

Sand-bed defect recognition for 3D sand printing based on deep residual network

Lan-xiu Wang, *Xuan-pu Dong, Shu-ren Guo

State Key Laboratory of Materials Processing and Die & Mould Technology, School of Materials Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract: The 3D sand printing (3DSP), by binder jetting technology for rapid casting, has a pivotal role in promoting the development of the traditional casting industry as a result of producing high-quality and economical sand molds. This work presents an approach for monitoring and analyzing powder sand-bed images to serve as a real-time control system in a 3DSP machine. A deep residual network (ResNet) is used to classify the defects occurring during the powder spreading stage of the process. Firstly, a pre-trained network was applied as the initial parameter; then it was fine-tuned on the labelled defective sample dataset to accomplish the task, which defines the sand-bed defects induced in the 3DSP processing. Furthermore, the recognition and positioning of sand-bed defects were readily achieved by dividing the sand-bed images into blocks. Experiments show that the fine-tuned network has a 98.7% classification accuracy on the validation dataset of sand-bed defects and 95.4% recognition accuracy for the sand-bed images.

Key words: 3D sand printing; sand-bed defects; deep residual network; sand-bed images

CLC numbers: TG221; Document code: A; Article ID: 1672-6421(2021)04-344-07

1 Introduction

Three-dimensional sand printing (3DSP) technology is increasingly becoming a significant instrument in the rapid sand casting^[1]. Compared with the traditional sand-casting process, it overcomes many limitations in traditional mold design which was used to assemble many parts in the mold.

3DSP machines operate by spreading a thin layer of sand powder over a build plate using a spreading roller. After powder spreading, the nozzle sprays binder in locations corresponding to a 2D slice of a 3D part. After the jetting is complete, the build plate is lowered, another layer of powder is spread, and the process repeats until the part is finished^[2]. Any defects occurring in processing affect the quality of the production. Figure 1 shows the powdered sand bed with defects. Human inspection and correction are commonly used to avoid defects, but it lacks intelligent recognition. One of the best methods is to establish a monitoring and analysis system for the working process of 3DSP to form a negative feedback process of 3DSP molding, and the key is to identify the defects of the sand bed.

There has been extensive work on the monitoring of the additive manufacturing (AM) build process^[3]. Computer vision algorithms and machine learning are widely applied to monitor the melt pool and the powder bed. Scime et al.^[4] studied molten pool defects generated by laser beams in selective laser sintering using a supervised machine learning method and computer vision. Scime and Beuth^[5-6] used deep learning, especially convolutional neural network (CNN), to automatically detect and classify the powder bed's powder state. Yuan et al.^[7] used the improved convolutional neural network to carry out real-time monitoring on the quality of laser track weld, and finally achieved the purpose of



*Xuan-pu Dong

Male, born in 1964, Professor, mainly engages in teaching and scientific research of casting process, application and development of materials and additive manufacturing in casting industry for a long time.

E-mail: dongxp@hust.edu.cn

Received: 2021-06-01

Accepted: 2021-07-08

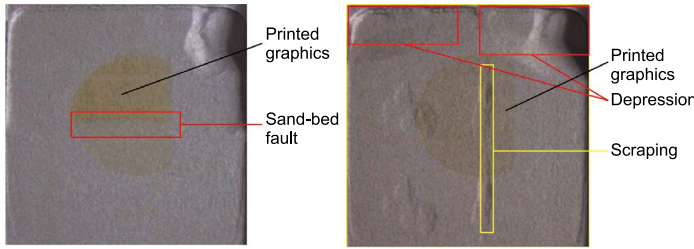


Fig. 1: Sand-bed defects occurring in powdering process

predicting track width, continuity and so on. These are all examples of automated monitoring using neural networks in the field of additive manufacturing, most of which are focused on selective laser sintering and welding quality evaluation. In additive manufacturing, because of its high degree of automation and continuous production, the use of machine vision has a higher application value [8]. In this work, a similar method could be used to detect sand-bed defects. However, there are still some unresolved issues due to the different materials (i.e., different particle size and physical properties between sand and metal powders).

