





Research Article

Plastic Gasket Defect Detection Based on Transfer Learning

Xieyi Chen ¹, Dongyun Wang ^{1,2}, Jinjun Shao ^{1,2} and Jun Fan ^{1,2}

¹Department of Engineering Institute, Zhejiang Normal University, Jinhua, Zhejiang 321005, China

²Zhejiang Provincial Key Laboratory of Urban Rail Transit Intelligent Operation and Maintenance Technology and Equipment, Zhejiang Normal University, Jinhua, Zhejiang 321005, China

Correspondence should be addressed to Dongyun Wang; zsdwdy@zjnu.edu.cn

Received 22 April 2021; Revised 10 August 2021; Accepted 25 August 2021; Published 13 September 2021

Academic Editor: Zhaoqing Pan

Copyright © 2021 Xieyi Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To automatically detect plastic gasket defects, a set of plastic gasket defect visual detection devices based on GoogLeNet Inception-V2 transfer learning was designed and established in this study. The GoogLeNet Inception-V2 deep convolutional neural network (DCNN) was adopted to extract and classify the defect features of plastic gaskets to solve the problem of their numerous surface defects and difficulty in extracting and classifying the features. Deep learning applications require a large amount of training data to avoid model overfitting, but there are few datasets of plastic gasket defects. To address this issue, data augmentation was applied to our dataset. Finally, the performance of the three convolutional neural networks was comprehensively compared. The results showed that the GoogLeNet Inception-V2 transfer learning model had a better performance in less time. It means it had higher accuracy, reliability, and efficiency on the dataset used in this paper.

1. Introduction

Compared with metal gaskets, plastic gaskets have excellent insulation, corrosion resistance, and nonmagnetic properties, and they are lightweight, which make them very common in semiconductor, automobile, and aerospace industries, interior decoration, and other fields. However, in the production process, it leads to defective products because of immature processes or imperfect production conditions. To ensure the quality of their products, the manufacturer needs to sort out the plastic gaskets and remove unqualified products. Traditional plastic gasket sorting is mainly performed by human eyes. Artificial visual observation and sorting not only have low detection efficiency and high labor intensity but are also prone to be affected by the subjective factors of the testing personnel, resulting in false detection and missing detection. Moreover, long-term manual detection will cause certain damage to vision. In addition, using a manual method is not conducive to database management and storage.

Currently, noncontact detection is widely adopted with great advances in imaging technology [1]. Surface defect detection technology has been used in pharmaceutical

packaging, textile, leather, steel plate surface, and machine part testing. Convolutional neural networks (CNNs) have already been widely used in image recognition and classification because they can effectively extract data features [2]. In this respect, it is a bit like a soft sensor [3–5]. However, it becomes a challenge to apply convolutional neural network to directly detect and classify plastic gaskets for the wide variety of plastic gasket defects, as shown in Figure 1. To solve this problem, a pretrained network is applied to our dataset which is composed of two types of plastic gaskets, namely NG (the defective product) and OK (the undefective product) based on the known technology transfer learning [6, 7]. To meet the industrial requirements and realize the automatic detection of the plastic gasket, one of the first popular pretrained networks, “GoogLeNet Inception-V2 [8],” was applied. In this work, data augmentation [9] was applied to our data (Section 5.1) to generate new images from the original pictures. The hyperparameters of this model were then modified to fit our data. The use of transfer learning also saves a lot of model training time and reduces the probability of model overfitting due to the few datasets. In this paper, a GoogLeNet [10] classification network for plastic gasket images is proposed (Section 4).

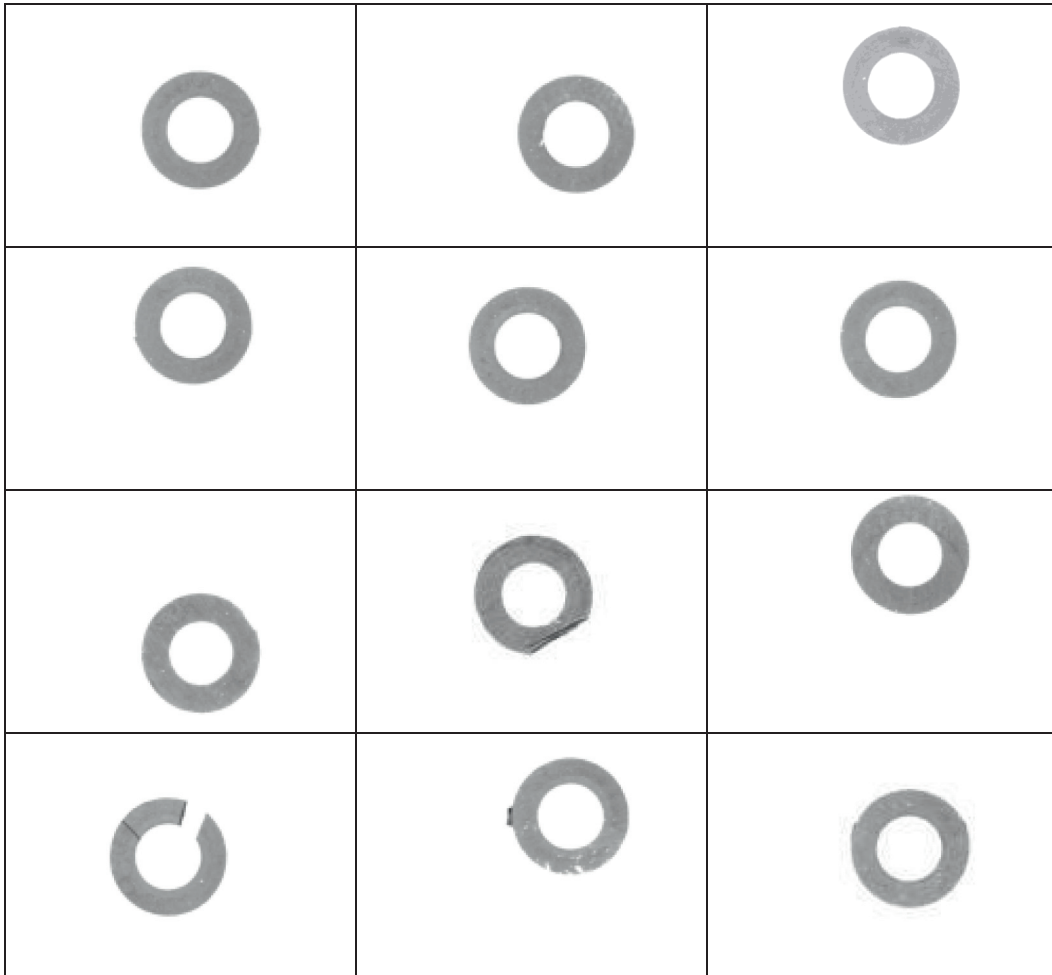


FIGURE 1: The first and second rows are the images without defects, whereas the third and fourth rows are the images with defects.

