

Article

An Effective Surface Defect Classification Method Based on RepVGG with CBAM Attention Mechanism (RepVGG-CBAM) for Aluminum Profiles

Zhiyang Li ¹, Bin Li ¹, Hongjun Ni ^{1,*}, Fuji Ren ², Shuaishuai Lv ^{1,*} and Xin Kang ³¹ School of Mechanical Engineering, Nantong University, Nantong 226019, China² School of Computer Science and Engineering, University of Electronic Science and Technology, Chengdu 611731, China³ Graduate School of Advanced Technology and Science, Tokushima University, Tokushima 770-8506, Japan

* Correspondence: ni.hj@ntu.edu.cn (H.N.); lvshuaishuai@ntu.edu.cn (S.L.)

Abstract: The automatic classification of aluminum profile surface defects is of great significance in improving the surface quality of aluminum profiles in practical production. This classification is influenced by the small and unbalanced number of samples and lack of uniformity in the size and spatial distribution of aluminum profile surface defects. It is difficult to achieve high classification accuracy by directly using the current advanced classification algorithms. In this paper, digital image processing methods such as rotation, flipping, contrast, and luminance transformation were used to augment the number of samples and imitate the complex imaging environment in actual practice. A RepVGG with CBAM attention mechanism (RepVGG-CBAM) model was proposed and applied to classify ten types of aluminum profile surface defects. The classification accuracy reached 99.41%, in particular, the proposed method can perfectly classify six types of defects: concave line (cl), exposed bottom (eb), exposed corner bottom (ecb), mixed color (mc), non-conductivity (nc) and orange peel (op), with 100% precision, recall, and F1. Compared with the existing advanced classification algorithms VGG16, VGG19, ResNet34, ResNet50, ShuffleNet_v2, and basic RepVGG, our model is the best in terms of accuracy, macro precision, macro recall and macro F1, and the accuracy was improved by 4.85% over basic RepVGG. Finally, an ablation experiment proved that the classification ability was strongest when the CBAM attention mechanism was added following Stage 1 to Stage 4 of RepVGG. Overall, the method we proposed in this paper has a significant reference value for classifying aluminum profile surface defects.



Citation: Li, Z.; Li, B.; Ni, H.; Ren, F.; Lv, S.; Kang, X. An Effective Surface Defect Classification Method Based on RepVGG with CBAM Attention Mechanism (RepVGG-CBAM) for Aluminum Profiles. *Metals* **2022**, *12*, 1809. <https://doi.org/10.3390/met12111809>

Academic Editors: Alain Pasturel and Alberto Moreira Jorge Junior

Received: 23 August 2022

Accepted: 21 October 2022

Published: 25 October 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: aluminum profile; surface defect classification; RepVGG; CBAM; attention mechanism

1. Introduction

Aluminum profiles are widely applied in construction, automobiles, and high-end equipment manufacturing due to their advantages of low density, corrosion resistance, strong plasticity, and recyclability [1–3]. There are many factors that affect the surface quality of aluminum alloy materials, such as microscale deformation that affects its surface roughness [4], and the crystal structure and grain orientation of the material also have an impact on its roughness [5]. Additionally, due to the complexity of the production process and many other factors, there will inevitably be ten types of defects on the surface of aluminum profiles: concave line (cl), dirty spot (ds), exposed bottom (eb), exposed corner bottom (ecb), graze (gra), mixed color (mc), non-conductivity (nc), orange peel (op), paint bubble (pb), and spray paint flow (spf). These defects not only affect the aesthetics of the product but also reduce its life and durability. At present, most aluminum profile production enterprises still rely on the traditional manual visual inspection method to classify defects, due to the fact that the surface of the aluminum profile itself contains patterns and defects that are not well-differentiated. Various types of defects have different

shapes and sizes, and their distribution is irregular. The manual visual method has a high labor intensity and low detection efficiency, and it is difficult to ensure the stability and accuracy of classification [6].

In recent years, there has not been much research on aluminum profile surface defect classification based on machine vision technology, but it has been widely used in the field of industrial production quality inspection for steel strips, printed circuit boards (PCBs), and so on [7–9]. For example, Zaghdoudi R. [10] proposed a defect classification method for steel strips based on a binary Gabor pattern (BGP) algorithm and a support vector machine (SVM). The local texture features of the strip steel defect images were first extracted and then an SVM was used to classify the strip steel defects into six categories. Hu et al. [11] proposed extracting geometric features, grayscale features, and shape features from steel strip defect images as well as their binary images, and then used an SVM to classify the strip surface defects. Chondronasios A. et al. [12] proposed a feature statistical method based on gradient-only co-occurrence matrices (GOCMs) to classify two types of defects (blisters and scratches) in extruded aluminum profiles. The essence all of these is the same, which can be regarded as the problem of defect classification on the workpiece surface. All the above methods are based on manually designed feature operators and classifiers to classify surface defects, which have achieved some results in solving the problem of categorizing workpieces surface defects. However, these methods often fail to obtain high classification accuracy. In addition, these manually designed feature operators are poorly adaptable and effective only for specific defects and imaging environments [13], which is not conducive to solving the task of classifying aluminum profile surface defects.

In recent years, with the continuous development of artificial intelligence and improvements in computing power, classification methods based on deep learning have attracted widespread attention, and many scholars have conducted related experiments and research. For example, Duan et al. [14] proposed a two-stream convolutional neural network based on gradient images to effectively classify aluminum profile defects. Abualighah et al. [15] designed a deep neural network classifier and combined it with the DesneNet201 pre-training model to achieve a classification accuracy of 98.43% for seven types of strip steel defects. Zhang et al. [16] proposed a novel dual-stream neural network that was used to generate a large number of defect images to pre-train a classification network and classify the surface defects of steel strips via transfer learning. Liu et al. [17] proposed a dual-convolutional neural network by integrating VGG16 and AlexNet, and its aluminum profile classification accuracy reached 95.1%. The defect classification methods based on deep learning are highly versatile and transferable and can be adapted to various classification tasks and achieve satisfactory results. However, the current research still has shortcomings. On one hand, such methods often require a large amount of data to train the model but the number of images in most workpiece surface defect datasets is very small [18–20]. On the other hand, neural networks integrating attention mechanisms are able to attach more importance to valuable units in the object, which is beneficial to improving classification accuracy [21]. Still, there is a lack of scientific research in the classification of aluminum profile surface defects.

However, the number of surface aluminum profile defect images is too small to support a training model. Augmenting the dataset becomes an urgent problem to be solved. Additionally, there is a lack of uniformity in the size and spatial distribution of aluminum profile surface defects, which makes it difficult for the model to adequately extract the features of the surface defects. These problems pose a great challenge to the deep learning model, making it unable to obtain high classification accuracy. In this context, this research was conducted to achieve high classification accuracy by designing a novel method to classify aluminum profile surface defects.

