Metal Character Detection Based on Improved Deformable Detr

Li Liu, Jiawei Zeng†

School of computing University of South China Hengyang, China

Abstract

To address the problem of inefficient and inaccurate inspection of workpiece characters In this paper, a fusion YOLOv5, we make datasets of the metal workpieces and propose character recognition methods based on deep learning. Firstly, Adding the attention mechanism (CBAM) to the backbone module of the Deformable DETR model to increase the initial picture feature extraction ability. By combining the benefits of Smooth-L1 loss and GIoU loss, the intercepted characters are passed through the ResNext50 model,thus further enhancing the recognition of characters and improving the overall recognition accuracy. The identification accuracy of the average metal workpiece based on the improved model can reach 84.5%, and the identification time of a single object is about 0.5s, which can be actually used in production.

Keywords

deep learning; Deformable Detr; CBAM; character recognition; ResNext50;

1. INTRODUCTION

Nowadays, image processing technology is almost closely related to people's lives, and image recognition technology is greatly brought convenience to our life, in many industrial production processes to reduce the labor intensity of staff, reduce the error rate of the industrial production processes, and greatly improve the production efficiency. The OCR generally contains two stages: text detection and text recognition. Traditional OCR text detection mainly focuses on image processing methods for text positioning, such as feature description algorithms such as HOG [1] algorithm, and the processing objects are often limited to clear imaging and regularly arranged document images, which cannot well handle the complex background images. Subsequently,a series of deep learning-based text detection algorithms have sprung up in the field of text detection in OCR. The CTPN algorithm proposed by Document [2] uses anchor regression mechanism to effectively detect the target area; Document [3] proposes the SegLink algorithm and introduces Segment and Linking to realize the detection of text with a certain rotation Angle;Literature [4] proposed Faster R-CNN algorithm to realize the shared convolutional features of regional suggestion network and detection network, greatly improving the operation speed; the YOLO series algorithm proposed by literature [5~9] unified the candidate box extraction, feature extraction, target classification and positioning in a neural network, with the advantages of fast operation speed and high detection accuracy. Most of the above algorithms rely on the convolutional neural network (CNN) and develop on its basis, but the convolutional neural network focus more on the local features and ignores the global features, so the CNN-based detection algorithm is insufficient to detect small-scale defects. Transformer[10] is able to capture a large range of feature information, so more focused on global features.As scholars applied Transformer to various fields [11,12], Carion et al. [13] proposed the Transformer-based end-to-end object detection model DETR model. The DETR model transforms the target detection task into an ensemble sequence prediction task, extracting features by simple CNN and using the encoding-decoding structure of the base Transformer in parallel. The proposal and application of DETR model provide a brand new idea for solving the target detection task. However, due to the restrictions of the Transformer attention module in handling image feature maps,its convergence is slow with limited feature spatial resolution. To alleviate these problems, literature [14] proposes the Deformable DETR model that enhances the ability of sparse spatial sampling by using deformable convolutions. The Deformable DETR can obtain

ICCEIC2022@3rd International Conference on Computer Engineering and Intelligent Control
EMAIL: 2249691300@qq.com (Jiawei Zeng)

better performance than DETR, and the model is easier to converge, reducing the training cycle. For the OCR text recognition phase, multiple characters can be identified using a combination of LSTM [15] and CTC [16], and a single character by using VGG [17] or ResNet[18].

This article uses Deformable DETR to identify the position of workpiece symbols and enhance the recognition accuracy of the model by adding an attention mechanism (CBAM [19]) to the module of the backbone of the Deformable DETR model to increase the feature extraction capability of the model. In addition, this paper enhances the robustness of the model to small data sets and noise by using a combination of Smooth-L1 loss function [20] and GIoU loss function [21] as edge regression loss to improve the regression accuracy and model training efficiency for small size defects.Finally, the characters with the identified positions are clipped and sent to the ResNext50[22] network for symbol recognition. This paper ensures the speed of character recognition of metal workpieces, but also ensures the accuracy of recognition, which can meet the current industrial application needs.

2. ALGORIYHM DETR

2.1. ResNext

The ResNext network is generated by combining the stacking idea of the VGG network with the idea of the Inception[23]'s split-transform-merge. The core idea of ResNext is grouping convolution, turning the single-channel convolution into a multi-branch and multiple convolution, and finally merging the features extracted from the multiple branches. Where the number of branches is variable and is controlled by the variable base (Cardinality). As shown in Figure 1 is the ResNext block of Cardinality=32. ResNext makes use of the topology of submodules to improve accuracy without increasing parameter complexity. In addition, ResNext uses the topology structure to enhance the portability and universality of the model.

