

上机实验二：基于卷积神经网络的CIFAR-10数据集分类

网络设计

选择ResNet[1]，由于本实验仅为一次简单的上机实验，因此我并没有选择resnet原论文在cifar10上使用的一些更小的模型，而是直接选用Resnet18的架构

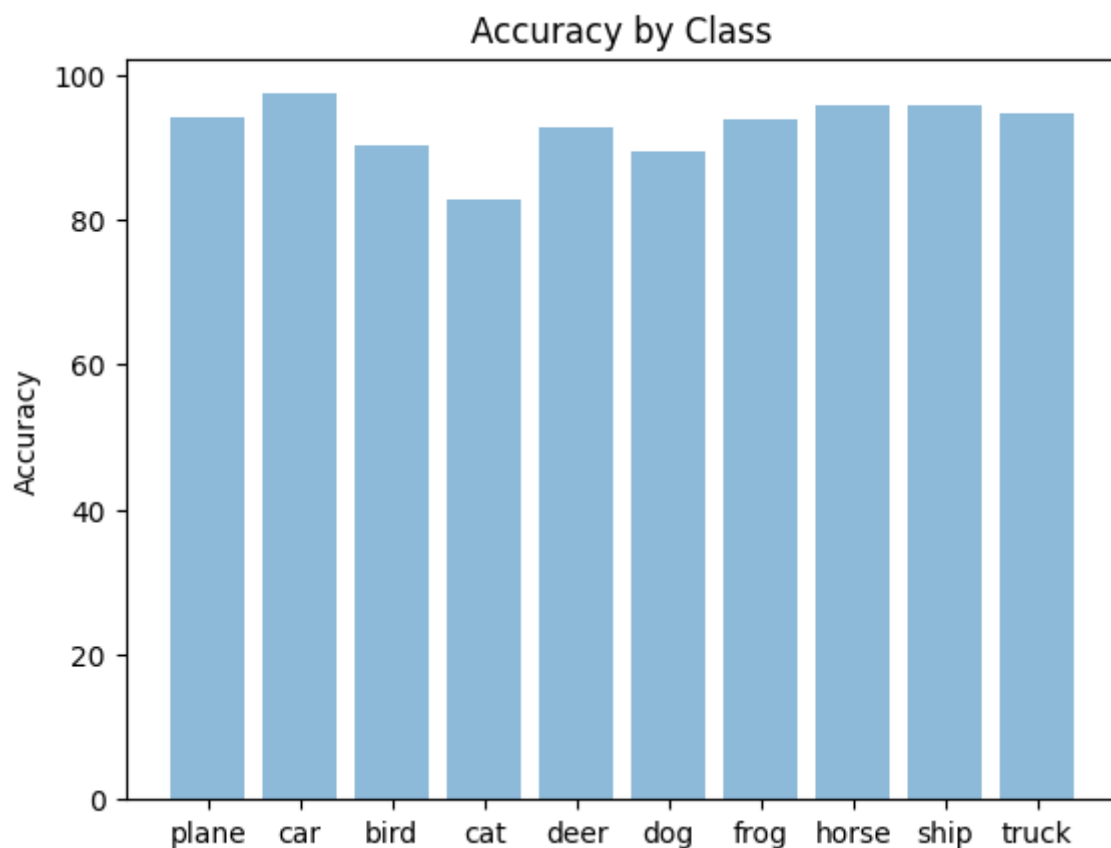
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

训练策略

由于本实验仅为一次简单的上机实验，因此我并没有选择resnet原论文的训练策略，而是选择了自己在其他研究项目常用的设置，并大幅度缩减了训练epochs数（原论文的15%）

- 使用cifar10上普遍被采用的随机翻转与裁剪作为数据增强
- 优化器使用Adam，其超参数为pytorch默认，未使用权重惩罚
- 学习率schedule：在训练整体epochs数的最后10个epochs使用余弦函数曲线使学习率从初始学习率衰减至0
- 初始学习率：0.01
- num of epochs: 30
- 训练batch size: 512

实验结果与分析



```
epoch: 29, loss: 0.0018
-----train acc-----
Accuracy of plane : 99.68 %
Accuracy of car : 99.92 %
Accuracy of bird : 99.54 %
Accuracy of cat : 98.56 %
Accuracy of deer : 99.50 %
Accuracy of dog : 98.96 %
Accuracy of frog : 99.68 %
Accuracy of horse : 99.74 %
Accuracy of ship : 99.98 %
Accuracy of truck : 99.72 %
Overall accuracy : 99.53 %
test loss: 0.0152
-----test acc-----
Accuracy of plane : 94.10 %
Accuracy of car : 97.30 %
Accuracy of bird : 90.10 %
Accuracy of cat : 82.60 %
Accuracy of deer : 92.60 %
Accuracy of dog : 89.40 %
Accuracy of frog : 93.80 %
Accuracy of horse : 95.60 %
Accuracy of ship : 95.80 %
Accuracy of truck : 94.50 %
Overall accuracy : 92.58 %
test loss: 0.4484
```

最终测试集上准确率如上图所示，由于本次实验内容较为简单，我并没有反复调整训练策略，仅训练一次后直接汇报了结果，可以看到最终的训练集整体准确率为92.58%，证明模型能够较为成功地在cifar10上进行分类。值得注意的是在cifar10上的动物上，分类准确率明显低于其他工业制品的类别。

实验建议

如果更仔细地调整其他训练超参、增加训练时间，或者甚至简单地照抄[1]对于cifar10数据集采用的简化版resnet与调整好的超参，取得更高的准确率并不困难，但我并不认为这样做有更多的意义，也并不建议将90%以上的更高准确率作为刷分标准。

[1] He, Kaiming, et al. "Deep Residual Learning for Image Recognition."