An improved target detection algorithm based on EfficientNet

Tao Wu¹, Hongjin Zhu², Honghui Fan²,³ and Hongyan Zhou¹

¹ School of Mechanical Engineering, Jiangsu University of Technology, Changzhou 213001, China
² School of Computer Engineering, Jiangsu University of Technology, Changzhou 213001, China
³ E-mail: fanhonghui@jsut.edu.cn

Abstract. In order to improve the detection accuracy for small-scale targets in complex scenes, an improved target detection algorithm based on EfficientNet is proposed. Firstly, the EfficientNet network is used to optimize the DarkNet53 feature extraction network. Compressing the standard convolution with the depthwise separable convolution, and increasing the depth of the neural network with the residual network that can effectively achieve feature extraction, reduce the number of parameters and improve the detection speed. Secondly, a feature pyramid network is used to design four scales of features for multi-scale feature extraction, which improves the detection of small targets. The experimental results show that the YOLOv3 target detection algorithm improves the detection accuracy by 4.98% compared to the original algorithm on VOC dataset, which improves the detection accuracy and ensures real-time detection for small targets.

1. Introduction

Target detection has a wide range of application scenarios, which has been applied in many fields such as face recognition, intelligent transportation, medical and surveillance security. The traditional target detection and identification methods usually combine the features extracted by human design with the corresponding classification algorithm in the field of machine learning, such as Viola [1], who extracted the Haar features and combined them with the Adboost algorithm in machine learning to train a series of cascading classifiers to detect faces, and the final results obtained are also very optimistic. Along with the rapid development of deep learning, the convolutional neural network model quickly becomes the mainstream to replace the traditional feature extraction methods. At the same time, the deep learning-based series of target detection and identification methods have become an irreversible trend and gradually develop and grow, currently performing better target detection algorithms, such as: Single shot detection(SSD) [2-3], You only look once(YOLO) [4-6] and other regression-based algorithms and the Region-CNN(R-CNN) [7-8] series of region proposal algorithms.

The speed and accuracy of detection is critical in a number of important areas, especially in medicine for disease prevention and detection. The balance of the acquired high-dimensional data such as images and text can be further improved by performing a series of pre-processing techniques on the data set in order to reduce the time spent on model training [9]. Among them, the use of PCA algorithm helps to reduce the associated features in the high-dimensions dataset by transforming them to low dimensions, which has the advantage of reducing the overfitting and reducing the time required for the subsequent model training [10-11].
The depth of the convolutional network model and the number of parameters to be processed from the initial VGG16 to Xception [12] are doubled, and these factors have made a good contribution to the increasing accuracy of the model. What’s more, the hardware conditions required for detection are gradually relied upon, which has an impact on the real-time detection. The Faster R-CNN and R-FCN series have achieved the optimization of detection accuracy but the detection speed is relatively slow, and the YOLO series has achieved the improvement of detection speed at the cost of detection accuracy. The EfficientNet [13] network achieves a reasonable balance between accuracy and speed, and by compressing the standard convolution with deeply separable convolution and increasing the depth of the neural network with residual network, the detection effect is improved by using a smaller number of parameters. In addition, for the multi-scale transformation problem caused by the complexity of the scene, current studies including STDN and SNIPER [14] have focused on how to effectively design the multi-scale size to achieve the combination of detection accuracy and time and cost.

In this paper, based on the YOLOv3 [6] deep learning network, certain improvements are made to the network structure in order to improve the detection speed and accuracy, the number of detection layers of YOLOv3 is increased to four, and small target images are detected by feature fusion as much as possible. We compare the improved network with other detection algorithms, and the results show that the accuracy and speed are significantly improved.

2. Brief introduction to YOLOv3 and EfficientNet

The logistic regression problem as the main problem of target detection is the difference between the YOLO series networks and networks such as R-CNN. The image is input and segmented into a grid of N × N size, and then the centroid of the target is detected, the location of the cell is determined and the bounding box of the detected target is predicted as well as the confidence level [15].

2.1. Network structure of YOLOv3

YOLOv3 has achieved a major improvement over the previous YOLOv1 and YOLOv2. Compared with the backbone network Darknet-19 of YOLOv2, YOLOv3 summarizes the residual network ResNet [16] and creates a new network named Darknet-53 as the feature extraction network of v3. By taking advantage of the easy optimization and high depth of the residual network, the accuracy of the detection is further increased. The network structure of Darknet-53 [17] is shown in Figure 1.

