An Improved Crowd Counting Method Based on YOLOv3

Shuang Zheng, Junfeng Wu, Fugang Liu(✉), Yunhao Liang, and Lingfei Zhao

Heilongjiang University of Science and Technology, Harbin 150022, China
liufugang_36@163.com

Abstract. This paper proposes a method of crowd counting. We use ResNeSt-50 as the backbone network of YOLOv3. After the backbone network, we add SPP (Spatial Pyramid Potential) and PANet (Path Aggregation Network) to enhance the receptive field of convolutional neural network and improve the accuracy of stream of people or crowd counting in real application scenarios. In the application scenario of high-density crowd counting, an improved VGG network is used to design a deep network to capture high-level semantic information. At the same time, a shallow network is constructed to detect the head blob of people far away from the camera. The deep network and the shallow network are combined to detect high-density crowd. Finally, through the effective fusion of the above two network models, the accuracy and applicability of the algorithm are further improved. It can improve the detection accuracy in the case of small number of people and occlusion, and effectively reduce the estimation error in the scene with high density crowd.

Keywords: Crowd density · Target detection · Convolutional neural network · YOLOv3

1 Introduction

Computer vision, including human detection, human posture recognition, crowd counting and other technologies, is one of the research hotspots in recent years, and has become an important branch of artificial intelligence industry. Among them, the crowd density estimation technology has a broad prospect in the field of security, new retail and other applications because of its far higher accuracy and speed than the naked eye count. According to different application scenarios, the technology is mainly divided into low-density crowd estimation and high-density crowd estimation. In the aspect of low density crowd estimation, the detection algorithms mainly include the target detection algorithm based on Region Proposal represented by RCNN (including RCNN, SPP-NET, FAST RCNN, FAST RCNN, etc.) and the target detection algorithm based on regression represented by YOLO (including YOLO series, SSD, etc.). In the aspect of high-density flow estimation, the method of Density Map is mainly used at present. Zhang et al. [1] used multi-column convolutional neural network to extract head features of different scales. The disadvantage of this multi network model is that it has many parameters

X. Jiang (Ed.): MLICOM 2021, LNICST 438, pp. 337–350, 2022.
https://doi.org/10.1007/978-3-031-04409-0_31

and large amount of calculation, so it can't carry out real-time crowd count detection. Vishwanath et al. [2] proposed four modules: Global Context Estimator (GCE), Local Context Estimator(LCE), Density Map Estimator(DME) and Fusion-CNN (F-CNN), which can generate high-quality crowd density and count estimation by explicitly incorporating global and local contextual information of crowd images. The authors in [3] used a dilated CNN as the back-end and uses the dilated kernel to provide larger reception fields and to replace pooling operations. It made CSRNet easy to train. CSRNet is used in four data sets (ShanghaiTech dataset, UCF_CC_50, WorldEXPO'10 and UCSD dataset),and high accuracy is obtained.

According to the research of current scholars, the current research in this field mainly tends to the high-density crowd estimation, but this kind of high-density crowd estimation network is difficult to accurately estimate the low-density crowd. To solve this problem, we propose an improved algorithm based on YOLOv3. In this paper, we modify the backbone network, add feature enhancement module, design deep network and shallow network to improve the detection accuracy and applicability of the algorithm. It can improve the detection accuracy in the case of small number of people and occlusion, and effectively reduce the estimation error in the scene with high density crowd.

2 YOLOv3

2.1 The Structure of YOLOv3

The schematic diagram of YOLOv3 structure module is shown in Fig. 1. Inside the red dotted line is the backbone network Darknet-53.The modules in the three dotted boxes below are the core structure of YOLOv3. DBL contains the corresponding convolution layer, batch normalization layer (BN)and leaky relu, which is the basic unit of YOLOv3. In order to reduce the over fitting problem caused by too many layers, YOLOv3 adds a new residual module ('n' in 'resn' is the number, which means n residual units). In Fig. 1, y1, y2 and y3 are characteristic maps of YOLOv3 with three different scales. Taking the experiment of coco data set [4] as an example, because there are 80 categories in this data set, each box outputs a probability for each category, and each grid cell detects 3 boxes, and each box needs five basic parameters, so the depth of the three characteristic graphs is $(3 \times (5 + 80))$.

