Optimization Research and Defect Object Detection of Aeroengine Blade Boss Based on YOLOv4

Wei Chen¹, Xintong Zhong¹,∗ and Jing Zhang²

¹College of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao Shandong, China
²Qingdao Beihai Ship Building Heavy Industry CO., LTD., Qingdao Shandong, China

Abstract. This paper proposes an object detection algorithm based on YOLOv4 (You Only Look Once) aiming at detecting the defects of the blade boss inside the aeroengine. The algorithm uses transfer learning to load the weights of the pre-trained model trained on the coco public data set. In order to adapt to the characteristics of small targets and complex structures better in the detection of bosses, the size of the bounding boxes is adjusted through the cluster analysis. In the meantime, the original data set is enhanced through the Mosaic method. The experimental results show that the improved YOLOv4 model increased the detection accuracy by 15.85%, the recall rate by 21%, with an average intersection ratio of 0.75, and the detection performance is significantly better than the data applying the SSD object detection algorithm in the same data set.

1. Introduction

With the accelerated development of Chinese militarization process and the increase of air force, the missions of aircraft are increasing day by day. The aeroengine is the most important component of an aircraft, and its normal operation will directly determine the stable performance of the aircraft. At present, turbofan engine is widely used in aircraft. It improves the ratio of thrust to weight by continuously increasing the temperature in front of turbine, which makes the working temperature of turbine blade very high, and the boss is prone to deformation and dislocation, which causes potential safety hazard to aircraft.

In general, the blade boss is located inside the aeroengine, and the outside is covered by a fan fairing, which can not be directly observed and inspected by naked eyes. Moreover, the internal mechanical structure of aeroengine is complex and precise, and the space is narrow, which increases the maintenance difficulty of blade boss. At present, the maintenance of blade boss mostly adopts the method of taking the engine apart and inspecting it with naked eyes. It costs a lot of manpower and time to dismantle and assemble the blade boss. Moreover, the existing target detection of engine blade is mostly focused on the detection of blade surface defects [1,2], and the defect detection of blade boss is rarely paid attention to.

Therefore, the algorithm proposed in this paper fills the blank of target detection for blade boss from the internal of aeroengine, and greatly improves the maintenance efficiency. Early warning is very important for the maintenance of engine blades, which is of great significance to ensure flight safety and improve the level of China's aviation industry.

In recent years, computer vision algorithm based on deep learning[3] has developed rapidly and achieved good results in the application of object detection, object tracking and semantic
At present, object detection models based on deep learning can be roughly divided into two categories: object detection models based on region proposal, such as R-CNN, Faster R-CNN, R-FCN, etc.; and object detection models based on box regression method, such as YOLO, SSD, etc.

In this paper, the YOLOv4 model proposed by Alexey Bochkovskiy et al. in 2020 is used to detect the object of blade boss, and then the data set is processed by data enhancement, and the default candidate box is improved by clustering analysis method to further improve the accuracy of object detection.

2. YOLO Object Detection Model

The core idea of the YOLO (You Only Look Once) series is very concise, that is, to use the entire picture as the input to the network, to return the location and category of the candidate box directly at the output level.

2.1. Composition of the YOLO Model

The YOLO model, as a check box regression based target detection algorithm, is usually composed of three parts, as shown in Figure 1.

1) Backbone: Features extracted from images, such as VGG and ResNet-50 are common backbone networks;
2) Neck: Used to connect the backbone network to the header structure;
3) DenseHead: Used for dense location-related operations on feature maps, where RPN, Retina, FCOS Heads are representative operations.

![Figure 1. Object detection model based on region proposal.](image)

2.2. YOLOv4 Object Detection Model

YOLOv4 is the fourth generation of YOLO object detection model. Its main purpose is to design a fast object detection system which can be applied to practical work environment and can be optimized in parallel. The model structure of YOLOv4 is shown in Figure 2.

![Figure 2. YOLOv4 model structure.](image)
YOLOv4 uses self adversarial training\cite{15} at the input end. In the backbone feature extraction network, drawing on the experience of CSPnet (cross stage partial networks)\cite{16} in 2019, a new backbone structure CSPDarknet53 is generated, and the Mish activation function is used to improve the accuracy:

\[
Mish = x \times \text{tanh}(\ln(1 + e^x))
\]  

At the same time, SPP (spatial pyramid pooling)\cite{17} and PAnet (path aggregation network)\cite{18} feature extraction structures are used in the feature pyramid. The SPP structure is shown in Figure 3, which can greatly increase the receptive field and separate the most significant context features. PAnet extracts features repeatedly to improve the ability of feature extraction.

\[
\text{CIOU} = \alpha \rho + \gamma - \text{IoU} + \frac{D(b,b^p)}{c^2}
\]  

CIOU takes into account the distance between the target and the candidate frame, the overlap rate, the scale and the distance between the center point of the boundary box, which makes the target regression more stable, and the speed and accuracy of the prediction box regression are higher.