The main contributions of the current work are as follows. Firstly, more defect images were obtained for classification training by means of digital image processing, such as rotation, flip, brightness, and contrast transformation. Secondly, a novel model RepVGG with a convolutional block attention module (RepVGG-CBAM) was proposed and used

to classify ten types of aluminum profile surface defects, and the classification accuracy reached 99.41%. Moreover, the superiority of the proposed method was demonstrated by comparative experiments and ablation experiments. Our method provides a reference for solving the problem of classifying aluminum profile surface defects.

The rest of this article is organized as follows. The second section introduces the methods involved in the experiment. The third section presents the experiment and results. The proposed method is discussed in the fourth section. The fifth section summarizes our article.

2. Methodology

2.1. Data Augmentation

It can be seen from Table 1 that although the types of aluminum profile defect images collected in this dataset are relatively rich, both the number of various defects and the overall number are significantly smaller compared to large datasets such as ImageNet and COCO. In addition, the distribution ratio of each type of defect image is shown in Figure 1, where eb defect images are the most numerous, accounting for 19.38% of the dataset, which is much higher than the other types of defects. The smallest amount is for the pb defect, which accounts for only 2.95%. The percentage of other types of defects also varies, which indicates that there is an uneven distribution of the number of samples with different types of defects in this dataset.

Table 1. Number of aluminum profile defect images of each type.

Type	cl	ds	eb	ecb	gra	mc	nc	op	pb	spf
Number	407	261	538	346	128	365	390	173	82	86
Total	2776									

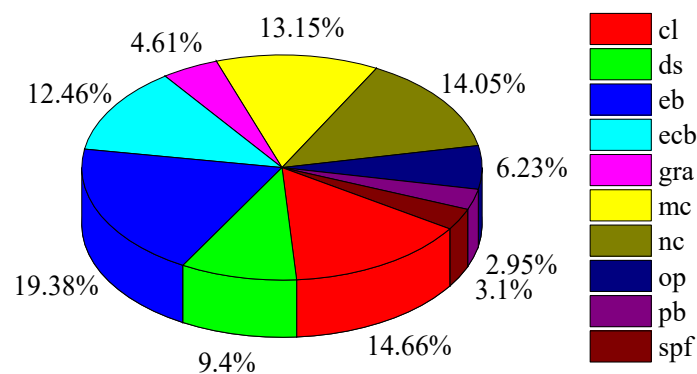


Figure 1. The distribution ratio of each type of defect image.

In the field of industrial defect classification based on deep learning, it is extremely easy for the overfitting phenomenon to occur during the training process, influenced by the lack of samples. Moreover, the unbalanced number of samples will also negatively contribute to the defect classification results. The most direct and effective way to avoid the overfitting phenomenon and improve the classification accuracy of the model is to augment the dataset. In view of the above two problems, we augmented the database based on the original defect images using several traditional image-processing means as follows. Figure 2 shows the image before and after transformation.

- (1) Rotation: By generating samples of aluminum surface defects at different angles for the classification model to learn, the sensitivity of the model to defects at arbitrary angles was improved.
- (2) Flip: Changing the position distribution of defects in the image provided image samples with richer defect position distribution.

- (3) Brightness transformation: Considers environmental factors. We simulated the actual variation of brightness in an aluminum profile production plant so as to improve the adaptability of the model to a complex brightness environment.
- (4) Contrast transformation: By changing the contrast of defect images, defect samples with different contrasts were added for classification model training.

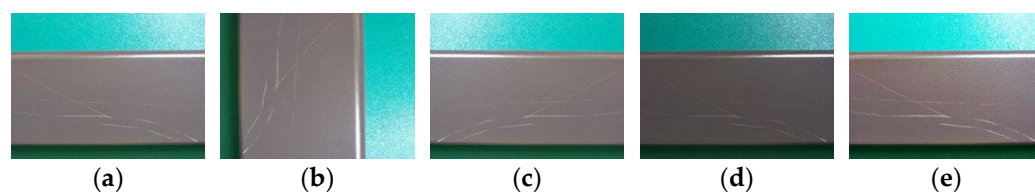


Figure 2. Data augmentation. (a) Original image; (b) Rotation; (c) Flip; (d) Brightness transformation; (e) Contrast transformation.

The distribution of the number of defect images of each type after data augmentation is shown in Table 2. It can be seen that the problem of small and unbalanced numbers of various types of defect images in the original dataset was solved.

Table 2. The number of defect images in the augmented dataset.

Type	cl	ds	eb	ecb	gra	mc	nc	op	pb	spf
Number	814	783	1076	1038	768	730	780	1038	738	774
Proportion (%)	9	9	13	12	9	9	9	12	9	9

2.2. RepVGG

Before 2015, research on deep learning models focused on single-branch networks, among which the visual geometry group (VGG) network received the most attention. It is fast, flexible, and has excellent feature-fitting ability due to the fact that the VGG [22] network includes only 3×3 convolution, the ReLU activation function, and pooling layers. Since the deep residual network (ResNet) [23] network was proposed, multi-branch models have shown more powerful characterization capability. Many researchers have shifted their research interests to designing complex model structures. The class of models with VGG architecture gradually faded out of the limelight. However, complex multi-branch structures will slow down inference, reduce memory utilization, and make models difficult to deploy. In 2021, Ding et al. [24] were inspired by the idea of the residual structure of ResNet. They designed the RepVGG algorithm by adding 1×1 branches and identity branches to the VGG-style network. The structure of RepVGG is shown in Figure 3. The network uses two residual structures: RepVGG block A, which contains only conv 3×3 and conv 1×1 branches, and RepVGG block B, which contains conv 3×3 , conv 1×1 , and identity branches. As shown in Figure 3d, the RepVGG training network is obtained by stacking RepVGG blocks and the ReLU activation function. Benefiting from multiple branches, the training RepVGG not only mitigates the gradient disappearance problem of the deep network, but also obtains more robust feature representations. After training, it is equivalently transformed into a single-branch deployment model, as shown in Figure 3e, through model re-parameterization, which has a faster inference speed.

