MULTI-SCALE DEFECT DETECTION NETWORK FOR TIRE X-RAY IMAGES

Ren Wang[1,2], *Qiang Guo*[1,2], *Caiming Zhang*[3]

[1]School of Computer Science and Technology,
Shandong University of Finance and Economics, Jinan, China
[2]Shandong Provincial Key Laboratory of Digital Media Technology, Jinan, China
[3]Software College, Shandong University, Jinan, China

ABSTRACT

Though automatic detection method has been tremendous improved, with the development of deep learning. Defect detection in many industrial processes is one of the remaining challenging tasks due to the diversity of products. In this work, we focus on detection tasks in tire industry and develop a *Multi-scale Defect Detection Network (MDDN)*, which contains two parallel sub-networks to capture multi-scale defect features. Specifically, high-abstracted semantic features containing defect shapes and locations are mined via a *Semantic-aware sub-network*, simplified by an off-the-shelf fully convolutional network. Furthermore, to complement the details filtered by the deep network, a novel *Texture-aware Sub-network* is used to cover edge features and small defects as much as possible. Finally, pixel-wised detection results are obtained by fusing features with semantic and texture information. Extensive experiments demonstrate that *MDDN* can produce comparable results and achieve significantly performance improvement in small defects detection.

***Index Terms*—** Defect detection, Fully convolutional network, Semantic segmentation, Multi-scale context

1. INTRODUCTION

Automatic defect detection, used to improve quality and accelerate production, has become an indispensable part in industrial processes [1, 2, 3]. Especially in tire manufacturing, numerous detection algorithms have been proposed [4, 5, 6] and aroused extensive attention recently. In most real-world applications, tire defect detection is first carried out by deriving the defective region from tire X-ray images, which contains various types of defects caused by unclean raw materials and undesired manufacturing facilities [7]. Then, the defective product is hierarchical processed according to the location and area of defects. Due to unique properties of the tire image, for instance complexity and low-quality, illustrated in previous study [8, 9], most inspection processes are performed by human observers, which increases the risk and reduces the efficiency. Therefore, tire defect detection remains one of the most challenging inspection tasks.

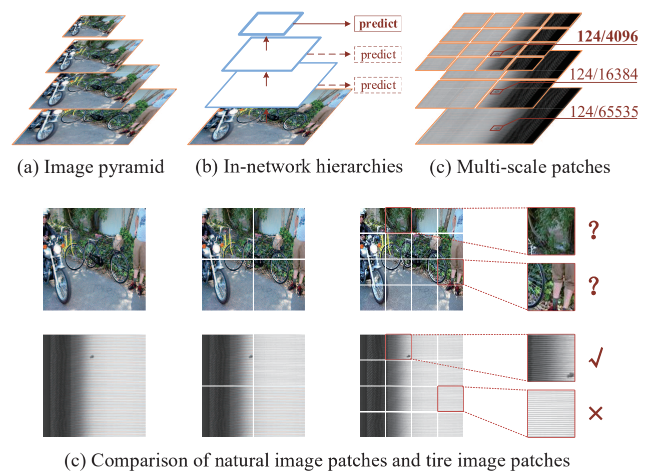

Fig. 1. (a)shows the image pyramids. (b) indicates in-network feature hierarchies. Prediction results can be derived from each layer. (c) represents multi-size tire image patches, which can increase the relative scale of small defects and backgrounds. (d) illustrates that the semantic information is more easily retained in tire image peaches by comparison.

At present, existing computer vision based detection methods are mostly devoted to distinguishing difference between defective regions and background (defective-free regions). A key issue for such methods is feature extraction. Guo *et al*. [10] proposed a component decomposition based method to detect tire defects, which separated the background from the image by means of two designed filters. Then through an adaptive thresholding processing, defects were derived from the residual image. Besides, Independent component analysis was also used for defect detection tasks [11, 12]. A major disadvantage of these fundamental methods is the limitation of the information contained in low-level clues and domain features. To address the limitation, Zhang *et al*. [8, 13] introduced mulit-scale wavelet and curvelet transform, in detection tasks respectively. Furthermore, optimized edge detection and total variation algorithm are used to achieve more accurate results [14]. However, the repre-

