

**专业综合实践论文**

|  |  |
| --- | --- |
| **题 目：** | 基于机器学习的蔬菜图像分类 |
| **学 院：** | 人工智能学院 |
| **专 业：** | 数据科学与大数据技术 |
| **年级班别：** | 2021级数据科学与大数据技术1班 |
| **学 号：** | 421470141 |
| **学生姓名**： | 彭智宇 |

|  |  |
| --- | --- |
| **日期** | 2024.11.27 |

**提交日期：** 2024 **年** 11 月

[1 绪论](#_Toc7844)**[目 录](#_Toc7844)**

[2](#_Toc7844)

[1.1 选题背景 2](#_Toc20635)

[1.2 研究内容 2](#_Toc31626)

[2 数据收集与处理 4](#_Toc4786)

[2.1 数据收集 4](#_Toc13836)

[2.1.1 数据收集方法 4](#_Toc16246)

[2.1.2 数据内容描述 4](#_Toc32514)

[2.2 数据处理 5](#_Toc19062)

[2.2.1 数据缩放 5](#_Toc12679)

[2.2.2 数据标准化 5](#_Toc10491)

[2.2.3 数据增强 6](#_Toc25047)

[3 图像特征提取 7](#_Toc10759)

[3.1 颜色直方图特征 7](#_Toc18397)

[3.2 灰度化特征 7](#_Toc24350)

[3.3 边缘检测特征 7](#_Toc2879)

[3.4 纹理特征 8](#_Toc9370)

[3.4 特征合并与数据集划分 9](#_Toc25858)

[4 模型建立和评估 10](#_Toc3413)

[4.1 kNN算法模型建立 10](#_Toc27370)

[4.2 决策数据算法模型建立 10](#_Toc21742)

[4.3 xgboost算法模型建立 11](#_Toc15176)

[4.4 模型评估 12](#_Toc21754)

[5 结论 14](#_Toc13272)

[得 分 15](#_Toc7378)

# 1 绪论

## 1.1 选题背景

在现代农业中，蔬菜种类繁多，不同种类的蔬菜在种植、管理和销售等方面有着显著差异。传统的蔬菜种类识别主要依赖于人工，这不仅耗时耗力，而且容易出现误判。随着科技的不断进步，特别是人工智能和机器学习技术的迅速发展，将这些技术应用于农业领域已经成为一种趋势。通过机器学习算法进行蔬菜图像分类，不仅能够提高识别效率，还能减少人工成本，提升农业生产的自动化和智能化水平。

图像分类作为计算机视觉领域的重要任务，已经在许多实际应用中取得了显著成果。蔬菜图像分类作为其中一个分支，能够帮助农民和农业从业者快速准确地识别不同种类的蔬菜，从而优化生产和销售流程。比如，在蔬菜收获后，利用图像分类技术可以快速将不同种类的蔬菜进行分拣，提高分拣效率和准确性。此外，在农贸市场和超市中，应用该技术也可以方便顾客识别和选购蔬菜，从而提升购物体验。

综上所述，基于机器学习的蔬菜图像分类具有重要的现实意义和应用价值。它不仅能够推动农业生产方式的革新，还能为农业领域的信息化和智能化发展提供技术支持。本研究旨在通过探索和优化机器学习算法，构建高效、准确的蔬菜图像分类模型，为农业生产和管理提供科学依据和技术手段。

## 1.2 研究内容

本研究主要围绕蔬菜图像分类这一主题展开，旨在通过机器学习技术对不同种类的蔬菜进行有效分类。具体研究内容如下：

（1）首先，进行数据收集与处理。通过手机网络，收集多种常见蔬菜的图像数据，确保数据的多样性和标签的准确性。然后，对收集到的数据进行预处理，包括图像缩放、标准化和数据增强，以提高模型的泛化能力和训练效果。

（2）其次，进行图像特征提取。通过颜色直方图、灰度化、边缘检测和纹理特征等方法，提取图像的关键特征。这些特征不仅能够反映蔬菜图像的颜色和形状，还能捕捉图像的细节和纹理信息。将不同特征进行合并，形成完整的特征向量，为后续的模型训练提供输入。

