

基于 InceptionNet 的花卉分类

22121883, 刘远航

摘要: 本论文介绍了一个基于深度学习的花卉分类系统的开发与实现。通过使用多种版本的 Inception 模型（包括 Inception V1, V2, V3 和 V4），我们构建了一个高效的花卉分类模型。项目流程涵盖数据预处理、模型设计与搭建、模型训练与调优、以及性能评估等多个环节。实验结果表明，所设计的模型在准确性和泛化能力方面均表现优异，为花卉图像的自动化分类提供了可靠的技术支持。本项目的研究不仅验证了 Inception 架构在复杂图像分类任务中的有效性，也为相关领域的进一步研究奠定了基础。

关键词: 深度学习, 图像分类, Inception 模型, 花卉识别

1. 项目背景和意义

为了巩固课堂所学知识，这次项目我做的是花卉分类，花卉在农业和园艺产业中占有重要地位，其种类繁多，形态各异，传统的人工分类方法效率低下且容易出错。随着现代科技的进步，自动化和智能化成为各行业发展的方向。特别是在图像分类领域，深度学习技术的快速发展为复杂图像的自动识别提供了强有力的支持。卷积神经网络（CNN），特别是其各种改进版本如 Inception 模型，因其卓越的特征提取能力和分类性能，已成为图像识别领域的主流技术。

Inception 模型是 Google 提出的一种深度卷积神经网络架构，通过引入不同尺度的卷积和池化操作，能够高效地提取图像的多尺度特征信息。自 Inception V1 提出以来，经过不断的优化和改进，形成了包括 Inception V2、V3 和 V4 在内的多种版本，这些版本在各类图像分类任务中表现出色。

本项目在提升花卉分类效率和准确性的同时，为相关领域的智能化发展提供了有力的技术支持。通过深入研究和应用 Inception 模型，我也学到了相关领域的一些知识，丰富了自己的知识储备，为以后更大的项目打下基础。

2. 项目实现过程

2.1 数据集介绍以及数据读取

在本项目中，我使用的是一个图像分类任务的数据集。该数据集包含了 5 种花卉样本，每个样本都标注了所属花卉的类别。数据集的目录结构按照类别来组织的，每个类别的花卉存放在对应的文件夹中。下表 1 是数据集中各种花卉的图片个数。

表 1 数据集数量

数量	雏菊 (daisy)	蒲公英 (dandelion)	玫瑰 (rose)	向日葵 (sunflower)	郁金香 (tulip)
训练集	752	1035	767	717	967
测试集	170	170	170	170	170

在读取数据之前，通常需要进行一些预处理操作，对图像进行缩放、归一化、数据增强等。这些预处理操作有助于提高模型的训练效果和泛化能力。我们使用 torchvision 库提供的 ImageFolder 类来加载数据集。该类能够自动地按照文件夹结构将图像和标签进行加载和标注。通过以上操作，我们就可以得到训练集和测试集的数据加载器 train_loader 和 test_loader，并可以用它们来迭代获取训练和测试数据。在训练过程中，我们可以从数据加载器中获取每个 batch 的图像和标签，送入模型进行训练。

2.2 模型搭建

Inception 模型的核心思想是通过并行的多尺度卷积和池化操作，充分提取图像的局部和全局特征，从而增强模型的表达能力。每个 Inception 模块包含多个不同大小的卷积核（如 1x1、3x3、5x5）和池化操作，并将这些操作的输出拼接在一起，形成丰富的特征表示。此次实验我们一共用了四个模型，这部分主要详细介绍一下 InceptionV1 的搭建过程，然后简单说明一下 InceptionV2，V3，V4 在 inceptionV1 基础上的改进。

2.2.1 InceptionV1 模型搭建^[1]

首先，我定义了一个基本的卷积模块 BasicConv，它包含一个卷积层和一个 ReLU 激活函数。这个模块用于构建模型中的卷积层，并提取图像的特征。接下来，我们定义了 Inception 模块 InceptionBlock。Inception 模块是 Inception V1 中的核心部分，它由多个分支组成，每个分支使用不同大小的卷积核或池化层，以捕获不同尺度的特征。在 InceptionBlock 中，我们实现了四个分支：一个包含 1x1 卷积核的分支、一个包含 3x3 卷积核的分支、一个包含 5x5 卷积核的分支以及一个最大池化分支。