The particles used in this study are generally relatively coarse, with the particle size of about 70–100 mesh [9], while the metal powder mentioned above is generally relatively fine. Because the spacing caused by large particles will affect the final recognition accuracy, the finer the particles, generally speaking, the better the recognition efficiency. The residual neural network (ResNet) [10] is selected as the basic recognition network according to the characteristics of the sand mold. The greatest advantage of this network, compared with the traditional neural network, such as VGG net, is that its high recognition accuracy can still be guaranteed when the number of layers of neural network increases. In the meantime, the deeper the network, the more the features can be extracted, which enhances the learning ability and generalization ability of the network. At the same time, the corresponding image block segmentation measures were designed according to the characteristics of the sand particles. This kind of image segmentation can not only increase the sample size to a certain extent, but also make the final recognition results realize a simple positioning effect [11].

ResNet is a convolution neural network framework proposed by He et al [10]. Compared with other convolutional neural network structures, ResNet proposes the residual learning unit, as shown in Fig. 2. The output feature of each unit, which is the input of the next unit, is changed from $F(x)$ to $F(x)+x$. Therefore, the network learning expressed as $F(x)$ is no longer the feature value, but the differences of feature values between the input and the output (called residual). Compared with the early network structures, residual learning is more sensitive to the output changes; thus, it could be trained more effectively. Therefore, a pre-trained ResNet was used in this work. ResNet is an example of convolutional neural network (CNN) and requires large data sets and significant computational resources to train; fortunately, transfer learning allows for a pre-designed and pre-trained ResNet to be applied to a unique classification problem. Experimental results confirmed that the method proposed is effective on the 3DSP sand-bed defects recognition [12].

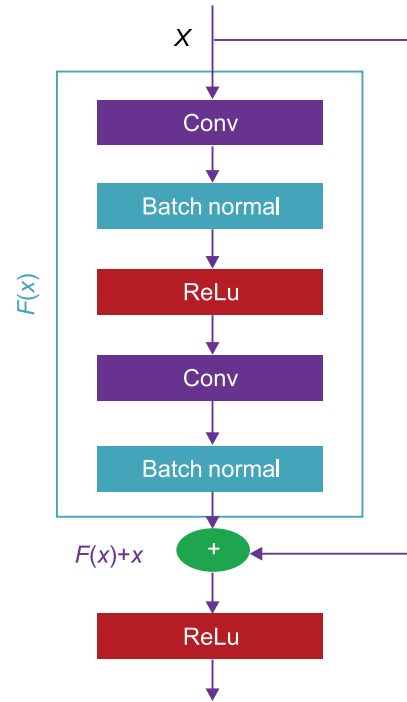


Fig. 2: A residual learning unit

2 Theory

2.1 Deep residual network

The basic network used in the study is ResNet18. The proposed network is built by four modules, a full connection layer, and a softmax layer. Figure 3 presents the structure of the proposed network. Each of the modules is composed of two residual units. The network extracts the image features from the input images and outputs a 1×3 vector. The output vector expresses the probabilities that the input image belongs to each category. The softmax vector is calculated by the following equation to normalize the input vector V , which is the output of the full connection layer:

$$s_i = \frac{e^{V_i}}{\sum_j e^{V_j}} \quad (1)$$

where V_i represents the i th element in V , and $\sum_j e^{V_j}$ represents the sum of exponents of all the elements. Thus, the equation can convert the input interval to the output interval between 0 and 1.

2.2 Strategy of fine-tuning proposed network

Usually, a CNN requires large datasets (on the order of 10^5 – 10^6 samples) for training [13]. However, the constructed sand-bed defects dataset is relatively small. Therefore, a fine-tuning strategy is applied to train the network [14], which could help the network achieve enhanced performance in the extraction of high-level features.

