Deep learning-based defect detection for hot-rolled strip steel

Binglong Si \(^1\), Musha · Yasengjiang \(^1\)*, Huawen Wu \(^2\)

\(^1\)College of Mechanical Engineering, Xinjiang University, Urumqi, Xinjiang 830047, China

\(^2\)Experimental Training Center, Xinjiang Institute of Technology, Aksu City, Xinjiang, 843100, China

\(*\)Corresponding author’s email: sunwenxj@sina.cn

Abstract. Defects in the strip's surface can have an impact on the strip company's product sales. As a result, it's crucial to spot flaws on the strip steel's surface. The Faster R-CNN network is structurally upgraded for the issue of strip steel surface defect detection by employing FPN for feature fusion in the feature extraction layer, RoI Align instead of RoI Pooling in the pooling layer, and Softer-NMS in the final prediction fully connected layer. In addition, for training, transfer learning is employed. The experimental findings demonstrate that the suggested technique mAP outperforms the original Faster R-CNN by 7.6%, and the detection time is fast enough for strip steel detection online.

1. Introduction

Hot-rolled steel strip is an indispensable raw material in the manufacturing industry. Defects in strip steel can lead to production problems in steel manufacturing companies and thus affect their reputation and product sales, and the quality of the steel surface is one of the most important indicators of its quality. This results in different sizes, shapes, and textures of defects on the steel surface during the production process due to differences in manufacturers, processing equipment, and processing techniques [1].

In the past, the inspection of steel surfaces was mainly carried out by manual visual inspection, which had disadvantages such as high labor intensity and low confidence level. With the increasing maturity of computer vision technology, the visual inspection of product surface defects and defect calibration have brought about rapid development. In particular, the hottest deep learning techniques have gradually started to be used on various engineering problems. In the last three decades, computer vision-based automatic inspection of steel surfaces has been considered as one of the most effective methods and has gradually developed significant research results. To identify pinholes on the surface of mitered joints, D Choi et al. [2] suggested a pinhole detection method for steel plate pictures based on Gabor filters and morphological characteristics. Based on a multivariate discriminant function, Liu Weiwei et al. [3] devised a technique for identifying surface flaws in strip steel. Minor flaws are efficiently recognized by segmenting the sample picture and extracting relevant characteristics. D Martin et al. [4] proposed and applied a multimodal defect detection approach to identify strip steel faults. By calculating the similarity between the sample and template pictures, Wang Heying et al. [5] used to detect defects on the strip surface by calculating the similarity between the sample image and the template image. Wang et al. [6] suggested a Faster R-CNN method combining multi-level features. To address the disadvantage of limited precision in identifying surface defects on the workpiece, Dai et al. [7] improved the Faster R-CNN and the detection effect is much higher than the traditional method.

The network model of Faster R-CNN was selected and improved by analyzing the types of defects
on the steel strip surface. The following three improvements are made to the algorithm model: (1) utilizing the Feature Pyramid Networks (FPN) [8] structure for feature fusion in the feature extraction network; (2) utilizing RoI Align (Region of Interest Align) [9] instead of RoI Pooling in the pooling layer; and (3) incorporating the Softer-NMS (Softer-Non Maximum Suppression) [10] module.

2. Faster R-CNN Model

As demonstrated in Figure 1, the faster R-CNN is made up of four sections. Module 1 is a feature extraction network that extracts features and sends the extracted feature maps to Modules 2 and 3; Module 2 is a regional suggestion network (RPN) that generates a series of prediction candidate boxes in the feature map and passes them to Module 3; Module 3 is the RoI Pooling layer that transforms the incoming feature maps into a uniform fixed size and sends the result to Module 4. Module 4 is a prediction and classification network that classifies and predicts feature maps as they arrive.

Faster R-CNN has several advantages, such as high detection accuracy and relatively good generalization ability, but it also has some disadvantages. For example, only the last layer of the feature extraction network is used for prediction. In addition, after convolution and pooling, the image produces a small-resolution feature map, and the two quantization operations in RoI Pooling result in a loss of information, which makes it difficult to identify small defects on the strip surface. According to the characteristics of strip surface defects, some improvements are mainly made.