In the following section, the current research status of combining convolutional neural networks with various disciplines to solve problems in the corresponding fields will be discussed. Then, the design of the defect detection device is introduced in Section 3. In Section 4, the software research methods used in this study are described. In Section 5, the experimental process is described, the experimental results are analyzed, and the GoogLeNet Inception-V3 [11] and MobileNet-V1 [12] are compared with the chosen deep convolutional neural network (DCNN). Finally, Section 6 provides the conclusion.

2. Related Work

Research on image classification using the DCNN to solve specific needs has been prominent [13–21], as shown in Table 1. In Zhou et al.'s paper [15], a visual perception technology (VPT) framework based on deep learning was proposed, which relied on the image preprocessing (IP) scheme and the DCNN WR-IPDCNN. The framework was based on the improved DCNN [22] of LeNet-5 [23] to mine the newly established WR dataset, which significantly improved the automation and intelligence level of steel wire surface damage detection. The work reported in

Apostolopoulos et al.'s paper [16] implemented VGG-19 [24], MobileNet-V2 [25], Inception [10], Xception [26], and Inception ResNet-V2 [27] as a pretrained CNN [28] to detect COVID-19 from X-ray images. Using COVID-19, bacterial pneumonia, viral pneumonia, and health status images as datasets, the pretrained DCNN was applied to case dichroism and triad classification. In Alqahtani et al.'s paper [17], an autonomous method for the detection and classification of fatigue crack damage and the associated risk assessment of machinery components that are often made of ductile materials was proposed. The underlying algorithms in the proposed method were built upon the concept of the DCNN, where the execution time is much less than the time in visual inspection, and the detection and classification process is expected to be significantly less error-prone than that of visual inspection. In Fujioka et al.'s paper [18], the DCNN was used to distinguish between benign and malignant lesions on the maximum intensity projection of dynamic contrast-enhanced breast magnetic resonance imaging (MRI), and the model showed comparable diagnostic performance. In Alencastre-Miranda et al.'s paper [19], computer vision and deep learning networks were used to select and plant healthy billets, which increased the plant population and yield per hectare of sugarcane planting. In

TABLE 1: The classification of the related works.

Characteristics	Title
DCNN semantic segmentation ecological protection	Semantic segmentation of major macroalgae in coastal environments using high-resolution ground imagery and deep learning
Pretrained DCNN transfer learning medical diagnosis	Detection of epileptic seizure using pretrained deep convolutional neural network and transfer learning COVID-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks Deep learning enabled multiwavelength spatial coherence microscope for the classification of malaria-infected stages with limited labeled data size
DCNN defect detection risk assessment	Automatic detection of industrial wire rope surface damage using deep learning-based visual perception technology Classification of fatigue crack damage in polycrystalline alloy structures using convolutional neural networks Developing machine learning-based models for railway inspection
DCNN medical diagnosis	Deep learning approach with convolutional neural network for classification of maximum intensity projections of dynamic contrast-enhanced breast magnetic resonance imaging
Pretrained DCNN transfer learning quality assurance	Convolutional neural networks and transfer learning for quality inspection of different sugarcane varieties

this study, the DCNN and transfer learning were used to process image datasets to extend the results to different sugarcane varieties.

The aforementioned studies proved that using the DCNN for feature extraction and classification can be an effective method that can be applied to different production and living needs. In this study, based on machine vision technology, DCNN transfer learning was used to identify and classify plastic gasket surface defects to produce automatic and intelligent detection of the product.

3. Design of the Detection Device

The front and left views of the plastic gasket are shown in Figure 2. Plastic gasket products are prone to defects such as scratches, cracks, unfilled corner, and pits. They are different in form and have no obvious distribution. To improve the detection efficiency and accuracy of the defect characteristics and determine the actual detection requirements of the products, a set of software and hardware devices for automatic online detection and sorting of plastic gaskets based on the machine vision was designed.

The device was composed of a hardware and software part. The hardware part was divided into the main test bench and subtest bench. The details of the device are presented in Figure 3 and Table 2. The software part mainly includes the software platform, image acquisition, and image processing algorithms.

The lower end face of the gasket was taken as the detection object, and the detection method of the upper end face was the same, so it was not repeated. Because the image features collected from defective gaskets were numerous and inconsistent, GoogLeNet [10] transfer learning was used for image feature recognition and sorting to meet the detection requirements. The entire process of plastic gasket defect detection and classification is shown in Figure 4.

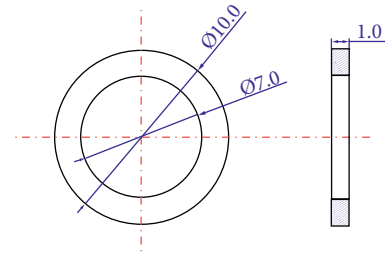


FIGURE 2: Structure of a plastic washer (units: mm).

4. Gasket Defect Classification Based on Transfer Learning

4.1. Problem Statement. The original dataset used in this study is shown in Figure 1. It was characterized by a wide variety of defects, and some of them were difficult to distinguish from normal products. Therefore, the accuracy and efficiency of the original manual detection method were extremely low. However, deep learning can easily solve this problem by automatically extracting features for image recognition and classification. In addition, the transfer learning method can greatly shorten the training model's time, make the model converge faster, and improve efficiency. However, deep learning requires the use of a large number of datasets for training, and because of the limitations of conditions, the datasets were insufficient. Therefore, we used the expanded the dataset (Section 5.1). In addition, the amount of data in the expanded dataset was still not enough. To prevent network overfitting caused by this, the parameter of the dropout layer was set to 0.5.