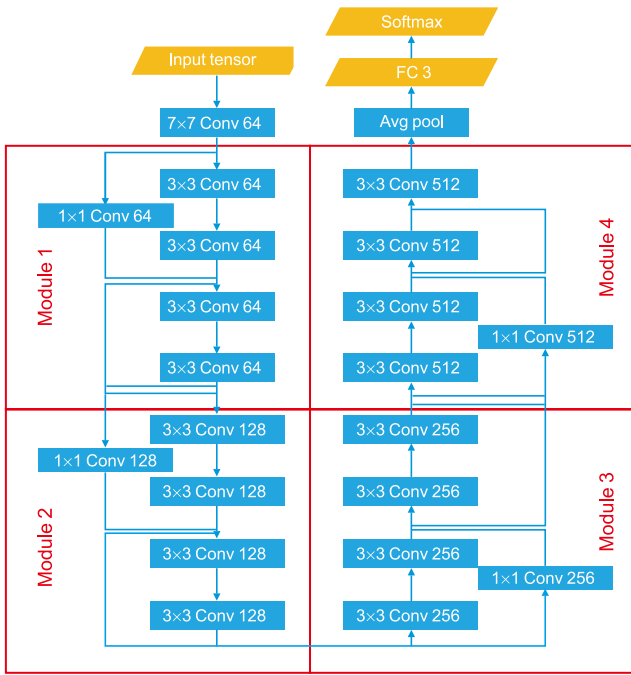


Fig. 3: Structure of purpose network

The fine-tuning of the neural network was carried out using the following training strategy (Fig. 4):

- (1) Obtain the pre-trained network parameters to initialize the purposed network, namely, the feature knowledge learned by the pre-trained network is transferred to the field of defect recognition.
- (2) For the task of sand-bed defects recognition, re-initialize the number of output categories of the full connection layer to adapt to defects.
- (3) Freeze the two shallow modules of the proposed network, named Module 1 and Module 2, and keep their

parameters from changing during training.

- (4) Use the sand-bed defects dataset as input data, then retrain the Module 3, Module 4, and full connection layer with selected loss function and optimization algorithm.

There are two advantages in terms of training the network through the above strategy. On the one hand, the number of parameters to be trained is reduced. Therefore, the time consumed for a single step shortens. On the other hand, the proposed network with the initial parameters inherited from the pre-trained network enables us to extract the basic features of the image. Therefore, the number of steps required for training also decreases.

The loss function was used to evaluate the degree to which the predicted value of the model is different from the real value. Usually, the smaller the loss function value, the better the performance of the model. Among them, the cross entropy loss function is effective in multi-classification problems and is very effective in unbalanced data set distribution, which is suitable for the application scenarios of this study. Thus, the loss function selected is the cross-entropy loss function [15], which is calculated based on the following equation:

$$L(\omega) = \frac{1}{N} \sum_i \sum_c^M y_{ic} \log(P_{ic}) \quad (2)$$

where ω represents the parameter needed to be trained in the network, M represents the number of the categories, y_{ic} represents whether the input's actual category belongs to Type c or not, if so, $y_{ic}=1$, otherwise, $y_{ic}=0$, and p_{ic} represents the predicted probability for the input's category of c calculated by the network.

The purpose of training is to find the value of ω that minimizes the loss function $L(\omega)$. The optimization algorithm by Adam [16] was used to iteratively update ω , and its iteration is as follows:

$$\begin{aligned} m &= \beta_1 m + (1 - \beta_1) \partial L \\ v &= \beta_2 v + (1 - \beta_2) (\partial L)^2 \\ \omega_k &= \omega_{k-1} - \alpha \frac{m}{\sqrt{v} + \epsilon} \end{aligned} \quad (3)$$

where α is the learning rate of the optimization algorithm, and represents the magnitude of each update of ω . By adjusting different learning rates and comparing the influence of them on the convergence rates of the loss value and accuracy, as shown in Fig. 5, it is found that either too large or too small a learning rate has a bad effect on the network. When the learning rate is 0.0001, the convergence speed of the loss function is faster than that of the other networks, and the accuracy rate is also improved steadily, regardless of the full training network or the pre-trained network. Consequently, the learning rate is set as 0.0001. The three hyperparameters in the Adam optimization algorithm, named β_1 , β_2 , and ϵ , were set to 0.9, 0.9999, and 10^{-8} , respectively, according to the literature [16].

