



# Detection of defects in voltage-dependent resistors using stacked-block-based convolutional neural networks

Tiejun Yang<sup>1</sup> · Tianshu Zhang<sup>1</sup> · Lin Huang<sup>1</sup>

Published online: 2 July 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

Voltage-dependent resistors (VDRs) are important circuit-protection devices. Their performance is affected by packaging quality. To identify VDR packaging defects more accurately and efficiently, we have proposed a convolutional neural network (CNN)-based VDR appearance quality inspection method that includes four stages: image acquisition, data augmentation, neural architecture design, and CNN training and testing. In designing the neural architecture, we have proposed two VDR-oriented network blocks, which consist of a compressed subnet and a multiscale subnet. Then, a stacking-block-based neural architecture design (BlockNAD) strategy is employed to determine the number of blocks. The last block is connected to a classification layer composed of a global average pooling (GAP) layer and a full connection (FC) layer. Further, using a VDR dataset containing 8058 images, we compared the identification performances of the candidate networks with different structures on 12 categories of VDR defects by adopting a variety of indicators, such as the mean average precision (mAP) and average test time per sample. The experimental results of the proposed method demonstrate competitive results compared to the state-of-the-art methods in identifying VDR defects, with a mAP value of approximately 99.9% and an average test time per sample of approximately 3 ms.

**Keywords** Neural architecture design · Deep convolutional neural networks · Defect detection · Voltage-dependent resistors

## 1 Introduction

Voltage-dependent resistors (VDRs) are important circuit-protection devices and have been widely used in various fields such as household appliances, power systems, and security systems. A VDR has two parts: a resistor body and pins (Fig. 1). The former consists of a round casing, and the latter consists of two fine wires. Due to the fragility of the round casing structure, the resistor body is highly susceptible to damage during the packaging process, resulting in defects such as surface damage, incomplete wrapping of pin joints, and surface protrusion; these affect VDR performance and may thus cause unpredictable consequences. Therefore, it is necessary to inspect the quality of a VDR's appearance before use. The traditional manual inspection method [1] is

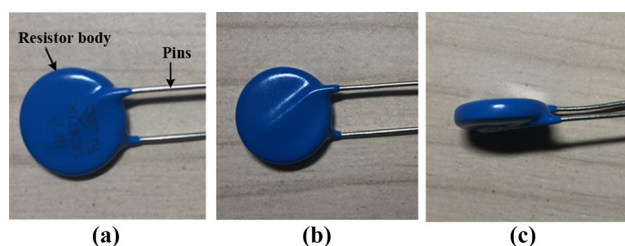
prone to being influenced by subjective judgment, making it difficult to achieve good efficiency and accuracy.

Machine learning enables learning from empirical data for automatic classification. Compared with manual inspection, automatic classification has the advantages of high speed and high precision, has been widely used in the field of industrial inspection [2–4], and is gradually replacing the manual inspection method. Compared to common inspection objects, VDRs are unique. First, due to a VDR's fine pin structure and smooth pin material, it is difficult to obtain a clear picture of the pins. Second, due to uneven illumination, the acquired image often has significant noise and poor contrast. Third, VDR defects are randomly distributed on the surface, and there are various defect types. These issues all pose challenges to VDR defect identification. Currently, machine-learning-based VDR defect recognition has been rarely investigated, even though there are references to defect-recognition methods developed in other fields. Chondronasios et al. [5] used a gradient co-occurrence matrix to extract statistical features of an image and achieved the classification of surface defects of an extruded aluminum surface through an artificial neural network. Li and Tsai [6]

---

✉ Lin Huang  
hlcucu@qq.com

<sup>1</sup> Guangxi Key Laboratory of Embedded Technology and Intelligent System, Guilin University of Technology, Guilin, Guangxi, China



**Fig. 1** VDR structure: **a** front view, **b** back view, and **c** side view

inspected defects of polycrystalline silicon solar cells using a single wavelet coefficient as a feature. Zhang et al. [7] used the Gaussian pyramid and Gabor filter to extract copper surface-defect features to generate a saliency map and accomplished defect-detection tasks using a Markov model. Ravikumar et al. [8] used a histogram method to extract defect features and achieved surface-defect recognition on machine parts using a decision-tree method. These manual feature-extracting methods depend largely on the quality of the manually extracted features and primarily address only a few defect types; thus, they are unusable in VDR defect detection.

Deep learning [9] does not require manual feature extraction. It can automatically learn the effective features of a target based on empirical data and has allowed breakthroughs in solving many image-recognition problems. The deep convolutional neural network (CNN) [10] is a deep learning technique that has attracted much attention and has been widely applied. It was initially applied to highly challenging tasks such as handwritten character recognition [11]. With their strong learning ability, CNNs have been successfully used in various computer vision tasks [12–15].