4.2. Overview of the GoogLeNet Inception-V2 Structure. GoogLeNet is a new deep learning structure proposed by Szegedy in 2014 [10]. Prior to this, AlexNet [29], VGG-19, and other structures all increased the depth of the network to achieve better training, but the increased number of layers

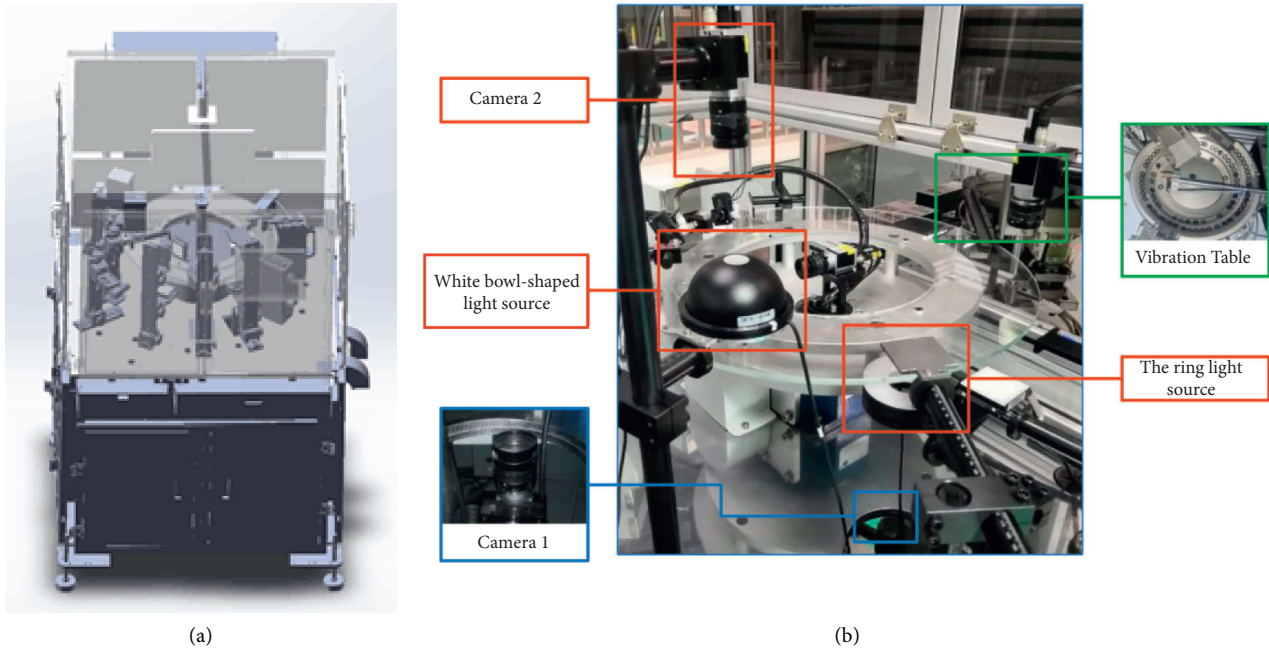


FIGURE 3: The setup of the device. (a) 3D modeling diagram. (b) 3D physical drawing.

TABLE 2: The components of the detecting device.

Sequence	Name	Parameter	Brand	Amount
1	Vibration plate	Working power: ≤ 1500 W Work pressure: 0.4–0.65 MPa	Weft lang	1
2	PLC	Model: CPU 1215C	Siemens	1
3	Cameras	Resolution: 659 px \times 494 px Frame rate: 120 ps Color: black and white Pixel size: 3.75 dpi \times 3.75 dpi	Basler	2
4	Lens	Target surface size: 2/3" The focal length: 35 mm Aperture range: F1.6 \sim F16 Minimum object distance: 0.3 m Minimum object distance field of view: 2/3" 8.1 mm \times 6.0 mm Interface: C Interface Deformation rate: -0.8% ($\gamma = 5.5$)	Computar	2
5	White bowl-shaped light source	Size: W140 mm \times H73 mm Angle: 30° Voltage: 24V Power: 5.6W Color: white, 6600K (color temperature)	Weft lang	1
6	Ring light source	Size: OD100 mm \times ID65 mm \times H24 mm Voltage: 24V Power: 12W Color: white, 6600K (color temperature)	Weft lang	1

brought many downsides, such as overfit, gradient disappearance, and gradient explosion. The Inception model is proposed to improve training results from a different perspective, namely the ability to use computational resources more efficiently and extract more features with the same amount of computation, thereby improving training results.

GoogLeNet Inception is currently available in four versions, the classic being Inception-V2. Inception-V2 was derived from the original Inception-V1, which changed the convolution from a size of 5×5 in the third branch to two convolutions of sizes 3×3 , as shown in Figure 5. Owing to the introduction of batch normalization (BN), the network

