. Since then, a number of excellent target-detection algorithms have been proposed. These algorithms are mainly classified into single-stage detection algorithms and two-stage detection algorithms. Two-stage detection algorithms include Region-CNN (R-CNN) [14], Fast Region-based CNN (Fast R-CNN) [15], Faster Region-based CNN (Faster R-CNN) [16], and so on. Single-stage detection algorithms such as the You Only Look Once (YOLO) series [17–21] and single-shot multibox detector (SSD) [22] further enhance the performance of deep learning models in detection.

The aim of this research is to use target-detection algorithms in deep learning techniques to address the shortcomings of previous research in industrial defect detection, such

as the high cost of detection, cumbersome detection steps, the ability to classify defects and the inability to accurately detect the location of defects.

## 2. Related Work

In recent years, deep learning has been widely used in industrial product inspection. M. P. Muresan et al. [23] proposed to classify bushing locations on injection moulds of automotive parts by using Gaussian filtering to remove white noise from the images and converting the images into greyscale images, followed by using morphological methods to segment out the black holes in the fixed templates and by each boundary to extract the region of interest and subsequently classify the bushings by using LeNet-5. However, this method requires that the regions of interest are extracted according to a fixed template and the extracted images are fed into the network for classification, which is not applicable for detection objects without a template. The method is mainly suitable for classification, but cannot effectively classify the same image with multiple defects, and cannot predict the exact location of the defects itself. R. Usamentiaga et al. [24] used a target-detection algorithm to detect defects on the steel surface separately, and experimentally showed that YOLOv5 has better performance for the six defects on the steel surface compared to the rest of the target-detection algorithms. However, this study only demonstrates the detection performance of YOLOv5 and does not investigate how YOLOv5 can be improved to achieve better detection performance. Markus Schmitz [25] performed detection of defects generated by laser welding, first by using the Canny algorithm for detected edges, with training of Inception3 to obtain weights, and secondly by using mobilenet as the backbone network, by using SSD's object detector. Finally, they used the K-mean++ algorithm, which clusters the detection frame sizes that match the dataset. Experiments have shown that this method has good results, but its training steps are too cumbersome. Gao, H. et al. [26] proposed the use of deep convolutional adversarial production network (DCGAN) for data augmentation and CNN for classification of defects in order to solve the problem of small samples for industrial gear defect detection. Experiments have shown that the model has good results in classification, but it only does classification work, and no further research on detection is carried out. Jiang, Q. [27] et al. proposed an improved Faster R-CNN model for the detection of bearing surface defects by using ResNet-101 as a feature-extraction network, followed by a general method for improving positive samples, which was experimentally shown to have better performance than the base algorithm. He Di et al. [28] addressed the problem of steel surface sample images (mostly unlabelled images) and proposed a way to annotate images based on convolutional self-coding (CA) and semi-supervised generative adversarial network (SGAN), which can give an insight to the rest of the scholars and has solved the problem of deep learning to manually annotate labels. Xu, Y. et al. [29] proposed the YOLOv3 algorithm as a basis for detecting surface defects in aluminium profiles by adding a detection layer and using the k-means++ algorithm instead of the original k-means algorithm in the network for clustering analysis to obtain a more accurate anchor frame, thus improving the detection effect. However, only three categories of defects have been studied, and the types of defects are insufficient. Yongxiong Wang et al. [30] proposed a new network based on CNN for the problem of tiny defects in aluminium alloy castings. First, the X-ray images of aluminium alloy defects were obtained, and general feature network (GFN) and subtle feature network (SFN) were proposed to extract the image. The GFN and SFN are proposed to extract general features and subtle features of the images respectively, so as to achieve the purpose of detecting minor defects. Although the experiments show that the model has a good detection effect for aluminium alloy castings, it is difficult to obtain data about the X-ray defects, and the X-ray irradiated images reduce a lot of disturbing information, so it is not suitable for practical scenarios. Chen Song et al. [31] proposed an improved Faster-RCNN algorithm for the detection of five different categories of defects in order to solve the problem of missed detections in the detection of surface defects in aluminium tubes, firstly by expanding the resulting dataset through data augmentation, and secondly by proposing a new region method for the detection of defects in aluminium profiles. However, two-stage detection algorithms require more detection time than single-

stage detection algorithms and therefore may not meet the demands of industrial inspection in real time.

Based on the above work, it is demonstrated that it is feasible to apply target detection to industrial defect detection, but there are still problems with existing algorithms for aluminium, such as slower detection, fewer types of defects detected, and poor extraction of certain missing features, which leads to average detection performance. Therefore, this paper proposes an improved model based on YOLOv5, MS-YOLOv5, for the detection of seven aluminium profiles, which meets the need for real-time performance in industrial inspection while ensuring high detection capability. The work done in this paper is as follows.

- (1) We propose a PE-Neck by using poly-scale convolution (PSCConv) with efficient channel attention (ECA) to incorporate it into the appropriate position of the neck part of the original algorithm and change its structure. This is done to overcome the model's problem of extracting and locating defective features with too large a scale difference.
- (2) A multi-streamnet is proposed, borrowing the idea of pyramid convolution (PyConv) to change its calculation, adding residual connections and incorporating the first detection head of the original algorithm, thus improving the recognition of randomly distributed defects.
- (3) We intend to address the situation where industrial defect samples are small. In addition we propose data-augmentation techniques by using traditional geometric transformations for the training set, and image processing techniques are also used. This produces similar but different data to increase the size of the training set while reducing the model's reliance on certain features.

