**Application of AcNet in lightweight models for image classification**

# Abstract

This paper proposes a novel approach to optimize the lightweight model by combining global inference with local inference to obtain the mobile-platform optimized image classification model. Two novel image classification models, the ACNet-based MobileNet and the ACNet-based GhostNet are proposed in this paper, respectively. One excellent example of lightweight models was MobileNet, created by the Google team in recent years. However, the model parameters of MobilNet are so few that its accuracy still can not compare with other large-scale network models. Although the recent GhosNet lightweight model proposed by the Huawei team is more accurate in image classification than MobileNetV3, it is still a far cry from the larger network model. According to the characteristics of the traditional convolutional neural network, ACNet [2] proposes an improved algorithm, which can flexibly change the characteristic internal performance of global reasoning and local inference and improve the classification accuracy. We believe that ACNet can compensate for traditional convolution problems, thus improving the performance of lightweight models. Therefore, two models, The ACNet-based MobileNet and The ACNet-based GhostNet have been proposed to make The model more widely adaptable. It also could improve the accuracy of image classification slightly. The code is available at https://github.com/tommywhy/acnet\_mobilenetv3

# Introduction

With the development of the deep learning domain, lightweight image classification models have been proposed. Google's MobileNet significantly reduces the parameters of the model by using depthwise separable convolutions(DWS)[3], which makes a meaningful contribution to porting to mobile devices. Even though mobileNetV3 has been improved by 3.2% in accuracy compared with the mobileNetV2[4] in ImageNet classification through the inverted residual with linear bottleneck and squeeze and excitation structure, the accuracy is still not comparable to other large-scale network models, such as ResNet[5] and VGG[6]. Ghost Module, mentioned by the Huawei team in GhostNet, can further improve the lightweight model’s image classification accuracy. However, it still has the disadvantage of low accuracy compared with the large-scale network. How to improve the accuracy of lightweight deep neural networks under the premise that the model volume is small enough has become the key to the successful application of deep learning in the field of the mobile platform.

Simultaneously, more and more models use the Convolutional neural network (CNN) as a vital part of the model with the large-scale application of deep learning in image classification and target detection. However, the limitations of CNN itself have also been continuously confirmed. Due to CNN only extracts information from local neighboring pixels, each layer in the convolutional network does not have an excellent global inference ability. Therefore, the convolution operation cannot distinguish two similar objects well. Nevertheless, ACNet - Adaptively Connected Neural Networks can effectively solve this problem. ACNet ingenious combines global reasoning with local reasoning. In the author’s opinion, ACNet can be used to optimize and reconstruct the DWS in MobileNet and effectively avoid CNN focusing too much on local reasoning and improving the accuracy. ACNet can also be used to distinguish the redundancy in feature maps for Ghost Module in GhostNet better.

# Background and Literature review

Image classification technology is becoming more and more mature, and it is trying to be applied to more and more fields. Many excellent models have been proposed, and the classification of images can achieve a high accuracy rate. It is due to the widespread use of traditional convolution operation. Since LeCun's[7] paper on CNN was published, the classical structure of CNN has been widely used in image classification. At the same time, large-scale datasets and the continuous improvement of computer computing power, deep CNN is continuously applied to image classification. The most famous is that Krizhevsky[8] proposed a prominent AlexNet CNN structure and gained a significant breakthrough in image recognition. AlexNet was a great success, setting off a research boom in convolutional neural networks. After this, the researchers put forward other improvement techniques. Based on the AlexNet model, a more layered and deeper VGGNet[6] model was proposed to solve image classification and achieve higher accuracy. GoogLeNet[9] adopts the idea of an Inception structure to enrich models' diversity. ResNet[5] took the lead in proposing the concept of residual network, which effectively suppressed the overfitting of the model. During this period, CNN's model is continuously developing in a more complex and deeper direction. ResNet's score in the ILSVRC 2015[10] competition is more than 20 times that of AlexNet and more than eight times that of VGGNet. Then ResNeXt proposed a group convolution module to once again break through the accuracy of image classification. By increasing the depth, the network can obtain a more accurate non-linear objective function so that the model can better reflect the characteristics. However, increasing the network's overall complexity makes the model too large, only staying in the laboratory stage.

For this reason, how to apply deep learning models to real-life has attracted more and more attention. The Google team has made many contributions to this. The Google team has proposed MnasNet[8] and MobileNet. Among them, MobileNet uses technologies such as an inverted residual network and DW convolution to reduce the model parameters significantly. MobileNet The method of DW convolution used by MobileNet is essentially the extreme form of packet convolution in ResNeXt, which means each channel is convolution in a group. Although the model is small enough to be applied to mobile devices, the reduced model parameters also bring a side effect of decreasing accuracy. The new GhostNet proposed by the Huawei team uses Ghost Module to optimize the redundant feature map, thus obtaining a lightweight image classification model. The optimization contributes to GhostNet better classification accuracy than mobilenetv3. However, The accuracy is still not at the same level as the extensive networks.