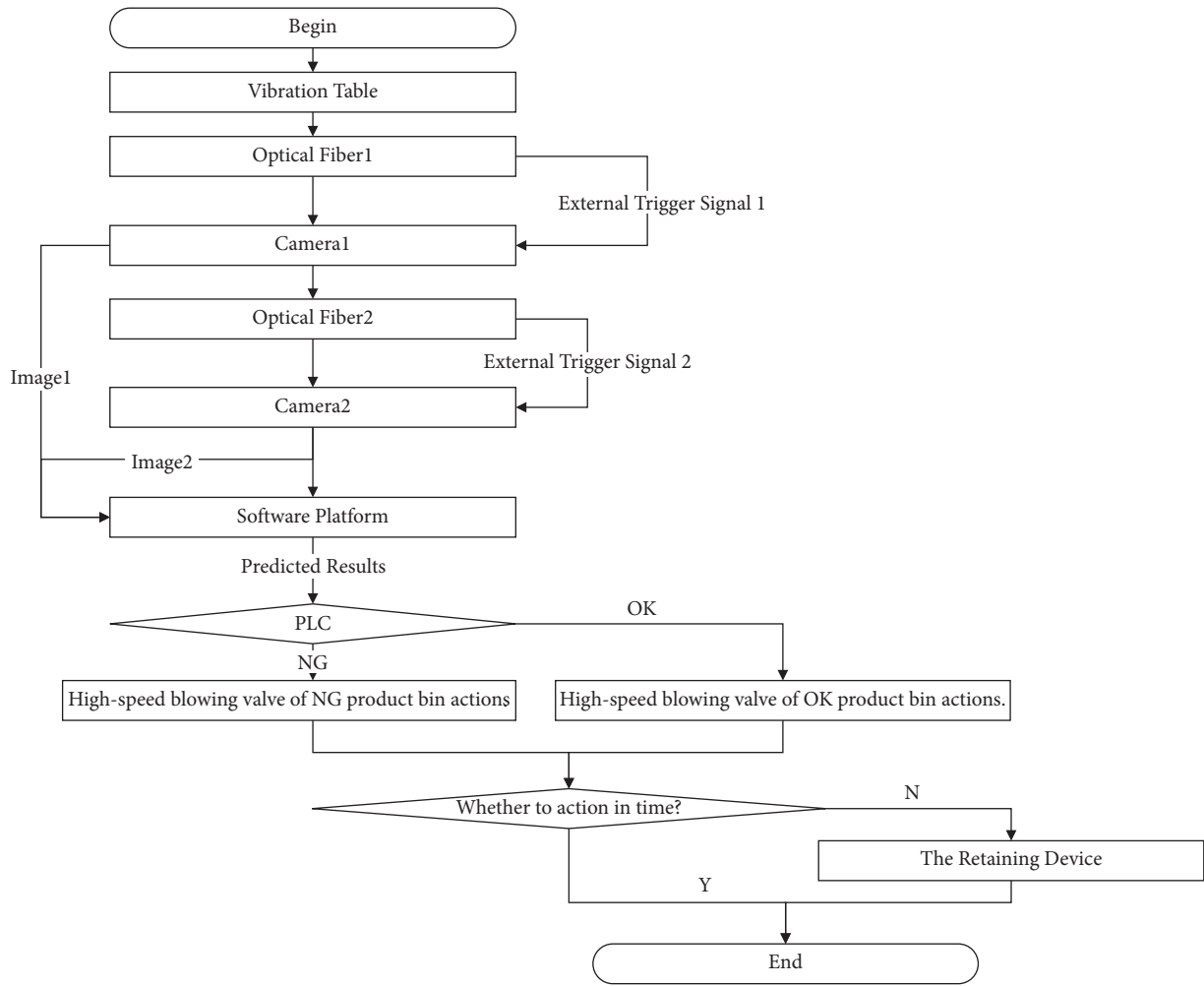


FIGURE 4: Flowchart of the device.

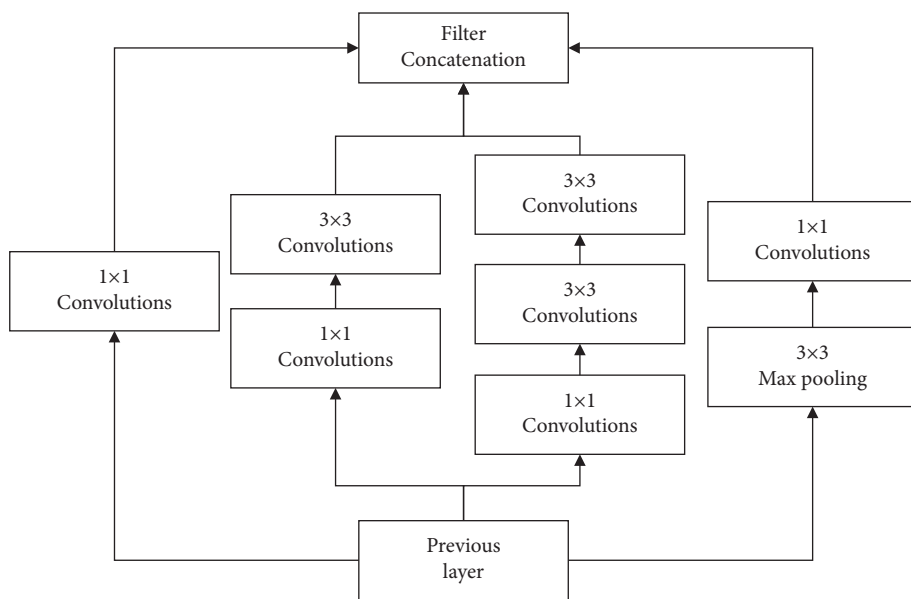


FIGURE 5: The structure of Inception-V2.

can set a higher learning rate and make the model converge faster. Furthermore, the training samples can be more thoroughly disturbed so that the network can achieve a higher verification accuracy.

4.3. Pretraining and Fine-Tuning Learning. Transfer learning computer vision technology [30] is widely used for it can accelerate the convergence of models. Using migration parameters to initialize the network can improve the generalization performance, even if the target task has been significantly adjusted. Many studies have demonstrated the feasibility of the approach [31]. They showed how to transfer the trained model parameters to the new model to help with the new model training. In this study, the initial weight parameters came from the pretrained GoogLeNet network model, and we only changed the last layer such that it corresponded to our label number. In addition, although the softmax cross-entropy loss function degenerates into a sigmoid cross-entropy loss function in a binary classification problem, the two are completely equivalent mathematically. However, in various materials used in our network model training process, we found that the effect of the sigmoid cross-entropy loss function was better than that of the softmax cross-entropy loss function. Therefore, we chose the sigmoid cross-entropy loss function for training and chose the sigmoid function for the output layer. Then, depending on the network training situation, we adjusted the choice of the optimizer (Section 4.4). We chose a learning rate of 0.01, which remained unchanged in subsequent iterations. The batch size selected in the training process was 16, and the maximum number of iterations was 20 000, which was 320 epochs.

4.4. Choice of Optimizer. The goal of the DCNN training process is to minimize the loss value, and after we define the loss function, the optimizer becomes useful. There are many different types of optimizers that can be chosen. This study focused on comparing the two types of optimizers, namely the gradient descent method and the momentum optimization method. The mathematical formulas for the two optimizers are as follows:

(1) Gradient descent:

$$W_{t+1} = W_t - \eta_t \Delta J(W_t). \quad (1)$$

(2) Momentum:

$$v_t = \alpha v_{t-1} + \eta_t \Delta J(W_t, X^{(i)}, Y^{(i)}), \quad (2)$$

$$W_{t+1} = W_t - v_t.$$

By comparing the formulas, it can be found that for the GDA optimizer, the update adjustment of model parameters is related to the gradient of the loss function on the model parameters. Therefore, the training speed of the model using this optimizer is relatively slow, and it is easy for the model to fall into the local optimal solution. The momentum optimization algorithm introduces

momentum-accumulating historical gradient information, which can solve the aforementioned problems adequately. In the actual training process, the model did occasionally fall into the local optimal solution, resulting in a low accuracy and a loss value that could not be reduced. Finally, using the momentum optimization method, the model jumped out of the local optimal solution. The momentum was set to 0.9.

4.5. Indicators for Model Evaluation. The seven indicators used to evaluate the performance of the DCNN were accuracy (AC), sensitivity (SE), specificity (SP), F_1 score, receiver operating characteristic (ROC) curve, precision-recall (PR) curve, and area under curve (AUC) values [32].

