International Journal of Automation and Computing
www.ijac.net

18(6), December 2021, 963-972
DOI: 10.1007/s11633-021-1287-y

DLA+: A Light Aggregation Network for Object Classification and Detection

Fu-Tian Wang[1] Li Yang[1] Jin Tang[1] Si-Bao Chen[1] Xin Wang[2,3]

[1]School of Computer Science and Technology, Anhui University, Hefei 230601, China

[2]Shenzhen Raixun Information Technology Co., Ltd., Shenzhen 518000, China

[3]Peking University Shenzhen Institute, Shenzhen 518000, China

Abstract: An efficient convolution neural network (CNN) plays a crucial role in various visual tasks like object classification or detection, etc. The most common way to construct a CNN is stacking the same convolution block or complex connection. These approaches may be efficient but the parameter size and computation (Comp) have explosive growth. So we present a novel architecture called "DLA+", which could obtain the feature from the different stages, and by the newly designed convolution block, could achieve better accuracy, while also dropping the computation six times compared to the baseline. We design some experiments about classification and object detection. On the CIFAR10 and VOC data-sets, we get better precision and faster speed than other architecture. The lightweight network even allows us to deploy to some low-performance device like drone, laptop, etc.

Keywords: Light weight, image classification, channel attention, efficient convolution, object detection.

Citation: F. T. Wang, L. Yang, J. Tang, S. B. Chen, X. Wang. DLA+: A light aggregation network for object classification and detection. *International Journal of Automation and Computing*, vol.18, no.6, pp.963–972, 2021. http://doi.org/10.1007/s11633-021-1287-y

1 Introduction

Rapid development with architecture of deep convolution neural network (DCNN) in many fields like computer vision (CV), natural language processing (NLP), fingerprint recognition (FR), etc., has been seen in the past decade. With the increase of complexity and difficulty in many artificial intelligence (AI) implementations, we need a higher performance network architecture to cope with various computer tasks. In the earlier computer vision tasks, for any network, the accuracy of models was mainly limited by the depth and structure. Like LeNet[1] or AlexNet[2], the depth of the architecture is no more than 10, it is not far enough to achieve a better result. Subsequently, many excellent network architectures like VGGNet[3], ResNet[4], DenseNet[5, 6], etc. have come out, making great progress in final accuracy. The main reason is that the network architecture is optimized in many aspects, and the capacity to extract semantic information was improved. Of course, it is inevitable, with the depth and width also rising, the number of parameters and Flops have explosive growth. A high-power device is a prerequisite for any network training. The number of parameters tends to over 500 MB or even 1 000 MB, and the demand on a running device's performance is rising. In some recent studies of network structure, it is difficult to employ on new devices or low-performance devices[7, 8], such as mobile phones, ultra-books, laptops, drones, etc. All these devices have poor performance compared with the desktop computers or servers. With the demand for computer vision and development of communication technology, the model after training may run in various devices. If we take a long time to train model and get a bigger training model, it will be difficult to deploy in various low-performance devices.

In prior popular skeletons, because the computation of fully connection has a large proportion of the neural network, then various convolution operation appeared. In order to reduce the computation, in some prior networks, such as LeNet, fully connections are often used in the last layers for classification operations. This operation adds a lot of computation. However, in recently studies such as single shot multibox detector (SSD)[9], researchers found that using a convolution layer instead of fully connection layer still achieved good results. For one thing, this could increase the network's flexibility. On the other hand, it can reduce the computation of forward propagation. So convolution connections are now commonly used instead of fully connections. All the same VGGNet uses a small

Research Article
Manuscript received October 24, 2020; accepted February 1, 2021; published online March 27, 2021
Recommended by Associate Editor Jangmyung Lee
Colored figures are available in the online version at https://link.springer.com/journal/11633

convolution kernel and other methods, but it also requires too many computing resources. So the question becomes how to construct a more efficient skeleton, besides reducing the number of parameters without extra accuracy. Generally, for any network like VGGNet, in terms of structure, which consists of several similar conv-units, each unit consists of a convolution operation, a batch normalization (BN) and a non-linear function (mostly it is ReLu). Researchers use many methods to restrain the parameter size, like replacing the two or three same small convolution kernels with a big convolution kernel to achieve a larger receptive field and save parameter size, when connecting these block-by plane-connections, the final parameters are still enormous. Usually, input image size of classification or detection network is fixed by 224×224 or bigger, sometimes researchers may add pooling operations to reduce the computation and decrease noise to improve the stability and performance of the network. After these adjustments, VGGNet not only raises the depth of the network but also wins the championship for object location in ILSVRC 2014.

Although VGGNet mades impressive progress in depth and accuracy, if researchers use the same method to construct a deeper network, a number of issues come up such as gradient explosion or gradient decent. The depth of the network is helpless to improve the capacity for extracting semantic information. Well, for VGGNet, the limit of depth is 19. The key for constructing networks becomes how to avoid these questions above. In deep residual learning for image recognition, this architecture is also called ResNet. In [4], the main content is given by adding a shortcut to learn the training loss from the shallow layer, such as Fig. 1. This method addresses the overfitting caused by no shortcut. In this way, the depth of the network exceeds 100 even in 1000.

