processes

MDPI

Article

Defect Detection Algorithm for Battery Cell Casings Based on Dual-Coordinate Attention and Small Object Loss Feedback

Tianjian Li [1], Jiale Ren [1], Qingping Yang [2,*], Long Chen [1] and Xizhi Sun [2]

[1] School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China; litianjian99@163.com (T.L.); renjjj111@163.com (J.R.); cl@usst.edu.cn (L.C.)
[2] College of Engineering, Design and Physical Sciences, Brunel University London, Kingston Lane, Uxbridge UB8 3PH, UK; xizhi.sun@brunel.ac.uk
* Correspondence: qingping.yang@brunel.ac.uk

Abstract: To address the issue of low accuracy in detecting defects of battery cell casings with low space ratio and small object characteristics, the low space ratio feature and small object feature are studied, and an object detection algorithm based on dual-coordinate attention and small object loss feedback is proposed. Firstly, the EfficientNet-B1 backbone network is employed for feature extraction. Secondly, a dual-coordinate attention module is introduced to preserve more positional information through dual branches and embed the positional information into channel attention for precise localization of the low space ratio features. Finally, a small object loss feedback module is incorporated after the bidirectional feature pyramid network (BiFPN) for feature fusion, balancing the contribution of small object loss to the overall loss. Experimental comparisons on a battery cell casing dataset demonstrate that the proposed algorithm outperforms the EfficientDet-D1 object detection algorithm, with an average precision improvement of 4.23%. Specifically, for scratches with low space ratio features, the improvement is 13.21%; for wrinkles with low space ratio features, the improvement is 9.35%; and for holes with small object features, the improvement is 3.81%. Moreover, the detection time of 47.6 ms meets the requirements of practical production.

Keywords: low space ratio feature; small object feature; dual coordinate attention; small object loss feedback; defect detection of battery cell casings

Citation: Li, T.; Ren, J.; Yang, Q.; Chen, L.; Sun, X. Defect Detection Algorithm for Battery Cell Casings Based on Dual-Coordinate Attention and Small Object Loss Feedback. *Processes* **2024**, *12*, 601. https://doi.org/10.3390/pr12030601

Academic Editor: Yo-Ping Huang

Received: 29 January 2024
Revised: 2 March 2024
Accepted: 12 March 2024
Published: 18 March 2024

1. Introduction

High-quality metal casings play a crucial role in various fields, such as automobile manufacturing and aerospace. However, during the product manufacturing process, defects such as scratches, dents, dirt, and wrinkles not only affect the appearance of the product but also reduce its mechanical performance, leading to a decline in product quality and even causing serious safety and quality incidents [1]. For example, defects in metal casings like those of electric vehicle battery cells may lead to chemical leaks and battery overheating, resulting in fires or explosions [2]. Therefore, the quality inspection of metal surfaces has become critically important. Current inspection methods mainly rely on manual visual inspection, traditional machine vision, and deep learning methods [3]. Manual inspection is time-consuming, labor-intensive, and not very accurate, while traditional machine vision heavily depends on the quality of the data. On the other hand, deep learning-based object detection methods have gradually matured, achieving significant success in quality inspection and effectively improving the efficiency of metal surface quality inspection. Compared with traditional machine learning, deep learning has advantages in feature learning and extraction, complexity processing, model performance, and generalization capabilities. Deep learning can automatically learn features without manual feature engineering. When processing high-dimensional and complex data, it performs better and can better capture the complex patterns and correlations of data, thereby improving model

performance. It also has strong generalization capabilities and can better adapt to and predict new data in addition to training data.

Currently, in the field of metal surface quality inspection, deep learning methods have become mainstream [4–7]. Chen et al. [8] proposed an adaptive convolution and anchor network for metal surface quality inspection, introducing a multi-scale feature adaptive fusion that effectively extracts and integrates features from different levels and scales, considering both channel and spatial attention, thereby improving the performance of the detector. Zhao et al. [9] proposed a semi-supervised, transformer-based multi-scale feature pruning fusion method that can reasonably detect metal surface quality even in cases with limited samples of rust, scratches, and labeled samples. Zabin et al. [10] proposed a self-supervised representation learning model based on a contrastive learning framework, supported by an enhanced pipeline and a lightweight convolutional encoder, which effectively extracts meaningful representations from unlabeled image data and can be used for defect classification. For the battery casing, Zhang et al. [11] proposed an improved YOLOv5s model, which can accurately and quickly detect three defects on the bottom surface of lithium batteries. YOLOv5s adds a layer to the network output layer to improve the detection effect of small defects, employs the convolutional block attention module attention mechanism to extract important information in the feature map, and then uses a new positional loss function to improve the position prediction accuracy of the model. In order to realize the automatic detection of surface defects in lithium battery pole pieces, Xu et al. [12] proposed a multi-feature fusion and PSO-SVM surface defect detection method of lithium battery pole pieces, using the support vector machine (SVM) optimized by particle swarm optimization (PSO) to identify defects. For online real-time detection of surface defects in lithium-ion battery electrodes, Yu et al. [13] proposed an adaptive threshold segmentation algorithm based on gray histogram reconstruction.

Attention mechanisms can significantly improve the performance of detectors in the field of object detection. Hou et al. [14] proposed Coordinate Attention (CA), which embeds positional information into channel attention to enhance the representation of objects of interest. Feature fusion is an important component of metal surface quality detection that can fuse features from different layers, scales, or branches. The use of feature fusion modules can effectively improve the performance of detectors [15–19]. Tan et al. [20] proposed the BiFPN (bidirectional feature pyramid network) for feature fusion and reuse in their EfficientDet network, which has shown significant performance in multi-scale feature object detection. Mekhalfi et al. [21–24] made some improvements based on EfficientDet. Small object features in metal surface defects are also an important issue. In the field of small objects [25,26], a feedback-driven loss function is proposed to address the poor detection performance and low accuracy of small objects. By increasing the weight of small object loss, the network treats objects with different scales more equally.

Low space ratio features and small object features have always been difficult problems in the field of object detection [27]. The low space ratio features are long and narrow and account for a small area in the detection frame. They have strong directionality. It is necessary to pay more attention to such highly directional low space ratio features in the detection frame. Small object features are a very common and important problem. In the past, feature fusion was generally used to solve it, starting from FPN to BiFPN and other feature fusion methods. GAN [28] is also used to generate high-resolution images or high-resolution features to solve the problem, and some utilize the environmental information of the small object or the relationship with other easily detected objects to assist in the detection of small objects.

At present, deep learning methods have achieved certain success in metal surface quality detection, but there are problems with insufficient directional accuracy and missed detection of small object features when detecting low space ratio features. In response to this issue, this article proposes dual-coordinate attention and a small target loss feedback network (Efficient Attention Feedback Net, EAF-Net). The EAF-Net adopts the dual coordinate attention (DCA) mechanism to solve the problem of low directional accuracy,

that is, using average pooling and maximum pooling to form dual branches, embedding more positional information into channel attention, decomposing channel attention into four one-dimensional feature encoding processes, and achieving more accurate positional information preservation. On the basis of the DCA mechanism, the small object loss feedback (SOLF) block is further integrated, and the loss of statistical information is used as feedback to guide the next iteration update, enrich the small object features, and effectively solve the problem of missing small object features. In order to verify the effectiveness of the proposed network, 1908 surface images of electric vehicle battery cell shells were collected through a visual platform, and a battery cell shell dataset containing six defect categories was created. The accuracy and usability of the network were verified through experiments using the dual coordinate attention and small object loss feedback network proposed in this paper.