In recent years, CNNs have also been used in the field of defect identification. To detect surface defects in steel, Soukup and Huber-Mörk [16] designed a CNN that consists of two convolutional layers and two pooling layers and uses one fully connected layer to integrate the features. This method can identify only a few categories of image defects and is not suitable for multi-category defect classification. Tao et al. [17] introduced CNNs into the field of spring wire defect detection for the first time. They used the convolutional and pooling layers of a VGG-16 network to extract a region of interest (ROI) feature map [18] that was introduced to an ROI inspection module and a classification module with full connection and softmax classifiers, respectively. The results of the inspection and classification modules were combined to detect the spring wire defects, achieving good detection results. Feng et al. [19] detected infrastructure surface defects based on the Resnet [20] and AL technologies [21], which can reduce the number of images that need to be annotated and thus reduce the workload of field experts. Huang et al. [22] proposed a surface

defect detection model, mainly using U-net as the backbone network, combined with a saliency image generator and a defect localization network. Good results have been achieved in the surface defect detection of magnetic tiles, and the time cost has been significantly reduced. Wang et al. [23] used a multilayer CNN to conduct two-stage classification on six-category defect samples in the DAGM2007 dataset and achieved good results. Faghieh-Roohi et al. [24] established a CNN containing three convolution layers, three pooling layers, and three fully connected layers for rail-surface-defect detection tasks and achieved a recognition rate of 92% on their dataset. Chen et al. [25] constructed a cascading detection network from thick to fine sizes to detect defects in high-speed rail fasteners; it uses the SSD [26] and YOLO [27] detectors to locate the cantilever node and its fasteners and then employs a classifier containing four convolution layers and two fully connected layers to classify the fastener defects. Tao et al. [28] designed an automatic metal-surface defect detection system with inspection and classification modules that uses a cascaded automatic encoding structure for the location segmentation of defects and then exports the semantically segmented images into a CNN with five convolutional layers, three maximum pooling layers, and a fully connected layer for classification; it is very effective on industrial defect datasets.

All of the above methods rely on fully connected layers to integrate features for classification at the classification stage. However, fully connected layers have many parameters and primarily rely on the dropout technique to prevent overfitting. Yu et al. [29] developed a fully convolutional network (FCN) framework based on the FCN to detect surface defects in an industrial environment; it combines image segmentation and inspection tasks and has achieved good results. Cha et al. [30] used a combination of CNN and the sliding-window technique to scan images to perform a two-category inspection of concrete cracks; the first three convolutional layers are used for feature extraction of input images and the last convolutional layer is used to output the two-category feature map, which is ultimately classified by the softmax classifier; this achieves classification without using a fully connected layer, which reduces the number of network parameters while producing good results. However, these CNNs are constructed layer by layer manually, which requires considerable effort to adjust the network architecture and parameters.

To construct CNN more efficiently and to identify VDR appearance quality defects more accurately and efficiently, we propose here a CNN-based VDR defect detection method. The main contributions of this paper are as follows:

- (1) We propose an efficient and effective neural architecture design method based on stacking blocks, named BlockNAD, for VDR appearance quality inspections.

- (2) Using BlockNAD, two blocks have been designed and applied to VDR defect detection. Each block consists of a compressed subnet and a multiscale subnet. The compressed subnet adjusts the number of feature-mapping channels of the input block to maintain the size of the block parameters. The multiscale subnet contains three branches to extract and merge features of different scales. They have a mean average precision (mAP) of approximately 99.9% on the VDR test set and an average inspection time per sample of approximately 3 ms, which meets the requirements of online real-time inspection.

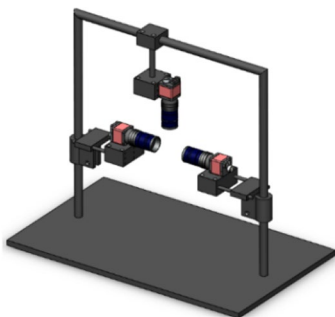
The remainder of the paper is organized as follows. In Sect. 2, the VDR image acquisition process and the dataset are introduced; in Sect. 3, the proposed method is described in detail; in Sect. 4, the algorithm evaluation criteria are introduced, and the experimental results are presented; and the final section contains a summary.

## 2 Materials

This section describes the details of our VDR dataset. Firstly, VDR images were collected using a designed 3-angle camera. Then, the collected images were subjected to data augmentation operations such as rotation and brightness adjustment. Finally, a 12-class VDR dataset was produced.

### 2.1 VDR image acquisition

VDR images were acquired using 0.3 M-pixel industrial cameras and the image acquisition device shown in Fig. 2; it consists of three imaging devices, separately capturing images of the VDR from three angles (front, back, and side). Each imaging device consists of a camera, lens, and coaxial light source. The coaxial light source avoids reflection from the smooth VDR surface, enabling the acquisition of clean VDR 640×480-pixel color images.

