Defect detection: Defect Classification and Localization for Additive Manufacturing using Deep Learning Method

Feng Han
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
Wuhan, China
fenghan@hust.edu.cn

Sheng Liu
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
Wuhan, China
victor_liu63@vip.126.com

Sheng Liu
School of Power and Mechanical Engineering
Wuhan University
Wuhan, China
victor_liu63@vip.126.com

Jingling Zou
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
Wuhan, China
Jinglong_zou@163.com

Yuan Ai
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
Wuhan, China
d201880255@hust.edu.cn

Chunlin Xu
School of Mechanical Science and Engineering
Huazhong University of Science and Technology
Wuhan, China
xuchunlin663@sina.com

***Abstract*—In this paper, a novel visual-based defect detection method using deep learning algorithms is reported to classify and localize the defect derived from the process of Additive Manufacturing (AM). The processing conditions and costs make it impossible to manually inspect the built parts defect. Vision-based in situ real-time defect monitoring method was conducted to acquire image data and then recognize the defect class and locality in the manufacturing process. Nevertheless, the traditional image processing technology is of poor performance when dealing with defect detection in sophisticated fabrication scenarios. Therefore, we research the viable technique, using deep learning algorithms to extract the defect image feature in continuous frame images and trained our network to learn the defect class label as well as its position. In order to test this method, we also report a defect dataset called AM Defect Recognition Benchmark Dataset, we proved the method can achieve the state-of-art defect detection performance in actual scene dataset.**

Keywords—Defect Detection, Defect Localization, Additive Manufacturing, Deep learning method

INTRODUCTION

Additive Manufacturing (AM) technology[1], or 3D printing technology, has been widely applied in industrial high-level precision manufacturing with geometric and material complexities, such as aerospace high-end manufacturing[2], biomedical application[3], etc. However, the current AM systems inevitably manufacture parts with structural defect (e.g., porosity, surface pit, surface scratch, porosity array, crack), which is unable to adapt the requirement of high-value industries. AM process parameters, such as heat source power, scan speed, hatch space, layer thickness as well as powder materials and chamber environment, are the main factors affecting the formation of defects.

There are many attempts to monitor and control the manufacturing process in recent researches. Eddy current (EC) detection[4] applied into AM manufacturing subsurface detection by Wei Du et al., which proved EC signal can be used to detect the parts subsurface quality. The defect of laser powder bed fusion material cross-contamination is monitored in real-time by applying a spectral graph-theoretic approach[5]. Rikesh Patel et al. use spatially resolved acoustic spectroscopy[6] (SRAS: a laser ultrasound inspection technique) to image material microstructure of metals and alloys. In situ, high-speed thermal imaging of the entire build area[7] has been reported, which can be used to defect detection and melt pool track. Various kinds of sensor (e.g. visual, infrared, X-ray computed tomography, ultrasound) is used to defect defection, but most detect methods have pivotal shortcomings, either the solution is too expensive and intricate to apply, or the approach is unable to perform in-situ, real-time, on-line detection and recognition with obtaining the class and localization of built parts defect at the same time. Here we purpose a novel defect detection method using deep learning object detection algorithm to acquire the defect class and its localization (even geometric shape) while only use the relatively simple and economical hardware system. We refer to Mask-RCNN[17] (a kind of region-based convolution neural network) method and construct our defect detection system.

The rest of the paper content is organized as follows: the second part is introducing the deep learning method with defect defection model architecture and implementation details. In the third section, we will show the self-made defect dataset, which is one of the most important contributions of our work. As we know, the dataset provided here have quite reference and application value. Then based on our own dataset, we trained the neural network model and give the experiment configuration details. Next section we will show defect detection result predicted with our trained model and analyze the method's performance. Finally, we will conclude our research with some further study improvement.

REGION-BASED DEFECT DETECTION

Defect detection method

Traditional image recognition technology extract image local feature (e.g., LBP[8], SIFT[9], HOG[10]) to classify and understand image semantics[15]. While these algorithms cannot catch enough information for visual recognition, in this case, Krizhevsky et al.[12] trained a convolution neural network(CNN) on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC)[13] building on the

network backpropagation algorithm of Rumelhart et al.[14], since then deep neural network got fast development.