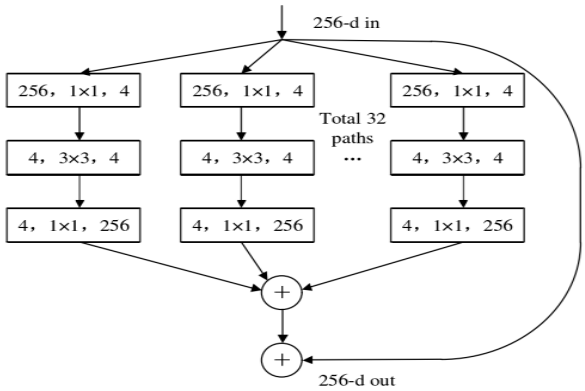

Figure 1. ResNext's branch structure diagram

2.2. Deformable DETR

Figure 2 depicts the structure of DETR

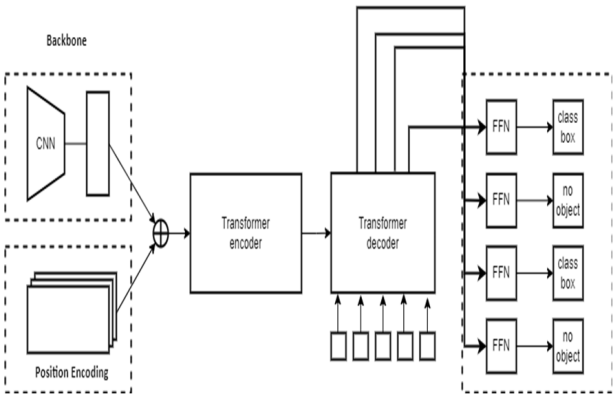

Figure 2. DETR illustraton of model

However, when DETR is initialized, the attention model is almost uniform for all pixel weights on the feature graph, leading to the need to focus on sparse meaningful positions with long training period learning. This results in a long training model and unfriendliness to small target identification. In order to make the weight of the encoder initialization no longer a unified distribution, that is, no longer the similarity to all key calculations, but to the more meaningful key calculation. By borrowing the idea of Deformable convolution [24], Deformable DETR was then proposed, and its model is illustrated in Figure 3.

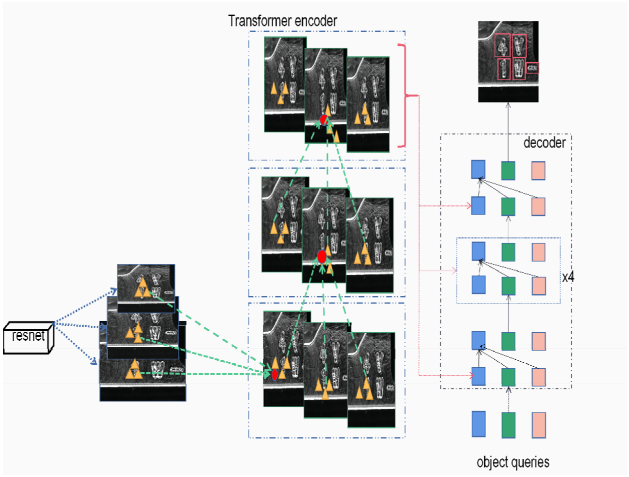

Figure 3. Deformable DETR model diagram

Deformable DETR fusion Deformable conv of sparse spatial sampling with Transformer correlation modeling capabilities in the overall feature map pixels, the model focuses on the sampling location of small sequences as a pre-filtering. In addition, the DETR model uses single-scale features, while the Deformable DETR uses multiscale features to enhance the detection effects, especially for small targets. Its structure is illustrated in Figure 4.