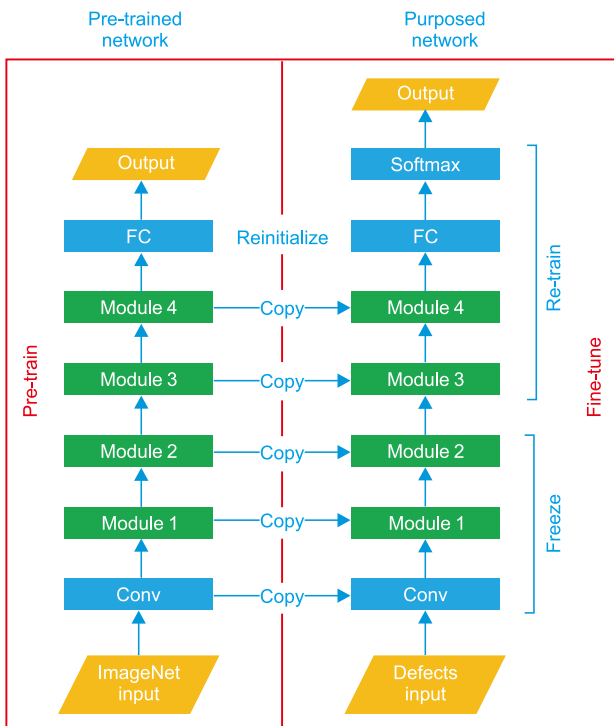


Fig. 4: Strategy diagram of fine-tuning purposed network

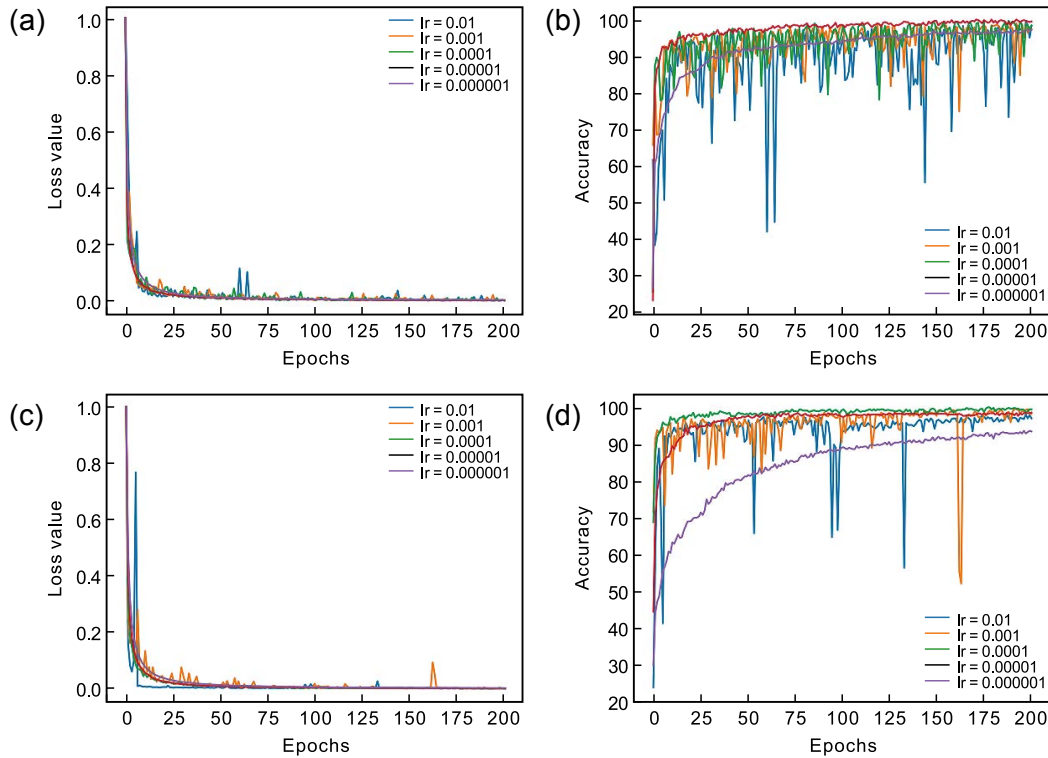


Fig. 5: Influence of learning rate on value of loss function and accuracy: (a) fully training loss function; (b) fully training accuracy; (c) fine-tuned loss function; (d) fine-tuned accuracy

3 Experiment

The method needs to learn from the marked sand-bed defects dataset to establish the relationship between the sand-defect image and the category. The image data were collected on the HXTC 200 3DSP printing machine. A schematic representation of the machine is shown in Fig. 6. The particle size of the molding sand was 150 μm–214 μm (70–100 mesh), the machine build plate measured 200 mm × 200 mm, and the thickness of each layer was 0.3 mm–0.5 mm. The resolution of the camera was set between 450 μm/pixel–600 μm/pixel. Besides, the image size after cropping was 450 pixels × 450 pixels. Consequently, each pixel represents a 450 μm × 450 μm field of view. During printing, images were automatically captured after the fusion of each layer.

