An Optimized SSD Target Detection Algorithm Based on K-Means Clustering

Yonggang Chi(✉), Jialin Fan, Bo Pang, and Yuelong Xia

Harbin Institute of Technology, Harbin 150000, China
chiyg@hit.edu.cn

Abstract. In response to the problem that the default box size and shape of the SSD network model need to be manually set based on experience and the lack of specificity for different data, this paper uses the k-means clustering method to optimize the default box setting method of the SSD network to make the default box more consistent with the data, enhancing the self-adaptive ability of SSD default box positioning regression, thereby improving detection accuracy and detection speed. The algorithm is applied to actual aluminum defect detection, the defect detection accuracy reaches 77.6% mAP, which is 2.86% higher than the original SSD512 model, and the detection speed is increased from 37 FPS to 39 FPS.

Keywords: SSD network · Target detection · K-means · Deep learning

1 Introduction

Artificial naked eye recognition is a commonly used target detection method. This method has the problems of low work efficiency, and high product cost, and is easily affected by many factors such as the quality of the inspection personnel, the naked eye resolution, and the eye fatigue [1, 2]. With the gradual maturity of machine learning, detection methods based on machine vision have developed rapidly [3]. When using this type of method for detection, not only a series of pre-processing such as denoising of the image but also feature extraction of the image, such as Haar [4], HOG [5], SHIFT [6] and other features extraction have to be done. In addition, in the face of the increasingly complex detection environment, such methods have the problems of single detection target, poor robustness, low efficiency.

In recent years, deep learning algorithms based on convolutional neural networks have performed well in computer vision such as target detection [7], avoiding the difficulty of manually extracting features based on machine learning detection methods. At present, target detection methods based on convolutional neural networks can be divided into two categories: the first two-stage scheme is based on candidate region algorithms, such as RCNN [8], Fast-RCNN [9], Faster-RCNN [10] and other models, to perform target detection through two steps, region proposal and region classification. The second scheme is one-stage scheme which is based on regression algorithms, such as YOLO [11], SSD [12] and so on. Unlike the two-step working mode of the R-CNN series network, one-stage can complete the above two steps in a single step. And the

M. Guan and Z. Na (Eds.): MLICOM 2020, LNICST 342, pp. 148–156, 2021.
https://doi.org/10.1007/978-3-030-66785-6_17

SSD network shows better detection accuracy and detection speed compared with the YOLO network. The SSD network unifies the area selection, image feature extraction and classification into a deep convolutional neural network, which realizes the automatic selection and automatic extraction of the target detection area. This method effectively improves the detection speed and detection accuracy of the detection network. But the default box size and shape of the SSD network model need to be manually set according to experience, and this setting lacks adaptability to different data objects, which may cause to some default boxes not match the real boxes and thus miss the target. To solve this problem, this paper uses the k-means clustering method to improve the setting of the default box. By clustering the calibration box size of the data set, the default box setting that more closely matches the calibration box in the data set is obtained, so that the default box is more precise and streamlined, and this method optimizes the regression positioning process of the SSD default box. The experimental results show that the improved SSD model has obviously improved in accuracy and speed.

2 SSD (Single Shot Multibox Detector)

Unlike the Faster RCNN network that first extracts candidate regions and then extracts candidate region features for classification, the SSD network uses independent convolution kernels to predict target position offsets and target categories on multi-scale feature maps. And unlike each cell in the YOLO network that only predicts two candidate boxes, the SSD network extracts a total of 30 or 36 types of candidate boxes in the multi-scale feature layer.

The structure of the SSD model can be divided into two parts: the basic network and the auxiliary network structure according to the network implementation function, as shown in Fig. 1. The basic network is a standard architecture for image classification. The basic network of the SSD model uses the truncated VGG-16 [13]. The main auxiliary structures include: (a) Multi-scale feature map layer; (b) Convolutional filters; (c) Default box; (d) Non-maximum Suppression (NMS).

