mathematics

Article

A ResNet50-Based Method for Classifying Surface Defects in Hot-Rolled Strip Steel

Xinglong Feng [1,†], Xianwen Gao [1,*] and Ling Luo [2,†]

[1] College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; 1610238@stu.neu.edu.cn
[2] Moviebook Technology Co., Ltd., Beijing 100027, China; ling_luo@moviebook.cn
* Correspondence: gaoxianwen@mail.neu.edu.cn
† These authors contributed equally to this work.

Abstract: Hot-rolled strip steel is widely used in automotive manufacturing, chemical and home appliance industries, and its surface quality has a great impact on the quality of the final product. In the manufacturing process of strip steel, due to the rolling process and many other reasons, the surface of hot rolled strip steel will inevitably produce slag, scratches and other surface defects. These defects not only affect the quality of the product, but may even lead to broken strips in the subsequent process, seriously affecting the continuation of production. Therefore, it is important to study the surface defects of strip steel and identify the types of defects in strip steel. In this paper, a scheme based on ResNet50 with the addition of FcaNet and Convolutional Block Attention Module (CBAM) is proposed for strip defect classification and validated on the X-SDD strip defect dataset. Our solution achieves a classification accuracy of 94.11%, higher than more than a dozen other compared deep learning models. Moreover, to adress the problem of low accuracy of the algorithm in classifying individual defects, we use ensemble learning to optimize. By integrating the original solution with VGG16 and SqueezeNet, the recognition rate of oxide scale of plate system defects improved by 21.05 percentage points, and the overall defect classification accuracy improved to 94.85%.

Keywords: hot rolled strip steel; deep learning; surface defects; defect classification

check for updates

Citation: Feng, X.; Gao, X.; Luo, L. A ResNet50-Based Method for Classifying Surface Defects in Hot-Rolled Strip Steel. *Mathematics* **2021**, *9*, 2359. https://doi.org/10.3390/math9192359

Academic Editors: Florin Leon, Mircea Hulea and Marius Gavrilescu

Received: 13 August 2021
Accepted: 18 September 2021
Published: 23 September 2021

1. Introduction

Hot-rolled strip steel is produced by rolling the billet at a temperature higher than the recrystallization temperature and then going through a series of processes such as phosphorus removal, finishing, polishing, edge cutting and straightening. Hot-rolled strip steel has good processing performance and strong coverage ability, which is widely used in automobile manufacturing, home appliance manufacturing, shipbuilding and chemical industry, etc. In the manufacturing process of strip steel, for various reasons [1–3], surface defects will inevitably arise, and these defects cannot be completely overcome by improving the process. Therefore, the detection of surface defects in hot rolled strip is an important part of hot rolled strip production and is closely related to the surface quality of the strip. Figure 1 shows the quality inspection process of surface defects in the actual production of a steel mill.

As shown in Figure 1, the hot rolled strip is first inspected by the hot rolled strip quality inspection system. The system takes high speed images of the top and bottom surfaces of the strip steel and determines the images that may have surface defects and passes them to the quality inspector. Since hot rolled strip passes through the quality inspection system very quickly, often in less than two minutes for a roll of strip to pass through the system, the quality inspector must judge the pictures coming from the system quickly. Strip steels judged to be normal by the quality inspectors will go directly to the next process, while coils judged to require further treatment will be given more specific treatment by the next batch of quality inspectors. Since the previous steps would have blocked the problematic steel coils, this batch of quality inspectors have more time to analyze the steel coils with surface defects and thus give the next instructions. After a series of processing, the finished strip coil is finally obtained as shown in Figure 2.

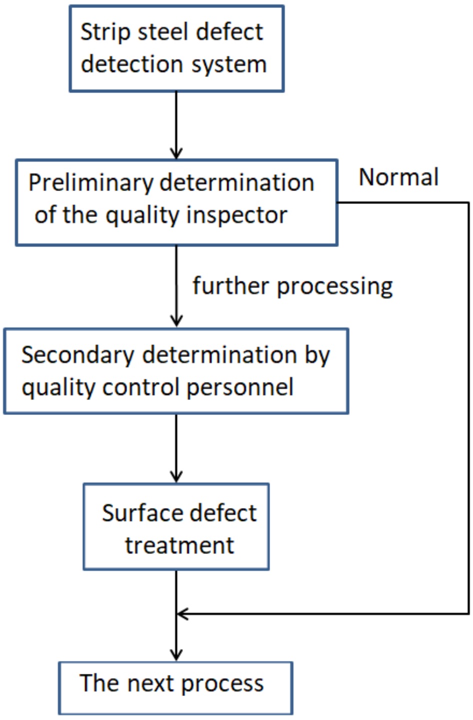

Strip steel defect detection system
Preliminary determination of the quality inspector
Normal
further processing
Secondary determination by quality control personnel
Surface defect treatment
The next process

Figure 1. Flow chart of strip defect detection.

Figure 2. The finished steel coils.

Although the above solution can meet the steel mill's requirements for strip surface quality, this solution has the following shortcomings: Firstly, strip production often takes place throughout the day, which requires quality inspectors who make preliminary judgments to work at night, and long hours of night work are detrimental to their health [4]. Secondly, for quality inspectors, the work of observing defective pictures for a long time is not only easy to produce visual fatigue but also very boring, and therefore easy to produce errors [5]. Last but not least, the work of quality inspection increases the cost of the steel mill because of the large amount of manpower required.

The main reason for the current use of manual further testing on the basis of the strip surface defect detection system is that the accuracy of the existing system is not yet as good as that of the quality inspectors. So the key question is how to improve the accuracy of this system to reach the average level of quality control workers. The strip surface defect detection system commonly used in steel mills today is shown in Figure 3.