2. Dual Coordinate Attention and Small Object Loss Feedback Network

A dual coordinate attention and small object loss feedback network was designed to address the low space ratio and small object characteristics of metal surface defect data, as shown in Figure 1. The network mainly consists of EfficientDet-D1 as the main structure, including EfficientNet-B1 feature extraction, dual coordinate attention (DCA), bidirectional feature pyramid network (BiFPN), box/class prediction networks, and small object loss feedback module [26]. The EAF-Net first extracts features through the EfficientNet-B1 [29] backbone network, obtains effective feature layers, and enters them into dual coordinate attention to acquire attention weights and positional information. It then performs multi-scale feature fusion through the BiFPN and finally obtains loss values through classification and regression, which are used through the loss feedback module to calculate the proportion of small object losses. If the loss proportion is less than the threshold in the current iteration, the concatenated image is used as the input data in the next iteration, otherwise, the regular image continues to be used as the data input. This will dynamically train and adjust the proportion of small object loss, and train the network in a balanced manner.

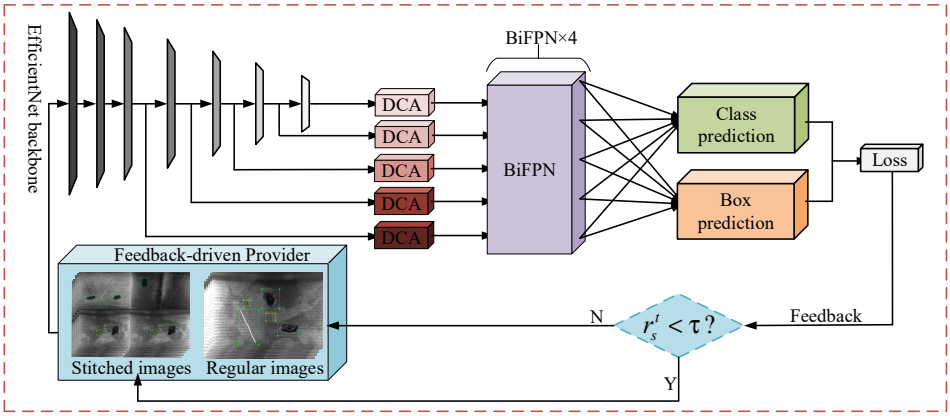


Figure 1. EAF-Net architecture.

2.1. Feature Extraction

The EAF-Net utilizes the EfficientNet-B1 backbone as the feature extraction component. EfficientNett-B1 employs a stacked structure of mobile inverted bottleneck convolution (MBConv) during feature extraction and adds Squeeze and Excitation (SE) [30] and Swish activation function to MBConv. After a head and seven stages of MBConv multiplexing and skip connections, three effective feature layers C3, C4, and C5 are obtained for feature extraction. Followed by downsampling and channel number adjustment, a total of five effective feature layers including C6 and C7 were obtained. The specific structure of EfficientNett-B1 is shown in Table 1.

Table 1. EfficientNet-B1 architecture.

Stage	Operator	Layers
1	Conv3 × 3	1
2	MBConv6, k3 × 3	2
3	MBConv6, k3 × 3	3
4	MBConv6, k5 × 5	3
5	MBConv6, k3 × 3	4
6	MBConv6, k5 × 5	4
7	MBConv6, k5 × 5	5
8	MBConv6, k3 × 3	2

2.2. Dual Coordinate Attention (DCA)

DCA attention not only considers information between channels but also emphasizes positional information. It employs a dual-branch approach to decompose channel attention into four independent directional awareness feature maps. Two maps are dedicated to the horizontal direction (X), and two maps are dedicated to the vertical direction (Y). Branch 1 decomposes through global average pooling, while Branch 2 decomposes through global max pooling. This results in two directional awareness feature maps, each for both the horizontal (X) and vertical (Y) directions. By utilizing the dual-branch approach, more positional information is embedded into channel attention. This method maximizes the preservation of precise positional information, with the specific structure illustrated in Figure 2.

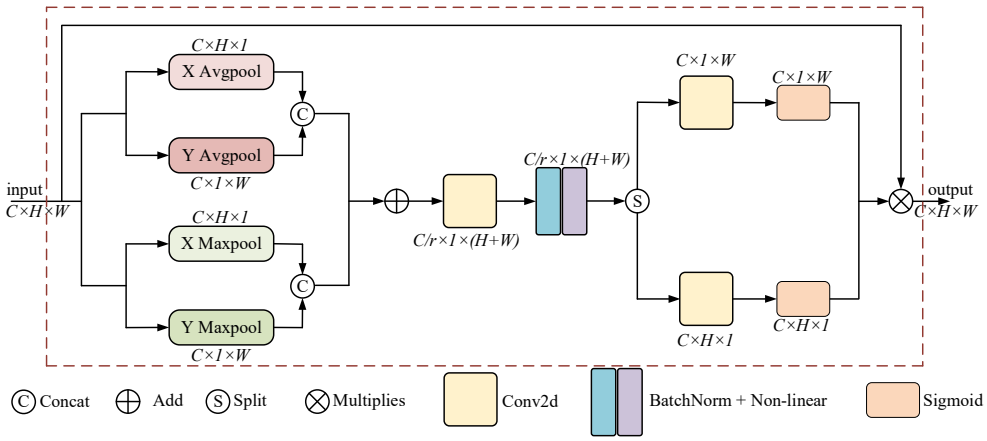


Figure 2. Dual Coordinate Attention architecture.

Branch 1 utilizes average pooling to decompose the input **x**, using pooling kernels $(H, 1)$ and $(1, W)$ global average pooling along the horizontal X and vertical Y directions. The output of the c-th channel with a height of h is:

$$z_c^{h_1}(h) = \frac{1}{W} \sum_{0 \le i \le W} x_c^1(h, i) \tag{1}$$

$$z_c^{w_1}(w) = \frac{1}{H} \sum_{0 \le j \le H} x_c^1(j, w) \tag{2}$$

Similarly, Branch 2 employs max pooling to decompose the input **x**, using pooling kernels $(H, 1)$ and $(1, W)$ global max pooling along the horizontal X and vertical Y directions. The output of the c-th channel with a width of w is:

$$z_c^{h_2}(h) = \frac{1}{W} \sum_{0 \le i \le W} x_c^2(h, i) \tag{3}$$

$$z_c^{w_2}(w) = \frac{1}{H} \sum_{0 \le j \le H} x_c^2(j, w) \tag{4}$$

We obtain two pairs of direction-aware feature maps, perform concatenation operations on them respectively, add the concatenated results, and then use a convolution and activation function to generate the feature map:

$$f = \delta(F_1(\left[Z^{h_1}, Z^{w_1}\right] + \left[Z^{h_2}, Z^{w_2}\right])) \tag{5}$$

where $f \in R^{C/r \times (H+W)}$ is the intermediate features containing horizontal and vertical spatial information, r is reduction factor, F_1 is 1×1 convolution, [ ] is Concat operation, and δ is the nonlinear activation function.