（3）然后，进行模型建立与评估。分别使用kNN、决策树和XGBoost等机器学习算法构建蔬菜图像分类模型。通过交叉验证和参数调优，优化各个模型的性能。采用混淆矩阵、准确率、精确率、召回率和F1值等指标，对模型的分类效果进行评估和比较，选出最佳模型。

（4）最后，对研究结果进行总结和分析。讨论不同算法在蔬菜图像分类中的表现，分析模型的优缺点，并提出改进方案和未来研究方向。

# 2 数据收集与处理

## 2.1 数据收集

### 2.1.1 数据收集方法

本研究的数据收集主要通过手机进行。首先，在多个超市、农贸市场和家庭种植园中拍摄各种蔬菜的图像。为了确保数据的多样性和真实性，每种蔬菜的图像都在不同的光照条件和背景下拍摄。

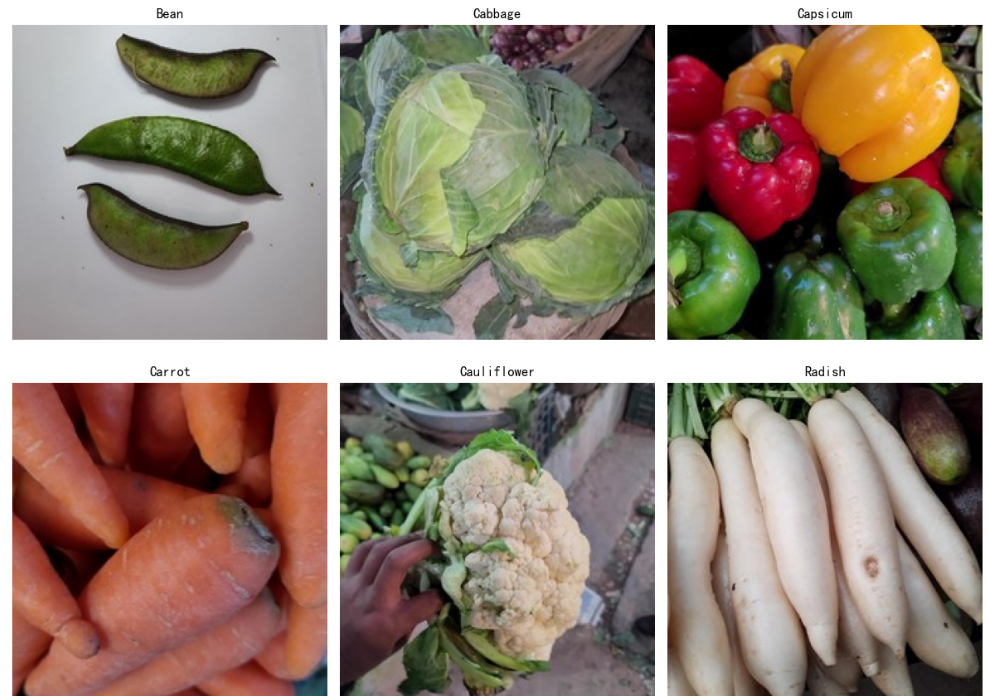


图2-1 数据示例

### 2.1.2 数据内容描述

本研究的数据集包含六种常见蔬菜的图像，分别是豆类（Bean）、卷心菜（Cabbage）、辣椒（Capsicum）、胡萝卜（Carrot）、花椰菜（Cauliflower）和萝卜（Radish）。