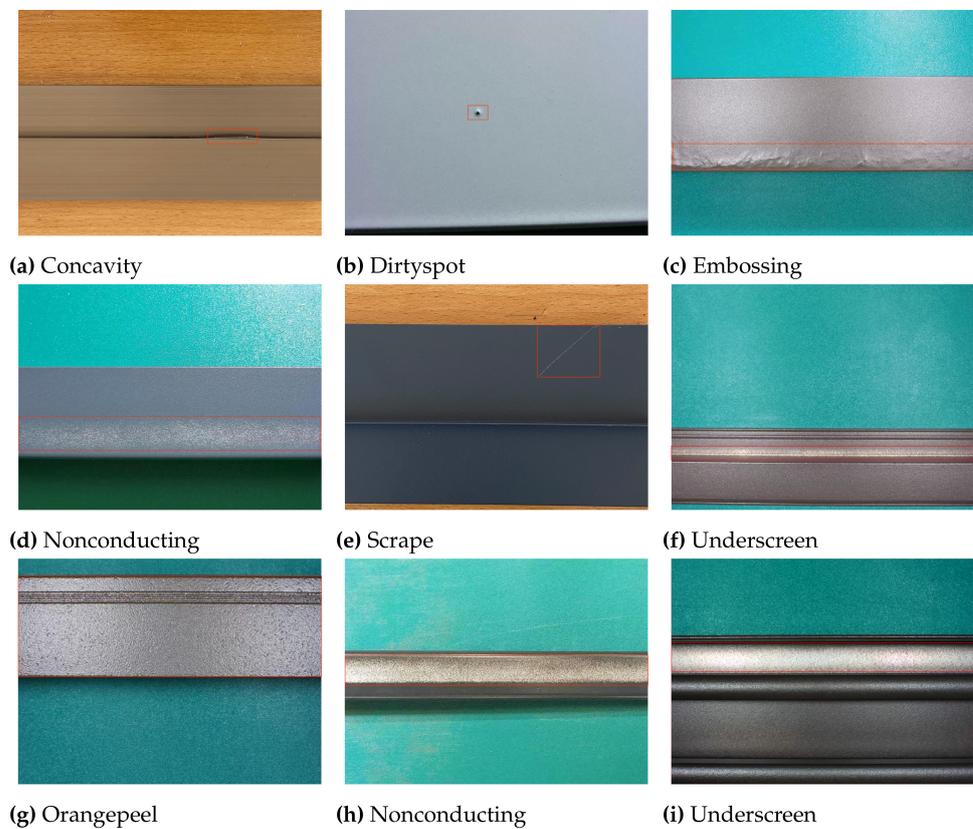
### 3. Materials and Methods

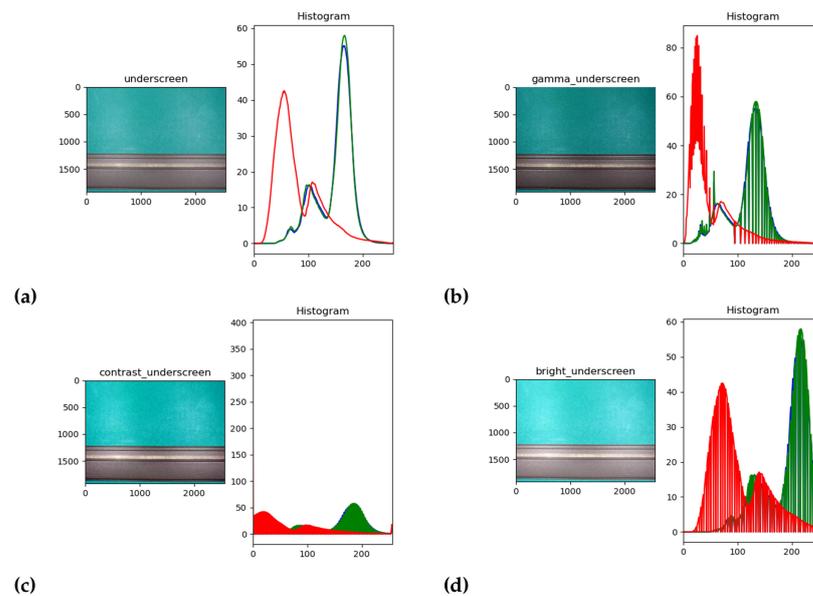
#### 3.1. Aluminium Profile Defect Dataset

The dataset in this paper is constructed from the Ali Tianchi database [32] together with the actual defect images produced by an aluminium profile factory in Guangxi. The dataset contains a total of 3098 images of seven types of defects such as Concavity, Dirtyspot, Orangepeel, Nonconducting, Scrape, Underscreen, and Embossing, all at a resolution of  $2560 \times 1920$ . The dataset described in this paper is shown in Figure 1. The acquired images are inadequate for the detection of surface defects in 7 types of aluminium profiles, and the lack of data can lead to problems such as overfitting of the training process, poor detection accuracy and poor generalisation, so we need to perform a data-enhancement strategy on the training set, which also uses gamma variation, contrast variation, and bright variation as image-processing techniques in addition to the traditional geometric variation, to produce similar but different data to increase the size of the training set while reducing the model's reliance on certain features. The final training and test sets were 7777 and 1987 images respectively, and the composition of the data for each type of defect in aluminium profiles is shown in Table 1. The enhanced RGB contrast histogram is shown in Figure 2.

**Table 1.** Dataset consisting of various classes of defects.

	Concavity	Dirtypspot	Scrape	Embossing	Underscreen	Nonconducting	Orangepeel
<b>Train</b>							
Original	185	143	158	135	178	185	127
H_flip	185	143	158	135	178	185	127
V_flip	185	143	158	135	178	185	127
HV_flip	185	143	158	135	178	185	127
Gamma	185	143	158	135	178	185	127
Contrast	185	143	158	135	178	185	127
Bright	185	143	158	135	178	185	127
Total				7777			
<b>Test</b>							
Original	320	256	216	320	277	278	320
Total				1987			

**Figure 1.** A partial dataset of aluminium profiles containing different types of defects.



**Figure 2.** RGB comparison histogram for different data enhancement methods. (a) Original image; (b) gamma variation image; (c) contrast variation image; (d) brightness variation image.

### 3.1.1. Gamma Variation

The same defect can appear in different feature states depending on the light or the angle of the image acquisition in a practical inspection environment. In image processing, gamma variation is a technique that enhances dark details by making nonlinear changes to an image. The equation for gamma change is shown in Equation (1):

$$g(x, y) = f(x, y)^\gamma \quad (1)$$

where  $f(x, y)$  represents the normalised grey value of row  $x$ , column  $y$ , and  $g(x, y)$  represents the grey value of row  $x$ , column  $y$  of the output image.

