

Defect Detection for Polymeric Polarizer Based on Faster R-CNN

Hai-Wei Lei*, Bin Wang,

School of Computer Science and Technology
North University of China
No. 3, University Road, Taiyuan, 030051, Shanxi

He-He Wu and An-Hong Wang

Institute of Digital Multimedia and Communication
Taiyuan University of Science and Technology
No. 66, Waliu Road, Taiyuan, 030024, China

*Corresponding Author:lhwh0312@163.com

Received March, 2018; Revised August, 2018

ABSTRACT. *A polymeric polarizer is a crucial part of thin-film transistor (TFT) liquid crystal display (LCD) panels. Its physical defects have a serious impact on the quality of TFT-LCD panels, so quality problems associated with the polarizer cannot be ignored. At present, defects are detected by using manual work and few methods based on traditional image processing. It is a promising and challenging task to implement the detection of defects in the polarizer using the deep learning method, so a method based on Faster R-CNN was used in this paper, it typically contains both the process of training models and detecting defects. The experimental results showed that this method can address the problem of the quality of the polarizer image, complete the detection quickly, and locate and mark the defect accurately.*

Keywords: Deep learning, Faster R-CNN, Polarizer defect detection, Industrial vision inspection

1. **Introduction.** With the rapid development of society, people's lifestyles also have changed significantly, and more electronic products are being used, such as television sets, computer monitors and mobile phones, and they have become an indispensable part of our day-to-day lives. Thin-film transistor (TFT) liquid crystal display (LCD) panels are used extensively in these electronic devices as the display terminal since they have the advantages of full-color display capabilities, low power consumption, and small size [1]. The polymeric polarizer is a crucial part of a TFT-LCD panel, and it is attached to two sides of the liquid crystal and is responsible for adjusting the optical path. However, defects, such as bubbles and scratches, inevitably occur in the process of manufacturing the polarizer. Also, dust sometimes causes problems even though the entire production process is conducted in a "dust-free" workshop. These defects reduce the quality of the TFT-LCD panels because they affect the display. Therefore, detecting any defects in the polarizer is an indispensable step before integrating the polarizer into the liquid crystal panel.

Currently, manual detection of the defects in the polarizer occurs to some extent, but this is a very labor-intensive, slow process that lacks reliability and consistency. Several recent papers have proposed the use of traditional image processing to detect defects in the polarizer. Wu et al. [2] proposed an SD&R filter for detecting the visual defects on polarizers, which was better than the traditional image enhancement filters, including Robert, Sobel, and Sharpening. In this kind of image enhancement technique, the contrast of the image is enhanced, and, thus, the visual defects are easier to address by further detection approaches. In [3], the images with defects were segmented using an adaptive threshold method, and then the locations of the defects were marked by an improved, fast-clustering algorithm based on density. This method is better at detecting fingerprints, dirt, and dust. Guo et al. [4] developed a set of on-line and real-time systems for detecting defects in the image of a polarizer. First, the image is down-sampled, and the boundary of the defect is identified by Laplace transform. Then, the boundary is segmented using the statistical decision method, and the type of defect is determined by Huffman transformation. In [5], a computer vision system was presented that can be used to detect the tiny bump defects in a polarizer. The system projects a straight, black-and-white fringe pattern onto the polarizer film and then captures the image using a charge couple device (CCD) camera. The defects can be identified because the location of the defect is deformed. Sang-Wook Sohn et al. [6] used a laser transmission method to obtain the polarizer image, and then they applied a least mean squares adaptive filtering technique to remove background noise and identified the defects based on a binary image that is obtained through a multi-threshold. Most of the methods described above only can detect certain defects, and they require specific lighting conditions and lack robustness.

