

实训报告

课 程 名 称： 暑期实习

实验项目名称： 实训报告

专 业 班 级： 计科1803

姓 名： 周诗沛

学 号： 201808010328

信息科学与工程学院

目录

[一、试验记录 3](#_Toc79146716)

[Shufflenet 3](#_Toc79146717)

[Shufflenet v1 3](#_Toc79146718)

[ShuffleNet V2 7](#_Toc79146719)

[源码解读 9](#_Toc79146720)

[二、实验步骤 11](#_Toc79146721)

[2.1 眼疾识别数据集iChallenge-PM 11](#_Toc79146722)

[2.2 数据集准备 11](#_Toc79146723)

[2.3 查看数据集图片 12](#_Toc79146724)

[2.4 训练过程 14](#_Toc79146725)

[2.5 网络搭建 16](#_Toc79146726)

[三、运行结果 17](#_Toc79146727)

[四、总结 17](#_Toc79146728)

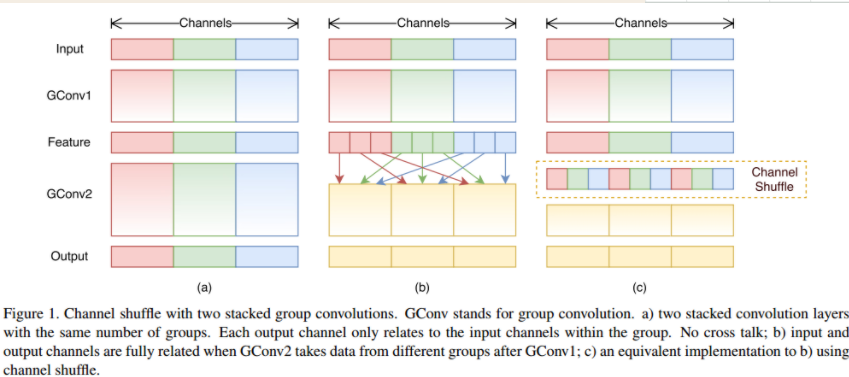
# 一、试验记录

## Shufflenet

### Shufflenet v1

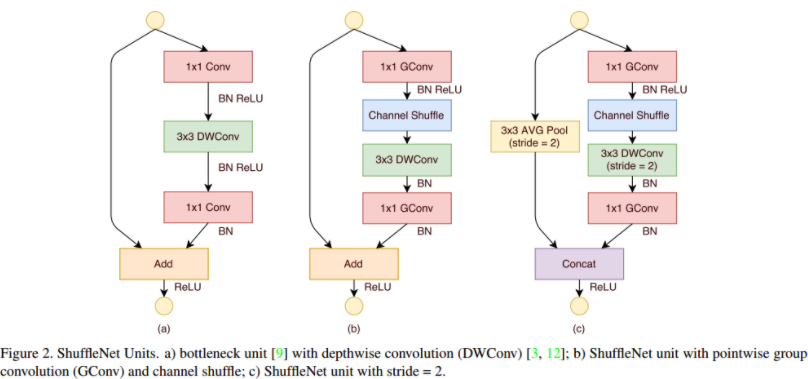
该网络提出于2017年，论文为《ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices》。

由于Xception和ResNeXt中存在大量密集的1\*1卷积，导致网络十分低效。因此，旷世提出了pointwise group convolutions来减少1\*1卷积的计算复杂度。但是，这样会带来副作用：通道间的信息没有得到很好的交流融合。所以，继而提出channel shuffle来帮助信息在不同特征通道中的流动。可以看出，ShuffleNet V1的主要两个创新点是pointwise group convolutions和channel shuffle。



如Fig.1所示，左图是普通的group卷积，卷积核只有相应的输入通道进行作用，不同通道之间信息没有交流。如果组数等于通道数，就可以理解成mobilenet中的depthwise convolution。(b)图是将通道进行重新打乱排列，使得不同通道信息之间能相互交流。(c)图是(b)图的等价，使用了channle shuffle进行通道重新排列。如果**group卷积能够获得不同group中的输入数据，输入特征和输出特征就能很好的关联起来。**通过channle shuffle操作，能构建更加强有力的网络结构。