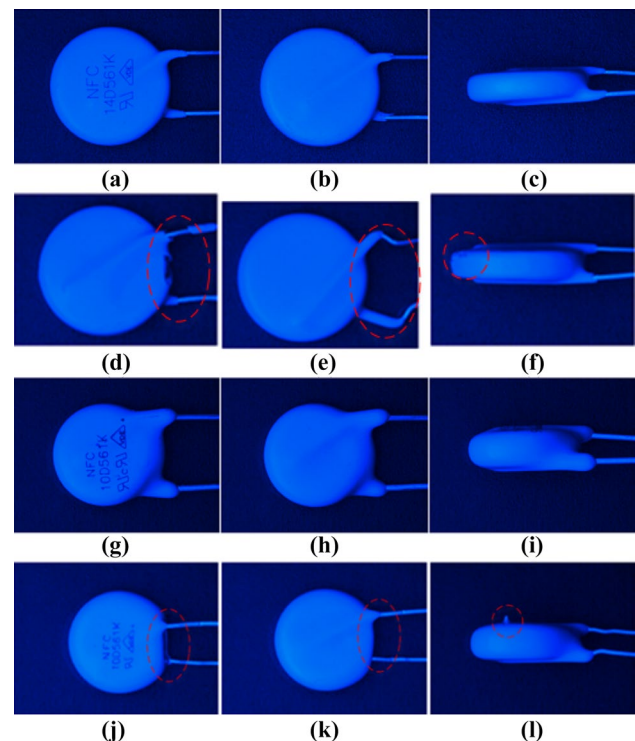


**Fig. 2** Image acquisition device that takes VDR pictures from three angles (front, back, and side)

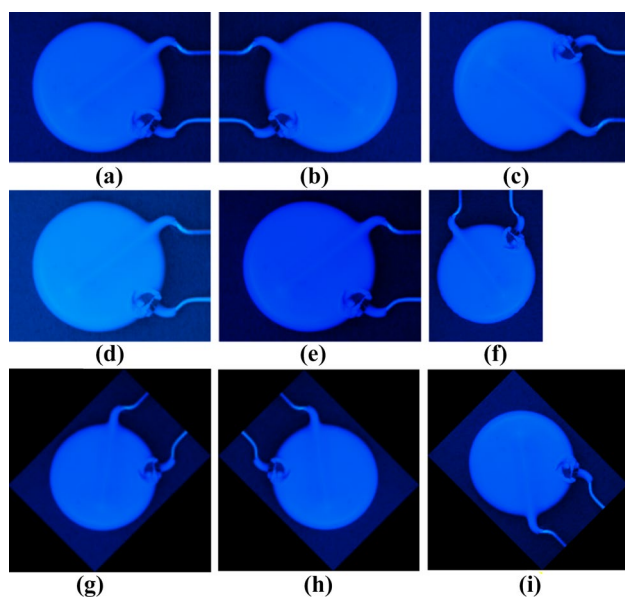
The acquired VDR images were then divided into two types according to the VDR body diameter: R14 (body diameter: 14 mm) and R10 (body diameter: 10 mm). For each VDR sample, three images from three angles (front, back, and side) were acquired, as shown in Fig. 3.

### 2.2 Data augmentation

To train a more reliable CNN model, we performed a series of data augmentation operations, including rotation, flipping, brightening, and dimming, on the acquired raw VDR images. First, the raw VDR image was augmented through rotations (45° and 90°); second, its brightness was adjusted through a gamma correction with gamma values of 0.6 and 1.4; third, the raw VDR image and the adjusted image were augmented through flipping. Some of the results of the data augmentation are shown in Fig. 4.



**Fig. 3** Examples of VDR images. Nondefective R14 sample images in **a** front, **b** back, and **c** side views; defective R14 sample images in **d** front view, showing the surface and damaged pins, **e** back view, showing bent pins, and **f** side view, showing damaged pins. Nondefective R10 sample images in **g** front, **h** back, and **i** side views; defective R10 sample images in **j** front view, showing missing pin wrap, **k** back view, showing an insufficient pin wrap, and **l** side view, showing a protruding surface. The positions of the defects are indicated with red dashed circles



**Fig. 4** Results of data augmentation. **a** original image; **b** vertical flipping of the original image; **c** horizontal flipping of the original image; **d** brightening of the original image (gamma value: 0.6); **e** dimming of the original image (gamma value: 1.4); **f** 90° and **g** 45° rotations of the original image; **h** vertical flipping of (g) and **i** horizontal flipping of (g)

### 2.3 VDR dataset

We acquired a total of 1344 images from three angles (front, back, and side) of 448 VDR samples collected from a production line using the image acquisition device described above. These images were then subjected to data augmentation to generate 8058 images that composed the final VDR dataset, which included 3894 R14 samples (2214 nondefective and 1680 defective samples) and 4164 R10 samples (2160 nondefective and 2004 defective samples). The VDR samples of the two models (R14 and R10) were divided into two categories (nondefective and defective samples), and images from the three angles (front, back, and side) were divided into three categories, for a total of 12 categories, as shown in Table 1. The samples in each category were divided into training, validating, and test sets in a ratio of approximately 7:1:2. All images were scaled to a size of  $64 \times 64$ .