The accuracy of classification (AC) is defined as follows:

$$AC = \frac{TP + TN}{P + N} \times 100\% \quad (3)$$

$$= \frac{TP + TN}{TP + FN + TN + FP} \times 100\%,$$

where TP refers to the number of correctly classified defective products, FP refers to the number of incorrectly labeled nondefective products, TN refers to the number of correctly classified nondefective products, FN refers to the number of incorrectly labeled defective products, P refers to the number of defective products, and N refers to the number of nondefective products.

Sensitivity (SE) is defined as the percentage of the product correctly identified as defective and is expressed as follows:

$$SE = \frac{TP}{P} \times 100\%, \quad (4)$$

$$= \frac{TP}{TP + FN} \times 100\%.$$

Specificity (SP) is defined as the percentage of a non-defective product that is correctly classified as nondefective.

$$SP = \frac{TN}{N} \times 100\%, \quad (5)$$

$$= \frac{TN}{TN + FP} \times 100\%.$$

F_1 score is defined as the harmonic mean of precision and sensitivity:

$$\text{precision} = \frac{TP}{TP + FP},$$

$$\text{recall} = \frac{TP}{TP + FN}, \quad (6)$$

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

$$= \frac{2TP}{2TP + FN + FP},$$

where the precision rate is defined as the proportion of correctly classified defective products to all defective products divided by the classifier, and the recall rate is defined as the proportion of correctly classified defective products to the number of defective products.

ROC is a curve connected by points of the false-positive rate (FPR) (7) and true-positive rate (TPR) (8). For the ROC curve, the closer the curve is to the upper left of the graph, the better the performance of the model is. AUC is the area formed between the ROC curve and the horizontal axis. The larger the value, the better the model, and its optimal value is 1. The PR is a curve connected by the points of recall and precision. For the PR curve, the closer the curve is to the upper right of the graph, the better the performance of the model is. In general, when a model's performance evaluation is better than that of another model under both ROC and PR curves, it can be proved that the performance of that model is better than that of the other models.

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \quad (7)$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad (8)$$

5. Experimental Results

5.1. Image Preprocessing. In deep learning, a large amount of data must be used for training to suppress model overfitting. Generally, geometric transformations can be used to enhance datasets. This paper uses the Python language and the "opencv" library to expand the dataset in Visual Studio. The main methods used were brightness enhancement, contrast enhancement, angle rotation, and image flipping, and the transformations are shown in Figure 6. Because some pictures lost their original features after having their brightness enhanced, they could not be used as training samples, so they were removed. In addition, because the picture contained many blank parts, it did not contain any useful information, and it increased the amount of calculation of the neural network and slowed down the training speed. This also caused the program to occupy too much video memory, which could easily cause video memory overflow and crashing of the program. When the image angle was rotated and transformed, black bars appeared on both the sides of the image, which could easily cause a neural network feature extraction error and reduce the accuracy of the neural network. Moreover, the length and width dimensions of the pictures were not consistent. Thus, when the subsequent picture size was modified, it could easily cause distortion of the picture. To solve these problems, all pictures were cropped. Indeed, in the actual training process, we found that the accuracy of the model in the validation set improved after data enhancement. Therefore, we concluded that when the dataset was too small, proper data enhancement could improve network performance and reduce overfitting. In total, there were 1666 processed images.

Because the input image channel of the neural network must be three channels, the size requirement was

224 dpi × 224 dpi. Therefore, when reading a picture, a single-channel picture was copied to generate a three-channel RGB picture, and the picture size was modified to meet the network input picture size requirements. In addition, to prevent the neural network from overfitting, the accuracy of the neural network in the training set was high at the beginning. However, this led to another problem because the gradient could not fall. Therefore, the picture was randomly scrambled when making and reading TFRecord files.

5.2. Results and Discussion. The final number of pictures used in the experiment was 1666, including 1350 NG pictures and 316 OK pictures. They were divided into training, validation, and test sets (Table 3) with a ratio of 8:2:2. During training, verification, and testing, the computer configured was a 64-bit Windows 10 operating system, the GPU is NVIDIA GeForce GTX 1050 Ti, the deep learning framework was TensorFlow, and the programming language was Python.

In general, the network model parameters used for comparison were set according to those recommended in GoogLeNet Inception-V2. It can be seen from Figure 7 that when the three models reached the 1000th iteration, that is, the 16th epoch, the accuracy rate on the verification set reached approximately 80%. At the end of the training, the loss values converged to 0%, the accuracy rate on the training set converged to 100%, and the accuracy rate on the verification set reached approximately 90%. It can be seen that GoogLeNet Inception-V3 converged the fastest.

Figures 8(a)–8(c) correspond to the confusion matrices of GoogLeNet Inception-V2, GoogLeNet Inception-V3, and MobileNet-V1, respectively. Figure 9(a) shows the ROC curves corresponding to the three DCNNs. It can be seen from the figure that the ROC curves of the three networks are projected to the upper left corner, among which GoogLeNet Inception-V2 is closest to the upper left corner of the graph and completely surrounds the ROC curves of GoogLeNet Inception-V3 and MobileNet-V1, whereas the ROC curves of GoogLeNet Inception-V3 and MobileNet-V1 intersect. Figure 9(b) shows the corresponding PR curves of the three DCNNs. It can be seen from the figure that the PR curves of the three networks project to the upper right corner of the graph, among which GoogLeNet Inception-V2 is closest to the upper right corner of the graph and completely surrounds the PR curves of GoogLeNet Inception-V3 and MobileNet-V1, whereas the PR curves of GoogLeNet Inception-V3 and MobileNet-V1 intersect.

Table 4 shows the accuracy, sensitivity, specificity, F_1 score, AUC indexes, network training time, and test time of recognition obtained by the three DCNNs. It can be seen from the table that in terms of accuracy, specificity, F_1 score, and AUC, GoogLeNet Inception-V2 had the highest values, the accuracy was as high as 95.495%, and the specificity value reached 100%. The second was GoogLeNet Inception-V3, whereas MobileNet-V1 was always the lowest, even with a specificity value of only 44.444%. In terms of sensitivity, the