### 3.1.2. Contrast Variation and Brightness Variation

To make sure that the model can cope with different external variations, this paper uses contrast transformations and brightness variations to enhance the dataset. Contrast transformation is an image-processing method that improves the quality of an image by changing the contrast of the image elements. Brightness variations is used to simulate the characteristics reflected by the object being detected in high-intensity light.

## 3.2. Methods

### 3.2.1. MS-YOLOv5

The YOLOv5 algorithm is one of today's more advanced single-stage target detection algorithms, which can guarantee detection accuracy while spending less time on recognition, but it is not accurate enough to detect defects in aluminium profiles with random defect distributions and large-scale differences. Therefore, this paper proposes an improved model MS-YOLOv5 based on the YOLOv5 algorithm. The structure of the MS-YOLOv5 model is shown in Figure 3. The algorithm consists of three parts: backbone, neck, and detection. The backbone adopts the CSPDarknet-53 structure. The neck uses our proposed PE-Neck. Detection uses our proposed multi-streamnet as the first detection header.



The PSConv formula is shown in Equation (4):

$$\mathcal{H}_{c,x,y} = \sum_{K=1}^{C_{in}} \sum_{i=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{j=-\frac{K-1}{2}}^{\frac{K-1}{2}} \mathcal{G}_{c,K,i,j} \mathcal{F}_{K,x+iD_{c,K},y+jD_{c,K}} \quad (4)$$

where  $D \in R^{C_{out} \times C_{in}}$  denotes the matrix associated by the channels in the orthogonal dimension and the specific channels in the convolution. From the above equations, it can be seen that PSConv generates multi-scale kernels by adding feature factors to different kernels in the convolution, and that the different scale kernel calculations alternate by channel to process the information at different scales.

### 3.2.3. Efficient Channel Attention

Most studies in recent years have indicated that adding attentional mechanisms in convolutional neural networks can boost the performance of the model as a whole [34–36]. Most attention mechanisms obtain better performance by using more complex structures, which results in problems such as a larger network, more training time and longer inference times. By analyzing the SE-Net, ECA [37] found that the reduction in dimensionality does not allow for effective learning of channel information, thus leaving the overall network without better overall performance. Appropriate cross-channel interaction of information can reduce the complexity of the model while maintaining good performance. Therefore, a dimensionless local cross-information interaction strategy is proposed, which is mainly implemented by 1-dimensional convolution and an adaptive selection of 1-dimensional convolution kernels. A schematic of ECA is shown in Figure 5.

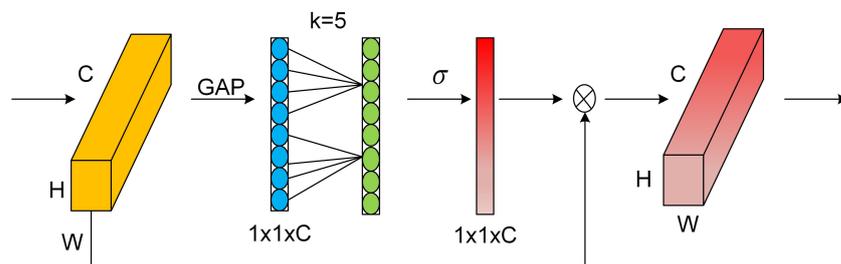


Figure 5. ECA structure schematic. Local cross-channel information interaction by 1D convolution of size  $k$ .

In order to be able to guarantee a good performance and a simple structure of the ECA module, we use  $W_k$  to denote the learned channel attention and avoid the complete independence of different groups. In performing the weight  $y_i$ , only the information interactions between that weight and its  $K$  neighbours are considered. The calculation formula is shown in Equation (5):

$$w_i = \sigma \left( \sum_{j=1}^k w_i^j y_i^j \right), y_i^j \in \Omega_i^k. \quad (5)$$

To further improve performance, so that all channels share weight information, the formula is shown in Equation (6):

$$w_i = \sigma \left( \sum_{j=1}^k w^j y_i^j \right), y_i^j \in \Omega_i^k. \quad (6)$$



### 3.2.5. Multi-Streamnet

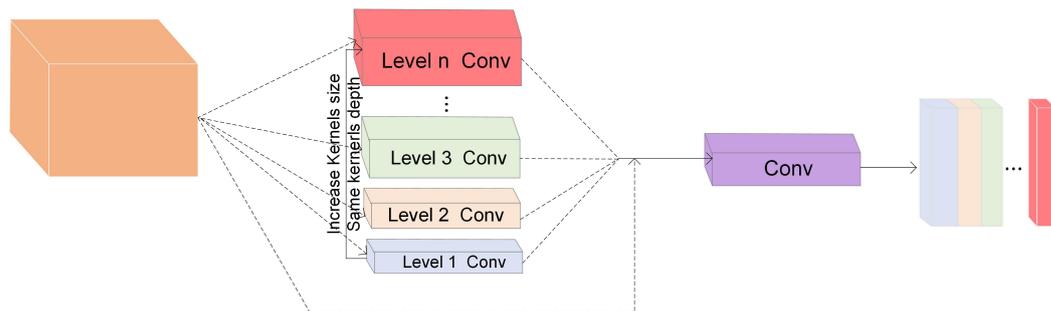
When the input image is convolved to extract features of the image, each kernel is responsible for capturing the local image, whereas larger kernels are able to capture a larger range of feature information. For example, PyConv [38], OctaveConv [39], Res2Net [40], ScaleNet [41], etc. can expand the perceptual field of the model mainly by using structures composed of different convolutional kernel sizes.