2.2 Backbone Network Darknet-53

YOLOv3 uses Darknet-53 network for feature extraction. Table 1 shows the network structure of Darknet-53. Darknet-53 is to deepen the convolution layer on the original network structure of YOLOv2 [5] and use the residual structure module. The network uses continuous 3×3 and 1×1 convolution kernels. 1×1 convolution kernel is used to reduce the dimension and 3×3 convolution kernel is used to extract features. Multiple convolution kernels are used alternately to reduce the dimension and extract features.In the process of forward propagation, the step size of convolution kernel is changed to realize the tensor size transformation. For example, the step size of convolution kernel in the red box in Table 1 is 2, which is equivalent to the image side length is reduced to

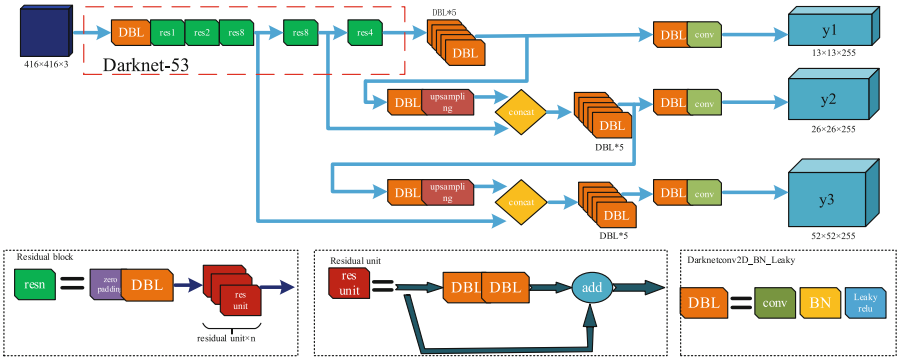

Fig. 1. Schematic diagram of the structure module of YOLOv3.

half of the original, and the area is reduced to one fourth of the original. In Table 1, a total of 5 times of reduction, the area became 1/32 of the original. When the size of the input feature map is 416 × 416, the output is 13 × 13.

Table 1. Darknet-53 network structure.

Type	Output channel	Convolution kernel	Output size	
Conv	32	3×3	256×256	
Conv	64	3×3/2	128×128	
Conv	32	1×1		×1
Conv	64	3×3		
Residual			128×128	
Conv	128	3×3/2	64×64	
Conv	64	1×1		×2
Conv	128	3×3		
Residual			64×64	
Conv	256	3×3/2	32×32	
Conv	128	1×1		×8
Conv	256	3×3		
Residual			32×32	
Conv	512	3×3/2	16×16	
Conv	256	1×1		×8
Conv	512	3×3		
Residual			16×16	
Conv	1024	3×3/2	8×8	
Conv	512	1×1		×4
Conv	1024	3×3		
Residual			8×8	
Average pooling	1024	Global pooling	1×1	
FC			1000	
Softmax			1000	

3 Improvement of YOLOv3+ResNeSt-50 Algorithm

3.1 Backbone Network ResNeSt-50

ResNeSt-50 [6] network extends the attention in channel direction to feature mapping group representation and uses unified CNN algorithm for modularization and acceleration. The network structure of its core module (Fig. 2) combines the ideas of multi-path mechanism, packet convolution, channel attention mechanism and feature mapping attention mechanism respectively. Among them, the multipath mechanism has a significant effect in GoogleNet [7], in which each network block is composed of different convolution cores. Packet convolution [8] appears in ResNet "bottle block" of ResNeXt [9], which realizes the transformation of multi-path structure into unified operation. Channel attention mechanism is introduced in SE-Net [10] to realize adaptive recalibration of channel feature response. The attention mechanism of feature mapping is proposed in SK-Net [11] and applied in two branches of network.

In order to compare with ResNet variant series [12], Hang Zhang et al. cut all network input images to 224×224 for training. In order to train better inference speed and reduce the amount of model, they moved the average pooling operation to the 3×3 convolution layer, and built ResNeSt fast model on the feature map sampled under convolution layer. Table 2 uses the mean average precision (mAP) of each category to quantify the network performance. It shows the experimental comparison results, which show that the accuracy of ResNeSt is the best in the ResNet variant series when only the backbone network is replaced, and the ResNeSt structure is better than the ResNet structure with the same network parameters.