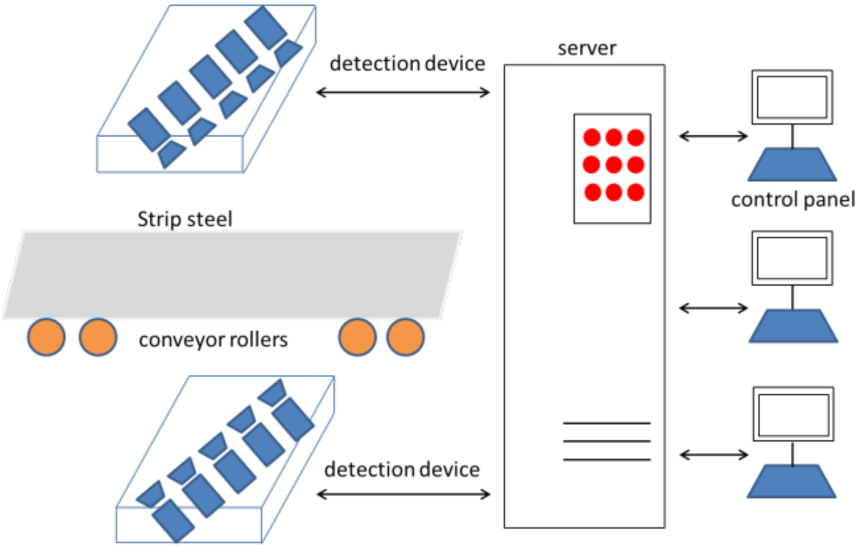


Figure 3. The strip surface defect detection system.

As shown in Figure 3, a schematic diagram of strip surface defect detection [6] is shown. The conveyor rollers rotate and drive the strip through the inspection device at high speed, and the inspection device takes high-speed images of the strip surface. Inspection devices generally include industrial cameras, industrial light sources, protection devices, etc.; images of the strip surface taken by the inspection devices are transmitted to the server, which processes them by the algorithm in the server. The server extracts samples that may be defective and sends them to the quality control personnel at the console for determination while storing them for later inspection. The hardware of the current strip surface defect detection system is sufficient to meet the use of detection, while the algorithm in the server determines the final accuracy of the strip defect classification. Therefore, to address the shortcomings of the existing system mentioned in the previous section, we try to improve the classification accuracy of the interserver algorithm. the contributions of this paper are shown below:

- We combine ResNet50, FcaNet and CBAM to propose a fused network for the classification of surface defects in hot-rolled steel strips.
- We validate the proposed algorithm on the X-SDD dataset [7], compare it with several deep learning models, and design ablation experiments to verify the effectiveness of the algorithm.

2. Related Work

2.1. Machine Learning Based Methods

There are many ways to classify surface defects in strip steel, and scholars have conducted many studies and proposed many schemes in this field. Xiao et al. [8] proposed an evolutionary classifier with Bayesian kernel (BYEC), which can be tuned with a small sample set to better fit the model of a new production line. Firstly, the classifier is designed by introducing rich features to cover the details of the defects. A series of support vector machines (SVMs) are then constructed from a random subspace of features. Finally, the Bayesian classifier is trained as an evolutionary kernel that is fused with the results of the sub-SVMs to form a comprehensive classifier. Gong et al. [9] proposed a novel multiclass classifier, i.e., support vector hyper-spheres with insensitivity to noise (INSVHs), in order to improve the classification accuracy and efficiency of steel plate surface defects. On the one hand, the INSVHs classifier introduces the bouncing sphere loss to reduce its sensitivity to the noise around the decision boundary. On the other hand, the INSVHs classifier reduces the detrimental effect of label noise and enhances the beneficial effect of important samples by increasing the local intra-class sample density weights. Chu et al. [10] proposed a novel support vector machine with adjustable hyper-sphere (AHSVM) focusing on the classification of strip surface defects. Meanwhile, a new multi-class classification method is proposed. AHSVM originates from the support vector data description and employs hyper-spheres to solve the classification problem. AHSVM can follow two principles: marginal maximization and intra-class dispersion minimization. In addition, the hypersphere of AHSVM is tunable, which makes the final classification hypersphere optimal for the training dataset. Luo et al. [11] proposed a generalized completed local binary patterns (GCLBP) framework. Two variants of the improved completion local binary pattern (ICLBP) and the improved completion noise-invariant local structure pattern (ICNLP) are developed under the GCLBP framework for steel surface defect classification. Unlike the traditional local binary pattern variants, descriptive information hidden in non-uniform patterns is innovatively mined for better defect representation. After binarizing the strip surface defect images, Hu et al. [12] combined the defect target images and their corresponding binarized images to extract three types of image features, including geometric features, grayscale features and shape features. For the support vector machine-based classification model, they use Gaussian radial basis as the kernel function, determine the model parameters by cross-validation, and use a one-versus-one approach for multi-class classifiers. Zhang et al. [13] proposed a feature selection method based on a filtering approach combined with an implicit Bayesian classifier to improve the efficiency of defect identification and reduce the complexity of computation. The details of the method are: a large set of image features is initially obtained based on the discrete wavelet transform feature extraction method. Then three feature selection methods (including correlation-based feature selection, consistency subset evaluator [CSE], and information gain) are used to optimize the feature space.

2.2. Deep Learning Based Methods

Although the above traditional machine learning-based schemes are effective to some extent for the classification of strip defects, their effectiveness often relies on feature extraction. The feature extraction-based schemes often require manual operations and expert knowledge, which limits the generality of the algorithms. In recent years, convolutional neural networks (CNN) have gradually received more and more attention from scholars due to their advantages of automatic feature extraction. Fu [14] proposed a compact and effective CNN model that emphasizes the training of low-level features and combines multiple receptive fields for fast and accurate classification of steel surface defects. The solution uses a pre-trained SqueezeNet as the backbone architecture. It requires only a small number of defect-specific training samples to achieve high accuracy recognition on a diversity-enhanced test dataset containing steel surface defects with severe non-uniform illumination, camera noise and motion blur. Liu et al. [15] used GoogLeNet as the base model and added identity mapping to it, which was improved to some extent. The network achieved a measured speed of 125 FPS (Frames Per Second), which fully meets the real-time requirements of the actual steel strip production line. Zhou et al. [16] designed a CNN containing seven layers, including two convolutional layers, two subsampling layers, and two fully connected layers. The experimental results confirm that their proposed method is quite simple, effective and robust for the classification of surface defects in hot rolled steel sheets. Konovalenko I et al. [17] used a deep learning model based on ResNet50 as the base classifier to perform classification experiments on planar images with three types of damage, and the results showed that the model has excellent recognition ability, high speed and accuracy at the same time. Yi et al. [18] proposed an end-to-end surface defect recognition system for steel strip surface inspection. The system is based on a symmetric wrap-around salinity map for surface defect detection and a deep CNN that uses the defect images directly as input and the defect class as output for defect classification. CNNs are trained purely on the original defect images and learn the defect features from the network training, which avoids the separation between feature extraction and image classification, resulting in an end-to-end defect recognition pipeline. Deep learning-based strip defect classification schemes have shown relatively better performance than traditional machine learning schemes, however, the current research has the following shortcomings: Firstly, most of the current studies are based on the NEU surface defect dataset [19], which is balanced among the six categories. However, in the actual field of strip production, the frequency of various types of defects is not the same. Therefore, on the one hand, it is necessary to study on a dataset with unbalanced samples. On the other hand, the attention mechanism has been shown to improve the accuracy of CNN [20–22] for it can make the algorithm focus more attention on the valuable information in the image; while current research rarely introduces the attention mechanism to improve the classification accuracy of strip surface defects.