## 3 Method

Next, we constructed a CNN suitable for VDR appearance defect identification. The structures of conventional CNNs generally include several convolution layers and several pooling layers that alternately connect with each other and one or several fully connected layers. Considerable time was

**Table 1** VDR dataset

Category	Label	# Train	# Validate	# Test
R14 D-S	0	420	60	120
R14 D-B	1	388	54	110
R14 D-F	2	373	50	105
R14 ND-S	3	517	74	147
R14 ND-B	4	517	74	147
R14 ND-F	5	517	74	147
R10 D-S	6	469	66	133
R10 D-B	7	469	66	133
R10 D-F	8	469	66	133
R10 ND-S	9	504	72	144
R10 ND-B	10	504	72	144
R10 ND-F	11	504	72	144

D and ND represent defective and nondefective, and S, B, F represent side, back, and front, respectively

spent in choosing the layers, layer-to-layer connections, and parameters. To design a proper neural architecture more efficiently and effectively, we used a stacking-block strategy for neural architecture designing we call BlockNAD.

### 3.1 Block-stacking-based neural architecture design

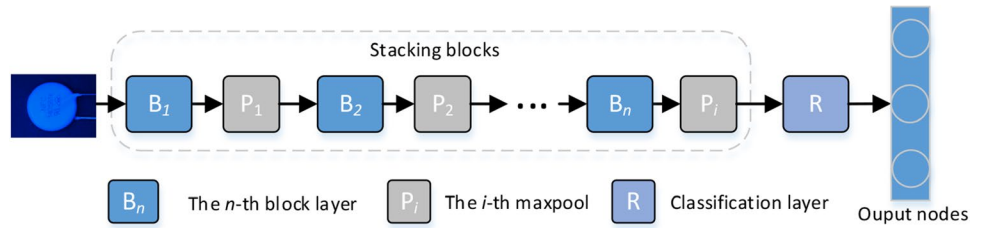
The proposed neural architecture is designed based on block stacking (as shown in Fig. 5). A block is a reusable sub-network that can be stacked  $K$  times in a network. Each block is followed by a maximum pooling layer (maxpool), where down-sampling is performed to allow the main features to be retained while reducing the number of parameters. To improve the accuracy of classification, the number of channels of all convolutional layers in the next block is set to be twice that in the previous block.

A classification layer is connected to the last block to output the classification results. The number of parameters of the fully connected (FC) layer is often too high, which causes various problems such as slow network training and the tendency of overfitting during training; therefore, a global average pooling layer (GAP) [31, 32] or GAP combined with the FC [33] is used to replace the traditional FC layer. The proposed network was composed of three types of components: block layers, pooling layers and a classification layer.

Once the CNN is established, network training and validating can be performed. Based on this method, we started the search from a network with one block and continued to increase the number of the blocks until we found a satisfactory network model, which has the advantage of reducing the search space of the neural architecture.



**Fig. 5** The proposed neural architecture is built based on stacked blocks with three types of components (block layers, pooling layers, and a classification layer). Each block layer is followed by a maxpool



### 3.2 Building blocks

Two types of blocks are constructed. Each block consists of a compressed subnet and a multiscale subnet, including several convolutional layers (conv) and an average pooling layer (avgpool). The structure of Blocks is shown in Fig. 6.

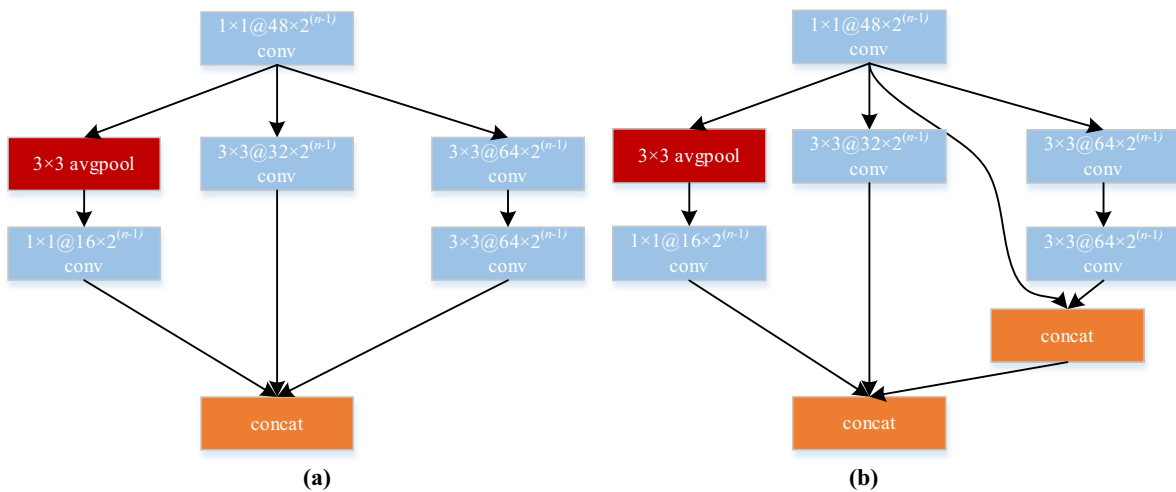
Figure 6a shows the structure of Block-A, which consists of a compressed subnet and a multiscale subnet. The compressed subnet has a  $1 \times 1$  convolutional layer whose function is to adjust the number of feature-mapping channels of the input block to maintain the size of block parameters. After adjustment, the outputs are sent to the three branches of the multiscale subnet separately. The first branch has an avgpool to first obtain the low-frequency features and further uses a  $1 \times 1$  convolutional layer for compression; the second branch is a  $3 \times 3$  convolutional layer, and the third branch has two adjacent  $3 \times 3$  convolutional layers, which is equivalent to a  $5 \times 5$  convolutional layer [18]. All  $3 \times 3$  convolutional layers use the Rectified Linear Unit (ReLU) activation function. Finally, the outputs of three branches are fused through the concat operator as the output of the block.