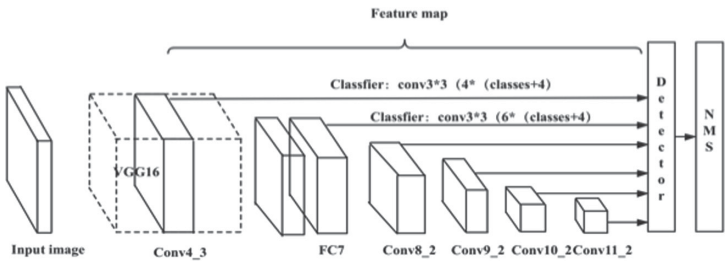

Fig. 1. SSD network structure diagram

For the target detection algorithm such as SSD, the loss function is more complicated than the general convolutional neural network, because in addition to identifying

the target category, it also optimizes the position information of the target in the image. The loss function of SSD is composed of two parts: default box positioning loss (L_{loc}) and confidence loss (L_{conf}). The loss function of SSD is

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}f(x, c) + \alpha L_{loc}(x, l, g)) \tag{1}$$

In the formulation: N is the number of default boxes that meet the IOU greater than a certain threshold, c is the predicted value of category confidence, and α is the weight parameter.

Positioning loss is

$$L_{loc}(x, l, g) = \sum_{i \in pos}^{N} \sum_{m \in \{cx,cy,w,h\}} x_{ij}^{k} smooth_{L1}(l_i^m - \hat{g}_j^m) \tag{2}$$

$$smooth_{L1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases} \tag{3}$$

In the formulation: the g of the real box needs to be encoded to obtain $\hat{g}$ (offset), because the predicted value l is also the encoded value.

The confidence loss is

$$L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^{p} \log(\hat{c}_i^p) - \sum_{i \in Neg} \log(\hat{c}_i^0) \tag{4}$$

$$\hat{c}_i^p = \frac{\exp(c_i^p)}{\sum_p \exp(c_i^p)} \tag{5}$$

SSD networks have great advantages in accuracy and real-time, but SSD networks also have some disadvantages. For example, the very important default box in the SSD network needs to be set according to human experience, the size and shape of the default box can not be obtained directly through learning.

3 Optimization of SSD Network

Although the classic SSD network has reached a high level in detection speed and detection accuracy, there is still room for improvement. The detection performance of the SSD network model is related to the setting of the default box. The speed and accuracy of SSD model detection are affected by the number of default boxes. On the one hand, selecting a smaller number of default boxes can increase the speed of model detection but will reduce the detection accuracy, while selecting a larger number of default boxes will increase the accuracy of the model but will reduce the detection speed. On the other hand, the default box size and shape of the SSD network model are manually set based on experience and lack specificity for different data objects. If the

initial default box size and number are more in line with the characteristics of the marked box in the data set, then the model can accelerate convergence while improving the speed and accuracy of the detection algorithm.

In view of the above problems, in order to obtain a more reasonable default box setting, this paper provides a new idea for the selection of the default box by performing k-means clustering on the calibration box of the training set, making the default box generated during training and prediction more accurate, and training also converges better.

3.1 K-Means Algorithm

k-means [14] belongs to a clustering algorithm. The algorithm accepts an unlabeled data set and then clusters the data into different groups. The k-means algorithm can be defined abstractly: given a series of data(x_1, x_2,..., x_n), each data is d-dimensional, the k-means algorithm divides the n data into k clusters $S = \{S_1, S_2, ..., S_n\}$,$\mu_i$ is the cluster center of each cluster, so that the internal mean square sum of the cluster is minimum, the objective function can be expressed as

$$\arg\min \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2 = \arg\min \sum_{i=1}^{k} VarS_i \tag{6}$$

3.2 The Optimization Method of Default Box Setting

The original SSD network default box generation rules are as follows:

For each cell in each feature map, multiple default boxes are generated in the center of the default box in the original image. Each feature map has a minimum value min_size and a maximum value max_size. The maximum value is the minimum value of the next layer feature map and defines the aspect ratio $\alpha \in \{1, 2, 3, 1/2, 1/3\}$. For the case where the aspect ratio is 1, square default boxes with side lengths of min_size and max_size are generated, and for the case where the aspect ratio is not 1, the width w and height h of the default box are generated as follows

$$\begin{cases} w = \sqrt{\alpha} \times \min\_size \\ h = \dfrac{1}{\sqrt{\alpha}} \times \min\_size \end{cases} \tag{7}$$

For the original SSD network model, each cell of feature map generates 4 or 6 default boxes. These parameters are selected manually. A major advantage of neural networks is that it can reduce the experience requirements for non-professionals. Therefore, k-means clustering method is used to cluster the calibration box of the training data, which provides new ideas for the selection of the default box. At the same time, by choosing a more accurate default box, it helps to improve the accuracy of network detection.