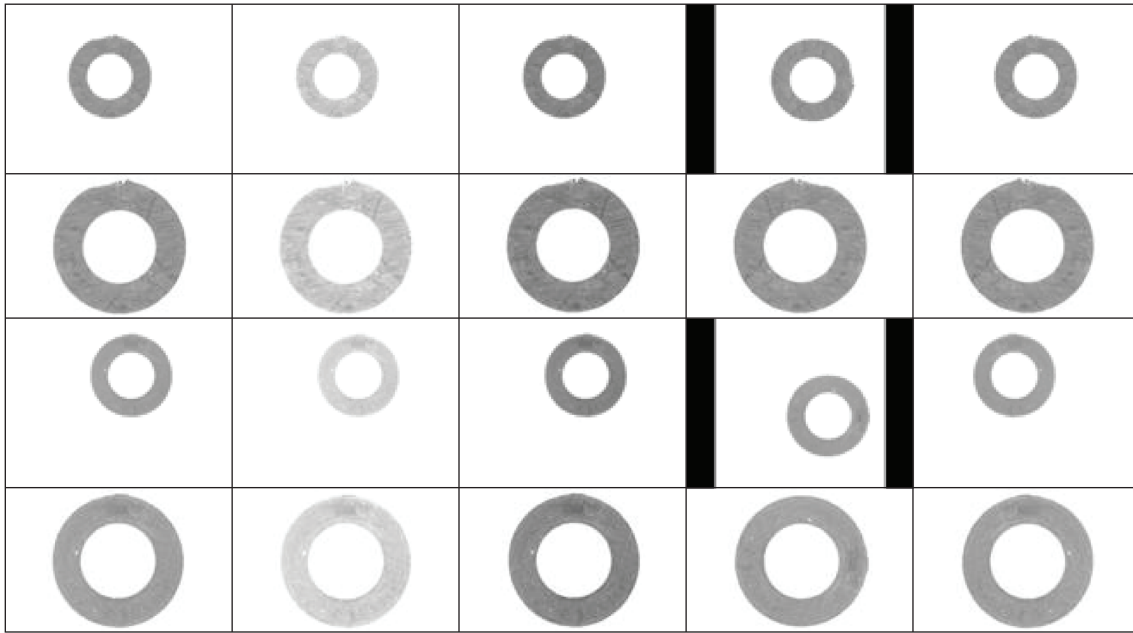


FIGURE 6: The first row contains the original NG images and the images transformed by brightness enhancement, contrast enhancement, angle rotation, and image flip. The second row includes the cropped images of the first row. The third and fourth rows are OK pictures arranged like NG pictures.

TABLE 3: Image quantity allocation table.

	Train	Verification	Test
NG	810	270	270
OK	190	63	63

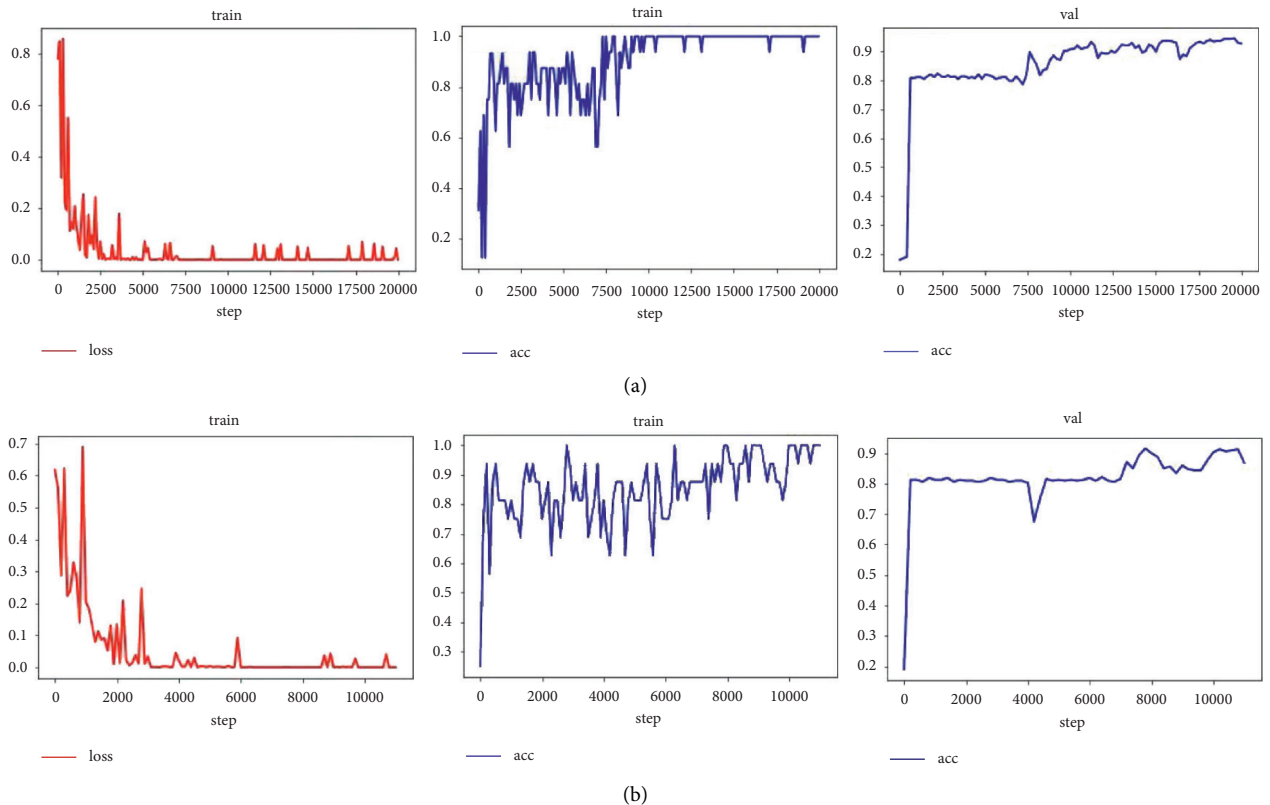


FIGURE 7: Continued.

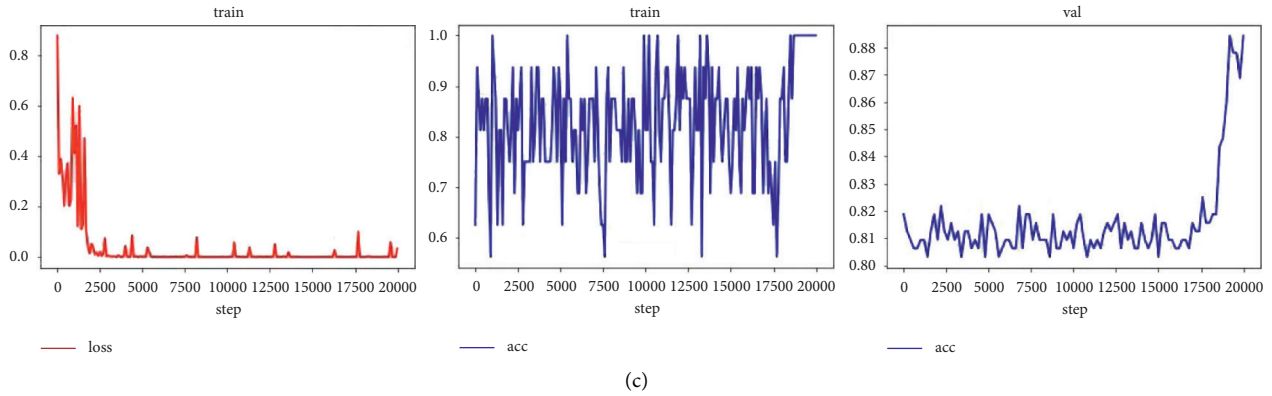


FIGURE 7: The loss values and accuracy on the training set and accuracy on the validation set of (a) GoogLeNet Inception-V2, (b) GoogLeNet Inception-V3, and (c) MobileNet-V1.

