**~~ACNet-based MobileNet for image classification~~**

# Abstract

~~In this paper, we propose a novel ACNet-based~~ ~~MobileNet(Adaptively Connected Neural Networks based MobileNet) for image classification. Google's MobileNet[1] gets a significant achievement in image classification on the mobile device platform in recent years.~~ ~~However, MobileNet has fewer model parameters, making its accuracy still not comparable to other large-scale network models. Previously,~~ ~~ACNet[2] proposed to improve the traditional convolutional neural networks (CNNs), can flexibly change the global and local reasoning in the internal feature performance, and it also enhances classification accuracy. We believe that ACNet can adequately compensate for the above-mentioned MobileNet problems. Therefore, our ACNet – based MobileNet has benefited is that while retaining the inverted residual architecture of the MobileNet model, the model parameters are small enough. It also could improve the accuracy of image classification slightly. The code is available at~~ *~~https://github.com/TOMMYWHY/acnet\_mobilenetv3~~*

# Introduction

1. 图片分类技术日趋成熟，可应用领域广发。很多模型对图片分类可以达到很高的准去率。这归功于传统卷积广泛应用。

经典轻量级模型也为图像分类在实际生活中的一广泛应用。

然而, 经典网络参数过大，进停留于试验阶段。很难应用到实际生活中。经典轻量级模型可以有效的嵌入移动设备，但准确率不高。

如何提高轻量级模型准确率，使得深度学习更好的应用到实际生活生产中，十分关键。

1. Cnn 与 mlp结合出很多优秀模型，但他们具有局限性。 Acnet 可以 结合cnn 与mlp。确保信息的完整性。

# 本文中经过优化的基于ACNet的思想架构对 现有经典移动端模型MobileNet 和 ghostNet 进行改进与优化，从而获得更高的准确率。

# related work and Literature review

Cnn

Alex vgg googlenet

Resnet

Mobilenet 123

Shullf net

Ghostnet

Acnet

介绍 mobilenet 结构

介绍 ghostnet结构

介绍acnet

# 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, author proposed formula 2 which contains two parts of global reference 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.