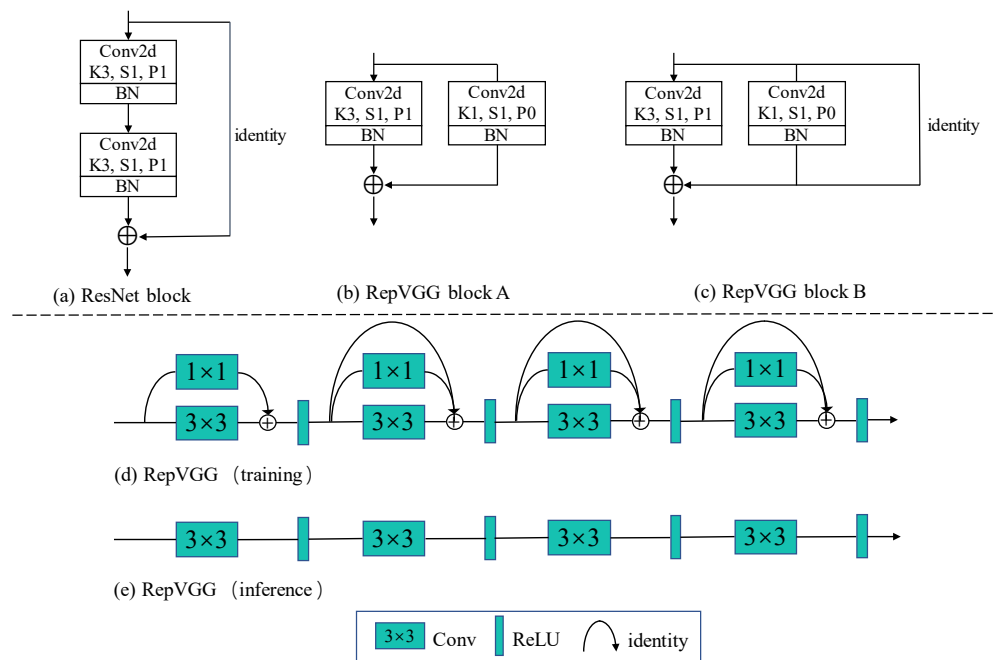


Figure 3. The structure of RepVGG: (a) Residual block of ResNet; (b) Residual block A of RepVGG; (c) Residual block B of RepVGG; (d) Sketch of RepVGG structure during training; (e) Sketch of RepVGG structure during inference.

Figure 4 displays the process of re-parameterization of the RepVGG residual block. The RepVGG block contains batch normalization (BN) layers in each branch, which can effectively solve gradient disappearance and gradient explosion, but the BN layers occupy a large amount of memory during forward inference, which increases the model’s inference time [25,26]. Therefore, to improve inference speed, the convolutional and BN layers are merged. The formulas for the convolution and BN layers can be written as:

$$Conv(x) = W(x) + b \tag{1}$$

$$BN(x) = \gamma \times \frac{(x - mean)}{\sigma} + \beta \tag{2}$$

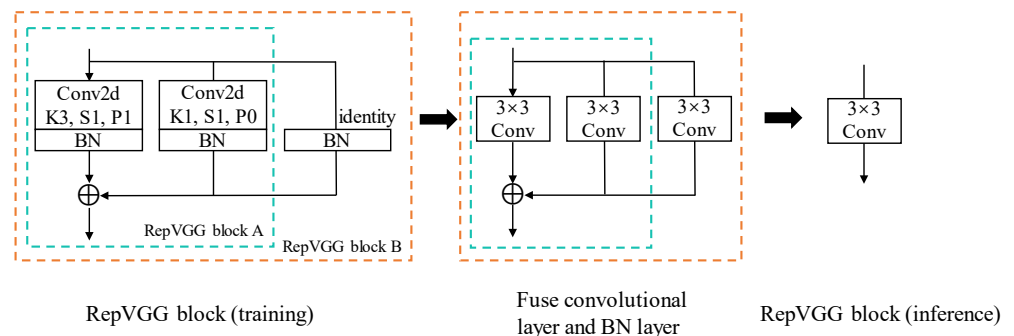


Figure 4. The process of structural re-parameterization of RepVGG block A and RepVGG block B.

Substituting Equation (1) into Equation (2), we have:

$$\begin{aligned} BN(Conv(x)) &= \gamma \times \frac{[(W(x)+b)-mean]}{\sigma} + \beta \\ &= \frac{\gamma \times W(x)}{\sigma} + \frac{\gamma \times \mu}{\sigma} + \beta \end{aligned} \tag{3}$$

where $W()$ denotes the convolution kernel operation, b is the bias, γ is the scaling factor, β is the bias, σ is the standard deviation, and $\mu = b - mean$ is the cumulative mean. As

shown in Figure 4, after fusing the convolutional and BN layers in each branch into a convolutional layer with bias, the 1×1 branch and the identity branch are converted into 3×3 convolutional kernels, respectively. The three 3×3 convolutional kernels are then summed. In this way, each RepVGG block can be converted into a 3×3 convolutional layer and the output is exactly the same before and after the conversion. Therefore, the trained model can be converted to a single-channel model with only 3×3 convolutional layers.

From the perspective of parameters, the RepVGG block transformation process can be described as the following form: when the number of channels and the length and width of the feature map before and after the convolution operation are equal, we have

$$\begin{aligned} M^{out} &= BN(M^{input} * W^{(3)}, \mu^{(3)}, \sigma^{(3)}, \gamma^{(3)}, \beta^{(3)}) \\ &+ BN(M^{input} * W^{(1)}, \mu^{(1)}, \sigma^{(1)}, \gamma^{(1)}, \beta^{(1)}) \\ &+ BN(M^{input} * W^{(0)}, \mu^{(0)}, \sigma^{(0)}, \gamma^{(0)}, \beta^{(0)}) \end{aligned} \quad (4)$$

where BN is the inference-time BN function. $W^{(k)} \in R^{C_2 \times C_1 \times k \times k}$ ($k = 3, 1$) denotes a convolution kernel of size $k \times k$ with C_1 input channel and C_2 output channel. $\mu^{(k)}$, $\sigma^{(k)}$, $\gamma^{(k)}$, and $\beta^{(k)}$ ($k = 3, 1, 0$) denote the cumulative mean, standard deviation, scaling factor, and bias of the BN layers of each branch, respectively. When $k = 3$, they are for the conv 3×3 branch, when $k = 1$, they are for the conv 1×1 branch, and when $k = 0$, they are for the identity branch. $M^{(1)} \in R^{N \times C_1 \times H_1 \times W_1}$ and $M^{(2)} \in R^{N \times C_2 \times H_2 \times W_2}$ are the input and output, respectively, and $*$ is the convolution operator. Particularly, $C_1 = C_2$, $H_1 = H_2$, and $W_1 = W_2$. Otherwise, we used RepVGG block A, hence Equation (4) only has the first two terms. Formally, according to Equation (2), when $\forall 1 \leq i \leq C_2$, Equation (5) can be obtained.

$$\begin{aligned} BN(M, \mu, \sigma, \gamma, \beta)_{:,i,:} &= (M_{:,i,:} - \mu_i) \frac{\gamma_i}{\sigma_i} + \beta_i \\ &= \frac{\gamma_i}{\sigma_i} M_{:,i,:} + (\beta_i - \frac{\mu_i \gamma_i}{\sigma_i}) \end{aligned} \quad (5)$$

In this way, the convolutional and BN layers on each branch are first converted into a convolution with bias vector. Let W' denote the convolution kernel and b' be the bias vector. They can be formulated as:

$$W'_{:,i,:} = \frac{\gamma_i}{\sigma_i} M_{:,i,:} \quad (6)$$

$$b'_i = \beta_i - \frac{\mu_i \gamma_i}{\sigma_i} \quad (7)$$

Then, it is easy to verify that when $\forall 1 \leq i \leq C_2$, we can get Equation (8):

$$BN(M, \mu, \sigma, \gamma, \beta)_{:,i,:} = (M * W')_{:,i,:} + b'_i \quad (8)$$

Essentially, the identity branch can be regarded as a special 1×1 convolution kernel with the weights of different channels fixed to 1. The 1×1 convolution kernel can be regarded as a special 3×3 convolution kernel. By means of padding, two 1×1 convolution kernels can be converted into 3×3 convolution kernels with the middle element being 1 and the other elements being 0. Therefore, the final convolution kernel can be obtained by adding the three 3×3 convolution kernels in the three branches, and the final bias is equal to the sum of the three biases.