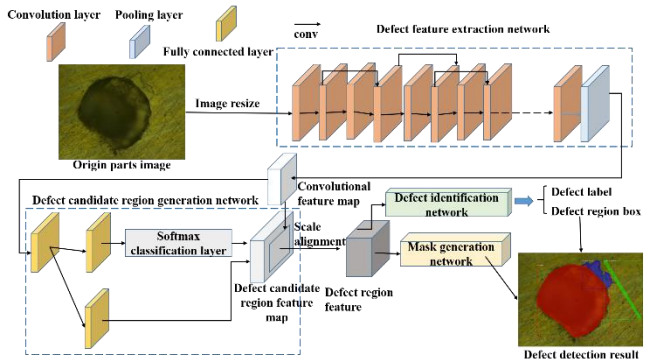

Fig. 1. Defect Detection Network Architecture

Aiming at the case segmentation task of defects in additive manufacturing parts, a model of defect instance segmentation based on mask branch is proposed. The schematic diagram of this method is shown in Figure 1. The method of part defect instance segmentation consists of four modules, which are Defect feature extraction network, Defect candidate region generation network, Defect identification network and mask generation network. The part defect detection model first reads the part image to be detected, adapts to the input size of the defect feature extraction network after size transformation, extracts the defect feature through the residual network, and outputs the convolution feature map of the original image, then passes the convolution feature map through the Defect candidate generation network to obtain the candidate area. The candidate area and the corresponding convolution feature map are aligned through the RoIAlign layer area to obtain the defect area feature map. The defect area feature map passes to two parallel network modules, the defect identification network obtains the defect type and corresponding target box coordinate of the defect area, the mask generation network obtains the binary mask of the defect instance, and finally combines the results of the two parallel networks to obtain the result of the defect instance segmentation.

The two parallel network modules in the above method for segmentation of part defect images combine target detection and image semantic segmentation. The defect recognition network finds all defects in the image. The mask segmentation network uses a fully convolutional neural network to complete defect semantic segmentation . Each module of the defect instance segmentation network can be flexibly modified and optimized, enabling end-to-end training and prediction.

Network loss function

The training purpose of the defect area generation network is to ensure that the minimum difference between the defect class and the coordinates of the bounding box predicted by the network and the true label of the defect in the training data set is continuously reduced. This difference is measured by the loss function. Each of the two types of the network output need to design a loss function, and then combine the two loss functions. For the loss function L_{cls} of the defect classification, it is necessary to perform binary classification on whether the object in each anchor box has a defect, so the log loss function is used to measure:

$$L_{cls} = LCE(p_i, p_i^*) = -[p_i ln p_i^* + (1 - p_i)\ln(1 - p_i^*)] \quad (1)$$

Where LCE () represents the log loss function, i = [1,2, ..., batch_size] is the index of the anchor box in each batch when the batch gradient descends; p_i represents the prediction probability of defects contained in each anchor box; p_i^* indicates the true label of the defect. If the current anchor frame is a positive sample then $p_i^* = 1$, and the anchor frame is a negative sample, $p_i^* = 1$. If the IOU(intersection ratio of defective areas) between anchor box and the true label is higher than 0.75, the anchor box is judged as a positive sample. When it is lower than 0.3, the anchor frame is judged as a negative sample.

For the bounding box coordinate regression loss function L_{reg}, refers to the $smooth_{L1}(x)$ in Fast-RCNN[16] research work:

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & if\ |x| < 1 \\ |x| - 0.5 & otherwise, \end{cases} \quad (2)$$

Our defect origin generation network bounding box loss function:

$$L_{reg} = smooth_{L1}(t_i - t_i^*) \quad (3)$$

Among above, i = [1,2, ..., batch_size] is the index of the anchor box in each batch during batch gradient descent; t_i represents the four position parameters (x_a, y_a, w_a, h_a) of the predicted bounding box, t_i^* represents the true defect area label associated with the anchor box of the positive sample.

