A Multi-class Defect Detection Network for Cooker Surface

Wenbiao Zhou^{a,b}, Jiao Zhang^b, Wenbin Liu^b, Pengfei Wan^b, Jiakui Liu^b, Ke Zhou^b

^aSchool of Information and Electronics, Beijing Institute of Technology,
5 South Zhongguancun Street, Haidian District, Beijing, Beijing, 100000, China

^bASD Robotic, No. 969, Changjiang Road, Huangpu District, Shanghai, Shanghai, 200000, China

*Corresponding Author: zhouwenbiao@bit.edu.cn

Abstract

In the actual production, a large amount of workers are needed for quality inspection, they inspect various defects by subjective eyesight and experience. This paper proposes a multi-class defect detection network for cookers(CDDNet). It also designs a training procedure to preserve end-to-end model accuracy post quantization. It applies CDDNet training and testing the ability of multi-classification and transfer learning, and show that multi-classification and transfer learning can be successfully applied using image data from an entirely different domain. The database is obtained from many days of automated camera recordings. As a result, the proposed quantization scheme improves the tradeoff between accuracy and device latency.

Keywords: defects detection, CDDNet, multi-classification, transfer learning.

1. Introduction

With the increasing demand of automation technology, more and more cooker assembly line need automatic contactless defect inspection. Most factory attach great importance to surface defect inspection of products, some defects are shown in the Fig. 1, and now such inspection tasks are done mostly by human workers. However, the human workers only can work for a limited time without experiencing fatigue, and with rising labor costs, there is an increasing demand for automating such defect inspection tasks.

Traditional defect detection methods used in manufacturing, such as edge detection⁽²⁷⁾, segmentation⁽²³⁾, texture analysis⁽²⁸⁾ method and line detection⁽²⁶⁾, only can treat some defects that have relatively regular shape and result in a low accuracy and low recall rate for non-regular

shape. And we usually use in conjunction with each other method, such as edge detection with segmentation⁽²⁵⁾.

Fig. 1. The typical defects on cookers' surface by some defect detection results.

The feature learned by deep neural networks makes up for the disadvantages of artificial feature. Deep neural networks have shown state-of-the-art performance in classification and detection tasks^(10,16,20,24). And Seunghyeon Kim has proved that transfer learning can be successfully employed while lack of data⁽⁸⁾. By feeding raw data into a neural network with various linear or non-linear operations, we are in effect transforming these raw data into a feature space. Since this transformation is differentiable, one can use a gradient-descent type of optimization techniques to find a good transformation from a random one. With such frameworks, we can use a deep neural network to model very complex input-output relationship.

In this paper, we propose CDDNet for cooker surface defect detection. Our network combines SSDNet⁽¹³⁾ and transfer learning. And the results show CDDNet performs excellently in kinds of defects, especially low resolution defect. Here are some examples of typical defects on

cookers' surface by some defect detection results in Fig. 1. Most of defective samples can achieve higher confidence, that is provident CDD network learnable on cooker surface defect detection. The contributions of our paper are summarized as follows:

- (a) We confirm the behavior of multi-class method through the detection accuracy. We classify and label training data according to their performance.
- (b) We propose a novel training method for domain adaptation that transfers learning model weights between difficult data.
- (c) We extensively evaluate our method on various tasks: defect finding, defect classification.

The remainder of the paper is structured as follows. In Section 2 we provide a brief overview of related work, followed by the description of the proposed architecture for defect detection in Section 3. The experimental setup is described in Section 4 while results are presented in Section 5. We conclude with the discussion in Section 6.

2. Related work

Classical approaches in defect detection on object surfaces follow more or less the same paradigm, such as SVM⁽²⁹⁾, LDA, PCA⁽¹⁾, Hough transform⁽¹²⁾, decision trees or KNN and so on. Classical approaches train on feature descriptors obtained from preprocessed images. Here, the preprocessing stage is crucial as it ensures that the problem is well-conditioned for the process of hand-engineering suitable features. Commonly defect detection approaches include: (i) Filtering approaches based on Wavelet transform⁽⁴⁾, independent component analysis⁽²¹⁾ and Gabor filter⁽³²⁾; (ii) Structural approaches based on morphological operators⁽¹⁵⁾ and edge detection⁽¹⁸⁾; (iii) Model based approaches such as the Hidden Markov model⁽¹⁷⁾ and auto-regressive models⁽²⁾; (iv) Statistical approaches based on histograms⁽¹⁹⁾, co-occurrence matrices⁽⁹⁾ and auto-correlation⁽²⁸⁾. Deep learning highly contrast these approaches by performing automated feature learning instead of hand-designing suitable and at times sub-optimal features.

