



Article

An Efficient Network for Surface Defect Detection

Zesheng Lin 10, Hongxia Ye 1,*0, Bin Zhan 2 and Xiaofeng Huang 3

- Key Laboratory for Information Science of Electromagnetic Waves (MoE), Fudan University, Shanghai 200433, China; zslin18@fudan.edu.cn
- ² NVIDIA Technology Shanghai Co., Ltd., Shanghai 201210, China; bzhan@nvidia.com
- School of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China; xfhuang@hdu.edu.cn
- * Correspondence: yehongxia@fudan.edu.cn; Tel.: +86-21-31242590

Received: 22 July 2020; Accepted: 27 August 2020; Published: 2 September 2020



Abstract: Convolutional neural networks (CNN) have achieved promising performance in surface defect detection recently. Although many CNN-based methods have been proposed, most of them are limited by the few samples available for training, and the imbalance of positive and negative samples. Hence, their detection performance needs to be further improved. To this end, we propose a multi-scale cascade CNN called MobileNet-v2-dense to detect defects more efficiently. Specifically, the multi-scale cascade structure used in our network can help capture the weak defect semantics that may be lost in the deep network. Then, we propose a novel asymmetric loss function to further improve detection performance. Lastly, a two-stage augmentation method effectively enlarges the training dataset. Experimental results show that, compared to the state-of-the-art, the area under the receiver-operating characteristic curve (AUC-ROC) score of our method increased by 0.16.

Keywords: convolution neural network; data augmentation; multi-scale cascade connection; surface defect detection

1. Introduction

Surface defect detection aims to recognize samples that exhibit dissimilar properties when compared with defect-free samples. Surface defect detection commonly occurs in many industrial applications, such as steel defect detection [1], textile defect detection [2], and glass bottle bottom defect detection [3]. The defects of industrial products that occur during manufacturing have negative effects on the product quality and even functionality in certain cases. To guarantee the quality of final products, the defective products must be detected and removed manually or automatically.

In the past decades, many traditional methods of surface defect detection have been developed such as statistics-based methods [4–7], spectrum-based methods [8–10], and model-based methods [11–13]. These traditional detection algorithms rely on the features of artificial design, which are limited to specific defect modes. Therefore, they cannot handle detection of texture defects that involve complex textures or any new defect mode.

Recently, Convolutional Neural Networks (CNN) have been widely applied in industrial defect detection task and achieved promising performance. Jonathan et al. [14] applied CNN for the first time to detection of steel surface defects. Later, more and more methods based on CNN have been proposed. The CNN-based methods have reached better defect detection results than traditional algorithms through the design of efficient network structure. However, their network structures were streamlined which cannot extract the information of the input image completely. In addition, although they used some data augmentation methods to increase the number of training samples, these methods are too simple to significantly improve the detection performance. Most importantly, they did not consider the imbalance of positive and negative samples. Generally speaking, the number

of negative samples (defect-free samples) is much larger than positive samples (defective samples). In this case, the network learns more features of negative samples, resulting in a decrease in the detection accuracy of positive samples.

To tackle these problems, this paper makes the following contributions:

- 1. an efficient network called MobileNet-v2-dense is proposed. It starts from MobileNet-v2 and innovates on a multi-scale cascade structure to improve defect detection performance;
- 2. a two-stage data augmentation method is proposed for network training. This method can effectively increase the number of data sets and improve detection performance;
- 3. to solve the imbalance between positive and negative samples, an asymmetric loss function is proposed, which can make the network pay more attention to the loss of positive samples.

The rest of the paper is organized as follows. Section 2 reviews some related work on surface defect detection. Section 3 describes the proposed MobileNet-v2-dense network in detail, including the data augmentation method and asymmetric loss function. To evaluate the model and compare the overall performance, the experimental results are presented in Section 4. Section 5 summarizes the work.

2. Related Work

Traditional Surface Defect Detection Methods. The traditional methods for surface defect detection can be categorized into three classes according to the image processing techniques. (a) Statistics-based methods; (b) spectrum-based methods; (c) model-based methods. Statistics-based methods commonly use the grayscale distributions of image regions to describe texture characteristics, such as the gray-level co-occurrence matrix method [4], the autocorrelation method [5], the morphology method [6], and the histogram feature statistics [7]. Spectrum-based methods focus on finding the textural structure of the texture image and are particularly suitable for textures with an obvious structure, such as the Fourier feature method [8], Gabor feature method [9], and wavelet feature method [10]. Model-based methods describe texture patterns by modeling special distributions or other attributes with certain models, such as the fractal body model [11], random field model [12], and back scattering model [13], and so on.