3. Design and Improvement of YOLOv4 Algorithm

3.1. Introduction of Transfer Learning

Transfer learning is a machine learning method, which means that a pretrained model is reused in another task. The main idea of transfer learning is to transfer annotated data or knowledge structure from related fields, and complete or improve the learning effect of target domain or task. At present, transfer learning is widely used in training neural networks. In order to further improve the efficiency of model training, this paper uses the method of transfer learning to load the pretrained model before training.

In view of the characteristics of the data sets in this paper, the coco data sets with good versatility and rich types are selected for transfer learning. The weight of the detection model trained on coco is loaded into the YOLOv4 model, and the model is modified and adjusted continuously during the training, so that the model does not need to be trained from scratch.

3.2. Improvement of Bounding Box Size

Since the network weight of the YOLOv4 pretrained model is trained on coco data set, the coco data set includes 80 kinds of targets, and the detection target of this data set is different from the aircraft blade boss detected in this paper, which does not conform to the prior frame size of this data set. To solve this problem, cluster analysis is used to improve the default bounding box size. Through the cluster analysis of the labeled bounding box in the boss training set, the suitable size of the bounding box is found, and the original bounding box size is modified to make the bounding box size generated by the feature layer better match the target to be measured.

K-means clustering is one of the most commonly used methods in clustering analysis. It is based on the similarity of the distance between points to calculate the best category. In this paper, K-means clustering algorithm is used to cluster the aircraft boss training set data, and the corresponding bounding box size is obtained.
3.3. Data Set Improvement

3.3.1. Production of original data set. The data set of this experiment is to use the endoscope to take the blade boss video from the inside of the aircraft engine. The video is extracted frame by frame to get the blade boss image. The image is cut and filtered, and then the label tool LabelImg is used to make the data set of this paper. LabelImg is a data set specially designed for the series of YOLO algorithms. With this annotation tool, the rectangular box is labeled manually, and the XML tag file storing the coordinate information is obtained, as shown in Figure 4. The format is PascalVOC data set format, a total of 2613 annotated images are obtained.

![Figure 4. Data set production process.](image)

3.3.2. Improvement of data set. Most of the targets detected in the original algorithm are fixed angle shooting targets. Because the video angle captured in this paper is not fixed, the detection difficulty is increased. In order to overcome the shortcomings of single image, such as single angle and poor robustness, this paper uses Mosaic data enhancement method to process the data set. Mosaic data enhancement is a new data enhancement method which combines four training images into one, which can enrich the context information of the image and enhance the robustness of the model.

4. Experimental Results and Analysis

4.1. Experimental Process and Platform
The experimental equipment used in this paper is physical server, Ubuntu operating system, GPU geforce GTX 1080ti graphics card, tensorflow1.13, keras2.1.5, numpy1.17.4, cuda10.0, cudn7.4.1.5, python3.7.

In this experiment, 2309 boss images with a resolution of 300*300 are used as training set and 304 as test set. The initial learning rate is 1e-3, and the preheating learning rate is 1e-5. A total of 50 epochs are trained, and the overlap threshold is set to 0.5.

4.2. Experimental Results

4.2.1. Test results of network model in this paper. In this paper, based on the yolov4 algorithm, K-means algorithm is used to detect the target of the aircraft engine blade boss. The blade boss of each angle is selected as the test result, as shown in Figure 5 (blue is the true value of the detection frame, green is the detection box with correct prediction, and red is the detection box with wrong prediction).

![Figure 5. Object detection result.](image)
4.2.2. Results before and after Modification of Bounding Boxes. In this experiment, K-means clustering analysis is used to modify the bounding boxes size of the model. The size of bounding boxes before and after modification is shown in Table 1.

Table 1. Improve the bounding box size.

| Before improvement | After improvement |
|--------------------|-------------------|
| (12,16)            | (49,79)           |
| (19,36)            | (66,117)          |
| (40,28)            | (81,158)          |
| (36,75)            | (102,52)          |
| (76,55)            | (108,219)         |
| (72,146)           | (120,97)          |
| (142,110)          | (156,88)          |
| (192,243)          | (180,123)         |
| (459,401)          | (216,185)         |

As can be seen from the above table, after the cluster analysis of the tagbox of the data set in this paper, the unsuitable bounding box which were originally too large and too small were removed.

An example of detection results before and after modifying the bounding box is shown in Figure 6, and the P-R curve (precision-recall) is shown in Figure 7. The abscissa in the figure is the recall rate, and the ordinate is the accuracy rate of the model. The area of the shadow part in the figure can reflect the performance of the model. The larger the area, the better the performance of the model.

Figure 6. Detection results before and after.

![Detection results before and after](image6.png)

Figure 7. The P-R curve before and after modifying bounding box.

From the comparison of the test results before and after the modification of the bounding box, it can be seen that the accuracy of the model for boss detection is improved from 76.62% to 89.99%. By adjusting the size of the prior frame, the detection ability of the blade boss is improved, and the detection effect under multi-angle is very good. Moreover, due to the removal of the larger bounding box size, the false detection of the gap similar to the boss is reduced, as shown in Figure 6 of 805. JPG detection. The experimental results show that the modification of the bounding box size improves the detection accuracy.

