



南開大學
Nankai University

计算机学院
深度学习与应用实验报告

卷积神经网络实验报告

学号：2112529

姓名：赵廷枫

2024 年 6 月 20 日

目录

1	原始版本卷积网络	2
1.1	网络结构	2
1.2	Loss 曲线和准确率曲线	2
2	ResNet	3
2.1	网络结构	3
2.2	Loss 曲线和准确率曲线	3
3	DenseNet	4
3.1	网络结构	4
3.2	Loss 曲线和准确率曲线	5
4	MobileNet	5
4.1	网络结构	5
4.2	Loss 曲线和准确率曲线	6
5	实验总结	7

1 原始版本卷积网络

卷积神经网络（Convolutional Neural Network, CNN）是一种深度学习模型，它通过卷积层提取局部特征，池化层降低特征维度，充分利用空间和局部信息，具有强大的表征学习能力。

实验指导中原始卷积网络的结构如下：

1.1 网络结构

```
Net(
  (conv1): Conv2d(3, 6, kernel_size=(5, 5), stride=(1, 1))
  (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (conv2): Conv2d(6, 16, kernel_size=(5, 5), stride=(1, 1))
  (fc1): Linear(in_features=400, out_features=120, bias=True)
  (fc2): Linear(in_features=120, out_features=84, bias=True)
  (fc3): Linear(in_features=84, out_features=10, bias=True)
)
```

图 1.1: 初始版本 CNN 结构

1.2 Loss 曲线和准确率曲线

使用上述网络结构经过 10 个 epoch 的训练，训练过程中的准确率达到 75.5%，训练过程的损失曲线和准确率曲线如下：

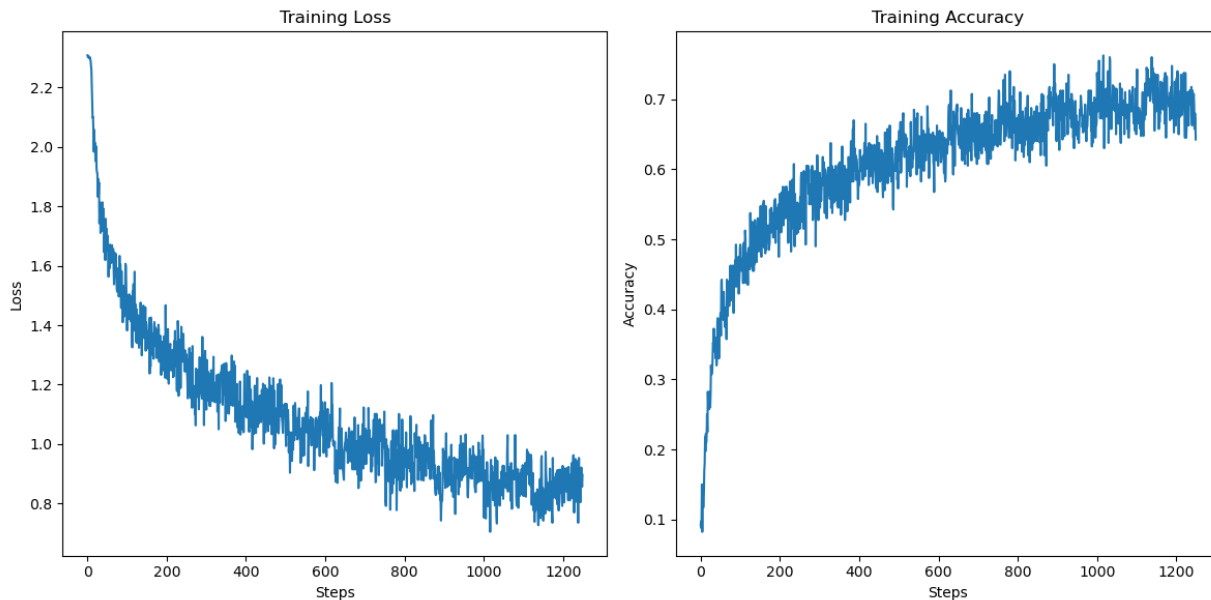


图 1.2: 初始版本 CNN 训练 Loss 曲线和准确率曲线，每 100 张图片为 1 个 step

之后我们在训练集上进行测试得到结果：

Class	Overall	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Accuracy (%)	61.0	68.1	67.2	46.1	39.2	59.3	49.4	68.3	67.9	80.9	64.9