We then split f into f^h and f^w along the spatial dimension, and perform 1×1 convolutional dimensionality enhancement combined with sigmoid activation function for feature transformation to make its dimension consistent with the input **x**:

$$g^h = \sigma(F_h(f^h)) \tag{6}$$

$$g^w = \sigma(F_w(f^w)) \tag{7}$$

Finally, we multiply the input image with attention weights to obtain the output:

$$y_c(i, j) = x_c(i, j) \times g_c^h(i) \times g_c^w(j) \tag{8}$$

By assigning weights to different parts of the input, it determines what important information the model focuses on during processing. Analyzing these weights gives one an idea of how much attention the model pays to the input in its decision-making process.

The DCA attention module retains more accurate positional information from two branches, making the directionality stronger. It can make the network more precise in directional position information without bringing too much computational complexity, overcoming the problem of insufficient information retained by the average pooling used in the basic CA attention decomposition, and can alleviate to some extent the problem of low accuracy in low space ratio feature detection, as shown in Figure 3.

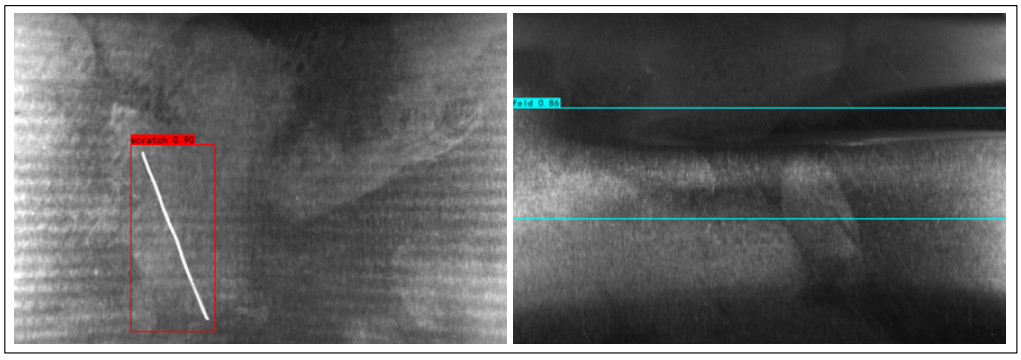


Figure 3. Detection results of low space ratio features.

2.3. Feature Fusion BiFPN

For multi-scale feature fusion problems, FPN [15] is a typical feature fusion network, which introduces a top-down path to fuse multi-scale features at levels 3–7. NAS-FPN [16] is an irregular feature network obtained through neural structure search, while PANet [17] adds an additional bottom-up channel on the basis of FPN. In previous feature fusion, since different input features have different resolutions and their contributions to the output features are also different, learnable weights can be introduced to learn the importance of different features. On the basis of the above feature fusion network, BiFPN [20] adds skip connection output nodes to fuse more features, and uses bidirectional paths from

top to bottom and bottom to top for reuse. During feature fusion, fast normalization fusion features are used to reduce computational complexity. Firstly, each bidirectional path is regarded as a feature network layer, and the same layer is repeated multiple times to achieve more advanced feature fusion. Secondly, during the feature fusion process, BiFPN needs to resize the input features to the same level due to different resolutions. The resolution has different contributions to the output of the final feature network. BiFPN adopts the weighted fusion method of fast normalized fusion:

$$O = \sum_i \frac{\omega_i}{\varepsilon + \sum_j \omega_j} I_i \tag{9}$$

By applying Relu to ω_i make $\omega_i \geq 0$, and at the same time $\varepsilon = 0.0001$ to ensure numerical stability, this weighted fusion method of fast normalized feature fusion greatly improves the training speed by avoiding the Softmax operation. BiFPN is obtained using a weighted fusion method of fast normalized feature fusion during bidirectional cross-scale connection. The structure of BiFPN is shown in Figure 4.

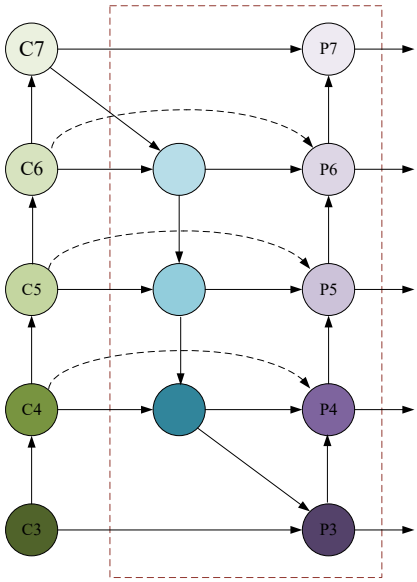


Figure 4. BiFPN Structure.

2.4. Small Object Loss Feedback Block

In the training iteration, if the contribution of small object losses to total losses is too low, it results in a decrease in optimization performance due to imbalance. The EAF-Net adds a small object loss feedback (SOLF) block after the prediction module, which dynamically adjusts the scale of the input image and increases the proportion of small object losses, with its specific structure detailed in Figure 5.

Small object loss feedback is a feedback-driven data provider that utilizes loss statistics as feedback. It selects different data as inputs, dynamically uses concatenated or regular images to balance the multi-scale features of the data, and introduces more small object features to increase the supervision signal of small objects. In the small object loss feedback module, by reducing the image size and then using k images for stitching, the same size as the regular image is obtained. When k is 1, the image is a regular image. When k is specified as 4, it is the stitched images shown in Figure 5. If the loss ratio of small objects in this iteration is lower than the threshold, the next iteration will use the concatenated image as the data input. The smaller objects in the concatenated image will be more abundant, which can alleviate the problem of insufficient supervision signals for small objects in the dataset. Because the size of the image remains unchanged after stitching, no additional calculations will be introduced. The area of the small object is defined as $h \times w$, and when

the area is less than 1024 (32 × 32), the regression loss of the object is defined as the small object loss:

$$a_0 \approx h_0 \times w_0 \tag{10}$$

After obtaining the loss value, the proportion of small object losses can be calculated. Based on the comparison between the proportion and the threshold, the appropriate input can be selected in the next iteration:

$$L_s^{reg} = L_{a_0 < A_s}^t \tag{11}$$

$$r_s^t = \frac{L_s^t}{L^t} \tag{12}$$

where L_s^t represents the small object loss and L^t represents the total loss.