In order to balance the computation, speed and precision, we present the DLA+, the whole architecture can be seen as Fig. 2, and the details about depth-wise asymmetric convolution blocks can be seen as Fig. 3.

2 Related work

In this section, we will review the most related techniques in this paper, mainly the attention for the network, the evolution of the convolution kernel and network architecture.

2.1 Importance of width and depth of network

There has been interesting in designing a deeper and wider neural network. From a depth point of view, in the first few years, the depth of the neural network is very shallow like LeNet, AlexNet, etc., and the ability to extract semantic information is too weak, it is not enough for some computer vision task[10–12]. In some object detection tasks, it requires rich representation, the size of feature map need various scales. Therefore, we need a deeper network[13]. On the other hand, most construct methods of computer vision networks is stack the same conv-unit. With the depth deepen, receptive field (RF) becomes larger and the size of feature map becomes smaller. In most cases, we would shrink the size of feature map by a factor 16× or 32×[14, 15], it brings new problems: In the last layer of network, the feature map size is too small and more sensitive for large object. In the front layers, the feature map size is too large and more sensitive for small object. So the final result cannot always be satisfied because when the multi-scale object exists as a single image, this only improves the depth of the network and cannot make the ability of the last layer more sensitive to the multi-scale objects.

So there are various feature aggregation architectures like FPN[4], please refer to Fig. 4. This research is mainly to solve the problem in multi-scale detection, now many of the networks are accessed using a single high-level features (such as faster-R-CNN using the convolution layer from Conv4, which is used for subsequent object classification and bounding box regression), but there is an obvious flaw in this method: The small pixel information of the object is less, it is easy to be lost in the process of down sampling. In order to deal with the detection problem with an obvious difference in multi scale feature, the classical method is to use an image pyramid to enhance the multi-scale variation, but this will bring a great amount of computation. FPN provides a good solution. From the structure of the network, FPN is a multi-branch architecture, it has mainly two path: top-down and bottom-up. The bottom-up path is a convolution path with batch-normalization and pooling, the output feature map size is the input one-four. In annother top-down pathway, reasearchers upsample the spatial resolution by a factor of 2. The top-down pathway merge with buttom-up pathway by lateral connection. By doing so, this architecture solved the problem of low precision on a small object, but due to the addition of extra bottom-up branch and many 1×1 convolutions, there is a significant increase in computation.

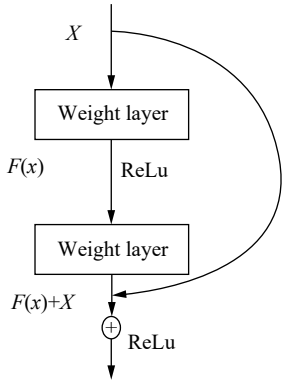


Fig. 1 Brand new depthwise asymmetric convolution block

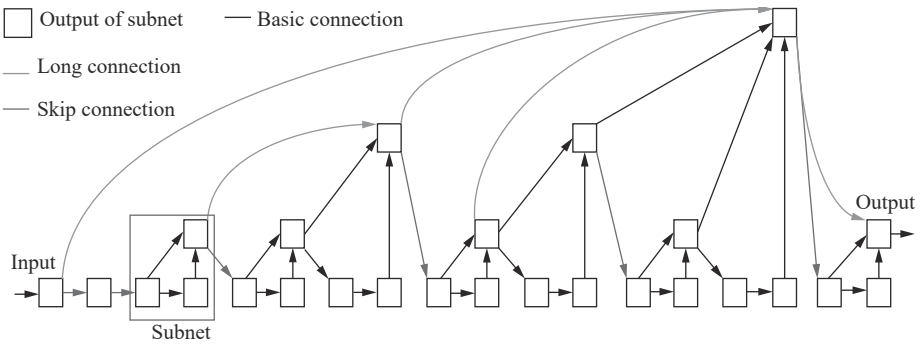


Fig. 2 Architecture of DLA+

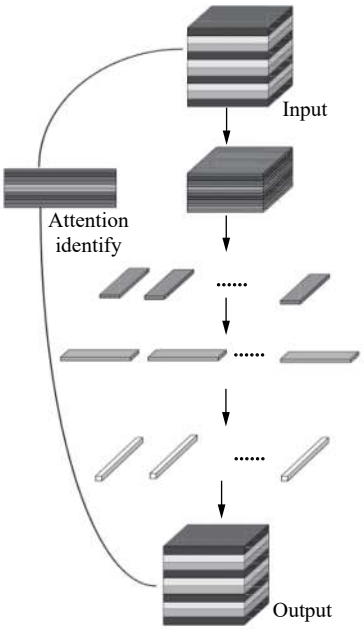


Fig. 3 Architecture of residual block of ResNet

For computer visual tasks, another type is multi-scale detection like SSD, which uses the last 2 layers of backbone and adds an extra 4 layers to detect a multi-scale object directly without the last layer to predict. This form is faster than FPN, besides it could get a Satisfactory accuracy[16, 17]. But this backbone is modified from VGGNet, and because it uses an extra convolution layer, the parameter size is still large. So we present DLA+, which is based on deep layer aggregation (DLA)[18], it is constructed by several same stages, and every stage consists of the same conv-units. For the first stage at the bottom of the network, we use neighboring conv-units to construct a mini network. And we use the same method to construct the upper layer. In order to be able to balance accuracy of large and small objects, we add a long-connection to connect the upper feature and lower feature.