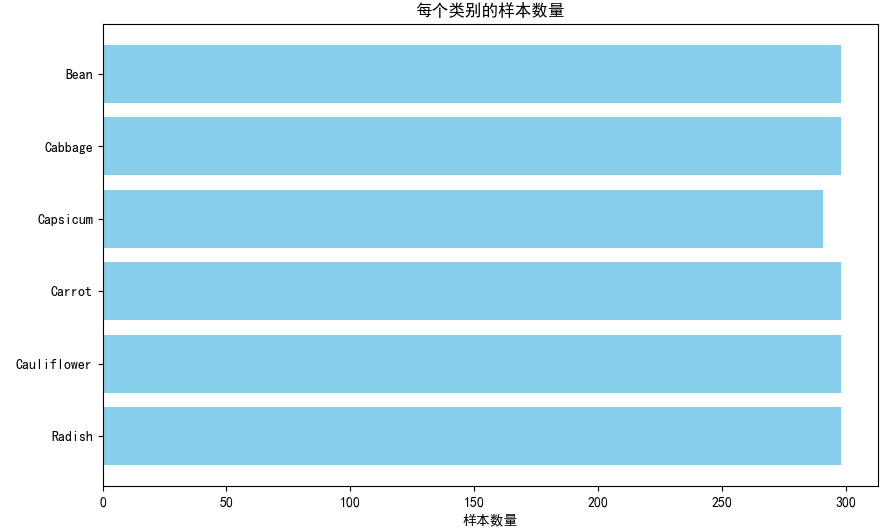


图2-2 数据数量

每种蔬菜的图像数量如下：类别 'Bean' 的数据量为298张，类别 'Cabbage' 的数据量为298张，类别 'Capsicum' 的数据量为291张，类别 'Carrot' 的数据量为298张，类别 'Cauliflower' 的数据量为298张，类别 'Radish' 的数据量为298张。总体而言，数据集包含1781张高质量的蔬菜图像。

## 2.2 数据处理

### 2.2.1 数据缩放

读取'./data'目录下的图像文件，使用Python中的PIL库（Pillow）来读取图像数据并统一大小为140x140。此外，使用一些库来处理图像文件名并将其映射到对应的标签。

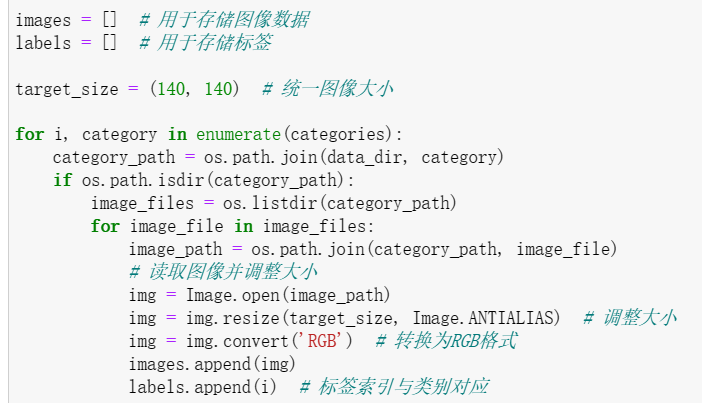


图2-3 数据缩放代码

### 2.2.2 数据标准化

在数据预处理阶段，对图像数据进行了标准化处理。以下是标准化的步骤：

（1）转换为数组：

首先，将图像数据转换为了NumPy数组，以便于数值计算和处理。

（2）归一化处理：

接着，对图像数据进行了归一化处理。通过除以255.0，将像素值缩放到 [0, 1] 的范围内。这种归一化方法可以帮助模型更快地收敛，并且在一定程度上提高模型的稳定性和训练效果。

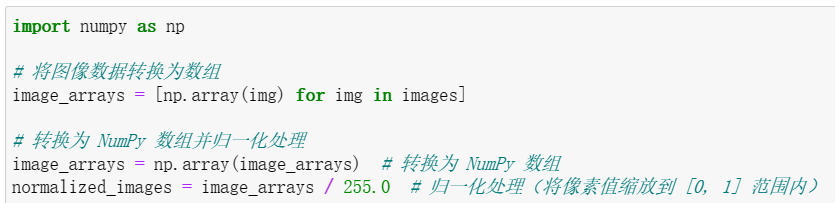


图2-4 数据标准化代码

### 2.2.3 数据增强

在数据预处理过程中，数据增强是一种有效的技术，用于通过对现有数据进行变换和扩充来增加数据集的大小和多样性。以下是数据增强的步骤和操作：

数据增强操作：使用ImageDataGenerator库创建了数据增强的生成器。应用了多种数据增强操作，包括随机旋转、移动、错切、缩放、翻转等，增加了图像数据的多样性和鲁棒性。