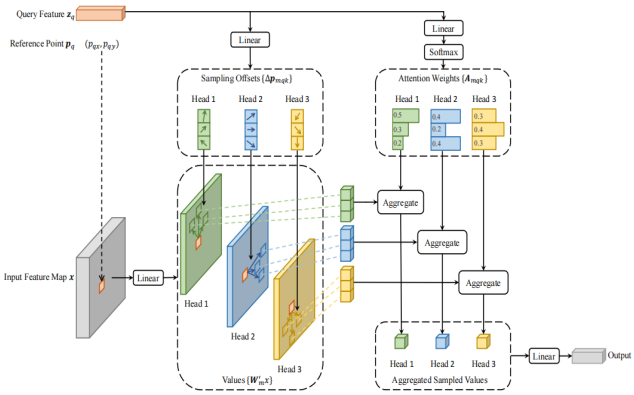

Figure 4. Attention Module for Deformable DETR

3. ALGORITHM IMPROVEMENT

3.1. CBAM attention mechanism

By introducing CBAM in the Deformable DETR backbone feature extraction network, the model's attention to strings is improved. CBAM contains both channel attention and spatial attention, which can enhance the feature extraction performance of the backbone network and enhance the detection accuracy, light weight and high efficiency. Figure 5 shows the structure of the CBAM module.

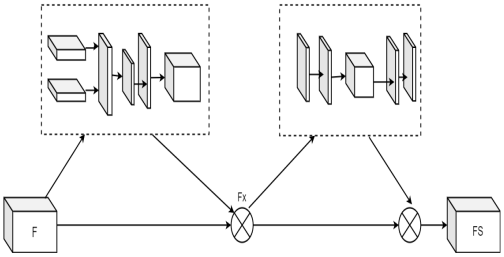

Figure 5. CBAM model structure diagram

First, the module of channel attention receives the feature map F, uses maximum pooling and average pooling to get the information of the individual channels of the feature map, and then the MLP superimposes the obtained parameters, and then activated through the Sigmoid function to get the channel attention feature Mc(F), which is calculated as

$$M_C(F)=\sigma\begin{bmatrix}MLP(AvgPool(F))+\\MLP(MaxPool(F))\end{bmatrix} \quad (1)$$

Among them, $\sigma(\bullet)$ denotes the Sigmoid activation function, MLP denotes the multi-layer perceptron, $Avgpool(\bullet)$ is the average pooling, and $Maxpool(\bullet)$ stands for the maximum pooling. The feature graph F_X is formed by multiplying the channel attention features and the input features element by element, and then the feature map is entered into the spatial attention module.The spatial feature map is generated through average pooling and maximum pooling, with 7×7convolution and Sigmoid function activation, finally, it is multiplied element by element with F_X to gain the spatial attention feature map F_S.

$$F_S=\sigma\left[f^{7\times7}\left(\begin{bmatrix}AvgPool(F_X);\\MaxPool(F_X)\end{bmatrix}\right)\right]\otimes F_X \quad (2)$$

Among them, $\otimes$ for the element-by-element multiplication. Introducing CBAM in the extracted feature extraction network can solve the problem of no attention preference in the original network, increasing the network's focus on the target to be detected during the detection phase. The modified Deformable DETR model is illustrated in Figure 6.

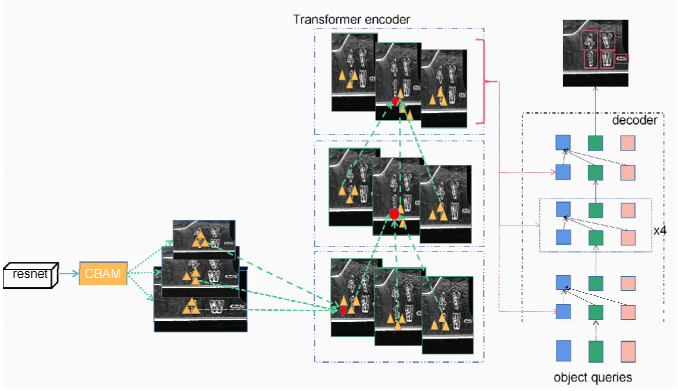

Figure 6. The improved Deformable DETR model structure diagram

4. LOSS FUNCTION DESIGN

The Deformable DETR model decodes and predicts the embedded output vector of the decoder,The model is optimized by continuously reducing the deviation between the predicted and labeled values through the loss function.To enhance detection precision, this paper combines the Smooth-L1 and GIoU loss functions as a regression loss to perform a prediction regression on the detection border.