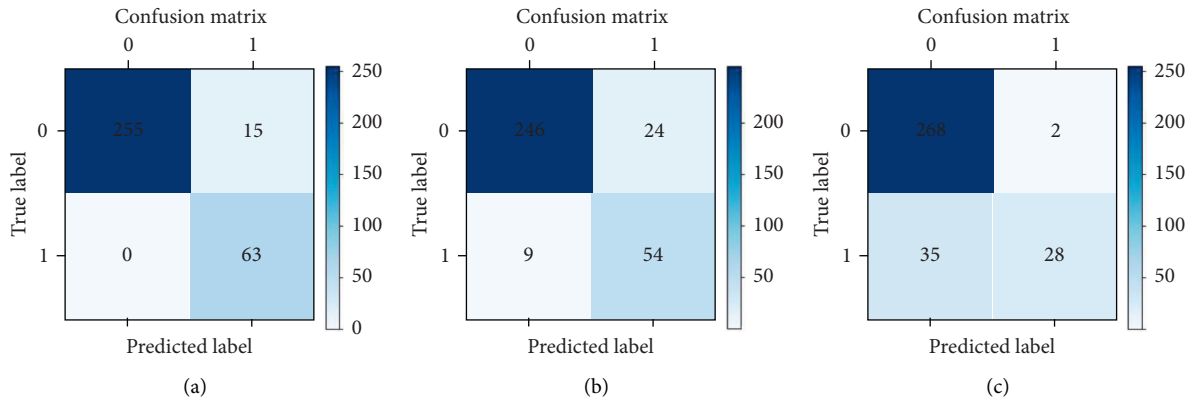


FIGURE 8: The confusion matrix of (a) GoogLeNet Inception-V2, (b) GoogLeNet Inception-V3, and (c) MobileNet-V1.

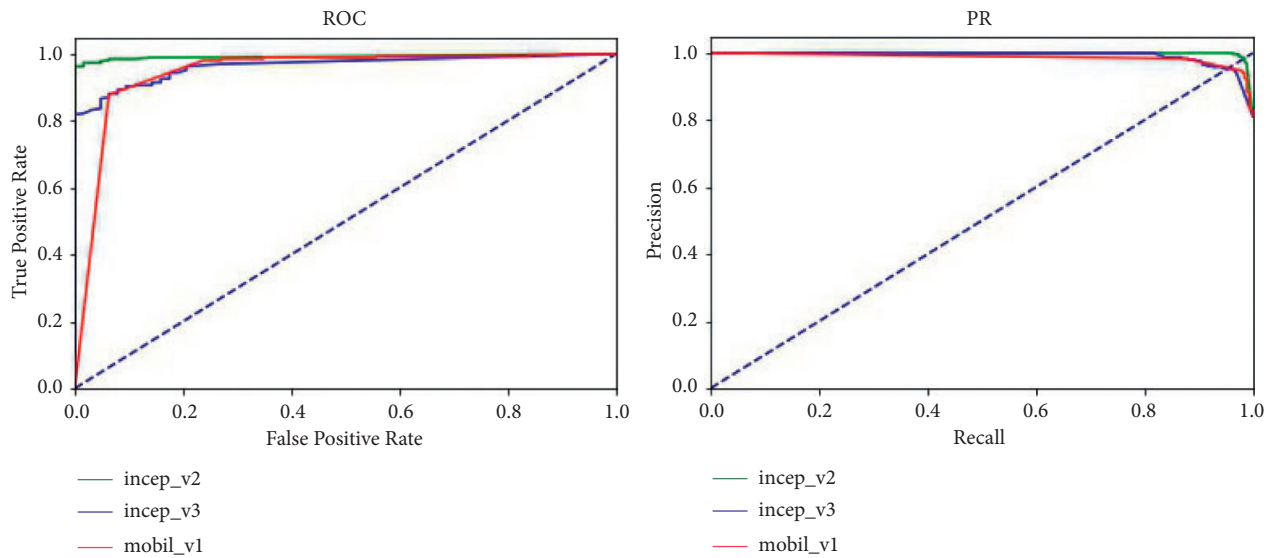


FIGURE 9: The ROC curves and PR curves of GoogLeNet Inception-V2, GoogLeNet Inception-V3, and MobileNet-V1.

MobileNet-V1 value reached the highest value, followed by GoogLeNet Inception-V2 and GoogLeNet Inception-V3. Owing to the application of transfer learning, all three

network models converge in a very short time. MobileNet-V1 had the shortest identification time, followed by GoogLeNet Inception-V2 and GoogLeNet Inception-V3. The

TABLE 4: Performance metrics of DCNN models with train and test time.

CNN	Accuracy (%)	Sensitivity (%)	Specificity (%)	F_1 score	AUC	Train time (h)	Test time (ms)
GoogLeNet Inception-V2	95.495	94.444	100.000	0.971	0.988	1.81	12.455
GoogLeNet Inception-V3	90.09	91.111	88.889	0.937	0.953	2.26	30.123
MobileNet-V1	88.889	99.259	44.444	0.935	0.909	1.33	5.26

recognition time of GoogLeNet Inception-V2 for each image was 12.455 ms, which was only 7.795 ms longer than that of MobileNet-V1 and could fully meet the requirements of industrial production.

Based on the above analysis, it can be found that although GoogLeNet Inception-V3 is an improvement of GoogLeNet Inception-V2, such an improvement only reflects the advantage of faster convergence in our dataset because our dataset features were not complicated. In the overall performance evaluation, GoogLeNet Inception-V2 was better than the other two networks.