CNN-based Surface Defect Detection Methods. Many CNN-based methods have emerged in recent years. Jonathan et al. [14] applied CNN for the first time to detection of steel surface defects and proved that the method of CNN was superior to the traditional SVM classifier. Soukup et al. [15] gathered the image of the rail surface and experimented with classical CNN, and the result showed that classical CNN already distinctly outperforms the model-based methods. Faghih et al. [16] used deeper convolutional neural networks (DCNNs) to detect railway surface defects. Ren et al. [17] proposed a supervised CNN architecture for image patch classification. Weimer et al. [18] and Park et al. [19] also proposed novel DCNNs architectures that obviously improved the performance of automated defect detection. Wang et al. [20] cut defect images into patches for detection. Benjamin et al. [21] proposed deep metric learning using triplet networks for defect detection.

Multi-scale Cascade Network. As CNN becomes deeper, a new research problem emerges: after the information of input images passes through many layers, it can vanish when it reaches the end of the network. Highway Networks [22] bypasses the signal from one layer to the next layer via identity connections. FractalNet [23] repeatedly combines several parallel layer sequences with a different number of convolutional blocks to obtain a large nominal depth while maintaining many short paths in the network. Although these different approaches vary in network topology, they all have common characteristics: they create short paths from the early layers to later layers. In [24–27], the use of multi-scale features in CNNs through skip connections has been found to be effective for various vision tasks. However, in the task of using CNN for defect detection [15–19], the network structure of CNN is usually streamlined, and does not use shallow network information.

MobileNet-v2. To extract richer features, current convolutional neural networks tend to increase the width and depth of the network. These methods have resulted in large network parameters and

Appl. Sci. 2020, 10, 6085 3 of 12

are difficult to apply on hardware devices with limited resources. MobileNet [28] was proposed to solve this problem. MobileNet-v1 [28] replaces standard convolution with depth separable convolution and divides the traditional convolution process into two steps: filtering and merging. This convolution approach greatly reduces the computational cost and model size and is easier to deploy in resource-constrained environments. The design of MobileNet-v2 [29] is based on MobileNet-v1 and borrows ideas from ResNet [30] to add the residual structure of the shortcut.

3. Proposed Methods

In this section, the proposed MobileNet-v2-dense model is described in detail. The overall architecture of the model is shown in Figure 1. In the training phase, training data are augmented with a two-stage augmentation method. Furthermore, the improved loss function, called asymmetric loss function, is introduced. In the testing phase, the learning-completed model is used to detect whether an image is defective or not. Specific illustrations are given as follows.

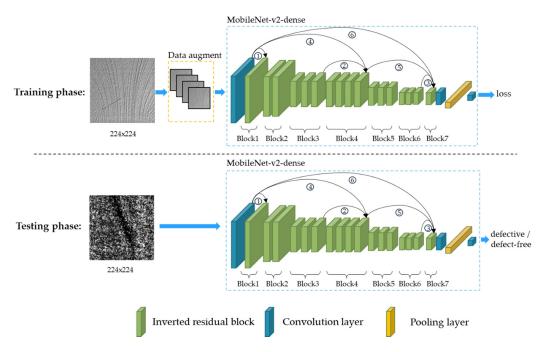


Figure 1. The structure of the MobileNet-v2-dense network. Where the block names correspond to Table 1. Solid lines in a diagram are concatenation from a shallow feature map to the deep one.

3.1. A Multi-Scale MobileNet-v2-Dense Network

Considering the limited storage and computing power of the hardware in practical applications, we chose a network with fewer parameters called MobileNet-v2 as our backbone.

As stated above, the transmission of multi-scale information can prevent the feature vanish when the network goes deeper. Therefore, we incorporated multi-scale design into MobileNet-v2 and designed our defect detection network MobileNet-v2-dense. The network structure is shown in Figure 1.

The main component of the network is composed of multiple cascaded inverted residual blocks with downsampling. The structure of the inverted residual block is shown in Figure 2. The inverted residual block takes a low-dimensional compressed representation as an input, which first expanded to high dimension and filtered with lightweight depthwise convolution. Features are subsequently projected back to a low-dimensional representation. The dense concatenation is used between inverted residual blocks to carry shallow features to deeper layers of the network, which is the fusion of multi-scale information. The network structure of MobileNet-v2-dense is formed by dense cascades

Appl. Sci. 2020, 10, 6085 4 of 12

between multi-scale channels, with six cascades. ① ②, and ③ are cascades of feature maps with the same scale, ④ and ⑤ are cascades of feature maps with 1/4 down sampling, and ⑥ is a cascade of feature maps with 1/16 down sampling.