Simultaneously, another defect of CNN is also reflected on MobileNet and GhostNet: CNN only extracts information from neighboring local pixels, so each layer inside the convolutional network does not have an excellent global knockdown capability[9]. Therefore, the convolution operation cannot distinguish two similar objects well in lightweight models. The ability to distinguish between two similar objects on a mobile device is another major bottleneck for deep learning applications.

This paper's primary purpose is to discuss how to use AcNet to build a deep learning model with a small parameter model and high precision. Aiming at the shortcomings of CNN, we use ACNet's flexible parameter storage method to learn the ability to transform general data local and global reasoning and propose the ACNet-based MobileNet and the ACNet-based GhostNet network models. In this way, a brand-new, mobile-friendly, and high-precision image classification model is obtained.

# Methodology

**Architecture**

Guangrun Wang proposed the ACNet mention formula(1) which uses the weights of β and γ to control CNN and MLP adaptively. It makes a specific layer of the model have both global inference and local inference. The formula also mentions a weight α, which is the weight that controls the transformation of itself. The three weight values are automatically updated by backpropagation to make it self-adaptive. It is easy to see from the formula that when the weight of α and γ is 0, the formula represents the traditional CNN. When the weights of α and β are 0, the formula represents an MLP operation. When the weight of β and γ is 0, the formula represents a 1\*1 convolution operation.

Inspired by ACNet, We proposed the formula(2) which contains two parts of global inference and self-transformation and is controlled by weights γ and α, respectively. When the formula(2) applied to the 1\*1 convolution operation, it can change the dimension and at the same time has the ability of global inference.

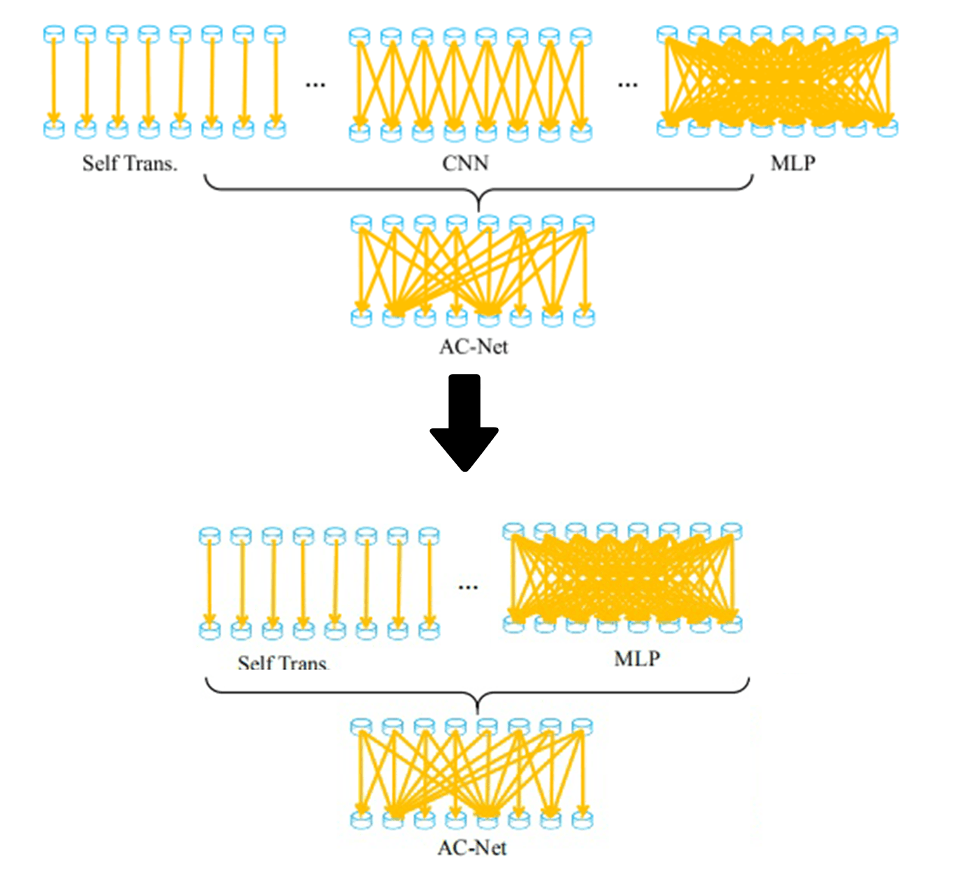


Figure 1: formula(2) - The variant structure of ACNet.