6. Conclusion and Future Work

In this study, a surface defect visual classification device for plastic gaskets was designed based on GoogLeNet Inception-V2 transfer learning. The results show that when the datasets are few, the training effect can still be achieved through the network model transfer learning. This proves that under the restriction of conditions, when sufficient training sets cannot be obtained, the mode of transfer learning can be used to solve the dilemma. Using the test set to compare the performance indexes of the three types of networks, it is shown that the GoogLeNet Inception-V2 network model had higher accuracy, reliability, and efficiency on the dataset used in this paper. This indicates that the device designed in this study can fully meet production needs and obtain higher profits for manufacturers in this industry.

Since our experimental results basically meet the current industrial production testing requirements, we will not perform extensive research on the defect detection of plastic gaskets. In the next stage, we plan to conduct research on the surface defects of the camshaft. In this study, we will classify each defect to achieve subsequent automatic repair and elimination. In future studies, CNNs can be integrated with different disciplines (such as environmental science and biomedical science) to solve key problems in corresponding disciplines and make such research more socially valuable.

Data Availability

All the picture data in this study were taken from the experimental equipment. The code of this paper is not available to the public.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this study.

Acknowledgments

This work was supported by the National University Students Science and Technology Innovation Project (201910345063X) and the Key Research and Development Program of Zhejiang Province under Grant 2020C01153 and Grant 2019C01134, and in part by the National Social Science Foundation of China under Grant 17BGL086.

References

- [1] Z. Liu, W. Liu, and Z. Han, "A high-precision detection approach for catenary geometry parameters of electrical railway," *IEEE Transactions on Instrumentation & Measurement*, vol. 66, no. 7, pp. 1–11, 2017.
- [2] Z. Zhen, J. Liu, Z. Zhang et al., "Deep learning based surface irradiance mapping model for solar PV power forecasting using sky image," *IEEE Transactions on Industry Applications*, vol. 56, no. 4, pp. 3385–3396, 2020.
- [3] D. Slišković, R. Grbić, and Ž. Hocenski, "Methods for plant data-based process modeling in soft-sensor development," *Journal for Control, Measurement, Electronics, Computing and Communications Automatika*, vol. 52, 2017.
- [4] H. Darvishi, D. Ciuonzo, R. Eide, and P. S. Rossi, "A data-driven architecture for sensor validation based on neural networks," in *Proceedings of the 2020 IEEE Sensors*, Rotterdam, Netherlands, Oct 2020.
- [5] R. Eide, D. Ciuonzo, P. S. Rossi, and H. Darvishi, "Sensor-fault detection, isolation and accommodation for digital twins via modular data-driven architecture," *IEEE Sensors Journal*, vol. 99, 2020.
- [6] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [7] L. Shao, F. Zhu, and X. Li, "Transfer learning for visual categorization: a survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, 2014.
- [8] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," 2015, <https://arxiv.org/abs/1502.03167>.
- [9] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, 2019.
- [10] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, June 2015.
- [11] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015, <https://arxiv.org/abs/1512.00567>.
- [12] A. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, <https://arxiv.org/abs/1704.04861>.
- [13] J. Balado, C. Olabarria, J. S. Martinez, J. R. P. Rodriguez, and A. Pedro, "Semantic segmentation of major macroalgae in coastal environments using high-resolution ground imagery

- and deep learning,” *International Journal of Remote Sensing*, vol. 42, no. 5, pp. 1785–1800, 2021.
- [14] H. S. Nogay and H. Adeli, “Detection of epileptic seizure using pretrained deep convolutional neural network and transfer learning,” *European Neurology*, vol. 83, no. 6, pp. 602–614, 2021.
- [15] P. Zhou, G. Zhou, H. Wang, D. Wang, and Z. He, “Automatic detection of industrial wire rope surface damage using deep learning-based visual perception technology,” *Ieee Transactions on Instrumentation and Measurement*, vol. 70, no. 14, pp. 523–538, 2021.
- [16] I. D. Apostolopoulos and T. Bessiana, “Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks,” 2020, <https://arxiv.org/abs/2003.11617>.
- [17] H. Alqahtani, S. Bharadwaj, and A. Ray, “Classification of fatigue crack damage in polycrystalline alloy structures using convolutional neural networks,” *Engineering Failure Analysis*, vol. 119, no. 8, pp. 104–122, 2021.
- [18] T. Fujioka, Y. Yashima, J. Oyama et al., “Deep-learning approach with convolutional neural network for classification of maximum intensity projections of dynamic contrast-enhanced breast magnetic resonance imaging,” *Magnetic Resonance Imaging*, vol. 75, no. 16, pp. 1–8, 2021.
- [19] M. Alencastre, R. M. Johnson, and H. I. Krebs, “Convolutional neural networks and transfer learning for quality inspection of different sugarcane varieties,” *Ieee Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 787–794, 2021.
- [20] N. Singla and V. Srivastava, “Deep learning enabled multi-wavelength spatial coherence microscope for the classification of malaria-infected stages with limited labelled data size,” *Optics & Laser Technology*, vol. 130, no. 3, pp. 106–335, 2020.
- [21] C. Yang, Y. Sun, C. Ladubec, and Y. Liu, “Developing machine learning-based models for railway inspection,” *Applied Sciences-Basel*, vol. 11, no. 1, pp. 13–33, 2021.
- [22] G. Huang, Z. Liu, V. D. M. Laurens, and K. Q. Weinberger, “Densely connected convolutional networks,” 2016, <https://arxiv.org/abs/1608.06993>.
- [23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [24] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, <https://arxiv.org/abs/1409.1556>.
- [25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Inverted residuals and linear bottlenecks: mobile networks for classification, detection and segmentation,” 2018, <https://arxiv.org/abs/1801.04381>.
- [26] F. Chollet, “Xception: deep learning with depthwise separable convolutions,” 2016, <https://arxiv.org/abs/1610.02357>.
- [27] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” 2016, <https://arxiv.org/abs/1602.07261>.
- [28] K. Fukushima, “Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.
- [29] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” *Communication of the ACM*, vol. 60, pp. 1097–1105, 2012.
- [30] H. C. Shin, H. R. Roth, M. Gao et al., “Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning,” *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.
- [31] E. T. Moore, W. P. Ford, E. J. Hague, and J. Turk, “An application of CNNs to time sequenced one-dimensional data in radiation detection,” in *Proceedings of the Volume 10986, Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imagery*, July 2019.
- [32] T. Pham, “Classification of COVID-19 chest X-rays with deep learning: new models or fine tuning?” *Health Information Science and Systems*, vol. 9, pp. 11–22, 2020.