ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

Steel surface defect detection based on improved YOLOv8 neural network

Siyi Wu[1,a], Jiarui Li[2,b], Zhongyi Zhao[1,c], Zihao Wang[3,d], Junxi Li[4,e,*]

[1]*Nanjing University of Science and Technology, Nanjing, China*
[2]*Xian University of Technology, Xi'an, China*
[3]*Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russian Federation*
[4]*Herzen University, St. Petersburg, Russian Federation*
[a]*chownate623@163.com,* [b]*lijiarui0702@163.com,* [c]*18391767001@163.com,* [d]*van39.ts@edu.spbstu.ru,* [e]*lijunxi111@yandex.com*
[*]*Corresponding author*

Abstract: *In this study, we use a neural network approach instead of the traditional manual approach to solve the problem regarding the detection of defects on steel surfaces. We introduce several attention mechanisms to improve the Yolo v8 neural network trained on a steel defect detection dataset. The results show that our improved Yolo v8 model improves the robustness of the model more significantly and can detect more detailed steel surface defects.*

Keywords: *Attention Mechanism; Object Detection; Steel Surface Defects*

1. Introduction

The steel industry occupies a pivotal position in the global economy and is regarded as one of the pillars of modern industry [1]. As well as providing key materials for infrastructure construction, manufacturing, the automotive industry and other sectors, it is a strong guarantee of economic development and national defence. Steel's strength, durability and versatility make it a material that is used in a wide range of applications such as building bridges, manufacturing machinery and equipment, and producing household appliances and means of transport.

The quality of the steel surface plays a crucial role in the production and use of steel. Defects on the surface of steel are mainly scratches, pits, cracks and oxidized skin, which can lead to degradation of material properties, affecting the aesthetics, corrosion resistance and mechanical properties of the product. Particularly in high-precision and demanding industries such as aerospace, automotive and power generation, small defects in surface quality can have a direct impact on product reliability and safety.

Surface defects may affect steel properties and service life. In terms of mechanical properties, surface defects reduce the strength, toughness and hardness of steel, weakening its ability to withstand external and fatigue stresses. In terms of corrosion resistance, rust spots and areas of surface failure are more susceptible to environmental attack, leading to accelerated corrosion of the material, especially when wet [2]. The rate of corrosion is further accelerated in acid and salt spray environments. In terms of fatigue life, small defects such as cracks and scratches can significantly reduce the fatigue life of steel under repeated mechanical stress. In terms of appearance and aesthetics, surface defects can affect market competitiveness and consumer acceptance of products with stringent appearance requirements.

The most used methods for detecting surface defects on steel are manual inspection and traditional vision inspection. Manual inspection was the earliest method of detecting defects on steel surfaces and relied on workers using the naked eye or simple tools to inspect steel surfaces for defects. Workers can make empirical judgements about complex defects with some flexibility [3]. However, some test results are too dependent on the experience and skill of the operator, making it difficult to ensure consistency and reliability. In mass production, manual inspection cannot keep up with the demands of the production line, especially during long hours of continuous work. Fatigue can easily lead to missed inspections, it is often difficult to detect small defects, and the use of manual inspection can lead to increased labor costs.

Traditional vision-based inspection methods capture images of the steel surface from a camera and

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

then use pre-defined algorithms such as edge detection and threshold segmentation algorithms to identify surface defects. Traditional image processing inspection methods are related to manual inspection, the degree of automation is higher, can improve the detection speed, can detect the more obvious defects, applicable to the appearance of the inspection. However, traditional vision detection methods are less effective at detecting complex defects. And the noise immunity is poor, there is a lack of adaptivity, and there will be a high rate of false detections and missed detections.

We have improved the YOLO v8 algorithm by introducing new modules, optimizing the loss function and enhancing the feature extraction module. The improved model is better able to cope with the diverse characteristics of steel surface defects and excels in identifying complex morphology and small defects [4]. By improving the YOLO v8 algorithm, it not only provides a more efficient and accurate solution for detecting steel surface defects, but also provides a practical direction for improving the shortcomings of traditional detection methods, which is expected to achieve a wide range of applications in quality inspection in the iron and steel industry.