The performance of split attention module in ResNeSt has been improved in classification, detection, instance segmentation and semantic segmentation. Based on the above characteristics, we modified Darknet-53, the backbone network of YOLOv3 in Fig. 1, to ResNeSt-50. The experiment in the fifth part of this paper shows that, compared with the original network, this scheme can improve the detection accuracy by about 3 percentage points.

3.2 Feature Enhancement

In order to strengthen the receptive field [13] of the network, SPP (spatial pyramid potential) [13] and PANet (Path Aggregation Network) [14] are added after the backbone network. Figure 3 is the structure diagram of ResNeSt-50 adding SPP and PANet. By adding SPP, multiple windows are used for any size feature map to get fixed size feature vector, so as to achieve the effect of enhancing network receptive field. The information flow is enhanced by PANet, and the path enhancement method from bottom to top is adopted to improve the accuracy of feature location in the lower level structure and shorten the information path of the lower and upper level features. In this paper, the adaptive feature pool is used to connect the feature grid and all feature layers, so that

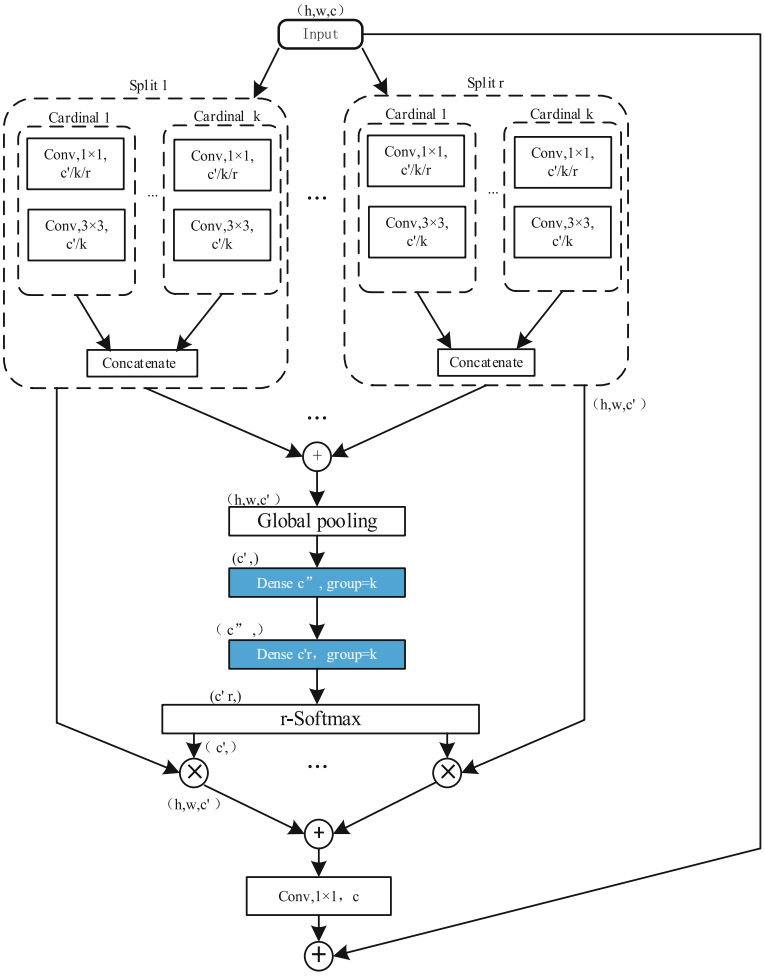

Fig. 2. Network structure of ResNeSt core module.

the useful information in each feature layer can be directly propagated to the following sub network. The network can extract feature information more quickly and improve the processing speed of the network. As far as CNN network is concerned, the addition of SPP and PANet has no effect on the network structure, which is equivalent to replacing the original pooling layer.

Table 2. mAP of ResNeSt vs. ResNet variant network.