2.2 Standard convolution with asymmetric factorization

Factorization convolutions have been raised in rethinking the inception architecture for computer vision (Inception-V3) from Szeged et al.[4] In inception-V3[19], Ku et al. replace any 1×1 convolution by a $1 \times n$ followed by 1×1 convolution. The theory basis of factorization convolution is simple: If the rank of the 2D-kernel is one, the asymmetric convolution after factorization is equal to standard convolution. Suppose $n = 3$, it could reduce parameters by 33%. And with the value of n changed, the parameter size may decrease by factor n. Fig. 5 shows asymmetric convolution.

In general, the ability to extract semantic information will become more powerful with the improvement of network depth, so this undoubtedly will make the size of the parameter larger, and the nonlinear network will be improved with the depth deepening. However, factorizing convolutions is an amount to deepen the depth, not only

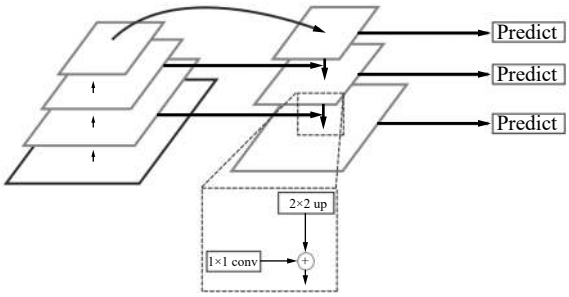


Fig. 4 The typology of FPN, the left part is the bottom-up branch and the right side is top-down branch. Researchers concat the features in the same stage by literal connected, which is consisted of 1 convolution and 2×2 up-sample.

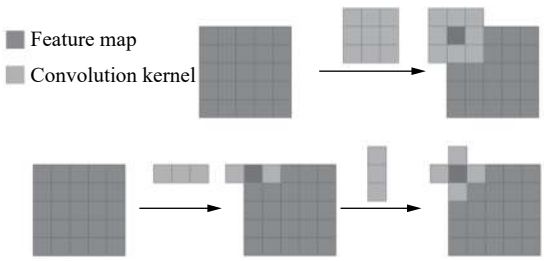


Fig. 5 The asymmetric convolution. Square convolution is decomposed into several rectangle convolutions, it could drop the parameter size by 5, and learn features from horizontal and vertical directions.

reduce the parameter size but also improve the fitting effect. Yang et al.[20, 21] introduces that asymmetric convolution is not applicable for replacing the standard convolution for the whole architecture, but it will get a good result when asymmetric convolution located in the mid-posterior of a network, otherwise easily lead to the side effect[22].

2.3 Details regarding depth-wise convolution

Depth-wise separable convolution is a special form of group convolution[23], which first appeared in AlexNet and could be running over two GPUs, because the GPU-memory is too low to undertake AlexNet. In the recent study such as MobileNet, Xception, because of the use of depth-wise convolution, researchers designed more efficient architecture. Mostly, researchers will directly perform a batch normalization (BN) followed by a non-linear function operation after depth-wise separable convolution as shown in Fig. 6. For any standard convolution layer with a input size as $D_I \times D_I \times M$, output size is $D_O \times D_O \times N$, and kernel size is $D_K \times D_K$, where D_K represents the size of filters, M and N represent numbers of input and output channels, the standard convolution computations is

$$C_{\text{std}} = D_I \times D_I \times M \times N \times D_k \times D_k. \tag{1}$$

Compared with standard convolution, the computation of depth-wise convolution can be represented as C_{dw}:

$$C_{\text{dw}} = D_I \times D_I \times M \times D_I \times D_I. \tag{2}$$

But the initial process of depth-wise separable convolution is channeled one by one[24], and we have found this approach may lose the relevance between channels. In order to narrow down the gap, channel attention is available to this problem. The idea is proposed in [18, 25], and as squeeze-and-excitation networks[26], one of the most important is the new channel attention block. To mitigate the problem of weak relevance, she squeezes the global spatial information by using global average pooling as weight, and multiplies it with the output after depth-wise convolution. If the size of input feature map D_I is $H \times W \times C$, by this operation, the feature size goes into $1 \times 1 \times C$. The weight w is calculated by

$$w = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} D_I(i, j). \tag{3}$$

Suppose the standard convolution output is D_O, after the channel attention, the output can be expressed as wD_O, and the squeeze block is easy to insert into any skeleton after nonlinear function for each conv-unit. There are not many parameters of the squeeze block and it is especially helpful for the issue of depth-wise convolution[27].

3 Network architecture

In this section, we will give instructions about the constitution of DLA+.

In the original DLA, the conv-unit consists of convolution (mostly the size is 3×3), BN and a non linear function (ReLu or others) as. In DLA+, we use the depth-wise asymmetric convolution blocks instead of the original conv-unit, Fig. 3 displays the shape of the new unit. The re-designed conv-unit is based on depth-wise and asymmetric convolution. As we can see in Section 2, the large computation of the standard convolution depends on D_I, D_K, M and N. Suppose $N = 3$, skeleton uses 3×3 depth-wise convolutions which use about 8 times less computation than standard convolutions.