In reference [24], the authors proposed a series of RepVGG networks. We selected the RepVGG_{B3g4} network in our study, whose feature extraction structure is shown in Table 3.

Table 3. Structure of RepVGG_{B3g4} network.

Stage	Output Size	Layers of Each Stage	Number of Channels
1	112 × 112	1	64
2	56 × 56	4	192
3	28 × 28	6	384
4	14 × 14	16	768
5	7 × 7	1	2560

2.3. CBAM Attention Mechanism

An attention mechanism is a way to achieve adaptive attention in the network. Generally speaking, it lets the network attach more importance to effective units and suppress invalid units during feature extraction. Common attention mechanisms include squeeze and excitation networks (SENet) [27], convolutional block attention modules (CBAMs) [28], efficient channel attention modules (ECAs) [29], etc. The structure of a CBAM is shown in Figure 5. It consists of two parts, the channel attention module (CAM) and the spatial attention module (SAM), which means that it can pay attention to the channel information and the location information of the object. From our perspective, it will contribute to addressing the problem of lack of uniformity in size and spatial distribution that exist in the aluminum profile surface defects. Therefore, the attention mechanism used in our study was CBAM. For the input feature map F , it can perform attention operations in the channel and spatial dimensions successively.

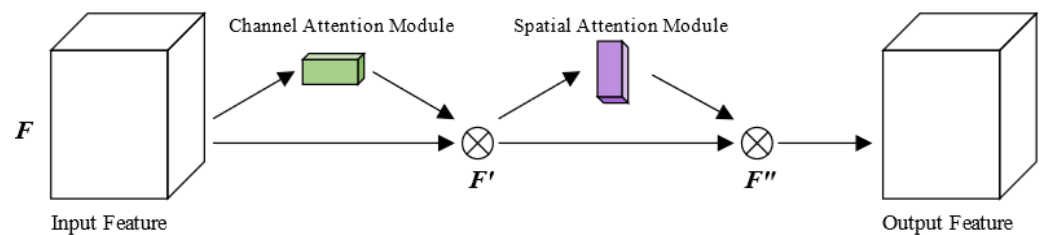


Figure 5. Structure of CBAM attention mechanism.

Let M_c be the attention mapping operation in the channel dimension and M_s be the attention mapping operation in the spatial dimension. Then the attention channel operation can be formulated as:

$$\begin{aligned}
 M_c(F) &= \sigma(MLP(AvgPool(F))) + \sigma(MLP(MaxPool(F))) \\
 &= \sigma(W_1(W_0(F_{avg}^c))) + \sigma(W_1(W_0(F_{max}^c)))
 \end{aligned}
 \tag{9}$$

where F denotes the input, σ is the sigmoid function, and MLP denotes the multi-layer perceptron model, $W_0 \in R^{C/r \times C}$ and $W_1 \in R^{C \times C/r}$. The CAM compresses the spatial information of a feature map by using both global max pooling and global average pooling to obtain two different spatial context descriptors: F_{avg}^c and F_{max}^c . Then, they are computed using a shared network composed of the shared MLP . The feature vectors of the MLP output are summed element by element. In addition, the channel attention feature map is produced by the sigmoid function. Finally, the output of the CAM is obtained by multiplying the original feature map with the channel attention feature map, as shown in Equation (10).

$$F' = M_c(F) \otimes F
 \tag{10}$$

where \otimes denotes element-wise multiplication.

The SAM takes the CAM output feature map F' as input, and its calculation process can be written as

$$\begin{aligned}
 M_s(F') &= \sigma(f^{7 \times 7}([AvgPool(F'); MaxPool(F')])) \\
 &= \sigma(f^{7 \times 7}([F'_{avg}^s; F'_{max}^s]))
 \end{aligned}
 \tag{11}$$

where $f^{7 \times 7}$ denotes a convolution operation with a filter size of 7×7 . Firstly, global max pooling and global average pooling are performed on the feature maps F' across the channel to obtain two 2D feature maps, F'^{avg} and F'^{max} . Then, they are concatenated and convolved by a 7×7 convolution kernel. Furthermore, the sigmoid function is used for normalization to obtain the spatial attention feature map. Finally, as shown in Equation (12), the feature map of the CBAM is obtained by element-wise multiplying $M_s(F')$ with F' .

$$F'' = M_s(F') \otimes F' \tag{12}$$

2.4. Our Proposed Method (RepVGG-CBAM)

At the time of its presentation, the RepVGG network demonstrated a strong classification capability on the ImageNet dataset. In a subsequent study, Feng et al. [30] proposed RepVGG_B3g4 + SA by combining RepVGG with a spatial attention module. The model was successfully applied to the strip steel surface defect classification task and obtained a classification accuracy of 95.10%, which was higher than that of the basic RepVGG network. According to current experience, adding an attention mechanism to the network can improve network performance. The CBAM attention module focuses on the channel and location information of the object, which is suitable for solving the problem of large size variation and irregular location distribution of aluminum profile surface defects. Based on this idea, we combined RepVGG_{B3g4} with CBAM to propose the RepVGG-CBAM model. Its structure is shown in Figure 6. The CBAM module is added following Stage 1 through Stage 4 of the basic RepVGG_{B3g4}. From our perspective, the performance of the RepVGG-CBAM will be greatly improved over the basic network. In the later parts of this paper, we will provide experiments to verify this conjecture and compare it with other networks to demonstrate the superiority of our proposed method.

