Article

DSRNet: A Novel Feature Extraction Network achieving trade off between accuracy and speed

Laigan Luo [1], Hailan Kuang[1], Xinhua Liu [1]* and Xiaolin Ma [1]

[1] Key Laboratory of Fiber Optical Sensing Technology and Information Processing, Ministry of Education, and Hubei Key Laboratory of Broadband Wireless Communication and Sensor Networks, School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China;

* Correspondence: liuxinhua@whut.edu.cn

Abstract: It is important to reduce the computation complexity while maintaining the accuracy of convolution neural networks. We deem it is possible to further reduce the network complexity while ensuring the accuracy. In this paper, we propose a novel feature extraction network called DSRNet which is lightweight but effective. DSRNet follows the basic ideas of stacking modules and short connection, introduces Depthwise Separable convolution and utilizes the Dilated convolution. The proposed network has fewer parameters and achieves outstanding speed. We conducted comprehensive experiments on CIFAR10, CIFAR100 and STL10 datasets, and the results showed the DSRNet has great performance improvement in terms of accuracy and speed.

Keywords: Depthwise; Dilated; Neural Network; Network Complexity

1. Introduction

The general practice in image recognition is roughly divided into two categories. The first one is about traditional mathematical statistics and machine learning theory. Initially, Image analysis was treated as two parts, low-level feature extraction with image process and mathematic modeling. For example, the feature vectors are extracted from target images by Sobel operator or Candy operator, then count the frequency of some certain vectors. Finally, researchers build mathematic model to predict and classify some new images. With the popularity of Bayesian theory [1]. and the extensive application of SVM algorithm, such supervised algorithms were becoming increasingly popular in medical image analysis. Machine learning methods greatly improve the accuracy of intelligent medical systems. However, they both highly rely on the handcraft feature extraction and man-made rules.

Deep learning is a great choice to handle the problem of image recognition. It brought a revolution for computer vision. It gradually receives the attention of the researchers thanks to the GPU-accelerated systems and large annotated datasets. Deep learning algorithms, especially the convolution neural networks can extract more relevant features than previous methods. LeNet-5 [2] was proposed in 1998 to solve the handwritten digits identify problem. It is regarded as the first generation of Convolution Neural Networks (CNN). The weight-sharing network structure makes CNN more similar to biological neural networks, reducing the complexity of the network model and reducing the number of weights. In convolution layers, local connection makes it possible that it can take a batch of original images as input. This network structure has some invariance to translation, scaling, and other forms of deformation.

Similar with LeNet , AlexNet [3] has deeper architecture. It utilizes the dropout to avoid the problem of overfitting and replace the sigmoid activation of ReLU (Rectified Linear Unit) activation, which can alleviate the problem of gradient vanishing in training procedure of deep networks. K.Simonyan et proposed VGG-16 and VGG-19 [4] in 2014. They are both simply stacked by convolution layers and pooling layers. However, VGG-Nets have more than ten hundred parameters especially in fully connected layers. L.Ming et al believe that average pooling layers can replace the first fully connected layer, then the amount of parameters can be reduced by an order of magnitude. Unlike VGG to increase network depth, Inception [5]; [6]; [7]; [8] is to expand the width of the network. Inception structure improves network performance by combining kernels with different receptive fields, while using 1x1 convolution kernel to avoid excessive network parameter.

With the increasing use of convolution networks in computer vision, many network structures have been developed based on the original networks, and most of the works are to increase the depth of the network or build a more complex network to get better performance. The deeper network, however, the harder it is to train. And deep networks suffer the gradient vanishing. Until in 2015, H.Kaimin et al proposed the residual structure named Residual Networks (ResNet) [9] which is the sum of the identify mapping and residual mapping to avoid this problem. DenseNet [10] is the kind of ultra version of ResNet and can extract the features in image more comprehensively. In 2018, DetNet [11] was proposed to optimize the feature extraction network, it is a feature extraction network dedicated to the target detection algorithm.

At the same time, we consider building a more lightweight network that guarantees accuracy and reduces the amount of parameters. In this paper, we propose a lightweight feature extraction network model. We use short connection to solve gradient problems, and redesign the residual blocks with Depthwise Separable convolution to reduce the amount of the parameters. Meanwhile, Dilated convolution is utilised to extract more information.