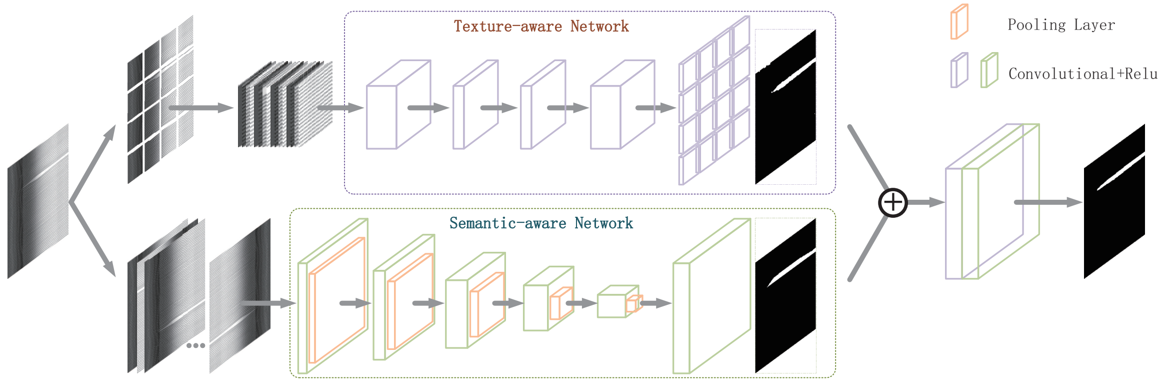

Fig. 2. Overall architecture of the proposed MDDN.

sentation capability of fixed kernels is not comprehensive enough. Moreover, transform processes are computationally expensive. Recently, Cui *et al.* [15] attempted to classify tire defects by means of convolutional neural networks, which has outstanding performance in the recognition and segmentation tasks of natural images. With the excellent feature extraction capability of deep convolutional network (ConvNets), Wang *et al.* [9] further implemented the detection and segmentation in tire images by a fully convolutional network (FCN) [16]. However, due to the existence of pooling layers, FCN is not sensitive to small defects and edge details, which is similar to that in dealing with natural image tasks.

To overcome these shortcomings, many methods have been proposed in benchmark datasets. Most of them are based on multi-scale strategies and can be roughly classified into image pyramids and in-network feature hierarchies. Image pyramids were directly scaled to get multi-scale images and extensively used in the era of hand-crafted features [17, 18], as shown in Figure 1(a). Even if the crafted features have largely been replaced by self-learning features, multi-scale testings on the image pyramid are still used to verify the adaptability and robustness (*e.g.*, [19]). Nevertheless, image pyramid based methods is impractical for real applications due to the considerable increase in inference time. In-network feature hierarchies are formed by the forward propagation within deep ConvNets. Through several of sub-sampling layers, in-network hierarchies produce feature maps of different spatial resolutions, with an multi-scale and pyramid shape[20]. By fusing these multi-scale feature maps, features in shallow and deep layers can be perceived. The Single Shot Detector (SSD) [21] is one of the first attempts at combining predictions from these features maps to detect objects of various sizes. Generally, shallow features are used to predict small objects, and deep features with large receptive fields are used to detect large objects. However, the lack of semantic information is harmful to the detection of small targets in shallow layers. Another fusing way can effectively address this problem by concatenating multi-scale features and detecting on top of the expanded feature maps, as shown by the red line in Figure 1b[]. For example, FCN defined a skip architecture to produce more accurate segmentation. Similar top-down skip architectures are popular in recent research[22, 23]. There exists a basic problem that it is still not enough to mine the detail texture in these structures[24]. Bai *et al.* proposed a novel multi-task generative adversarial network (MTGAN), which improve the detection performance by up-sampling a small object to a larger scale using super-resolution.

Inspired by MTGAN, we construct a end-to-end network named Multi-scale Defect Detection Network (MDDN) consisting of a semantic-aware sub-network and a texture-aware sub-network. To detect small tire defects, image patches [25, 26] are fed in the texture-aware sub-network. Unlike natural image patches, defects (objects) are still significant and discernible in the tire image patches, illustrated in Figure 1(d). In addition, as shown in Figure 1(c), the proportion of defects in a 256×256 tire image increases from 124/65535 to 124/4096, which is advantageous for better capturing of detailed information.

2. MULTI-SCALE DEFECT DETECTION NETWORK

This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method. This is the proposed method.This is the proposed method.This is the proposed method.This is the proposed method.