3. Method

3.1. Introduction of ResNet

As the deep learning-based network evolves, its structure is deepening; while this helps the network to perform more complex feature pattern extraction, it may also introduce the problem of gradient disappearance or gradient explosion. "Gradient disappearance" and "gradient explosion" can lead to the following shortcomings: (1) Long training time but network convergence becomes very difficult or even non-convergent. (2) The network performance will gradually saturate and even begin to degrade, known as the degradation problem of deep networks. To solve such problems, He et al. [23] proposed the ResNet network, which makes it possible to obtain a good performance and efficiency of the network even when the number of network layers is very deep (even over 1000 layers). The deep residual learning framework of ResNet is shown in Figure 4.

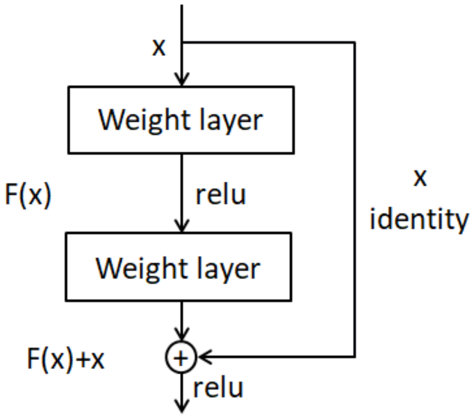


Figure 4. Residual learning: a building block.

As shown in Figure 4, there is an identity mapping in the residual module of ResNet that causes the output of the network to change from F (x) to F (x) + x. The training error of a deep network is generally higher than that of a shallow network. However, adding multiple layers of constant mapping (y = x) to a shallow network turns it into a deep network, and such a deep network can get the same training error as a shallow network. This shows that the layers of constant mapping are better trained. For the residual network, when the residual is 0, the stacking layer only does constant mapping at this time, and according to the above conclusion, theoretically the network performance will not degrade at least.

3.2. *Introduction of CBAM*

Woo et al. [24] proposed the convolutional block attention module (CBAM) in 2018, a simple and effective attention module for feed-forward convolutional neural networks. The significance of attention has been extensively studied in the previous literature [25–28]. Attention not only tells people where to focus their attention, it also improves representation of interest. Representation can be improved by using attentional mechanisms: focusing on important features and suppressing unnecessary ones. The structure of CBAM is shown in Figure 5. The CBAM module has two sequential sub-modules: channel attention model and spatial attention model.

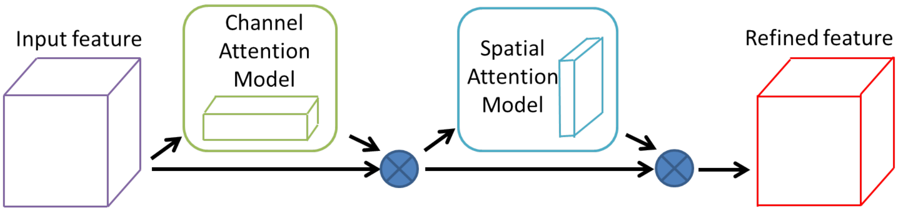


Figure 5. The Convolutional Block Attention Module.

Given an intermediate layer feature map named **F** with dimension $C \times H \times W$ as input, CBAM sequentially generates a 1-dimensional channel attention map (with dimension $\mathbf{Mc} \in C \times 1 \times 1$) and a 2-dimensional spatial attention map (with dimension $\mathbf{Ms} \in 1 \times H \times W$). The overall CBAM attention process can be summarized by the following equation:

$$\mathbf{F}' = \mathbf{Mc}(\mathbf{F}) \otimes \mathbf{F}, \tag{1}$$

$$\mathbf{F}'' = \mathbf{Ms}(\mathbf{F}) \otimes \mathbf{F}', \tag{2}$$

where $\otimes$ represents the one-to-one multiplication of the corresponding elements, and during the multiplication, the attention values are broadcasted (copied) accordingly: the channel attention values are broadcasted along the spatial dimension and vice versa. F$''$ is the output of the final attention weights. The schematic diagrams of the channel attention mechanism and the spatial attention mechanism are shown in Figure 6.

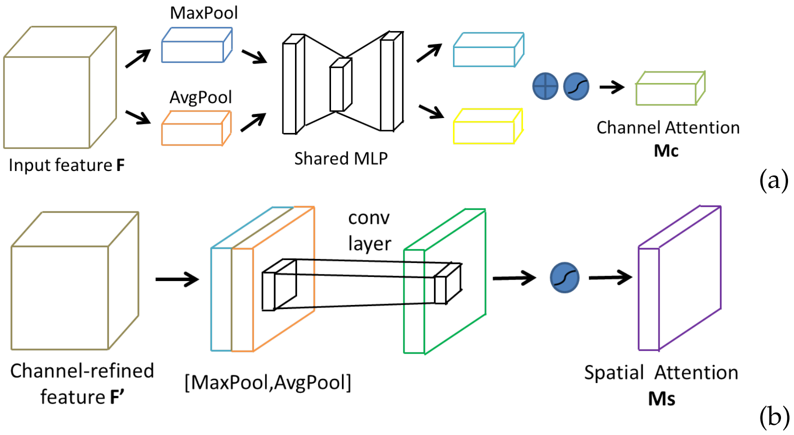


Figure 6. The Convolutional Block Attention Moudle: (**a**) Channel Attention Moudle. (**b**) Spatial Attention Moudle.

