

实验题目	实验三			实验日期	2022. 5. 27
班级		学号		姓名	郑晟赫、朱永葵、林煜鹏

CS32262 模式识别与深度学习实验

实验报告

一、实验目的（介绍实验目的）

1. 基于 PyTorch 实现 VGG/ResNet/SENet 等结构
2. 选用上述最优模型进行优化器对比、数据增强对比
3. 自选模型优化性能

二、实验环境（介绍实验使用的硬件设备、软件系统、开发工具等）

1. 硬件设备：CPU: i7-9750H; GPU: NVIDIA GeForce GTX 1650; 16G RAM; 512GHD Disk
2. 软件系统：Windows 10; Python 3.7.10; PyTorch 1.9.0
3. 开发工具：PyCharm

三、实验内容（介绍实验过程、设计方案、实现方法、实验结果等）

1. 数据集分析

选用的数据集是 Kaggle 上的植物种子分类问题。数据集分为训练集和测试集，其中训练集分为 12 中不同的种子类别，需要预测测试集中每张图片属于哪一种种子，最终将预测结果提交 kaggle 得到预测成绩。

2. 网络定义与调优

2.1 VGG

VGG 相比于 AlexNet 的一个改进是采用连续的几个 3×3 的卷积核代替 AlexNet 中的较大卷积核 (11×11 , 7×7 , 5×5)。对于给定的感受野（与输出有关的输入图片的局部大小），采用堆积的小卷积核是优于采用大的卷积核，因为多层非线性层可以增加网络深度来保证学习更复杂的模式，而且代价还比较小（参数更少）。简单来说，在 VGG 中较小的卷积核堆叠来代替较大的卷积核，这样在具有相同感受野的情况下就提高了神经网络的效果。VGG 的结果可见图 1。

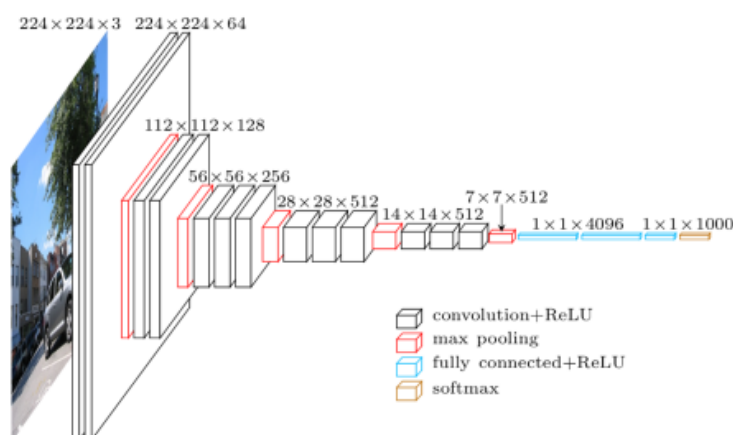


图 1 VGG 结构

2.2 Resnet

ResNet 针对退化现象发明了“快捷连接”，极大的消除了深度过大的神经网络训练困

实验题目	实验三			实验日期	2022. 5. 27
班级		学号		姓名	郑晟赫、朱永葵、林煜鹏

难问题。神经网络的“深度”首次突破了 100 层、最大的神经网络甚至超过了 1000 层。快捷连接的最直接操作就是将卷积、池化后的结果与原始的输入相加之后通过激活函数作为这一个小部分的输出，这样做的目的是有效解决了梯度消失问题，且残差学习相比原始特征直接学习更容易。残差学习单元的结构示意图见图 2。

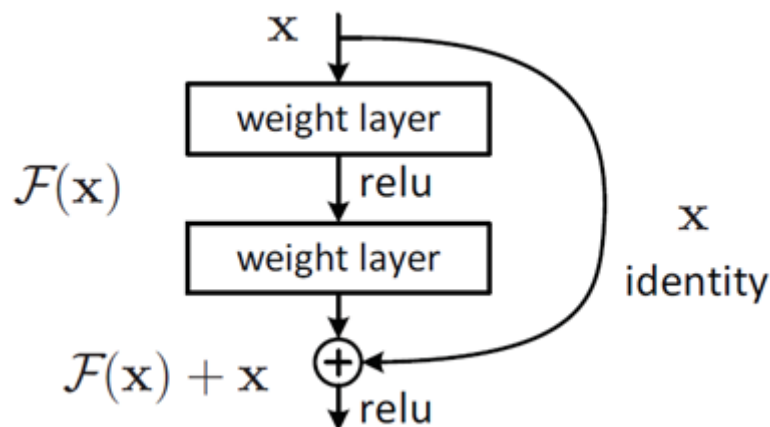


图 2 残差学习单元

从整体结构上来看，ResNet 网络是参考了 VGG19 网络，在其基础上进行了修改，并通过短路机制加入了残差单元，变化主要体现在 ResNet 直接使用 stride=2 的卷积做下采样，并且用 global average pool 层替换了全连接层。ResNet 的一个重要设计原则是：当 feature map 大小降低一半时，feature map 的数量增加一倍，这保持了网络层的复杂度。ResNet 的具体结构可见图 3。