Table 1.	Mobilenet-v2-Dense	configuration.	("Conv"	and	"Concat"	denote	convolution	and
concatena	ntion, respectively).							

Layers	Output Size	Operation
Conv1	112 × 112 × 24	Conv 3 × 3, BN, LRelu
Block1	112 × 112 × 16	Inverted Residual Block \times 1
C1	$112 \times 112 \times 40$	Concat (Conv1, Block1)
Block2	$56 \times 56 \times 40$	Inverted Residual Block \times 2
Block3	$28 \times 28 \times 48$	Inverted Residual Block × 3
Block4	$28 \times 28 \times 56$	Inverted Residual Block \times 4
C2	$28 \times 28 \times 124$	Concat (C1/4, Block3, Block4)
Block5	14 × 14 × 96	Inverted Residual Block × 3
Block6	$7 \times 7 \times 120$	Inverted Residual Block × 3
Block7	$7 \times 7 \times 160$	Inverted Residual Block \times 1
C3	$7 \times 7 \times 320$	Concat (C1/16, C2/4, Block6, Block7)
Conv2	$7 \times 7 \times 1280$	Conv 1 × 1, BN, LRelu
Pooling	$1 \times 1 \times 1280$	Global Pooling
Conv3	$1 \times 1 \times 2$	Conv 1 × 1

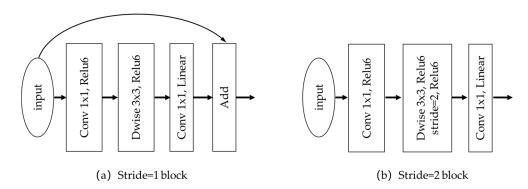


Figure 2. The inverted residual block in MobileNet-v2. It has two structures, and when the stride = 2, the size of the feature map is downsampled by two times. Dwise is a deep-wise separable convolution proposed by MobileNet.

The configuration of MobilNet-v2-dense is shown in Table 1. BN indicates batch normalization layer, and LRelu denotes leaky relu layer. All the activation functions of the network are leaky_relu. Additionally, C1/4 means 4 times downsampling of the C1 layer. Inverted Residual Block x2 states there are two inverted residual blocks in Block2. The input image size of the network is 224×224 , and the resolution of the feature image changes to 112×112 after Conv1. Subsequently, features are extracted through several blocks. The "Concat layer" concatenate the features of the shallow layer. Finally, after aggregation by a Pooling layer, global features are fed into a softmax layer to obtain the final probability scores, which indicate the possibility that the image is a defect.

The network topology can transfer and fuse multi-scale features via shortcuts from the shallow to deep layers to avoid the feature missing of selected samples in the streamlined network. The experimental results in Section 4 show that the structure improves the detection accuracy.

Appl. Sci. 2020, 10, 6085 5 of 12

3.2. Two-Stage Augmentation Method

In this section, we introduce the two-stage data augmentation method, which can effectively increase the number of data sets and improve detection performance.

Currently, training of CNNs often uses static augmentation [28] or dynamic augmentation [31] to expand the dataset. Static augmentation refers to the dataset is augmented with image processing methods before feeding into the convolutional neural network. Dynamic augmentation is that the original data remains the same and augmentation operations are performed per mini-batch during the training process. However, the static augmentation pattern typically increases the samples exponentially, thus the training time is also increased exponentially. In the experiment, we found that for a small number of data sets, only use dynamic augmentation is not helpful to improve the detection performance.

Therefore, this paper proposes a two-stage augmentation method for data augmentation during the training process via static augmentation and dynamic augmentation. Barret et al. [31] indicated that the operation most commonly used in good strategies they search out is rotate. Therefore, under the premise of ensuring regular texture of the defect image, static augmentation of first stage only performs five types of operation: horizontal flip, vertical flip, 90° rotation, 180° rotation, and 270° rotation. These operations preserve the global texture of the original image. The statically expanded dataset is subsequently used to train the neural network. When the loss value stops decreasing and begins to jitter, the dynamic augmentation process of second stage is applied to further enhance the statically augmentation data set during the training process. Dynamic augmentation operations are performed per mini-batch. The operations modes are listed in Table 2, and these operations occur with a certain probability for each batch. The probability of enhanced operation is determined by our prior knowledge. We think that first stage augmentation data is always more important, and the second stage augmentation method may damage data original texture structure and lead to model degradation, so the probability of enhanced operation is set below 30%. The intensity of the occurrence is random within a certain range, and the augmentation types can occur in combination. The one-stage augmentation is only trained once, and the training data is dynamically augmented based on static augmentation during training.