It is easy to see the computation cost after depth-wise convolution operation depends on the value of D_I, D_K and M. But the values of D_I, M cannot change easily, in general, there is a more popular method to save computation by control D_I, M, like pooling and changing the value of stride. If we still reduce the size of the feature map, it may have a bad effect on the representation abilities. So we decomposed the standard convolution into a asymmetric convolution for each depth-wise convolution. As introduced in Section 2.2, after replacing the standard convolution with depth-wise convolution, the computation can be represented as follows:

$$\frac{D_I \times D_I \times M \times D_I \times D_I}{D_I \times D_I \times M \times N \times D_I \times D_I} = \frac{1}{N}. \tag{4}$$

If we depose the square kernel to asymmetric kernel,

$$\begin{aligned} \cos t &= \frac{D_I \times D_I \times M \times D_k \times 1 + D_I \times D_I \times M \times 1 \times D_k}{D_I \times D_I \times M \times N \times D_I \times D_I} \\ &= \frac{2}{N \times D_k}. \end{aligned} \tag{5}$$

By this equation we can see if the convolution kernel size is 3×3, it will save about 33% further parameters.

3.1 Attention on depth-wise separable convolution

In Section 2.2, we introduced the problem of depth-wise convolution: information transmits obstructed caused by the relevance between channels. Each channel of the feature map provides significant guidance information for analyzing its image. So it is essential to pay more attention to find where it is meaningful for any input image.

In order to capture the relationship, Woo et al.[25] suggests learning a relevance between channels, the atten-

tion maps are multiplied to input feature. The multiplier is caculate by average pooling for each input feature channels. Beyond the previous study, we empirically argue that which could improve representation ability. The core of DLA+ is the multi-path conv-unit. Fig. 6 shows the depth-wise asymmetric convolution branch. Fig. 7 shows the final conv-unit after combining 2 branches.

As shown in Fig. 7, suppose shown in an input feature map $D_I \subset \mathbf{R}^{H\times W\times C}$, D_O is the output without attention, and make same padding to keep the same size of input and output, the value of channel attention can be seen as a 1D vector and the value can be obtained by the next function:

$$Mc = \delta(AvgPool(D_I)) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} v(i,j) \tag{6}$$

and then multiplied with

$$D_o^{'} = M_c \otimes D_O. \tag{7}$$

The $\otimes$ represents the giving weights to the output.

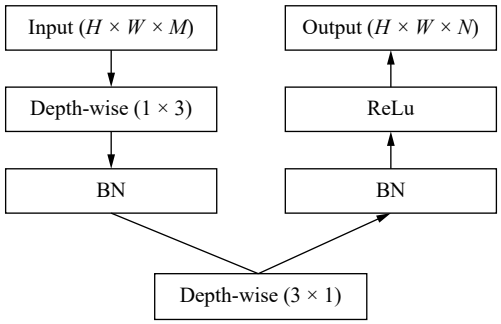


Fig. 6 Typology of asymmetric depth-wise separable convolution branch

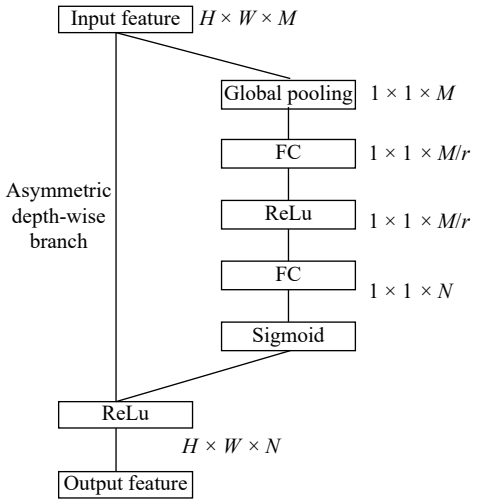


Fig. 7 Architecture of attention branch embedded in depth-wise asymmetric convolution as the above figure demonstrates. The left side: asymmetric depth-wise convolution could be seen as Fig. 6.

Characteristics of depth-wise separable are convolution operations based on a single-channel, and combined into an output feature. These values of each channel in the 2D-kernel are not equal, so the feature of each channel may lose relevance. Therefore, in order to utilize the information between channels, we sequence the channel information before depth-wise. If the input feature map size is $H \times W \times M$, and output size is $H \times W \times N$, we change the shape to $1 \times 1 \times M$ by using the global average pooling. We adopt the channel attention method, which is similar to SeNet, and we use two fully connected layers in attention branch. And notice that if the channel of input is not equal to output, in the second fully connected, we will multiply M/r by factor rM/N to keep the number of attention channels equal to those of output channels. The final conv-unit is shown as Fig. 7.

3.2 Densely connected architecture with subnet

In order to take advantage of subnet features from previous stages and layer, we add more long-connected in DLA+, as depicted in Fig. 2, which can be seen as an deep supervision structure[28, 29]. Owing to the dense long connect, DLA+ generates more features without arising the parameter size and computation compared with original network.

Finally, there are three difference from DLA and original baseline: 1) DLA+ have depth-wise convolution unit; 2) Compare with original architecture, these are more densely connected between each subnet; 3) We add an attention branch to obtain the semantic information.