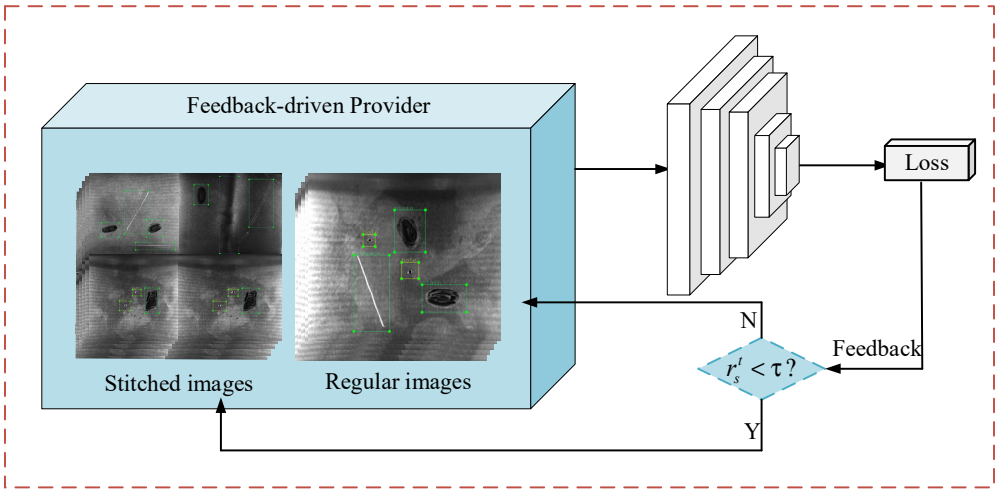


Figure 5. Small object loss feedback architecture.

3. Data Collection and Dataset Generation

3.1. Data Collection

In order to better validate the effectiveness of the EAF-Net in industrial environments, an experiment was conducted to detect defects in the outer shell of electric vehicle battery cells. A total of 1908 images were captured for six common defect categories, and the data collection used the BTW vision platform, including BS-LN4217 strip light source, BS-M-CV65W-24V-4 light source controller, BL-1218-6MP lens and BC-M2592-48UM camera, sourced from the BTW company in Shanghai, China, with some data samples shown in Figure 6.

The data set contains six different defect types, namely scratches, wrinkles, holes, dirt, foreign objects, and pits. These defect types represent common problems that may arise during actual production or manufacturing processes. Below is a detailed description of these defect types:

Scratches: Scratches refer to linear or curved marks on the surface caused by scratches by sharp objects or other hard objects. Scratches may result in reduced surface quality, affecting the appearance and performance of the product.

Wrinkles: Wrinkles are creases that appear on the surface of a material or product. This may be caused by the material being squeezed, stretched, or folded during the production process, affecting the flatness and aesthetics of the product.

Holes: Holes are round or irregularly shaped cavities on the surface of a material or product. This may be due to internal defects in the material, errors during processing, or other reasons, that can affect the structural integrity and functionality of the product.

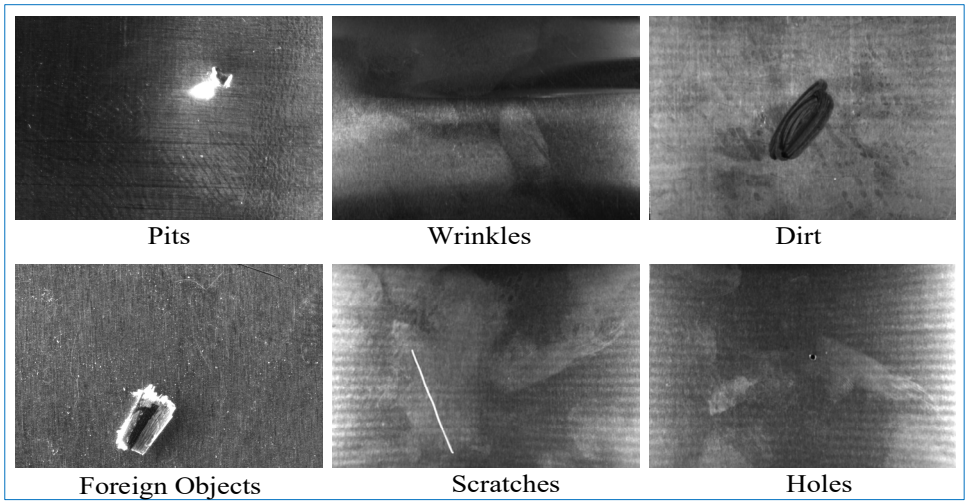


Figure 6. Partial data samples.

Dirt: Dirt refers to stains, dirt, or other impurities attached to the surface of materials or products, causing the surface to lose its original cleanliness and smoothness. Dirt may affect the appearance and hygiene of the product.

Foreign matter: Foreign matter refers to impurities or substances that are not originally part of the product and accidentally enter the product or adhere to the surface of the product. Foreign objects may affect the functionality, safety, and hygiene of the product.

Pits: Pits are dents or depressions in a surface that may be caused by impacts from external objects, pressure, or other factors. Pits will reduce the appearance quality and surface smoothness of the product.

We found that, to a certain extent, the higher the image resolution, the better the detection accuracy because high-resolution images can provide more detailed information and help the model detect targets more accurately. However, high-resolution images also result in slower detection because more pixels need to be processed. This requires a trade-off between performance and computational efficiency.

3.2. Dataset Generation

Six types of common defect features were analyzed and labeled using image labeling tools (such as LabelImg) to create a dataset of electric vehicle battery cell shells. The specific numbers of samples are shown in Table 2.

Table 2. Dataset categories.

Defect Category	Scratches	Wrinkles	Holes	Dirt	Foreign Objects	Pits	Total
Number	1233	338	621	1083	404	247	1908

Our data set contains different defect types, and there is a significant difference in the number of samples. The largest category has 1233 samples, while the smallest category has only 247 samples. The imbalanced sample sizes can potentially introduce bias in the machine learning models. This will be investigated in the training results to check if we can still achieve the model generalization ability when the training data is highly imbalanced.

3.3. Dataset Partitioning

In the experiment, the test set was divided according to the ratio of (training set + validation set): test set = 9:1, and the training set and validation set were divided according to the ratio of training set: validation set = 9:1. The battery shell dataset was created in the format of the PASCAL VOC dataset.

This proportional division method can provide enough data for training the model as much as possible, and also have enough data for verifying the performance and generalization ability of the model. The purpose of the validation set is to adjust the hyperparameters of the model and monitor the performance of the model during the training process to avoid overfitting. The test set is used to ultimately evaluate the performance of the model. The performance of the model on unseen data is measured against the test set.

4. EAF-Net Experimental Analysis

The computer hardware used in the experiment is Intel (R) Xeon (R) CPU E5-2680 v3 @ 2.50 GHz, with 64 GB RAM and 11 GB GTX1080Ti GPU. The software environment is Python 3.9.16 and PyTorch 2.0.1. The training parameters were set to 120 epochs, with a batch size of 4, and maximum and minimum learning rates of 3×10^{-4} and 3×10^{-6}, respectively. The Adam optimizer and Cos learning rate reduction method were selected. The specific training parameters are shown in Table 3.

Table 3. Training parameters.