In recent years, there have been rapid developments and improvements in computer vision areas, including the ability to classify images, detect objects, and identify faces. These achievements have the benefit of a variety of convolutional neural networks (CNNs), which are a class of deep, feed-forward artificial neural networks that has been used successfully in analyzing visual imagery. Among the CNNs, the region-based convolutional neural network (R-CNN) is very effective for detecting objects. With the optimization of the structure of the network and the improvement of the performance of the hardware, the accuracy and speed of detection have been improved significantly. In particular, the advent of Faster R-CNN network offers hope for real-time detection, and it certainly will be used in industrial inspections in the near future.

In this paper, Faster R-CNN was used to detect and locate the defects on a small polarizer that is used in mobile phones. Both the dirt and the marker (a mark for indicating defects) were detected. The experimental results showed that the Faster R-CNN can identify these two defects and achieve precise locating and labeling.

The remainder of this paper is organized as follows. The second part introduces the principle of the Faster R-CNN and the overall framework of the network, and the third part details the method of detecting defects based on the Faster R-CNN, including constructing the database, training the network, and testing the process. The fourth part presents the experimental results, and our conclusions are given in the final part.

2. Faster R-CNN Networks. Faster R-CNN is one of the state-of-art methods for detecting objects, and it was developed originally by CNN. CNN can classify images by identifying their visual features. Krichevsky et al. use CNN to solve the single image classification problem in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

2012, and won the first place [7]. However, the classification and localization of multiple objects in an image are still problems. To solve these problems, Girshick et al. [8] proposed a region-based convolutional neural network (R-CNN). R-CNN generates region proposals by using a selective search (SS) algorithm [9], and each one of the region proposals is resized to a fixed size and fed as input to the CNN for feature extraction. The CNN is followed by a support vector machine (classifier) to estimate the kind of image and by a regressor to refine the position of the bounding box. The disadvantages of R-CNN are that multiple stages must be trained, the steps are cumbersome, the training is time consuming, and it takes up a lot of space on the disk. In addition, it has very slow training and detection speeds, e.g., the VGG16 [10] model requires about 47s to deal with an image by using GPU.

In 2015, Girshick et al. proposed an improved scheme for R-CNN, i.e., Fast R-CNN[11]. It processes the entire image with several convolutional and max pooling layers to produce a convolutional feature map. For each region proposal, a fixed-length feature vector is extracted from the feature map through a region of interest (RoI) pooling layer and fed into a sequence of fully connected (fc) layers. Instead of Support Vector Machine (SVM) and the regressor, two sibling output layers at the end of the network produce softmax probability estimates for K classes and refine the position of the bounding box. Fast R-CNN avoided repeating the convolution operation for each region proposal and unified the classification and the regression of the position of the bounding box. Therefore, it improved the speed of training deeper neural networks, such as VGG16. Compared to R-CNN, the speed for the Fast R-CNN training stage is nine times faster and the speed for the test is 213 times faster.

Although Fast R-CNN resolves many of the disadvantages associated with R-CNN, the first step of detecting candidate regions using the SS algorithm still makes the whole network slow. To solve this bottleneck, Faster R-CNN [12] adds a regional proposal network (RPN) to Fast R-CNN, which is an alternative to the SS algorithm and can share convolutional layers. Fig. 1 shows the architecture of the Faster R-CNN. For an input image, first, feature maps are computed by the shared convolutional layers, and the RPN predicts a set of object proposals and corresponding scores indicating which are objects and which are not based on these feature maps. Then, a RoI pooling layer extracts a fixed-length feature vector for each proposal from the above feature maps. After that, the feature vector is fed into fully-connected layers. Finally, the classification layers calculate which category each proposal belongs to and outputs the probability of each category, after which the bounding box regressor refines the spatial location for the proposals.