3.3. *Introduction of FcaNet*

In general, when calculating the channel attention, each channel will need a learnable scalar value to calculate the attention weight behind the scalar calculation function is generally used Global Average Pooling (GAP). However, GAP is not so perfect, and the simple de-averaging method discards a lot of information and does not fully capture the diversity of each channel. In order to obtain sufficient information about the diversity of each channel, Qin et al. [29] proved that GAP is a special form of discrete cosine transform (DCT), and based on this proof, generalized channel attention to the frequency domain and proposed FcaNet, a channel attention network using multiple frequencies. Assuming that X is the input feature map, the channel attention mechanism can be written as Equation (3) [30,31]:

$$att = sigmoid(fc(gap(X))), \tag{3}$$

where att reprents the attention vector, $sigmoid$ reprents the sigmoid function, fc is the maping functions and gap is GAP. Once this attention vector is obtained, each channel can be scaled by the corresponding elements of this attention vector to obtain the output of the channel attention mechanism:

$$X^*_{:}, i :, := att_i X :, i, :, :, \qquad s.t. \quad i \in 0, 1, \ldots, C-1 \tag{4}$$

where X^* reprents the out of attention mechanism, att_i is the i-th element of attention vector, and $X :, i, :, :$ is the i-th channel of input. The DCT is defined as Equation (5) [32]

$$f_k = \sum_{i=0}^{L-1} x_i cos(\pi k/L(i + 1/2)), \qquad s.t. \quad k \in 0, 1, \ldots, L-1 \tag{5}$$

Here, f is the spectrum of the DCT, x is the input, and L is the length of x. The 2-dimensional DCT can be written as:

$$\begin{aligned} f^{2d}_{h,w} = \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} x^{2d}_{i,j} cos(\pi h/H(i+1/2)) cos(\pi \omega/W(j+1/2)), \\ s.t. \quad i \in 0, 1, \ldots, H-1, j \in 0, 1, \ldots, W-1 \end{aligned} \tag{6}$$

where f is the 2D DCT frequency spectrum, x is the input, H is the heght of x, and W reprents the width of x. The inverse transformation of 2D DCT can be written as:

$$\begin{aligned} x^{2d}_{i,j} = \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} f^{2d}_{h,w} cos(\pi h/H(i+1/2)) cos(\pi w/W(j+1/2)), \\ s.t. \quad i \in 0, 1, \ldots, H-1, j \in 0, 1, \ldots, W-1 \end{aligned} \tag{7}$$

With the definition of channel attention and DCT, we can summarize two points: (1) Existing methods use GAP as preprocessing when doing channel attention. (2) DCT can be viewed as a weighted sum of inputs, and the weights are the cosine part of Equations (6) and (7). For more details, please refer to the reference [29].

3.4. Our Method

In terms of model selection, we choose CNN as the backbone network because the CNN model has the following advantages: The CNN learns local patterns and captures promising semantic information. Moreover, it is also known to be efficient compared to other model types for it has less number of parameters [33,34]. Considering the excellent performance achieved by ResNet50 in the field of strip classification defects, we decided to use it as the backbone network of our method. On this basis, since CBAM, FcaNet attention mechanism can weight the relevant parameters, making the algorithm focus on more and more valuable information; therefore, we add CBAM and FcaNet to improve the performance of the original model. The overall structure diagram of our proposed method is shown in Figure 7.

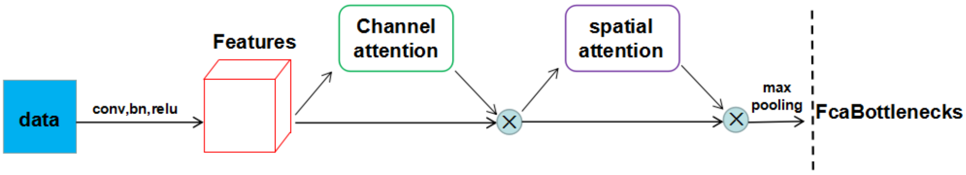


Figure 7. The overall structure of the method in this paper.

We adopt the FcaBottleneck instead of the Bottleneck structure in the original ResNet50 and place the spatial attention mechanism and the channel attention mechanism before the FcaBottleneck. In other words, we adopt CBAM outside the Bottleneck of ResNet for improvement and FcaNet inside the Bottleneck for improvement, so that the original Bottleneck, is converted to FcaBottleneck. The difference between the original Bottleneck in ResNet50 and the FcaBottleneck after the addition of FcaNet is shown in Figure 8.

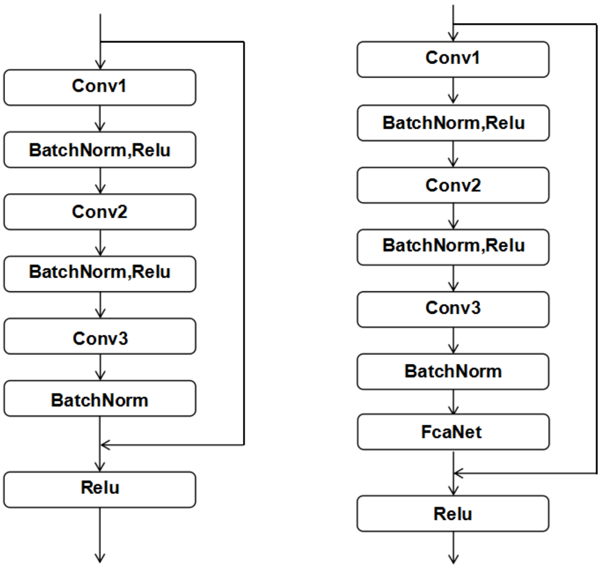


Figure 8. The Bottleneck and FcaBottleneck.

As shown in Figure 8, The flow chart on the left is Bottleneck and the flow chart on the right is FcaBottleneck. We can see the main difference between Bottleneck and FcaBottleneck: FcaBottleneck has an additional layer of FcaNet than Bottleneck. The code details can be found at: https://github.com/Fighter20092392/ResNet50-CBAM-FcaNet (accessed on 5 July 2021).

In contrast to other studies that added attentional mechanisms, we paired two different attentional mechanisms instead of adding only a single one. Moreover, we place CBAM and FcaNet inside and outside of the block, so that the attention mechanism can be fully functional. Whether such an improved scheme will improve the classification accuracy of strip surface defects will be verified by experiments next.