The loss function of Defect origon generation network can be represented as:

$$Loss_{total} = \frac{1}{N_{cls}} \sum_i L_{cls} + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg} \quad (4)$$

N_{cls} is the regularization coefficient of the defect classification loss function, which represents the number of samples of each batch of batch gradient descent; N_{reg} is the regularization coefficient of the anchor frame position loss function, which represents the current total number of anchor boxes ; λ is the weight ratio of the total loss of the defect generation network to the two types of losses, using the same setting as in the related study [19] $\lambda = 10$; $p_i^* L_{reg}$ indicates that the regression loss is only activated in the positive sample anchor box, and for the negative sample anchor box, the regression loss of the anchor box position is not calculated.

Defect classification and mask generation

The output feature map through Defect feature extraction backbone network and defect area generation network module have concluded the defect area information of the part image, but the defect type and the actual position of the defect have not been segmented and positioned. In order to achieve the instance segmentation of the defect, the key step is to perform class identification on the defect target area, and at the same time, perform mask segmentation on the defect instances.

The schematic diagram of the defect classification and mask segmentation network structure is shown in Figure 2. First, the fixed-size feature map generated by the defect area alignment operation, and the 7 × 7 × 1024 feature map are passed through the 5th conv block of ResNet neural network. The convolution block outputs an intermediate feature map of size 7 × 7 × 2048. The average pooled output 2048-dimensional feature vector of the intermediate feature map 1

is used as the input of the defect classification network module. After mapping through two fully connected layers, the two sets of feature vectors output similar to the defect area generation network. One set of feature vectors outputs the predicted defect probability value vector, and the other set of feature vectors outputs the defect area box coordinate vectors obtained by further coordinate regression. The same as the loss function of defect origin generation network, which is divided into two parts: the classification loss and the defect box position regression loss. The mask segmentation module passes the output of the intermediate feature layer 1 through the convolution layer to obtain the intermediate feature map 2, and then outputs a 14 × 14 × 6 binary mask through a convolution layer, that is, generates a 14 × 14 size mask. The last dimension of the mask feature map indicates that the number of identified defect categories is 6 types of part manufacturing defects. The loss function of this module is defined as a binary cross-entropy loss function. For the defect area of the model, only the mask corresponding to the predicted positive sample defect area and the real defect label is calculated, and the contribution of the remaining masks to the loss function is not calculated. Finally, only the defect mask predicted by the defect classification module is retained.

The final calculation result of the model obtains a binary mask of the defect type identified by the target area and the corresponding defect instance. The binary mask is transformed into the same size as the target area by size conversion, that is, the instance segmentation of the defect image is completed.

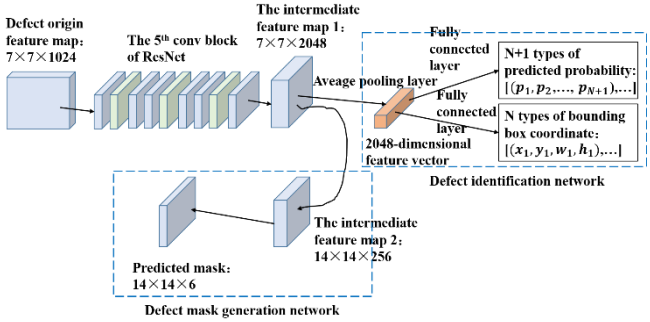

Fig. 2. Defect identification and mask generation network

DEFECT DETECTION NETWORK TRAINING

The hardware configuration of the defect detection model is a desktop host equipped with an Intel i7-7800 processor clocked at 3.5GHz, based on the Ubuntu 16.04 operating system, the machine is equipped with 16GB of memory, and 12GB of video memory NVIDIA 1080Ti graphics card. The model training data set includes 6 types of defect examples, including pore defects, crack defects, pit defects, inclusion defects, pore arrays and surface scratch defects. The training data set is 773 manual labeled defect images of additive manufacturing parts, The test data set is 103 defect pictures that are also manually labeled. The size of the data set picture is uniformly set to 1200 × 1600. For each picture, VIA (VGG Image Annotator) image annotation software is used to mark the defect category and defect outline for each defect instance, as shown in Figure 3.