为了使用好channel shuffle的操作，作者提出了ShuffleNet网络。首先，我们先看一下其基本单元ShuffleNet Unit。如Fig.2所示，(a)图是ResNeXT的残差模块，经过1\*1的卷积来降低通道数，然后使用3\*3的group卷积，最后使用1\*1的卷积将通道数提升回原来。(b)图是作者提出的ShuffleNet Unit，与残差模块类似，将残差模块中的1\*1卷积换成1\*1的group卷积，并加入了channel shuffle操作。值得注意的是，3\*3的group卷积后，没有接激活函数。(c)图是步长为2的情况，基本类似，但最后是concat操作，而不是add，这样做的目的是在很小的计算成本下，更容易扩大通道数。



接下来分析一下，ShuffleNet的FLOPs的变化。假设输出尺寸为c∗h∗w

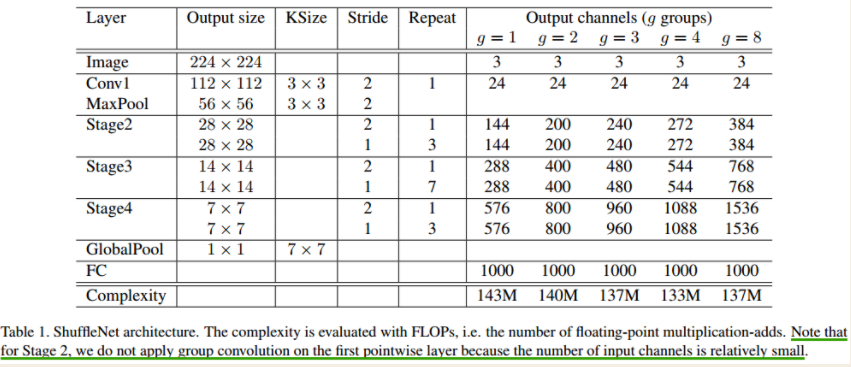
ResNet的FLOPs为hwcm+9hwm2+hwmc=9hwm2+2hwcm

ResNeXt的FLOPs为hwcm+9hwm2/g2+hwcm=9hwm2/g+2hwcm

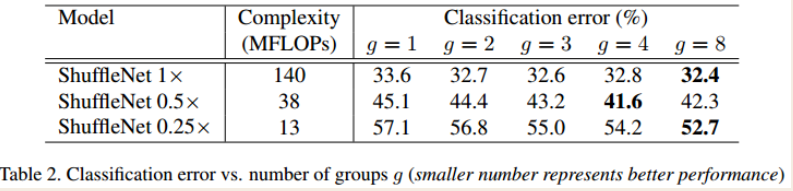
ShuffleNet的FLOPs为9hwm+2hwcm/g

可以看出，ShuffleNet相对的FLOPs较小。换而言之，在相同的计算资源限制下，ShuffleNet能使用更宽的特征图。这个对轻量化网络来说，是非常重要的。因为轻量化网络通常由于特征图宽度不足而无法更好处理信息。

基于提出的ShuffleNet Unit，可以构建出Shuffle网络， 如Table.1所示。在整体计算资源变化不大下（~140MFLOPs），当组数g扩大的时候，可以采用更加宽的特征图，来更好的编码信息。



为了评估pointwise group convolution的重要性，作者在上面的网络中使用了不同的g值（通道数会变化的情况），结果如Table.1所示。利用看出，当g>1时，性能会比g=1的情况好（g=1的网络类似于Xception）。所以，小网络会因为分组而得到准确率上的提升。作者认为这是因为更大的特征图宽度对特征的编码起到更多作用。但也值得注意的是，g值越大并不一定越好，网络可能会饱和或者准确率出现恶化。



最后作者在ARM平台上测试了ShuffleNet的真实速度，发现，尽管g=4或者g=8有更高的准确率，但效率却不是最高的。最终采用g=3来折中准确率和最终运行速度。作者还尝试了加入SE模块，但速度大大变慢了。

### ShuffleNet V2

该网络提出于2018年，《ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design》。主要的创新点是提出了4条轻量化的原则，并基于此4条原则，在v1基础上提出了ShuffleNet V2。