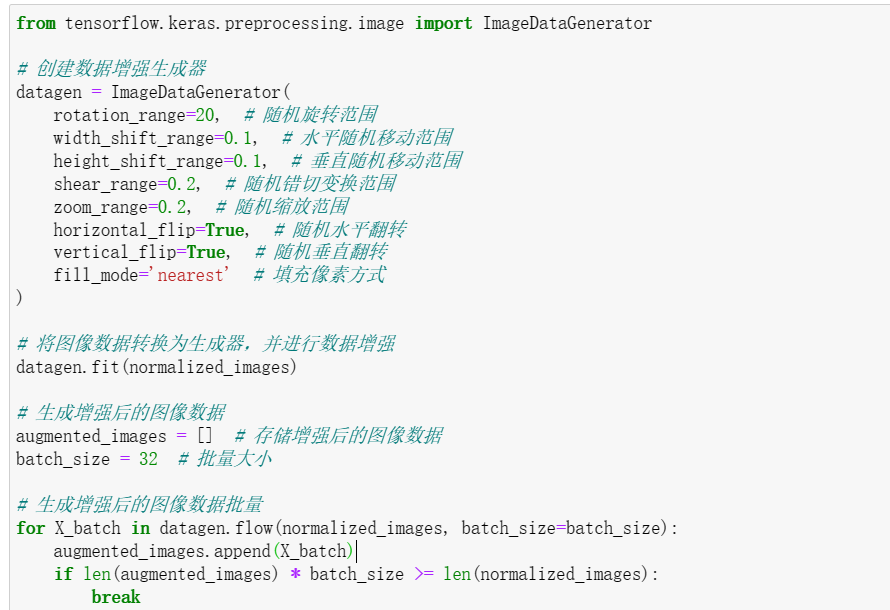


图2-5 数据增强代码

增强后的数据合并：将增强后的图像数据与原始标准化图像数据合并为一个更大的数据集。通过NumPy库的函数确保了合并后的图像和标签数据一一对应。

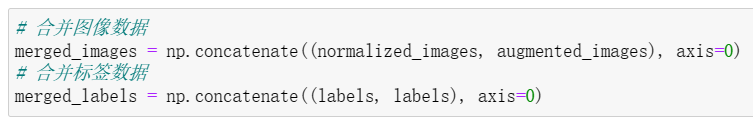


图2-6 数据特征合并代码

# 3 图像特征提取

## 3.1 颜色直方图特征

使用extract\_color\_histogram函数，通过cv2.calcHist计算每张图像的颜色直方图。这个直方图是基于图像的BGR通道，使用8x8x8的直方图来表示颜色分布情况。之后对直方图进行了归一化处理，并将其展平为一维数组，以便于后续的处理和分析。

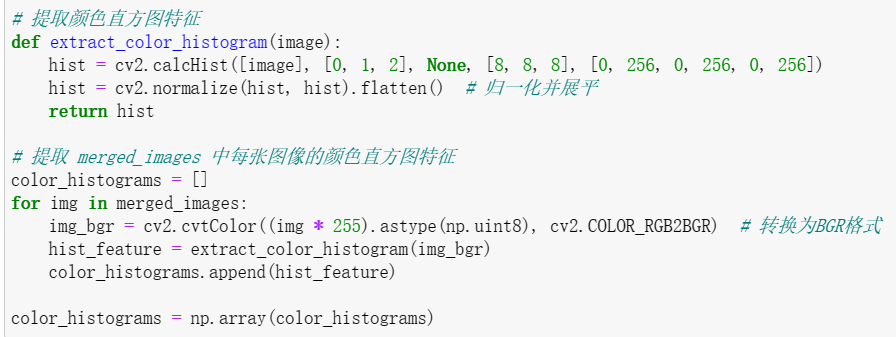


图3-1 图像颜色直方图特征提取代码

这种特征提取方法有助于将图像转换为数值特征，以便于机器学习模型的训练和分析。颜色直方图特征反映了图像在颜色空间中的分布情况.