By analysing PyConv, we found that although the perceptual field of features can be increased by convolution of different sizes, convolution of different depths can lead to the generation of too much redundant feature information, which affects network inference and increases the difficulty of defect detection, and this unnecessary feature information also increases the number of parameters and computation. To address the above problems, we propose a multi-streamnet, which first, in order to reduce the amount of computation and the number of parameters, proposes to use the number of convolutions to control the depth of the convolution, so that the depth of the convolutions at different scales remains the same, and thus the number of feature maps obtained is the same. To facilitate understanding, if the input contains  $FM_i$  channels, and the size of each convolutional layer is  $K_1^2, K_2^2 \dots K_n^2$ , and the number of convolutions is  $n$ , then the depth is  $FM_i/n$ , and the corresponding feature dimensions are  $FM_{01}, FM_{02} \dots FM_{0n}$ , and its parametric quantity and computation formula are shown in Equations (9) and (10):

$$\left. \begin{array}{l} \text{parameters} = \\ K_n^2 \times FM_{0n} \times \frac{FM_i}{n} + \\ \vdots \\ K_3^2 \times FM_{03} \times \frac{FM_i}{n} + \\ K_2^2 \times FM_{02} \times \frac{FM_i}{n} + \\ K_1^2 \times FM_{01} \times \frac{FM_i}{n} + \end{array} \right\} \quad (9)$$

$$\left. \begin{array}{l} \text{FLOPs} = \\ K_n^2 \times FM_{0n} \times \frac{FM_i \times (W \times H)}{n} + \\ \vdots \\ K_3^2 \times FM_{03} \times \frac{FM_i \times (W \times H)}{n} + \\ K_2^2 \times FM_{02} \times \frac{FM_i \times (W \times H)}{n} + \\ K_1^2 \times FM_{01} \times \frac{FM_i \times (W \times H)}{n} + \end{array} \right\} \quad (10)$$

Each row represents the computational cost of a different convolution kernel size, in such a way that a certain number of feature images are generated while ensuring a certain amount of computation is reduced to reduce the complexity of the model. In addition, this paper borrows ideas from Resnet to use residual connectivity to learn redundant structures into constant mappings without performance degradation. These structures are eventually incorporated into the first detection head of the original algorithm to improve the model's ability to identify randomly distributed defects. A schematic of multi-streamnet is shown in Figure 7.



**Figure 7.** Multi-streamnet structure schematic. By using the same convolution depth for different scales of convolution kernels on top of PyConv, adding residual connections and incorporating detection layer convolution.

**4. Experimental Environment, Evaluation Indicators, and Model Training**

*4.1. Experimental Environment*

The experimental environment was defined as follows: CPU, Intel(R) Core(TM)i7-11700KF@3.60GHz 8-core CPU; GPU, NVIDIA GeForce RTX3080Ti; SSD, 1TB; programming languages, Python3.7; Framework, Pythorch 1.7.1; deep learning accelerator, CUDA 11.0 and CUDNN 11.1; IDE: Pycharm.

*4.2. Experimental Evaluation Indicators*

For the target detection algorithm, the performance metric used to evaluate the effectiveness of the model is *mAP* (mean average precision). *mAP* is used to measure the average recognition performance of the model over all categories and is given in Equation (11):

$$mAP = \frac{1}{N} \sum_1^N AP, \tag{11}$$

where *N* denotes *n* classifications and *AP* denotes the average accuracy rate, which is calculated as shown in Equation (12):

$$AP = \int_0^1 P(R)dR, \tag{12}$$

where *P* represents precision and *R* represents recall, which is calculated as in Equation (13):

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN}. \tag{13}$$

True positive (*TP*) means that the correct target type is predicted, false positive (*FP*) means that the predicted target type is wrong, and false negative (*FN*) means that the target that should be detected is missed. Thus, from this formula we can see that *P* and *R* should be negatively correlated in a model.

*4.3. Model Training*

YOLOv5 uses the idea of transfer learning in the training process and obtains pre-weights after testing with a large amount of data. MS-YOLOv5 continues the YOLOv5 training approach by using the initial weights YOLOv5x.pt with the best results to train the training set in the dataset. We set the hyperparameters before training to: learning rate, 0.001; Optimizer, stochastic gradient descent (SGD); momentum, 0.937; weight decay, 0.005; batch size, 8; epoch, 150. The target detection algorithm uses the training loss to determine whether the model is stable, and the loss of YOLOv5 is divided into box\_loss, obj\_loss and cls\_loss. The final model with three training losses is shown in Figure 8. We can see that after 60 calendar hours, all losses change minimally and the model is close to stable.

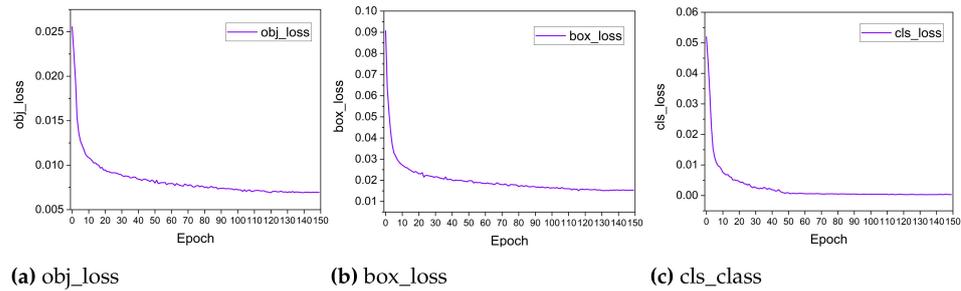


Figure 8. MS-YOLOv5 three types of training losses.

### 5. Results

#### 5.1. Validation of the MS-YOLOv5 Model

MS-YOLOv5 model was validated by adding P-Neck, PE-Neck and multi-streamnet to the YOLOv5 algorithm, respectively, and the results of each comparison step are shown in Table 2. Compared to the original YOLOv5 algorithm, the addition of P-Neck improved the recall by 2.5% and the *mAP* by 1.4%, thus demonstrating that the addition of P-Neck’s effectiveness. When PE-Neck was added in place of P-Neck, the precision and recall improved by 0.2% and 0.6%, respectively, compared to P-Neck, whereas *mAP* improved by 1.7%, thus demonstrating the effectiveness of adding PE-Neck. The inclusion of both PE-Neck and multi-streamnet resulted in a 1.2% increase in precision and a 0.2% increase in *mAP* compared to the inclusion of PE-Neck only. The experimental results show a 0.4% decrease in precision, a 1.1% increase in recall, and a 3.3% increase in *mAP* for MS-YOLOv5 compared to YOLOv5, with only a 1 FPS increase in detection time. The above sections of this paper have shown that precision and recall are negatively correlated in the model, with *mAP* being the most important. The MS-YOLOv5 *mAP* validation results for comparison is shown in Figure 9.