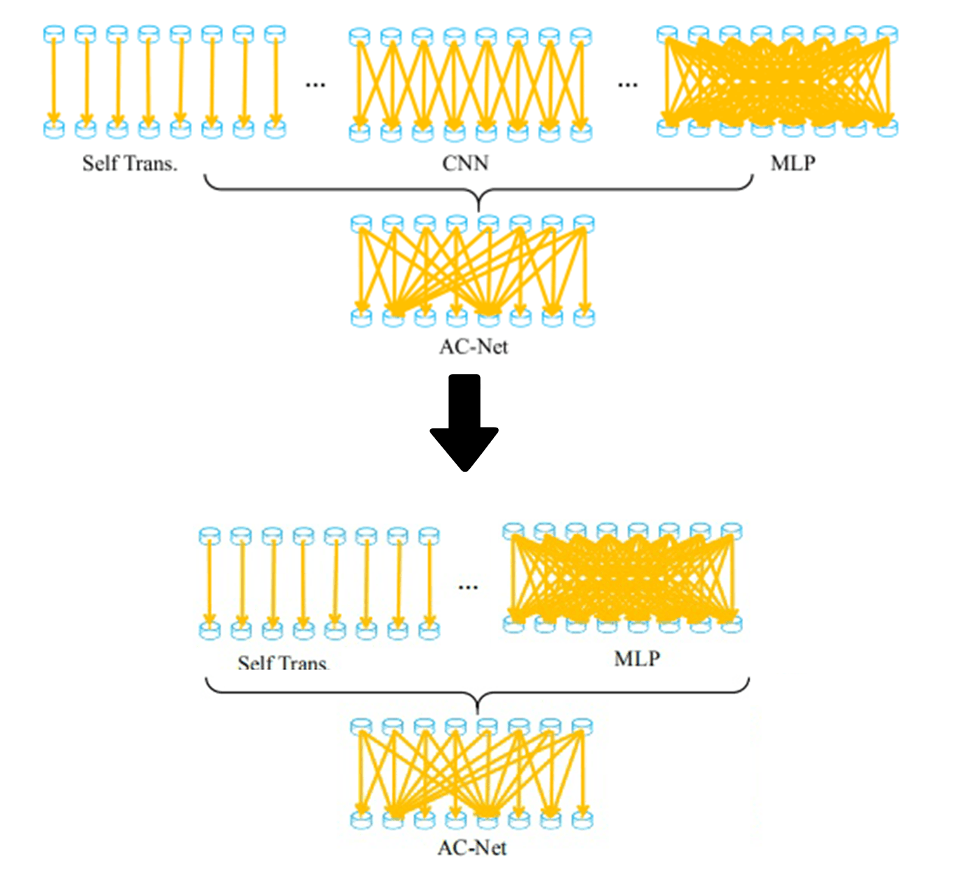


Figure 1: formula 2

**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 the 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 inverted residual module have certain global reference ability. 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 image's redundancy information to the greatest extent by global reference parameters.

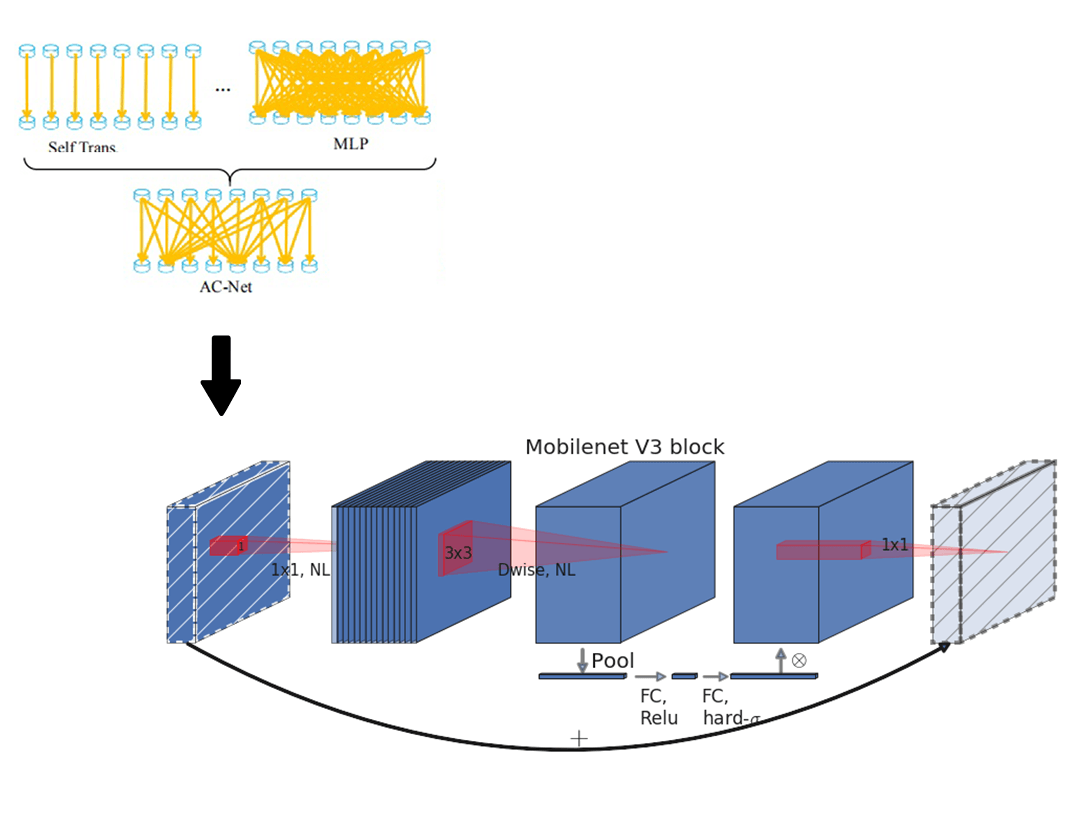


Figure1: the adaptive inverted residual module (AIR)

Mobilenetv3 uses the inverted residual with linear bottleneck module and the structure of squeeze and excitation. We propose an adaptive inverted residual module (AIR) with the same structure. In Figure 2 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.