Figure 6b shows the structure of Block-B, which is similar to that of Block-A. The only difference is that the output

of third branch and the output of the compressed subnet are fused first through a concat operator. That is, the shallow and deep features are fused first as one output; then, this output is fused together with the outputs of first and second branches through the second concat operator as the output of Block-B.

### 4 Experiments

The experiments were based on CNNs constructed with caffe 1.0 [34] to perform the training and testing. The experiments were implemented on a PC (Intel Core i5 CPU, 8 GB DDR4, and 1050Ti NVIDIA GPU) with Windows 10. The stochastic gradient descent (SGD) method [35] was adopted in optimization during network training in which the learning rate, momentum, and maximum number of iterations were set to 0.001, 0.9, and 10,000, respectively. To evaluate the performance of the proposed method, we conducted experimental comparisons from three aspects. First, through the comparison of GAP and GAP + FC, the classification layer of BlockNAD was determined; then, based on the BlockNAD and Blocks, two types of CNNs were constructed. In the training and validating sets, the classification performances of CNNs



**Fig. 6** The structures of two types of the proposed block. **a** Block-A and **b** Block-B. They consist of a compressed subnet and a multiscale subnet, including several convolutional layers and avgpool. For example,  $1 \times 1 @ 48 \times 2^{(n-1)}$  conv represents the convolutional layer with

a convolution kernel size of  $1 \times 1$  and  $48 \times 2^{(n-1)}$  channels;  $n$  is the index of stacking blocks,  $n = 1, 2, 3, \dots$ ; and  $3 \times 3$  avg pool represents the average pooling layer with a pooling window of  $3 \times 3$

with different stacking blocks were compared to find the optimal CNN. Finally, the optimal CNN was compared with the state-of-the-art methods on the test set.

### 4.1 Evaluation indicators

We used average precision (AP) and mean average precision (mAP) to evaluate the performance of the proposed algorithm and compare it with other algorithms. For each class, we first calculated its AP and then the mAP of all classes, using the following equations:

$$AP_j = \frac{1}{R} \sum_{i=1}^M I_i * \frac{R_i}{i} \tag{1}$$

$$mAP = \frac{1}{N} \sum_j AP_j \tag{2}$$

In these equations, the category to be calculated was regarded as a positive sample and the remaining categories were regarded as negative samples;  $R$  is the number of all positive samples in the test set, and

$M$  is the total number of samples in the test set. When the  $i$ th sample is a positive sample,  $I_i = 1$ ; otherwise,  $I_i = 0$ .  $R_i$  represents the number of positive samples in the first  $i$  samples,  $N$  represents the number of categories, and  $AP_j$  represents the average precision of the  $j$ th category.

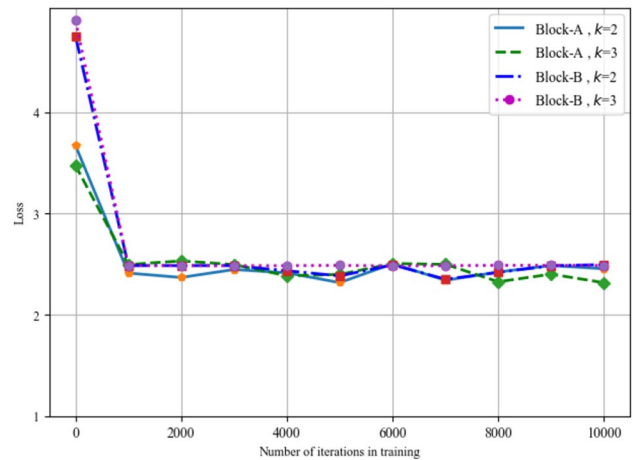
### 4.2 Classification layer

First, through the comparison of GAP and GAP+FC, the classification layer of BlockNAD was determined. Based on Block-A and Block-B, 4 CNNs were constructed using BlockNAD with  $K=2, 3$  and were trained 10,000 times on the training set. Figures 7 and 8 show the training times—error curves of the 4 CNNs using GAP or GAP+FC as the classification layer. As can be observed from the figures, when GAP was adopted as the classification layer, the Loss of the CNNs can be reduced to only approximately 2.4, while when GAP+FC was adopted, the Loss dropped to close to 0 rapidly.