2 ResNet

ResNet^[1] 通过引入残差块 (Residual Block), 在网络层之间增加直接的短连接 (skip connection), 解决了深层神经网络中的梯度消失和梯度爆炸问题, 从而使得训练更深层次的网络成为可能。根据论文我实现的模型结构如下:

2.1 网络结构

```
ResNet(
  (conv1): Conv2d(3, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
  (bn1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (layer1): Sequential(
    (0): BasicBlock(
      (conv1): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (bn1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (conv2): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (short_cut): Sequential()
    )
    (1): BasicBlock(
      (conv1): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (bn1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (conv2): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (short_cut): Sequential()
    )
  )
  (layer2): Sequential(
    (0): BasicBlock(
      (conv1): Conv2d(64, 128, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), bias=False)
      (bn1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (conv2): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (short_cut): Sequential(
        (0): Conv2d(64, 128, kernel_size=(1, 1), stride=(2, 2), bias=False)
        (1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      )
    )
    (1): BasicBlock(
      (conv1): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (bn1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (conv2): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1), bias=False)
      (short_cut): Sequential()
    )
  )
  (avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
  (linear): Linear(in_features=128, out_features=10, bias=True)
)
```

图 2.3: ResNet 网络结构

2.2 Loss 曲线和准确率曲线

使用上述网络结构经过 10 个 epoch 的训练, 训练过程中的准确率达到 91.750%, 训练过程的损失曲线和准确率曲线如下:

之后我们在训练集上进行测试得到结果:

Class	Overall	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Accuracy (%)	81.0	79.9	94.4	75.0	65.8	77.0	76.9	86.7	87.9	88.5	87.5

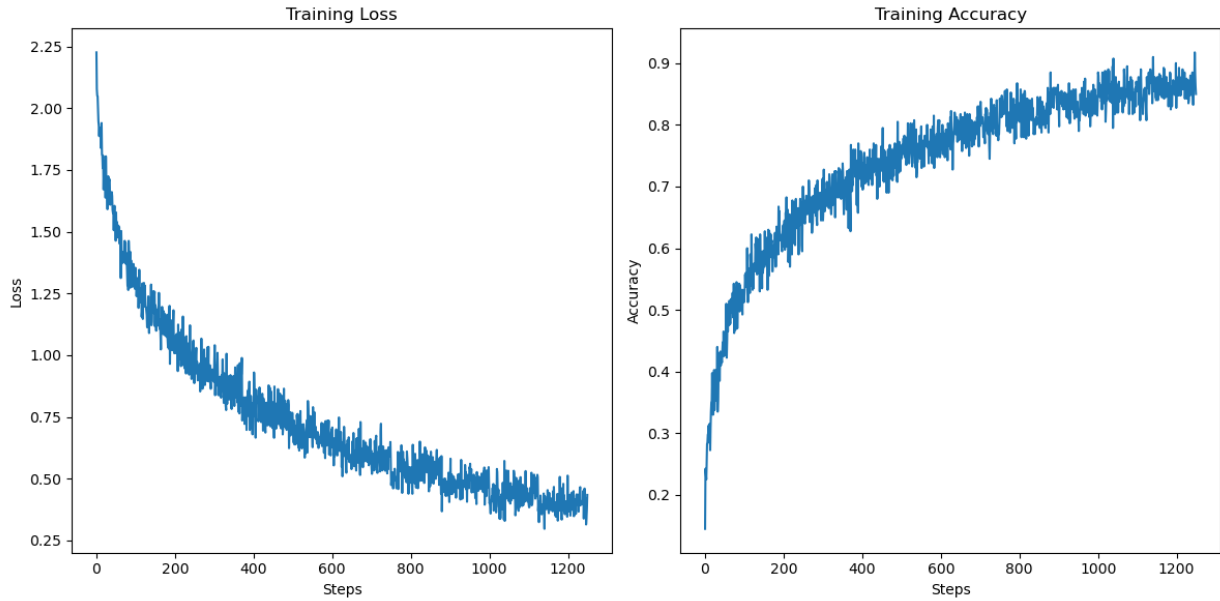


图 2.4: ResNet 训练过程的损失和准确率曲线

3 DenseNet

DenseNet[3] 在 ResNet[1] 的基础上通过在层之间建立密集连接，确保每一层都接收到前面所有层的特征图，有效地缓解了梯度消失问题，促进了特征的重用，提高了模型的效率和性能。根据论文我实现的模型结构如下：