Figure2: 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 redundant information is filtered. However, redundant information still represents certain information. Using the formula (2) will add a global reference 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**

Primary 上有更好的结果。

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 a cheap operation and a primary operation. An analysis of Ghostnet’s paper here shows that the Linear transformations in question are equivalent to cheap operations. And 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 3 is the Ghost module structure, which is divided into two operations to obtain the same number of characteristic graphs as ordinary 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.

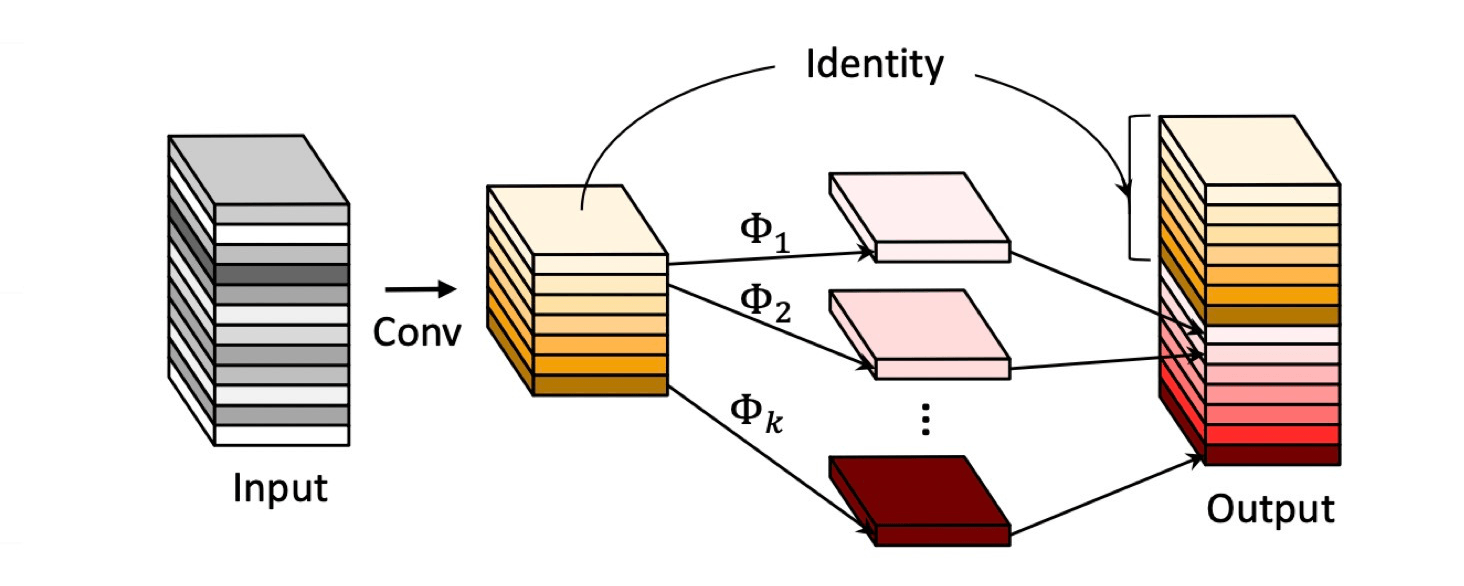


Figure3: Ghost module

The authors believe that the combination of global and local reference 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 reference 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 reference 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 reference, local reference, 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.