4 Experiments

We evaluate DLA+ on these object classification tasks: Cifar10[30] and ImageNet100[31]. The latter is a subset of ImageNet2012[32], which is divided into 100 classes with 1 000 JPEG images; and PASCAL VOC2007[33, 34] for object detection. We compare DLA+ with networks which have been reproduced in the Pytorch framework.

In order to prove the validity of DLA+, we first perform experiments with the baseline: original deep layer aggregation[35, 36]. Besides, we have performed several experiments on object classification and detection.

4.1 Ablation study

4.1.1 Separable convolution

In order to verify the effectiveness, we only use separable convolution in DLA+, first we compare it with other networks that use depth-wise convolution. All networks trains 200 epochs and on 1 Titan Xp GPU. The final results are shown in Table 1. We evaluate three aspects: parameter size, Flops and accuracy. Notice that compared with ShuffleNet, the parameter size is lower than DLA+, but the Flops is too high because the pro-

cess of the channel shuffle operation in ShuffleNet costs many computations. The result can be seen in Table 1.

Table 1 Result of adding depth-wise convolution on Cifar10

Net	Lr	Train Acc(%)	Test Acc(%)	Flops(MB)	Comp(%)
VGG16_bn	0.01	97.84	91.13	527.8	527.8
ShuffleNet	0.01	98.71	92.93	30	30
DenseNet	0.01	98.53	92.37	43.63	43.63
Baseline	0.01	98.64	92.51	163	163
DLA+(dw)	0.01	95.93	86.83	70	70

In the Table 1, dw represents the depth-wise convolution. The result of Cifar10 after using depth-wise convolution can be seen in the last line, the parameter size has decrease of about 52%. And then, we test our model in Cifar10 and compare the video-memory cost with other networks shown in Fig. 8.

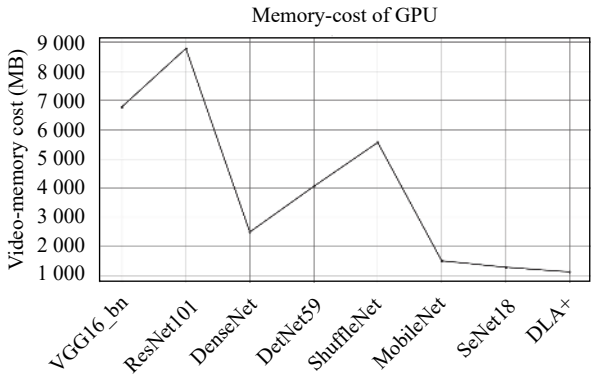


Fig. 8 GPU video-memory cost on Cifar10

4.1.2 Asymmetric convolution

The 2D-kernel shape of the depth-wise convolution is $n \times n$, we split the square 2D-kernel into $n \times 1$ and $1 \times n$ in each depth-wise convolution, which lets the network learn features by two directions: vertical and horizontal, besides reducing more than 33% parameters in DLA+. In the original DLA, each conv-unit consists of a two layer convolution operation, but in DLA+, which is replaced by asymmetric convolution, the new conv block can be seen as a four layer convolution operation as in Fig. 7. In the prior work[37], researchers propose a hypothesis: In the process of designing a network or conv-unit, ReLu could increase the ability of nonlinear and reducing the computation, in addition, ReLu could reducing the possibility of over-fitting. But if too many ReLus are used in the same conv-unit, it will could adversely impact. So we removed ReLu and put it after the concat feature[38] (add by asymmetric depth-wise branch and channel-attention branch).

Compared with the original depth-wise convolution, the depth after deposing the square convolution for each depth-wise convolution block is the amount to deepen, so we made some adjustment for depth-wise asymmetric convolution blocks: We remove the ReLu between two asymmetric convolutions, in order to obtain a better training result, we replace the ReLu to ReLu6. This operation is in favor of deployment on the mobile device.

4.1.3 Channel attention

Because the initial process of depth-wise convolution in channel is one, the relevance in each feature is underused, in order to obtain this relation, we used channel attention on both sides with the asymmetric depth-wise convolution. Results are shown in Table 2.

In Table 2, ch represents the channel attention, and 1n represents the deposed standard square convolution with asymmetric convolutions.

It is easy to see from Table 3, compared with other multi-branch or multi-scale architecture such as FPN, DLA+ achieves a faster speed on the Cifar10. The main reason is that a large number of depthwise asymmetric convolution modules are used in the DLA+, depthwise separable convolution disintegrates the standard convolution, asymmetric convolution also has a very good ability to make up for the square convolution feature representation from vertical and horizontal. In general, square convolution is more suitable for running on GPU. These structures are more suitable for processing large amounts of parallel data. But for none GPU device or low performance device, it is more suitable for running DLA+, because depthwise separable tends to serial computing. In order to further enhancing the capability of feature representation, we use the channel attention module to get better channel features to achieve better results on advanced visual tasks, meanwhile it does not generate too many extra parameters.

Table 2 Results of adding channel attention and asymmetric convolution on Cifar10

Net	LR	Train Acc	Test Acc	Comp(MB)
Baseline	0.025	97.25	88.02	60
DLA+(dw)	0.025	93.3	85.2	9
DLA+(dw+ch)	0.025	94.1	84.8	10.1
DLA+(dw+1n)	0.025	93.93	87.64	50.8
DLA+	0.025	97.49	90.45	9.81

Table 3 Result on Cifar10. Final result and the computation is calculated by fixing the input size as 224 × 224.