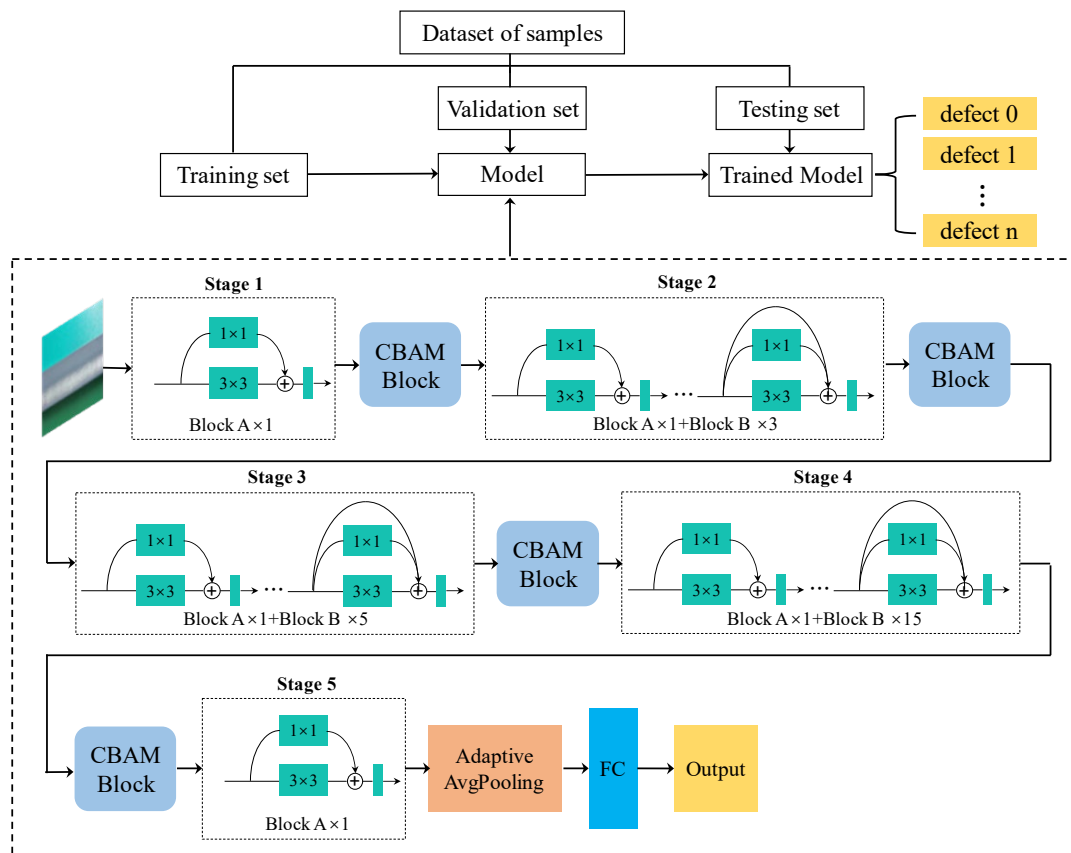


Figure 6. Our method.

The overall process of our method is as follows: firstly, the dataset is augmented by using digital image-processing methods (PyCharm Community Edition 2021.2.2, JstBrains s.r.o., Prague, Czech Republic). Then, the augmented dataset is divided into a training set, a validation set, and a testing set. Finally, the test set is responsible for evaluating model performance and the output of classification results.

3. Experiment and Results

3.1. Dataset

The dataset used in our research is available on the AliCloud platform. As shown in Figure 7, this dataset contains ten types of aluminum profile surface defects, including concave line (cl), dirty spot (ds), exposed bottom (eb), exposed corner bottom (ecb), graze (gra), mixed color (mc), non-conductivity (nc), orange peel (op), paint bubble (pb), and spray paint flow (spf). To reduce the training time and computational complexity, these images were scaled down to 224×224 pixels and applied to the network training.

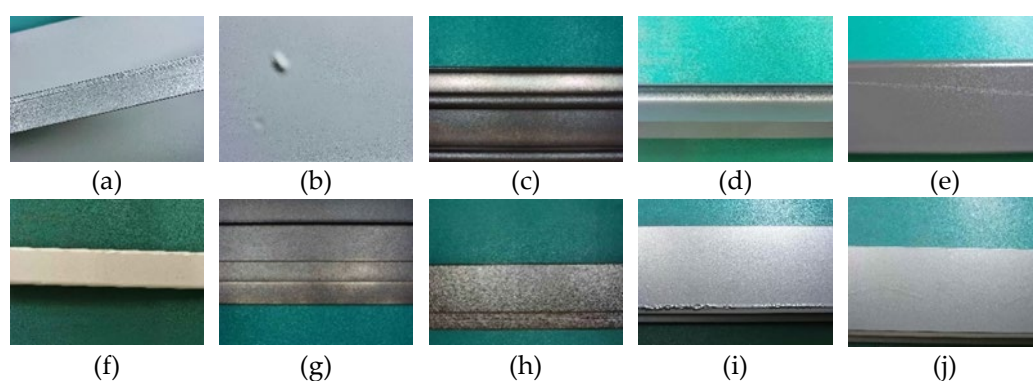


Figure 7. Aluminum profile surface defect images in ten categories: (a) concave line (cl); (b) dirty spot (ds); (c) exposed bottom (eb); (d) exposed corner bottom (eb); (e) graze (gra); (f) mixed color (mc); (g) non-conductive (nc); (h) orange peel (op); (i) paint bubble (pb); (j) spray paint flow (spf).

The augmented dataset contains a total of 8539 defect images. As shown in Table 4, the dataset was divided into a training set, a validation set, and a testing set; 10% of all images were randomly selected as the testing set. Among the remaining images, 80% were randomly selected as the training set and 20% as the validation set to train the model.

Table 4. The division of training set, validation set, and testing set images.

Defect Class	Training Set	Validation Set	Testing Set
cl	587	146	81
ds	564	141	78
eb	776	193	107
ecb	748	187	103
gra	554	138	76
mc	526	131	73
nc	562	140	78
op	748	187	103
pb	532	133	73
spf	558	139	77
total	6155	1535	849

3.2. Experimental Environment and Training Parameters

All experiments were performed on a computer (Lenovo Legion R7000P2021H, Lenovo (Beijing) Ltd, Beijing, China) with an AMD CPU Ryzen 7-5800H@3.20Ghz, 512 GB DDR4 memory, an NVIDIA GEFORCE RTX3060 graphics processing unit (GPU) with 6 GB memory, and Windows10 operating system with 16 GB memory. All experiments were

performed using Python 3.8, NVIDIA CUDA-11.1.1 and cuDNN-11.2, and the compiler environment was established by PyTorch 1.8 deep learning framework.

The parameters of the model have a great influence on model performance. Suitable parameters can improve the convergence speed and accuracy of the model. The main parameters of the network during pre-training were set as shown in Table 5. We chose the Adam optimizer with a learning rate of 0.0001. We set the batch to 16 and the epoch to 100.

Table 5. Parameters of the training process.

Parameters	Setting
Optimizer	Adam
Learning rate	0.0001
Batch size	16
Epoch	100

Figure 8 displays the accuracy and loss curves of RepVGG-CBAM for the training process; train_acc and train_loss represent the accuracy and loss of the training process, respectively, and val_acc and val_loss represent the accuracy and loss of the validation process, respectively. It demonstrates that after the network was initialized, the classification ability of the model was weak, and the initial training accuracy was only 59.63%. Moreover, the accuracy values of the training and validation sets increased rapidly until the first 15 iterations, and then showed a slow increasing trend. Correspondingly, the loss value decreased rapidly at the initial stage, and gradually converged with the increase of the number of iterations. After training, the loss was close to zero. During the iteration, when the epoch was 98, the training accuracy and loss were 99.74% and 0.0066, respectively, and the validation accuracy and loss were 97.41% and 0.0172, respectively, which were the best in the whole training process. Therefore, after the training was completed, the weights of this epoch were adopted for testing.