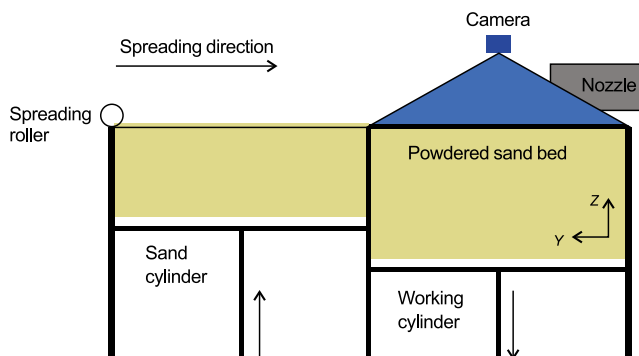


Fig. 6: Schematic diagram of 3DSP machine (the arrows indicate moving direction of schematic components following printing of current layer)

3.1 Definition of sand-bed defects

The sand-spreading process of 3DSP is divided into two parts:

- (1) The sand cylinder ascends to provide certain content of molding sand, and the working cylinder descends to provide forming space.
- (2) The spreading roller spreads the molding sand on the working cylinder through the rotation and translation.

Based on the above two crucial points, the study defines the categories of 3DSP sand-bed defects, as shown in Table 1 and Fig. 7.

The size and position of the printed area may be different in each layer. As long as the defects appear on the sand layer, they may affect the final printed result. Therefore, there is no need to specifically distinguish the defects between the printed area and the non-printed area. In addition, because the color of the printed area is lighter and close to the color of the non-printed area, they are not distinguished in the training data, which is also confirmed by the final recognition results in Table 2 and Table 3.

The depressions and faults caused by sand deficiencies often appear on the top of the sand bed. Once these overlap with the printed graphics, the produced sand mold would be staggered and deformed. As the sand scraping defects often penetrate into the entire sand bed, the surface of the printed sand mold could be consequently impaired, and a significant reduction of the structural strength of the printed parts makes it challenging to meet the quality requirements.

3.2 Sand-bed defects dataset creation

The collected sand-bed images may contain multiple different defective and non-defective areas, such as those shown in Fig. 1.

Table 1: Categories of sand-bed defects and their color codes indicated

Name	Description	Overlay color codes
Sand deficiency	The top layer of the sand bed has depressions and faults because of insufficient sand feeding	Red
Sand scraping	The sand scrapes parallel to the sand spreading direction in the sand bed because of surface defects of the powder spreading roller	Yellow
Normal	No sand-bed defects	Transparent

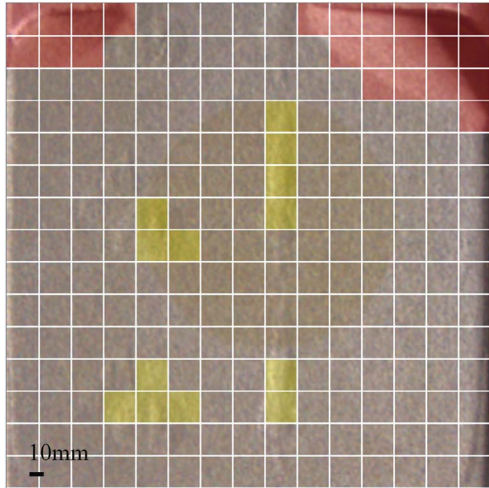


Fig. 7: Sand-bed image with labels, with red and yellow blocks belonging to category of sand deficiency and sand scraping, respectively