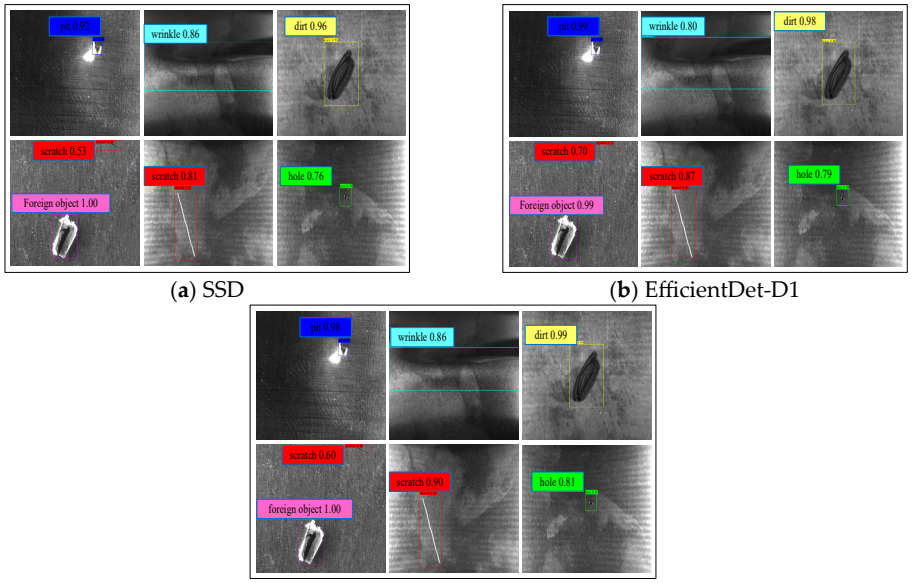
Parameter Name	Parameter Value	Note
Epoch	120	Epochs of training
Batch size	4	Number of samples used in one training iteration
Init_lr	3×10^{-4}	Maximum learning rate
Lr_decay_type	cos	Learning rate reduction method
Optimizer_type	Adam	Optimizer type
Momentum	0.9	Momentum
Mini_lr	3×10^{-6}	Minimum learning rate

The number of epochs determines the number of times the entire training set is traversed. It is designed to ensure that the model has enough time to learn the data features while avoiding overfitting. More epochs can provide more training opportunities, but also increase training time. Experiments show that the loss has converged when Epoch is 120. The Batch Size is 4 due to memory constraints. The learning rate determines the magnitude of weight updates. Choosing a dynamic learning rate range helps to quickly converge in the early stages of training, and improves the stability and performance of the model by reducing the learning rate in the later stages of training. This range set helps to find balance. Adam is a widely used optimization algorithm that can both speed up the training process and automatically adjust the learning rate to suit the needs of most problems. Adam was chosen because it has shown good convergence speed and stability in practice. The Cos learning rate decay strategy is a method of periodically adjusting the learning rate. This method helps the model find better local optimal solutions in multiple cycles and avoid falling into the initial local optimal, thereby improving model performance.

4.1. EAF-Net Validity Validation

To verify the effectiveness of the EAF-Net, SSD [31], EfficientDet-D1, and EAF Net were compared on the cell shell dataset. The average accuracy of our method EAF-Net in the battery shell dataset is 98.49%, which is better than SSD and EfficientDet-D1's 92.35% and 94.26%, respectively. The score of each type of detection result is generally higher than the other two models, especially in the categories of scratches, wrinkles, and holes with low space ratio and small object features. The detection results and scores are shown in Figure 7, and the score is enlarged for easy viewing, with the specific data shown in Table 4. According to data comparison, EAF-Net demonstrates an average accuracy increase of 6.14% and 4.23% compared to SSD and EfficientDet-D1, respectively. It leads in varying degrees across six categories, with the most significant improvement seen in Scratches at 14.18% and 13.21% over the two counterparts. It is worth noting that the unbalanced sample sizes have not caused any noticeable bias in the model detection performance. In terms of detection time, EAF-Net only lags behind SSD and EfficientDet-D1 by 16.4 ms and

5.3 ms, respectively. Therefore, our approach, EAF-Net, achieves the highest accuracy in both six categories and average accuracies without significantly increasing detection time.



(**a**) SSD (**b**) EfficientDet-D1

(**c**) EAF-Net

Figure 7. Detection results examples using different detection algorithms.

Table 4. Detection results.

Algorithms	Average Accuracy/%						Mean Average Accuracy/%	Detection Time/ms
	Scratches	Wrinkles	Holes	Dirt	Foreign Objects	Pits		
SSD	83.04	85.34	94.16	97.54	96.87	98.91	92.35	31.2
EfficientDet-D1	84.01	86.42	96.19	98.12	98.92	99.82	94.26	42.3
EAF-Net	97.22	95.77	100.00	98.48	100.00	100.00	98.49	47.6

After 120 iterations of training, the loss curve converged, and the EAF-Net loss curve is shown in Figure 8.

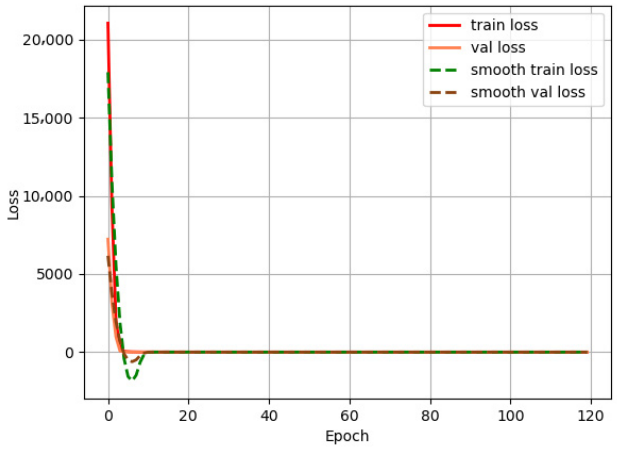


Figure 8. EAF-Net loss curve.

4.2. Validation of Dual Attention Effectiveness

To verify the effectiveness of dual attention, four rounds of experiments were conducted on the battery shell dataset using three types of attention for analysis. The first

group is EfficientDet-D1, the second group is EfficientDet-D1 with coordinated attention, the third group is EfficientDet-D1 with Efficient Channel Attention (ECA) [32], and the fourth group is EfficientDet-D1 with DCA attention. The detailed experimental results are shown in Figure 9. It can be seen from the four sets of experiments that the mean accuracy performance (MAP) of EfficientDet-D1 with the DCA attention group was 3.11%, 1.09%, and 1.04% higher than the other three groups, respectively. Among the six categories, five had the highest accuracy, with scratches and wrinkles being 8.47%, 2.40%, 1.41%, and 8.80%, 3.91%, and 4.84% higher than the other three groups, respectively. This proves that DCA attention can effectively solve the problem of low space ratio feature detection.

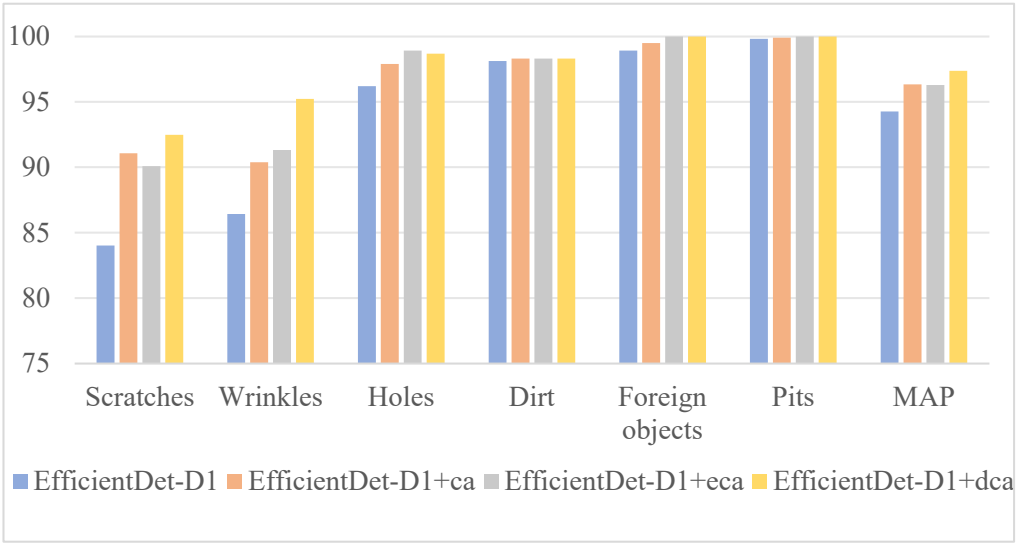


Figure 9. Model detection results.