首先介绍一下FLOPs，这里引用了知乎博主的介绍。

FLOPS: floating point operations per second

FLOPs: floating point operations

FLOPS： 全大写，指每秒浮点运算次数，可以理解为计算的速度。是衡量硬件性能的一个指标。（硬件）

FLOPs： s小写，指浮点运算数，理解为计算量。可以用来衡量算法/模型的复杂度。（模型） 在论文中常用GFLOPs（1 GFLOPs = 10^9 FLOPs）

在衡量计算复杂度时，通常使用的是FLOPs（the number of float-point operations），但FLOPs是一个间接衡量的指标，它是一种近似但不等价于直接衡量指标。在实际中，我们所关心的是速度或者延时，而这些才是直接衡量的指标。因此，使用FLOPs作为唯一衡量计算复杂度的指标，是不充分的，而且容易导致次优的网络设计。

直接衡量指标（速度）和间接衡量指标（FLOPs）之间的差异，主要源于以下两个原因：

1. 还有多个影响速度的指标，并未纳入FLOPs的考虑范围中。例如MAC（memory access cost，内存访问成本），MAC对group卷积的速度影响很大；平行度（degree of parallelism），高平行度的网络的推理速度更快。
2. 不同平台对推理时间的影响。例如，CUDNN对3\*3的卷积进行了优化，不能简单认为其是1\*1卷积的9倍。

所以，接下来作者对网络的衡量指标有两点，i)使用直接衡量指标（速度）；ii)在特定的同一个平台上进行衡量。

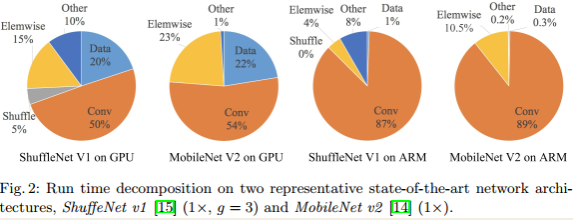
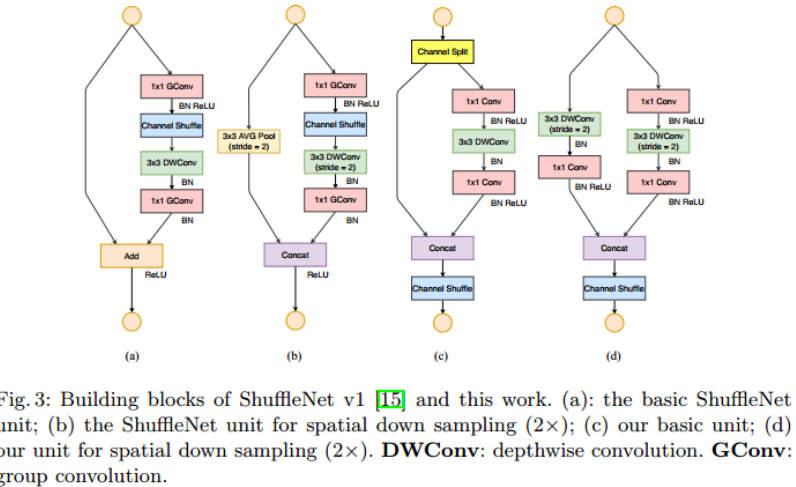


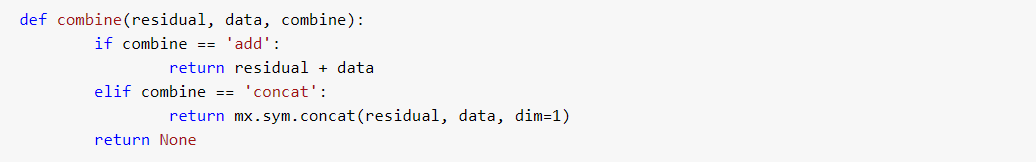
Fig.2是ShuffleNet v1和MobileNet v2的时间占比图。从图中可以看出，FLOPs仅仅是考虑了卷积部分，但其他时间同样不能忽略，例如数据的读取，Element-wise等操作。