For this reason, training of the network is performed using image patches. The advantage of such processing is that, on the one hand, the data size is expanded, which makes the original small sample size increase; on the other hand, the method of

using image patches can make the distribution of dataset more balanced, remove a lot of redundant information, and improve the quality of samples. To develop the training dataset, sand-bed images are split into patches and each image patch is marked by a ground-truth label as one of the three categories listed in Table 1. The patches and their attached labels are stored in a database for access by the methods. The image patches with labels are shown in Fig. 7. The current training dataset includes a total of 5,600 image patches consisting of 2,100 non-defective blocks, 1,400 of sand deficiency, and 2,100 of sand-scraping. The sand-bed defects classification network can learn how to extract image features and establish the relationships between image features and categories based on the provided dataset. Moreover, the network can reduce the influence of printed graphics and particles on the classification by correctly marking patches when constructing the dataset.

3.3 Training results

The experimental computer has an i7-9750H processor, 16 GB of installed RAM, and GeForce RTX 1660Ti graphics card. The experiment used the PyTorch deep learning framework to implement, train, and test the algorithm on the Windows 10

Table 2: Results of sand-bed image recognition with fine-tuned network

Sand-bed image after detection						
	Accuracy	95%	96%	95%	96%	95%
	Time cost	1.74 s	1.72 s	1.72 s	1.73 s	1.69 s

Table 3: Results of sand-bed image recognition with fully trained network

Sand-bed image after detection						
	Accuracy	90%	93%	97%	98%	92%
	Time cost	2.59 s	2.47 s	2.49 s	2.45 s	2.50 s

operating system.

The effectiveness of the sand-bed defect-recognition network was verified by the following two aspects: (1) 10% of the sand-bed defect data were extracted randomly as the verification set to validate the classification effect of the network, and a comparison between a fine-tuned network with the above strategy and fully trained network was executed. (2) Several sand-bed images were taken to evaluate the network.

3.3.1 Single sand-bed defect classification

Confusion matrices and the receiver operation characteristic (ROC) graph are a metric commonly used to evaluate machine learning algorithms [17]. Figure 8 shows the convergence of the loss function value and accuracy with the step size. The fine-tuned network produced 98.7% accuracy after 200 steps. However, the fully trained network only produced 95.6%

accuracy after 200 steps. Figure 8 also compares the fine-tuned network with the fully trained network on loss value and total accuracy. The fine-tuned network has a higher starting point and faster convergence. It has also achieved a higher overall accuracy. Figure 9 shows the confusion matrix of the classification results in sand-bed defects. The diagonal elements on the confusion matrix represent the proportion of correctly classified samples. The correctly classified samples account for most of the tested samples, which confirms the effectiveness and accuracy of the classification model.

The ROC graph of the network is shown in Fig. 10. The area closer to the upper left corner of the ROC curve will result in a better model. The fine-tuned network has a good recognition effect for every category of the sand-bed defects. Moreover, it also receives superior evaluations in every category over that of the fully trained.

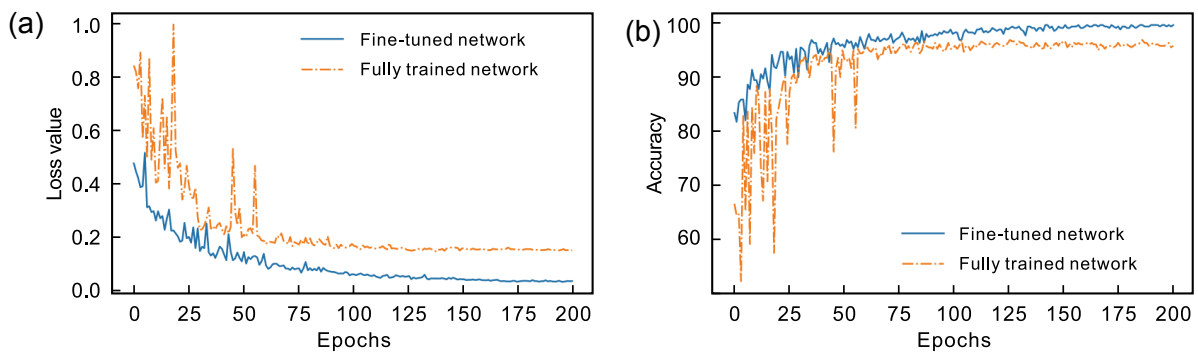


Fig. 8: Comparisons between fine-tuned network and fully trained network: (a) loss value; (b) accuracy