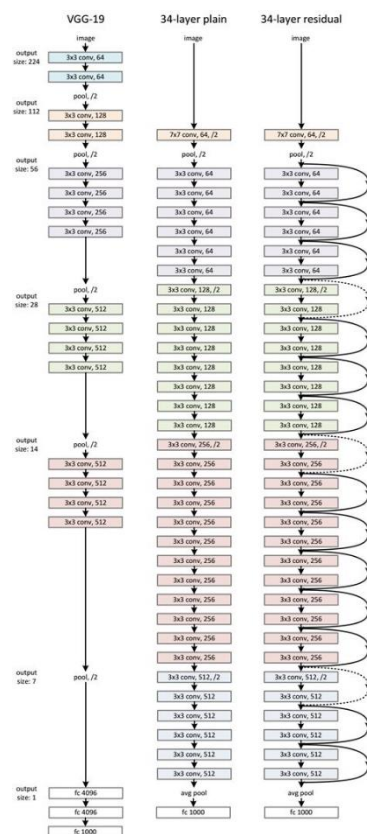


图 3 ResNet 网络结构图

实验题目	实验三			实验日期	2022. 5. 27
班级		学号		姓名	郑晟赫、朱永葵、林煜鹏

2.3 Resnet + SE Block

SE: Squeeze-and-Excitation 的缩写，特征压缩与激发的意思。

我们可以看到，已经有很多工作在空间维度上提升网络的性能。那么就自然的想到，网络是否可以从其他层面来考虑去提升性能，比如：考虑特征通道直接的关系？基于这一点提出了 SENet。在 SENet 中，Squeeze 和 Excitation 是两个非常关键的操作，所以我们以此命名。我们的动机是希望显式地建模特征通道之间的相互依赖关系。另外，我们并不打算引入一个新的空间维度来进行特征通道间的融合，而是采用了一种全新的「特征重标定」策略。具体来说，就是通过学习的方式来自动获取到每个特征通道的重要程度，然后依照这个重要程度去提升有用的特征并抑制对当前任务用处不大的特征

SE Block 的结构：

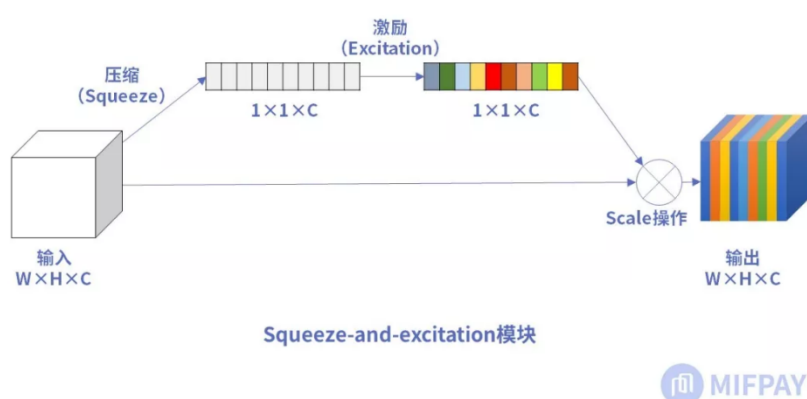


图 4 SENet 结构

下图表示一个 SE 模块。主要包含 Squeeze 和 Excitation 两部分。W, H 表示特征图宽，高。C 表示通道数，输入特征图大小为 $W \times H \times C$ 。

压缩 (Squeeze):

首先是 Squeeze 操作，我们顺着空间维度来进行特征压缩，将每个二维的特征通道变成一个实数，这个实数某种程度上具有全局的感受野，并且输出的维度和输入的特征通道数相匹配。它表征着在特征通道上响应的全局分布，而且使得靠近输入的层也可以获得全局的感受野，这一点在很多任务中都是非常有用的。

激励 (Excitation):

其次是 Excitation 操作，它是一个类似于循环神经网络中门的机制。通过参数 w 来为每个特征通道生成权重，其中参数 w 被学习用来显式地建模特征通道间的相关性。

Scale 操作:

最后是一个 Reweight 的操作，我们将 Excitation 的输出权重看做是经过特征选择后的每个特征通道的重要性，然后通过乘法逐通道加权到先前的特征上，完成在通道维度上的对原始特征的重标定。

将 SE 模块嵌入网络:

SE 模块的使用是很灵活的，可以在已有网络上添加而不打乱网络原有的主体结构。

ResNet 中添加 SE 模块形成 SE-ResNet 网络，SE 模块是在 bottleneck 结构之后加入的。

2.4 性能调优公共部分

优化器选择:

优化器选择主要考虑在 SE-ResNet 上测试 SGD 优化器与 Adam 优化器。这两种优化器中

实验题目	实验三			实验日期	2022. 5. 27
班级		学号		姓名	郑晟赫、朱永葵、林煜鹏

SGD 使用了一阶梯度信息，而 Adam 使用了二阶动量信息，我们看到，SGD-M 在 SGD 基础上增加了一阶动量，AdaGrad 和 AdaDelta 在 SGD 基础上增加了二阶动量。把一阶动量和二阶动量都用起来，就是 Adam 了——Adaptive + Momentum。Adam 在控制优化方向的同时合理控制了优化器。Adam 优化器的优点如下：计算高效，方便实现，内存使用也很少；更新步长和梯度大小无关，只和 α 、 β_1 、 β_2 有关系；并且由它们决定步长的理论上限；对目标函数没有平稳要求，即 loss function 可以随着时间变化能较好的处理噪音样本，并且天然具有退火效果；能较好处理稀疏梯度，即梯度在很多 step 处都是 0 的情况。最后经过实验发现，Adam 优化器的训练速度相较于 SGD 而言更快，且训练的最终的结果也更加稳定，因此最终实验决定选取 Adam 优化器。

数据增强：

数据增强的意义在于我们的训练集的大小未必能够使得神经网络的训练达到较优的效果，或者数据集已经足够大，但是由于数据集的分布与真实的分布之间存在一定的差异，例如我们想使用神经网络对车的品牌做二分类，但是训练集中 A 品牌车均朝右，B 品牌车均超左，神经网络有可能将这作为分类的一个依据，但是这显然不合理，因此可以进行水平翻转，垂直翻转等操作，减少数据集中不相关的特征。

本实验中采用了水平翻转、随机裁剪两种方式，理论上该数据集采用上下翻转，改变色调、饱和度、明亮度均可，但是经试验证明使用后面这几种数据增强之后对于正确率的提升程度有限，但是生成数据时间上升比较明显，因此最终决定只使用水平翻转与随机裁剪两种方式。数据增强后正确率略有提升（最终测试集上大约 0.3%），但增加了训练时间。经过数据增强，在最终测试集上的得分大约提升 1%~2%。

2.5 性能调优自选部分

性能调优部分测试了两种网络——DenseNet 和 Inception Resnet V2。最终 Inception Resnet V2 达到了比 ResNet 更优的效果。以下简要介绍这两种网络。

DenseNet 的基本思路和 ResNet 一致，但是它建立了靠前的层和靠后的层之间的密集连接，同时它还通过特征在通道上的连接实现了特征重用。从 ImageNet 数据集上的表现上来看，DenseNet 在参数和计算成本更少的情形下比 ResNet 的性能更优，在本数据集上相较于 ResNet 效果不佳也许是数据集并不适用于 DenseNet。DenseNet 的结构可见图 5。

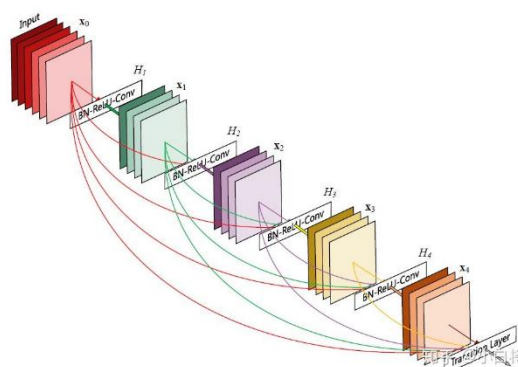


图 5 DenseNet 结构

接下来介绍 Inception Resnet V2。首先简单介绍 Inception 网络，这一网络事实上就是通过是在同一个层级上运行多个尺寸的滤波器，使得网络本质上变得更宽一些，从而不是很深，该架构的主要特点是更好地利用网络内部的计算资源，这通过一个精心制作的设计来实

实验题目	实验三			实验日期	2022. 5. 27
班级		学号		姓名	郑晟赫、朱永葵、林煜鹏

现，该设计允许增加网络的深度和宽度，同时保持计算预算不变。而 Inception Resnet V2 结合了 ResNet 和 Inception 的优势之处，事实上就是在 Inception 模块的卷积运算中添加了残差连接，本质上还是一个 Inception 网络。其简单结构图可见图 6。

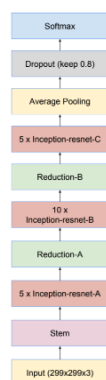


图 6 Inception Resnet V2 结构

使用了这种网络之后使得网络可以又宽又深，从而提高了网络的性能，而 Inception-ResNet v2 和 Inception V4 的大小相当，比 ResNet18 参数量大很多，训练的速度也比较慢，同时预测一张图的时间也要长很多，但是相对于 ResNet18 的预测性能来说也要搞出不少。详细的比较数据可见第 3 部分实验比较部分。

3. 实验比较

算法	测试时间 (ms/image)	Score
VGG11	17.134	0.8072
ResNet	31.131	0.9212
ResNet+SE	33.243	0.9357
DenseNet	效果不佳，无测试	0.9121
Inception Resnet V2	27.161	0.9559

四、实验结论（总结实验发现及结论）

经过本次实验，对于 VGG、ResNet 以及 ResNet 的各种变体有了更加深刻的了解，同时也认识到 ResNet 对于深度网络的意义。