Figure 1. DarkNet-53 network structure diagram.

The backbone of the DarkNet53 network consists of 52 convolutional layers, followed by a 1×1 convolutional layer for a total of 53 layers. The backbone network consists of one filter of 32 convolutional layers and five Residual Blocks, where the Residual Block structure consists of a convolutional layer with a step size of 2 and a set of convolutional layers that are repeatedly executed, so that the input image is finally downscaled 5 times from 416×416 to 13×13, and the output image size is 13×13, and output 3 feature maps for detecting different objects.
2.2. DarknetConv2D structure

The use of Conv2D structure in Darknet makes each convolution part of DarkNet53 is followed by L2 regularization and BatchNormalization and LeakyRelu activation function after the completion of the convolution, in which the ordinary Relu function will set the value of less than zero to zero, LeakyRelu sets the non-zero slope, which can be expressed by Formula (1) as follows:

\[
Y_i = \begin{cases} 
X_i & X_i \geq 0 \\
X_i / K_i & X_i < 0 
\end{cases}
\]

where \(X\) is the input value and \(K_i\) is the fixed parameter.

2.3. Introduction to EfficientNet algorithm

In this paper, the backbone network of YOLOv3 is replaced by EfficientNet, which compresses the standard convolution by depthwise separable convolution and increases the depth of the neural network by increasing the residual network, enabling deeper neural networks to extract features and reduce the number of parameters. YOLOv3’s original scale detection structure is augmented to meet the complexity and large-scale variation of the application.

EfficientNet [13] is a highly efficient network model proposed by Google in 19 years. By borrowing the residual network to increase the depth of the neural network, EfficientNet is able to extract features from a deeper neural network. In addition, to obtain more features, EfficientNet can change the number of feature layers for each layer to achieve more layers of feature extraction. Lastly, the resolution of the input image can be increased to allow the network to learn and express more information, which helps improve accuracy. The detection accuracy of EfficientNet B7 is 84.4%. While the Gpipe is only 84.3%, which has the highest detection accuracy at the present time. In terms of detection speed, Gpipe's detection speed is about 6 times slower than EfficientNet. In addition, EfficientNet improves the detection accuracy by 6% with the same flops for a wide range of residual networks. For further improvement of the detection accuracy, more attention may be paid to the scale-up of convolutional neural networks, such as the development of residual networks from ResNet18 to ResNet200. EfficientNet has a good balance between accuracy and efficiency and is implemented simply by scaling each dimension through the use of ratios, such as composite scaling, which optimizes both accuracy and speed [13]. The backbone of the EfficientNet model is borrowed from MBCConv in MobileNet V2, and to further optimize the network structure, EfficientNet has summarized the squeeze and excitation methods from SENet. The B0 network structure created after summarizing for MBCConv and SENet is shown in Table 1.

| Stage | Operator | Resolution | Channel | Layer |
|-------|----------|------------|---------|-------|
| 1     | Conv3×3  | 224×224    | 32      | 1     |
| 2     | MBConv1, k3×3 | 112×7112 | 16      | 1     |
| 3     | MBConv6, k3×3 | 112×112   | 24      | 2     |
| 4     | MBConv6, k5×5 | 56×56     | 40      | 2     |
| 5     | MBConv6, k3×3 | 28×28     | 80      | 3     |
| 6     | MBConv6, k5×5 | 14×14     | 112     | 3     |
| 7     | MBConv6, k5×5 | 14×14     | 192     | 4     |
| 8     | MBConv6, k3×3 | 7×7       | 320     | 1     |
| 9     | Conv, Pool, FC | 7×7       | 1280    | 1     |

The EfficientNet network structure consists of a Stem, 16 MBConvBlocks, Conv2D, GlobalAveragePooling2D, Dense, and other layers as shown in Figure 2, where the critical part is the 16 MBConvBlocks. The Block structure is borrowed from the residual network, in which the 1×1 convolution is not used to increase the dimension, followed by the N×N convolution, and then the
attention mechanism is used to add the corresponding weights to each channel to extract important information.