Net	Lr	FPS	Train Acc(%)	Test Acc(%)	Comp(%)
VGG16_bn	0.025	4 761	97.84	91.13	527.8
DenseNet121	0.025	2 381	98.71	92.93	30
SENet_Resblock	0.025	4 761	98.53	92.37	43.63
Preact_Res101	0.025	2 941	98.64	92.51	163
DetNet59	0.025	4 347	95.93	86.83	70
ResNet101	0.025	2 500	98.55	91.97	339
DLA+	0.025	4 981	99.74	90.45	9.81

4.2 Image classification

From Sections 2 and 3, we know that an efficient neural network could make the best use of each conv-unit. To prove the principle, we adopted train from scratch for all experiments. First, we perform ImageNet100 classification to evaluate our network, and we compare DLA+ with other networks such as ResNet, DetNet and VGGNet. The results of experiments are shown in Table 4.

Table 4 Results of classification on ImageNet100

Net	Lr	FPS	Top1 Acc	Top5 Acc
ResNext	0.1	205	61.47	84.36
ResNet	0.1	302	62.94	85.19
VGGNet6	0.1	228	50.04	76.52
DetNet	0.1	230	63.84	85.36
SequenceNet	0.1	163	35.50	63.21
Baseline	0.1	147	63.25	85.16
DLA+	0.1	222	65.31	86.23

We train the network for 90 epochs and batch size is 256. The initial learning rate is 0.1, and lowered by 10 times at epoch 30 and 60, respectively. As we can see from Table 4, the performance of DLA+ is beyond baseline and other networks. Notice that the parameter size and Flops of DLA+ are smaller than other networks[39].

4.3 Object detection

From Sections 2 and 3, we know that an efficient neural network could make the best use of each conv-unit. To prove the principle, we adopted train from scratch for all experiments.

CenterNet[40] was used as the object detection framework, and we use the PASCAL VOC dataset, which contains 5 011 training images and 4 952 test images, the final training converges in 50 epochs. The original skeleton deep layer aggregation (DLA) was replaced with DLA+. We fix the input size as 384×384, other hyper-parameters are the same as CenterNet.

We compare it with faster RCNN based on VGG16, and SSD, MobileNet and EfficientNetB3. The results are shown in Table 5. All devices use TITANXP Xp GPU and Xeon E5-2 620 CPU. It is easy to see the DLA+ is better than baseline on both accuracy and speed from, as shown Table 1. Besides, the two-stage detection framework such as faster RCNN cannot undertake the training from the scratch task so that RCNN cannot converge to global feature. Likewise, compared with other implements, DLA+ shows the trade-off between accuracy and speed.

5 Conclusions

In order to address the large computation cost in neural network training and employment for many mobile devices, we present DLA+, which is a smaller and faster skeleton. Compared to other multi-branch networks, the result also indicates that the measures of DLA+ achieve a reasonable trade-off of accuracy and speed or parameter size compared with the baseline and another popular skeleton. We put many experiments with many visual tasks like object classification and detection to prove the implementation in DLA+ is valid. And we will release the code in Pytorch.

Acknowledgements

The authors would like to thank the editor and anonymous reviewers for their valuable comments and sug-

Table 5 Final training result on Pascal VOC. We compare the DLA+ with some advanced networks, including EfficientNet, MobileNet, etc.

Method	Plane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table
FRCNN	0	0	0	0	0	0	0	0	0	0	0
EfficientNetB	0.65	0.71	0.43	0.47	0.2	0.69	0.74	0.67	0.37	0.5	0.65
MobileNet	0.52	0.6	0.34	0.3	0.1	0.59	0.64	0.56	0.28	0.36	0.58
SSD_VGG	0.62	0.67	0.4	0.42	0.17	0.67	0.74	0.57	0.31	0.5	0.58
Baseline	0.58	0.72	0.54	0.36	0.36	0.6	0.77	0.65	0.35	0.43	0.5
DLA+	0.63	0.73	0.52	0.34	0.44	0.66	0.77	0.65	0.41	0.39	0.49
	Dog	Horse	Motor	Person	Potted	Sheep	Sofa	Train	TV	mAP	FPS
	0	0	0	0	0	0	0	1	0	0	5
	0.61	0.76	0.7	0.67	0.28	0.46	0.64	0.76	0.57	0.58	31
	0.51	0.69	0.65	0.58	0.16	0.36	0.52	0.69	0.49	0.48	41
	0.53	0.73	0.67	0.65	0.22	0.45	0.53	0.72	0.53	0.53	54
	0.53	0.68	0.67	0.73	0.32	0.35	0.58	0.66	0.52	0.54	33
	0.62	0.75	0.75	0.73	0.74	0.5	0.56	0.66	0.55	0.59	39

gestions, which are very helpful in improving this paper. And this work was supported by University Synergy Innovation Program of Anhui Province (No. GXXT-2019-007), Corporative Information Processing and Deep Mining for Intelligent Robot (No. JCYJ20170817155854115), Major Project for New Generation of AI (No. 2018AAA0100400), Anhui Provincial Natural Science Foundation (No. 1908085MF206).