The RPN is a key part of Faster R-CNN, which is constructed by adding an $n \times n$ (typically $n = 3$) convolutional layer and two sibling 1×1 convolutional layers on the top of shared convolutional layers. The first convolutional layer maps each 3×3 sliding window in the feature maps to a lower-dimensional feature vector (512-d for VGG16). The two following layers are the classification layer and the regression layer, which output the object-ness score and the bounding box coordinates for each window, respectively. An anchor-based method is proposed to address the problem of multiple scales and sizes of objects. Each anchor is at the center of a spatial window and has a specific scale and aspect ratio. For each spatial window, three scales (128^2 , 256^2 , and 512^2 pixels) and three aspect ratios (1:1, 1:2, and 2:1) are used, i.e., $k = 9$ anchors are generated. Thus, the RPN classifies and regresses k proposals of different sizes that correspond to k anchor boxes in

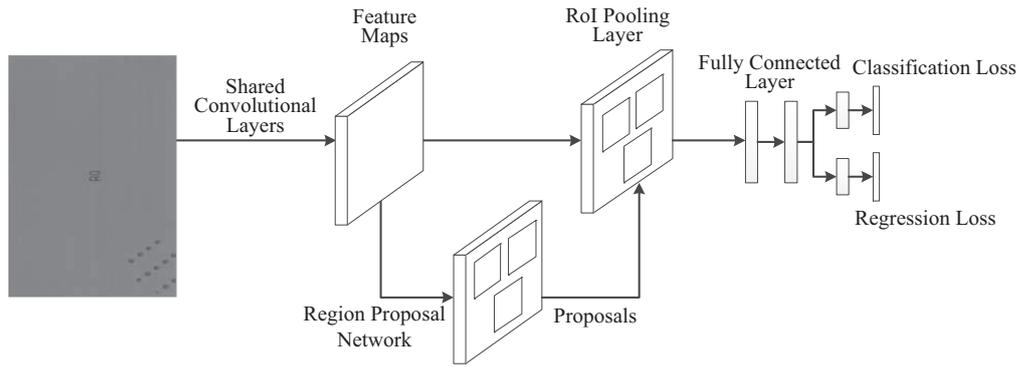


FIGURE 1. Architecture of the Faster R-CNN

each window position. Consequently, the classification layer outputs 2k possibility scores of being an object, and the regression layer outputs 4k box coordinates.

3. Polarizer Detection Based on Faster R-CNN.

3.1. Basic framework of visual inspection system. In order to realize the automatic detection of the polarizer, a set of visual inspection system is necessary. Fig. 2 shows the prototype of our visual inspection system. In the system, polarizer images are obtained by a line-scan camera under the control of an optocoupler trigger and an encoder. The detection algorithm in the industrial personal computer (IPC) performs defect detection on the image and gives a result. Then, the control signal is sent to a controller which achieves rejection of the defective product.

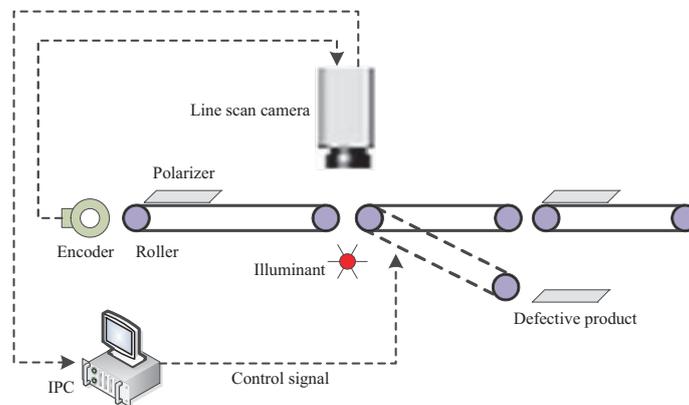


FIGURE 2. Prototype of our visual inspection system

3.2. Datasets. As mentioned above, polarizer images were taken by a line-scan Complementary Metal Oxide Semiconductor (CMOS) camera, with the light transmitted from the bottom. For each of the captured images, the categories and locations of the defects were labeled by “LabelImg”, which is a graphical image annotation tool that can be used to label object bounding boxes in images. Two types of defects are marked in our dataset, i.e., dirt and marker.