## 3.2 灰度化特征

为从彩色图像中提取灰度特征，将图像转换为灰度图像并提取其灰度直方图。将每张彩色图像转换为灰度图像，并计算了灰度直方图特征。

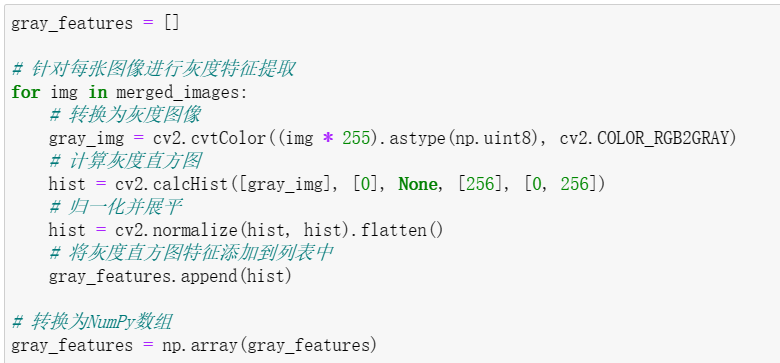


图3-2 图像灰度直方图特征提取代码

最终，gray\_features列表包含了所有图像的灰度直方图特征。这些特征可以用于后续的机器学习模型训练任务。

## 3.3 边缘检测特征

边缘检测是一种图像处理技术，用于识别图像中的边界和轮廓。在OpenCV中，常用的边缘检测算法包括Canny边缘检测器。



图3-3 图像边缘检测特征提取代码

将每张图像转换为灰度图像，并使用Canny边缘检测算法检测边缘。然后计算了边缘直方图特征，最终得到了edge\_features列表，包含了所有图像的边缘特征。

## 3.4 纹理特征

纹理特征可以通过不同的方法提取，其中一种常用的方法是使用局部二值模式（Local Binary Patterns，LBP）来描述图像的纹理信息。

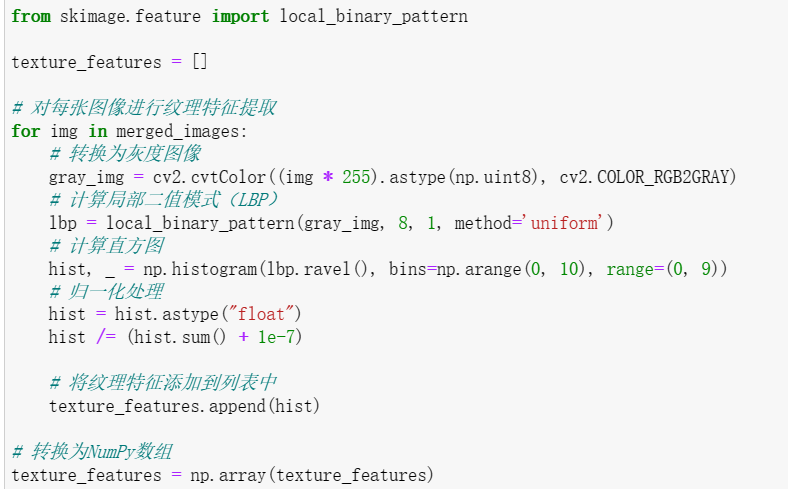


图3-4 图像局部二值模式特征提取代码

使用了scikit-image库中的local\_binary\_pattern函数来计算局部二值模式（LBP），然后计算了LBP直方图特征。最终得到了texture\_features列表，包含了所有图像的纹理特征。