**ACNet-based MobileNet**

By analyzing the network structure of MobileNet, the unique design method of the inverted residuals module is using a 1\*1 convolution operation to change the dimension of inputs. The dimensional change will lose a large of redundancy information. It is the most crucial reason why MobileNet is less accurate than other large networks. The ACNet-based MobileNet, which the author proposed, uses formula(2) to optimize the 1\*1 convolution operation in the inverted residual module. It makes the inverse residuals module have a certain global reasoning ability. The X is the input layer, ,​, represent the learnable weights. Use the global transformation or the self-transformation for self-inference through the two weight parameters of α,γ. Due to the adaptive updating weights via backpropagation, the convolution operation can preserve the part of the image's useful information to the greatest extent by global inference parameters.

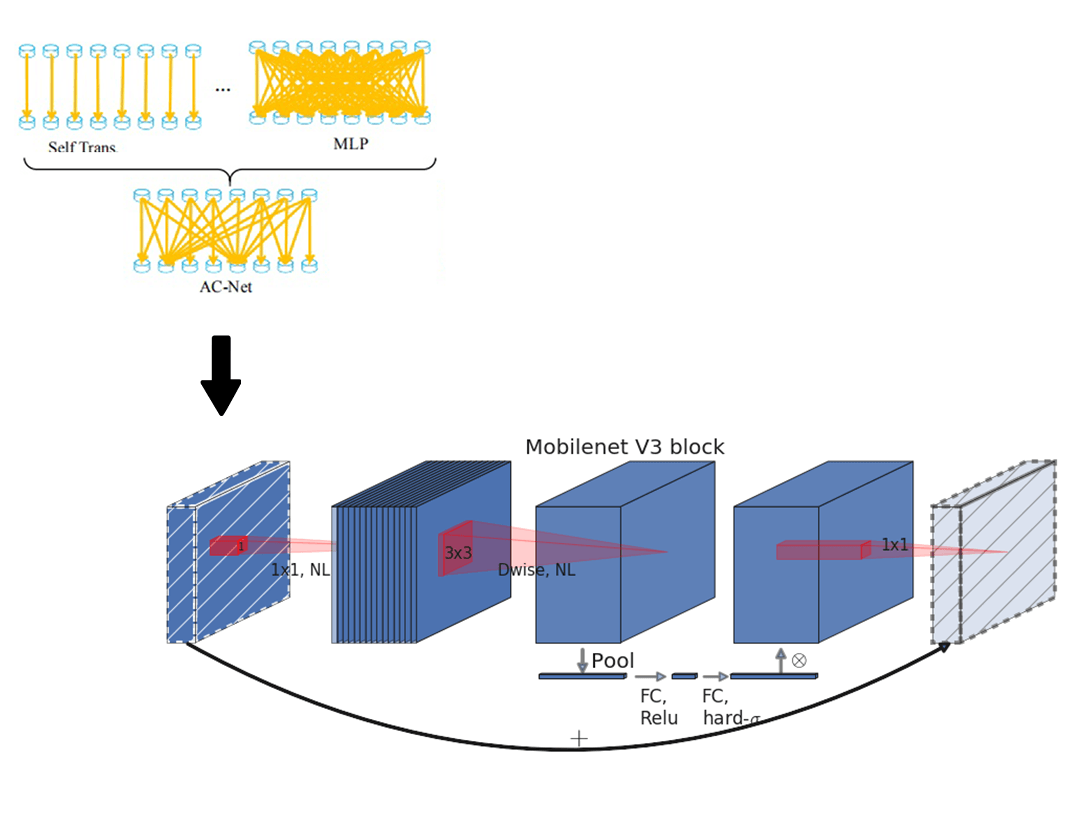


Figure2: the adaptive inverted residual module (AIR)

Mobilenetv3 uses the inverted residual with linear bottleneck module and the structure of squeeze and excitation(SE). We propose an adaptive inverted residual module (AIR) with the same structure. In Figure 2 the AIR is defined by a 1\*1 Expansion layer followed by depth-wise convolutions and a 1\*1 projection layer. MnasNet[11] built upon the MobileNetV2 structure by introducing lightweight attention modules based on squeeze and excitation into the bottleneck structure. The Expansion layer use its change deformation and global optimization.



Figure3: adaptive inverted residual module (AIR)

The author designed the model only to optimize the Expansion layer in AIR. The reason is that the author believes that the input dimension changed after into the Expansion layer, leading to some valuable information is filtered. Using the formula(2) will add a global inference to a two-dimensional image in the Expansion layer to efficiently pass more input information to the Depthwise layer. The author did not optimize the Projection layer because the Projection layer's effect is only a restoration of the dimension to the same with input. Even with any variation of AcnNet Formula 1, the original input layer can not be recovered from the Depthwise layer extension. This is the reason the AIR module is only optimized for the Expansion layer.