3. Improving the Faster R-CNN

Faster R-CNN has three enhancements. (1) using the FPN (Feature Pyramid Networks) structure for feature fusion in the feature extraction network; (2) using RoI Align instead of RoI Pooling in the pooling layer; (3) adding the Softer-NMS structure in the final prediction bounding box regression layer. As shown in Figure 2, the red box is the improved part.
3.1 Feature fusion
Due to the differences in defect size, shape, and distribution, the feature extraction process on the original image has to undergo several co-evolutions, and after pooling, the information of small target defects in the image will be seriously lost, resulting in smaller defects being easily missed. As a result, merging the underlying characteristics with the high-level features increases the expression of the features and, as a result, improves target identification accuracy, particularly for tiny targets. Figure 3 shows the fusion process using the FPN structure at the feature extraction layer, which includes bottom-up, top-down, and lateral connections. The bottom-up process is the process of feature extraction by ResNet50, and the FPN records the layers whose size and number of channels do not change as a stratum, which is divided into five strata, C1-C5. The top-down process is the process of fusing the high-level feature map with the bottom feature map, starting from C5, using bilinear interpolation upsampling to adjust the feature map to the same size as C4. C4 and C3 are adjusted similarly. The horizontal linkage has two steps. Firstly, the feature maps of the C2-C5 strata are convolved 1×1 so that the number of channels in the upper and lower strata is the same, and secondly, the fusion method is the sum of the corresponding pixel points. The final step removes the overlap generated by upsampling by means of a 3 x 3 filter, resulting in a new feature map, noted as P2, P3, P4 and P5.

The improved Faster R-CNN feeds the fused feature maps P2–P5, and the feature map P6 obtained by maximum pooling in the C5 layer feature map, into the RPN network. Single-scale anchor frames with different scales (1:1, 1:2, 2:1) are preset at each pixel position of these five feature maps, respectively. The pixel areas of the anchor boxes vary with the scale of the feature maps, and are arranged from small to large 16²,32²,64²,128²,256², respectively.

![Figure 3. Feature fusion structure diagram](image)

3.2 RoI Align
The Faster R-CNN model employs RoI Pooling, which is based on the notion of mapping the network's regions of interest to matching spots in the feature map based on the input image. The regions of interest are of diverse sizes and include floating-point numbers since they are formed by an offset correction and selection process using region schemes of various sizes and scales. Furthermore, feature maps of a specified size are required for successive fully linked network inputs. As a result, the floating-point regions of interest are first mapped to the feature maps' corresponding locations, and then the quantized feature maps are scaled to a fixed size. During the RoI pooling phase, there are two rounding quantization steps that introduce a mismatch of information between the regions of interest and the derived features, lowering detection accuracy. To reduce the error of RoI Pooling, introduce RoI Align.
RoI Align eliminates the rigorous quantization process of RoI Pooling. It avoids quantizing the region of interest, retains all floating-point data, and then calculates the exact value of several sampling points by bilinear interpolation, combining the maximum or average value of several sampling points to obtain the final value. Thus, the whole feature aggregation process becomes a continuous operation to improve performance while preserving as much of the original region as possible. The improvement is obvious to regions with small defects. The principle is shown in Figure 4. The purple arrows indicate the values of the cyan points obtained by bilinear interpolation. The dashed arrow indicates the maximum value of the 4 cyan points obtained. The red arrow points to the final value of the 4 cyan points after merging them into a feature.

Figure 4. Bilinear interpolation and maximum or average pooling should be aligned.

3.3 Softer-NMS
Non-maximum suppression algorithms (NMS) are important in object detection. In Faster R-CNN networks, the RPN network generates a large number of predicted candidate frames, in which many duplicate candidate frames are positioned on the same target, and NMS is used to eliminate these duplicate frames and extract the true candidate frames. On the other hand, NMS has a drawback. When two defects are close to each other, the two candidate frames containing the defective region will produce overlapping regions, and the candidate frames with lower scores will be removed, leading to a missed detection situation and reducing the average detection accuracy of the algorithm. Softer-NMS improves the drawbacks of NMS by ranking the candidate frames according to the confidence score. Softer-NMS also weights the candidate boxes within the predicted labeled variance, giving higher confidence scores to the candidate boxes with better prediction positions and lower classification confidence to the predicted candidate boxes with lower suppression scores. The modified network structure is shown in Figure 5, where the red boxes represent the newly added parts compared with the original model, and AbsVal (absolute value layer) is used to make the predicted bounding boxes Gaussian distributed, which can improve the prediction accuracy of the real boxes.