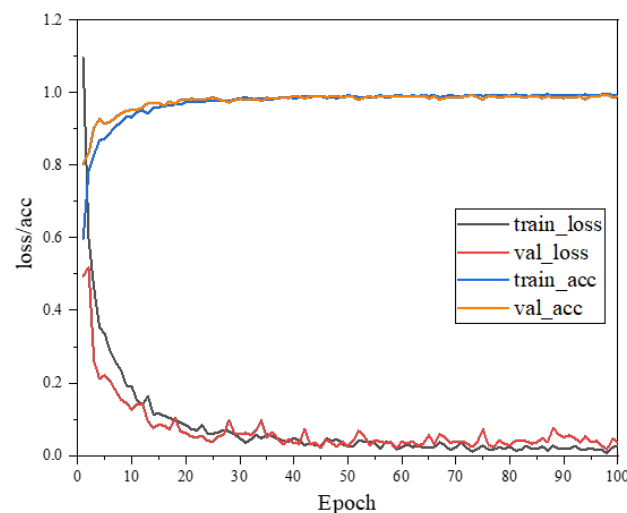


Figure 8. Accuracy and loss curves of RepVGG-CBAM during training.

3.3. Evaluation Method

The classification results can be divided into four cases: true positive (*TP*), false positive (*FP*), true negative (*TN*), and false negative (*FN*). In this paper, *Precision*, *Recall*, and *F1* values were used to evaluate the classification performance of the model for various types of defects. *Accuracy*, *Macro-precision*, *Macro-recall*, and *Macro-F1* were used to evaluate the overall performance of the model. They can be expressed as:

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

$$Recall = \frac{TP}{TP + FN} \quad (14)$$

$$F1 = \frac{2PR}{P + R} \quad (15)$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (16)$$

$$Macro - Precision = \frac{1}{N} \sum_{i=1}^N \frac{TP_i}{TP_i + FP_i} \quad (17)$$

$$Macro - Recall = \frac{1}{N} \sum_{i=1}^N \frac{TP_i}{TP_i + FN_i} \quad (18)$$

$$Macro - F1 = \frac{1}{N} \sum_{i=1}^N \frac{2 \times Precision_i \times Recall_i}{Precision_i + Recall_i} \quad (19)$$

3.4. Defect Classification Test Results

In order to graphically show the distribution of the prediction results of our method for each type of defect, Figure 9 displays a confusion matrix of the defect classification generated by the testing. The columns of the confusion matrix represent the real types of defects, and the rows represent the types of defects predicted by the model. It can be seen that two ds defect images were incorrectly classified as gra and three spf defect images were incorrectly classified as pb. This is mainly because the background area of some ds defect images had features similar to gra defects, leading the model to recognize the wrong units. Pb defects are small in size and spf defects are extremely inconspicuous and have similar color features to the background area. These factors mean the model cannot fully extract their defect features and the model is prone to misclassification.

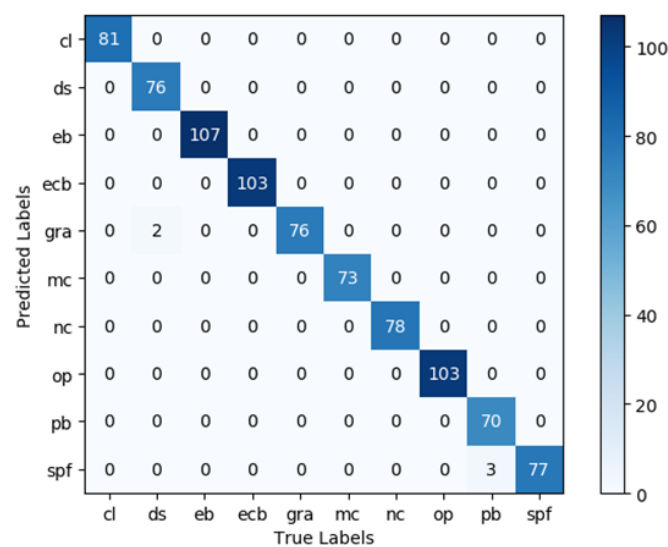


Figure 9. Confusion matrix.

So as to present the classification performance of RepVGG-CBAM more intuitively for various types of defects, the four indicators shown in Equations (13)–(16) were used to evaluate the classification results. The specific values are shown in Table 6. As can be seen, the model can perfectly classify six types of defects: cl, eb, ecb, mc, nc, and op. The precision, recall, and F1 reached 100% for all of them. In addition, the probability of misclassification between defects spf and pb was the highest, and the precision for the spf defect was the lowest, 96.25%. The recall and F1 for the pb defect were the lowest, 95.89% and 97.89%, respectively. However, the classification accuracy of the model reached

99.41% overall, which indicates that our proposed method has an excellent ability to classify aluminum profile surface defects.

Table 6. Evaluation of classification performance.

Label	Precision (%)	Recall (%)	F1 (%)	Accuracy (%)
cl	100	100	100	
ds	100	97.44	98.70	
eb	100	100	100	
ecb	100	100	100	
gra	97.44	100	98.70	
mc	100	100	100	99.41
nc	100	100	100	
op	100	100	100	
pb	100	95.89	97.9	
spf	96.25	100	98.09	

4. Discussion

4.1. Comparison of Different Defect Classification Algorithms

In order to verify the superior performance of our method, six classification algorithms—VGG16, VGG19, ResNet34, ResNet50, ShuffleNet_v2, and RepVGG_{B3g4}—were selected to classify aluminum profile surface defects under the same experimental conditions. The classification results of each model are shown in Table 7. It can be seen that all four indicators of our proposed method are higher than 99.00%, which is better than other methods. This also indicates that our proposed method has better feature extraction ability and robustness. The classification accuracy of our RepVGG-CBAM is 4.85% better than that of the basic RepVGG algorithm, indicating that the CBAM plays a positive role.

Table 7. Comparison results of different models.

Methods	Accuracy (%)	Macro Precision (%)	Macro Recall (%)	Macro F1 (%)
VGG16	98.35	98.21	98.16	98.18
VGG19	97.53	97.25	97.32	97.28
ResNet34	97.41	97.23	97.18	97.29
ResNet50	97.76	97.84	97.78	97.80
ShuffleNet_v2	97.64	97.48	97.43	97.42
RepVGG _{B3g4}	98.82	98.77	98.71	98.73
RepVGG-CBAM (ours)	99.41	99.37	99.33	99.34

Figure 10 shows the accuracy curves for the training set for each method. It demonstrates that the curves of each method can be stable after the completion of the iteration. With the exception of VGG19 and ShuffleNet_v2, the accuracy of all the methods could reach more than 95%, mainly because VGG19 has a large number of parameters and needs more samples to train the model and achieve satisfactory accuracy. While ShuffleNet_v2 is a lightweight network, its network layer is shallow, which diminishes its recognition ability. In terms of convergence speed, VGG19 was the slowest due to the huge number of parameters, in contrast to RepVGG_{B3g4} and our method. It is worth noting that compared to RepVGG_{B3g4}, the network structure of our method becomes more complex after adding multiple CBAM blocks. It nevertheless maintained almost the same convergence speed as RepVGG_{B3g4}, and our method had the highest accuracy. This indicates that our enhancement of the RepVGG network was positive.