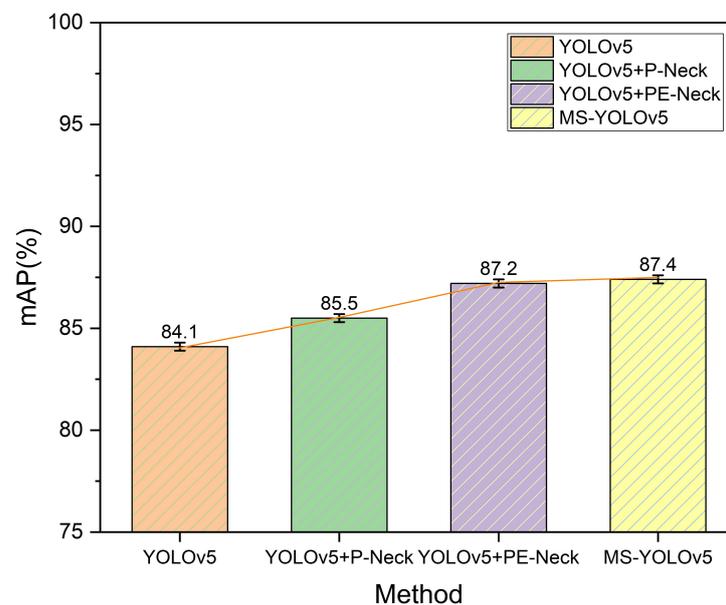
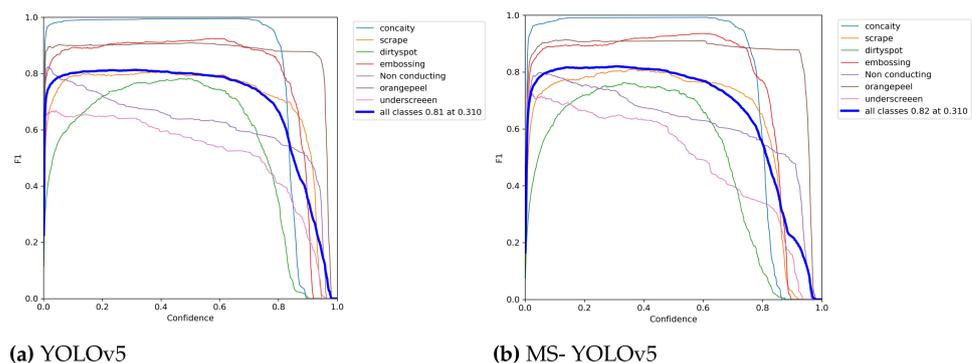


Figure 9. YOLOv5 improved to MS-YOLOv5 *mAP* comparison chart for each step.

**Table 2.** Performance comparison of various method.

Method	Precision(%)	Recall(%)	<i>mAP</i> (%)	FPS
YOLOv5	91.6	75.4	84.1	20.1
YOLOv5+P-Neck	89.8	77.9	85.5	19.8
YOLOv5+PE-Neck	90	78.5	87.2	19.5
YOLOv5 + PE-Neck + Multi-streamnet(MS-YOLOv5)	91.2	76.5	87.4	19.1

In order to visualize the performance of the MS-YOLOv5 model, we compare the performance of MS-YOLOv5 with YOLOv5 in terms of F1\_Cure, PR\_Cure, and actual detection results. F1\_cure is a metric that takes into account both the accuracy and recall of the model, and the higher the curve, the better the performance of the model. As shown in Figure 10, we can see that the F1\_Cure for MS-YOLOv5 is higher than that of YOLOv5 for all categories. PR\_Cure is a curve formed by taking the accuracy and recall of the model as coordinates, the area of which is the *mAP*. As shown in Figure 11, we can see that the area enclosed by MS-YOLOv5 is larger than that enclosed by YOLOv5. The actual inspection results are shown in Figure 12. The labels in the graph show the results and probabilities of detection. The first row shows the detection results for each defect with YOLOv5, and the second row shows the detection results for each defect with MS-YOLOv5. It can be seen that the two algorithms have similar detection results for well-defined surface defects in aluminium profiles such as Concavity, Orangepeel and Embossing. However, in the case of Scrape, which has a large variation in scale and is densely distributed, MS-YOLOv5 can detect it, whereas YOLOv5 cannot. In the case of Dirtyspot, which has a small and random distribution, YOLOv5 can only identify one defect, whereas MS-YOLOv5 can identify all of them. In the case of Nonconducting, which has a large difference in aspect ratio, YOLOv5 can only identify a single defect, whereas MS-YOLOv5 can identify them all. In defects such as Underscreen, where the colour is highly similar to the background colour, MS-YOLOv5 has a higher recognition level than YOLOv5. This shows that MS-YOLOv5 has a better detection performance than YOLOv5 and still meets the real-time requirements of industrial inspection. More actual test results are shown in Figure 13. The labels in the graph show the results and probabilities of detection.

**Figure 10.** F1-Cure comparison of YOLOv5 and MS-YOLOv5 for seven different defects.

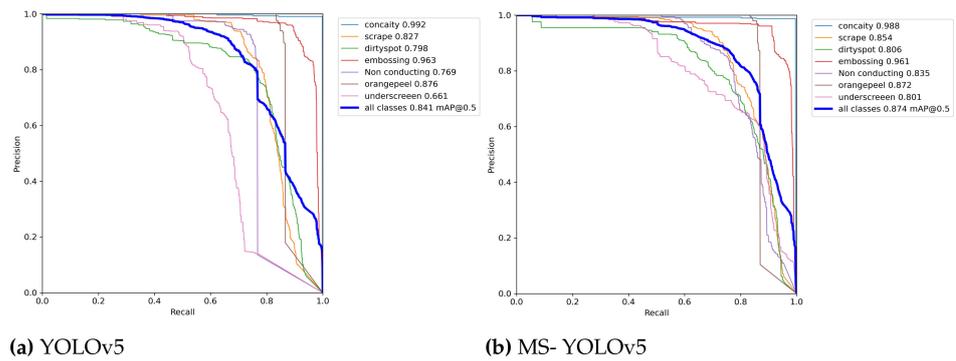


Figure 11. PR-Cure comparison of YOLOv5 and MS-YOLOv5 for seven different defects.