## 3.4 特征合并与数据集划分

将四种特征合并为最终的特征表示通常涉及将它们连接起来，形成一个单一的特征向量。这个矩阵可以作为最终的特征表示，用于机器学习模型的训练或其他任务。

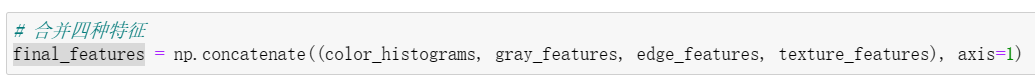


图3-5 图像特征合并代码

对提取的特征进行可视化时，使用直方图或其他图形来展示特征的分布情况。绘制每种特征的直方图，结果如下所示。

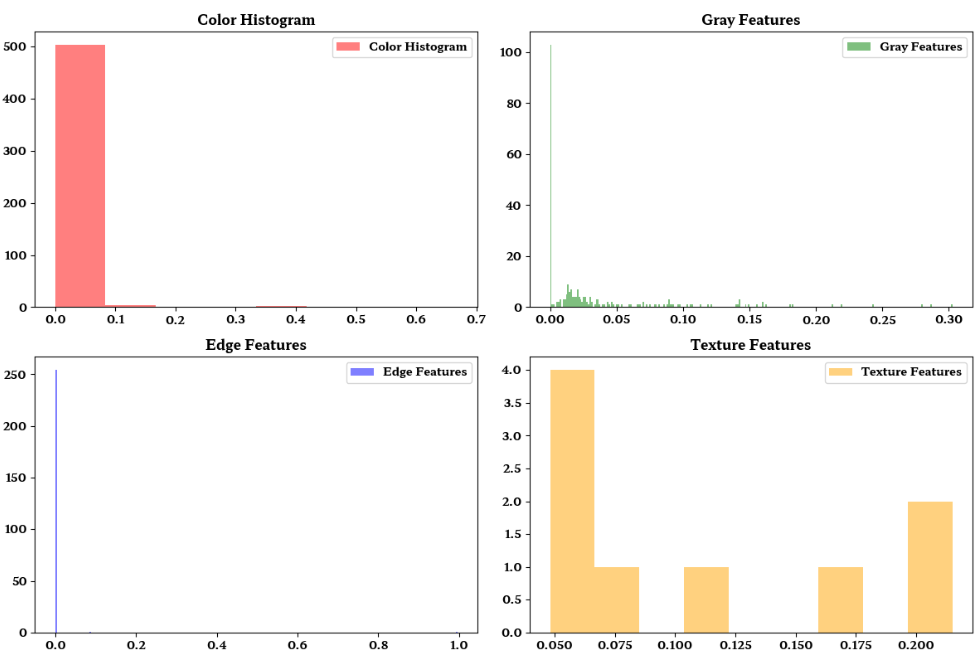


图3-6 图像特征分布直方图

这四种特征提取方法产生了各自独特的直方图分布。颜色直方图展现了图像在颜色空间中的分布情况，灰度特征呈现了图像像素强度的分布，边缘特征揭示了图像中边缘的出现频率，而纹理特征则捕捉到了图像局部纹理结构的信息。

划分数据集为训练集和测试集是为了在模型训练过程中验证模型的性能和泛化能力。使用train\_test\_split函数将特征集final\_features和标签merged\_labels划分为训练集和测试集。test\_size参数指定了测试集的比例20%。

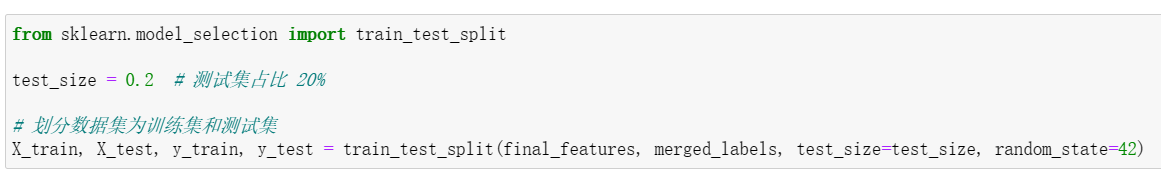


图3-7 数据集划分代码

# 4 模型建立和评估

## 4.1 kNN算法模型建立

kNN是一种基于实例的学习方法，它根据特征空间中相邻点的类别进行分类。其原理是通过测量不同特征之间的距离，找到离待分类样本最近的k个训练样本，通过多数表决来确定待分类样本所属的类别。

使用KNeighborsClassifier类建立kNN模型。首先，导入KNeighborsClassifier类，然后创建一个实例，可以设定k值和其他参数，最后使用fit()方法拟合训练数据。

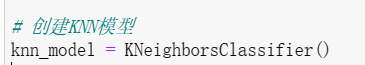


图4-1 KNN模型构建代码

对KNN模型进行交叉验证，搜索不同邻居数量（n\_neighbors）下的最佳性能。在搜索过程中，会尝试不同的邻居数量，从1到20，并绘制出每个邻居数量下的平均准确率曲线图。