2. Dataset of Steel Surface Defects

Data on rail surface defects typically include information on the type, location, size and severity of the defect. This data can be obtained through a variety of inspection equipment and techniques, such as laser scanners, high-definition cameras, and so on. These data are important for assessing the condition of rails, developing maintenance programmers and ensuring the safety of railway transport. By analysing and processing these data, accurate detection and classification of rail surface defects can be achieved, helping to improve the efficiency and safety of rail maintenance.

The dataset we have collected comes from two main sources. The first part comes from the different stages of the steel production process, such as ironmaking, steelmaking and steel rolling, and collects data on the testing of steel samples at these stages. The second part comes from data on substandard products from several steel producers. The two sets of data together form the final image, which contains 4688 images of steel with surface defects [5]. Images of four different degrees and types of steel surface defects in the dataset are shown in Figure 1.

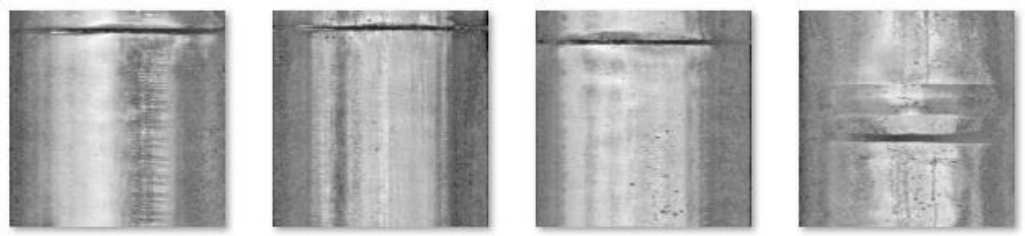

Figure 1: Four different types of steel surface defects

We obtain 4 classes of defective datasets that can be used for target detection tasks and are suitable for yolov8 algorithm model training tasks.

3. Improvement of Yolo v8 Neural Network

We have made changes to the Yolo v8 model structure, in three parts. The first part is the introduction of the attention mechanism [6]. The second part is the introduction of the improved pyramid structure of the void space [7]. The third part is the introduction of the CoT3 module incorporating the Transformer design [8].

The Pooling Layer is one of the important components in CNNs to reduce the spatial size of the input data and extract key features. In CNNs, the pooling layer is usually followed by the convolutional layer, and its main function is to perform spatial down sampling, i.e., to reduce the number of parameters in the model by reducing the size of the feature map.

The pooling layer achieves this by performing aggregation operations within each local region. The common operations of the pooling layer are Max Pooling and Average Pooling. Max Pooling takes the maximum value of a local region as the output of that region, while Average Pooling takes the average value of a local region as the output of that region. Figure 2 shows the results of maximum pooling and average pooling.

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

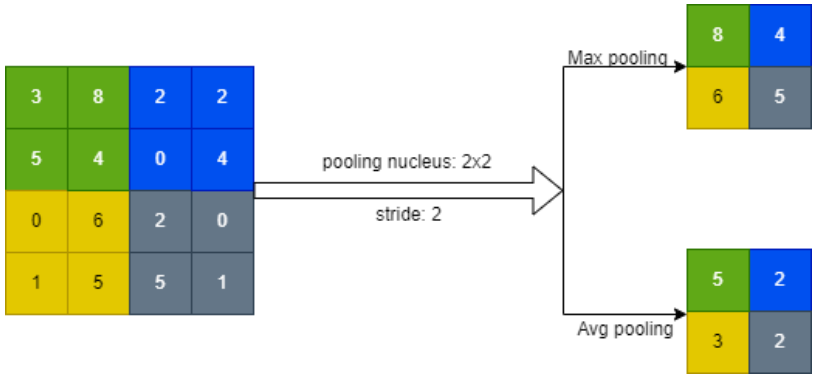

Figure 2: Maximum and average pooling

As shown in Figure 2, the size of the input feature map is 4 x 4, the size of the pooling kernel is 2 x 2, and the step size is 2. In the Maximum Pooling operation, the pooling kernel is first located in the green region, and after selecting the maximum value of 8, the pooling kernel moves two steps to the right, enters the blue region, and takes the maximum value of 4, and so on until the entire pooling operation is completed. Average Pooling follows the same steps as Maximum Pooling, except that the maximum value is changed to the average of the 2 x 2 pooling kernels.