Enhanced OperationProbabilityIntensityHorizontal (or vertical) zoom0.10-10% PixelAdd Gaussian noise0.1Variance 0.005-0.02rotation0.20-15°Horizontal (or vertical) translation0.30-10% Pixel

Table 2. Second stage augmentation operation.

The proposed augmentation method helps to expand the dataset more efficiently, reduce the training time, and improve the convergence of the neural network training and the robustness of the trained model. This method can also be used to fine-tune the network and improve the accuracy further.

3.3. Asymmetric Loss Function

Class imbalance is common in detection problems. To solve this problem, we propose an asymmetric loss function. This section will introduce the loss in detail.

In the field of defect detection, the number of positive samples (defective images) is usually smaller than the number of negative samples (defect-free images). This imbalance will cause two problems: (1) the training efficiency is low because negative samples take up a large portion of time during the training process; and (2) the training performance is poor because the network learns features from a large number of negative samples instead of the effective defect features, which leads to model degradation.

Appl. Sci. 2020, 10, 6085 6 of 12

A common solution is to perform a form of hard negative mining [32] or to give different weights to different classes during training [33]. In contrast, certain scholars attempted to solve the problem by designing special loss functions to allow efficient training on all examples without sampling or weights. The most common loss function used for binary classification is cross-entropy (CE), which expression as follows:

$$L_{CE} = \begin{cases} -\log(y') & if \ y = 1\\ -\log(1 - y') & if \ y = 0 \end{cases}$$
 (1)

where y' is the prediction and y is the expected value. y = 1 is a positive sample, y = 0 is a negative sample. However, whether the ground truth of the sample is defective or not, the attenuation degree on CE loss is the same.

The focal loss function [34] modified the CE-loss function to reduce the relative loss of well-classified samples and placed more attention on bad-classified samples. A remarkable characteristic of both CE-loss and focal loss is that although the loss of defective samples is large, since the number of defective samples is less, network training does not focus on defective samples. Thus, we redesigned the loss to give more attention to the loss value of the defective sample by using the exponential function as the attention mechanism. It is written as:

$$L_{asy} = \begin{cases} -e^{y'} \log(y') & \text{if } y = 1\\ -e^{-y'} \log(1 - y') & \text{if } y = 0 \end{cases}$$
 (2)

Figure 3 compares three kinds of loss functions. Intuitively, they are symmetric in the center of (a) and (b), which means that no matter the positive sample or the negative sample, their loss value is the same. Thereby, it pays more attention to the samples of minority class, alleviating the imbalance of positive and negative samples.

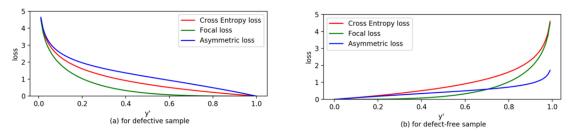


Figure 3. Comparison between different loss functions.

4. Experiments and Discussion

4.1. Experimental and Algorithm Setup

In order to verify the effectiveness of our proposed multi-scale cascaded network structure, asymmetric loss, and two-stage data augmentation, we use the network structure of Figure 1 to train and test on the DAGM data set [35], and compare it with the other three advanced defect detection algorithms.

All experiments were trained on a single NVIDIA GeForce GTX 1060 6 GB graphics card. We use the Adam optimizer [36] with a mini-batch size of 8. While the network is training, the training set is processed by our two-stage augmentation method. Network is trained to minimize asymmetric loss. The learning rate (LR) is set to 1e-4 and is decayed by a factor 10 when the loss does not decrease within 10 epochs.

4.2. Evaluation Indicators

TPR and TNR. True positive rate (TPR) means that the classifier predicts the proportion of correct positive samples to all positive samples. Similarly, true negative rate (TNR) represents the proportion

Appl. Sci. 2020, 10, 6085 7 of 12

of correct negative samples predicted by the classifier to all negative samples. TPR and TNR are defined as Equations (3) and (4),

$$TPR = TP/(TP + FN), (3)$$

$$TNR = TN/(FP + TN)$$
 (4)

where TN refers to the defect-free samples that are identified as defect free, FP denotes those that are identified as defective, TP refers to the defective samples that are identified as defective, and FN represents those that are identified as defect-free. TPR and TNR are values between 0 and 1, higher is better.