**ACNet-based GhostNet**

Huawei team has recently proposed the GhostNet, an innovative lightweight model, in which it defines the term Ghost module. The module consists primarily of a cheap operation and a primary operation. The module consists primarily of the Cheap Operation and the Primary Operation. Reducing the number of redundant feature graphs by Cheap Operation. An analysis of GhostNet’s paper here shows that the Linear transformations in question are equivalent to cheap operations. Furthermore, the paper also mentions: the linear operations Φ operate on each channel whose computational cost is much less than the ordinary convolution. In practice, there could be several different linear operations in a Ghost module, e.g. 3 \* 3 and 5 \* 5 linear kernels, which will be analyzed in the experiment part.

Figure 4 is the Ghost module structure, which is divided into two operations to obtain the same number of characteristic graphs as standard convolution, namely Primary Operation and Cheap Operation. The paper mention that the whole Ghost module here needs to be stressed the same number of feature maps. Firstly, the input enters Primary Operation, which using a small amount of convolution. For example, suppose the regular operation uses 32 convolution cores. In that case, the Primary Operation uses 16 convolution cores here, which cuts the computation in half. The previous output is then continued with the Cheap Operation for the second step. The Cheap Operation uses Depth-wise convolution φ such as 3 \* 3. It is the key that performs the GhostNet model more effective than MobileNet. At last, the Primary Operation and Cheap Operation are spliced together to obtain an output with the same number of channels as the input.

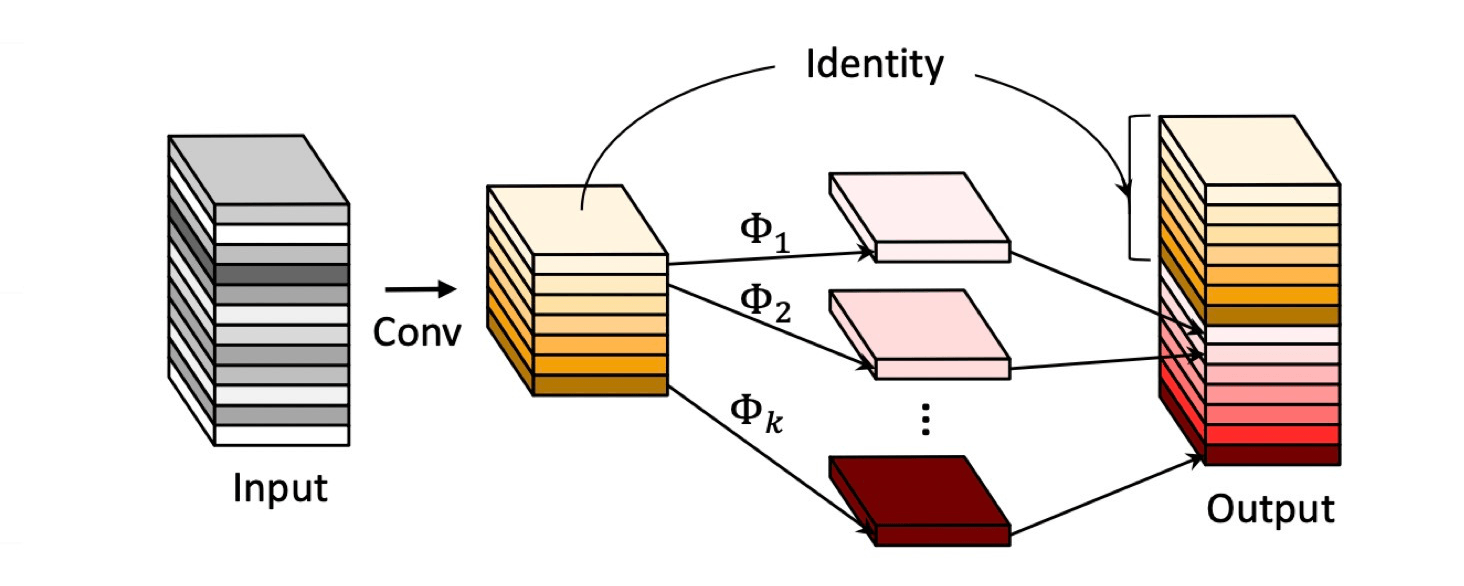


Figure4: Ghost module

We believe that the combination of global and local inference in Primary Operations to build an adaptive Ghost module. It not only can reduce the number of convolution but also improve the model's overall inference ability and further improve the model's accuracy. The reason is that when the input enters Primary Operation, it can carry both global and local inference information. It allows more essential information to pass through the Depth-wise convolutional of Cheap Operations. It makes the model has the resulting referential capability more plausible. The authors used the structure of Figure 4 on Operation Primary to introduce three factors: global inference, local inference, and self-transformation, which allowed the Ghost feature maps into the subsequent Cheap Operation to carry more helpful information and achieve the goal of improving accuracy.