基于上述的观察，提出了4条关于轻量化网络设计的原则：

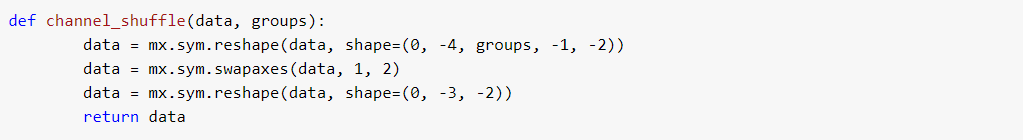
1. 当输入通道数和输出通道数的值接近1:1时，能减少MAC时间；
2. 过多的group卷积，会增加MAC时间；
3. 网络的分裂会降低平行度；
4. Element-wise操作是不可以忽略的，包括ReLU，AddTensor，ADDBias等

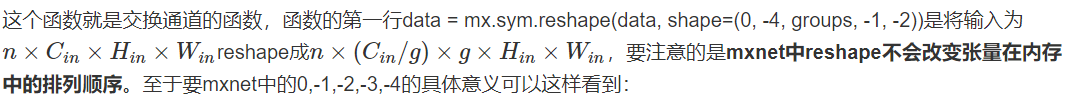


## 源码解读

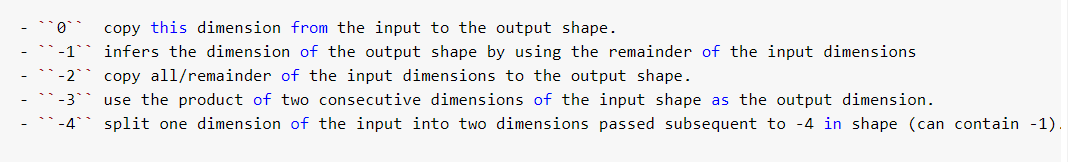


add是代表图6中的单元b），concat是代表图6中的单元c）。



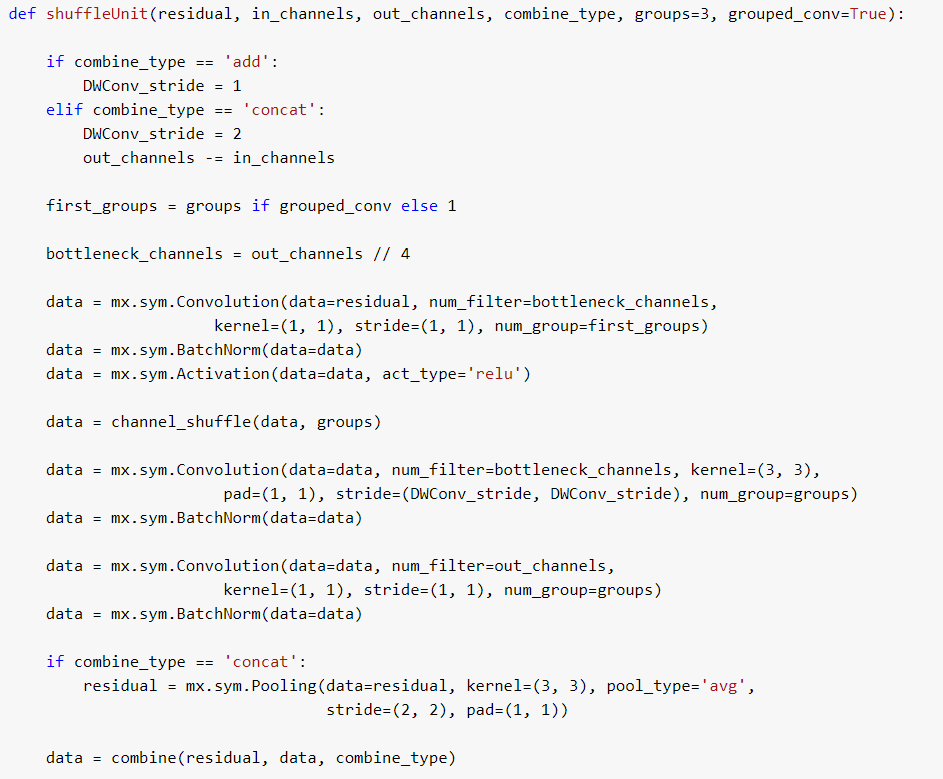


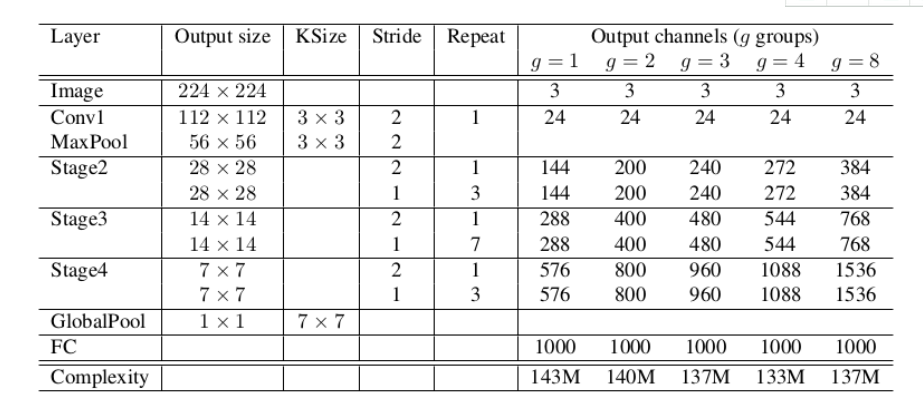
可以看到输出以下（只提取出一小部分，其余的可用上述方法查看），这里有各个参数的具体意义：



函数的第二行是交换第一与第二个维度，那么现在这个symbol的符号的shape就变成了n×g×(Cin/g)×Hin×Win。这里的第零个维度是n。要注意的是交换维度改变了张量在内存中的排列顺序，改变了内存中的顺序实现上就是完成了图5c)中的Channel Shuffle操作，不同的颜色代码数据在原来内存中的位置。