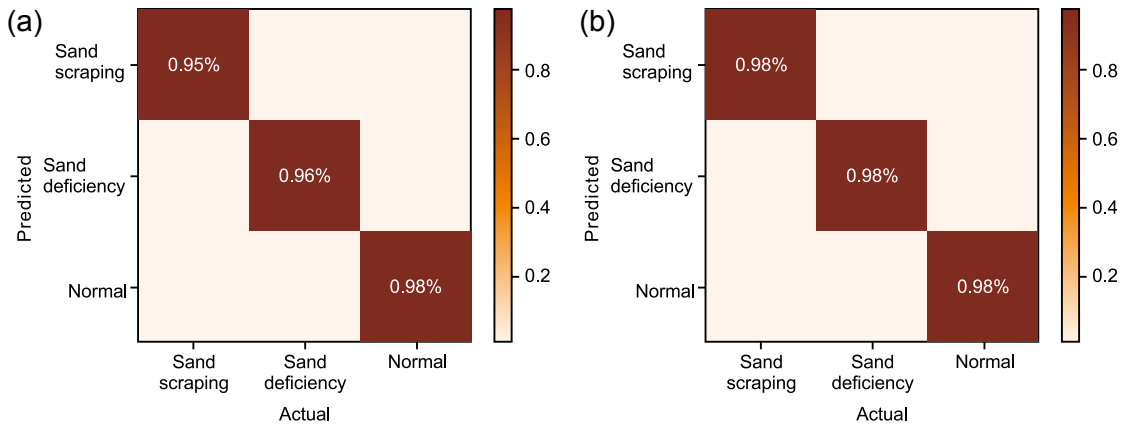


Fig. 9: Confusion matrix of proposed network: (a) fully trained network; (b) fine-tuned network

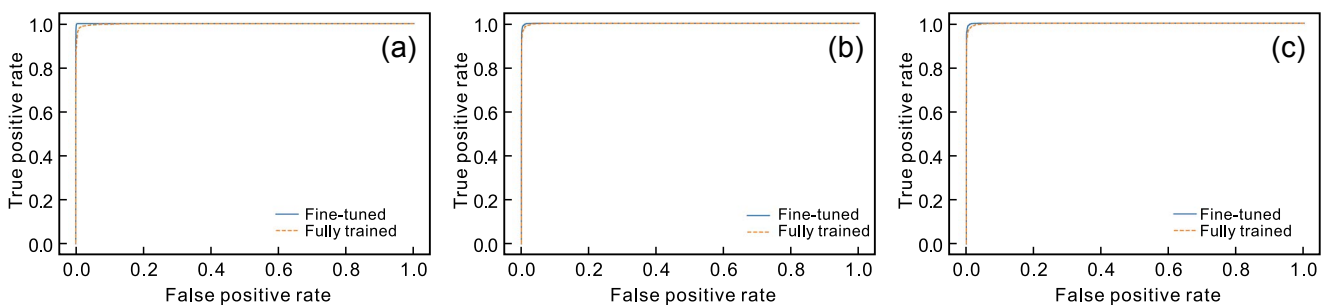


Fig. 10: ROC curve of each category of sand-bed defects: (a) sand deficiency; (b) sand scraping; (c) normal

3.3.2 Multiple sand-bed defects recognition

The sand-bed image usually contains multiple defective or non-defective areas, while the convolutional neural network trained can only discriminate the defective types of a single area. During implementation of the sand bed monitoring algorithm, the pre-processed sand-bed image is divided into 10×10 patches through the Python imaging library (PIL). Then, each patch is input into the trained network for classification. Five collected sand-bed images are recognized to test the recognition method, and the results are shown in Table 2 and Table 3. For 500 split image patches, 477 (95.4%) of them were correctly identified by the fine-tuned network, and the average recognition time for each image was 1.72 s, while in the fully training network, 470 (94%) were correctly identified, and the average recognition time for each image was 2.5 s. By contrast, it can be seen that the recognition effect of the pre-trained network after fine-tuning is better than that of the fully trained network both in terms of the recognition accuracy and time. Misrecognition mostly occurs for the sand scraping defects. Further research will be conducted in this aspect.