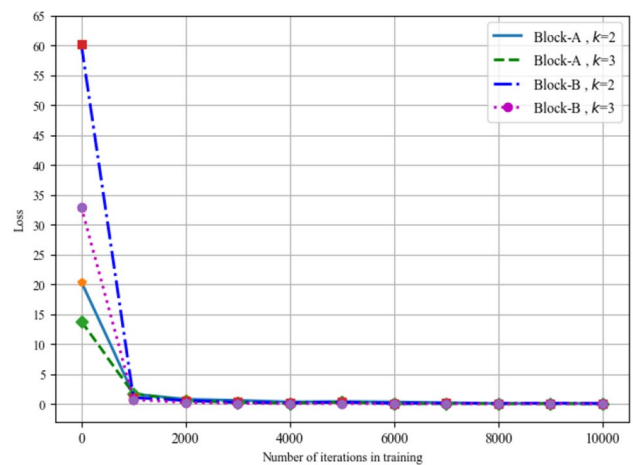
Based on the above analysis, convergence of CNNs with GAP+FC was better than that of CNNs with GAP, therefore, GAP+FC was used in the classification layer of BlockNAD.

### 4.3 Block layers

Based on the proposed CNN network structures (Fig. 5) and BlockNAD, multiple CNNs with different stacked blocks were constructed and evaluated. After experimental comparison and analysis, we chose the representative structures



**Fig. 7** The training times - error curve of the CNNs using GAP as the classification layer. For example, for Block-A,  $K=2$  means that this CNN has two stacked Block-A's



**Fig. 8** The training times - error curve of the CNNs using GAP+FC as the classification layer

with high performance for VDR defect identification. The CNNs constructed based on Block-A and B were termed BlockNAD-A and B, respectively.

Figure 9 shows the classification accuracy of the two CNNs with different numbers of stacking Blocks. As can be observed from the figure, when  $K=1-5$ , the mAP of the two CNNs generally showed an upward trend, and when  $K=3, 4$  and  $5$ , both mAPs approached 100%. When  $K=1, 2$ , the mAPs of BlockNAD-B were higher than those of BlockNAD-A, indicating the fusion of shallow and deep features was more effective for VDR defect detection.

Figure 10 shows the size of parameters of the two CNNs when  $K=1-5$ . It can be observed from Fig. 10 that the number of parameters of the two CNNs varied little at different  $K$  values. When  $K=1-4$ , the number of parameters of the CNNs was less than 10 M and increased slowly. When  $K=5$ ,

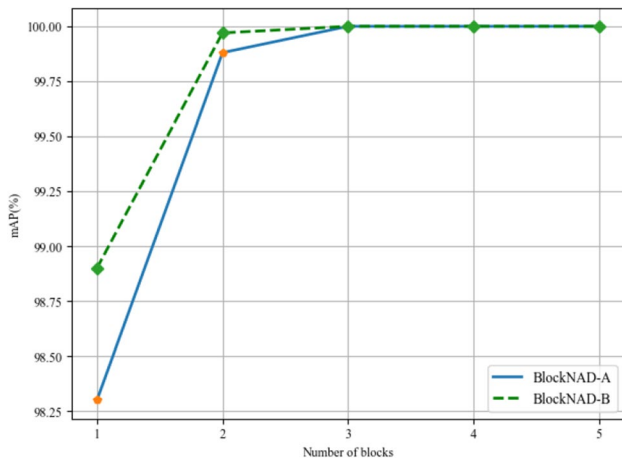


Fig. 9 The relationship between mAP and  $K$ , the number of stacking blocks

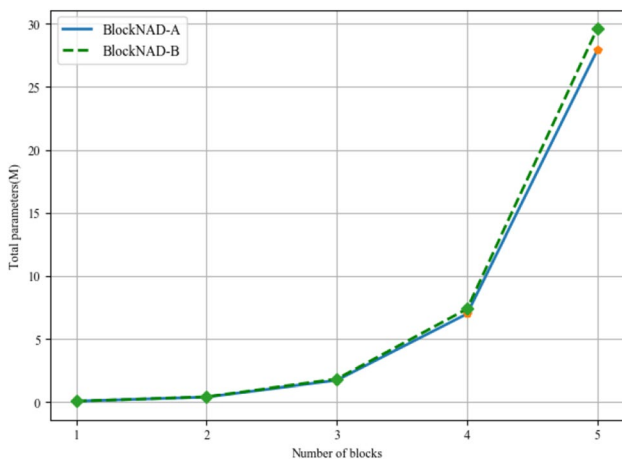


Fig. 10 The relationship between the size of the parameters and  $K$ , the number of stacking blocks

the number of parameters started to increase rapidly, since the number of channels increases as  $2^{(n-1)}$ , where the  $n$  is the index of stacking blocks.

Based on the comprehensive analysis of Figs. 9 and 10, either BlockNAD-A-3 or BlockNAD-B-3 can be used as a CNN for defect detection after considering the detection accuracy and efficiency.

#### 4.4 Experimental results

We compared the accuracies of the proposed networks in recognizing VDR defects with VGG-16 [18], Resnet-18 [20], DBCC [36], and an 11-layer CNN [23]. Of these, VGG-16 and Resnet-18 are classic CNNs that perform well in large-scale classification tasks, such as ImageNet, and DBCC and the 11-layer CNN are manually constructed CNNs for surface-defect detection.