In order to verify the performance difference between the dual-coordinate attention mechanism and the traditional attention mechanism in defect detection of different categories, we also compared their precision and recall in different categories and analyzed the performance differences between the two mechanisms in different categories in detail. Compared with the other three groups, the EfficientDet-D1 + DCA group has the highest precision in four of the six categories, and has a greater advantage in the recall, with the highest recall in five of the six categories, as shown in Figures 10 and 11.

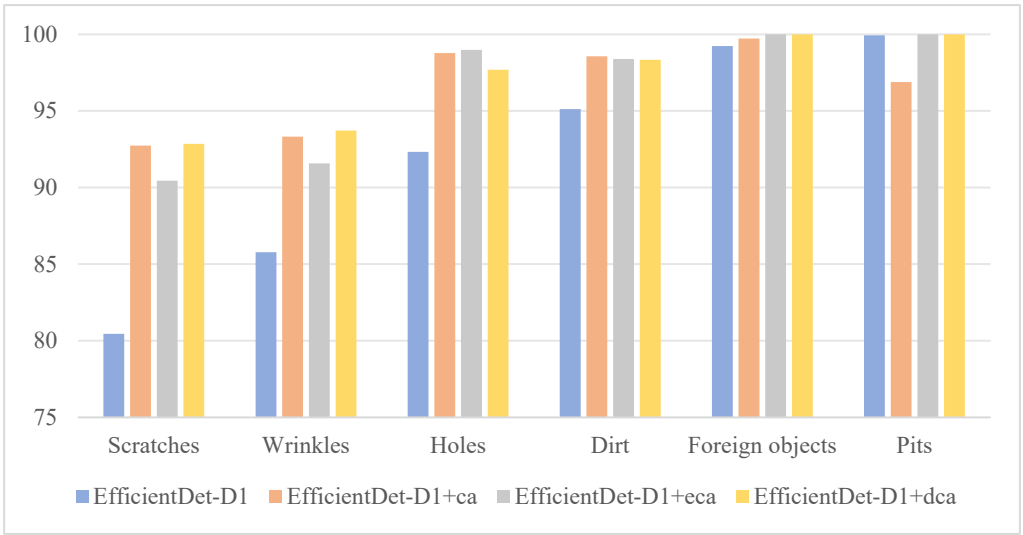


Figure 10. Recall results.

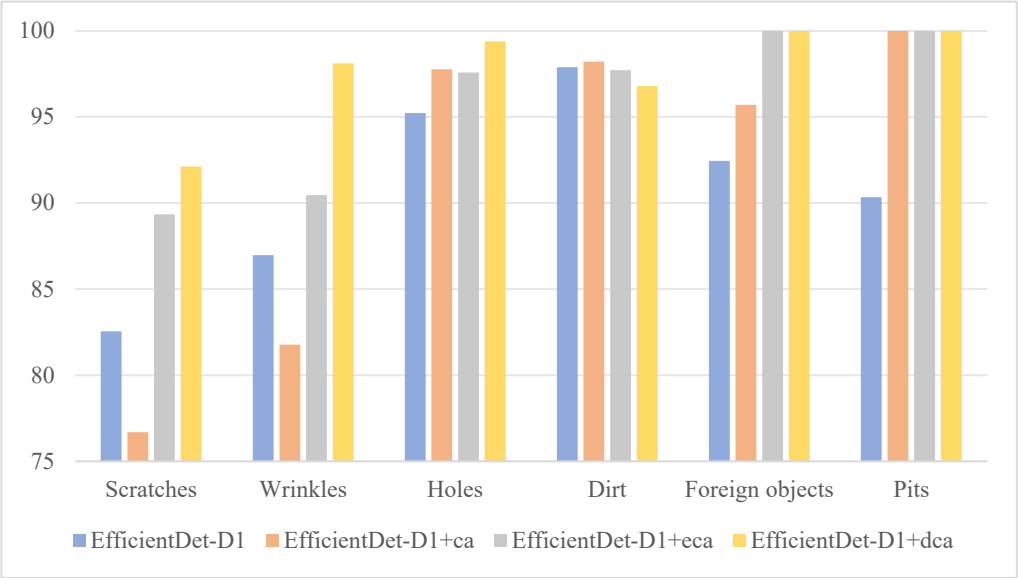


Figure 11. Precision results.

4.3. Ablation Experiment

To verify the EAF-Net that integrates DCA and SOLF, this study conducted four rounds of ablation experiments on the cell shell dataset, namely EfficientDet-D1, EfficientDet-D1+DCA, EfficientDet-D1+SOLF, and EfficientDet-D1+DCA+SOLF. As shown in Table 5, the test results on the cell shell dataset were significantly improved. Compared with EfficientDet-D1, with only an additional cost of 5.3 ms, the average accuracy of the EAF-Net has increased by 4.23%, among which scratches with low space ratio and small object characteristics increased by 13.21%, wrinkles with space ratio characteristics increased by 9.35%, and holes with small object characteristics increased by 3.81%. For the DCA module, compared with EfficientDet-D1, the accuracy of EfficientDet-D1+DCA increased by 8.47% and 8.80% in the categories of scratches and wrinkles respectively, and the average accuracy increased by 3.11%, with the detection time only increased by 4.5 ms. Compared with EfficientDet-D1, EfficientDet-D1+SOLF has improved accuracy by 6.55% and 2.33% in the categories of scratches and holes respectively, the average accuracy has increased by 1.98%, and the detection time has only increased by 1.1 ms. The specific data are shown in Table 5.

Table 5. Model detection results.

EfficientDet-D1	DCA	SOLF	Average Accuracy/%						Mean Average Accuracy/%	Detection Time/ms
			Scratches	Wrinkles	Holes	Dirt	Foreign Objects	Pits		
√			84.01	86.42	96.19	98.12	98.92	99.82	94.26	42.3
√	√		92.48	95.22	98.68	98.31	100.00	100.00	97.37	46.8
√		√	90.56	86.64	98.52	98.42	98.96	99.91	96.24	43.4
√	√	√	97.22	95.77	100.00	98.48	100.00	100.00	98.49	47.6

In the table, "√" indicates that the algorithm utilizes the current module.

The results indicate that the EAF-Net can meet the requirements of time and accuracy, effectively detecting various defects in the battery cell casing. Again, it is worth noting that the unbalanced sample sizes have not caused any noticeable bias in the model detection performance.