3.1 网络结构

```

1 DenseNet(
2     conv:
3         BN_Conv2d(Conv2d(3, 32, kernel_size=7x7, stride=2, padding=3, bias=False),
4                     BatchNorm2d(32))
5     blocks:
6         Sequential(
7             DenseBlock(),
8             BN_Conv2d(Conv2d(64, 32, kernel_size=1x1, stride=1, bias=False),
9                         BatchNorm2d(32)),
10            AvgPool2d(kernel_size=2, stride=2),
11            DenseBlock(),
12            BN_Conv2d(Conv2d(64, 32, kernel_size=1x1, stride=1, bias=False),
13                        BatchNorm2d(32)),
14            AvgPool2d(kernel_size=2, stride=2),
15            DenseBlock(),
16            BN_Conv2d(Conv2d(64, 32, kernel_size=1x1, stride=1, bias=False),
17                        BatchNorm2d(32)),
18            AvgPool2d(kernel_size=2, stride=2),
19            DenseBlock()
20 )

```

```

21 fc :
22     Linear(in_features=64, out_features=10, bias=True)
23 )

```

3.2 Loss 曲线和准确率曲线

使用上述网络结构经过 10 个 epoch 的训练，训练过程中的准确率达到 97.000%，训练过程的损失曲线和准确率曲线如下：

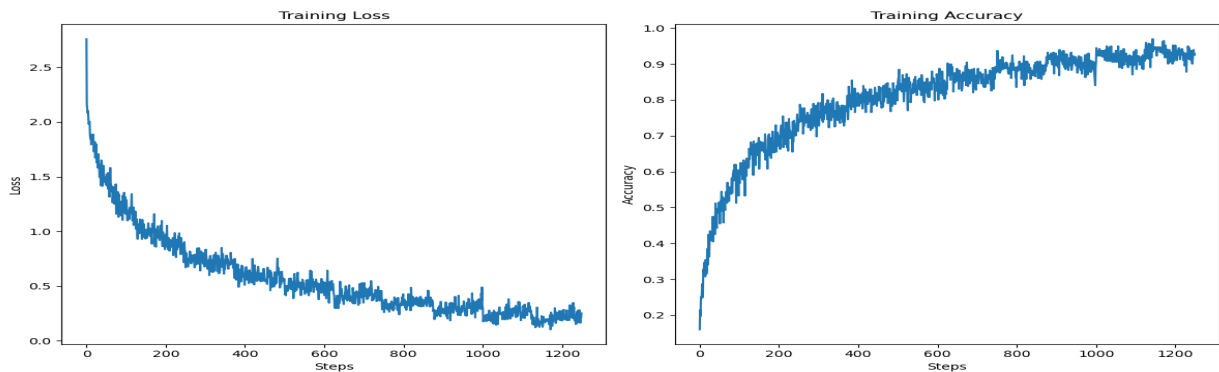


图 3.5: DenseNet 训练过程的损失和准确率曲线

之后我们在训练集上进行测试得到结果：

Class	Overall	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Accuracy (%)	83.7	81.2	95.1	81.2	65.8	81.8	77.6	87.6	89.5	85.8	89.4

4 MobileNet

MobileNet[2] 是一种轻量级深度卷积神经网络，主要用于资源受限的设备如移动和嵌入式设备。它采用深度可分离卷积 (Depthwise Separable Convolution) 来显著减少参数量和计算量，同时保持较高的准确性。

根据上述论文我设计的网络结构如下：

4.1 网络结构

```

1 MobileNetV1(
2     feature: Sequential(
3         # Initial Conv Layer
4         Conv2d(3, 4, kernel_size=3x3, stride=2, padding=1, bias=False)
5         BatchNorm2d(4)
6         ReLU()
7
8         # Depthwise Separable Conv Blocks
9         DepthwiseSeparableConv(4, 8, stride=2)
10        DepthwiseSeparableConv(8, 8)

```

```

11 DepthwiseSeparableConv(8, 16, stride=2)
12 DepthwiseSeparableConv(16, 16)
13 DepthwiseSeparableConv(16, 32, stride=2)
14 DepthwiseSeparableConv(32, 64)
15 DepthwiseSeparableConv(64, 128, stride=2)
16
17 # Adaptive Pooling
18 AdaptiveAvgPool2d(output_size=1)
19 )
20 classifier: Sequential(
21     Linear(in_features=128, out_features=10, bias=True)
22 )
23 )
24
25 def DepthwiseSeparableConv(in_channels, out_channels, stride=1):
26     return Sequential(
27         Conv2d(in_channels, in_channels, kernel_size=3x3, stride=stride, padding=1,
28             groups=in_channels, bias=False),
29         BatchNorm2d(in_channels),
30         ReLU(),
31         Conv2d(in_channels, out_channels, kernel_size=1x1, stride=1, bias=False),
32         BatchNorm2d(out_channels),
33         ReLU()
34     )