References

[1] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Q. Jia, K. M. He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. [Online], Available: https://arxiv.org/abs/1706.02677, 2017.

[2] X. H. Ding, Y. C. Guo, G. G. Ding, J. G. Han. ACNet: Strengthening the kernel skeletons for powerful CNN via asymmetric convolution blocks. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, Korea, pp. 1911–1920, 2019. DOI: 10.1109/ICCV.2019.00200.

[3] A. Borja, A. B. Josefson, A. Miles, I. Muxika, F. Olsgard, G. Phillips, J. G. Rodríguez, B. Rygg. An approach to the intercalibration of benthic ecological status assessment in the North Atlantic ecoregion, according to the European Water Framework Directive. *Marine Pollution Bulletin*, vol. 55, no. 1–6, pp. 42–52, 2007. DOI: 10.1016/j.marpolbul.2006.08.018.

[4] S. Ioffe, C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, ACM, Lille, France, pp. 448–456, 2015.

[5] H. Law, J. Deng. CornerNet: Detecting objects as paired keypoints. In *Proceedings of the 15th European Conference on Computer Vision*, Springer, Munich, Germany, pp. 734–750, 2018. DOI: 10.1007/978-3-030-01264-9_45.

[6] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, K. Keutzer. DenseNet: Implementing efficient ConvNet descriptor pyramids. [Online], Available: https://arxiv.org/abs/1404.1869, 2014.

[7] B. Hu, J. C. Wang. Deep learning based hand gesture recognition and UAV flight controls. *International Journal of Automation and Computing*, vol. 17, no. 1, pp. 17–29, 2020. DOI: 10.1007/s11633-019-1194-7.

[8] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 770–778, 2016. DOI: 10.1109/CVPR.2016.90.

[9] B. Xu, N. Y. Wang, T. Q. Chen, M. Li. Empirical evaluation of rectified activations in convolutional network. [Online], Available: https://arxiv.org/abs/1505.00853, 2015.

[10] C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, 2015. DOI: 10.1109/CVPR.2015.7298594.

[11] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. DOI: 10.1109/5.726791.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. H. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, F. F. Li. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. DOI: 10.1007/s11263-015-0816-y.

[13] S. Q. Ren, K. M. He, R. Girshick, J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. DOI: 10.1109/TPAMI.2016.2577031.

[14] T.-Y. Lin, P. Dollár, R. Girshick, K. M. He, B. Hariharan, S. Belongie. Feature pyramid networks for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Honolulu, USA, pp. 2117–2125, 2017. DOI: 10.1109/CVPR.2017.106.

[15] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015. DOI: 10.1109/TPAMI.2015.2389824.

[16] N. N. Ma, X. Y. Zhang, H.-T. Zheng, J. Sun. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In *Proceedings of the 15th European Conference on Computer Vision*, Springer, Munich, Germany, pp. 116–131, 2018. DOI: 10.1007/978-3-030-01264-9_8.

[17] J. Hu, L. Shen, G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 7132–7141, 2018. DOI: 10.1109/CVPR.2018.00745.

[18] F. Yu, D. Q. Wang, E. Shelhamer, T. Darrell. Deep layer aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 2403–2412, 2018. DOI: 10.1109/CVPR.2018.00255.

[19] Y. D. Ku, J. H. Yang, H. Y. Fang, W. Xiao, J. T. Zhuang. Optimization of grasping efficiency of a robot used for sorting construction and demolition waste. *International Journal of Automation and Computing*, vol. 17, no. 5, pp. 691–700, 2020. DOI: 10.1007/s11633-020-1237-0.

[20] X. Yang, H. Sun, X. Sun, M. L. Yan, Z. Guo, K. Fu. Position detection and direction prediction for arbitrary-oriented ships via multitask rotation region convolutional neural network. *IEEE Access*, vol. 6, pp. 50839–50849, 2018. DOI: 10.1109/ACCESS.2018.2869884.

[21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 2818–2826, 2016. DOI: 10.1109/CVPR.2016.308.

[22] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, K. Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size. [Online], Available: https://arxiv.org/abs/1602.07360, 2016.

[23] A. G. Howard, M. L. Zhu, B. Chen, D. Kalenichenko, W. J. Wang, T. Weyand, M. Andreetto, H. Adam. MobileNets: Efficient convolutional neural networks for mobile vis-

ion applications. [Online], Available: https://arxiv.org/abs/1704.04861, 2017.

[24] M. Sandler, A. Howard, M. L. Zhu, A. Zhmoginov, L.-C. Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 4510–4520, 2018. DOI: 10.1109/CVPR.2018.00474.

[25] S. Woo, J. Park, J. Y. Lee, I. S. Kweon. CBAM: Convolutional block attention module. In *Proceedings of the 15th European Conference on Computer Vision*, Springer, Munich, Germany, pp. 3–19, 2018. DOI: 10.1007/978-3-030-01234-2_1.

[26] X. X. Chu, B. Zhang, R. J. Xu. MoGA: Searching beyond MobileNetV3. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, Barcelona, Spain, pp. 4042–4046, 2020. DOI: 10.1109/ICASSP40776.2020.9054428.

[27] K. Simonyan, A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, USA 2015.