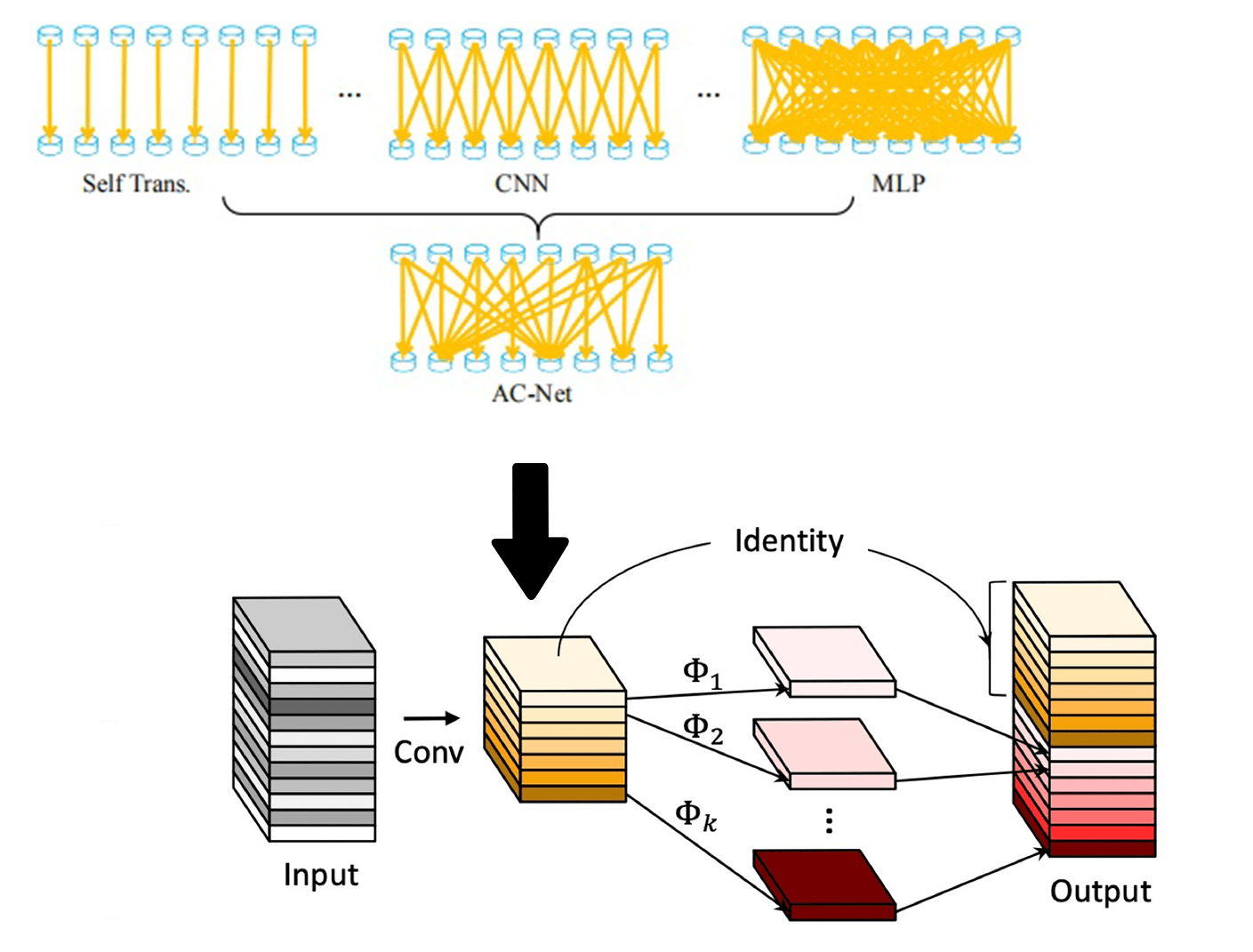


Figure5: the Adaptive Ghost module

The authors did not optimize the cheap operation because the input of the Cheap Operation is the output of the Primary Operation. At this point, the Cheap Operation input result already has the factors of global and local inference. The Cheap Operation does a regular Depth-wise convolutional convolution and then combines with the feature maps after Primary Operations. Therefore, in the Adaptive Ghost Module, the AcNet Formula(1) only optimizes the part of primary operation. The results of the controlled experiment in the following paper can confirm the author’s point of view.

# Experiments

**Dataset**

This article uses Cifar-100[12] dataset. The dataset has 100 classes, and each class includes 600 images. Among the 60000 images, there are 50000 training images and 10000 testing images. The reason for using cifar100 as the data set this time is to verify that ACNet-based MobileNet has a certain global inference ability through more detailed classification. According to a convention, two error rates will be provided: top-1 and top-5. The top-5 error rate means that the testing image's correct label is not among the five most likely notes considered by the model. In the data preprocessing stage, the data pictures are uniformly cropped to a fixed size of 224\*224 at the centre point. We will mirror and flip the image randomly to achieve the purpose of expanding the dataset.