Table 2 shows the performance of the compared methods on the 12-category identification. The mAP, parameter size, and Model size of proposed networks BlockNAD-B-3 and BlockNAD-A-3 were both in the top 2, and the average detection time was only half that of Resnet-18, whose mAP ranked in the 3rd place. The mAP of BlockNAD-B-3 was higher than that of BlockNAD-A-3; however, the parameter size and Model size were slightly larger, and the detection time is slightly longer (~0.1 ms). The parameter size and Model size of VGG-16 were the largest and were more than 20 times of those of the proposed CNNs. The detection time of DBCC was the shortest; however, its parameter size and Model size were slightly larger than those of the proposed CNNs. The detection time of the 11-layer CNN was shorter than that of the proposed CNNs, however, its parameter size and Model size were 5 times of those of the proposed CNNs.

In addition to detection time, the results of the above comparison experiments demonstrate that the proposed networks had the best overall performance. In addition, BlockNAD-based CNNs are simpler and more efficient than the CNNs constructed layer by layer manually. Additionally, the average detection time of approximately 3 ms can meet the demand of real-time detection. Therefore, in practice, the proposed networks can be used as models for VDR defect detection.

### 5 Conclusions

In this study, we proposed an automatic inspection method for VDR appearance defects based on CNN, which achieved VDR defect identification based on VDR images from three angles (front, back, and side) using a simple and efficient neural architecture designing method called BlockNAD. High classification accuracy and efficiency in the 12-category classification of VDR defects were achieved that can meet the requirements of online real-time inspection. In the future, we will develop the classification and positioning methods for VDR defects in additional defect categories.

Table 2 Classification performance comparison

Method	mAP (%)	Paras (M)	Model size (MB)	Time (ms)
VGG-16 [18]	99.38	400	152	6.5
Resnet-18 [20]	99.82	110	42.7	6.6
DBCC [36]	98.79	20	7.62	1.5
11-layer CNN [23]	99.72	97	37.1	1.8
BlockNAD-A-3	99.84	17	6.57	3.2
BlockNAD-B-3	99.92	18	6.66	3.3

**Acknowledgements** This research was supported in part by the National Natural Science Foundation of China (61941202), the Guangxi Natural Science Foundation (2018GXNSFBA281081), and the Guangxi Key Laboratory Fund of Embedded Technology and Intelligent System (2019-02-01, 2019-01-08).