	Method	Backbone	mAP%
Prior Work	Faster-RCNN	ResNet101	37.3
		ResNeXt101	40.1
		SE-ResNet101	41.9
	Faster-RCNN+DCN	ResNet101	42.1
	Cascade-RCNN	ResNet101	42.8
Experiment results	Faster-RCNN	ResNet50	39.25
		ResNet101	41.37
		ResNeSt50	42.33
		ResNeSt101	44.72
	Cascade-RCNN	ResNet50	42.52
		ResNet101	44.03
		ResNeSt50	45.41
		ResNeSt101	47.50
	Cascade-RCNN	ResNeSt200	49.03

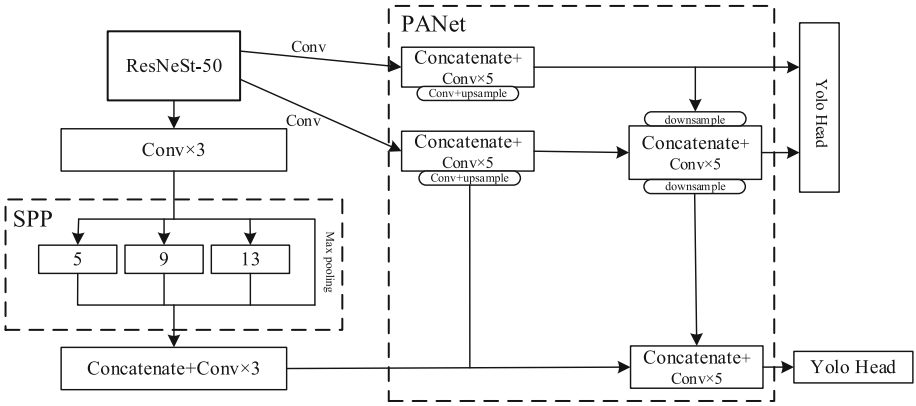

Fig. 3. Feature enhancement module SPP and PANet.

4 Density Map Network

In this paper, deep and shallow networks are designed to detect high density traffic. The density map is generated by combining the deep layer and the shallow layer network, and the Gauss density algorithm is used to detect the traffic of the generated density map. The overall convolution network structure is shown in Fig. 4.

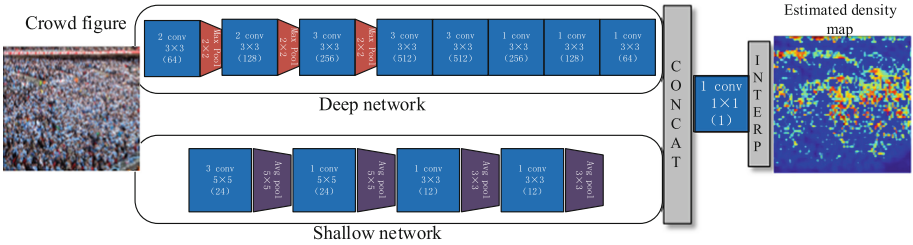

Fig. 4. Network structure of density graph.

4.1 Deep Network Structure

Deep network is a model based on VGG-16 [15]. Because crowd density estimation requires super-pixel detection, which is different from image classification. The latter is to assign a single discrete label to the whole image, so it is necessary to remove the full connection layer in VGG-16 network to obtain pixels for detection, so that the network is completely convoluted in structure. VGG-16 network has five maximum pooling layers. In this paper, the fourth maximum pooling layer and the fifth pooling layer are deleted, and the dilated convolution with dilated rate of 2 is added to the last six convolution layers. In dilated convolution, if the convolution core with $k \times k$ size is amplified to the size of $k + (k-1)(k+1)$ with the expansion step of r, the receiving field can be expanded without increasing the number of parameters and the amount of calculation (as shown in Fig. 5). At the same time, dilated convolution can aggregate multi-scale context information and keep the same resolution, so that sparse kernel can replace pooling layer and convolution layer.

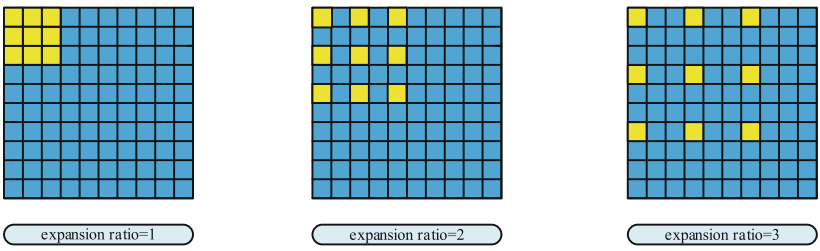

Fig. 5. 3×3 Convolution kernels with dilations of 1, 2 and 3

The formula of dilated convolution is as follows:

$$y(m,n) = \sum_{i=1}^{M} \sum_{j=1}^{N} x(m + r \times i, n + r \times j) w(i,j) \tag{1}$$

$y(m, n)$ is the output of the dilated convolution of the input $x(m, n)$ and the convolution kernel $w(i, j)$, whose length and width are M and N respectively. The parameter r is the dilated rate. If $r = 1$, the dilated convolution becomes normal convolution. It is found that when $r = 2$, the effect is the best. This enables the network to detect when the input is 1/8 times the original resolution, and increases the receptive field of the network.