AUC-ROC. AUC-ROC is the area under the receiver-operating characteristic (ROC) curve. ROC is an important evaluation metric for assessing the performance of classification models. The ROC curve demonstrates the relationship between false positive rate (FPR) and TPR. FPR is defined as:

$$FPR = FP/(FP + TN)$$
 (5)

where FPR represents the proportion of false positive samples predicted by the classifier to all negative samples. A notable advantage of the ROC curve is that when the distribution of positive and negative samples changes, its shape can remain generally unchanged. In other words, in the case of an unbalanced number of positive and negative samples, the ROC curve is a more stable indicator for reflecting the quality of the model. The ROC curve is closer to the (0, 1) point, the model performance is better. The AUC-ROC score is closer to 1, the model performance is better.

4.3. Data Sets

We perform experiments on DAGM data sets. The German DAGM 2007 dataset includes 10 types of woven fabrics, as shown in Figure 4. These texture defects usually exist on the surface of textile cloth and wallpaper. The red lines in the figures indicate the defect. There are 8050 training and testing sets each, and the ratio of positive and negative samples for each type is approximately 1:7. This dataset is often used in industrial optical defect detection.

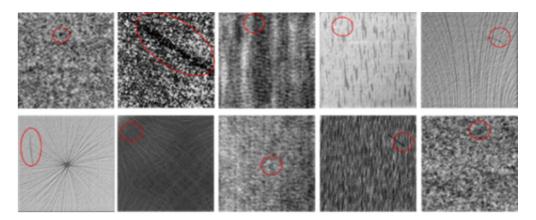


Figure 4. Defects detect data set.

4.4. Experimental Results

Visual Evaluation. The visual result of the model evaluation is shown in the Figure 5. We can accurately identify whether the surface of the object is defective and display its label on the image. Good means that the surface is free of defects; bad means that there are defects.

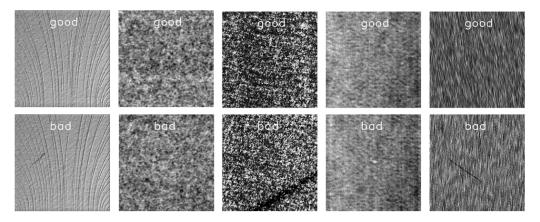


Figure 5. The surface defect detection result of our model.

In addition, in order to verify the effectiveness of the proposed MobileNet-v2-dense model, we compare the performance of the model to other latest three CNN-based defect detection algorithms, in terms of TNR/TPR and AUC-ROC indicators.

AUC-ROC Evaluation. We compare the AUC-ROC of defect detection results to Benjamin's method [21] on the task of detecting defects in DAGM test dataset, which is part of the DAGM dataset. The results are shown in Table 3. From these results, we see that our model has higher AUC-ROC scores for the ten classes of defects in the DAGM data set, so the performance of our model is better than Benjamin.

Table 3. Area under the receiver-operating characteristic (AUC-ROC) curve performance comparison, higher mean is better.

Class	1	2	3	4	5	6	7	8	9	10	Mean
Benjamin [21]	0.98	0.47	1	0.49	1	0.98	0.95	1	0.98	0.57	0.83
Ours	1	0.94	1	1	1	1	1	0.99	1	1	0.99

Furthermore, Figure 6 presents the ROC curves for the 10 classes of defects in the DAGM test data set. According to the ROC curve, the model achieves good classification performance in all classes except the second and eighth classes. The reason for the not perfect performance of the second and eighth classes of defects may be that their defects and textures are too similar. It should be noted that far fewer positive samples exist than negative samples in each class of DAGM dataset and that the proportion of positive and negative samples is approximately 1:7. So it can be concluded that our model performs well on ROC and can perform well even when positive and negative samples are unbalanced, benefitting from our asymmetric loss function.

TPR/TNR Evaluation. We calculate the TPR/TNR scores of defect detection results of our network, Weimer's [18] and Wang's [20].

As shown in Table 4, we detect all 10 classes of defects in DAGM dataset, whereas other methods only performed defect detection for six classes. The experimental results show that our method get the highest TPR/TNR mean values, which means that the detect results of our method have best detect performance in this three methods. The reason why TPR is slightly lower than TNR might be that texture of the sample is irregular, and selected positive samples with defects are mistaken for defect-free.