The original k-means algorithm uses Euclidean distance to find the closest center point, minimizing the distance from each point to any center point. In the SSD network, the clustering dimension of the calibration data is based on the width and height of the calibration box. If use Euclidean distance as k-means distance function, a large calibration box will produce a larger loss than a small calibration box. But in target detection we want to generate a default box with a higher overlap rate with the real box, that is, We hope that the IOU score is higher. The IOU here is the parallel ratio of the default box (DB) and the calibration true box (GT), which can be expressed as

$$IOU = \frac{DB \bigcap GT}{DB \bigcup GT} \tag{8}$$

In order to reduce the influence of the size of the calibration box, the improved k-means distance function is

$$d(box, centroid) = 1 - IOU(box, centroid) \tag{9}$$

Using the improved k-means to cluster the calibrate box of aluminum defect data, SSD512 has 7 feature layers, so k takes 7, and the clustering result is shown in Fig. 2. It can be seen from the figure that the cluster center 1 is a small size box, so the min _size of the first layer feature map is set to the abscissa of the cluster center 1. Since the size and shape change of the small target is small, the aspect ratio of 2 and 1/2 of the first feature layer are removed to reduces model complexity. The min_size of the second feature layer is set to the abscissa of cluster center 2, and the aspect ratio of 3 and 1/3 of the second feature layer are removed. Since the widths of the other five clustering centers are all 512, and these clustering centers detecting large-sized targets, and the calibration box of these clustering center vary greatly in width and height, so the min_size of the last one feature layer is set to 512, and an aspect ratio of 15 is added. The default box of the remaining feature layers are set according to the original model. The average IOU of the original network default box and the aluminum defect calibration box is 36.54%. After the optimized k-means clustering algorithm, the average IOU is 40.05%, and the default box number is reduced from 36 to 33.

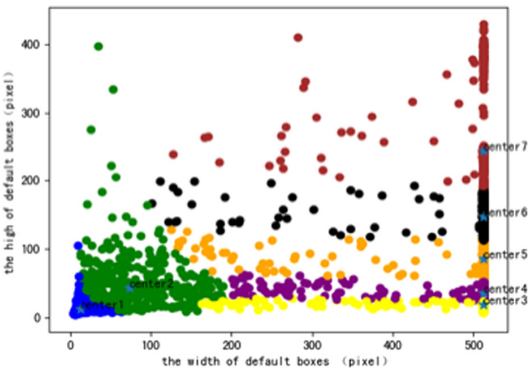

Fig. 2. K-means clustering result

4 Experimental Results and Analysis

4.1 Experimental Data

The aluminum defect detection images used in this article are from the Alibaba Cloud Tianchi platform. In the actual production process of aluminum, due to the influence of many factors, defects such as scratches, dirty spots, and paint bubbles will occur, and these defects will seriously affect the quality of aluminum. The data is derived from the actual production data of the enterprise, and each picture contains one or more defects. The pixel of the aluminum material defect image is 2560 * 1920. There are the following ten types of defects. Table 1 shows the number of various types of defective samples, and divides the data in Table 1 into a training verification set and a test set according to 9:1. It can be seen from the table that the number of each type of defects is very uneven, such as the number of samples in jet flow is very few, and the number of sample defects such as bottom leakage is much more than other categories. The scale of the defect vary largely, the size of some defect is very small, which means that the defects only occupy a small part of an image, and the size of some defect is large. These all increase the difficulty of designing the flaw detection algorithm.