4.4. Verification of Model Generalization Ability with Limited Training Data

In order to verify the generalization ability of the model when the training data is limited, we prepared three sets of data sets with different data sizes by reducing the

amount of data. Set 1 has 477 images, Set 2 has 954 images, and Set 3 has 1908 images. The experiment compared their MAP and detection time on EAF-Net. As shown in Figure 12, on Set 1 with too little data, the MAP will drop significantly, which is 3.91% and 4.14% lower than Set 2 and Set 3, respectively, while Set 2 is only 0.23% lower than Set 1. In terms of detection time, as shown in Figure 13, the detection times of the three groups of experiments are similar, and there are no significant differences in the data set size.

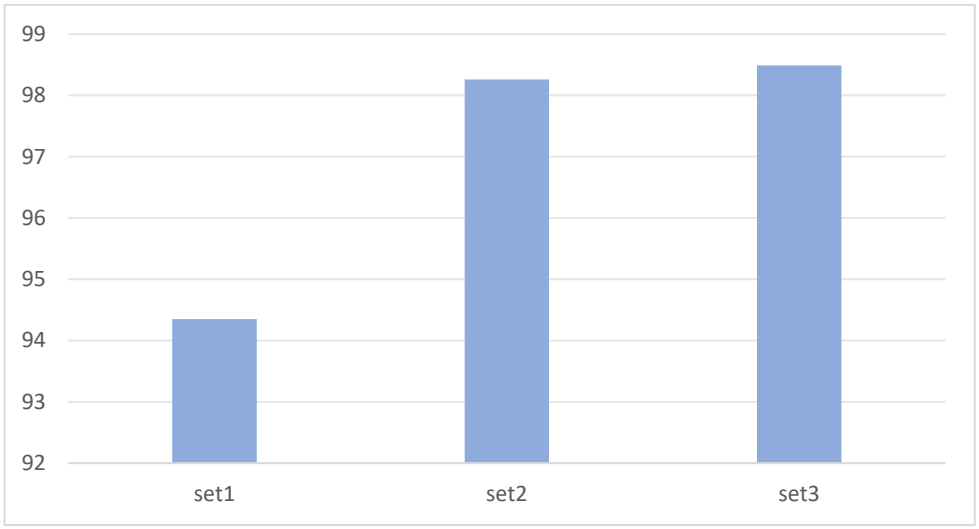


Figure 12. Model generalization ability with limited training data.

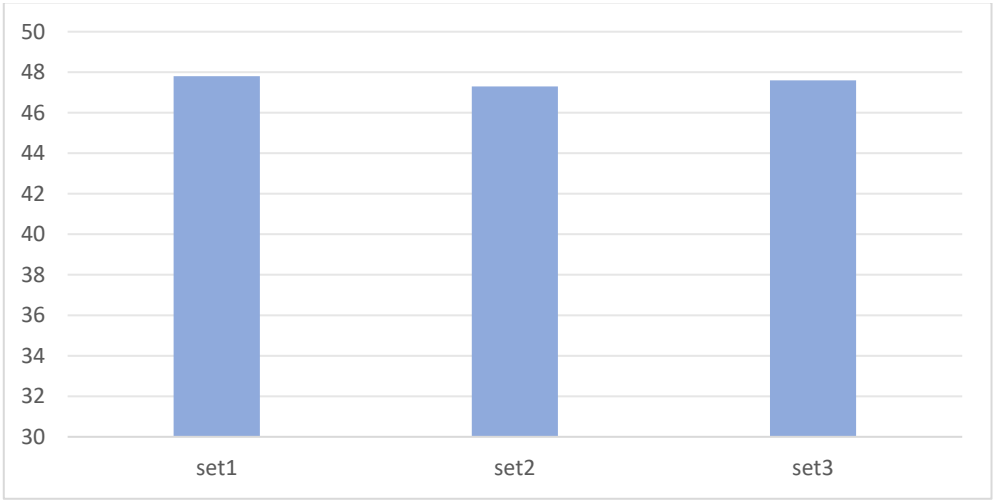


Figure 13. Detection time with limited training data.

5. Discussions and Conclusions

A dual coordinate attention and small object loss feedback network, EAF-Net, has been proposed to address the low space ratio and small object characteristics of metal surface defects. After experimental verification on the battery shell dataset, the average accuracy can reach 98.49%, and the detection time is 47.6 ms, meeting the actual production requirements. Its detection accuracy is 4.23% higher than that of the EfficientDet-D1 network. The dual coordinate attention (DCA) proposed in the EAF-Net preserves precise positional information from the dual branches, enhances directionality, and improves the detection accuracy of scratches and wrinkles with low space ratio features by 13.21% and 9.35%, respectively; The added small object loss feedback module enhances the supervision signal of small objects, improving the detection accuracy of scratches and holes with small

object features by 13.21% and 3.81%. Therefore, the EAF-Net provides a feasible and effective method for metal surface quality detection.

This paper employs dual branches to decompose the channel attention into four independent direction-aware feature maps, embed more positional information into the channel attention, and retain accurate positional information, with good results achieved in directionality. It solves the common and important problem of small object features without generating high-resolution images or high-resolution features. At the same time, the SOLF adopted in this article made use of loss statistical information as feedback to introduce more small object features and increase the supervision signals of small objects to solve the problem of missed detection of small object features.

We recognize that the transferability of the algorithm to different types of materials or different defect types is critical to its scalability in the broader quality inspection domain. This algorithm has strong adaptability to different defect types and materials. Our algorithm is suitable for defect detection of common materials and is not limited to certain special materials. We have verified the algorithm's ability to identify common defect types in experiments. However, because it mainly solves the problems of the low space ratio features and small object characteristics as well as the limited data set, the experimental verification mainly focuses on six specific types of defects. Our algorithm has the potential for wider applicability and applications in different fields and conditions. At the same time, new defect types can be updated and expanded, and incremental learning can be performed. Our algorithm currently only needs to load the previous model as a pre-training model for retraining to adapt to new defect types and does not need to be retrained from scratch, so that the model is more flexible and has certain robustness. This updated strategy can not only effectively deal with the introduction of new unseen defect types, but also alleviate the problem of model performance drift over time to a certain extent and conduct defect detection system life cycle management, including updating, retraining, and monitoring of performance degradation because it allows the model to adapt to changes in the manufacturing process without retraining. Likewise, we recognize that error analysis of algorithms is critical in the broader field of quality inspection. Our error analysis focuses on two common failure cases: wrinkles and scratches, both categories characterized by low duty cycle. Although our algorithm has achieved certain improvements, there is still room for improvement. Especially in extremely low-contrast scenes, scratches are often light and easily missed, which shows that our algorithm still faces challenges in such complex scenes. In terms of the cost-effectiveness of algorithms, integrating the proposed algorithms into existing systems may require certain integration costs, including aspects such as data acquisition, model training, debugging, testing, etc., as well as some one-time investments in equipment. However, compared with the long-term costs of manual visual inspection, it is an economically viable choice in the long run.