The main contributions can be summarized as follows:

- Most of feature extraction networks still have many parameters. We take into account that this situation can be improved. We utilise Depthwise Separable convolution instead of the common convolution. The Depthwise Separable convolution factorizes common convolution operation into two steps, reducing spatial dimensions. In this way, we reduce the parameters of the network and increase the speed.
- In deep networks, the networks will loss some important information, which will causes many problems, especially for small targets. The Dilated convolution can improve the receptive field and extract more features, it is introduced in first convolution layer and residual modules of network in order to alleviate the problem of information loss caused by deep networks.
- We conducted experiments to validate our network on CIFAR10, CIFAR100 and STL10, we compared speed and accuracy. The results showed that our network does not increase the complexity and computational complexity of the network, and the speed of the network has been greatly improved. And most importantly, we also ensure the accuracy.

2. Related Work

More and more deep convolution networks stacked with different scale convolution have emerged since AlexNet achieved excellent performance in image tasks. However, as the networks deepen, deep convolution networks expose the gradient degradation. ResNet addresses the problem, similar to the Highway Networks [12] idea, ResNet uses the shortcut connection, residual blocks add a linear layer connected from the input to the output, it sums the identify mapping and residual mapping and ensures the lost image information in the deep network is supplemented, the integrity of the image information is guaranteed. In this paper, we keep the idea of residual structure to address the problems deep networks suffer from.

Fisher Yu proposed Dilated convolution [13] in 2016. The Dilated convolution can increase the receptive field, allowing the output contain a larger range of image information, and the convolution is capable of learning more image information. At present, it has a good application in image segmentation, speech synthesis, machine translation, and image classification [14], and has achieved impressive improvement. In this paper, we will add Dilated convolution in an appropriate way, so that the feature maps of different scales will contain more image information. In a sense, the problem of information loss caused by deep networks can be alleviated. Our experiments conducted on datasets also show promising result.

In order to utilise the convolution neural network more effectively, the researchers have improved from the network structure, convolution layer parameters, loss function and many other aspects, and have achieved excellent results. Most of network structures, however, are still too complicated, have too many parameters, and cost too much calculation time. The researchers pursue more efficient improvements: reduce the amount of parameters and the cost of calculation while ensuring accuracy or even improving accuracy. The proposed Depthwise Separable convolution [15] addresses this problem by ensuring accuracy while reducing the amount of parameters and the cost of calculation. In 2017, Mobilenet [16] was proposed. The core of this model is Depthwise Separable convolution. Different network models verify that Depthwise Separable convolution has a good effect. In this paper, we design the network structure based on the Depthwise Separable convolution to reduce the complexity of network.

3. The DSRNet Architecture

In this paper, based on a design criterion: to reduce network computation and parameters while maintaining network accuracy, we design a more efficient and lightweight feature extraction network. In this section, we introduce the detailed structure of the proposed network.

The common ResNet101/152 or deeper network structures, for practical purposes, blocks will use 1x1, 3x3, 1x1 convolution for less computation and less parameter quantities. On this basis, we consider further reduce the complexity of calculation and the amount of parameters. So we redesigned the residual blocks.

We retain the residual blocks module and the shortcut connection, and redesign the layers of the residual blocks using the Depthwise Separable convolution. We reserve the first layer of 1x1 convolution and use it only to make dimensional changes to extract features, and redesign the 3x3 convolution based on Depthwise Separable convolution and remove the last convolution(1x1 convolution) of the residual blocks. Since the Depthwise Separable convolution factorizes one-step convolution operation into two steps, we can argue that the redesigned residual blocks contain only two convolution, while the redesigned second convolution has less parameters and calculation than the original common convolution. This guarantees two points: First, the introduction of shortcut connection ensures that the network still has the ability to solve the gradient problem caused by deep networks; Second, the introduction of Depthwise Separable convolution and the abandonment of third convolution guarantee that the network greatly reduce the amount of calculation and the amount of parameters without affecting the accuracy. The detailed of the residual block structure is shown in Fig. 1.

The reason why deep networks have good effects is that they can learn image features at different levels, and the deeper the features extracted by the network, the more abstract and more semantic information. In deep networks, however, image information loss will occur, such as loss of internal data structure, loss of spatial level information, and especially the loss of small target information! The design of the Dilated convolution addresses these problems. Compared to common convolution, the Dilated convolution has a special hyper-parameter called the dilation rate (the number of intervals in the kernel). Dilated convolution increases the receptive field while keeping the convolution kernel parameters constant, so each convolution output contains more image information, which solves the problem of missing image information caused by deep networks. Unexpectedly, using Dilated convolution multiple times in the network can also cause image information discontinuity. In this paper, in order to allow the network to transmit more image information, we will add the Dilated convolution in the appropriate place of the network structure and set appropriate dilated rate.