**ACNet-based MobileNet**

***Training***

We use multiple controlled trials to test ACNet-based MobileNet. The first is to verify the validity of the ACNet-based MobileNet theory. Use the MobileNetV3-Small structure mentioned in the mobilenetv3 paper for training on cifar100. Then use ACNet-based MobileNet (Table1)with the same structure and params to classify and predict cifar100. Use a single factor to verify the functionality of ACNet-based MobileNet.

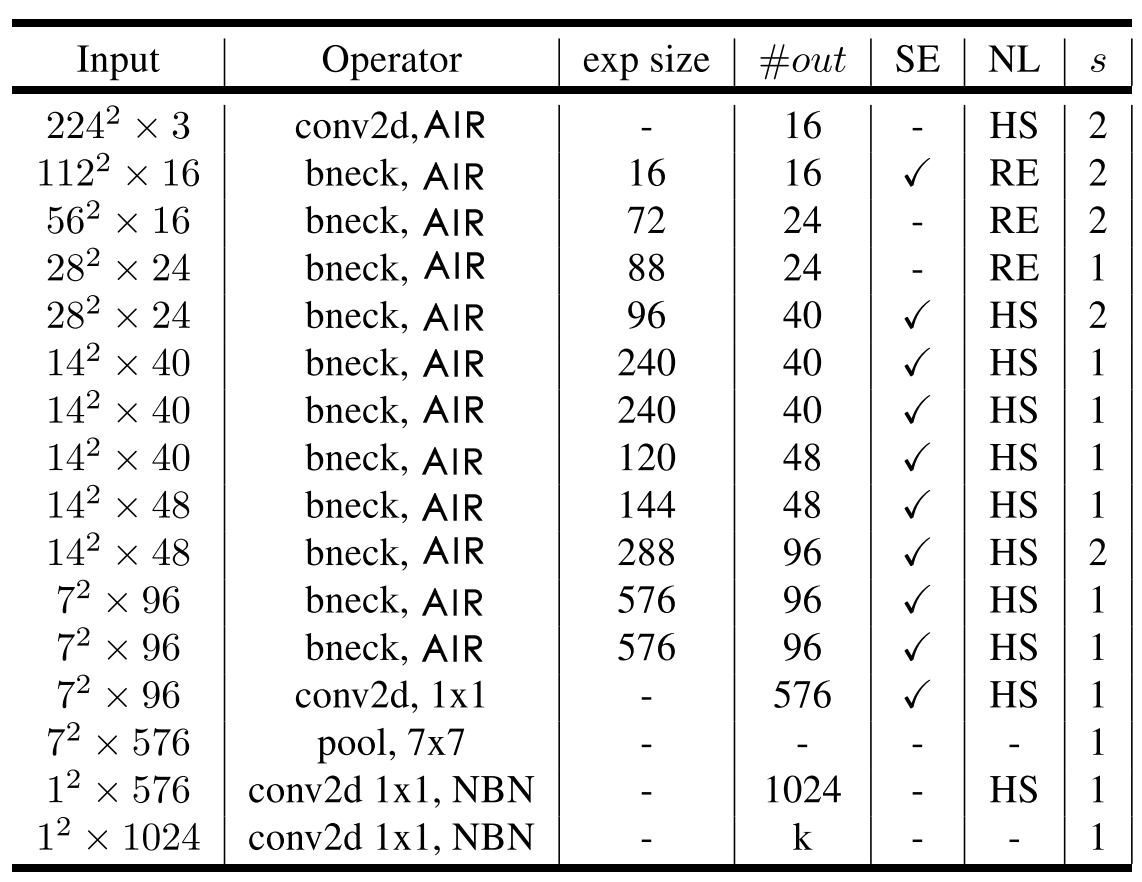


Table1: Using the same architecture as mobilenetv3, which uses AIR instead of bneck.

**﻿**

***Experiment results***

The table2 compares the accuracy of mobilennetv3 with that of ACNet-based MobileNet after the iteration of 100 epochs. The Author trained the models on a single RTX 2080 GPU with 10GB of memory. The maximum batch size of 128 is used, and the classification accuracy of Top-1 and Top-5 models was recorded. CNet-based MobileNet achieves the best top-1 and top-5 test set accuracy rates of 67.67% and 90.17%, respectively. The accuracy is 0.92% higher than what MobileNetV3(SMALL) had for Top-1. It is due to the use of the AIR module to optimize the MobileNetV3. The model is adaptive with the global reasoning ability, which can adapt to more complex image separation problems and improve accuracy. It is in line with our above speculation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Top-1 | Top-5 | Epoch | Training Time |
| MobileNetV3(SMALL) | 66.75 | 89.79 | 100 | 2h35min |
| ACNet-based MobileNet | **67.67** | **90.17** | 100 | 3h10min |