Regarding energy consumption, during model training and inference, energy consumption mainly comes from the computers used. However, compared with traditional manual inspection methods, our algorithms can save a significant amount of manpower and resources, improve production efficiency, and ultimately save energy. Additionally, in our research, we are committed to emphasizing the positive impact of automated detection on reducing workload, freeing up labor, and ensuring personnel safety in production.

In this study, we have explored in depth the application of dual-coordinate attention and small object loss feedback in battery casing defect detection and proposed a defect detection algorithm that meets actual production applications. Although some progress has been made, we also realize that there are some limitations to our work. Future work can be carried out in the following two aspects: On the one hand, introducing multimodal data allows us to address the issue of poor model generalization when dealing with low-quality, single-modal data. At the same time, the fusion of multi-modal data can help improve the model generalization ability in different environments. On the other hand, we can also explore other methods that can improve the performance of small object detection. At present time, small object detection is still a major problem in the field of object detection.

Finally, we encourage further research on potential applications of related algorithms in other computer vision fields to better understand their scalability and generality.

Author Contributions: Conceptualization, T.L.; Methodology, J.R. and Q.Y.; Validation, T.L. and Q.Y.; Resources, T.L. and L.C.; Writing—original draft, J.R. and X.S.; Writing—review & editing, T.L., Q.Y. and L.C.; Supervision, T.L.; Project administration, L.C. and X.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors are grateful for the facilities and other support part by Shanghai Betterway for Automation Company.

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

References

1. Bhatt, P.M.; Malhan, R.K.; Rajendran, P.; Shah, B.C.; Thakar, S.; Yoon, Y.J.; Gupta, S.K. Image-based surface defect detection using deep learning: A review. *J. Comput. Inf. Sci. Eng.* **2021**, *21*, 040801. [CrossRef]
2. Arizona Public Service. *McMicken Battery Energy Storage System Event Technical Analysis and Recommendations*; Arizona Public Service: Phoenix, AZ, USA, 2020.
3. Tang, B.; Chen, L.; Sun, W.; Lin, Z.K. Review of surface defect detection of steel products based on machine vision. *IET Image Process.* **2023**, *17*, 303–322. [CrossRef]
4. Cumbajin, E.; Rodrigues, N.; Costa, P.; Miragaia, R.; Frazão, L.; Costa, N.; Fernández-Caballero, A.; Carneiro, J.; Buruberri, L.H.; Pereira, A. A Systematic Review on Deep Learning with CNNs Applied to Surface Defect Detection. *J. Imaging* **2023**, *9*, 193. [CrossRef] [PubMed]
5. Konovalenko, I.; Maruschak, P.; Brezinová, J.; Prentkovskis, O.; Brezina, J. Research of U-Net-based CNN architectures for metal surface defect detection. *Machines* **2022**, *10*, 327. [CrossRef]
6. Wang, C.; Xie, H. MeDERT: A Metal Surface Defect Detection Model. *IEEE Access* **2023**, *11*, 35469–35478. [CrossRef]
7. Li, Z.; Zhang, Y.; Fu, X.; Wang, C. Metal surface defect detection based on improved YOLOv5. In Proceedings of the 2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS), Chengdu, China, 7–9 July 2023; IEEE: Piscataway, NJ, USA, 2023; pp. 1147–1150.
8. Chen, F.; Deng, M.; Gao, H.; Yang, X.; Zhang, D. ACA-Net: An Adaptive Convolution and Anchor Network for Metallic Surface Defect Detection. *Appl. Sci.* **2022**, *12*, 8070. [CrossRef]
9. Zhao, L.; Zheng, Y.; Peng, T.; Zheng, E. Metal Surface Defect Detection Based on a Transformer with Multi-Scale Mask Feature Fusion. *Sensors* **2023**, *23*, 9381. [CrossRef] [PubMed]
10. Zabin, M.; Kabir AN, B.; Kabir, M.K.; Choi, H.J.; Uddin, J. Contrastive self-supervised representation learning framework for metal surface defect detection. *J. Big Data* **2023**, *10*, 145.
11. Zhang, Y.; Lu, X.; Li, W.; Yan, K.; Mo, Z.; Lan, Y.; Wang, L. Detection of Power Poles in Orchards Based on Improved Yolov5s Model. *Agronomy* **2023**, *13*, 1705. [CrossRef]
12. Xu, C.; Li, L.; Li, J.; Wen, C. Surface defects detection and identification of lithium battery pole piece based on multi-feature fusion and PSO-SVM. *IEEE Access* **2021**, *9*, 85232–85239. [CrossRef]
13. Liu, Y.; Chen, Y.; Xu, J. An automatic defects detection scheme for lithium-ion battery electrode surface. In Proceedings of the 2020 International Symposium on Autonomous Systems (ISAS), Guangzhou, China, 6–8 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 94–99.
14. 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, 20–25 June 2021; pp. 13713–13722.
15. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
16. Ghaisi, G.; Lin, T.-Y.; Pang, R. Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7029–7038.
17. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
18. Liu, S.; Huang, D. Receptive field block net for accurate and fast object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 385–400.
19. Zhang, Q.; Xiao, T.; Huang, N.; Zhang, D.; Han, J. Revisiting feature fusion for RGB-T salient object detection. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *31*, 1804–1818. [CrossRef]
20. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 10781–10790.

21. Mekhalfi, M.L.; Nicolò, C.; Bazi, Y.; Al Rahhal, M.M.; Alsharif, N.A.; Al Maghayreh, E. Contrasting YOLOv5, transformer, and EfficientDet detectors for crop circle detection in desert. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
22. Song, S.; Jing, J.; Huang, Y.; Shi, M. EfficientDet for fabric defect detection based on edge computing. *J. Eng. Fibers Fabr.* **2021**, *16*, 15589250211008346. [CrossRef]
23. Wang, Y.; Wang, T.; Zhou, X.; Cai, W.; Liu, R.; Huang, M.; Jing, T.; Lin, M.; He, H.; Wang, W. TransEffiDet: Aircraft detection and classification in aerial images based on EfficientDet and transformer. *Comput. Intell. Neurosci.* **2022**, *2022*, 2262549. [CrossRef] [PubMed]
24. Li, R.; Wu, J.; Cao, L. Ship target detection of unmanned surface vehicle base on efficientdet. *Syst. Sci. Control. Eng.* **2022**, *10*, 264–271. [CrossRef]
25. Liu, G.; Han, J.; Rong, W. Feedback-driven loss function for small object detection. *Image Vis. Comput.* **2021**, *111*, 104197. [CrossRef]
26. Chen, Y.; Zhang, P.; Li, Z.; Li, Y.; Zhang, X.; Qi, L.; Sun, J.; Jia, J. Dynamic scale training for object detection. *arXiv* **2020**, arXiv:2004.12432.
27. Cheng, G.; Yuan, X.; Yao, X.; Yan, K.; Zeng, Q.; Xie, X.; Han, J. Towards large-scale small object detection: Survey and benchmarks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 13467–13488.
28. Bai, Y.; Zhang, Y.; Ding, M.; Ghanem, B. Sod-mtgan: Small object detection via multi-task generative adversarial network. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 206–221.
29. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. *Int. Conf. Mach. Learn.* **2019**, *97*, 6105–6114.
30. 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.
31. 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 Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Proceedings, Part I 14. Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
32. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11534–11542.