2.1. Semantic-aware sub-network

This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.

This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network. This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.This is Semantic-aware sub-network.

2.2. Texture-aware sub-network

This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.

This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network. This is Texture-aware sub-network.This is Texture-aware sub-network.This is Texture-aware sub-network.

3. EXPERIMENTS

3.1. Implementation details

Dataset. This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset. This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset. This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.This is Dataset.

Parameter Setting. This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting. This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting. This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.This is Parameter Setting.

Metrics. This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices. This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices. This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.This is Metrices.

3.2. Ablation experiment

This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment. This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the

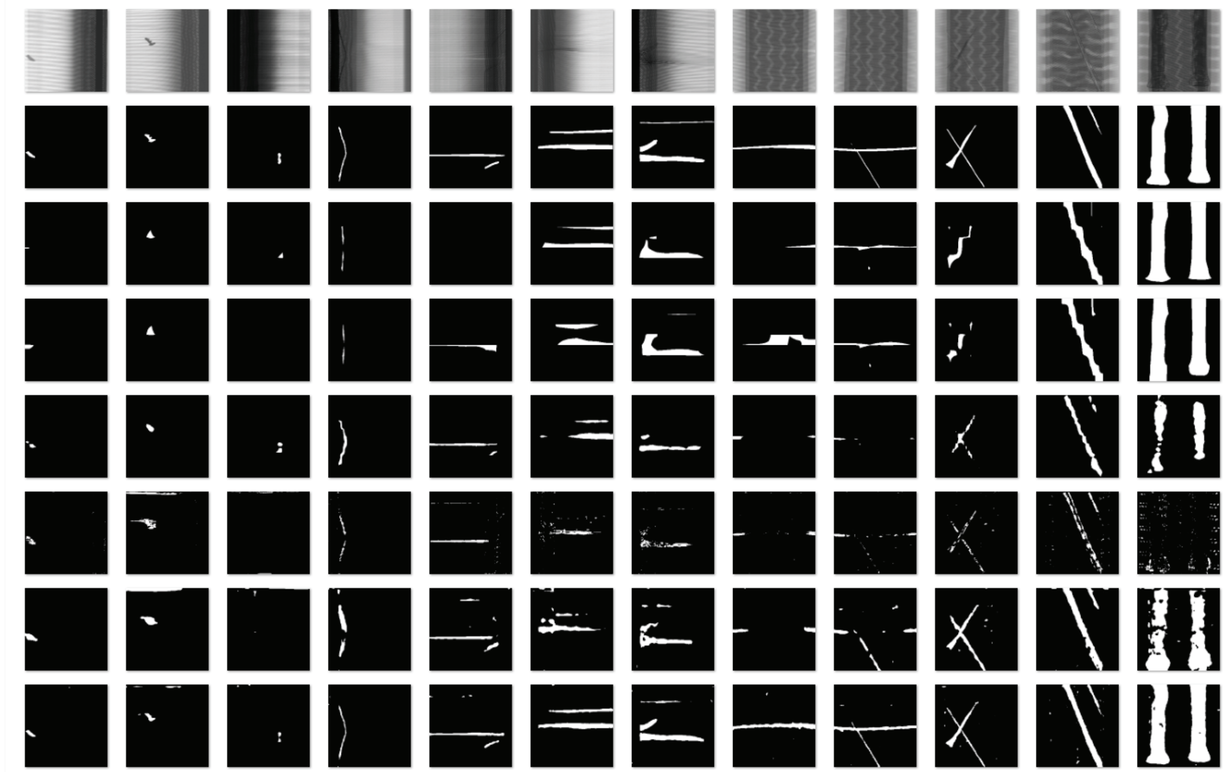

Fig. 3. Comparison of experimental results.

ablation experiment. This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment. This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment. This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.This is the ablation experiment.

3.3. Comparison with state-of-the-art methods

Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods. Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.Comparison with state-of-the-art methods.

4. CONCLUSION

In this paper, we proposed a MDDN model for detailed tire defect detection tasks. Through combining a semantic-aware network and a novel texture-aware network, MDDN can preserve the necessary detail features while mining the semantic information hidden in deep layers. We showed that an increase in the proportion of defective areas is critical for small defect detection. In addition, we have experimentally verified that the blocking strategy can effectively enhance the dataset and retain detailed information, in tire images. The experiments demonstrate that our MDDN has significantly improved over the existing tire defect detecting methods, and can produce more accurate small defect detection results. The future work includes reducing noises in the texture-aware network and increasing detection speed.