Figure 5. Faster R-CNN with Softer-NMS

4. Sources of data sets
The dataset was obtained using the Northeastern University (NEU) database and has six defect categories for hot-rolled strip steel. Plaque (Pa), cracks (Cr), rolled oxide (RS), pitting (PS), inclusions (In), and scratches (Sc). There are 300 sheets for each defect type, for a total of 1800 images. The size of each image is 200 x 200 pixels. As shown in Figure 6, the five defects selected for training and validation in this paper are plaque (Pa), rolled oxide (RS), pitting surface (PS), inclusions (In), and scratches (Sc).
5. Training process

The experimental equipment is a computer with Windows 10 installed and a computer configuration CPU model i7-10875H, 8 cores, 16 threads, 16 GB of memory, GeForceRTX3060 GPU, and a 6 GB video memory software environment configuration. PyTorch 1.7.1, Python 3.7, cuda11.1 and cudnn10.2.

For comparative testing, three network models were trained. Yolov3 [11], Faster R-CNN and Improved Faster R-CNN were the three. During the training period, 1500 images were divided 7:3, with 450 images in the validation set and 1050 images in the training set. The training methods were: online data augmentation was used in the preprocessing phase; data augmentation was done by flipping and scaling the image aspects horizontally; migration learning was used in training; the weight files were trained from the COCO dataset [12]. Using the SGD optimizer, the learning rate is set at 0.1 at the beginning and is adjusted to 0.3 times the original after two epochs.

Convergence occurs very fast because migration learning is used during training. The loss of the model reached a minimum after 20 epochs were trained. The loss values and learning rates are shown in Figure 7, where the loss value (loss) is on the left, the learning rate value (lr) is on the right, and the horizontal coordinate is the number of training sessions. The loss value during training eventually converges to 0.15.

\[
L = \frac{1}{N_{cls}} \sum_{i} L_{cls}(p_i, p_i^*) + \frac{1}{N_{reg}} \sum_{i} p_i L_{reg}(t_i, t_i^*)
\]

\[
L_{cls} = \left[ p_i \log(p_i) + (1 - p_i) \log(1 - p_i) \right]
\]

Figure 7. Training loss and learning rate graph

The training of the network includes two stages of training: RPN model training and Fast R-CNN network training. RPN model training includes classification and regression of prediction candidate frames. Fast R-CNN network training includes classification and regression of prediction targets. The loss function of RPN is shown in Equation (1).

\[L_{cls}\]—denotes the classification loss; \[L_{reg}\]—denotes the loss of the bounded regression frame. \[N_{reg}\]—denotes the number of anchor positions of approximately 2400; \[N_{cls}\]—denotes 256 samples in a batch.

The formula for \[L_{cls}\] is as in equation (2).
$p_i$—the likelihood that the $i$th anchor will be the real label; $p^*_i$ —if the $i$th anchor is a positive sample, it is 1, otherwise it is 0.

The formula for $L_{reg}$ is as in equation (3)

$$L_{reg}(t_i, t^*_i) = \sum_i smooth_{t_i}(t_i - t^*_i)$$  (3)

$t_i$—denotes the bounding box regression parameter for predicting the $i$th anchor; $t^*_i$—denotes the true box corresponding to the $i$th anchor.

The Fast R-CNN loss calculation uses a multiclassification loss as in Equation (4)

$$L(p,u,t^*,v) = L_{cls}(p,u) + \lambda[u \geq 1]L_{loc}(t^*,v)$$  (4)

$p$—is the soft max probability distribution $p = (p_0, ..., p_k)$ predicted by the classifier; $u$—corresponding target true class label; $u$—bounding box regressor predicted by the regression parameters of the corresponding class $u(t^u_x, t^u_y, t^u_w, t^u_h); v$—bounding box regression parameters corresponding to the true target $(v_x, v_y, v_w, v_h)$.

6. Analysis of test results

The effectiveness of the model detection is assessed in terms of the class-wide average correct rate mAP, which is the combined weighted average of the class detection and the combined weighted average of the average correct AP.

Precision is calculated as in equation (5) Recall is calculated as in equation (6).

$$Precision = \frac{TP}{TP + FP}$$  (5)

$$Recall = \frac{TP}{TP + FN}$$  (6)

$TP$ (True Positive)—defective areas are correctly detected as defects; $FP$ (False Positive)—non-defective areas are identified as defects; $FN$ (False Negative)—defective areas are considered non-defective.

| Model       | In  | Pa  | PS  | RS  | Se  |
|-------------|-----|-----|-----|-----|-----|
| YOLOv3      | 75.57 | 88.4 | 70.4 | 41.5 | 83.8 |
| Faster R-CNN| 76.1  | 88.3 | 76.7 | 44.3 | 81.9 |
| Ours        | 79.84 | 89.6 | 81.1 | 61.1 | 93.7 |

The detection accuracy of YOLOv3 and Faster R-CNN for rolling oxide flaws is quite poor, as Table 1, whereas the enhanced Faster R-CNN increases by 16.8%. The revised Faster R-CNN has improved while comparing the accuracy of each defect individually. The experimental findings show that the enhanced network increases detection accuracy significantly.