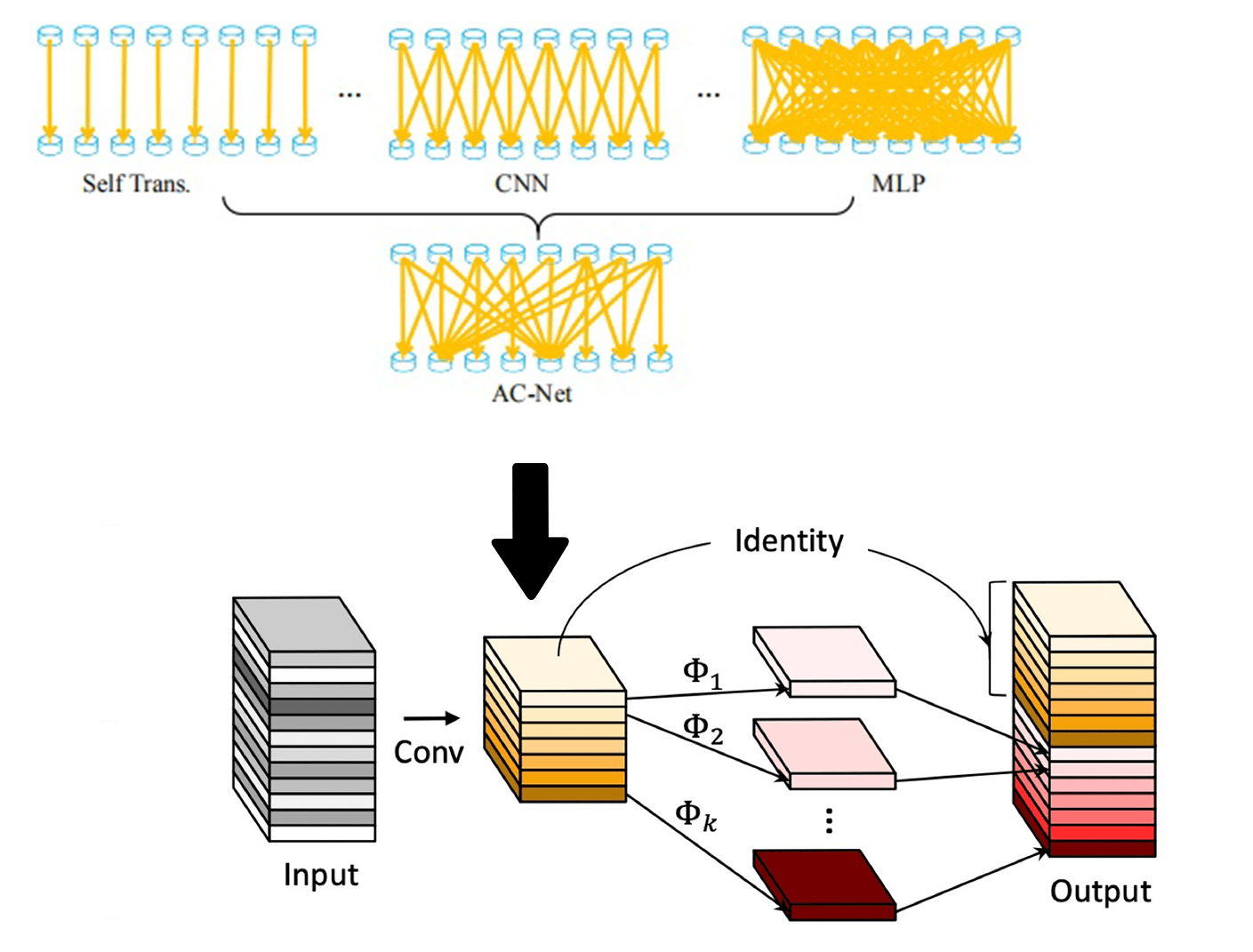


Figure4: the Adaptive Ghost module

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 reference. 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, 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 center point. We will mirror and flip the image randomly to achieve the purpose of expanding the dataset. ~~All the implementation details and experiment settings are the same as [13][14].~~

***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. ~~At the same time, we used the techniques in Tong's[15] paper to optimize the model parameters and tried to get the model with the highest accuracy.~~