5. REFERENCES

[1] A. Kumar, "Computer-vision-based fabric defect detection: A survey," *IEEE Trans. on Industrial Electronics*, vol. 55, no. 1, pp. 348–363, 2008.

[2] Y. Li, W. Zhao, and J. Pan, "Deformable patterned fabric defect detection with fisher criterion-based deep learning," *IEEE Trans. on Automation Science and Engineering*, vol. 14, no. 2, pp. 1256–1264, 2016.

[3] S. Ghorai, A. Mukherjee, M. Gangadaran, and P. K. Dutta, "Automatic defect detection on hot-rolled flat steel products," *IEEE Trans. on Instrumentation and Measurement*, vol. 62, no. 3, pp. 612–621, 2012.

[4] C. Zhang, X. Li, Q. Guo, X. Yu, and C. Zhang, "Texture-invariant detection method for tire crack," *Journal of Computer-Aided Design & Computer Graphics*, vol. 25, no. 6, pp. 809–816, 2013.

[5] Y. Zhang, X. Cui, Y. Liu, and B. Yu, "Tire defects classification using convolution architecture for fast feature embedding," *International Journal of Computational Intelligence Systems*, vol. 11, no. 1, pp. 1056–1066, 2018.

[6] C.D. Jones, A.B. Smith, and E.F. Roberts, "A dictionary-based method for tire defect detection," in *ICIA*. IEEE, 2014, pp. 519–523.

[7] Q. Guo, C. Zhang, H. Liu, and X. Zhang, "Defect detection in tire x-ray images using weighted texture dissimilarity," *Journal of Sensors*, vol. 2016, 2016.

[8] Y. Zhang, T. Li, and Q. Li, "Defect detection for tire laser shearography image using curvelet transform based edge detector," *Optics & Laser Technology*, vol. 47, pp. 64–71, 2013.

[9] R. Wang, Q. Guo, S. Lu, and C. Zhang, "Tire defect detection using fully convolutional network," *IEEE Access*, vol. 7, pp. 43502–43510, 2019.

[10] Q. Guo and Z. W. Wei, "Tire defect detection using image component decomposition," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 1, pp. 41–44, 2012.

[11] X. Cui, Y. Liu, and C. Wang, "Defect automatic detection for tire x-ray images using inverse transformation of principal component residual," in *AIPR*. IEEE, 2016, pp. 1–8.

[12] X. H. Cui, Y. Liu, C. X. Wang, and H. Li, "A novel method for feature extraction and automatic recognition of tire defects using independent component analysis," *DEStech Transactions on Materials Science and Engineering*, , no. icimm, 2016.

[13] Y. Zhang, D. Lefebvre, and Q. Li, "Automatic detection of defects in tire radiographic images," *IEEE Trans. on Automation Science and Engineering*, vol. 14, no. 3, pp. 1378–1386, 2015.

[14] Z. Yan, L. Tao, and L. Qing-Ling, "Detection of foreign bodies and bubble defects in tire radiography images based on total variation and edge detection," *Chinese Physics Letters*, vol. 30, no. 8, pp. 084205, 2013.

[15] X. Cui, Y. Liu, Y. Zhang, and C. Wang, "Tire defects classification with multi-contrast convolutional neural networks," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 32, no. 04, pp. 1850011, 2018.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440.

[17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[20] T. Y. Lin, P. Dollr, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 2117–2125.

[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*. Springer, 2016, pp. 21–37.

[22] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *ECCV*. Springer, 2016, pp. 483–499.

[23] G. Ghiasi and C. C. Fowlkes, "Laplacian pyramid reconstruction and refinement for semantic segmentation," in *ECCV*. Springer, 2016, pp. 519–534.

[24] P. Zhou, B. Ni, C. Geng, J. Hu, and Y. Xu, "Scale-transferrable object detection," in *CVPR*, 2018, pp. 528–537.

[25] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Sod-mtgan: Small object detection via multi-task generative adversarial network," in *ECCV*, 2018, pp. 206–221.

[26] L. C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *CVPR*, 2016, pp. 3640–3649.