Table 2: ACNet-based MobileNet results

The AIR module has some side effects as well as improved accuracy. The training time of the whole model is prolonged because more parameters are introduced for global inference. The training time of ACNet-based MobileNet is longer than MobileNetV3 35 minutes running on a single GPU. However, the time complexity was still within an acceptable range. We can conclude from the Top-1 accuracy in figure 5 that ACNet-based MobileNet has a low accuracy rate in the beginning. The reason is that the large amount of complex redundant information to be processed in the global inference. However, with the increase of epochs, the models of MobileNetV3 and ACNet-based MobileNet converged at 31 and 32 iterations, respectively. The Top-5 accuracy of Figure 5 shows the trends of accuracy almost identical in ACNet-based MobileNet and MobileNetV3. The experiment shows that the AIR module introduces adaptive global inference, which leads the model parameters to increase, which leads to the long training time and the lag of convergence. However, the accuracy of the model and the training time depends on the setting of hyperparameters. In future test work, we will search for the super-parameters to get a better model.

Figure6: The Top-1 and Top-5 accuracy of ACNet-based MobileNet vs MobileNetV3

The authors also experimented with other variants of the formula(2). It includes using local inference and global inference, and self-transition and global inference. Moreover, It is clear from the data in table 3, variations of the AIR can still play a positive role in the precise classification. One of the most significant improvements the AIR was mixed with an adaptive global inference in the Expansion layer. The authors concluded that the Expansion layer’s 1 \* 1 operation filters a lot of important information. Furthermore, the disadvantage of the inverted residuals module can be effectively mitigated by adding an adaptive global inference.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Top-1 | Top-5 | **Epoch** |
| MobileNetV3(SMALL) | 66.75 | 89.79 | 100 |
| ACNet-based MobileNet(self+local) | 67.27 | 89.45 | 100 |
| **ACNet-based MobileNet(self+global)** | **67.67** | **90.17** | 100 |
| ACNet-based MobileNet(local+global) | 66.91 | 88.83 | 100 |
| ACNet-based MobileNet(local+global+self) | 67.63 | 88.57 | 100 |

Table 3: variants of the formula(2) AIR working in Expansion layer

**ACNet-based ghostnet**

***Training***

The ACNet-based GhostNet used the same experimental environment as above. We were using Cifar-100 as the dataset. Furthermore, the data is preprocessed in the same as before. The training model uses the maximum 128batch size and iterates 100 times. As shown in table 4, the GhostNet network structure is similar to the MobileNet, and uses the SE structure. The following table, where # EXP represents the number of first G-Module output characteristic graphs of G-bneck. Using the Adaptive Ghost Module to optimize the primary operation of G-bneck, the ACNet-based GhostNet model is obtained.

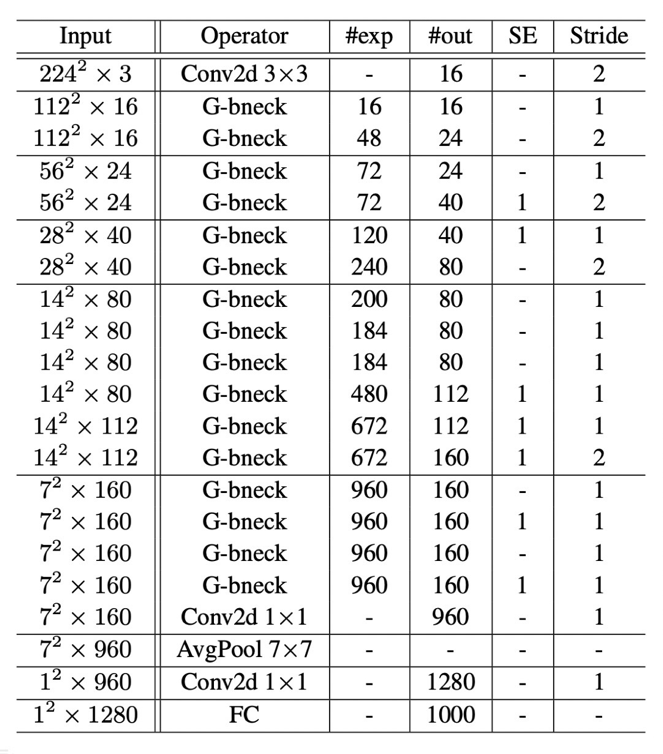


Table4: Using the same architecture as GhostNet, which uses Adaptive Ghost module instead of G-bneck

***Experiment results***

The table5 compares the accuracy of GhostNet with that of ACNet-based MobileNet after the iteration of 100 epochs. ACNet-based ghostnet achieves the best top-1 and top-5 test set accuracy rates of 69.18% and 90.89%, respectively. There is 0.69% more than GhostNet’s Top-1 accuracy rate. Although the ACNet-based GhostNet performance of classification was not perfect, the slight improvement in accuracy was consistent with the author’s previous predictions. From the results of the experiment, the adaptive ghost module is used to optimize the G-bneck, which makes the model adaptive with a certain global inference ability to adapt to more complex image separation problems and improve accuracy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Top-1 | Top-5 | Epoch | Training Time |
| GhostNet | 68.49 | 90.83 | 100 | 4h0min |
| ACNet-based GhostNet | **69.18** | **90.89** | 100 | 5h20min |