4.2.3. Results before and after data enhancement. In this experiment, Mosaic method is used to enhance the data set. Figure 8 shows the result of data enhancement.
Figure 8. Picture after Mosaic.

The improved data set is utilized for training, and the results of the two improved experiments are shown in Table 2. At the same time, the training results of SSD model in this dataset under the same experimental conditions are compared.

Table 2. Comparison of experimental results.

| Model                      | Accuracy /% | Recall rate /% | Average merge ratio /% |
|----------------------------|-------------|----------------|------------------------|
| Original SSD model         | 65          | 60             | 60                     |
| Improved SSD model         | 75          | 68             | 65                     |
| Original yolov4 model      | 76.62       | 69             | 70.45                  |
| Modify bounding box        | 89.99       | 86             | 71.13                  |
| Mosaic data enhancement    | 88.39       | 83             | 73.95                  |
| Modify bounding box+Mosaic | 92.47       | 90             | 75.24                  |

As can be seen from the experimental results in the table above, when the overlap threshold is set to 0.5, the accuracy of the improved YOLOv4 model is improved by 15.85% in terms of precision, and 21% in recall rate. However, since mosaic data enhancement does not change the size of the prior frame, the average intersection ratio is not greatly improved. In the comparison of precision, recall rate and average intersection ratio, the experimental model is better than SSD model and improved SSD model, which can prove that the experimental model is more suitable for convex target detection.

Figure 9. The P-R curve after Mosaic and modifying bounding box.

Finally, the P-R curve of the improved model is shown in Figure 9. Compared with the P-R curve of the original model, the proportion of shadow area in the figure is increased, which means that the improved model still maintains a high level of accuracy while increasing the recall rate, which proves that the performance of the improved yolov4 model has been improved to a certain extent.

5. Conclusion

In this paper, the problem of object detection for the internal vane boss of aircraft engine is studied. The recently open-source yolov4 model is applied to the boss detection system, and the transfer learning and mosaic data enhancement are added. In order to adjust the bounding box size, a network model suitable for blade boss detection is obtained, which improves the detection accuracy and system performance. The experimental results show that the improved YOLOv4 model can effectively solve the problem of object detection from the inside of aircraft engine. Next, we consider to classify the images of blade boss according to different shooting angles to further improve the detection performance.
References

[1] Abhishek R.V. Indragandhi, Logesh R.V. Subramaniyaswamy. Detection of Cracks and damage in wind turbine blades using artificial intelligence-based image analytics[J]. Measurement, 2019,147.

[2] A. Joshua, V. Sugumaran. Crack Detection and Localization on Wind Turbine Blade Using Machine Learning Algorithms: A Data Mining Approach[J]. Structural Durability & Health Monitoring, 2019, 13(2).

[3] Santana E, Hotz G. Learning a Driving Simulator [J]. ResearchGate. 2016 (8): 145

[4] Bo Z, Jiashi F, Xiao W, Shuicheng Y. A Survey on Deep Learning-based Fine-grained Object Classification and Semantic Segmentation [J]. International Journal of Automation and Computing, 2017, 14(02): 119-135.

[5] Chinthakindi B M, Mohammad F H, Neeraj D B, Zong W G. Investigations of Object Detection in Images/Videos Using Various Deep Learning Techniques and Embedded Platforms—A Comprehensive Review [J]. Applied Sciences, 2020, 10(9).

[6] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [C]. neural information processing systems, 2015: 91-99.

[7] Dai J, Li Y, He K, et al. R-FCN: object detection via region-based fully convolutional networks [C]. neural information processing systems, 2016: 379-387.

[8] Redmon J, Divvala S K, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection [C]. computer vision and pattern recognition, 2016: 779-788.

[9] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector [C]. European conference on computer vision, 2016: 21-37.

[10] Alexey B, Chien-yao W, Hong-yuan M L, et al. YOLOv4: Optimal Speed and Accuracy of Object Detection [J]. arXiv: Computer Vision and Pattern Recognition, 2020.

[11] Simonyan K, Zisserman A. VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION [C]. computer vision and pattern recognition, 2014.

[12] He K, Zhang X, Ren S, et al. Identity Mappings in Deep Residual Networks [C]. European conference on computer vision, 2016: 630-645.

[13] Lin T, Goyal P, Girshick R, et al. Focal Loss for Dense Object Detection [C]. International conference on computer vision, 2017: 2999-3007.

[14] Tian Z, Shen C, Chen H, et al. FCOS: Fully Convolutional One-Stage Object Detection [C]. International conference on computer vision, 2019: 9627-9636.

[15] Bekoulis G, Deleu J, Demeester T, et al. Adversarial training for multi-context joint entity and relation extraction [C]. Empirical methods in natural language processing, 2018: 2830-2836.

[16] Wang C, Liao H M, Yeh I, et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN [J]. arXiv: Computer Vision and Pattern Recognition, 2019.

[17] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[18] Liu S, Qi L, Qin H, et al. Path Aggregation Network for Instance Segmentation [C]. Computer vision and pattern recognition, 2018: 8759-8768.