In order to preserve more initial semantic information of images, we set dilated rate=3 in the first convolution layer of the network structure. In order to compensate for the information loss caused by down-sampling, set the dilated rate=2 when feature maps down-sample. In addition, in the residual

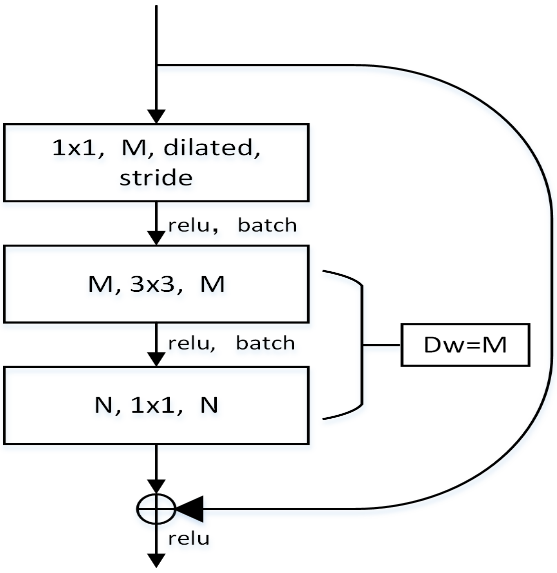

Figure 1. In the proposed blocks, the first layer of 1x1 convolution dynamically adds the dilated factor, and Depthwise Separable convolution replaces the second layer 3x3 convolution and the third layer 1x1 convolution layer. In addition, BatchNorm and Relu will follow the convolution. Here, Dw means Depthwise Separable convolution.

blocks we dynamically add the dilated rate, so that the feature maps of different levels can fuse more image features, and the dynamic dilated rate prevents the problem of information discontinuity.

We design a more efficient lightweight network. In order to further reduce network complexity, we redesign the residual blocks using Depthwise Separable convolutions. In order to improve the receptive field and extract more features, Dilated convolution is added. We have noticed that this approach not only does not increase the complexity and computational complexity of the network, but also improves the accuracy. We also adapt the modular design that uses residual blocks for stacking. The advantage of using modularity is that you only need to focus on a small number of parameter factors when adjusting the network. Detailed architecture can be seen in Table 1.

Table 1. DSRNet architecture. Dilated factor is used in the proposed architecture. Here, Dw means Depthwise separable convolutions.

layer name	outsize	DSRNet
conv1	112x112	7x7, 64, stride, dilated.
conv2	56x56	3x3 max pool, stride
		$\begin{bmatrix} \text{1x1, 64,stride,dilated} \\ \begin{bmatrix} \text{3x3, 64} \\ \text{1x1, 256} \end{bmatrix} \text{Dw = 64} \end{bmatrix}$ x3
conv3	28x28	$\begin{bmatrix} \text{1x1, 128,stride,dilated} \\ \begin{bmatrix} \text{3x3, 128} \\ \text{1x1, 512} \end{bmatrix} \text{Dw=128} \end{bmatrix}$ x4
conv4	14x14	$\begin{bmatrix} \text{1x1, 256,stride,dilated} \\ \begin{bmatrix} \text{3x3, 256} \\ \text{1x1, 1024} \end{bmatrix} \text{Dw=256} \end{bmatrix}$ x6
conv5	7x7	$\begin{bmatrix} \text{1x1, 64,stride,dilated} \\ \begin{bmatrix} \text{3x3, 64} \\ \text{1x1, 2048} \end{bmatrix} \text{Dw=512} \end{bmatrix}$ x3
	1x1	Global average pool 1000-d fc, softmax.

4. Experiment And Results

4.1. Datasets

We conducted our experiments on CIFAR10, CIFAR100 [17] and STL10 [18].

The CIFAR10 dataset consists of 60,000 (32*32) color images of 10 classes, each with 6000 images. There are 50,000 training images and 10,000 testing images. The dataset is divided into five training batches and one testing batch, each with 10,000 images. The testing batch contains exactly 1000 randomly selected images from each category. Training batches contain the remaining images in random order, but some training batches may contain more images from one category than the others. Overall, the sum of the five training sets contains exactly 5,000 images from each class.