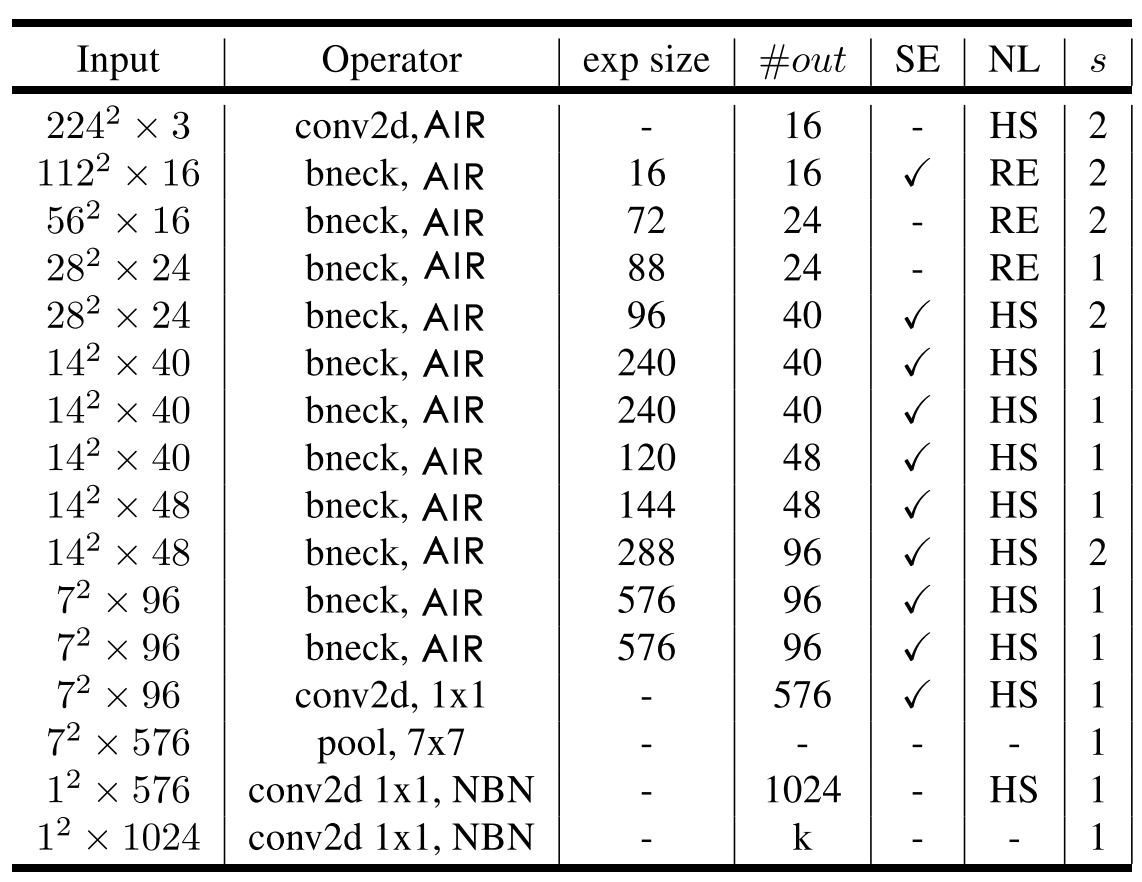


Table1: It uses the same architecture as mobilenetv3, which uses AIR instead of 3X3 and 5X5 convolution operations.

**﻿**

**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 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 reference. 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 reference. 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 reference, 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.

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 reference and global reference, and self-transition and global reference. 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 AIR was mixed with an adaptive global reference in the Expansion layer. The authors concluded that the Expansion layer’s 1 \* 1 operation filters a lot of important information. And the disadvantage of the inverted residuals module can be effectively mitigated by adding an adaptive global reference.

Projection 实验结果与分析。。。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 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**

ACNet-based ghostnet的实验中，使用了与MobileNetV3-Small相同的实验环境。~~之所以使用这样的结构是为了能能够严谨的与上述实验进行对比。~~使用同样的Cifar-100 数据集，并且对数据进行相同的预处理。训练模型使用最大的128 batch size，并迭代100次。从Table4可以看出，Ghostnet 的网络结构与与上述实验中的mobilenet相似。 由于ghostNet也采用了 与上述实验不同的地方是采用the Adaptive Ghost module。 ***ACNet-based ghostnet*** achieves the best top-1 and top-5 test set accuracy rates of 67.67% and 90.17%, respectively.

The adaptive Ghost module

**Experiment results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Top-1 | Top-5 | Epoch | Training Time |
| GhostNet | 68.62 | 89.42 | 100 | 2h35min |
| ACNet-based GhostNet | **67.67** | **90.17** | 100 | 3h10min |

# conclusions

The ACNet-based MobileNet model uses the AIR module, which in theory can effectively improve the global reasoning ability and retain more information in the inverted residual module. However, it has not been significantly confirmed in this experiment. Try to use the ACNet-based MobileNet model to distinguish objects with the same shape characteristics effectively, and then get a more accurate model. From the experimental results, the effect is not significant. 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 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.