The activation function in EfficientNet is designed as a Swish activation function, and the swish function is a Self-Gated activation function, expressed by the formula:

\[
swhish(X) = X\sigma(\beta X)
\]

where \(\sigma\) is the logistic function, the parameters \(\beta\) can be learned or set to fixed hyper-parameters.

**2.3.1. Attention mechanism.** In MBConvBlock, the attention mechanism of channels is utilized, i.e., according to the different key information obtained by different channels, by adding a weight to each channel for the correlation between the channel and the key information, the larger the weight value, the higher the correlation [18]. The attention mechanism is divided into three parts: squeeze, excitation, and attention [19].

Squeeze function:

\[
z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j)
\]

The role of global average is to implement the operation of summing and averaging the values of each channel, consistent with the global average pooling expression. Where \(H\) and \(W\) are the width and height of each channel and \(u\) is the average value of each channel.

Excitation function:

\[
s = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W^1_s \delta(W^2_z))
\]

The \(\delta(X)\) A function is the Relu activation function, the \(\sigma(X)\) B is the sigmoid activation function, and the weight values \(W_1\) and \(W_2\) learned from the training, which ultimately yields a one-dimensional weight value for activating each channel.

Scaling function:

\[
\tilde{x}_c = F_{scale}(u_c, s_c) = s_c \cdot u_c
\]

The enhancement of attention to key channel domains is achieved by multiplying the different weight values \(S\) obtained by the excitation function by different channels \(u\), essentially similar to scaling.
3. Improvement of network structure based on YOLOv3

3.1. Improving YOLOv3 with more scales

3.1.1. Multi scale feature extraction for small object detection. The YOLOv3 backbone network performs feature extraction to obtain three-layer feature maps with sizes of 13×13, 26×26, and 52×52, respectively. In order to cope with the complexity of the application scenarios, including a large number of detections and large-scale variations, the improved network amplifies the original feature map into four layers, with the amplified sizes of 13×13, 26×26, 52×52, and 104×104. Multiscale detection is an effective way to solve the problem of multi-scale variation by fully preserving the deep image semantic information and at the same time making reasonable use of the ignored shallow image feature information as far as possible [20].

The development history of the convolutional neural network shows that most of the research studies deepen the depth of the detection network model to have a better understanding and use of deep semantic information, such as the SSD backbone network VGG16, which only uses Conv4_3 feature information and ignores the shallow features. Of course, in order to improve the accuracy of target detection, the deepening of the network structure used to extract deeper semantic information has a significant effect, but if the ignored shallow information is fused and utilized, the accuracy of detection can be improved to some extent.

3.1.2. Feature pyramid network. The Feature Pyramid Network (FPN) [21] network is a method for multi-scale feature extraction, which is more prominently used in Faster R-CNN. The basic idea is to fuse the acquired high-level semantic information and the underlying spatial information, as the feature semantic information is more prominent in the deeper layers and rarer in the shallower layers of the network, so more semantic information can be extracted by fusing the two. The specific implementation is to change the shallow feature channel by connecting the 1×1 convolution horizontally, and then superimpose the obtained result with the up-sampling part of the feature layer.

![Improved YOLOv3 network structure diagram.](image)

3.2. Using EfficientNet as the backbone network

The EfficientNet network borrows from residual networks for deep augmentation, involving depthwise separable networks for compression and channel attention mechanisms for critical channel attention enhancement. After the basic Steam module and MBConvBlocks network structure comes the
Conv2D layer, BatchNormalization, Swish activation and pooling layers. In order to maintain consistency with the original YOLOv3 backbone network DarkNet53, before combining EfficientNet with YOLOv3, one of the key layers of EfficientNet - Steam and MBConvBlocks - should be kept and the number of layers behind should be removed, so that the number of downsampling operations on the whole network is the same as before. The modified EfficientNet backbone network consists of 17 layers in which the Steam structure is a convolutional layer followed by Batchnormalization which is used to compress the input image for downsampling. The improved feature extraction network is shown in Figure 3.