In addition, horizontal flip and rotation operations were used to increase the training samples, and 1757 Polarizer images were used as the training set, 200 images were used as the test set, and another 200 images were used as the validation set.

3.3. Network training. Faster R-CNN has two training methods, i.e., alternative training (alt-opt) and approximate joint end-to-end training. In our training process, we used the approximate joint end-to-end strategy.

In order to get the final model, we also used the four-step alternating training method. The specific steps are as follows: In the first step, we used an ImageNet pre-trained model to fine-tune the RPN based on our dataset and generated region proposals. Note that our dataset for detecting objects was too small, so we had to use another larger dataset, like ImageNet, to initialize the network, which is called pre-training. In the second step, we trained a separate detection network by Fast R-CNN, which also was initialized by the ImageNet pre-trained model and using the proposals generated by the RPN. At this point, the two networks did not share convolutional layers. In the third step, we used the model that was generated in the second step to initialize RPN training, but we fixed the shared convolutional layers and only fine-tuned the layers unique to RPN. Then, the two networks shared convolutional layers. In the fourth step, we kept the shared convolutional layers fixed and fine-tuned the FC layers of the Fast R-CNN. Through the four-step training, the model was obtained, and it was used in the subsequent testing process.

For training images, the pixels were 4096×3796 , and its original aspect ratio was unchanged. We trained the RPN and the Fast R-CNN with SGD, and the maximum iterations were 80k and 40k, respectively. The base learning rate, the learning policy, the weight decay, and the momentum of the RPN and the Fast R-CNN were similar. In detail, we set the base learning rate to 0.001, and the learning policy is step, and we used a weight decay of 0.0005 and a momentum of 0.9.

In RPN, non-maximum suppression (NMS) [14] was conducted to exclude some redundant region proposals after the bounding box regression. That is, the region proposal was removed when its Intersection-over-Union (IoU) with a higher scored proposal was greater than a given threshold, i.e., 0.7. We used the top 2k proposals per image for training after NMS, and only 300 proposals per image for testing in the following experiments section.

3.4. The Detection Process. It involves the following steps for detecting defects in the polarizer: (1) Taking the polarizer picture to be tested as the input picture, and the feature map was obtained through 13 layers of the convolutional layer. (2) Using the convolution feature map as the input of RPN network, a large number of proposal regions were generated. (3) The non-maximum suppression (NMS) operation was performed on the proposal region box, and the top 300 proposal regions with the highest scores were reserved. (4) Take the features in the proposal regions of the feature map to form high-dimensional feature vectors, and calculate the score of each class from the Fast R-CNN detection network and predict the position of defects.

Through the above steps, the identification and localization of the defects in polarizer images were determined.

4. Experiments. The experiments are conducted on a workstation that was equipped with a NVIDIA GeForce GTX 1080 GPU and 8 GB memory. The deep learning framework we used was Caffe, and the training and testing processes were implemented by modifying

the source code of the original Faster R-CNN [15]. The size of the image captured for detecting defects was 4096×3796 .

Faster R-CNN provides three training models, i.e., the ZF model, the VGG_CNN_M_1024 model, and the VGG16 model. We conducted the experiments separately on these three models, and the results are shown in Table 1. We used the mean Average Precision (mAP) as the evaluation criterion, a metric for object detection. The VGG16 model provided a higher mAP than ZF and VGG_CNN_M_1024, so we used the VGG model as our training network. The results of the VGG16 model show that the Faster R-CNN had good capability of identifying the defects on the polarizer, but they also indicate that this method has a certain fall-out ratio.