The CIFAR100 dataset has 100 classes. Each contains 600 images. Each class has 500 training images and 100 test images. The 100 classes in CIFAR100 are divided into 20 super-class. Each image has a 'fine' label (the class it belongs to) and a 'rough' label (the super-class it belongs to).

The STL10 is an image recognition dataset for developing unsupervised feature learning, deep learning, self-taught learning algorithms containing images of 10 types of objects, 1300 images per class, 500 images for training, and 800 images for testing, each with a resolution of 96*96. In addition to images with category labels, there are 100,000 images without category information.

4.2. Training Detail

Our network structure can be seen in Table 1. The first stage starts with a 7x7 convolution, sets the dilated rate factor to 3 and the stride factor to 2, then followed by four stages which is four blocks, where the first block we start with a 3x3 maxpool, and set the stride factor of 2. Finally, it ends with the average pool, full connection, Softmax. In addition, the dilated factor is set to 2 when the feature maps downsample.

In order to compared with other network structures, we set some hyper-parameters in advance, and the same processing is performed on the image data. We have randomly cropped, flipped, and normalized images. We set the initial learning rate LR = 0.1, and then reduce it by a factor of 1/10. The optimization method is mini-batch momentum-SGD, and the momentum is 0.9, the weight-decay is $5e^{-4}$, and L2 regularization is used.

In terms of the accuracies, all of the networks were not pre-trained due to some special reasons, as a result, the accuracies were not ideal!

4.3. CIFAR10

We compared three networks: ResNet, ResNext, DSRNet, DSRNet shows a more performance improvement. Results can be seen in Table 2.

Table 2. the detailed results of classification performance of DSRNet, ResNet50, ResNet100, ResNext50 and ResNext100 on CIFAR10 separately(BatchSize=128).

	DSRNet	ResNet50	ResNet101	ResNext50	ResNext101
Top1-error	9.20%	11.85%	11.88%	10.44%	10.91%.
fps/ms	0.148	0.719	1.406	0.406	0.977

From the Table 2, DSRNet has the best performance and the fastest speed which are 9.20% and 0.019s, respectively. The most important point is the speed. Our speed is 5 times faster than ResNet50, and is 3 times faster than ResNext50 approximately, and compared to the ResNet101, the speed almost be 10 times. Compared to ResNet and ResNext, the number of parameters has also been reduced. The results have shown that DSRNet has the ability to improve network accuracy while reducing the computational complexity and the parameter amount of the network as much as possible.

4.4. CIFAR100

We still compared three networks: ResNet, ResNext, DSRNet, and we added an experimental indicator: Top3-error which we got a much larger performance improvement than the others. Meanwhile, we conduct more experiments on CIFAR100, details can be seen in Table 3.

Table 3. the detailed results of classification performance of DSRNet, ResNet50, ResNet100, ResNext50 and ResNext100 on CIFAR100 separately(BatchSize=128).

	DSRNet	ResNet50	ResNet101	ResNext50	ResNext101
Top1-error	33.50%	38.48%	39.43%	36.53%	35.20%.
Top3-error	17.80%	22.02%	22.47%	20.09%	19.06%.
fps/ms	0.180	0.719	1.445	0.430	0.977

Based on the experiments conducted on CIFAR100, DSRNet has the best performance. From the Table 3, we can see that DSRNet has the best performance than others, and DSRNet shows 1.7% and 1.26% improvement than ResNext101 and ResNet101 on cifar100 respectively. Meanwhile, DSRNet has the fastest speed, we set BatchSize=128, it only costs 0.023s, which is better than other networks speed.

4.5. STL10

We keep the same hyper-parameters on STL10 dataset, the same processing is performed on this image data. However, different from the CIFAR dataset, the image size of STL10 is 96*96. We perform zero-padding on the image during processing, and then randomly crop the image to 96*96. The results are showed as below. Details can be seen in Table 4.

Table 4. the detailed results of classification performance of DSRNet, ResNet50, ResNet100, ResNext50 and ResNext100 on STL10 separately. BatchSize=128.

	DSRNet	ResNet50	ResNet101	ResNext50	ResNext101
Top1-error	25.26%	27.28%	28.11%	27.96%	26.90%.
Top3-error	7.25%	8.11%	8.59%	8.65%	8.37%.
fps/ms	0.430	0.633	1.953	1.031	2.969

From the Table 4, DSRNet got better performance. DSRNet achieved 25.26% and 7.25% separately, which is 1% higher than ResNet and ResNext approximately. At the same time, the BatchSize is 128, compared to ResNet and ResNext, we got the fastest speed which is 0.055s, which still have greater performance than ResNet and ResNext.

