开课单位： 计算学院 ；时间：\_2024\_年\_\_10\_月\_ 17\_ 日；

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 题序 | 一 | 二 | 三 | 四 | 五 | 总 分 |
| 得分 |  |  |  |  |  |  |
| 评卷人 |  |  |  |  |  |  |

|  |
| --- |
| 得分 |
|  |

一．\_\_选择题\_\_\_(本大题共\_\_5\_\_题，每题\_\_\_6\_\_\_分，共\_\_\_\_30\_\_\_\_分。)

1．卷积神经网络中卷积层的主要作用是什么（ C ）。

A. 增加数据维度 B. 对数据进行分类

C. 提取特征并减少计算量 D. 完全连接所有输入和输出节点

2．卷积神经网络中池化层（Pooling layer）的作用是什么（ B ）

A. 增加模型参数量 B. 特征图降维以减少计算复杂度

C. 增加特征图数量 D. 提高模型非线性能力

3．卷积神经网络中填充（Padding）的目的是什么 ( B )

A. 增加额外隐藏层 B. 在特征图边缘加零以保持空间尺寸

C. 对特征图降维 D. 增加卷积核数量

4．以下哪个不是经典的卷积神经网络模型（ D ）。

A. VGG16 B. ResNets

C. GoogleNet D. Mask-RCNN

5．1个5x5的卷积核在一张224x224的灰度图中卷积，stride为1，其参数量是多少个（ B ）

A. 25 B. 50176

C. 1024 D. 1254400

|  |
| --- |
| 得分 |
|  |

二．\_\_\_判断题\_\_\_\_(本大题共\_\_5\_\_题，每题\_\_\_\_2\_\_分，共\_\_\_10\_\_\_\_\_分。)

1．卷积神经网络只能处理图像数据。 ( x )

2. ReLU是解决梯度消失问题的有效方法。 ( √ )

3. 卷积神经网络中卷积层的步长越大，输出特征图越大。( x )

4. 卷积神经网络的降采样层只能使用Pooling层。 ( x )

5. GoogleNet可以训练很深的网络，原因是引入了跳变连接。 ( √ )

|  |
| --- |
| 得分 |
|  |

三． \_\_\_简答题\_\_\_\_(本大题共\_\_5\_\_题，每题\_\_\_\_6\_\_分，共\_\_\_30\_\_\_\_\_分。)

1． 卷积神经网络和全连接神经网络有何区别？简单描述之。

CNN使用卷积层来提取局部特征，并通过滑动窗口方式减少参数数量，而FCNN每层的每个神经元都与前一层的所有神经元相连，参数数量多，计算量大。CNN适用于图像等具有空间结构的数据，而FCNN适用于没有明显空间结构的数据。

1. 请描述卷积神经网络主要有哪些网络层组成，作用分别是什么。

卷积神经网络主要由卷积层、激活层、池化层、全连接层和归一化层组成。

卷积层：提取特征，减少计算量。

激活层：引入非线性，使模型能够学习更复杂的特征。

池化层：降维，减少参数数量和计算量，提取重要特征。

全连接层：用于分类或回归任务，将特征映射到输出空间。

归一化层：如Batch Normalization，用于加速训练过程，提高模型稳定性。

3．有哪些常用的经典卷积神经网络模型，请举例并说明各网络的主要创新点（至少答出3个）。

1.LeNet：最早的CNN之一，用于手写数字识别，引入了卷积层和池化层的概念。

2.AlexNet：在ImageNet竞赛中取得突破性成绩，使用了ReLU激活函数和Dropout防止过拟合。

3.VGGNet：通过堆叠多个3x3的卷积层来增加网络深度，证明了网络深度对于性能的提升。

4.GoogleNet：引入了Inception模块和跳变连接，提高了网络的效率和准确性。

4．**为什么在卷积神经网络中可以引入归一化（Local Response Norm, Batch Norm等）？它如何影响模型的训练和性能？**

归一化可以减少内部协变量偏移，加速训练过程，提高模型的稳定性和准确性。例如，Batch Normalization通过规范化层的输入来加速训练，使模型对初始化不敏感，并且可以部分替代Dropout的作用。