SE (Squeeze and Excitation) Attention Mechanism, CA (Coordinate Attention) Attention Mechanism and CBAM (Convolutional Block Attention Module) Attention Mechanism are widely used [9]. Proper use of the attention mechanism allows the model to focus on important information related to steel surface defects and suppress irrelevant information that interferes with the detection of steel surface defects. In this paper, we improve the YOLOv8 model by introducing the three popular attention mechanisms mentioned above. The SE attention mechanism can be trained to automatically learn the importance of each channel and assign different weights to the channels in the network.

$$g_C = F_s(u_C) = \frac{1}{HxW}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j),\ g \in R^C \tag{1}$$

W and H denote the width and height of the feature map, and C denotes the number of channels. The size of the input feature map is WxHxC. The SE attention mechanism works by first performing a compression operation on the feature map obtained by convolution to obtain global features for each channel. Excitation operations are then performed on the global features to learn the interrelationships between each channel and to obtain the weights of the different channels. Finally, the initial feature values are multiplied to get the final feature values. The structure of the SE attention module is shown in Figure. 3, where the formula for compression is shown in 1. F_s represents the compression operation, u represents the input information, C represents each channel, H represents the height of the feature image for a single feature channel, W represents the width of the feature image for a single feature channel, and $u_c(i, j)$ represents the value of each point on the feature mapping channel.

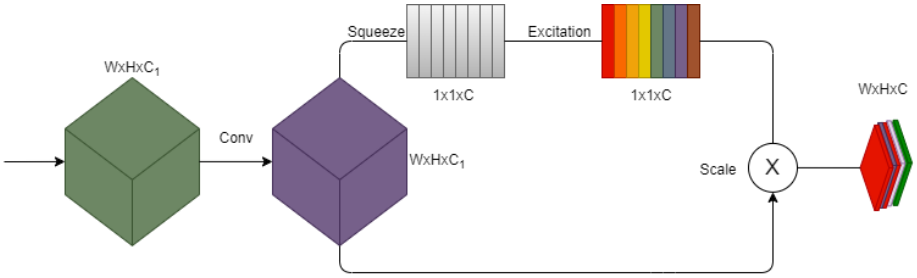

Figure 3: SE Attention Mechanism Structure Diagram

After compression, the excitation is computed as in Equation 2, where Fe represents the excitation operation, σ is the ReLU activation function, g represents the one-dimensional matrix obtained from Fe, W1, W2 are the fully connected layers, and r represents the hyper-parameter, which is 16 by default, and denotes the dimensionality reduction factor of the first fully connected layer.

$$t = F_e(g, W) = \sigma\left(W_2 Sigmoid(W_1 g)\right) \tag{2}$$

The SE attention mechanism is embedded in the backbone part of the network after each convolutional layer. This is because the backbone network is responsible for extracting features from

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

the input image that are analysed and extracted in subsequent layers. By introducing the SE attention mechanism into the backbone network, the network can pay more attention to the important features, thus improving the feature representation and helping to better capture the visual features of the lung nodules.

The CA attention mechanism, which not only captures the relationship between channels, but also captures the information related to the position, thus the CA attention mechanism can enhance the model's ability to perceive different positional information, thus improving the performance of the target detection task [10]. As shown in Figure 4, the CA module uses two one-dimensional global pooling operations (X: Horizontal Global Pooling, Y: Vertical Global Pooling) to aggregate the input features vertically and horizontally into two independent orientation-aware feature maps. The two feature maps embedded with directional information are then encoded into two attention maps, each of which captures the remote dependence of the input feature map in spatial direction. Finally, these two attention maps are multiplied by the input feature map to obtain the final attention feature, which effectively improves the feature map representation.