```

4.2 Loss 曲线和准确率曲线

使用上述网络结构经过 10 个 epoch 的训练，训练过程中的准确率达到 48.250%，训练过程的损失曲线和准确率曲线如下：

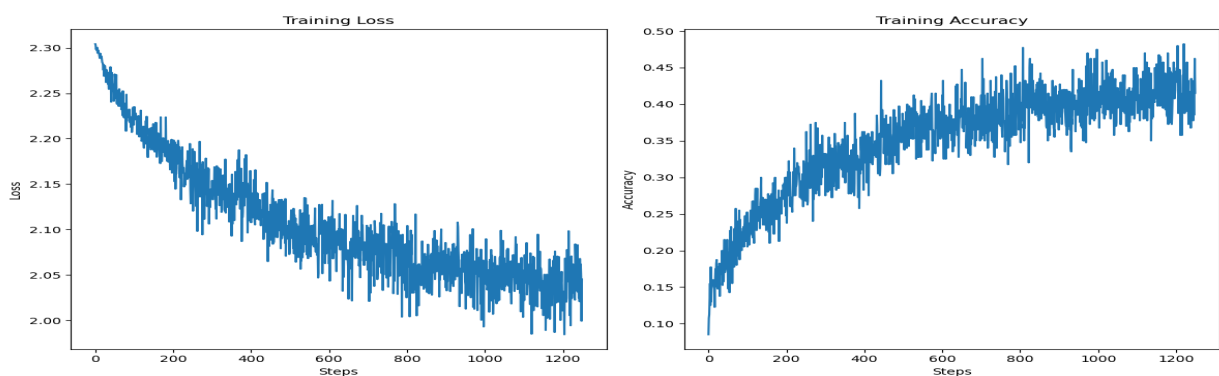


图 4.6: MobileNet 训练过程的损失和准确率曲线

之后我们在训练集上进行测试得到结果：

Class	Overall	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck
Accuracy (%)	42.0	52.3	52.2	0.0	25.7	36.6	38.0	58.2	51.5	56.8	49.6

5 实验总结

在本次实验中，我们比较了几种不同的卷积神经网络架构，包括原始版本的 CNN、ResNet[1]、DenseNet[3] 和 MobileNet[2]，并分析了它们在训练过程中的损失曲线和准确率曲线。首先，初始版本的 CNN 提供了一个基础的网络结构，但其性能在处理复杂任务时相对有限，损失和准确率曲线显示出较大的波动。

接下来，ResNet[1] 通过引入残差连接，显著改善了网络的训练效果。残差连接帮助解决了深度网络中的梯度消失问题，使得模型可以训练得更深，从而在复杂任务上表现出更好的性能。ResNet 的损失和准确率曲线相对平滑，表明其收敛速度和最终性能均有所提升。

DenseNet[3] 进一步通过密集连接（dense connections）优化了信息流动和梯度传播，每一层都接收所有前面层的特征图作为输入。这种设计提高了网络的参数效率和特征复用能力，DenseNet 的损失和准确率曲线显示出更稳定的收敛过程和较高的准确率。

MobileNet[2] 则采用深度可分离卷积（depthwise separable convolutions），显著减少了模型参数量和计算复杂度。尽管其参数较少，但 MobileNet 在任务中的表现依然优异，损失和准确率曲线表明其在保持较高效率的同时，仍能提供较好的分类效果。虽然我们在实验结果中 MobileNet 的分类效果并没有很好，但我们推测是因为训练轮数不够或者学习率过小导致训练不充分，后面我们会继续进行充分的实验来论证 MobileNet 的效果。

总体而言，不同的网络结构在处理复杂任务时各有优势。ResNet 和 DenseNet 在深度和特征复用方面表现出色，而 MobileNet 则在轻量化和高效性方面具有明显优势。这些实验结果为我们在实际应用中选择合适的网络结构提供了有力的参考。

总而言之，收获颇丰。

参考文献

- [1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.
- [2] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017.
- [3] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.