4.2 Shallow Network Structure

In this paper, we construct a shallow convolutional network to identify people in the photos who are far away from the camera. The depth of the network has six layers. The number of channels in the first four floors is 24, and that in the fifth and sixth floors is 12. The convolution kernel size of the first four layers is 5×5, and that of the fifth and sixth layers is 3×3. The shallow network is mainly used to detect small head blob, and the average pooling is used in the shallow network to prevent the loss of high-dimensional information caused by the maximum pooling.

5 Experimental Analysis

5.1 Data Preprocessing

The data set of this paper uses the data of Baidu's crowd density. Some training data refer to public data sets (such as ShanghaiTech, UCF-CC-50, WorldExpo'10, Mall, etc.), and the data annotation of data sets is in the corresponding json file. Because there are ignored areas in the data, and the annotation is not uniform, so this paper unifies the annotation of the data, and then fills the ignored areas in the image. In this paper, the size of all images is normalized to keep the same size of all input images. For the dimension of image output, we can have three choices: Directly output a density map without compression, or directly output a compressed density map, or directly output an actual value of the number of people detected. Through the comparison of the three schemes, the second scheme highlights the image features more and reduces the amount of calculation. So this paper chooses the second way.

5.2 Training Methods

In this paper, we use ShanghaiTech data set to detect the high-density of crowd. Firstly, the data set is trained once by using this network to get a detection model. Then the data are classified according to different scenes of the dataset, and the data of the same scene are trained separately, and the trained scene model and detection model are fused. Because of the complexity of the network in this paper, the transfer learning method is used in the training, and the weight of ResNeSt-50 model is used for pretraining, so

that the training can get a fast convergence effect. We finds that the loss rate is very low in the process of training by directly using density map as the result output, but it is quite different from the actual number of people. Therefore, this paper changes the loss function into the result of adding the mean square error of density map and the number of people.

Through the network structure designed in this paper, using a single Tesla V100 GPU to train, the "person part" iterates 20000 times, the "head part" iterates 30000 times, and the images classified for different scenes iterate 5000 times. The average training time of "person part" is about 23 h, the average training time of "head part" is about 16 h, and the training time of different scene images is about 3 h. In this paper, momentum is used to optimize the training, and the momentum is 0.9. The initial learning rate is 0.001. After every 1000 iterations, the learning rate is reduced to 0.1 times of that before, and the batch_size is 8. The single Tesla V100 GPU is used to train the density map network for 400 rounds. Using Adam to optimize training, the learning rate is 0.001, and the batch_size is 26.

5.3 Experimental Results

There are 11 different scenarios in Baidu's data set of crowd density. There is only one person with the least people flow, and there are hundreds of people with more people flow. There are thousands of people with more people flow in Shanghai's data set. To ensure the accuracy of the test results, this paper selects 25 images from the test set of Baidu traffic density data firstly. Among them, two images are randomly selected from each of the 11 scenes, and the remaining three images are randomly selected. Finally, 25 images are randomly selected from the test set of Shanghai data. The accuracy of the model is verified by 50 selected images. Figure 6 shows the comparison between the detection count and the actual count of each image in the selected 50 data sets. It is observed from Fig. 6 that the number of most people detection is close to the actual number. When the number of people in the image exceeds 2000, the error between the number of model detection and the actual number is large. The reason for this error may be that the density of crowd is too large, the number of heads in the picture is too dense, and the small objects are mistakenly considered as human in the detection and recognition of the model, thus causing interference to the experimental results.

Table 3 shows the comparison between the proposed algorithm and the original algorithm, as well as the comparison of the improved model of the original algorithm.

It can be concluded from Table 3 that the detection accuracy of the original YOLO3 model is 79.32. After the replacement of the network backbone, the accuracy of the model is improved by about 3 percentage points. On this basis, the accuracy of the final model is 86.76, which is 7.5 percentage points higher than that of the original model.

5.4 Result Analysis

1. Analysis of SPP and PANet.
Adding SPP module and PANet to ResNeSt-50 structure, the results of contrast enhancement are shown in Table 4. It is found that the accuracy of the improved YOLOv3 algorithm is improved, but the effect is not ideal.