4.6. The Parameters

We use Depthwise Separable convolution to design residual blocks instead of common convolution, meanwhile, we do not use the last 1x1 convolution in the block, which turns out that it does not have adverse effect from the results. From the Fig. 2, we can see our architecture has fewer parameters compared with other networks, which is the reason why our network has higher speed.

5. Conclusion

In this paper, we proposed a lightweight network. We designed the residual modules with Depthwise Separable convolutions, and the Dilated convolution is introduced. From the results, DSRNet achieved best speed which is 0.023s on CIFAR10 and CIFAR100. On STL10, it still had the fastest speed(0.055s). More importantly, our architecture had fewer parameters which was $2.19 * 10^7$ while improving the accuracy. By the experiments, we can see that our network does guarantee accuracy while improving speed. Compared with ResNet and ResNext, our network had less parameters, and we got the fastest speed. At the same time, our network presented gains in

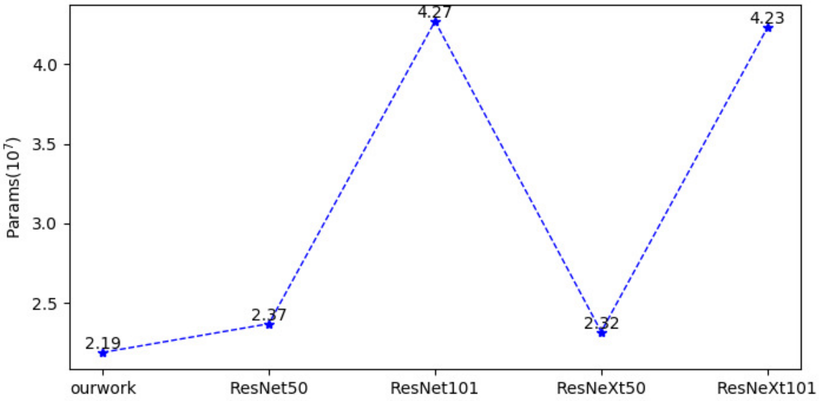

Figure 2. the parameters of networks (we can see our network has fewer parameters which is 2.19 * 10^7).

classification performance on the CIFAR10, CIFAR100 and STL10 datasets. The results have shown that DSRNet has the ability to improve network accuracy while reducing the computational complexity and the amount of the network parameters as much as possible.

References

1. Bernardo, J.; Smith, A. *Bayesian Theory*; 2008.
2. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE* **1998**, *86*, 2278 – 2324. doi:10.1109/5.726791.
3. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. International Conference on Neural Information Processing Systems, 2012.
4. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Computer Science* **2014**.
5. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift **2015**.
6. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. Computer Vision Pattern Recognition, 2016.
7. Szegedy, C.; Ioffe, S.; Vanhoucke, V. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning **2016**.
8. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions **2014**.
9. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. 2016, pp. 770–778. doi:10.1109/CVPR.2016.90.
10. Gao, H.; Zhuang, L.; Weinberger, K.Q. Densely Connected Convolutional Networks **2016**.
11. Li, Z.; Chao, P.; Gang, Y.; Zhang, X.; Jian, S. DetNet: A Backbone network for Object Detection **2018**.
12. Kumar Srivastava, R.; Greff, K.; Schmidhuber, J. Training Very Deep Networks. *2015 Neural Information Processing Systems (NIPS 2015 Spotlight)* **2015**.
13. Yu, F.; Koltun, V. Multi-Scale Context Aggregation by Dilated Convolutions **2016**.
14. Liu, S.; Di, H.; Wang, Y. Receptive Field Block Net for Accurate and Fast Object Detection **2017**.
15. Sifre, L.; Mallat, S. Rigid-Motion Scattering for Texture Classification. *Computer Science* **2014**, *3559*, 501–515.
16. G. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications **2017**.
17. Krizhevsky, A.; Hinton, G. Learning multiple layers of features from tiny images. *Computer Science Department, University of Toronto, Tech. Rep* **2009**, *1*.

18. Coates, A.; Y. Ng, A.; Lee, H. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. *Journal of Machine Learning Research - Proceedings Track* **2011**, *15*, 215–223.