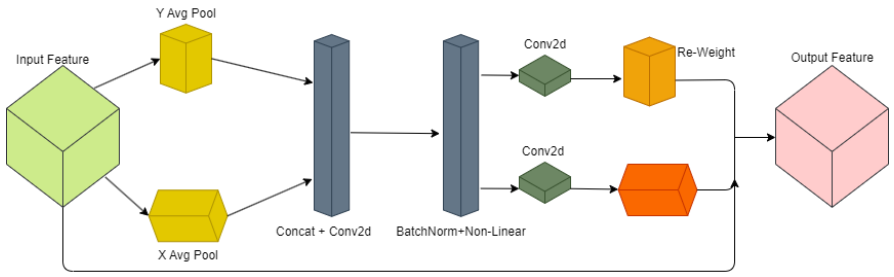

Figure 4: CA Attention Mechanism Structure Diagram

Since the CA attention mechanism is designed to solve the long-range feature dependency problem by capturing the spatial relationship between different locations in the feature map, and ultimately improve the model's ability to perceive the global information, this paper embeds the CA attention module in the larger layers of the feature map, which helps to improve the model's ability to understand the global information, and thus better capture the contextual information of the defective images of steel surfaces and the long-range dependency relationship.

CBAM is a simple and effective attention mechanism for feed-forward convolutional neural networks. CBAM enhances the perceptual and expressive capabilities of the network, thereby improving performance and generalisation. As shown in Figure 5, CBAM consists of two modules: Channel Attention Module (CAM) and Spatial Attention Module (SAM).

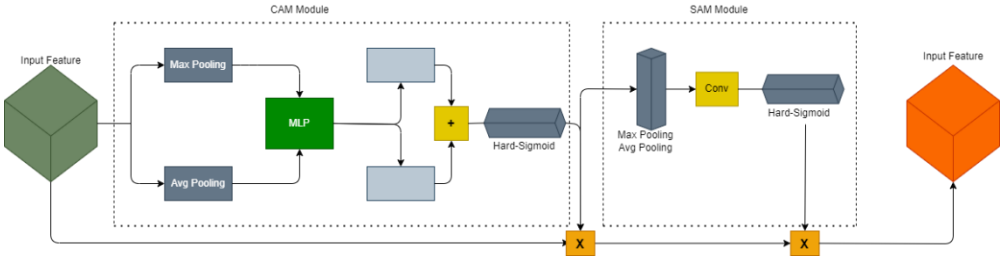

Figure 5: CBAM Attention Mechanism Structure Diagram

Among them, the channel attention module performs average pooling and maximum pooling on the input feature map $F \in R_{Cx1x1}$ at the same time, and then passes the merged features to a shared multi-layer perceptron with one hidden layer, and uses the Sigmoid activation function to add and activate the resulting features to generate a channel attention map $M_C \in R_{Cx1x1}$, as shown in Formula 3, where M_C represents the channel compression weight matrix, F represents the input feature map, σ represents the Sigmoid activation function, MLP represents the shared multi-layer perceptron, AvgPool represents the average pooling operation, and MaxPool represents the maximum pooling operation.

$$M_c(F) = \sigma\left(MLP\left(AvgPool(F)\right) + MLP(MaxPool(F))\right) \quad (3)$$

The improved channel attention F‘ is obtained by assigning the channel weights learnt by M_c to the different channels of F’. As shown in Equation 4, F’ represents the feature map for channel attention selection and ⊗ represents the matrix multiplication.

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

$$F' = M_c(F) \otimes F \tag{4}$$

In SAM, average pooling and maximum pooling operations are performed on F', and then the collected features are subjected to a 7x7 convolution operation to perform the activation of σ. The computation of the spatial attention map $M_s \in R_{1xHxW}$ is shown in Equation 5, where f_{7x7} represents the 7x7 convolution operation and M_s represents the spatial compression weight matrix.

$$M_s(F') = \sigma(f^{7x7}(AvgPool(F'); Max(Pool(F')))) \tag{5}$$

Finally, M_s is multiplied with F' to obtain the feature map F'' processed by the CBAM module.

$$F'' = M_c(F') \otimes F' \tag{6}$$

The global maximum pooling and global average pooling used in CAM complement each other, which can effectively extract compressed information. In SAM, 7×7 convolutions are used instead of traditional multiple 3×3 convolutions, because the former can effectively expand the receptive field and better obtain spatial information.