Appl. Sci. 2020, 10, 6085 9 of 12

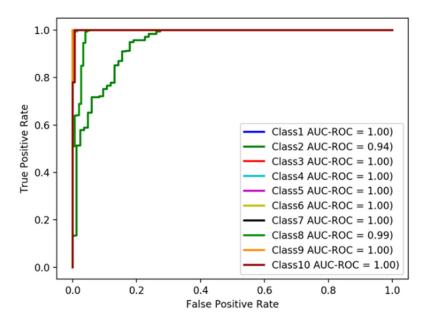


Figure 6. ROC curve and AUC-ROC for 10 classes of data in DAGM dataset.

Table 4. True negative r	ate Chivixi and	a true Dositive rate	t i it it i bei	IOI IIIAIICE COIIIDALISOII.

Class	O	urs	Weim	er [18]	Wang [20]	
Cluss	TPR (%)	TNR (%)	TPR (%)	TNR (%)	TPR (%)	TNR (%)
1	100	100	100	100	100	100
2	99.8	100	100	97.3	100	100
3	100	100	95.5	100	100	100
4	99.3	100	100	98.7	100	93.2
5	99.6	100	98.8	100	99.7	100
6	98.5	100	100	99.6	100	100
7	100	100	-	-	-	-
8	100	100	-	-	-	-
9	97.3	100	-	-	-	-
10	100	99.3	-	-	-	-
mean	99.45	99.93	99.05	99.27	99.95	98.87

Parameters Comparison. In practical applications, network models with too many parameters are difficult to use in resource-constrained hardware. In order to prove that our model is more convenient to apply in practical, we compared the parameters of our network to the three networks mentioned above. The experimental results are shown in Table 5. The parameters of the convolution filter in our model and other three methods are 2.12M, 6.07M, 6.98M, 135.59M. This result shows that our model has the fewest parameters, thus it is more convenient to apply on hardware with limited resources. We tested the speed of our model on small devices, and the detection rate can reach 24 ms per image on the NVIDIA Jetson Nano platform, which can meet industrial demands.

Table 5. Parameters of different networks.

Network	Benjamin's [21]	Weimer's [18]	Wang's [20]	Our
Parameters	6.07M	6.98M	135.59M	2.12M

4.5. Ablation Study

The ablation experiment is to verify the effectiveness of each innovation point. The idea of ablation experiment is to control variables and experiment on innovation points one by one.

Two-stage augmentation. In order to show the effectiveness of our two-stage augmentation, we performed the following ablation study. In this ablation study, we train the MobileNet-v2 without data augmentation, with dynamic augmentation, with static augmentation, with one-stage augmentation. Results are shown in Table 6. Our two-stage augmentation method has the highest TPR/TNR, which means that our augmentation method is the best and can effectively improve the performance of defect detection.

TPR	TNR
86.94%	82.65%
87.36%	83.15%
96.58%	86.72%
97.85%	82.22%
97.69%	87.76%
	86.94% 87.36% 96.58% 97.85%

Table 6. Comparison of TPR/TNR of different augmentation methods.

Multi-scale cascade. In order to verify the performance improvement of multi-scale cascade, we conducted a performance comparison experiment between our multi-scale cascade network MobileNet-v2-dense and MobileNet-v2. It should be noted that the accuracy of network changes is basically unchanged if no data augmentation is used. Therefore, we train the network based on the two-stage augmentation, and the loss function is the CE loss. The experimental results are shown in Table 7. Compared with MobileNet-v2, the TPR/TNR of our MobileNet-v2-dense is improved.

Network Model	TPR	TNR
MobileNet-v2	97.69%	87.76%
Our MobileNet-v2-dense	98.72%	88.81%

Table 7. Comparison of TPR/TNR of different networks.

Asymmetric loss. In experiment 3, we train with CE loss, focal loss, and asymmetric loss in the same MobileNet-v2-dense and two-stage data augmentation. The results are listed in Table 8. It can be seen that the asymmetric loss function can improve the TNR to 99.93% for the DAGM data set, with an increase of 2.28% compared to Focal loss.

 Loss Function
 TPR
 TNR

 CE loss
 98.72%
 88.81%

 Focal loss
 98.97%
 97.65%

 Asymmetric loss
 99.45%
 99.93%

Table 8. Comparison of TPR/TNR of different loss functions.