4 Conclusions

(1) Through the fine-tuning strategy of the classical convolution neural network model, the deep neural network with the ResNet transfer learning pre-trained by imageNet database has a good effect in the sand bed defect recognition task of 3D sand printing.

(2) For the fine-tuned ResNet18 network, when the two modules Module 1 and Module 2 are frozen and the cross entropy loss function is adopted, the accuracy of single defect classification is 98.7%, and the average recognition accuracy and recognition time of multi-defect classification are 95.6% and 1.72 s, which are better than the 95.4% accuracy of single defect classification, 94% accuracy of multi-defect classification, and average time of 2.5 s of the full training network. The average recognition time of the former is nearly 30% lower than that of the latter, which has a very important advantage in the real-time printing process.

(3) The network has a good recognition effect for 3D sand printing sand bed defects, but it cannot accurately locate the defects, and the positioning ability of the model can be optimized in the future.

(4) Once the defects are recognized in the printing process, the signal can be fed back to the printer machine for the spreading roller to relay the sand, and the corresponding adjustment can be made according to the defect type of the sand layer.

References

- [1] Upadhyay M, Sivarupan T, El Mansori M. 3D printing for rapid sand casting – A review. *Journal of Manufacturing Processes*, 2017, 29: 211–220.
- [2] Son H J, Jang S, Lee H J, et al. Mechanical properties and reliability of sand casting 3D printing materials. *Korean Journal of Materials Research*, 2020, 30(1): 38–43.
- [3] Walker J M, Prokop A, Lynagh C, et al. Real-time process monitoring of core shifts during metal casting with wireless sensing and 3D sand printing. *Additive Manufacturing*, 2019, 27: 54–60.
- [4] Scime L, Beuth J. Using machine learning to identify in-situ melt pool signatures indicative of flaw formation in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 2019, 25: 151–165.
- [5] Scime L, Beuth J. A multi-scale convolutional neural network for autonomous anomaly detection and classification in a laser powder bed fusion additive manufacturing process. *Additive Manufacturing*, 2018, 24: 273–286.
- [6] Scime L, Beuth J. Anomaly detection and classification in a laser powder bed additive manufacturing process using a trained computer vision algorithm. *Additive Manufacturing*, 2018, 19: 114–126.
- [7] Yuan B D, Guss G M, Wilson A C, et al. Machine-learning-based monitoring of laser powder bed fusion. *Advanced Materials Technologies*, 2018, 3(12): 1800136.
- [8] Xu X, Vallabh C K P, Krishnan A, et al. In-process thread orientation monitoring in additive manufacturing. *3D Printing and Additive Manufacturing*, 2019, 6(1): 21–30.
- [9] Hackney P M, Wooldridge R. Characterisation of direct 3D sand printing process for the production of sand cast mould tools. *Rapid Prototyping Journal*, 2017, 23(1): 7–15.
- [10] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, USA, 2016: 770–778.
- [11] Suzuki S, Abe K. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 1985, 30(1): 32–46.
- [12] Badrinarayanan V, Kendall A, Cipolla R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017, 39(12): 2481–2495.
- [13] Xu K L, Darve E. Solving inverse problems in stochastic models using deep neural networks and adversarial training. *Computer Methods in Applied Mechanics and Engineering*, 2021, 384, 113976, doi.org/10.1016/j.cma.2021.113976
- [14] Becherer N, Pecarina J, Nykl S, et al. Improving optimization of convolutional neural networks through parameter fine-tuning. *Neural Computing and Applications*, 2019, 31(8): 3469–3479.
- [15] Murphy K P. *Machine learning: A probabilistic perspective*. MIT Press, Massaridge, Massachusetts, USA, 2012.
- [16] Yi D, Ahn J, Ji S. An effective optimization method for machine learning based on ADAM. *Applied Sciences*, 2020, 10(3): 1073.
- [17] Fawcett T. An introduction to ROC analysis. *Pattern Recognition Letters*, 2006, 27(8): 861–874.