5．卷积层有哪些变体，请举例？

1.扩张卷积：增加卷积核的感受野而不增加参数。

2.转置卷积：用于上采样，常见于生成模型和分割任务。

3.分组卷积：将输入通道分组，每组使用独立的卷积核，减少参数数量。

|  |
| --- |
| 得分 |
|  |

四． \_\_\_分析计算题\_\_\_\_(本大题共\_\_2\_\_题，共15分)

1. 给定一个输入特征图尺寸为11x11，使用一个5x5的卷积核进行卷积操作，步长为2，填充为1。请计算输出特征图的尺寸。（10分）

(11+2\*1-5)/2+1=5 输出特征图的尺寸为5\*5

2．一个3层卷积网络，输入为3通道的224x224大小的RGB图像，第一层32个3x3的卷积核，stride为1，padding为same模式；第二层64个3x3卷积核，stride为1，padding为same模式；第三层是神经元个数为512的全连接层，不考虑偏置，请计算该模型的参数量（5分）

第一层： 3\*32\*（3\*3）=864

第二层：32\*64\*（3\*3）=18432

第三层：（224\*224\*64）\*512=301989888

Sum=864+18432+301989888=302019184

|  |
| --- |
| 得分 |
|  |

五． \_\_\_开放题\_\_\_\_(本大题共\_\_1\_\_题，共15分)

1． 请设计一个简单的图像分割项目，使用卷积神经网络构建分割模型。描述项目的实现步骤及预期效果，其中必须说明选用哪种算法，算法步骤（可用伪代码）等。

使用U-Net，这是一个流行的用于图像分割的卷积神经网络架构。

1.数据准备：

收集图像分割数据集。

对图像进行预处理，包括缩放、归一化等。

生成图像的掩码（mask），用于训练时的像素级标注。

2.设计网络结构：

实现U-Net架构，包括卷积层、激活层、池化层、上采样层和跳跃连接。

3.模型训练：

定义损失函数（如交叉熵损失）和优化器。

使用数据集训练模型，并使用验证集进行模型选择。

4.模型评估：

使用测试集评估模型的性能。

计算评估指标，如准确率、召回率、IoU（交并比）等。

5.模型部署：

将训练好的模型部署到实际的应用中。

# 数据准备

data, masks = load\_data()

# 定义U-Net模型

class UNet():

    def \_\_init\_\_(self):

        # 初始化模型层

        self.encoder = ...

        self.decoder = ...

        self.skip\_connections = ...

    def forward(self, x):

        # 前向传播

    def train\_step(self, image, mask):

        # 单步训练

        self.optimizer.zero\_grad()

        output = self.forward(image)

        loss = self.loss(output, mask)

        loss.backward()

        self.optimizer.step()

        return loss.item()

# 实例化模型

model = UNet()

# 训练模型

for epoch in range(num\_epochs):

    for image, mask in data:

        loss = model.train\_step(image, mask)

    if epoch % log\_interval == 0:

        print(f'Epoch {epoch}, Loss: {loss}')

# 评估模型

metrics = evaluate\_model(model, test\_data)

print(f'Accuracy: {metrics["accuracy"]}, IoU: {metrics["iou"]}')

# 保存模型

save\_model(model)

**预期效果**

模型能够准确地分割出图像中的前景物体和背景。

在测试集上达到高准确率和高IoU。

模型能够泛化到新的、未见过的图像上。

注：IoU 的值范围从 0 到 1。IoU 值为 1 表示预测分割区域与真实分割区域完全一致，即分割完全正确。IoU 值为 0 则表示两者之间没有任何重叠。在实际应用中，通常使用一个阈值（如 0.5）来判断分割是否成功：如果 IoU 大于或等于这个阈值，则认为分割是成功的。

第6页共 6 页