4. Experiments

4.1. Introduction of the Dataset

We choose the newly proposed X-SDD [7] strip surface defect dataset to validate the proposed method in this paper. The X-SDD dataset contains 7 types of 1360 surface defects in hot rolled strip: 238 slag inclusions, 397 red iron sheet, 122 iron sheet ash, 134 surface scratches, 63 oxide scale of plate system, 203 finishing roll printing and 203 oxide scale of temperature system. the size of original images is 128 × 128 pixels with 3 channel JPG format. The defect pattern in this dataset is shown in Figure 9.

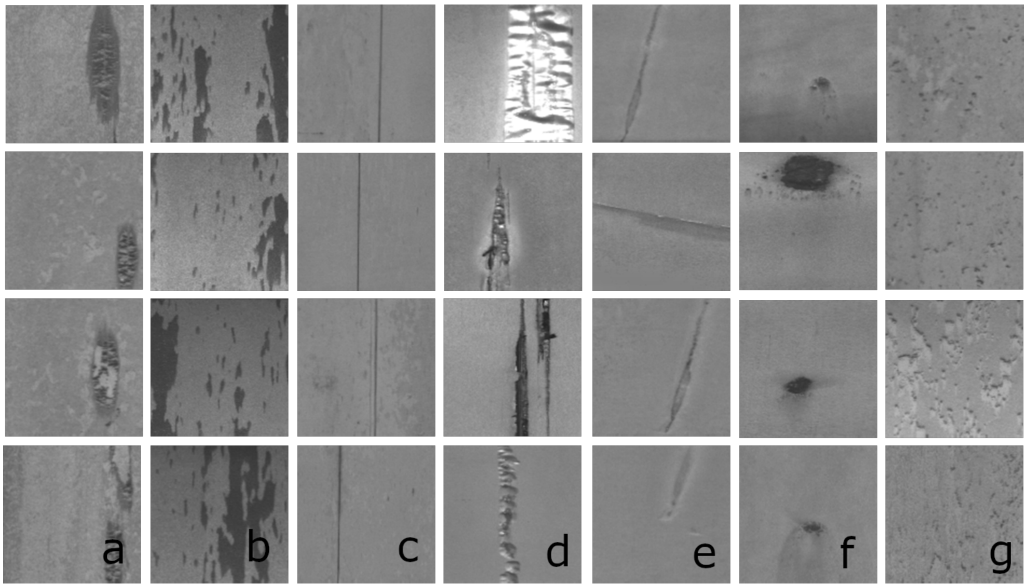


Figure 9. Samples of seven kinds of typical surface on X-SDD. (**a**) oxide scale of plate system. (**b**) red iron sheet. (**c**) surface scratches. (**d**) slag inclusions. (**e**) finishing roll printing. (**f**) iron sheet ash. (**g**) oxide scale of temperature system.

4.2. Experimental Settings

The experiments were conducted under the Win10 operating system and the PyTorch deep learning framework. The hardware configuration for the experiments was a single card NVIDIA RTX3060 GPU, an Intel Core i7-9700 CPU and a 64GB of RAM. In the experiment, the input size is set to 224 × 224 pixels, the batch size is set to 16 (Generally speaking, the batch size value should be set as large as possible within the allowed range of video memory), the learning rate is set to 0.0001 based on experience, the Adam optimizer is used for optimization, and the number of training epochs is 100 (Note that as the batch size increases, the epoch must be increased to force the model to maintain the same accuracy.). We use 70% of the defective images in the X-SDD dataset for the trainset and the remaining 30% of the images for the testset.

4.3. Experimental Results

In order to make the experimental results more convincing, we chose several indicators for comparison, including: Accuary, Macro-Recall, Macro-Precision and Macro-F1. The above indicators are derived as shown in Equations (8)–(12).

$$n\_correct = TP_0 + TP_1 + \ldots + TP_{N-1} \tag{8}$$

$$Accuary = \frac{n\_correct}{n\_total} \tag{9}$$

$$Macro - Recall = (\frac{TP_0}{TP_0 + FN_0} + \frac{TP_1}{TP_1 + FN_1} + \ldots + \frac{TP_{N-1}}{TP_{N-1} + FN_{N-1}}) \times \frac{1}{N} \tag{10}$$

$$Macro - Precision = (\frac{TP_0}{TP_0 + FP_0} + \frac{TP_1}{TP_1 + FP_1} + \ldots + \frac{TP_{N-1}}{TP_{N-1} + FP_{N-1}}) \times \frac{1}{N} \tag{11}$$

$$Macro - F1 = (\frac{2P_0R_0}{P_0 + R_0} + \frac{2P_1R_1}{P_1 + R_1} + \ldots + \frac{2P_{N-1}R_{N-1}}{P_{N-1} + R_{N-1}}) \times \frac{1}{N} \tag{12}$$

where n_total represents the total number of samples in the testset; N is the total number of defect types and in this paper the value of N is 7; $TP_0, TP_1, \ldots, TP_{N-1}$ represents the number of true cases in each category, i.e., the number that classifies the positive cases correctly. We have chosen several deep learning models for comparison: AlexNet [35], MobileNet v3 [36], Xception [37], ShuffleNet [38], EspNet v2 [39], GhostNet [40], VGG16, VGG19 [41], ResNet101 and ResNet152 [23]. The experimental results are shown in Table 1.

Table 1. The experimental results.

Model	Accuary	Macro-Recall	Macro-Precision	Macro-F1
AlexNet	90.69%	82.79%	88.95%	84.21%
MobileNet v3	91.67%	87.95%	91.83%	88.59%
Xception	91.18%	84.30%	90.28%	85.37%
ShuffleNet	89.71%	84.76%	89.44%	84.87%
EspNet v2	86.52%	82.46%	84.10%	81.88%
GhostNet	89.22%	82.99%	87.16%	83.91%
ResNet101	92.40%	86.29%	93.30%	88.02%
ResNet152	89.22%	**89.10%**	87.26%	87.54%
VGG16	89.71%	86.64%	88.68%	87.47%
VGG19	86.52%	86.06%	88.98%	86.86%
RegVGG B1g2	88.48%	80.33%	92.54%	81.34%
Our Method	**93.87%**	87.33%	**94.35%**	**88.71%**

As shown in Table 1, our method achieves better than other compared models in terms of Accuracy, Macro-Precision and Macro-F1. Among them, our method achieved 93.87% in Accuracy, which is 1.47 percentage points igher than the second place ResNet101. Our method achieved the third place in the Macro-Recall metric by 1.77 percentage points lower than ResNet152 and 0.62 percentage points lower than MobileNet v3. One possible reason for the low Recall metric of our method is that Accuracy and Recall tend to affect each other, and our method focuses on improving Accuracy at the expense of Recall to some extent. Nevertheless, considering that our method is better than ResNet152 and MobileNet v3 in other metrics; therefore, our method has an advantage over ResNet152 as well as MobileNet v3.