| Model       | mAP(%) | FPS |
|-------------|--------|-----|
| YOLOv3      | 71.93  | 37  |
| Faster R-CNN| 73.46  | 26  |
| Ours        | 81.06  | 21  |

Table 2 shows that, the model in this paper has improved in mAP by 9.13% and 7.6% compared to the Faster R-CNN and YOLOv3 algorithms respectively. Compared to the improvement in detection accuracy, there is only a small reduction in detection speed. However, it meets the practical requirements.
Table 3. Comparison of the three improvement methods

| Model                                      | mAP/ % |
|--------------------------------------------|--------|
| Faster R-CNN+ResNet50                      | 73.46  |
| Faster R-CNN+ResNet50+FPN                 | 77.44  |
| Faster R-CNN+ResNet50+FPN+RoI Align       | 79.61  |
| Faster R-CNN+ResNet50+FPN+RoIAlign+Softer-NMS | 81.06  |

From Table 3, it can be seen that using the FPN structured feature extraction network is very helpful in improving the detection accuracy improvement, with a 3.94% increase in mAP. mAP improved by 6.11% using the RoI Align pooling method and FPN compared to the original Faster R-CNN model, and several of the proposed improvement methods had a certain improvement in mAP, and the final network mAP improved by 7.6% compared to the original network, and the experiments proved that the improvement methods were indeed useful. Figure 8 shows a graph of the detection effect of the improved model.

7. Conclusion

For the band defects with small area and irregular shape, an improved Faster R-CNN model is proposed, including the use of FPN structure in ResNet50 to improve the ability to extract detailed information; the use of RoI Align instead of RoI Pooling to improve the ability to locate defect areas; and adding Softer-NMS to the fully connected layer of the predicted bounding regression box to reduce the probability of missed detection by the network. According to the training results, it can be concluded that the test effect of the modified Faster R-CNN model has greatly improved. The next phase will look at algorithms for pruning and increasing attention to further improve detection speed and accuracy.

Reference

[1] Sharifzadeh, M., Amirfattahi, R., Sadri, S., Alirezaeae, S. og Ahmadi, M. (2008) Detection of steel defect using the image processing algorithms. Í The International Conference on Electrical Engineering.pp. 1-7.
[2] Choi, D.-c., Jeon, Y.-j., Yun, J. P. og Kim, S. W. (2011) Pinhole detection in steel slab images using gabor filter and morphological features. Applied Optics, 50(26): 5122-5129.
[3] Weiwei, L., Yunhui, Y., Jun, L., Yao, Z. og Hongwei, S. (2008) Automated on-line fast detection for surface defect of steel strip based on multivariate discriminant function. Í 2008 Second International Symposium on Intelligent Information Technology Application,pp. 493-497.
[4] Martín, D., Guinea, D. M., García-Alegre, M. C., Villanueva, E. og Guinea, D. (2010) Multi-modal defect detection of residual oxide scale on a cold stainless steel strip. Machine Vision and Applications, 21(5): 653-666.
[5] Wang, H., Zhang, J., Tian, Y., Chen, H., Sun, H. og Liu, K. (2018) A simple guidance template-based defect detection method for strip steel surfaces. IEEE Transactions on Industrial Informatics, 15(5): 2798-2809.
[6] Wang, H., Wang, J. og Luo, F. (2020) Research on surface defect detection of metal sheet and strip based on multi-level feature faster r-cnn. Mechanical Science and Technology for Aerospace Engineering, 2020.
[7] Dai, X., Chen, H. og Zhu, C. (2020) Research on surface defect detection and implementation of
metal workpiece based on improved faster r-cnn. Surface Technology, 49(10): 362-371.

[8] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B. og Belongie, S. (2017) Feature pyramid networks for object detection. Î Proceedings of the IEEE conference on computer vision and pattern recognition.pp. 2117-2125.

[9] He, K., Gkioxari, G., Dollár, P. og Girshick, R. (2017) Mask r-cnn. Î Proceedings of the IEEE international conference on computer vision.pp. 2961-2969.

[10] He, Y., Zhang, X., Savvides, M. og Kitani, K. (2018) Softer-nms: Rethinking bounding box regression for accurate object detection. arXiv preprint arXiv:1809.08545, 2(3).

[11] Redmon, J. og Farhadi, A. (2018) Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[12] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. og Zitnick, C. L. (2014) Microsoft coco: Common objects in context. Î European conference on computer vision.pp. 740-755.