[28] Z. Hu, X. Li, T. Guan. A study on performance and reliability of urethral valve driven by ultrasonic-vaporized steam. *International Journal of Automation and Computing*, vol. 17, no. 5, pp. 752–762, 2020. DOI: 10.1007/s11633-016-1026-y.

[29] Q. V. Le, J. Ngiam, Z. H. Chen, D. Chia, P. W. Koh, A. Y. Ng. Tiled convolutional neural networks. In *Proceedings of the 23rd International Conference on Neural Information Processing Systems*, ACM, Red Hook, pp. 1279–1287, 2010.

[30] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Honolulu, USA, pp. 1251–1258, 2017. DOI: 10.1109/CVPR.2017.195.

[31] Z. W. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, J. M. Liang. UNet++: A nested U-Net architecture for medical image segmentation. In *Proceedings of the 4th International and 8th International Workshop Workshop on Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Springer, Granada, Spain, pp. 3–11, 2018. DOI: 10.1007/978-3-030-00889-5_1.

[32] A. Krizhevsky, I. Sutskever, G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, ACM, Lake Tahoe, Nevada, USA, pp. 1097–1105, 2012.

[33] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010. DOI: 10.1007/s11263-009-0275-4.

[34] Y. L. Li, S. J. Wang, Q. Tian, X. Q. Ding. Feature representation for statistical-learning-based object detection: A review. *Pattern Recognition*, vol. 48, no. 11, pp. 3542–3559, 2015. DOI: 10.1016/j.patcog.2015.04.018.

[35] S. N. Xie, R. Girshick, P. Dollár, Z. W. Tu, K. M. He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Honolulu, USA, pp. 1492–1500, 2017. DOI: 10.1109/CVPR.2017.634.

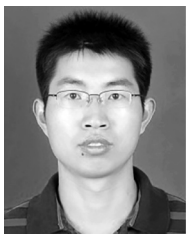
[36] O. Ronneberger, P. Fischer, T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Proceedings of the 18th International Conference on Medical image Computing and Computer-Assisted Intervention*, Munich, Germany, Springer, pp. 234–241, 2015. DOI: 10.1007/978-3-319-24574-4_28.

[37] Z. M. Li, C. Peng, G. Yu, X. Y. Zhang, Y. D. Deng, J. Sun. DetNet: A backbone network for object detection. [Online], Available: https://arxiv.org/abs/1804.06215, 2018.

[38] K. W. Duan, S. Bai, L. X. Xie, G. H. Qi, Q. M. Huang, Q. Tian. CenterNet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, Korea, pp. 6569–6578, 2019. DOI: 10.1109/ICCV.2019.00667.

[39] X. Y. Zhang, X. Y. Zhou, M. X. Lin, J. Sun. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 6848–6856, 2018. DOI: 10.1109/CVPR.2018.00716.

[40] M. X. Tan, Q. V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning*, Long Beach, California, USA, pp. 6105–6114, 2019.



Fu-Tian Wang received the B. Eng. degree in computer science and technology, the M. Eng. degree in computer application and the Ph. D. degree in computer software and theory from Anhui University, China in 2005, 2009 and 2017 respectively. He has been a teacher in Anhui University, China from 2009.

His research interests include image processing, computer vision and edge computing.

E-mail: wft@ahu.edu.cn

ORCID iD: 0000-0003-4181-8485

Li Yang received the B. Eng. degree in electrical engineering and automation from Luoyang Institute of Technology, China in 2017. He is currently a master student in computer science and technology, Anhui University, China.

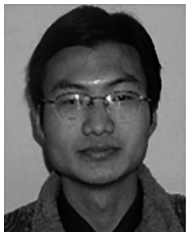
His research interests include computer vision, object detection and model compression.

E-mail: 543320032@qq.com

Jin Tang received the B. Eng. degree in automation and the Ph. D. degree in computer science from Anhui University, China in 1999 and 2007, respectively. He is currently a professor with School of Computer Science and Technology, Anhui University, China.

His research interests include computer vision, pattern recognition, machine learning and deep learning.

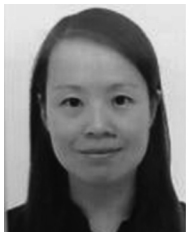
E-mail: tangjin@ahu.edu.cn



Si-Bao Chen received the B. Sc. and M. Sc. degrees in probability and statistics and the Ph.D. degree in computer science from Anhui University, China in 2000, 2003 and 2006, respectively. From 2006 to 2008, he was a postdoctoral researcher at Department of Electronic Engineering and Information Science, University of Science and Technology of China. From 2008, he has been a teacher in Anhui University. He was a visiting scholar at University of Texas at Arlington, USA from 2014 to 2015.

His research interests include image processing, pattern recognition, machine learning and computer vision.

E-mail: sbchen@ahu.edu.cn



Xin Wang received the B. Sc. degree from Department of Precision Machinery and Precision Instruments, University of Science and Technology, China in 1998. Now, she′s the technical director of Shenzhen Raixun Information Technology Co., Ltd., and the Researcher of Peking University Shenzhen Institute, China.

Her research interests include multimedia information processing, speech recognition and Internet Security.

E-mail: wangxin@imsl.org.cn (Corresponding author)

ORCID iD: 0000-0001-7042-2637