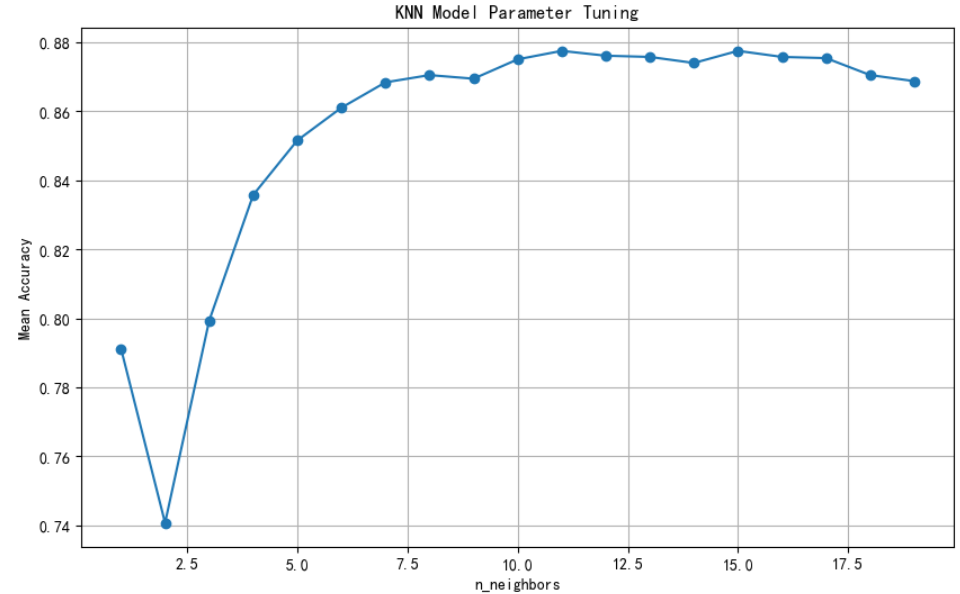


图4-2 KNN模型参数调优曲线图

根据曲线图选择最优的邻居数量15。

## 4.2 决策数据算法模型建立

决策树是一种基于树结构的分类模型，通过对特征进行递归划分来实现分类。它通过选择最优特征对数据进行划分，构建一个树状结构，在每个节点根据特征的值进行分支，直到叶子节点表示数据的最终分类。

使用Scikit-learn库中的DecisionTreeClassifier类可以建立决策树模型。导入类后，创建一个实例，可以设定树的深度、分裂标准等参数，然后使用fit()方法拟合训练数据。

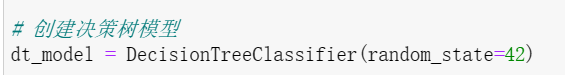


图4-3决策数模型构建代码

决策树算法时，常用的参数包括树的深度（max\_depth）、最小分割样本数（min\_samples\_split）、最小叶节点样本数（min\_samples\_leaf）等。使用交叉验证来搜索最佳的参数组合，进而提高模型性能。



图4-4决策数模型参数调优结果

## 4.3 xgboost算法模型建立

XGBoost是一种梯度提升树（Gradient Boosting Decision Tree）的算法。它采用多个决策树进行集成学习，通过迭代训练弱分类器来不断优化损失函数，并结合正则化项，使得模型更具泛化能力，提高对数据的拟合效果。

XGBoost是一个强大的梯度提升库，可以使用XGBClassifier类建立分类模型。首先导入该类，创建一个实例，可以设定树的数量、学习率等参数，然后使用fit()方法拟合训练数据。

使用GridSearchCV对XGBoost模型进行参数搜索。参数范围包括learning\_rate、max\_depth、subsample和colsample\_bytree等。GridSearchCV会通过交叉验证来寻找在给定参数范围内最优的参数组合。