函数的最后一行合并了第一与第二个维度，输出的张量与输入的张量shape都是n×Cin×Hin×Win。





# 二、实验步骤

## 2.1 眼疾识别数据集iChallenge-PM

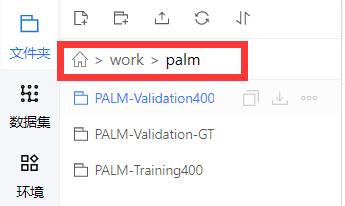
iChallenge-PM是百度大脑和中山大学中山眼科中心联合举办的iChallenge比赛中，提供的关于病理性近视（Pathologic Myopia，PM）的医疗类数据集，包含1200个受试者的眼底视网膜图片，训练、验证和测试数据集各400张。下面我们详细介绍LeNet在iChallenge-PM上的训练过程。

## 2.2 数据集准备

home/aistudio/data/data19065 目录包括如下三个文件，解压缩后存放在/home/aistudio/work/palm目录下。

* training.zip：包含训练中的图片和标签
* validation.zip：包含验证集的图片
* valid\_gt.zip：包含验证集的标签

注意要将/home/aistudio/work/palm/PALM-Validation-GT/目录下的PM\_Label\_and\_Fovea\_Location.xlsx文件转存成csv格式，本节代码示例中已经提前转成文件labels.csv。