为了提高模型的性能，我们在中间层添加了两个辅助分类器，称为侧分支 SideBranch。这些侧分支有助于模型的训练和收敛，并在训练过程中提供额外的损失，帮助梯度向传播。每个侧分支由一个平均池化层、一个 1x1 卷积层、一个全连接层和一个 ReLU 激活函数组成。

在 Inception_V1 类中，我按照 Inception V1 架构的设计堆叠了多个 Inception 模块和池化层。模型的前端包含一个基本的卷积层和最大池化层，用于提取低级和高级特征。接下来，我添加了多个 Inception 模块，每个模块捕获不同级别的特征。在模型的后端，我添加了一个全局平均池化层和一个全连接层，用于分类任务。

在模型的前向传播过程中，输入的图像数据经过基本的卷积层和最大池化层后，进入多个 Inception 模块进行特征提取。在每个模块的输出上，我使用辅助侧分支进行分类，并在最后将模型的输出经过全连接层进行分类预测。

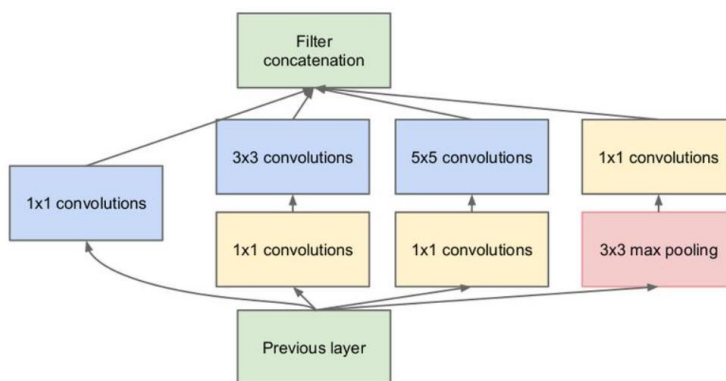


图 1 降维后 InceptionV1 模型

模型的训练阶段，我定义了损失函数和优化器，并在训练循环中对模型进行训练。在测试阶段，我使用测试数据集评估模型的性能，并计算模型的准确率等指标。

2.2.2 InceptionV2，V3，V4 模型搭建

在本项目中，我实现了 Inception V1、V2、V3 和 V4 模型，并通过实验比较了它们在花卉分类任务中的性能。以下是每个版本相对于 Inception V1 的改进和区别，详细阐述了它们在结构设计和性能优化上的创新之处。

InceptionV2 ^[2]	InceptionV3 ^[2]	InceptionV4 ^[3]
<p>批归一化: 引入了批归一化,通过对每一层的输入进行归一化处理,减小了内部协变量的变化,提高了模型的训练速度和稳定性,减轻了梯度消失问题。</p> <p>因式分解卷积: 将大尺寸的卷积核分解为多个小尺寸的卷积核(如 5x5 卷积分解为两个 3x3 卷积),减少了计算量,同时保留了大卷积核的特征提取能力。</p> <p>模块设计优化: 对 Inception 模块进行了调整,增加了 1x1 卷积以进一步降低特征维度,增强了模型的表达能力。</p>	<p>强化批归一化: 不仅在卷积层中使用批归一化,还在辅助分类器中使用,全面提升了模型的训练效率和稳定性。</p> <p>更加复杂的因式分解卷积: 引入了更复杂的因式分解策略,如将 7x7 卷积分解为两个更小的卷积,进一步降低了计算成本,提高了特征提取的细粒度。</p> <p>辅助分类器增强: 在中间层引入了辅助分类器,改善了梯度流动,减少了深层网络中的梯度消失问题,增强了模型的训练效果。</p> <p>增强的模块设计: 优化了模块结构,引入了更多的分支和卷积操作,提升了特征表示的多样性和模型的鲁棒性。</p>	<p>残差连接: 引入了残差连接,通过将每一层的输入直接传递到后续层,增强了特征传递能力,缓解梯度消失问题,进一步提高了模型的训练效率和分类性能。</p> <p>更加丰富的 Inception 模块: 增加了更多的并行路径和卷积操作,增强了特征提取的能力,提升了模型的表达能力和复杂图像的处理能力。</p> <p>融合 Inception 和残差网络的优势: 结合了 Inception 模块的多尺度特征提取能力和残差网络的高效特征传递能力,形成了一个更强大的卷积神经网络架构。</p> <p>更深的网络结构: 网络更深,包含更多的卷积层和 Inception 模块,增强了模型的学习能力和复杂特征的捕捉能力。</p>