4.4. Ablation Experiments

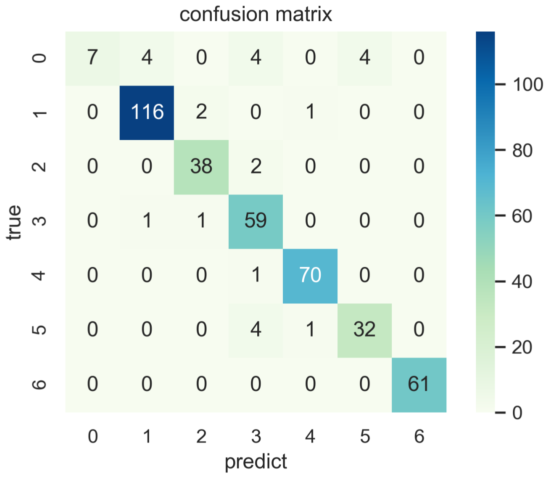
In order to verify the improvement of our method over the original ResNet50, the following ablation experiment is designed to analyze the effect. We compare the scheme proposed in this paper with ResNet50 and ResNet50+CBAM to analyze the effectiveness of our improved scheme. The results of the ablation experiments are shown in Table 2.

Table 2. The results of the ablation experiments.

Model	Accuary	Macro-Recall	Macro-Precision	Macro-F1
ResNet50	92.40%	86.45%	94.08%	88.32%
ResNet50+CBAM	92.65%	**88.62%**	91.40%	**89.71%**
ResNet50+CBAM+FcaNet	**93.87%**	87.33%	**94.35%**	88.71%

As can be seen in Table 2, the improved scheme of ResNet50+CBAM compared to ResNet50 has some improvement in Accuary, Macro-Recall and Macro-F1. This shows that the CBAM attention mechanism makes the algorithm pay more attention to the valuable information of images in spatial and channels, which in turn improves the classification ability of the algorithm. The only shortcoming is that the ResNet50+CBAM model is 2.68 percentage points lower than the ResNet50 model in the Macro-Precision metric. In contrast, our proposed ResNet50+CBAM+FcaNet scheme achieves higher scores than the ResNet50 model in all four metrics, which indicates that our proposed approach is more effective in improving the results. From a practical point of view, the most important of the four metrics is the Accuracy metric, and the scheme in this paper achieves the highest Accuracy. This indicates that our proposed ResNet50+CBAM+FcaNet method has more practical application value.

The confusion matrix of our proposed method is shown in Figure 10. The horizontal and vertical coordinates of 0–6 in the figure represent the oxide scale of plate system, red iron sheet, surface scratches, slag inclusions, finishing roll printing, iron sheet ash and oxide scale of temperature system, respectively. As can be seen from the confusion matrix, our method can classify most defect categories very accurately, with less accuracy only in the case of oxide scale of temperature system. Our model has 7 correct classifications and 12 incorrect classifications for oxide scale of plate system, with a correct classification rate of only 36.84% for this type of defect. The reason for this result may be that the amount of data on oxide scale of temperature system is small and the algorithm fails to learn effectively for this type of defect. A possible solution to this problem is to perform more data augmentation for this class of defects, using multiple models for cascading or ensemble learning. In the next part of this paper, we will try to solve the problem by using an ensemble learning approach.



true \ predict	0	1	2	3	4	5	6
0	7	4	0	4	0	4	0
1	0	116	2	0	1	0	0
2	0	0	38	2	0	0	0
3	0	1	1	59	0	0	0
4	0	0	0	1	70	0	0
5	0	0	0	4	1	32	0
6	0	0	0	0	0	0	61

Figure 10. The confusion matrix.

4.5. Comparison of Model Complexity

The comparison results of the number of parameters and computation of the model are shown in Table 3. As can be seen from Table 3, the proposed method in this paper is basically the same in terms of number of parameters and computational effort compared with the original ResNet50. This shows that the improvement in the effect of our proposed method does not come from an increase in the number of participants but from a more rational structure. Compared with heavyweight deep learning models such as ResNet101, ResNet152 and VGG16, our method has the advantage of smaller number of parameters and computational complexity. The computational and parametric quantities of our method are only 35.60% and 44.77% of those of ResNet152, respectively. As can be seen from Table 1, ResNet152 has some advantages over our method in terms of recall, but our method is much better than ResNet152 in terms of the number of parameters and computational complexity. Compared to lightweight deep learning models such as EspNet v2 with 0.092 G of computation and 0.638 M of parameters, our method requires more hardware resources. In the future, model pruning, quantization, and knowledge distillation can be used to reduce the computational effort and number of parameters of the model, making it easier to deploy.

Table 3. The comparision of model complexity.

Model	Flops (G)	Params (M)
AlexNet	0.309	14.596
MobileNet v3	0.300	4.317
Xception	4.617	20.822
ShuffleNet	0.132	0.860
EspNet v2	0.092	0.638
GhostNet	0.213	3.127
ResNet50	4.109	23.522
ResNet101	7.832	42.515
ResNet152	11.557	58.158
VGG16	15.484	138.358
VGG19	19.647	143.667
RepVGG B1g2	9.815	43.748
ResNet50+CBAM	4.111	23.523
Our Method	4.114	26.038

4.6. The Ensemble Model

We use three models for integration, the sub-models are ResNet50+CBAM+FcaNet, VGG16 and SqueezeNet. The three sub-models were chosen because they differ in principle and meet the need for diversity in ensemble learning. We set the weights of ResNet50+CBAM+FcaNet, VGG16 and SqueezeNet to 1.2, 0.9, 0.9 respectively. The weights of the models are not all set equal; this is to facilitate the final choice of the integrated model when all three sub-models have different output values. The output of the ensemble model is shown in Figure 11.