After feature extraction through the backbone network EfficientNet, the effective feature layers obtained from MBConvBlock for downsampling are extracted. After the feature extraction process, the feature map sizes of 13×13, 26×26, 52×52, 104×104 are obtained, and the bottom feature layer is up-sampled from the bottom to the neighboring feature layer, and then fused with the neighboring features until all four detection scales are completed. The combination of the shallow network, which is rich in location information, and the deep network, which is rich in semantic information, improves the ability of YOLOv3 to cope with the complexity of the scene and improves the detection accuracy.

4. Experimental analysis and results
In order to verify the effectiveness of EfficientNet's improved YOLOv3 target detection algorithm, this paper uses Pytorch to implement it. The experimental environment is as follows: Intel Core i5-10400 processor, 16G memory, Nvidia GeForce RTX 2060 SUPER, CUDA version 10.1, Win10 Professional 64-bit OS.

4.1. Data set production
The datasets used for this experiment were selected from the large public datasets PASCAL VOC2007 and VOC2012 for training. The PASCAL VOC dataset contains 20 types of objects including vehicles, pedestrians, etc. and all of them are labeled, and the total number of images in the 07trainval dataset is 5011 plus the total number of images in the 12trainval dataset is 16551 images. The four scales designed in this paper have three anchor boxes for each scale, the height and width of which are clustered by k-means in the VOC dataset as (15×26), (22×67), (38×116), (35×38), (62×64), (69×161), (106×230), (109×98), (162×164), (198×321), (304×207), and (366×362). All the input images are set to 416×416 pixels.

The network parameters used are as follows: the training process uses adaptive moment estimation to optimize the loss function, the initial learning rate is 0.001, and the batch size is set to 16. The Tensorboard tool can be used to view the trend of train_loss after training is complete. Figure 4 shows that the loss function begins to show a sharp downward trend and then slowly changes and eventually converges.

![Figure 4. Loss function training graph.](image-url)
4.2. Experiment results analysis

4.2.1. Algorithm performance comparison. The MAP (mean average precision) used in this paper is a common evaluation index for target detection evaluation systems. The basic meaning of MAP is the average result obtained by taking the average AP value (average precision) of each detected target class, and the MAP range is \([0, 1]\). The relationship between MAP (mean average - precision) and AP can be expressed by the formula:

\[
MAP = \left( \sum_{k=1}^{C} AP_k \right) / C
\]

Where AP is the average precision and C is the number of class, and the average precision AP is the area enclosed by the PR value curve. The basic idea of PR curve (decision-recall) is to identify the change of threshold value, so that the system can identify the test picture set sequentially and the fluctuation of decision and recall values caused by the change of threshold value can be plotted into a PR curve.

4.2.2. Results analysis. The data sets used for training in the performance comparison experiment are VOC2007 and VOC2012, and the data sets used for data testing are selected as VOC2007 test images, and the improved MAP results of the YOLOv3 algorithm are shown in Figure 5. From the data, we can see that the network model still has room for improvement on small target detection due to the number of large targets on the target area adjustment dominates in the network training process, the scale of the prediction box is too large resulting in small target prediction box deviates from the real area, there is a target loss appears omission ratio.

Table 2 shows the test results of different detection algorithms, where the Faster R-CNN, SSD series data are from the Reference [22], YOLOv3 and EfficientNet-based improved YOLOv3 test results obtained by this experiment. As shown in Table 2, compared with the Faster R-CNN series based on the selective search in the target detection algorithm, the improved YOLOv3 detection accuracy value of the highest increase of 11.18%, compared with the SSD500 detection accuracy quality improved by about 4%, compared with the improved YOLOv3 based on DarkNet53 detection accuracy value of YOLOv3 improved 4.98%.

![Figure 5. Improved YOLOv3 algorithm MAP result graph.](image)

Table 2. Comparison of results of different target detection algorithm on the dataset.

| Methods   | Net     | Data            | MAP   |
|-----------|---------|-----------------|-------|
| Faster R-CNN | Resicual-101 | VOC007+12 | 76.4  |
| Faster R-CNN | VGG     | VOC007+12 | 73.2  |
| SSD500    | Residual-101 | VOC007+12 | 80.6  |
| YOLOv3    | DarkNet53 | VOC007+12 | 79.4  |
| YOLOv3    | EfficientNet | VOC007+12 | 84.38 |
4.3. Comparison of target detection results
Because of the disadvantages of small targets, such as relatively small number and small size, the final size displayed in the feature map is even smaller and fewer features are acquired after multiple sampling of the detection model. In addition, the size of the target mapped from the feature map back to the original map varies greatly from the original size. In this paper, by increasing the number of target detection layers from three to four and the number of anchor boxes to twelve, more position and semantic information can be obtained in the feature map. As shown in Figure 6, three images are selected for detection using the original YOLOv3 algorithm and the improved YOLOv3 algorithm. Compared to the missed detection phenomenon present in the algorithm, the improved algorithm effectively detects obscured portraits and pedestrians with small targets on the road, and the detection is further improved.