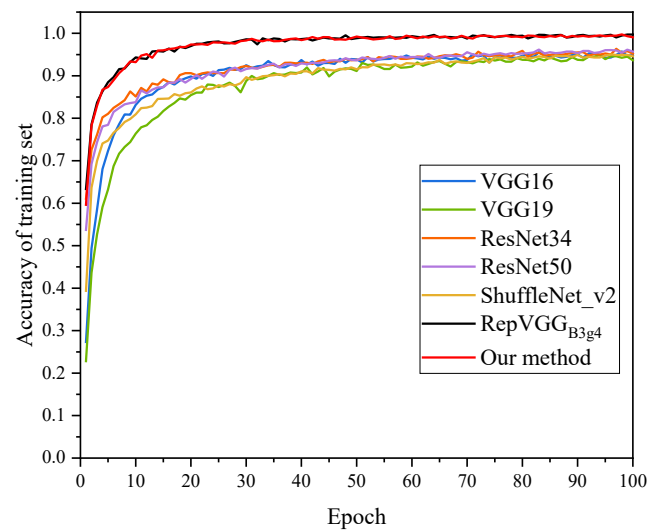


Figure 10. Comparison of the accuracy of each method training set.

The loss curves of each method in the validation set are shown in Figure 11. It can be seen that VGG16, VGG19, ResNet34, and ResNet50 have large fluctuations and are less stable. The curve of ShuffleNet_v2 is the smoothest. There are minor fluctuations in the curves of our method, but the loss values are the lowest and show an overall smooth decreasing trend. Compared with RepVGG_{B3g4}, our method has less fluctuation, which indicates that the stability of the network has been improved. Overall, our method achieved a stable training process, the lowest loss values, and the highest accuracy, so it is optimal for classifying aluminum profile surface defects.

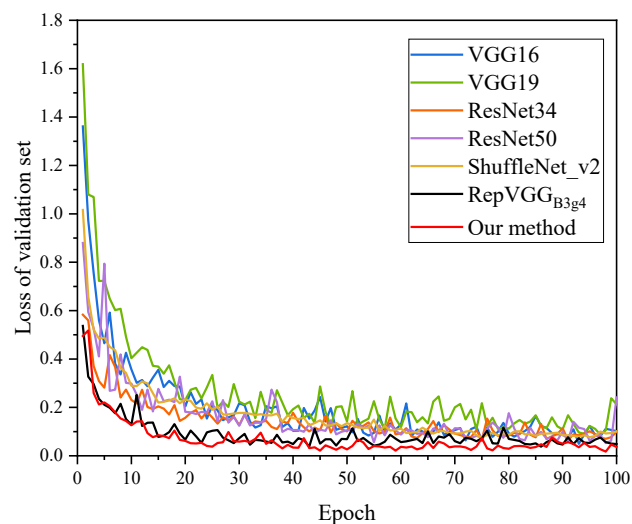


Figure 11. Comparison of the loss of each method validation set.

4.2. Ablation Study

An ablation study was conducted to enable us to better understand how the CBAM attention mechanism can help improve RepVGG performance. The CBAM attention module was added following different stages of RepVGG. The results of the ablation study are shown in Table 8. It can be seen that the classification accuracy was the lowest when the CBAM was added following all five stages, which was 98.58%, even lower than the basic RepVGG, which proves that it is not better to add more CBAMs. The highest classification accuracy of 99.41% was achieved when the CBAMs were added following Stage 1 through Stage 4, which was 4.85% better than the basic RepVGG. These results show that choosing

an appropriate way to integrate the CBAM into the original network can improve network performance. It also verifies the effectiveness of our proposed method.

Table 8. Results of the ablation study.

Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Accuracy (%)
✓	-	-	-	-	98.94
✓	✓	-	-	-	99.29
✓	✓	✓	-	-	99.17
✓	✓	✓	✓	-	99.41
✓	✓	✓	✓	✓	98.58

5. Conclusions

1. To address the problem of small and unbalanced numbers of various types of defect images in the original dataset, digital image-processing methods such as rotation, flip, contrast transformation, and brightness transformation were used to augment our dataset. Not only does this simulate the environment of the actual production conditions, but it also generates a large number of sample images for model training.
2. A RepVGG-CBAM model was proposed by combining CBAM based on the RepVGG_{B3g4} algorithm and used to classify ten types of aluminum profile surface defects. The training process of this model was stable without overfitting. Our RepVGG-CBAM algorithm achieved promising results. Six types of defects: cl, eb, ecb, mc, nc, and op, could be perfectly classified, and their precision, recall, and F1 reached 100%. The classification accuracy of our method was 99.41%. The outstanding performance of RepVGG-CBAM demonstrated the advantages of our method in classifying surface defects in aluminum profiles.
3. The classification accuracy of our RepVGG-CBAM was 4.85% better than that of the basic RepVGG algorithm, indicating that integrating a CBAM had a positive effect. In addition, the results of comparative experiments confirm that the accuracy, macro precision, macro recall, and macro F1 of our proposed method were the highest; it outperformed VGG16, VGG19, ResNet34, ResNet50, ShuffleNet_v2, and RepVGG_{B3g4}. It indicates that our proposed RepVGG-CBAM is an advanced algorithm for classifying surface defects in aluminum profiles. Moreover, the results of the ablation study demonstrated that the classification ability was strongest when the CBAM attention mechanism was added following Stage 1 through Stage 4 of RepVGG. This provides a certain basis for later related studies.

Although the experimental results demonstrated the effectiveness of the RepVGG-CBAM algorithm, we found that the algorithm performance was not very good on defects of small size such as pb. In the future, we will consider integrating CBAM into the residual blocks of RepVGG to further improve classification accuracy. In addition, we will also proceed with network lightweighting to try out practical applications in engineering.