Recently, advanced machine learning methods such as autoencoders and back propagation neural networks have been proposed for the intelligent diagnosis of mechanical systems in industry and research^(6,14) as these use data to construct models that can detect different conditions autonomously and are capable of learning valuable features from very large sets of raw data⁽³⁾. Others such as deep neural

networks employ a hierarchical structure composed of multiple neural layers that, in a layer-by-layer fashion, extract intricate structures from large volumes of raw data and learn useful features across multiple levels of abstraction⁽¹⁰⁾. In particular, convolutional neural networks (CNN)⁽¹¹⁾ are one of the main models and have been very successful when applied to learn features autonomously by exploiting the spatial structure in raw data. For instance, with the aim of automatically learning features to detect faults on gearboxes, the work reported in⁽⁷⁾ explored different CNN configurations applied to single-axis vibration data in the time, frequency and time-frequency domains. The results demonstrated that a CNN model is capable of ingesting raw data to undergo feature extraction, selection and classification, thus forming a feature learning system that outperforms traditional ML methods. Convolutional neural networks have proven superior in tasks where hand-designing features proves a difficult task. Early work on utilizing a CNN for surface defect detection can be found in⁽³³⁾. The motivation arises from the aforementioned difficulty, where even domain specialists struggle to devise accurate rules based on geometrical and shape features for certain defects. The classification error is reduced by half over the classical approach with a classifier trained on feature descriptors, which included a Multi Layer Perceptron (MLP) and SVM with RBF classifiers trained on features obtained via HOG, PHOG, rotation invariant measure of local variance, and Local Binary Patterns (LBP, LBP-Fourier). More specifically in cooker manufacturing, the defect detection problem has not been studied previously with any DL strategies yet, based on our literature survey.

3. Proposed method

With a generally applicable neural network, for domain-specific problems, like the defect detection problem in this paper, what we need to do is to: (i) prepare the data; (ii) select an existing deep neural network architecture; (iii) design a suitable loss function for optimization; (iv) tune hyperparameters for training to achieve a best performance. In fact, a popular type of deep learning methods, the convolutional neural network (CNN), was found to obtain good performance in this defect detection task^(22,24,31).

3.1 Datasets

For the source dataset we use the COCO dataset. This dataset contains 165,482 training images, 81,208 valing images, and 81,434 testing images of 80 classes. The original

cooker data are 749 images with size 2448×2048 .

Fig. 2. The experiment dataset we use to train CDD.

Fig. 3. A comparison between two single shot detection models: SSD⁽¹³⁾ and CDD.

One constraint on using a pre-trained network is that the size of the input image should be equal to the size of the input layer of the network (in our case, 300×300). Therefore, we extract patches from training images in the target dataset by using windows of size 300×300 , just as shown in Fig. 2. Additionally, we augment both defective and non-defective training sets by adding cropped patches. Finally, for the cooker dataset we use to detect defect consists 1,409 labeled defect images and 31,881 non-defect images. From the cooker dataset, the small cropped images are uniformly selected to generate training and validation sets. We selected 993 training images, 285 valing images and 131 testing images among 1409 defects images. Also the defects are labeled and fall into 5 classes: dirty, scratch, oil, pit and wire-drawing.

3.2 CNN Architecture

This section summarizes the entire process of our

framework. The overall process is illustrated in Fig. 3. The CDD approach is based on a SSD network that produces a multi-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections. The early network layers are based on a standard architecture used for high quality image classification (truncated before any classification layers), which we will call the base network. We then add shallow convolutional feature layers to the end of the truncated base network. These layers decrease in size progressively and allow predictions of detections at multiple mini-scales. We use a CDD network for both source and target networks. The CDD network is based on SSD that consists of 21 convolutional layers and three fully-connected layers. Also, there is a max-pooling layer after every two or three convolutional layers. The network has an input layer of size $300 \times 300 \times 3$. The exact configuration of the SSD network is described in⁽¹³⁾. The

Fig. 4. Statistics suggest that a different distribution the resolution of defects.

most important difference between the SSD and CDD. According to statistics the defects resolution, we notice the defects with rather small size as shown in Fig. 4, especially the resolution of pit are mainly in low size 38×38 and the activation area in feature map is much lower, as Fig. 8. and the feature maps on deeper convolution layers may not be helpful enough because of lower resolution. We decide to add shallow feature map with slightly larger size to the prediction layer. And the number of output layer depends on the number of classes in the dataset, which in our case consists of 80 output nodes for the COCO dataset and 5 output nodes for the cooker dataset.