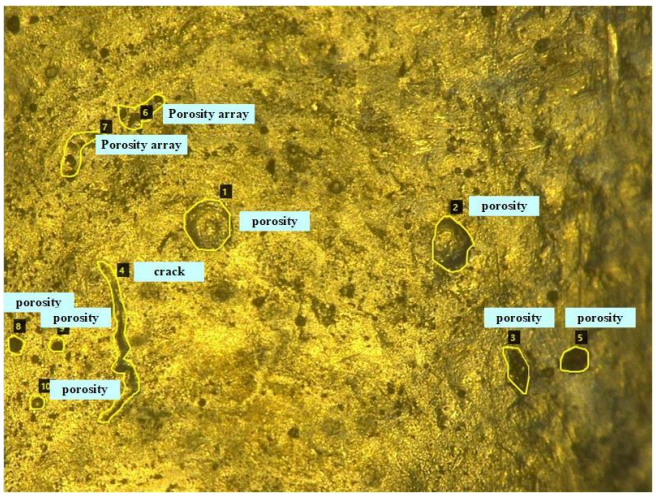

Fig. 3. Defect detection dataset labeled sample

The configuration of the model training parameters is as follows: the number of training steps per round of the data set is set to 500 steps, the initial learning rate of the model training is set to 10^{-4}, the model training optimization algorithm is the Adam algorithm, and the model training uses transfer learning technology. In this study, the weight of the model trained on the COCO image recognition data set was transferred to the defect recognition task, which effectively improved the model training effect and the accuracy of recognition in the case of a limited number of industrial defect detection data sets. During the model training process , the saved model weight data obtained in each round of training, we can migrate the model weight data obtained in the subsequent model training, so that the model can continue to train on the basis of the previous training, and enhance the flexibility of model training.

EXPERIMENT RESULT

The loss change curve of Defect instance segmentation model training 100 rounds of data set of each network module training and verification is shown in Figure 4 and Figure 5. In Figure 4, the training loss of each module has fallen close to 0 and Maintained at a stable level, the verification loss in Figure 5 has fallen rapidly to 0 while remaining stable. From the comparison of the total training loss value and the total verification loss value in Figure 6, the model's loss on the training set has converged when training for about 60 rounds. From the point of view of the loss value of the verification data set, the verification loss has been in a downward trend. After 80 rounds of training, the loss decline is no longer obvious, indicating that the model predicts the performance of the training data set and the verification data set already quite ideal.

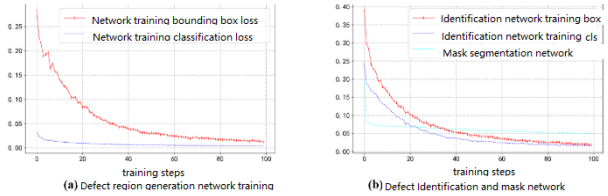

Fig. 4. Variation curve of training loss function of each network module

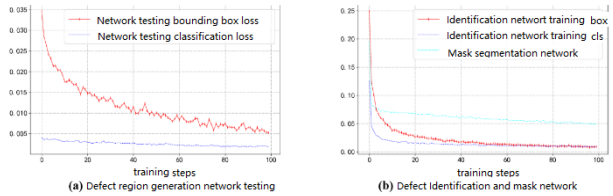

Fig. 5. Variation curve of testing loss function of each network module

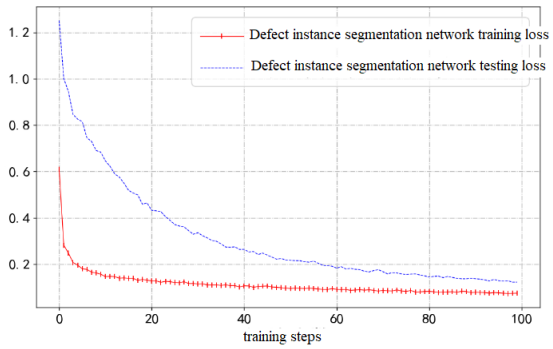

Fig. 6. Variation curve of total loss function of each network module

Using mAP (mean Average Precision) as the average of AP values of all types of defects, it can measure the comprehensive performance index of multi-type defect recognition. The defect instance segmentation prediction performance index mAP is shown in Table 1. Compared with SSD[19], Faster-RCNN[16], YOLOv3[18] and other current cutting-edge target detection frameworks to predict the performance of defects on the experimental part defect image dataset, it is derived from the table that the defect instance segmentation model proposed by the study is compared with the other three detection methods The average mAP value of the comprehensive defect identification is the highest, reaching 89.4mAP, but compared with the detection speed of the single-stage target detection method SSD[19] and YOLOv3[18], there is still some room for improvement. The model proposed in this study can predict about 12 defect images per second.