Transformers with self-attention mechanisms have created a sensation in the field of artificial intelligence [11]. However, most existing designs use self-attention directly on 2D feature maps to obtain attention matrices based on independent queries and key pairs at each spatial location, and do not take full advantage of the rich contextual information between neighbouring keys. The Context Converter (CoT) module is designed to make full use of the dynamic context information between adjacent keys, which is combined with the convolutional static context information and fused to the output. The CoT module absorbs the advantages of the traditional CNN and the Transformer, where the CNN captures the raw information of the input features, and the Transformer captures the global information of the input features.

The structure of the CoT module is shown in Figure 6. Calculations are shown in equations 7,8,9. Firstly, a K × K convolution operation is performed on K to obtain K with local context information, denoted by K_1. After that, a concatenation operation is performed on K_1 with Q and two consecutive 1 × 1 convolution operations are performed.

$$Q = X, K = X, V = XW_v \tag{7}$$

$$K^1 = [K, K]W_\theta \tag{8}$$

$$O = [K^1, Q]W_\theta W_\delta \tag{9}$$

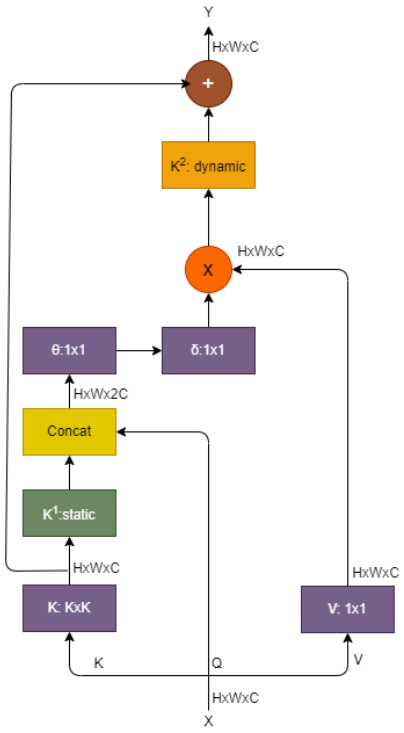

Figure 6: CoT module structure diagram

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

Multiply O by V to obtain the dynamic context information K^2. The calculation is shown in Formula 10, where X represents the input data, H, W, C represent the height, width and number of channels of the input data, Q, K, V represent the query vector, key vector and value vector in the self-attention mechanism, θ represents a 1x1 convolution with an activation function ReLU, δ represents a 1x1 convolution, and ⊗ represents matrix multiplication.

$$K^2 = V \otimes O \tag{10}$$

Eventually, the local static context information K^1 and global dynamic context information K^2 are fused as a result for output.

4. Evaluation and Results

To verify the performance of the model through experiments, we evaluate the experimental results according to three aspects: Accuracy, Sensitivity and mAP. The calculations are shown in equations 11, 12, 13 and 14. TP represents the number of true positive samples that are judged as positive samples, FP represents the number of negative samples that are misclassified as positive samples, FN represents the number of true positive samples that are judged as negative samples, AP represents the average precision of a given class of targets, AP_i represents the AP of class i, n represents the total number of classes, and p(r) represents the average precision obtained with the recall value.

$$precision = \frac{TP}{TP+FP} \tag{11}$$

$$sensitivity = \frac{TP}{TP+FN} \tag{12}$$

$$AP = \int_o^1 p(r)dr \tag{13}$$

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \tag{14}$$

In this study, the level of Precision represents the proportion of true positive nodes among all predicted positive nodes, the level of Sensitivity represents the proportion of the number of correctly predicted positive nodes among the total number of true positive nodes, and the mAP represents the mean accuracy of detection, with a higher mAP indicating a better detection performance of the model.