Author Contributions: Conceptualization, Z.L. and B.L.; methodology, B.L., Z.L. and F.R.; software, B.L.; validation, Z.L., X.K. and S.L.; formal analysis, Z.L., B.L. and F.R.; investigation, Z.L.; resources, H.N.; data curation, Z.L. and B.L.; writing—original draft preparation, B.L.; writing—review and editing, H.N. and F.R.; visualization, Z.L. and X.K.; supervision, Z.L. and S.L.; project administration, H.N.; funding acquisition, H.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Priority Academic Program Development of Jiangsu Higher Education Institutions, grant number PAPD; Jiangsu Province Policy Guidance Program (International Science and Technology Cooperation) Project, grant number BZ2021045; Nantong Applied Research Project, grant number JCZ21066; University-Industry Collaborative Education Program, grant number 202102236001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Zhou, W.; Shao, Z.; Yu, J.; Lin, J. Advances and Trends in Forming Curved Extrusion Profiles. *Materials* **2021**, *14*, 1603. [[CrossRef](#)] [[PubMed](#)]
2. Wang, Y.; Zhao, G. Hot Extrusion Processing of Al–Li Alloy Profiles and Related Issues: A Review. *Chin. J. Mech. Eng.* **2020**, *33*, 64. [[CrossRef](#)]
3. Liu, Z.; Li, L.; Li, S.; Yi, J.; Wang, G. Simulation Analysis of Porthole Die Extrusion Process and Die Structure Modifications for an Aluminum Profile with High Length–Width Ratio and Small Cavity. *Materials* **2018**, *11*, 1517. [[CrossRef](#)] [[PubMed](#)]
4. Romanova, V.; Balokhonov, R.; Zinovieva, O.; Shakhidzhanov, V.; Dymnich, E.; Nekhorosheva, O. The relationship between mesoscale deformation-induced surface roughness, in-plane plastic strain and texture sharpness in an aluminum alloy. *Eng. Fail. Anal.* **2022**, *137*, 106377. [[CrossRef](#)]
5. Wouters, O.; Vellinga, W.P.; van Tijum, R.; De Hosson, J.T.M. Effects of crystal structure and grain orientation on the roughness of deformed polycrystalline metals. *Acta Mater.* **2006**, *54*, 2813–2821. [[CrossRef](#)]
6. Wang, Q.; Yang, R.; Wu, C.; Liu, Y. An effective defect detection method based on improved Generative Adversarial Networks (iGAN) for machined surfaces. *J. Manuf. Process.* **2021**, *65*, 373–381. [[CrossRef](#)]
7. Karthikeyan, S.; Pravin, M.C.; Sathyabama, B.; Mareeswari, M. DWT Based LCP Features for the Classification of Steel Surface Defects in SEM Images with KNN Classifier. *Aust. J. Basic Appl. Sci.* **2016**, *10*, 13–19.
8. Wei, P.; Liu, C.; Liu, M.; Gao, Y.; Liu, H. CNN-based reference comparison method for classifying bare PCB defects. *J. Eng.* **2018**, *2018*, 1528–1533. [[CrossRef](#)]
9. Ricci, M.; Ficola, A.; Fravolini, M.L.; Battaglini, L.; Palazzi, A.; Burrascano, P.; Valigi, P.; Appolloni, L.; Cervo, S.; Rocchi, C. Magnetic imaging and machine vision NDT for the on-line inspection of stainless steel strips. *Meas. Sci. Technol.* **2012**, *24*, 25401. [[CrossRef](#)]
10. Zaghdoudi, R.; Seridi, H.; Ziani, S. Binary Gabor pattern (BGP) descriptor and principal component analysis (PCA) for steel surface defects classification. In Proceedings of the 2020 International Conference on Advanced Aspects of Software Engineering (ICAASE), Constantine, Algeria, 28–30 November 2020; pp. 1–7.
11. Hu, H.; Li, Y.; Liu, M.; Liang, W. Classification of defects in steel strip surface based on multiclass support vector machine. *Multimed. Tools Appl.* **2014**, *69*, 199–216. [[CrossRef](#)]
12. Chondronasios, A.; Popov, I.; Jordanov, I. Feature selection for surface defect classification of extruded aluminum profiles. *Int. J. Adv. Manuf. Technol.* **2016**, *83*, 33–41. [[CrossRef](#)]
13. Ma, Z.; Li, Y.; Huang, M.; Huang, Q.; Cheng, J.; Tang, S. A lightweight detector based on attention mechanism for aluminum strip surface defect detection. *Comput. Ind.* **2022**, *136*, 103585. [[CrossRef](#)]
14. Duan, C.; Zhang, T. Two-Stream Convolutional Neural Network Based on Gradient Image for Aluminum Profile Surface Defects Classification and Recognition. *IEEE Access* **2020**, *8*, 172152–172165. [[CrossRef](#)]
15. Abualighah, S.M.; Al-Naimi, A.F.; Duwairi, R.M. DD-SSD: Deep Detector for Strip Steel Defects. In Proceedings of the 2022 13th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 21–23 June 2022; pp. 9–14.
16. Zhang, J.; Li, S.; Yan, Y.; Ni, Z.; Ni, H. Surface Defect Classification of Steel Strip with Few Samples Based on Dual-Stream Neural Network. *Steel Res. Int.* **2022**, *93*, 2100554. [[CrossRef](#)]
17. Liu, X.; He, W.; Zhang, Y.; Yao, S.; Cui, Z. Effect of dual-convolutional neural network model fusion for Aluminum profile surface defects classification and recognition. *Math. Biosci. Eng.* **2022**, *19*, 997–1025. [[CrossRef](#)]
18. Mayr, M.; Hoffmann, M.; Maier, A.; Christlein, V. Weakly supervised segmentation of cracks on solar cells using normalized Lp norm. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 22–25 September 2019; pp. 1885–1889.
19. Tabernik, D.; Aela, S.; Skvarč, J.; Skočaj, D. Segmentation-based deep-learning approach for surface-defect detection. *J. Intell. Manuf.* **2020**, *31*, 759–776. [[CrossRef](#)]
20. Huang, Y.; Qiu, C.; Yuan, K. Surface defect saliency of magnetic tile. *Vis. Comput.* **2020**, *36*, 85–96. [[CrossRef](#)]
21. Hao, Z.; Li, Z.; Ren, F.; Lv, S.; Ni, H. Strip Steel Surface Defects Classification Based on Generative Adversarial Network and Attention Mechanism. *Metals* **2022**, *12*, 311. [[CrossRef](#)]
22. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
23. 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, 27–30 June 2016; pp. 770–778.
24. Ding, X.; Zhang, X.; Ma, N.; Han, J.; Ding, G.; Sun, J. Repvgg: Making vgg-style convnets great again. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13733–13742.
25. Qing, Y.; Liu, W.; Feng, L.; Gao, W. Improved YOLO Network for Free-Angle Remote Sensing Target Detection. *Remote Sens.* **2021**, *13*, 2171. [[CrossRef](#)]

26. Wu, Z.; Wang, X.; Chen, C. Research on Lightweight Infrared Pedestrian Detection Model Algorithm for Embedded Platform. *Secur. Commun. Netw.* **2021**, *2021*, 1549772. [[CrossRef](#)]
27. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2011–2023. [[CrossRef](#)] [[PubMed](#)]
28. Woo, S.; Park, J.; Lee, J.; 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.
29. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. Supplementary material for ‘ECA-Net: Efficient channel attention for deep convolutional neural networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 13–19.
30. Feng, X.; Gao, X.; Luo, L. X-SDD: A New Benchmark for Hot Rolled Steel Strip Surface Defects Detection. *Symmetry* **2021**, *13*, 706. [[CrossRef](#)]