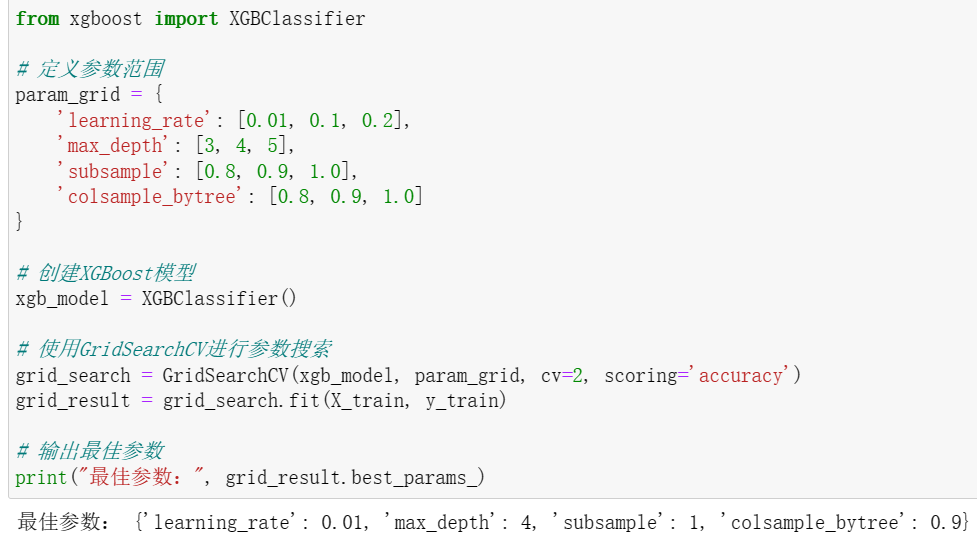


图4-5XGBoost模型参数调优结果

## 4.4 模型评估

计算每个模型的模型的准确率、精度、召回率和F1值。结果如下图所示。

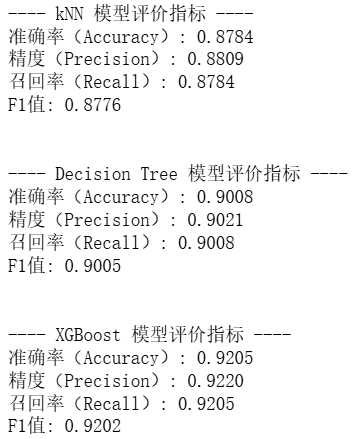


图4-6 三种模型的评估结果

从评估结果来看，XGBoost模型在所有指标上都表现最佳，准确率为0.9205，精度为0.9220，召回率为0.9205，F1值为0.9202，显著优于kNN和决策树模型。这表明XGBoost模型在处理我们的数据时具备更强的分类能力和泛化能力，能够更有效地捕捉数据的复杂关系。其优异表现主要得益于梯度提升算法的强大能力，通过迭代训练弱分类器不断优化损失函数，使模型在处理非线性问题时具有显著优势。

相比之下，决策树模型的表现次之，准确率为0.9008，精度为0.9021，召回率为0.9008，F1值为0.9005，虽然不如XGBoost，但也表现出较强的分类能力。决策树模型能够通过递归划分特征空间构建树状结构，实现较为精细的分类。然而，其易于过拟合的特点在一定程度上限制了其泛化能力。而kNN模型的表现相对较弱，虽然其准确率达到了0.8784，但在精度、召回率和F1值上均低于其他两个模型。这主要是由于kNN模型对数据的敏感性较高，且在处理高维数据时表现不佳。总体而言，XGBoost模型在本次分类任务中表现最佳，是最终推荐的模型。

# 5 结论

本研究以机器学习为基础，旨在对蔬菜图像进行分类，并通过对比不同算法的性能，选择最佳的分类模型。首先，通过手机采集了包含‘Bean’、‘Cabbage’、‘Capsicum’、‘Carrot’、‘Cauliflower’和‘Radish’六类共1781张图像，并进行了数据缩放、标准化和数据增强等预处理步骤。接着，提取了颜色直方图、灰度特征、边缘检测特征和纹理特征四种图像特征，并将它们合并为最终的特征表示。然后，分别建立了kNN、决策树和XGBoost三种分类模型，并通过交叉验证和参数调优来优化每个模型的性能。最终，通过准确率、精度、召回率和F1值对各模型进行了评估。

研究结果显示，XGBoost模型在所有评估指标上均表现最佳，其准确率达到了0.9205，显著优于kNN和决策树模型。这表明XGBoost模型在处理本研究中的图像分类任务时，具备更强的分类能力和泛化能力。决策树模型次之，虽然表现较好，但由于易于过拟合，其泛化能力稍显不足。kNN模型虽然简单易用，但在高维数据处理上表现较弱。通过本次研究，我们不仅掌握了图像预处理和特征提取的技术方法，还深入理解了不同机器学习算法在图像分类任务中的适用性和优缺点。