As can be seen in Figure 11, the number of correctly classified oxide scale of plate system defects is 11, and the number of incorrectly classified defects is 8. The classification accuracy of this category of defects is 57.89%, which is 21.05% higher than the 36.84% of the ResNet50+CBAM+FcaNet model. The results of the ensemble model on each metric are shown in Table 4.

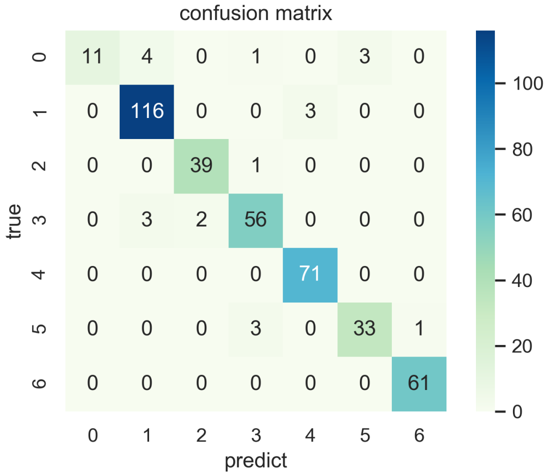


Figure 11. The output of the ensemble model.

Table 4. The effect of ensemble model.

Model	Accuary	Macro-Recall	Macro-Precsion	Macro-F1
The ensemble model	94.85%	90.71%	95.04%	92.06%

Comparing the results in Table 4 with those in Tables 1 and 2 shows that the ensemble model outperforms all comparison models in all four metrics. The ensemble model achieves a good score of over 90% on all indicators, which indicates that the improved model is more balanced on all indicators. In summary, it is effective to improve the model using ensemble learning.

5. Discussion and Conclusions

In this paper, we propose a ResNet50+CBAM+FcaNet model for the problem of classifying surface defects in hot-rolled strip steel. After validation on the newly proposed X-SDD dataset, our proposed algorithm achieves 93.87% accuracy on the testset, which is better than more than ten other comparative algorithms. In addition, our method still achieves better results relative to other comparison models on Macro-Precision, Macro-F1, and third place on Macro-Recall. The above results show the effectiveness of the algorithm proposed in this paper. Combining CBAM with FcaNet helps to improve the accuracy of ResNet50, and we argue that this improved approach will also be applicable to other models. In the next work, we may verify through more experiments which combination of this scheme and which model will achieve optimal results. Although our previous paper [7] showed that a RepVGG based scheme with an added attention mechanism may be superior in terms of effectiveness; according to Table 3, the ResNet50 based approach has an advantage over RepVGG in terms of number of parameters and computational complexity, i.e., it is easier to deploy in practice.

In order to further confirm the effectiveness of the attention mechanism, an ablation experiment is designed to verify. The ablation experiment verifies that adding the attention mechanism can effectively improve the classification accuracy of the algorithm, but while the overall accuracy is improved, the classification accuracy of individual categories may be reduced, which in turn affects the overall Recall and F1 metrics. Since the number of categories in the X-SDD dataset we use is unbalanced among categories, our approach will favor improving the accuracy of the categories with larger sample sizes at the expense of the accuracy of the categories with smaller sample sizes. In the case of a category with a smaller sample size, the accuracy of the category may be significantly reduced, as well as the overall Recall being significantly affected, simply because a few more samples are misclassified than before. The analysis of Figure 10 shows that the main factor affecting the Macro-Recall of our method is the low accuracy of the classification of oxide scale of plate system.

To solve the low classification accuracy of ResNet50+CBAM+FcaNet model on the oxide scale of plate system, we improve the original scheme. We introduce the concept of ensemble learning by integrating the original ResNet50+CBAM+FcaNet with VGG16 and SqueezeNet. We believe that the integration of multiple models can alleviate the problem of low classification level of a single model on a particular category to some extent, because the focus of different models may be different. In the selection of the ensemble sub-models, we fully consider the diversity of sub-models; the final selection of sub-models covers three models with different characteristics, such as with and without attention mechanism, heavy weight network and lightweight network. The final experimental results show that the ensemble model is optimal in all four indicators, and the classification accuracy of oxide scale of plate system has been improved substantially.

Although our proposed ResNet50+CBAM+FcaNet model and the improved ensemble model both achieve good results, there are still some areas that can be improved. Firstly, there is still some room for further improvement in the effectiveness of the model for the category imbalance problem. In this paper, the model integration is carried out in a weighting way, while other ensemble methods such as probabilistic summation can also be considered. In addition, modifying the loss function may also improve the classification accuracy for classes with small sample sizes. Secondly, we combine two attention mechanisms-CBAM and FcaNet with ResNet50, while more attention mechanisms can be considered for combination. Modification of the existing attention mechanism or proposing a new attention mechanism based on the characteristics of the steel strip surface defects may also yield good results.

After classifying the surface defects of hot rolled strip, different treatments are often required depending on the severity of the defects. Therefore, in the future, we will compile a dataset of the degree of surface defects of hot rolled steel strip and design an algorithm to classify the degree of surface defects of hot rolled steel strip. We may introduce the newly proposed MLP-mixer [42] algorithm into the field of strip defects and improve the original algorithm to make it more suitable for the context of strip defect classification.

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

Funding: This research was funded by National Science Foundation under Grant 61573087, 61573088, 62173072 and 62173073.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work is done when Xinglong Feng was an intern at xBang Inc., Shenyang, China. Thanks their support for this work.