3.3 Transferring learning in training stage

On the one hand, once the CDD network is trained on the COCO dataset, its weights are transferred to initialize the CDD network. Inspired by ⁽³⁰⁾, we transfer the CDD network weights up to convolutional layers on COCO dataset to cooker dataset; that is, all convolutional layers of the CDD on cooker dataset are initialized by the transferred weights, while the remaining fully-connected layers are randomly initialized. Since the weights in the fully-connected layers of the CDD are specific to the COCO dataset labels, transferring the whole layers of the CDD can degrade generalization performance on the defect detection. After transferring the weights, we train the entire CDD network by using a mini-batch Stochastic Gradient Descent (SGD) optimizer.

On the other hand, we optimize trained model by transferring learning in iteration of training data. With the lack of defect datas, we proposed an iteration training method, that is we make different defects labeled as to the added training data. Also, we label non-defective samples as non-defective class. So the difference of defects and non-defects can be learned mostly improved. We give the transferring learning solution in Fig. 5.

networks is the predictions in much shallow feature layer.

3.4 Testing the CDD

For each test image, we extract numbers slightly patches by windows of size 300×300 . Each cropped patch from the test image is forward propagated through the target network to yield the prediction result. The most frequently occurring pattern is selected as the pattern for the test image. Also, if there exists at least one patch predicted as defective, the test image is labeled as defective. An example of this process is illustrated in Fig. 6. Since the majority of patches are predicted as ‘Defect 2’, the pattern of the test image is labeled as ‘Defect 2.’ Also, since there exists one patch predicted as ‘ABNORMAL,’ the test image is labeled as ‘Defective.’ Therefore, the final prediction result of this test image is a ‘Defect 2 defective’ image.

Fig. 5. Our method of iteration transferring training.

Fig. 6. An example of obtaining a prediction of the test image.

Fig. 7. Here we give the illustration of experimental facility.

4. Experiment Results

4.1 Experimental setup

The experimental facility is shown as in Fig. 7. We show the main size and units installation site. Also we give the the physical map. And it's equipped with an acrylic light box, light source and five cameras. The height of the installation is 100 centimeters as reference surface. The light source on top surface is the ten u-shaped lights. The light source on four side surfaces is the ten v-shaped lights and is distributed in the space area of 600*700mm near the top. Here we need clear imaging to get enough information about the defects, the difficulty is that the defect not only on the plane, but also curve region. So we use the shaped lights combined diffuser to get high quality pics for defects. The industrial cameras we use are HikVison MVL-MF1620M-5MP lens with 5 mega pixel, focal length of 16mm and the photo taken has the resolution 2448*2048. And here we explain about our experimental facility. Firstly, the 5 arches with 50mm DIA is the position where the camera is mounted, and the four cameras on the side are required to be adjustable in the range of 60-80 degrees. Secondly, the whole camera

cannot be inside the box because it casts the shadow on the cooker we want to detect, only part of the lens is allowed inside the acrylic light box. Thirdly, we design a tray that can be moved up and down in the bottomless box to put the object.

The implementation is based on the framework in Keras 2.2.2(5). To decrease the unwanted variation due to different lighting and reflector texture conditions, we apply a simple normalization to all samples. The learning rate is initially set to 10^{-4} with a decay factor of 10^{-5} to help avoid overfitting to the training data. From the results acquired in our parameter adjustment runs, we set the mini-batch size to 22 for CDD. Then, we train each network over 1500 epochs. For each test we shuffle all the existing samples and divide each class into 5 sets. For 5 rounds of crossvalidation, we test each time on one out of 5 sets and use the rest for training. For the testing, we randomly sample the normal class to 250 test samples per round. This is done due to the huge imbalance of the class sizes, which can severely bias the test results if all normal test samples are evaluated. In order to convert the multi-class classification results to the binary classification of normal samples versus anomalies, we simply regard all the non-normal classes as one and compute

Fig. 8. We illustrate the feature maps and predictor layers of our CDDNet with other nets.

Fig. 9. We evaluate our CDDNet with other nets.

the numbers of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The binary classification accuracy: $(TP + TN) / (TP + TN + FP + FN)$ and F1-score: $2TP / (2TP + FP + FN)$ are then defined based on the reduced classification matrices.