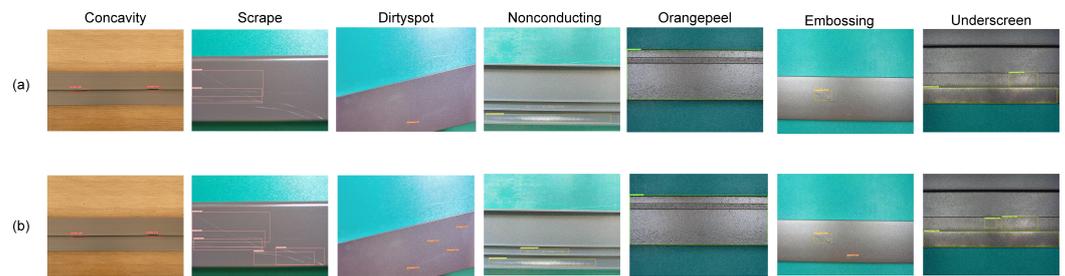


Figure 12. Comparison of YOLOv5 and MS-YOLOv5 results for seven defects. (a) Result of YOLOv5; (b) Result of MS-YOLOv5.

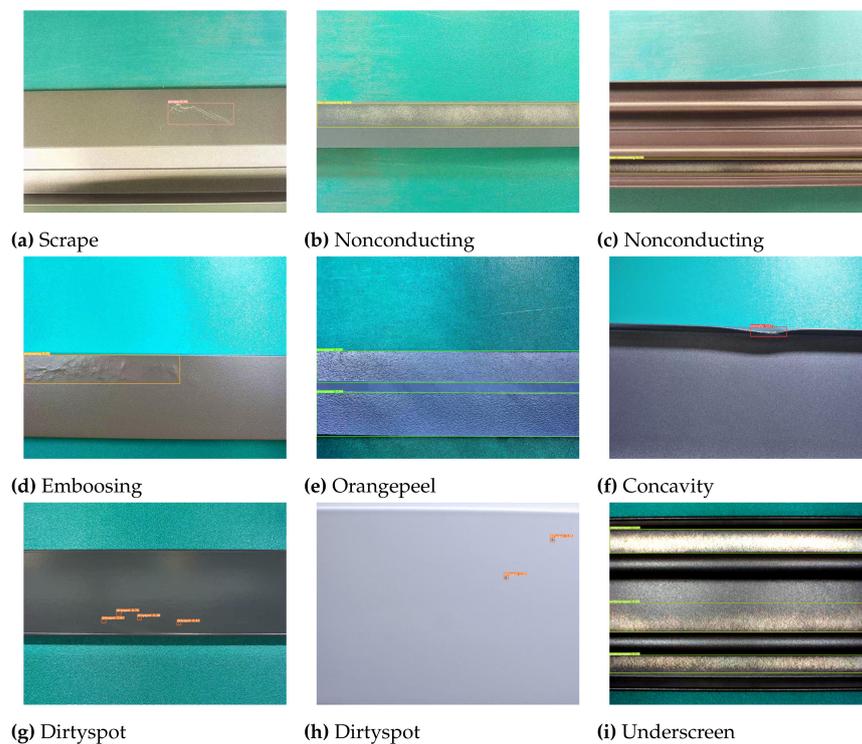


Figure 13. MS-YOLOv5 results for seven different defects.

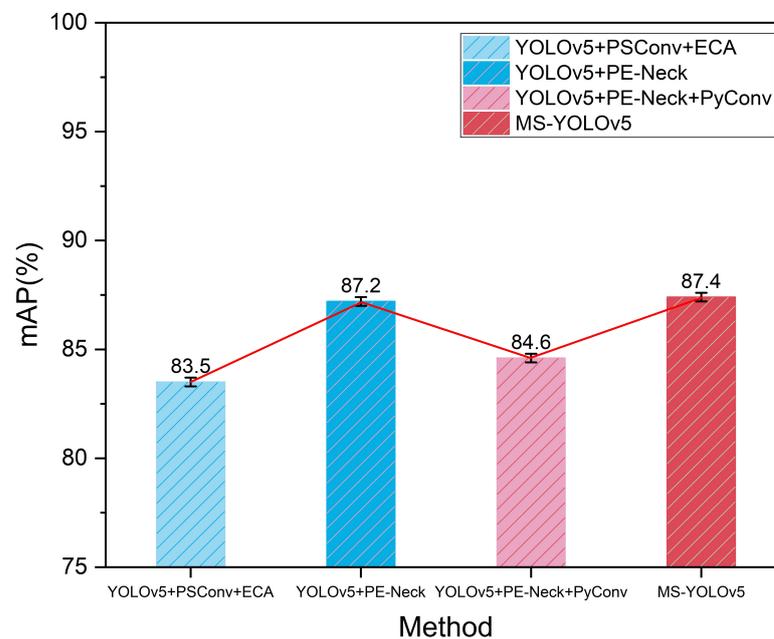
### 5.2. Ablation Comparison Experiments

In order to further verify the validity of the proposed structure, the unimproved original module (the same position as MS-YOLOv5) was added to YOLOv5 for comparison with MS-YOLOv5. The results obtained are shown in Table 3. Compared to the direct addition of the PConv and ECA modules to the YOLOv5 algorithm, the addition of PE-Neck alone improved accuracy and recall by 2.6% and 0.5%, respectively, increased

$mAP$  by 3.7%, and reduced detection time by 0.3 FPS. Compared to the direct addition of Pyconv to YOLOv5+PE-Neck, MS-YOLOv5's accuracy and  $mAP$  increased by 1.8% and 2.8% respectively, and reduced the detection time by 0.7 FPS. The results of the ablation experiment demonstrated the effectiveness of MS-YOLOv5 by improving both  $mAP$  and detection time compared to the original method. A comparative plot of  $mAP$  for the ablation experiments is shown in Figure 14.