TABEL I. DEFECT INSTANCE SEGMENTATION PERFORMANCE INDEX

Method	Blow-hole AP	Scratch AP	Crack AP	Blowhole array AP	Inclusion AP	Pit AP	Average mAP	Speed FPS
SSD ResNet101[19]	86.5	82.5	86.4	79.1	87.1	76.9	83.1	19
Faster RCNN[16]	85.7	82.6	82.7	71.9	63.6	78.5	77.5	5
Our method	94.5	87.1	89.8	90.7	87.5	86.6	**89.4**	12
YOLOv3[18]	76.5	68.9	71.8	82.1	72.0	80.3	75.3	**35**

CONCLUSION

The paper based on the segmentation and recognition of defect images, the defect instance segmentation is studied in depth. The purpose of researching defect detection instance segmentation is to distinguish the differences between similar defects, and to effectively segment and identify different instances of the same type of defect. Research proposes a well-designed deep network model. By training the defect recognition model proposed in the study, the training data set contains 773 images of the surface defects of the additive manufacturing parts marked at the instance level of the defect. At the same time, the training model is verified against 103 verification data sets. The training and verification loss curve both steadily fell to zero, indicating that the trained model has excellent defect recognition performance. Finally, the trained defect detection model is used to predict 101 parts pictures. The instance segmentation performance index mAP is higher than the three mainstream target detection algorithms in the part defect instance segmentation task of this study. The mAP reaches 89.4, and the defect instance segmentation speed reaches 12 images per second, which can realize real-time defect detection during the manufacturing process.

REFERENCES

[1] K. V. Wong and A. Hernandez, "A Review of Additive Manufacturing," ISRN Mechanical Engineering, 2012.

[2] A. Uriondo, M. Esperon-Miguez, and S. Perinpanayagam, "The present and future of additive manufacturing in the aerospace sector: A review of important aspects," Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering. 2015.

[3] I. Yadroitsev, P. Krakhmalev, and I. Yadroitsava, "Selective laser melting of Ti6Al4V alloy for biomedical applications: Temperature monitoring and microstructural evolution," Journal of Alloys and Compounds, 2014.

[4] W. Du, Q. Bai, Y. Wang, and B. Zhang, "Eddy current detection of subsurface defects for additive/subtractive hybrid manufacturing," International Journal of Advanced Manufacturing Technology, 2018.

[5] M. Montazeri, R. Yavari, P. Rao, and P. Boulware, "In-process Monitoring of Material Cross-Contamination Defects in Laser Powder Bed Fusion," Journal of Manufacturing Science and Engineering, vol. 140, no. November, 2018.

[6] R. Patel et al., "Imaging Material Texture of As-Deposited Selective Laser Melted Parts Using Spatially Resolved Acoustic Spectroscopy," Applied Sciences, 2018.

[7] N. Boone, C. Zhu, C. Smith, I. Todd, and J. R. Willmott, "Thermal near infrared monitoring system for electron beam melting with emissivity tracking," Additive Manufacturing, 2018.

[8] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, 2017.

[9] D. H. Timo Ojala, Matti Pietikainen, "Performance evaluation of texture measures with classification based on Kullback discrimination of distributions," international conference on pattern recognition, vol. 1, pp. 582–585, 1994.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, 2004.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 2005.

[12] A. Krizhevsky and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Neural Information Processing Systems, 2012.

[13] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," International Journal of Computer Vision, 2015.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Parallel Distributed Processing, p. 1:318–362, 1986.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2014.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," arXiv:1506.01497, Jun. 2015.

[17] M. R-cnn, P. Doll, and R. Girshick, "Mask R-CNN," arXiv:1703.06870v3, 2018.

[18] Joseph Redmon and Ali Farhadi, "YOLOv3: An Incremental Improvement," arXiv: 1804.02767,2018.

[19] Liu, Wei , Anguelov, Dragomir et.al, "SSD: Single Shot MultiBox Detector," Lecture Notes in Computer Science, 21–37, 2016.