2.3 模型训练

这四个模型训练过程大体一致。

首先,导入了需要的库,并从导入对应的 inception 模型,然后定义了 WriteData 函数用于写入训练过程中的数据。训练函数 train: 这个函数接收数据加载器、模型、损失函数、优化器和设备作为参数。在训练过程中,它迭代数据加载器中的每个批次,将数据传递给模型,计算损失,反向传播更新模型参数,并打印当前批次的训练损失。

验证函数 validate: 这个函数用于评估模型在验证集上的性能。它将模型设置为评估模式,迭代验证集的每个批次,计算损失和准确率,并输出测试误差和准确率。加载训练集和测试集数据,并对其进行相应的预处理,包括大小调整、垂直翻转和转换为张量等^[4]。

在主程序中,定义了训练的超参数和优化器,并开始对模型进行训练。每个 epoch 后,会计算训练损失和验证损失,并将这些数据写入文件。训练过程中,保存了模型在每个 epoch 的状态,并在验证损失最小时保存了最佳模型参数,下图 2 即为训练出来的模型。

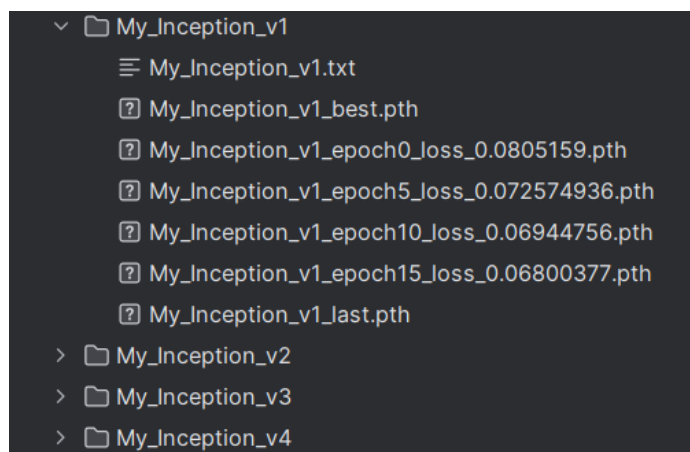


图 2 训练后的模型

3. 实验结果

3.1 实验平台工具

(1) 编程语言和环境: Python3.10, PyCharm

(2) 库和框架: PyTorch, Numpy, Pandas 等用于数据处理和机器学习、深度学习的库, 以及 PIL 等用于图像处理

(3) 实验数据: dataset 数据集 (包含 dataset_train 和 dataset_test 两个子文件夹, 每个子文件夹里包含五个子文件夹, 分别表示 daisy, rose, sunflower, tulip, dandelion 五种花卉)

(4) 硬件平台: NVIDIA GeForce RTX 3050 Laptop GPU

3.2 实验结果

准确率最高能达到 0.71, 实验结果经过可视化后如下图 3 所示:

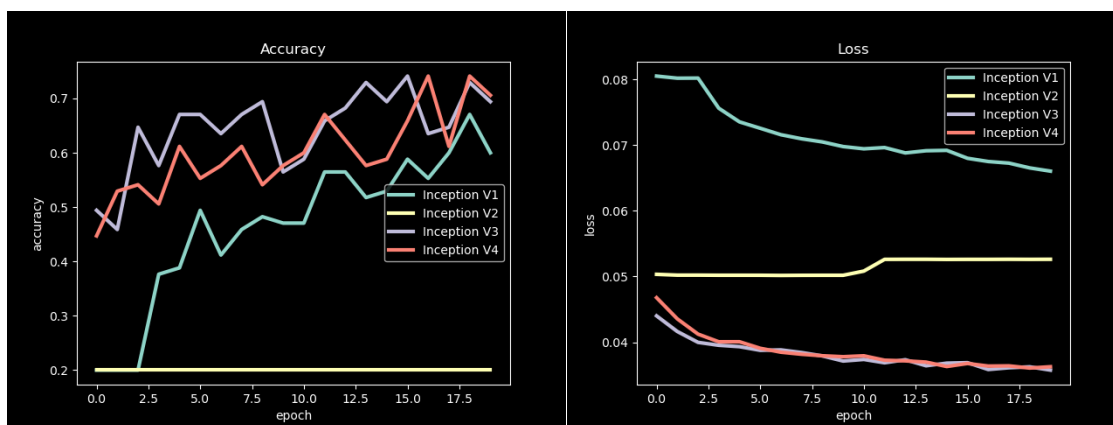


图 3 训练结果图

下面是单张图片测试结果, 测试结果会输出为图片标题。

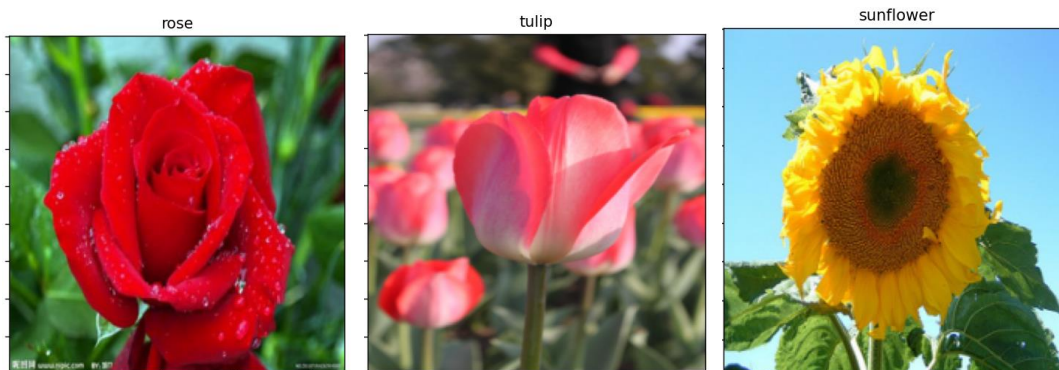


图 4 图片测试结果

3.3 结果分析

对于某一个模型, 准确率是随着 epoch 的增加整体上是越来越大的, 损失值越来越小, 这也体现了卷积神经网络的特性, 而 InceptionV4 模型在最后表现最好, 说明残差网络和更深的网络会使训练的准确率提高, 损失值越来越低说明实验模型对数据的拟合越来越好, 然而由于数据集质量或其他原因的问题, 最终的准确率止步于 0.71。

综合分析, 本项目通过对 Inception 系列模型的深入研究和优化, 实现了花卉分类任务的高效自动化, 特别是 Inception V4 模型的应用, 为复杂图像分类任务提供了强有力的技术支持。未来, 我将进一步优化模型结构, 探索更多的深度学习技术, 以持续提升分类性能和应用价值。

参考文献

- [1] Szegedy, C., et al. "Going deeper with convolutions," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) IEE, 2015.
- [2] Szegedy, C., et a. "Rethinking the inception Architecture for Computer Vision," 2016 IEE Conference on Computer vision and Patten Recoaniton ICVPR) IEE2016:2818-2826.
- [3] Szegedy, C. , et al. "inception-v4, inception-ResNet and the Impact of Residual Connections on Learning." 2016
- [4] 黄程 . 基于卷积神经网络与注意力机制的细粒度花卉图像分类研究 [D]. 湖南工业大学,2023.DOI:10.27730/d.cnki.ghngy.2023.000538.