## 2.3 查看数据集图片

iChallenge-PM中既有病理性近视患者的眼底图片，也有非病理性近视患者的图片，命名规则如下：

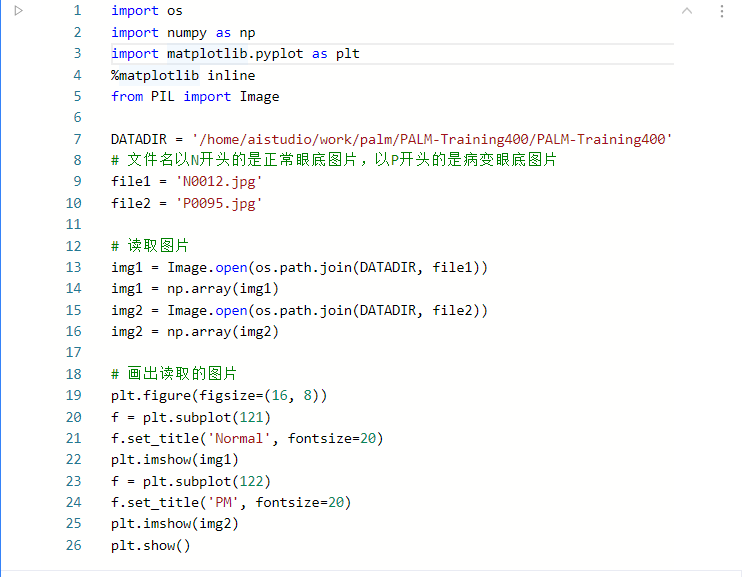
病理性近视（PM）：文件名以P开头

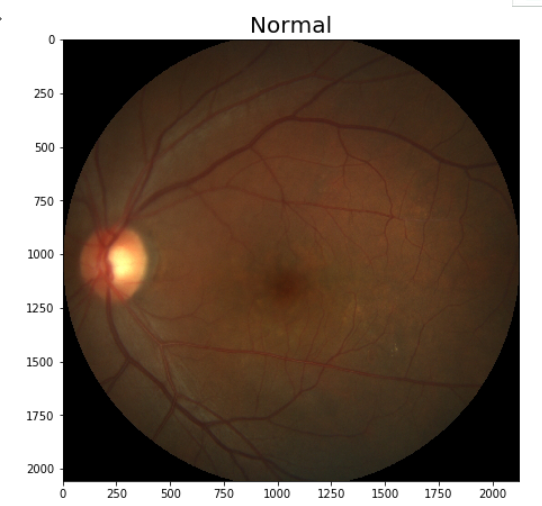
非病理性近视（non-PM）：

高度近视（high myopia）：文件名以H开头

正常眼睛（normal）：文件名以N开头

我们将病理性患者的图片作为正样本，标签为1； 非病理性患者的图片作为负样本，标签为0。从数据集中选取两张图片，通过LeNet提取特征，构建分类器，对正负样本进行分类，并将图片显示出来。代码如下所示：





定义数据读取器



训练集读取时通过文件名来确定样本标签，验证集则通过csvfile来读取每个图片对应的标签

请查看解压后的验证集标签数据，观察csvfile文件里面所包含的内容

csvfile文件所包含的内容格式如下，每一行代表一个样本，

其中第一列是图片id，第二列是文件名，第三列是图片标签，

第四列和第五列是Fovea的坐标，与分类任务无关

ID,imgName,Label,Fovea\_X,Fovea\_Y

1,V0001.jpg,0,1157.74,1019.87

2,V0002.jpg,1,1285.82,1080.47

打开包含验证集标签的csvfile，并读入其中的内容

filelists = open(csvfile).readlines()

## 2.4 训练过程





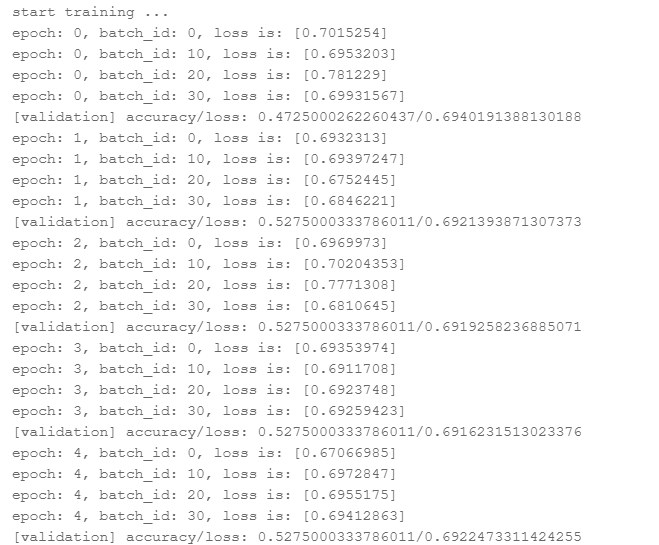


## 2.5 网络搭建





# 三、运行结果



# 四、总结

眼疾数据集是我第一次使用，主要就是数据预处理相对来说比较麻烦一些。对训练数据和验证数据的标签、图像获取方式不一样。Shufflenet实现起来和其他网络相比，也略微复杂，有分组卷积，也部分借鉴了resnet结构。并且这次实训也第一次解除了飞桨平台，十分好用，也有很多学习资源，收获很大。