4.1. Classification of loss

The cross-entropy loss function provides a good account of the distance between the predicted output and the expected output. by continuously learning to optimize the probability of the model predicting each category, and the distance between label categories in one-hot form, so as to achieve the correct classification.Assuming that the predicted output is the probability distribution P_i and C_i is the desired output, the cross-entropy loss function is defined as follows.

$$L_{BEC}(p_i, c_i) = -\sum_{i=0}^{n} \log p_i(c_i) \qquad (3)$$

4.2. Return to the loss

In this essay, the combination of Smooth-L1 loss function and GIoU loss function enables the algorithm to not only stably regression on small size defects, improve the detection accuracy, but also quickly converge to higher accuracy.

$$L_{Smooth-L1}(b_i, \hat{b}_i) = \begin{cases} \sum_{i=0}^{n} 0.5(b_i - \hat{b}_i)^2, if \left|\hat{b}_i - b_i\right| < 1 \\ \sum_{i=0}^{n} \left|b_i - \hat{b}_i\right| - 0.5, other \end{cases} \qquad (4)$$

Where, b_i represents the expectation box of the i-th index, $\hat{b}_i$ is the prediction box of the i-th index. Compared with the L1 and the L2 loss function, the Smooth-L1 loss function combines the advantages of both, as defined as equation (4). Early in the training process, there is an excessive gap between the expectation frame and the prediction frame,prediction box gradient is effectively constrained by the Smooth-L1 loss function. thus avoiding the gradient explosion, allowing the model to converge quickly and enhancing the robustness; while in the late training period, there is a problem that the distance between the prediction frame and the expectation frame is too small, and the derivative of the loss function also exists when fluctuating around 0, allowing the model to converge to a higher accuracy.Therefore, GIoU is also introduced in the calculation of regression loss, which regresses the prediction frame as a whole. Equation (5) shows the calculation process of GIoU. A denotes the prediction frame, B denotes the true frame, and C is the minimum enclosing frame of the prediction frame and the true frame.

$$L_{GIoU} = 1 - \frac{|A \cap B|}{|A \cup B|} + \frac{|C - A \cup B|}{|C|} \qquad (5)$$

5. ANALYSIS AND RESULTS OF EXPERIMENTS

5.1. Experimental data processing and experimental environment

The experiments were carried out on the same hardware environment to demonstrate fairness. (GPU: NVIDIA RTX3060 12GB, CPU: Intel Intel I5-10400F 4.60GHz, RAM: 32GB). The experimental data presented in this essay are the real industrial acquisition data. The data collected from the production line is built into a data set; The data set is annotated using the LabelImg annotation tool,and expand the data set by cutting, enlarging, shrinking, and rotating the pictures.

5.2. Evaluation indicators

The mean average precision (mAP) is applied in this essay as the index for evaluating the model.

$$\mathrm{AP} = \int_0^1 \mathrm{P}(R) dR \qquad (6)$$

$$\mathrm{m}AP = \frac{1}{Q}\sum_{q \in Q} AP(q) \quad (7)$$

Each category obtains AP values by pr curve integration, as shown in formula (6), and then averaging the AP values for all categories is mAP, as shown in formula (7). In purpose of verifying the excellent performance of the improved algorithm proposed in this work,the module improvement process is contrasted using the exact same number of test sets in the same configuration conditions. Table I shows the results. As demonstrated by the data,the improved Deformable DETR model is 7.87% higher than the average accuracy of the original model (mAP@0.5). Figure 7 shows the changes in error score rate, total loss, and mAP for different epochs between the two algorithms during training. Due to the improved Smooth-L1 loss function,As opposed to the previous model, the derivative of the loss function still persists even when the prediction frame and the expectation frame are very close to each other. This allows the model to converge to better accuracy and to regress steadily for small-size defects. Meanwhile, because the GIoU loss function has a gradient in the early stage, combined with the fast convergence of Smooth-L1 loss function in the early stage, the model can converge more quickly and stably and improve the training efficiency.