TABLE 1. Results of the three training models

training model	mAP(%)	dirty(%)	marker(%)
VGG16	67.5	62.6	72.4
ZF	43.8	30.1	57.5
VGG_GNN_m_1024	64.5	58.4	70.6

Fig. 3 shows the detection results of a polarizer image, where a) is the original polarizer image and b) and c) are the probability estimation and bounding box for dirty and marker, respectively. From Fig. 3, it can be seen that most of the defective areas are identified.



FIGURE 3. Example of the results of detecting defects in the image of the polarizer

The maximum iterations and the dropout rate affected the mAP of the Faster R-CNN. These two parameters were determined by training the mode repeatedly, and, as a result, the maximum number of iterations was 70,000, and the dropout rate was 0.5. In addition, the time spent on detecting a polarizer also was included in order to evaluate the testing speed. The average time for multiple tests was about 0.3 second, which shows that deep learning is a promising method for industrial real-time detection.

5. Conclusions. In this paper, a defects detection and localization method based on Faster R-CNN for polymeric polarizer is proposed for use in verifying the possibility of using and accuracy of the deep learning method for detecting defects in a polarizer. The

experimental results show that this method achieved good detection accuracy and a near real-time detection speed by using the GPU computing unit. The research work in this paper can serve as a reference for follow-up research associated with the detection of defects in polarizers.

In our future work, we will change the number of layers of the network and adjust some of the networks parameters to optimize the test model and thus improve the efficiency and accuracy of detection.

Acknowledgment. We would like to acknowledge the supports from the Science and Technology Innovation Cultivation Team Project of Shanxi Province (20171025).

REFERENCES

- [1] V. P. Luo, X. N. Yun, and K. Y. Kim, The manufacturing and developing of the TFT-LCD, *Advanced Display*, vol. 134, pp. 31-38, 2012.
- [2] C. H. Yeh and F. C. Wu, An Image Enhancement Technique in Inspecting Visual Defects of Polarizers in TFT-LCD Industry, in *2009 International Conference on Computer Modeling and Simulation*, 2009, pp. 257-261.
- [3] M. Cai, Z. Wu, and F. Tian, A deity-based improved fast clustering algorithm and its application to the inspection of Polarizer defects, *Technical Journal*, vol. 24, pp. 267-274, 2009.
- [4] C.-F. J. Kuo, C.-H. Chiu, and Y.-C. Chou, Research and Development of Intelligent On-Line Real-time Defect Inspection System for Polymer Polarizer, *Polymer-Plastics Technology and Engineering*, vol. 48, pp. 185-192, 2009/02/02 2009.
- [5] H. N. Yen and M. J. Syu, Inspection of polarizer tiny bump defects using computer vision, in *2015 IEEE International Conference on Consumer Electronics (ICCE)*, 2015, pp. 525-527.
- [6] S.-W. Sohn, D.-Y. Lee, H. Choi, J.-W. Suh, and H.-D. Bae, Detection of Various Defects in TFT-LCD Polarizing Film, *Adaptive & Natural Computing Algorithms*, vol. 4432, pp. 534-543, 2007.
- [7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, presented at the *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, Lake Tahoe, Nevada, 2012.
- [8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580-587.
- [9] J. R. R. Uijlings, K. E. A. v. d. Sande, T. Gevers, and A. W. M. Smeulders, Selective Search for Object Recognition, *International Journal of Computer Vision*, vol. 104, pp. 154-171, 2013.
- [10] K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, *CoRR*, vol. abs/1409.1556, / 2014.
- [11] R. Girshick, Fast R-CNN, in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440-1448.
- [12] S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137-1149, 2017.
- [13] J. Deng, W. Dong, R. Socher, L. J. Li, L. Kai, and F.-F. Li, ImageNet: A large-scale hierarchical image database, in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248-255.
- [14] A. Neubeck and L. V. Gool, Efficient Non-Maximum Suppression, in *18th International Conference on Pattern Recognition (ICPR'06)*, 2006, pp. 850-855.
- [15] Faster R-CNN https://github.com/shaoqingren/faster_rcnn.