Table 1. The number of various types of defects

Category	Non-conductive	Scratch	Corner bottom	Orange peel	Bottom	Jet	Paint bubble	Pit	Variegated	Dirty spots	Total
Number	360	128	346	173	538	86	82	407	365	251	2736

4.2 Model Training

The experimental platform configuration is shown in Table 2. The size of the model input picture is 512 * 512. The batch size is set to 32. The non-uniform learning rate decay strategy of multistep is used to train the SSD model. Through multiple experiments, the basic learning rate was finally selected to be 0.0001, and the step value of multi-step learning was 40,000 and 80,000, and the attenuation coefficient was 0.1. In order to increase the amount of data and prevent overfitting during training, data enhancement methods such as random mirroring, random cropping, rotation, translation, and grayscale transformation are used for the training data.

Table 2. Experimental platform configuration

Name	Configuration
Deep learning framework	Pytorch1.4
CPU	Intel Core i9-7900X, 3.3 GHz
GPU	NVIDIA GeForce GTX2080Ti, 11G
RAM	64 GB

Use the above parameters to train the SSD network and the optimized SSD network, and record the network loss value every 20 steps. Figure 3 is the original SSD training loss curve, and Fig. 4 is the optimized SSD training loss curve. From the figure, the loss curves show a downward trend with the increase of the number of iterations, and the loss value decreases greatly at the beginning of training, indicating that the learning rate is appropriate. And the loss curve tends to be stable after training to a certain stage, indicating that the model begins to converge and meets the expected requirements. It can be seen from Figs. 3 and 4 that the optimized SSD network has a smaller loss value and faster training convergence.

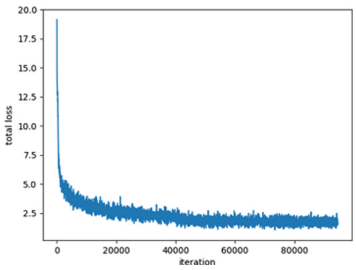

Fig. 3. Original SSD training loss

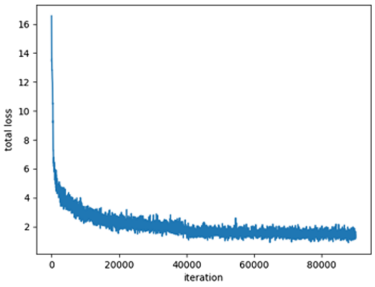

Fig. 4. Optimized SSD training loss

4.3 Experimental Results and Discussion

On the actual aluminum data set, the SSD network before and after optimization was tested and compared, using average precision (AP) and mean average precision (mAP) as comparison indicators.

Figure 5 compares the AP of each category before and after optimization. Table 3 shows the improvement ratio of AP in each category. It can be seen from Fig. 5 and Table 3 that the optimized SSD network greatly improves the detection average precision of the category with lower original detection average precision, among them, the improvement ratio of the "scratch" category has reached 37.8%. It is of great significance to improve the detection AP of category with original low AP. The improved default box of the SSD network model is more in line with defect calibration box, the regression positioning process of the network default box for defects is optimized, and many defect categories that are difficult to detect by the original network can be detected.

Judging from the mean average precision of all categories, the optimized SSD network mAP increased from 74.82% of the original SSD to 77.68%, and the detection speed was increased from 37fps to 39fps. In addition, using the optimized SSD network for detection, the position regression is more accurate compared to the original network. The overall performance of the algorithm is shown in Table 4.

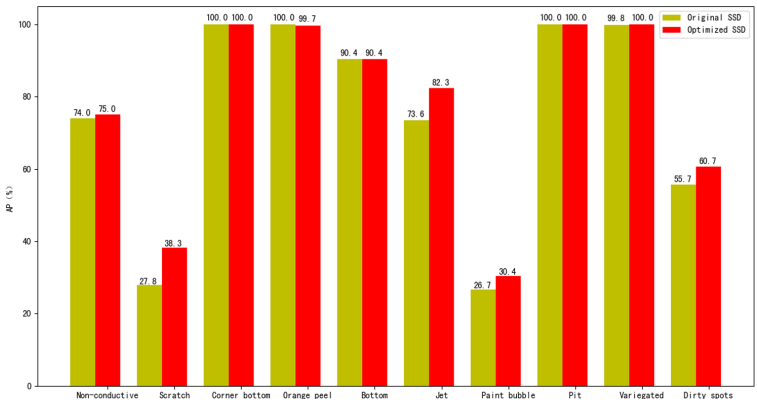

Fig. 5. Average precision

Table 3. Average precision and improvement ratio

Category	AP(%)		
	Original SSD	Optimized SSD	Improvement ratio
Non-conductive	74.0	75.0	+1.3%
Scratch	27.8	38.3	+37.8%
Corner bottom	100.0	100	0
Orange peel	100.0	99.7	−0.3%
Bottom	90.4	90.4	0
Jet	73.6	82.3	+11.8%
Paint bubble	26.7	30.4	+13.9%
Pit	100	100	0
Variegated	99.8	100	0.2%
Dirty spots	55.7	60.7	+9%