Figure 6. Comparison of small target detection result. (a) original image; (b) YOLOv3; (c) improved YOLOv3.

5. Conclusions
The improved YOLOv3 object detection algorithm of EfficientNet is proposed in this paper to reduce the number of model parameters and improve the detection accuracy for small objects. The experimental limitation is that it is not able to detect objects with heavy occlusion, so further research will focus on detecting objects with occlusion and further optimization of the network structure and parameters is needed.

References
[1] Viola P and Jones M 2001 Rapid object detection using a boosted cascade of simple features[C]. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001. Kauai, HI, USA: IEEE Comput. Soc. 11-511-1-518
[2] Liu W, Anguelov D and Erhan D 2016 SSD: Single Shot MultiBox Detector[G]. Leibe B, Matas J, Sebe N, et al. Computer Vision – ECCV 2016. Cham: Springer International Publishing 9905 21-37
[3] Zhang S, Wen L and Bian X 2018 Single-shot refinement neural network for object detection[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 4203-12
[4] Redmon J, Divvala S and Girshick R 2016 You only look once: Unified, real-time object detection[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 779-88
[5] Redmon J and Farhadi A 2017 YOLO9000: better, faster, stronger[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 7263-71
[6] Redmon J and Farhadi A 2018 YOLOv3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767
[7] Girshick R 2015 Fast R-CNN[C]. Proceedings of the IEEE International Conference on Computer Vision 1440-48
[8] Ren S, He K and Girshick R 2017 Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence 39(6) 1137-49
[9] Gadekallu T R, Rajput D S and Reddy M P K 2020 A novel PCA–whale optimization-based deep neural network model for classification of tomato plant diseases using GPU[J]. Journal of Real-Time Image Processing 1-14
[10] Reddy T, Bhattacharya S and Maddikunta P K R 2020 Antlion re-sampling based deep neural network model for classification of imbalanced multimodal stroke dataset[J]. Multimedia Tools and Applications 1-25
[11] Reddy G T, Reddy M P K and Lakshmanna K 2020 Analysis of dimensionality reduction techniques on big data[J]. IEEE Access 8 54776-54788
[12] Chollet F 2017 Xception: Deep learning with depthwise separable convolutions[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 1251-58
[13] Tan M and Le Q V 2019 Efficientnet: Rethinking model scaling for convolutional neural networks[J]. arXiv preprint arXiv:1905.11946
[14] Singh B, Najibi M and Davis L S 2018 Sniper: Efficient multi-scale training[C]. Advances in Neural Information Processing Systems 9310-20
[15] Xu Wei, Zhu Hongjin and Fan Honghui 2020 Research on improved YOLOv3 network in steel plate surface defect detection[J]. Computer Engineering and Applications 1-10
[16] He K, Zhang X and Ren S 2016 Deep residual learning for image recognition[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 770-8
[17] Wang H, Yu Y and Cai Y 2019 A comparative study of state-of-the-art deep learning algorithms for vehicle detection[J]. IEEE Intelligent Transportation Systems Magazine 11(2) 82-95
[18] Hu J, Shen L and Sun G 2018 Squeeze-and-excitation networks[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 7132-41
[19] Tan M, Chen B and Pang R 2018 MnasNet: Platform-Aware Neural Architecture Search for Mobile[J].
[20] Singh B and Davis L S 2017 An Analysis of Scale Invariance in Object Detection - SNIP[J].
[21] Lin T Y, Dollár P and Girshick R 2017 Feature pyramid networks for object detection[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2117-25
[22] Fu C Y, Liu W and Ranga A 2017 Dssd: Deconvolutional single shot detector[J]. arXiv preprint arXiv:1701.06659