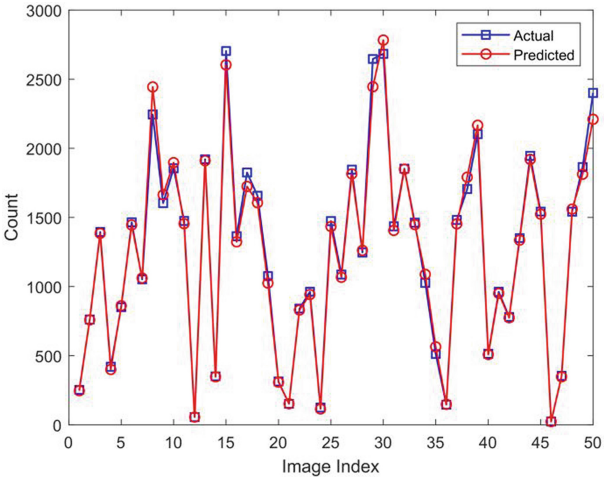

Fig. 6. Comparison between actual count and detection count.

Table 3. Comparison of recognition performance with different networks.

Method	mAP	FPS
YOLOv3	79.32	43.468
YOLOv3+ResNeSt-50+SPP+PANet (head)	82.15	54.605
YOLOv3+ResNeSt-50+SPP+PANet (person)	82.52	46.572
Fusion density map (head)	83.22	54.785
Fusion density map (person)	84.86	46.683
In this paper	86.76	53.861

The reason is that the original algorithm of YOLOv3 contains the fusion of high and low-level feature information, while PANet only improves the feature information fusion. In this paper, the data analysis is carried out in the data processing, and the compression algorithm is adopted for the image, and SPP module only ensures that the input image does not distortion when zooming. So the overall effect of SPP module and PANet only plays a supplementary role.

2. Analysis of density map fusion.
The performance comparison after enhancement is shown in Table 5. In this Table, "Fusion density map" means the density map fusion is added to the improved YOLOv3 algorithm. As can be seen from the figure, after fusing the density map, the accuracy of detecting some high-density pedestrian flow images will be improved by about 2 points, and the recognition speed will also be improved a little.

Table 4. Comparison of recognition performance after adding SPP and PANet.

Method	mAP	FPS
YOLOv3	79.32	43.468
YOLOv3+ResNeSt-50 (head)	81.56	54.401
YOLOv3+ResNeSt-50 (person)	81.79	45.214
YOLOv3+ResNeSt-50+SPP+PANet (head)	82.15	54.605
YOLOv3+ResNeSt-50+SPP+PANet (person)	82.52	46.572

Table 5. Comparison of recognition performance after merging density map.

Method	mAP	FPS
YOLOv3	79.32	43.468
YOLOv3+ResNeSt-50+SPP+PANet (head)	82.15	54.605
YOLOv3+ResNeSt-50+SPP+PANet (person)	82.52	46.572
Fusion density map (head)	83.22	54.785
Fusion density map (person)	84.86	46.683

3. Analysis of dilated convolution.
Compared with the effect of adding dilated convolution in the density map, it is found that the accuracy of this method is about 1% higher than that of the previous simple density map fusion method (Table 6). (as shown in Table 6). This shows that dilated convolution is better than simple pooling.

Table 6. Comparison of recognition performance with adding dilated convolution in density map.

Method	mAP	FPS
YOLOv3	79.32	43.468
Fusion density map (head)	83.22	54.785
Fusion density map (person)	84.86	46.683
Fusion density map (head)+Dilated convolution	84.65	53.954
Fusion density map (person)+Dilated convolution	85.64	46.124

4. Analysis of data enhancement.
In this paper, the Autoaugment technology is used to automatically select the optimal data enhancement operation. The data is improved by automatic search, and the optimal transformation strategy is found from the data itself, so that the neural network can produce the highest verification accuracy on the target data set.

The specific operation process is as follows: first, prepare 16 basic data enhancement operations, such as cropping, deformation, scaling, erasing, filling and a series of simple operations; Then five operations are randomly selected from them, and each operation is called a sub-policy; Finally, five operations are used to train each batch of images. After a certain epoch, the network begins to learn the effective transformation strategy. Through the feedback of the generalization ability of the training model in the verification set, the five sub-policies are concatenated, and then the final training is carried out to obtain the optimal enhancement algorithm. After data enhancement with Autoaugment, the comparison of accuracy and real-time performance of each algorithm is shown in Table 7.