**Table 3.** Performance comparison of ablation study.

Method	Precision(%)	Recall(%)	$mAP$ (%)	FPS
YOLOv5+PSConv+ECA	87.4	78	83.5	19.2
YOLOv5+PE-Neck	90	78.5	87.2	19.5
YOLOv5+PE-Neck+PyConv	89.4	77.7	84.6	18.4
YOLOv5 + PE-Neck + Multi-streamnet(MS-YOLOv5)	91.2	76.5	87.4	19.1



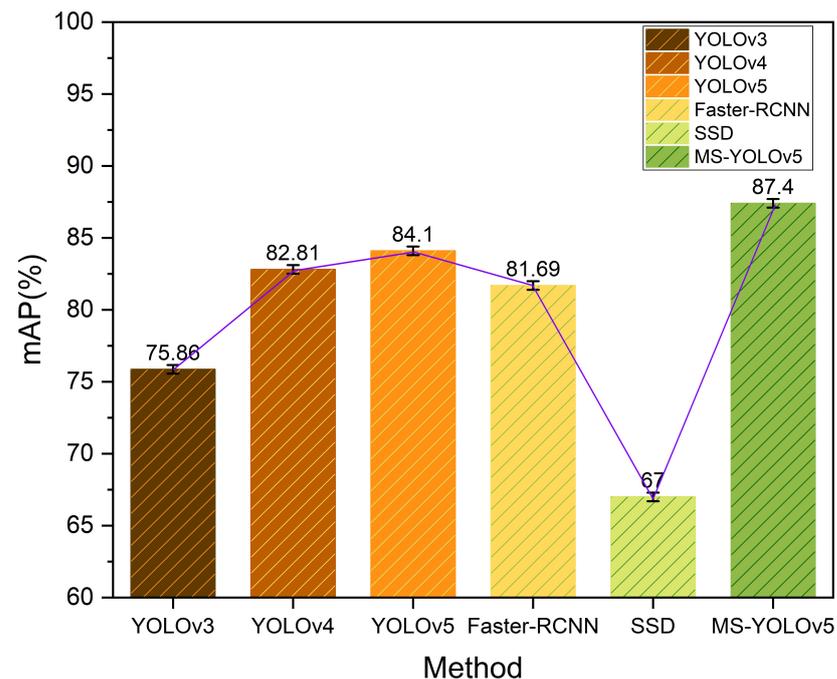
**Figure 14.** Comparison of  $mAP$  with the original module and with the improved module.

### 5.3. Experiments Comparing Different Algorithms

To further verify the effectiveness of MS-YOLOv5, its performance was compared with that of the YOLOv3, YOLOv4, SSD, Faster-RCNN, and YOLOv5 algorithms for the detection of surface defects in seven aluminium profiles, and the experimental results are shown in Table 4. It can be seen that among the detection results for each type of defect, MS-YOLOv5 is the best in terms of overall performance, with an  $AP$  of over 80% for each defect and no extreme imbalance in detection performance, whereas the other algorithms have extreme imbalance in detection performance for defects with large-scale differences, which is crucial in the actual detection process. MS-YOLOv5 achieved 87.4%  $mAP$ , 11.54% higher than YOLOv3, 4.59% higher than YOLOv4, 3.3% higher than YOLOv5, 20.4% higher than SSD, and 5.71% higher than Faster-RCNN. A comparison of the  $mAP$  of several algorithms is shown in Figure 15.

**Table 4.** Performance comparison of various algorithms.

Model	AP(%)							mAP(%)	FPS
	Concavity	Scrape	Dirty Spot	Embossing	Non Conducting	Orange Peel	under Screen		
YOLOv3	97.14	48.07	40.56	78.77	89.59	93.42	83.44	75.86	21.3
YOLOv4	97.71	74.43	51.05	81.15	94.69	96.42	84.25	82.81	20.4
YOLOv5	99.2	82.7	79.8	96.3	76.9	87.6	66.1	84.1	20.1
SSD	72.91	60.96	20.9	64.25	82.14	90.81	76.97	67	21.7
Faster-RCNN	92.14	70.74	38.7	94.03	90.36	97.40	88.49	81.69	12.6
MS-YOLOv5	98.8	85.4	80.6	96.1	83.5	87.2	80.1	87.4	19.1

**Figure 15.** Comparison chart of *mAP* using six different algorithms.

## 6. Conclusions

This paper proposes an improved MS-YOLOv5 model based on the YOLOv5 algorithm, proposes a PE-Neck by using PSConv with ECA and incorporating it into the appropriate position of the neck part of the original algorithm, and changing its structure in order to solve the problem of model extraction and localisation of defect features with too large a scale difference. Secondly, a multi-streamnet is proposed to improve the recognition of randomly distributed defects by changing its calculation by borrowing ideas from PyConv, adding residual connections and incorporating the first detection head of the original algorithm. At the same time, the problem of the lack of industrial samples is addressed by means of data augmentation. The experimental results show that MS-YOLOv5 achieves 87.4% detection for seven aluminium surface defects; compared to the mainstream target detection algorithm, the experimental results show that MS-YOLOv5 is the best in terms of overall performance, with an *AP* of over 80% for each defect, no extreme imbalance in detection performance, and a detection speed that meets the industrial inspection. However, it also has some problems, such as its sacrifice of some detection time. In future work, we will use knowledge distillation to reduce the size of the model as much as possible while guaranteeing model detection performance, thus reducing detection time, and place it in an embedded device.

**Author Contributions:** Conceptualization, T.W. and J.S.; methodology, T.W. and J.S.; software, T.W.; Data curation, C.X. and Y.Z.; validation, T.W. and J.S.; formal analysis, C.X. and Y.Z.; investigation, T.W. J.S., C.X., and Y.Z.; funding acquisition, J.S.; resources, J.S.; writing—original draft preparation, T.W.;

writing—review and editing, T.W and J.S.; visualization, T.W.; supervision, J.S.; project administration, J.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Innovation Project of Guangxi University of Science and Technology Graduate Education (GKYC202222); Special funding of horizontal projects of Hechi University (RH2000002165); Hechi University 2021 High-level Talent Research Initiation Project (2021GCC014).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** A part of the data used for this study is available in publicly available datasets, available online at <https://www.kaggle.com/datasets/tngw0702/defects-in-aluminium-profiles>(accessed on 1 February 2022). Another part of the data collected through the self-built system can be obtained from the corresponding author upon reasonable request.