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

References

1. Kumar, A.; Das, A.K. Evolution of microstructure and mechanical properties of Co-SiC tungsten inert gas cladded coating on 304 stainless steel. *Eng. Sci. Technol. Int. J.* **2020**, *24*, 591–604. [CrossRef]
2. Afanasieva, L.E.; Ratkevich, G.V.; Ivanova, A.I.; Novoselova, M.V.; Zorenko, D.A. On the Surface Micromorphology and Structure of Stainless Steel Obtained via Selective Laser Melting. *J. Surf. Investig. X-ray Synchrotron Neutron Tech.* **2018**, *12*, 1082–1087. [CrossRef]
3. Gromov, V.E.; Gorbunov, S.V.; Ivanov, Y.F.; Vorobiev, S.V.; Konovalov, S.V. Formation of surface gradient structural-phase states under electron-beam treatment of stainless steel. *J. Surf. Investig. X-ray Synchrotron Neutron Tech.* **2011**, *5*, 974–978. [CrossRef]
4. Youkachen, S.; Ruchanurucks, M.; Phatrapomnant, T.; Kaneko, H. Defect Segmentation of Hot-rolled Steel Strip Surface by using Convolutional Auto-Encoder and Conventional Image processing. In Proceedings of the 2019 10th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES), Bangkok, Thailand, 25–27 March 2019; pp. 1–5. [CrossRef]
5. Ashour, M.W.; Khalid, F.; Halin, A.A.; Abdullah, L.N.; Darwish, S.H. Surface defects classification of hot-rolled steel strips using multi-directional shearlet features. *Arab. J. Sci. Eng.* **2019**, *44*, 2925–2932. [CrossRef]
6. Luo, Q.; Fang, X.; Sun, Y.; Liu, L.; Ai, J.; Yang, C.; Simpson, O. Surface Defect Classification for Hot-Rolled Steel Strips by Selectively Dominant Local Binary Patterns. *IEEE Access* **2019**, *7*, 23488–23499. [CrossRef]
7. Feng, X.; Gao, X.; Luo, L. X-SDD: A New Benchmark for Hot Rolled Steel Strip Surface Defects Detection. *Symmetry* **2021**, *13*, 706. [CrossRef]
8. Xiao, M.; Jiang, M.; Li, G.; Xie, L.; Yi, L. An evolutionary classifier for steel surface defects with small sample set. *EURASIP J. Image Video Process.* **2017**, *2017*, 1–13. [CrossRef]
9. Gong, R.; Chu, M.; Yang, Y.; Feng, Y. A multi-class classifier based on support vector hyper-spheres for steel plate surface defects. *Chemom. Intell. Lab. Syst.* **2019**, *188*, 70–78. [CrossRef]
10. Chu, M.; Liu, X.; Gong, R.; Zhao, J. Multi-class classification method for strip steel surface defects based on support vector machine with adjustable hyper-sphere. *J. Iron Steel Res. Int.* **2018**, *25*, 706–716. [CrossRef]
11. Luo, Q.; Sun, Y.; Li, P.; Simpson, O.; Tian, L.; He, Y. Generalized completed local binary patterns for time-efficient steel surface defect classification. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 667–679. [CrossRef]
12. 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]
13. Zhang, Z.F.; Liu, W.; Ostrosi, E.; Tian, Y.; Yi, J. Steel strip surface inspection through the combination of feature selection and multiclass classifiers. *Eng. Comput.* **2020**, *38*, 1831–1850. [CrossRef]
14. Fu, G.; Sun, P.; Zhu, W.; Yang, J.; Cao, Y.; Yang, M.Y.; Cao, Y. A deep-learning-based approach for fast and robust steel surface defects classification. *Opt. Lasers Eng.* **2019**, *121*, 397–405. [CrossRef]

15. Liu, Y.; Geng, J.; Su, Z.; Yin, Y. Real-time classification of steel strip surface defects based on deep CNNs. In *Proceedings of 2018 Chinese Intelligent Systems Conference*; Springer: Singapore, 2019; pp. 257–266.
16. Zhou, S.; Chen, Y.; Zhang, D.; Xie, J.; Zhou, Y. Classification of surface defects on steel sheet using convolutional neural networks. *Mater. Technol.* **2017**, *51*, 123–131.
17. Konovalenko, I.; Maruschak, P.; Brezinová, J.; Viňáš, J.; Brezina, J. Steel Surface Defect Classification Using Deep Residual Neural Network. *Metals* **2020**, *10*, 846. [CrossRef]
18. Yi, L.; Li, G.; Jiang, M. An end to end steel strip surface defects recognition system based on convolutional neural networks. *Steel Res. Int.* **2017**, *88*, 1600068. [CrossRef]
19. Song, K.; Yan, Y. Micro Surface defect detection method for silicon steel strip based on saliency convex active contour model. *Math. Probl. Eng.* **2013**, *2013*, 429094. [CrossRef]
20. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6450–6458.
21. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
22. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019.
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. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
25. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
26. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the International Conference on Machine Learning (PMLR), Lille, France, 7–9 July 2015; pp. 2048–2057.
27. Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D.; Wierstra, D. Draw: A recurrent neural network for image generation. In Proceedings of the International Conference on Machine Learning (PMLR), Lille, France, 7–9 July 2015; pp. 1462–1471.
28. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial transformer networks. *arXiv* **2015**, arXiv:1506.02025.
29. Qin, Z.; Zhang, P.; Wu, F.; Li, X. FcaNet: Frequency Channel Attention Networks. *arXiv* **2020**, arXiv:2012.11879.
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, 18–23 June 2018; pp. 7132–7141.
31. Qilong, W.; Banggu, W.; Pengfei, Z.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. *arXiv* **2020**, arXiv:1910.03151.
32. Ahmed, N.; Natarajan, T.; Rao, K.R. Discrete cosine transform. *IEEE Trans. Comput.* **1974**, *100*, 90–93. [CrossRef]
33. Jeon, M.; Jeong, Y.S. Compact and accurate scene text detector. *Appl. Sci.* **2020**, *10*, 2096. [CrossRef]
34. Vu, T.; Van Nguyen, C.; Pham, T.X.; Luu, T.M.; Yoo, C.D. Fast and efficient image quality enhancement via desubpixel convolutional neural networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Glasgow, UK, 23–28 August 2018; pp. 243–259.
35. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
36. Howard, A.; Sandler, M.; Chu, G.; Chen, L.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 1314–1324.
37. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
38. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6848–6856.
39. Mehta, S.; Rastegari, M.; Shapiro, L.; Hajishirzi, H. Espnetv2: A light-weight, power efficient, and general purpose convolutional neural network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seoul, Korea, 27–28 October 2019; pp. 9190–9200.
40. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 1580–1589.
41. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
42. Tolstikhin, I.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J.; et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv* **2021**, arXiv:2105.01601.