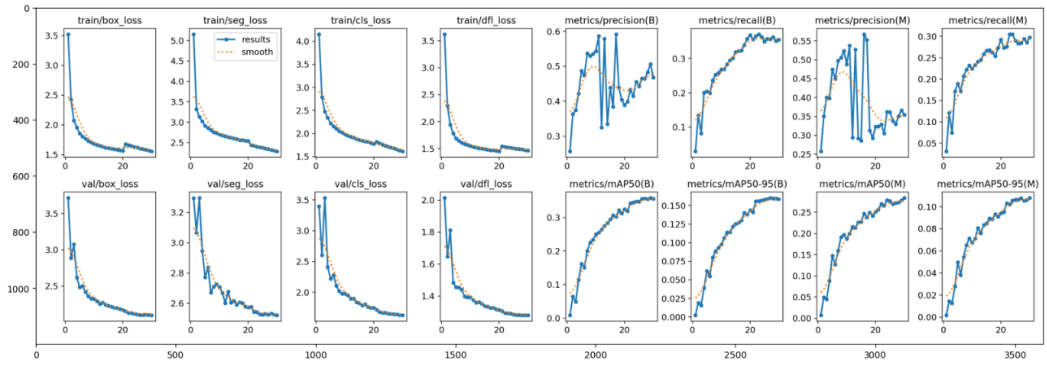

Figure 7: Training results on the steel defects dataset

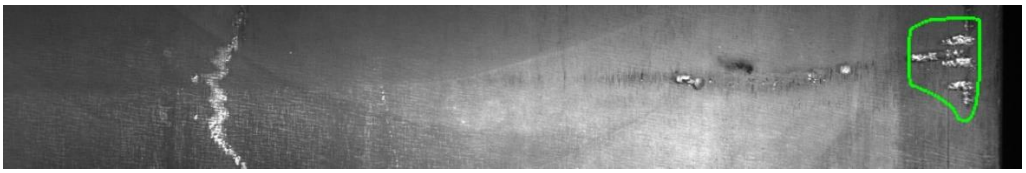

Figure 8: Effectiveness in detecting surface defects on steel

As shown in Figure 7 and Figure 8, when we improve the structure of the Yolo v8 network, the accuracy of the model can reach 97,6%, which is an improvement of 11% compared to the original model, and there is a significant improvement in the robustness of the model, and the defects on the surface of the steel can be detected in a more detailed way.

ISSN 2616-5880 Vol. 5, Issue 3: 110-116, DOI: 10.25236/AJMC.2024.050317

5. Discussion and Conclusions

The background in the steel defect detection image is complex, and there are other background interferences besides the steel defects, resulting in low precision measurement accuracy. Also, the robustness of the model may be affected under conditions such as dealing with complex scenes and large brightness variations. In addition, smaller steel surface defects in the image are difficult to detect accurately, because small targets may lose detail on lower resolution feature maps, making detection difficult.

To enhance the model's ability to suppress irrelevant information around steel surface defects, this study compares three popular attentional mechanisms, explores the superiority of each attentional mechanism, and optimises the overall network structure to improve the model's ability to pay attention to information about steel surface defects.

However, the robustness of our model has not been verified in industrial scenarios, and we need to continue to improve the model structure and optimise the loss function and optimiser after comparing it with different detection algorithms before putting it into industrial scenarios for testing.

References

[1] Chen Wanzhi, Zhang Chunguang. Improved YOLOv5 model for detecting surface defects on steel strip. Journal of Liaoning Technical University (Natural Science Edition)/Liaoning Gongcheng Jishu Daxue Xuebao (Ziran Kexue Ban), 2024, 43(3).

[2] Jain, S.; Seth, G.; Paruthi, A.; Soni, U.; Kumar, G. Synthetic data augmentation for surface defect detection and classification using deep learning. J. Intell. Manuf. 2020, 33, 1007–1020.

[3] Luo, Q.; Fang, X.; Su, J.; Zhou, J.; Zhou, B.; Yang, C.; Liu, L.; Gui, W.; Lu, T. Automated Visual Defect Classification for Flat Steel Surface: A Survey. IEEE Trans. Instrum. Meas. 2020, 69, 9329–9349.

[4] Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; IEEE: Piscatevi, NJ, USA, 2005; 1, pp. 886–893.

[5] Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. 2010, 32, 1627–1645.

[6] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. Commun. ACM 2017, 60, 84–90.

[7] He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2015, 37, 1904–1916.

[8] Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.

[9] Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. 2015, 28.

[10] Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.

[11] Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Los Alamitos, CA, USA, 18–22 June 2018; pp. 6154–6162.