Table 7. Comparison of recognition performance after data enhancement.

Method	mAP	FPS
YOLOv3	79.32	43.468
YOLOv3+ResNeSt-50+SPP+PANet (head)	82.15	54.605
YOLOv3+ResNeSt-50+SPP+PANet (person)	82.52	46.572
Fusion density map (head)	83.22	54.785
Fusion density map (person)	84.86	46.683
YOLOv3+Data enhancement	81.86	41.961
YOLOv3+ResNeSt-50+SPP+PANet (head)+Data enhancement	83.95	52.564
YOLOv3+ResNeSt-50+SPP+PANet (person)+Data enhancement	84.05	45.512
Fusion density map (head)+Data enhancement	85.55	52.684
Fusion density map (person)+Data enhancement	85.94	44.531

6 Conclusion

Aiming at the problem of detection density of crowd, this paper designs two algorithm models. In order to improve the YOLOv3 algorithm, ResNeSt-50 is used as the backbone network of YOLOv3, and the feature enhancement module SPP and PANet are added after the backbone network to enhance the receptive field of convolutional neural network and improve the detection accuracy of human flow density in real application scenarios; The other is density map network, which uses the improved VGG network to design a deep network for capturing high-level semantic information, and constructs a shallow network for detecting distant head spots in pictures, which combines the deep network with the shallow network to detect high-density people flow. Through the combination of the two models, we can improve the accuracy of the flow density detection. In this paper, the error rate of verification is only 0.1006 in the test set of Baidu, which achieves a relatively high accuracy, proving the feasibility and accuracy of the algorithm.

The main problems of further research are as follows: (1) There is no in-depth research in the complex situation of high-density pedestrian flow occlusion, the detection

of this situation is insufficient, and the number of heads is difficult to estimate in high-density. In the future, we need to further study the situation of severe occlusion and too dense number of heads. (2) The way of image processing will greatly affect the detection of network traffic. How to strengthen the characteristics of people in the image and improve the learning efficiency of the algorithm is worth studying. (3) The model of this paper is large, for the problem of hardware device embedding, it increases the cost, and the model will be lightweight later, so that the model of this paper can be better applied in practice.

Acknowledgements. This work has been partially supported by "Heilongjiang Science Foundation Project (LH2021F052)" .

References

1. Zhang, Y., Zhou, D., Chen, S., et al.: Single-image crowd counting via multi-column convolutional neural network. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3290–3298. IEEE, Las Vegas, NV, USA (2016)
2. Sindagi, V.A., Patel, V.: Generating high-quality crowd density maps using contextual pyramid CNNs. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 1879–1888. IEEE, Venice, Italy (2017)
3. Li, Y., Zhang, X., Chen, D.: CSRNet: dilated convolutional neural networks for understanding the highly congested scenes. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1091–1100, IEEE, Salt Lake City, UT (2018)
4. Redmon, J., Farhadi, A.: YOLOv3: An Incremental Improvement. arXiv:1804.02767 (2018)
5. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6517–6525. IEEE, Honolulu HI (2017)
6. Zhang, H., Wu, C., Zhang, Z.: ResNeSt: Split-Attention Networks. arXiv:2004.08955 (2020)
7. Szegedy, C., Liu, W., Jia, Y.Q., et al.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–9. IEEE, Boston, USA (2015)
8. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Adv. Neural. Inf. Process. Syst. **25**, 1097–1105 (2012)
9. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: 2017 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1492–1500. IEEE, Honolulu, Hawaii (2017)
10. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: 2018 In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141. IEEE, Salt Lake City, UT, USA (2018)
11. Li, X., Wang, W., Hu, X., Yang, J.: Selective kernel networks. In: 2019 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–519. IEEE, Long Beach, CA, USA (2019)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778. IEEE, Las Vegas, NV (2016)
13. He, K., Zhang, X., Ren, S., et al.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1904–1916 (2014)

14. Liu, S., Qi, L., Qin, H., et al.: Path aggregation network for instance segmentation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8759–8768. IEEE, Salt Lake City, UT, USA (2018)
15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: The 3rd International Conference on Learning Representations, pp. 7749–8758. IEEE, Banff, Canada (2014)