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

## References

1. Khatun, H.; Hazarika, S.; Sarma, U. Aluminium Plate Surface Defect Detection and Classification based on Piezoelectric Transducers. In Proceedings of the 2021 IEEE 18th India Council International Conference (INDICON), Guwahati, India, 19–21 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
2. Bustamante, L.; Jeyaprakash, N.; Yang, C.H. Hybrid laser and air-coupled ultrasonic defect detection of aluminium and CFRP plates by means of Lamb mode. *Results Phys.* **2020**, *19*, 103438. [CrossRef]
3. Ramirez-Pacheco, E.; Espina-Hernandez, J.H.; Caleyó, F.; Hallen, J. Defect detection in aluminium with an eddy currents sensor. In Proceedings of the 2010 IEEE Electronics, Robotics and Automotive Mechanics Conference, Washington, DC, USA, 28 September 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 765–770.
4. Chen, P.H.; Lin, C.J.; Schölkopf, B. A tutorial on  $\nu$ -support vector machines. *Appl. Stoch. Model. Bus. Ind.* **2005**, *21*, 111–136. [CrossRef]
5. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
6. Rzepakowski, P.; Jaroszewicz, S. Decision trees for uplift modeling. In Proceedings of the 2010 IEEE International Conference on Data Mining, Sydney, Australia, 13 December 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 441–450.
7. Yan, K.; Dong, Q.; Sun, T.; Zhang, M.; Zhang, S. Weld defect detection based on completed local ternary patterns. In Proceedings of the International Conference on Video and Image Processing, Singapore, 27–29 December 2017; pp. 6–14.
8. Aghdam, S.R.; Amid, E.; Imani, M.F. A fast method of steel surface defect detection using decision trees applied to LBP based features. In Proceedings of the 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, 18–20 July 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 1447–1452.
9. Duan, F.; Yin, S.; Song, P.; Zhang, W.; Zhu, C.; Yokoi, H. Automatic welding defect detection of x-ray images by using cascade adaboost with penalty term. *IEEE Access* **2019**, *7*, 125929–125938. [CrossRef]
10. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
11. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*. [CrossRef]
12. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
13. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
14. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 23–28 June 2014; pp. 580–587.
15. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
16. Faster, R. Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *9199*, 2969239–2969250.
17. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
18. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
19. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
20. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.

21. Glenn Jocher, Alex Stoken, Jirka Borovec, NanoCode012. 2021. Available online: <https://github.com/ultralytics/yolov5/> (accessed on 15 November 2021).
22. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin, Germany, 2016; pp. 21–37.
23. Muresan, M.P.; Cireap, D.G.; Giosan, I. Automatic vision inspection solution for the manufacturing process of automotive components through plastic injection molding. In Proceedings of the 2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 3–5 September 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 423–430.
24. Usamentiaga, R.; Lema, D.G.; Pedrayes, O.D.; Daniel, G. Automated surface defect detection in metals: A comparative review of object detection and semantic segmentation using deep learning. *IEEE Trans. Ind. Appl.* **2022**, *58*, 4203–4213. [[CrossRef](#)]
25. Schmitz, M. Machine Learning in Industrial Applications: Insights Gained from Selected Studies. Ph.D. Thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Erlangen, Germany, 2022.
26. Gao, H.; Zhang, Y.; Lv, W.; Yin, J.; Qasim, T.; Wang, D. A Deep Convolutional Generative Adversarial Networks-Based Method for Defect Detection in Small Sample Industrial Parts Images. *Appl. Sci.* **2022**, *12*, 6569. [[CrossRef](#)]
27. Jiang, Q.; Tan, D.; Li, Y.; Ji, S.; Cai, C.; Zheng, Q. Object detection and classification of metal polishing shaft surface defects based on convolutional neural network deep learning. *Appl. Sci.* **2019**, *10*, 87. [[CrossRef](#)]
28. Di, H.; Ke, X.; Peng, Z.; Dongdong, Z. Surface defect classification of steels with a new semi-supervised learning method. *Opt. Lasers Eng.* **2019**, *117*, 40–48. [[CrossRef](#)]
29. Xu, Y.; Zhang, K.; Wang, L. Metal Surface Defect Detection Using Modified YOLO. *Algorithms* **2021**, *14*, 257. [[CrossRef](#)]
30. Wang, Y.; Hu, C.; Chen, K.; Yin, Z. Self-attention guided model for defect detection of aluminium alloy casting on X-ray image. *Comput. Electr. Eng.* **2020**, *88*, 106821. [[CrossRef](#)]
31. Chen, S.; Wang, D.G.; Wang, F.B. Detecting aluminium tube surface defects by using faster region-based convolutional neural networks. *J. Comput. Methods Sci. Eng.* **2022**, 1–10. Preprint.
32. Tianchi Data Sets. Available online: <https://tianchi.aliyun.com/dataset/> (accessed on 15 November 2021).
33. Li, D.; Yao, A.; Chen, Q. Psconv: Squeezing feature pyramid into one compact poly-scale convolutional layer. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; Springer: Berlin, Germany, 2020; pp. 615–632.
34. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
35. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
36. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 25 June 2021; pp. 13713–13722.
37. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
38. Duta, I.C.; Liu, L.; Zhu, F.; Shao, L. Pyramidal convolution: Rethinking convolutional neural networks for visual recognition. *arXiv* **2020**, arXiv:2006.11538.
39. Chen, Y.; Fan, H.; Xu, B.; Yan, Z.; Kalantidis, Y.; Rohrbach, M.; Yan, S.; Feng, J. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3435–3444.
40. Gao, S.H.; Cheng, M.M.; Zhao, K.; Zhang, X.Y.; Yang, M.H.; Torr, P. Res2net: A new multi-scale backbone architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 652–662. [[CrossRef](#)] [[PubMed](#)]
41. Li, Y.; Kuang, Z.; Chen, Y.; Zhang, W. Data-driven neuron allocation for scale aggregation networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11526–11534.