5. Conclusions

This paper proposes a surface defect detection method based on MobileNet-v2-dense. A new multi-scale cascade is proposed, i.e., the lightweight convolutional neural network structure MobileNet-v2-dense, which can be used on small embedded systems. A two-stage augmentation method is proposed to enlarge the training dataset, such that the robustness of the defect detection model is improved and training time is conserved. Moreover, an asymmetric loss function is introduced

to alleviate the imbalance of positive and negative samples. The defect detection method that combines these three aspects was proven to greatly improve the performance of defect detection on three types of data sets.

Author Contributions: Methodology, Z.L., H.Y., B.Z. and X.H.; Project administration, H.Y., B.Z. and X.H.; Software, Z.L.; Writing—original draft, Z.L.; Writing—review & editing, H.Y., B.Z. and X.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China grant number 2018YFB0504900 and the National Natural Science Foundation of China grant number No.61871424.

Acknowledgments: This work was supported partially by National Key R&D Program of China, 2018YFB0504900 and 2018YFB0504902, and by National Natural Science Foundation of China No.61871424.

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

References

- 1. Liu, K.; Luo, N.; Li, A. A New Self-Reference Image Decomposition Algorithm for Strip Steel Surface Defect Detection. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 4732–4741.
- 2. Song, L.; Li, R.; Chen, S. Fabric Defect Detection Based on Membership Degree of Regions. *IEEE Access* **2020**, *8*, 48752–48760.
- 3. Zhou, X.; Wang, Y.; Zhu, Q.; Mao, J.; Xiao, C.; Lu, X.; Zhang, H. A surface defect detection framework for glass bottle bottom using visual attention model and wavelet transform. *IEEE Trans. Ind. Inform.* **2019**, 16, 2189–2201.
- 4. Tan, D.P.; Li, L.; Zhu, Y.L.; Zheng, S.; Ruan, H.J.; Jiang, X.Y. An Embedded Cloud Database Service Method for Distributed Industry Monitoring. *IEEE Trans. Ind. Inform.* **2018**, *14*, 2881–2893.
- 5. Yu, J.; Zheng, X.; Liu, J. Stacked convolutional sparse denoising auto-encoder for identification of defect patterns in semiconductor wafer map. *Comput. Ind.* **2019**, *109*, 121–133.
- Yin, L.; Wang, K.; Tong, T. Improved Block Sparse Bayesian Learning Method Using K-Nearest Neighbor Strategy for Accurate Tumor Morphology Reconstruction in Bioluminescence Tomography. *IEEE Trans. Biomed. Eng.* 2020, 67, 2023–2032.
- 7. Xiangru, Y.U.; Jianpei, D.; Jinping, L.I. Tire impurity defect detection based on morphology and projection histogram. *J. Univ. Sci. Technol. China* **2019**, *49*, 49–54.
- 8. Da, Y.; Dong, G.; Shang, Y.; Wang, B.; Liu, D.; Qian, Z. Circumferential defect detection using ultrasonic guided waves: An efficient quantitative technique for pipeline inspection. *Eng. Comput.* **2020**, *37*, 1923–1943.
- 9. Wang, Q.X.; Li, D.; Zhang, W.; Cao, D.; Chen, H. Unsupervised defect detection of flexible printed circuit board gold surfaces based on wavelet packet frame. In Proceedings of the 2nd International Conference on Industrial and Information Systems, Dalian, China, 10–11 July 2010.
- 10. Deotale, N.T.; Sarode, T.K. Fabric Defect Detection Adopting Combined GLCM, Gabor Wavelet Features and Random Decision Forest. *3D Res.* **2019**, *10*, 5.
- 11. Zhou, J.; Wang, J.; Bu, H. Fabric defect detection using a hybrid and complementary fractal feature vector and fcm-based novelty detector. *Fibres Text. East. Eur.* **2017**, *25*, 46–52.
- Zhang, H.; Jin, X.; Wu, Q.J.; Wang, Y.; He, Z.; Yang, Y. Automatic Visual Detection System of Railway Surface Defects with Curvature Filter and Improved Gaussian Mixture Mode. *IEEE Trans. Instrum. Meas.* 2018, 67, 1593–1608.
- 13. Chen, H.; Liu, J.; Wang, S.; Liu, K. Robust Dislocation Defects Region Segmentation for Polysilicon Wafer Image with Random Texture Background. *IEEE Access* **2019**, *7*, 134318–134329.
- 14. Masci, J.; Meier, U.; Ciresan, D.; Schmidhuber, J.; Fricout, G. Steel defect classification with max-pooling convolutional neural networks. In Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, Australia, 10–15 June 2012.
- 15. Soukup, D.; Huber-Mörk, R. Convolutional neural networks for steel surface defect detection from photometric stereo images. In *Advances in Visual Computing, Proceedings of the 10th International Symposium on Visual Computing, Las Vegas, NV, USA, 8–10 December 2014*; Bebis, G., Boyle, R., Parvin, B., Koracin, D., McMahan, R., Jerald, J., Zhang, H., Drucker, M.S., Kambhamettu, C., Choubassi, E.M., et al., Eds.; Springer: Cham, Switzerland, 2014.