## References

- Ng, H.F.: Automatic thresholding for defect detection. *Pattern Recognit. Lett.* **27**, 1644–1649 (2004)
- Wu, H., Zhang, X., Xie, H., Kuang, Y., Ouyang, G.: Classification of solder joint using feature selection based on bayes and support vector machine. *IEEE Trans. Compon. Packag. Manuf. Technol.* **3**, 516–522 (2013)
- Wang, Q., Li, D., Zhang, W., Cao, D., Chen, H.: Unsupervised defect detection of flexible printed circuit board gold surfaces based on wavelet packet frame. In: 2010 2nd International Conference on Industrial and Information Systems, Dalian, China, Jul. 10–11 2010, pp. 324–327. *IEEE*
- Cai, N., Lin, J., Ye, Q., Wang, H., Weng, S., Ling, B.W.K.: A new IC solder joint inspection method for an automatic optical inspection system based on an improved visual background extraction algorithm. *IEEE Trans. Compon. Packag. Manuf. Technol.* **6**, 161–172 (2016)
- Chondronasios, A., Popov, I., Jordanov, I.: Feature selection for surface defect classification of extruded aluminum profiles. *Int. J. Adv. Manuf. Technol.* **83**, 33–41 (2016)
- Li, W.C., Tsai, D.M.: Wavelet-based defect detection in solar wafer images with inhomogeneous texture. *Pattern Recognit.* **45**, 742–756 (2012)
- Zhang, X., Ding, Y., Duan, D., Fang, G., Xu, L., Shi, A.: Surface defects inspection of copper strips based on vision bionics. *J. Image Graph.* **16**, 594–599 (2011)
- Ravikumar, S., Ramachandran, K.I., Sugumaran, V.: Machine learning approach for automated visual inspection of machine components. *Expert Syst. Appl.* **38**, 3260–3266 (2011)
- Hinton, G.E., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. *Science* **313**, 504–507 (2006)
- Chang, L., Deng, X., Zhou, M., Wu, Z., Yuan, Y., Yang, S., Wang, H.: Convolutional neural networks in image understanding. *Acta Autom. Sin.* **42**, 1300–1312 (2016)
- LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L.: Handwritten digit recognition with a back-propagation network. In: *Advances in Neural Information Processing Systems 2*, San Francisco, CA, USA 1989, pp. 396–404. Morgan Kaufmann Publishers Inc
- Bi, L., Feng, D., Kim, J.: Dual-path adversarial learning for fully convolutional network (FCN)-based medical image segmentation. *Vis. Comput.* **34**, 1043–1052 (2018)
- Dong, Z., Wu, Y., Pei, M., Jia, Y.: Vehicle type classification using a semisupervised convolutional neural network. *IEEE Trans. Intell. Transp. Syst.* **16**, 2247–2256 (2015)
- Cao, Z., Mu, S., Dong, M.: Two-attribute e-commerce image classification based on a convolutional neural network. *Vis. Comput.* (2019). <https://doi.org/10.1007/s00371-019-01763-x>
- Park, J.-K., Kang, D.-J.: Unified convolutional neural network for direct facial keypoints detection. *Vis. Comput.* **35**, 1615–1626 (2019)
- Soukup, D., Huber-Mörk, R.: Convolutional neural networks for steel surface defect detection from photometric stereo images. In: *Bebis, G., Boyle, R., Parvin, B., Koracin, D., McMahan, R., Jerald, J., Zhang, H., Drucker, S.M., Kambhamettu, C., Choubassi, M.E., Deng, Z., Carlson, M. (eds.) International Symposium on Visual Computing*, Cham, Switzerland 2014, pp. 668–677. Springer
- Tao, X., Wang, Z., Zhang, Z., Zhang, D., Xu, D., Gong, X., Zhang, L.: Wire defect recognition of spring-wire socket using multitask convolutional neural networks. *IEEE Trans. Compon. Packag. Manuf. Technol.* **8**, 689–698 (2018)
- Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *Comput. Sci.*, [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)
- Feng, C., Liu, M.Y., Kao, C.C., Lee, T.Y.: Deep active learning for civil infrastructure defect detection and classification. In: *ASCE International Workshop on Computing in Civil Engineering 2017*, Seattle Washington, United States, May 1 2017, pp. 298–306. Mitsubishi Electric Research Laboratories, Inc
- He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 27–30 2016, pp. 770–778. *IEEE*
- Settles, B.: *Active Learning Literature Survey*. Computer Sciences Technical Report 1648, University of Wisconsin, Madison, Wisconsin (2009)
- Huang, Y., Qiu, C., Yuan, K.: Surface defect saliency of magnetic tile. *Vis. Comput.* **36**, 85–96 (2020)
- Wang, T., Chen, Y., Qiao, M., Snoussi, H.: A fast and robust convolutional neural network-based defect detection model in product quality control. *Int. J. Adv. Manuf. Technol.* **94**, 3465–3471 (2018)
- Faghih-Roohi, S., Hajizadeh, S., Nunez, A., Babuska, R., De Schutter, B.: Deep convolutional neural networks for detection of rail surface defects. In: *2016 International Joint Conference on Neural Networks (IJCNN)*, Vancouver, BC, Canada, Jul. 24–29 2016, pp. 2584–2589. *IEEE*
- Chen, J., Liu, Z., Wang, H., Nunez, A., Han, Z.: Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network. *IEEE Trans. Instrum. Meas.* **67**, 257–269 (2017)
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: *Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision—ECCV 2016*, Cham, 2016//2016, pp. 21–37. Springer International Publishing
- Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 27–30 2015, pp. 779–788. *IEEE*
- Tao, X., Zhang, D., Ma, W., Liu, X., Xu, D.: Automatic metallic surface defect detection and recognition with convolutional neural networks. *Appl. Sci.* **8**, 1575 (2018)
- Yu, Z., Wu, X., Gu, X.: Fully convolutional networks for surface defect inspection in industrial environment. In: *International Conference on Computer Vision Systems 2017*, pp. 417–426. Springer
- Cha, Y.J., Choi, W., Büyüköztürk, O.: Deep learning-based crack damage detection using convolutional neural networks. *Comput. Aided Civil Infrastruct. Eng.* **32**, 361–378 (2017)
- Lin, M., Chen, Q., Yan, S., (2013). Network in network. *arXiv e-prints* (2013)
- Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.: SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv preprint arXiv:1602.07360* (2016)
- Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: An extremely efficient convolutional neural network for mobile devices. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT 2018, pp. 6848–6856. *IEEE*
- Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture



for fast feature embedding. In: 22nd ACM International Conference on Multimedia, New York, NY, USA, Nov. 03–07 2014, pp. 675–678. ACM

35. Bottou, L.: Stochastic gradient descent tricks. In: Neural networks: Tricks of the trade, pp. 421–436. Springer, Berlin (2012)
36. Fu, L.L., Fei, M.W., Li, L., Cheng, L.: Research on detection algorithm for bridge cracks based on deep learning. *Acta Autom. Sin.* **45**(9), 1727–1742 (2019)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Tiejun Yang** received a B.S. degree in computer science and technology and an M.S. degree in computer software and theory from Central South University, China, in 2002 and 2005, respectively. He received a Ph.D. in computer application technology from the South China University of Technology, China, in 2008. He is current a professor at the School of Information Science and Engineering, Guilin University of Technology, China. His research interests are computer vision and deep learning.



**Tianshu Zhang** is currently pursuing a Master's degree at the School of Information Science and Engineering, Guilin University of Technology, China. His research interests include computer vision and deep learning.



**Lin Huang** received a M.S. degree in signal and information processing from Shantou University, China, in 2007. She is currently pursuing a Ph.D. at the Department of Electrical Engineering, Xidian University, China. Her research interests include computer vision and deep learning .