4.2 Classification results

We report two types of results from the experiments. Initially the networks are trained to classify the data into 5 classes.

We compare the confusion defects of the classification results trained on the CDD with tranfering learning. The defect classification results trained on the CDD are presented in Table 1 and Table 2. The cost time for CDD is 0.125s~0.142s. The rows of the matrices in Table 1 correspond to the network we trained on and the columns correspond to the defects predicted miss rate among all defective samples. After we transferred weights of CDD trained on COCO dataset by subsampling, we train CDD on cooker training

		SSD	SSD+Conv3_3	SDD+Conv3_3+TL(CDD)
BEFORE	score $\geq$ 50	25.73	22.56	15.78
	score $\geq$ 90	31.8	27.42	18.91
AFTER	score $\geq$ 50	16.69	13.86	8.31
	score $\geq$ 90	23.7	18	10.67

Table 1. The miss rate detected by several nets(%).

		SSD	SSD+Conv3_3	SDD+Conv3_3+TL(CDD)
BEFORE	score $\geq$ 50	4.51	4.6	2.46
	score $\geq$ 90	1.21	1.3	0.34
AFTER	score $\geq$ 50	2.46	2.23	1.97
	score $\geq$ 90	0.34	0.49	0.47

Table 2. The false rate detected by several nets(%).

data. In our paper, we compare the other nets with CDD. And the miss rate be lower by combining shallow feature map with transfer learning. In Table 2 for instance, the percentage of false classification value comparison between we then add non-defective samples into training data after add difficult defective samples and with net weights transferred. After we transferred weights of CDD trained on added training dataset by upsampling, we train CDD on non-defective data. As the same as miss rate, we compare the other nets with CDD. And the false rate be lower by training non-defective samples with transfer learning.

The comparison shows that in the assessment of multi-class classification accuracy, the one classes of normal should be integrated into three class (Normal). As a result, there are eight classes (pit, oil, dirty, scratch, wire-drawing, logo, texture and line) considered for the performance evaluation of the CDD models. The performance results of the CDD models for the multi-class (5 classes defects) classifications are summarized in Fig. 9. Here we runs predictions over the entire cooker dataset, then matches these predictions to the ground truth boxes, computes the precision-recall curves for each class, and finally computes the mean average precision over all classes. Our CDDNet reaches well performance on cooker data. We compares the obtained accuracy results of the CDD based on the means of the computed performance metrics and the corresponding standard deviations. As shown in Fig. 8, not only the little defect but also the normal area on cooker surface which can be easily been detected as scratch, we achieve a not bad balance in detection accuracy and false rate by CDD.

5. Conclusion

We present a new framework for solving an industrial

optical inspection problem by transferring weights from a CNN trained on an entirely different domain. Our experimental results show that if there is insufficient data, transferring weights training a deep CNN from defect is often effective. Also, in the case when the source and target domains are highly dissimilar, using the fine-tuned transferred weights does help in learning new representations of the target domain and the performance increases dramatically. Through CDD, most of the unnecessary variations encoded in the weights of the source network are deactivated. At the same time, meaningful variations of the Ccooker dataset are amplified to capture the new simple variations of the target domain.

References

- (1) Bissi, Lucia, et al. "Automated defect detection in uniform and structured fabrics using Gabor filters and PCA." *Journal of Visual Communication and Image Representation* 24.7 (2013): 838-845.
- (2) Bu, H-G., et al. "Detection of fabric defects by auto-regressive spectral analysis and support vector data description." *Textile Research Journal* 80.7 (2010): 579-589
- (3) Deng, Li. "A tutorial survey of architectures, algorithms, and applications for deep learning." *APSIPA Transactions on Signal and Information Processing* 3 (2014)
- (4) Ghazvini M, Monadjemi S A, Movahhedinia N, et al. Defect detection of tiles using 2D-wavelet transform and statistical features[J]. *World Academy of Science, Engineering and Technology*, 2009, 49(901-904): 1
- (5) Gulli, Antonio, and Sujit Pal. *Deep Learning with Keras*. Packt Publishing Ltd, 2017