16. Faghih-Roohi, S.; Hajizadeh, S.; Núñez, A.; Babuska, R.; De Schutter, B. Deep convolutional neural networks for detection of rail surface defects. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016.

- 17. Ren, R.; Hung, T.; Tan, K.C. A generic deep-learning-based approach for automated surface inspection. *IEEE Trans. Cybern.* **2018**, *48*, 929–940.
- 18. Weimer, D.; Scholz-Reiter, B.; Shpitalni, M. Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Ann.* **2016**, *65*, 417–420.
- 19. Park, J.K.; Kwon, B.K.; Park, J.H.; Kang, D.J. Machine learning-based imaging system for surface defect inspection. *Int. J. Precis. Eng. Manuf. Green Technol.* **2016**, *3*, 303–310.
- 20. Wang, T.; Chen, Y.; Qiao, M. A fast and robust convolutional neural network-based defect detection model in product quality control. *Int. J. Adv. Manuf. Technol.* **2017**, *94*, 3465–3471.
- 21. Staar, B.; Lütjen, M.; Freitag, M. Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP* **2019**, *79*, 484–489.
- Srivastava, R.K.; Greff, K.; Schmidhuber, J. Training very deep networks. In Advances in Neural Information Processing Systems 28, Proceedings of the 29th Annual Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; Cortes, C., Lee, D.D., Garnett, R., Lawrence, D.N., Sugiyama, M., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2015.
- 23. Larsson, G.; Maire, M.; Shakhnarovich, G. Fractalnet: Ultra-deep neural networks without residuals. *arXiv* **2016**, arXiv:1605.07648.
- Bharath, H.; Arbelaex, P.; Girshick, R.; Malik, J. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
- 25. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
- 26. Cao, Y.; Guan, D.; Huang, W.; Yang, J.; Cao, Y.; Qiao, Y. Pedestrian Detection with Unsupervised Multispectral Feature Learning Using Deep Neural Networks. *Inf. Fusion* **2019**, *46*, 206–217.
- 27. Yang, S.F.; Deva, R. Multi-scale detection with DAG-CNNs. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
- 28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andretto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* 2017, arXiv:1704.04861.
- Guo, Y.W.; Yao, A.B.; Chen, Y.R. Dynamic network surgery for efficient DNNS. In Advances in Neural Information Processing Systems 29, Proceedings of the 30th Annual Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; Lee, D.D., von Luxburg, U., Garnett, R., Sugiyama, M., Guyon, I., Eds.; Curran, Associates, Inc.: Red Hook, NY, USA, 2017.
- 30. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Deep residual learning for image detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Detection, Las Vegas, NV, USA, 27–30 June 2016.
- 31. Zoph, B.; Cubuk, D.E.; Ghiasi, G.; Lin, Y.-T.; Shlens, J.; Le, V.Q. Learning data augmentation strategies for object detection. *arXiv* **2019**, arXiv:1906.11172.
- 32. Shrivastava, A.; Sukthanka, R.; Malik, J.; Gupta, A. Beyond skip connections: Top-down modulation for object detection. *arXiv* **2016**, arXiv:1612.06851.
- 33. Bulo, S.R.; Gerhard, N.; Peter, K. Loss max-pooling for semantic image segmentation. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2016; pp. 7082–7091.
- 34. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
- 35. Mei, S.; Yang, H.; Yin, Z.P. An Unsupervised-Learning-Based Approach for Automated Defect Inspection on Textured Surfaces. *IEEE Trans. Instrum. Meas.* **2018**, *67*, 1266–1277.
- 36. Kingma, D.P.; Jimmy, B. Adam: A method for stochastic optimization. arXiv 2014, arXiv:1412.6980.



© 2020 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 (http://creativecommons.org/licenses/by/4.0/).