Table 5: ACNet-based GhostNet results

As can be seen from Figure 6 below, the Adaptive Ghost Module has a limited effect on ACNet-based GhostNet classification accuracy. ACNet-based GhostNet and GhostNet converged after 39 and 36 iterations, respectively. The adaptive Ghost module brings limited optimization effect to The model and considerable time complexity. The reason is that ACNet-based GhostNet introduces three factors using Formula 1: self-transition, local inference, and global inference. To compare with the AIR of ACNet-based MobileNet, The Adaptive Ghost Module have more computation and parameters. It is also why ACNet-based GhostNet requires more time for training.

Figure6: The Top-1 and Top-5 accuracy of ACNet-based GhostNet vs GhostNet

Table 6 lists different variants of the Adaptive Ghost Module, and the author experiments the models with different combinations of factors. The most accurate classification model is still an adaptive model with three factors. The author hypothesizes that three adaptive factors preserve the redundant information to the greatest extent and make the feature maps after the primary operations have the most favourable information. The author will carry on the further experimental demonstration in this direction in the follow-up work.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Top-1 | Top-5 | **Epoch** |
| GhostNet | 68.49 | 90.83 | 100 |
| acnet\_ghost (self+global) | 67.58 | 89.90 | 100 |
| acnet\_ghost (self+local) | 67.27 | 89.24 | 100 |
| acnet\_ghost (local+global) | ~~67.8400~~ | ~~89.08~~ | ~~100~~ |
| **ghost (local+global+self)** | **69.18** | **90.89** | 100 |

Table 7: variants of the adaptive Ghost module working in primary operation

# conclusions

By using global inference and local inference in ACNet, image classification accuracy based on lightweight model can be improved effectively. From The experimental results, it can be concluded that AIR module is used in the ACNet-based MobileNet model to improve The global inference ability and retain more useful information in the inverted residual module. It makes the accuracy of image classification of the model have a slight improvement. The ACNet-based GhostNet experiment showed little improvement in image classification accuracy. The reason may be the hyperparameter learning rate is set too small, and follow-up work will conduct search experiments on the learning rate to obtain a better performing model. In the next work, other different data sets will be used to verify the model to ensure the rigor of the conclusion.

The experimental results show that MLP and CNN's adaptive adjustment can determine to some extent the information loss caused by the dimension change of the inverse residuals module.

~~The results of the experiment did not meet expectations. However, we still got a conclusion that the method of adaptively adjusting MLP and CNN cannot solve the information loss caused by the dimensional changes in the inverted residual module.~~ ~~The author speculates that the reason is the inverted residual module is reduced to the loss of two-dimensional information, which cannot be compensated by simple global inference.~~ ~~In the follow-up work, a lot of experiments and explorations will be carried out to find an innovative inverted residual module that retains most of the information as much as possible.~~

In this way, image classification technology can be better applied to the mobile terminal, and deep learning related technology can be better applied to life.

# References

[1] A. Howard *et al.*, “Searching for mobileNetV3,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 1314–1324, 2019, doi: 10.1109/ICCV.2019.00140.

[2] G. Wang, K. Wang, and L. Lin, “Adaptively connected neural networks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 1781–1790, 2019, doi: 10.1109/CVPR.2019.00188.

[3] A. G. Howard and W. Wang, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” 2012.

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, 2018, doi: 10.1109/CVPR.2018.00474.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[6] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pp. 1–14, 2015.

[7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, 1998, doi: 10.1109/5.726791.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017, doi: 10.1145/3065386.

[9] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015, vol. 07-12-June-2015, doi: 10.1109/CVPR.2015.7298594.

[10] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[11] M. Tan *et al.*, “Mnasnet: Platform-aware neural architecture search for mobile,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 2815–2823, 2019, doi: 10.1109/CVPR.2019.00293.

[12] M. B. McCrary, “Urban multicultural trauma patients.,” *ASHA*, vol. 34, no. 4, 1992.

[13] G. Wang, P. Luo, and X. Wang, “Batch Kalman Normalization: Towards Training Deep Neural Networks with Micro-Batches.”

[14] G. Wang, J. Peng, P. Luo, X. Wang, and L. Lin, “Kalman normalization: Normalizing internal representations across network layers,” *Adv. Neural Inf. Process. Syst.*, vol. 2018-December, no. NeurIPS, pp. 21–31, 2018.

[15] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, “Bag of tricks for image classification with convolutional neural networks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 558–567, 2019, doi: 10.1109/CVPR.2019.00065.