Table 4. SSD performance comparison before and after optimization

Algorithm	mAP(%)	FPS
Original SSD	74.82	37
Optimized SSD	77.68	39

5 Conclusion

This article first describes the advantages of target detection methods based on convolutional neural networks, and introduces several classic target detection models. Among them, SSD network is an algorithm with relatively good detection performance, and it is superior in speed and accuracy. Then, in response to the lack of adaptability of

the default box settings of the SSD network model to different data objects, this paper uses the k-means clustering method to cluster the calibration box of the training data set, getting more accurate and simplify default box settings. In this way, the regression positioning process of the network default box to the target is optimized, so that the generated default box is more accurate and the training is better converged. By using actual data for experimental comparison, the optimized SSD model is superior to the original SSD network in AP, mAP and FPS. The optimized SSD network mAP reaches 77.68%, which is 2.86% higher than the original network, at the same time, the detection speed is increased from 37 frames per second to 39 frames per second.

References

1. Li, S., Yang, J., Wang, Z.: Review of development and application of defect detection technology. Acta Automatica Sinica (2020). https://doi.org/10.16383/j.aas.c180538
2. Zhou, L.: Present situation and development of modern precision measurement technology. Chin. J. Sci. Inst. **38**(8), 1869–1878 (2017)
3. Tang, B., Kong, J., Wu, S.: Review of surface defect detection based on machine vision. J. Image Graph. **22**(12), 1640–1663 (2017)
4. Panning, A., Al-Hamadi, A.K., Niese, R., et al.: Facial expression recognition based on haar-like feature detection. Pattern Recogn. Image Anal. **18**(3), 447–452 (2008). https://doi.org/10.1134/S1054661808030139
5. Dalal, N.: Histograms of oriented gradients for human detection. In: IEEE Proceedings of CONFERENCE 2015, CVPR, vol. 9999, pp 1640–1663. IEEE, California (2005)
6. Lowr, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94
7. Yang, J., Li, S., Gao, Z., et al.: Real-time recognition method for 0.8 cm darning needles and bearings based on convolution neural networks and data increase. Appl. Sci. **8**(10), 1857 (2018). https://doi.org/10.3390/app8101857
8. Girshick, R., Donahue, J., Dare, T., et al.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE CONFERENCE 2014, CVPR, pp. 580–587. IEEE, Columbus (2014)
9. Girshick, R.: Fast R-CNN. In: IEEE CONFERENCE 2015, CVPR, pp. 1440–1448. IEEE, Boston (2015)
10. Ren, S., He, K., Girshick, R., et al.: Faster R-CNN: towards real- time object detection with region proposal networks. In: Cortes, C., Lawrence, N.D., Lee, D.D., et al. (eds.) CONFERENCE 2015, NIPS, vol. 28, pp. 91–99. NIPS, Montreal (2015)
11. Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: Unified, real-time object detection. In: IEEE CONFERENCE 2016, CVPR, pp. 779–788. IEEE, Seattle (2016)
12. Liu, W., Anguelov, D., Erhan, D., et al.: Single shot multi-box detector: towards real- time object detection with region proposal networks. In: Cortes, C., Lawrence, N.D. (eds.) CONFERENCE 2016, LNCS, vol. 9905, pp. 21–37. Springer, Amsterdam (2016)
13. Simonyan, K., Ziserman, A.: Very deep convolutional networks for large-scale image recognition. In: CONFERENCE 2015, ICLR, pp. 1440–1448. Hilton (2015)
14. Ding, C., He, X.: Cluster structure of *K*-means clustering via principal component analysis. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 414–418. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24775-3_50