- (6) Jia, Feng, et al. "A neural network constructed by deep learning technique and its application to intelligent fault diagnosis of machines." *Neurocomputing* 272 (2018): 619-628
- (7) Jing, Luyang, et al. "A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox." *Measurement* 111 (2017): 1-10
- (8) Kim, Seunghyeon, et al. "Transfer learning for automated optical inspection." 2017 International Joint Conference on Neural Networks (IJCNN). IEEE, 2017
- (9) Latif-Amet, A., Aysin Ertüzün, and Aytül Erçil. "An efficient method for texture defect detection: sub-band domain co-occurrence matrices." *Image and Vision computing* 18.6-7 (2000): 543-553
- (10) LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature* 521.7553 (2015): 436
- (11) LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324
- (12) Li, Wei-Chen, and Du-Ming Tsai. "Defect inspection in low-contrast LCD images using Hough transform-based nonstationary line detection." *IEEE Transactions on industrial informatics* 7.1 (2011): 136-147
- (13) Liu, Wei, et al. "Ssd: Single shot multibox detector." *European conference on computer vision*. Springer, Cham, 2016
- (14) Luo, Bo, et al. "Early fault detection of machine tools based on deep learning and dynamic identification." *IEEE Transactions on Industrial Electronics* 66.1 (2019): 509-518
- (15) Mak, Kai-Ling, P. Peng, and Ka Fai Cedric Yiu. "Fabric defect detection using morphological filters." *Image and Vision Computing* 27.10 (2009): 1585-1592
- (16) Masci, Jonathan, et al. "Steel defect classification with max-pooling convolutional neural networks." *The 2012 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2012
- (17) Mukherjee, Anirban, et al. "An object-based coding scheme for frontal surface of defective fluted ingot." *ISA transactions* 45.1 (2006): 1-8
- (18) Najafabadi, Farzaneh Salimian, and Hossein Pourghassem. "Corner defect detection based on dot product in ceramic tile images." 2011 IEEE 7th International Colloquium on Signal Processing and its Applications. IEEE, 2011
- (19) Ng, Hui-Fuang. "Automatic thresholding for defect detection." *Pattern recognition letters* 27.14 (2006): 1644-1649
- (20) Pernkopf, Franz, and Paul O'Leary. "Visual inspection of machined metallic high-precision surfaces." *EURASIP Journal on Advances in Signal Processing* 2002.7 (2002): 650750
- (21) Polzleitner, W. "Defect detection on wooden surface using gabor filters with evolutionary algorithm design." *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*. Vol. 1. IEEE, 2001
- (22) Protopapadakis, Eftychios, and Nikolaos Doulamis. "Image based approaches for tunnels' defects recognition via robotic inspectors." *International Symposium on Visual Computing*. Springer, Cham, 2015
- (23) Ruz, Gonzalo A., Pablo A. Estevez, and Claudio A. Perez. "A neurofuzzy color image segmentation method for wood surface defect detection." *Forest products journal* 55.4 (2005): 52-58
- (24) Soukup, Daniel, and Reinhold Huber-Mörk. "Convolutional neural networks for steel surface defect detection from photometric stereo images." *International Symposium on Visual Computing*. Springer, Cham, 2014
- (25) Topal, Cihan, and Cuneyt Akinlar. "Edge drawing: a combined real-time edge and segment detector." *Journal of Visual Communication and Image Representation* 23.6 (2012): 862-872
- (26) Von Gioi, Rafael Grompone, et al. "LSD: A fast line segment detector with a false detection control." *IEEE transactions on pattern analysis and machine intelligence* 32.4 (2010): 722-732
- (27) Wen, W., and A. Xia. "Verifying edges for visual inspection purposes." *Pattern recognition letters* 20.3 (1999): 315-328
- (28) Xie, Xianghua. "A review of recent advances in surface defect detection using texture analysis techniques." *ELCVIA: electronic letters on computer vision and image analysis* 7.3 (2008): 1-22
- (29) Xue-Wu, Zhang, et al. "A vision inspection system for the surface defects of strongly reflected metal based on multi-class SVM." *Expert Systems with Applications* 38.5 (2011): 5930-5939
- (30) Yosinski, Jason, et al. "How transferable are features in deep neural networks?." *Advances in neural information processing systems*. 2014
- (31) Zhang, Lei, et al. "Road crack detection using deep convolutional neural network." 2016 IEEE international

conference on image processing (ICIP). IEEE, 2016

- (32) Zheng, Hong, Ling Xue Kong, and Saeid Nahavandi. "Automatic inspection of metallic surface defects using genetic algorithms." *Journal of materials processing technology* 125 (2002): 427-433
- (33) Zhou, Yuqing, and Wei Xue. "Review of tool condition monitoring methods in milling processes." *The International Journal of Advanced Manufacturing Technology* (2018): 1-15