TABLE I. COMPARISON OF THE MODEL IDENTIFICATION EFFECTS

Model	mAP@0.5
Deformable DETR	0.808
OURS	0.848

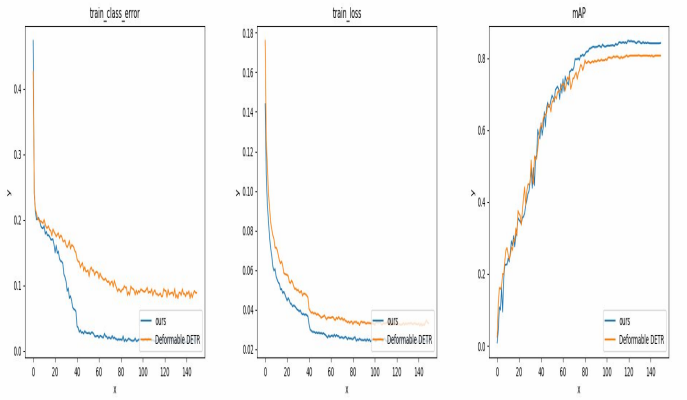

Figure 7. Training results diagram

In the character recognition module, this paper adopts the method of transfer learning, and fine-tuning in the existing model, which can enhance the learning speed of the model and make the model converge faster. In this paper, the four classification models are ResNet34, Resnet50, Resnext50, and Restnext101. After 80 iterations on the same dataset, through Table II, Figure 8, we can find that Resnext50 has the best effect in character recognition, with the highest accuracy and loss rate. Through Table II, it is concluded that the recognition effect of Resnext50 model for the same 50-layer network is much better than that of Resnet50 model. The experiment shows that the network structure of ResNext can increase the ability of the model to extract features. The contrast between the recognition rate of Resnet34 model and Resnet50 model, and between the recognition rate of Resnext50 model and Restnext101 model indicates that the increment in the number of layers of the network does not enhance the recognition rate of the model.

TABLE II. SYMBOL MODEL IDENTIFICATION AND COMPARISON

Model	Acc	loss
Resnet34	0.9730	0.125
Resnet50	0.9376	0.252
Resnext50	0.9764	0.094
Restnext101	0.9362	0.222

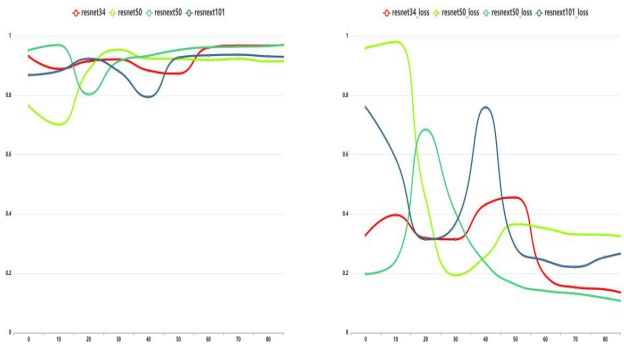

Figure 8. Model accuracy (left) and loss (right) comparison

6. CONCLUSIONS

This paper proposes an algorithm for metal artifact character recognition based on the improved Deformable DETR model. It upgrades the detection competence of the model by integrating the attention mechanism (CBAM), and then the detected image is cut and sent to the ResNext50 network for single character recognition. The experiment indicated that the cascade algorithm presents in this paper meets the detection speed to the needs of actual production, and has a positive role in the on-the-ground of deep learning in the industrial field. However, this paper is aimed at the low number of metal characters. When the future work will explore the situation when there are more characters on the metal, we can consider using the end-to-end text recognition method to reduce character recognition time.

7. REFERENCES

[1] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Ieee, 2005, 1: 886-893.

[2] Tian Z, Huang W, He T, et al. Detecting text in natural image with connectionist text proposal network[C]//European conference on computer vision. Springer, Cham, 2016: 56-72.

[3] Shi B, Bai X, Belongie S. Detecting oriented text in natural images by linking segments[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2550-2558.

[4] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. Advances in neural information processing systems, 2015, 28.

[5] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[6] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.

[7] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[8] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[9] Li C, Li L, Jiang H, et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications[J]. arXiv preprint arXiv:2209.02976, 2022.

[10] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[11] Devlin J , Chang M W , Lee K , et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. 2018.

[12] Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 10012-10022.

[13] Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers[C]//European conference on computer vision. Springer, Cham, 2020: 213-229.
[14] Zhu X, Su W, Lu L, et al. Deformable DETR: Deformable Transformers for End-to-End Object Detection[C]//International Conference on Learning Representations. 2020.
[15] Shi X, Chen Z, Wang H, et al. Convolutional LSTM network: A machine learning approach for precipitation nowcasting[J]. Advances in neural information processing systems, 2015, 28.
[16] Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks[C]//Proceedings of the 23rd international conference on Machine learning. 2006: 369-376.
[17] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[18] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[19] Woo S, Park J, Lee J Y, et al. Cbam: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.
[20] Wei B, Hao K, Tang X, et al. Fabric defect detection based on faster RCNN[C]//international conference on artificial intelligence on textile and apparel. Springer, Cham, 2018: 45-51.
[21] Rezatofighi H, Tsoi N, Gwak J Y, et al. Generalized intersection over union: A metric and a loss for bounding box regression[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 658-666.
[22] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1492-1500.
[23] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
[24] Dai J, Qi H, Xiong Y, et al. Deformable convolutional networks[C]//Proceedings of the IEEE international conference on computer vision. 2017: 764-773.