

东北大学计算机科学与工程学院

机器学习课程设计报告

题目 : Plant Seedlings Classification

2021 年 12 月

目录

1 课题概述	3
1.1 问题描述.....	3
1.2 评价指标.....	3
2 分工情况	3
3 数据处理	4
3.1 数据读取.....	4
3.2 图像掩模处理.....	4
4 特征工程	5
4.1 HOG 特征	5
4.2 颜色特征.....	6
4.3 SIFT	6
5 模型训练	7
5.1 模型选择.....	7
5.1.1 词袋模型.....	7
5.1.2 K-means.....	7
5.1.3 支持向量机.....	8
5.2 模型训练.....	8
6 特征与模型的选择和超参数的调整	9
6.1 特征的选择.....	9
6.1.1、HOG 特征。	9
6.1.2、LBP 特征。	10
6.1.3、sift+bag of words 特征。	10
6.1.4、颜色特征。	10
6.2 特征的融合.....	10
6.3 超参数的调整.....	11
7 模型评价与分析	12
7.1 结果分析.....	12
7.2 合理性与创新性.....	12
7.2.1 模型合理性.....	12
7.2.2 模型创新性.....	13
8 收获与体会	13

1 课题概述

1.1 问题描述

你能区分杂草和作物幼苗吗?

有效地做到这一点意味着提高作物产量和改善环境管理。

奥胡斯大学信号处理小组与南丹麦大学合作,最近发布了一个数据集,其中包含了大约 960 种独特植物的图像,属于 12 个不同生长阶段的物种。用于植物幼苗分类算法基准的公共图像数据库。

对于测试集中的每个文件,您必须预测物种变量的概率。

1.2 评价指标

提交的作品在 MeanFScore 上进行评估,在 Kaggle,采用 micro-F1 作为多分类问题的评价指标。结果分数以这种方式计算:

$$MeanFScore = F1_{micro} = \frac{2Precision_{micro}Recall_{micro}}{Precision_{micro} + Recall_{micro}} \quad (1)$$

其中 precision 为精确率,关注预测为正样本的数据(可能包含负样本)中,真正正样本的比例;recall 为召回率,关注真正正样本的数据(不包含任何负样本)中,正确预测的比例。计算时先计算所有类别的总的 Precision 和 Recall,然后计算出来的 F1 值即为 micro-F1。

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

其中 TP(真阳性):正样本被正确预测为正样本;FP(假阳性):负样本被错误预测为正样本;TN(真阴性):负样本被正确预测为负样本;FN(假阴性):正样本被错误预测为负样本。

micro-F1 指标综合了 Precision 与 Recall 的产出的结果,在计算公式中考虑到了每个类别的数量,所以适用于数据分布不平衡的情况。micro-F1 的取值范围为 0 到 1,1 代表模型的输出最好,0 代表模型的输出结果最差。

2 分工情况

3 数据处理

3.1 数据读取

数据集为 12 种不同类别的植物图像，需要先对图像数据进行导入。采用 python 中的计算机视觉库 cv2 中的 `imread` 函数对图片进行读取，由于数据集均为彩色图像，故返回的数据为图片每个像素的 B, G, R 值组成的三维数组。分别对训练集和测试集的图片进行读取，保存至相应的列表中。

3.2 图像掩模处理

掩模是由 0 和 1 组成的一个二值图像。当在某一功能中应用掩模时，1 值区域被处理，被屏蔽的 0 值区域不被包括在计算中。通过指定的数据值、数据范围、有限或无限值、感兴趣区和注释文件来定义图像掩模，也可以应用上述选项的任意组合作为输入来建立掩模。

在本实验中，由于输入的图像中存在植物的土壤、标签、容器等无关内容，故采用掩模处理将图像中的植物部分提取出来，消除无关背景对训练过程及预测过程的影响。

关于掩模的创建，先采用 cv2 中的 `cvtColor` 函数将图像的 RGB 表示转化成 HSV 表示，对绿色的区分度更明显，更易提取。给定绿色部分的 HSV 范围后，利用 cv2 中的 `inRange` 函数进行二值化处理，将属于这个范围的的部分的值变为 255，其余部分变为 0，用来将绿色部分提取出来。

初步的掩模创建完成后，采用图像形态学的开操作(先腐蚀后膨胀)用来消除小物体、在纤细点处分离物体、平滑较大物体的边界的同时并不明显改变其面积，使掩模处理的结果更加精确。这里采用 cv2 中的 `getStructuringElement` 函数，创建 kernel 操作矩阵，然后调用 cv2 中的 `morphologyEx` 函数即可完成形态学开操作。

最终的掩模创建完成后，调用 cv2 中的 `bitwise_and` 函数即可对图像进行掩模处理，提取出所需的植物部分。

4 特征工程

4.1 HOG 特征

方向梯度直方图（Histogram of Oriented Gradient, HOG）特征是一种在计算机视觉和图像处理中用来进行物体检测的特征描述子。它通过计算和统计图像局部区域的梯度方向直方图来构成特征。HOG 特征的主要思想是：在一副图像中，局部目标的表象和形状能够被梯度或边缘的方向密度分布很好地描述。由于 HOG 是在图像的局部方格单元上操作，所以它对图像几何的和光学的形变都能保持很好的不变性。

HOG 特征提取方法的主要过程为：

- 1) 图像灰度化；
- 2) 采用 Gamma 校正法对输入图像进行颜色空间的标准化（归一化）；目的是调节图像的对比度，降低图像局部的阴影和光照变化所造成的影响，同时可以抑制噪音的干扰；
- 3) 计算图像每个像素的梯度（包括大小和方向）；主要是为了捕获轮廓信息，同时进一步弱化光照的干扰。
- 4) 将图像划分成小 cells（例如 6*6 像素/cell）；
- 5) 统计每个 cell 的梯度直方图（不同梯度的个数），即可形成每个 cell 的 descriptor；
- 6) 每几个 cell 组成一个 block（例如 3*3 个 cell/block），一个 block 内所有 cell 的特征 descriptor 串联起来便得到该 block 的 HOG 特征 descriptor。
- 7) 将图像 image 内的所有 block 的 HOG 特征 descriptor 串联起来就可以得到该 image 的 HOG 特征 descriptor，即最终可供分类使用的特征向量。

在本实验中，对图像进行掩模处理后，先采用 cv2 中的 resize 函数统一输入图像的大小，使 HOG 特征个数相同。接着采用 cv2 中的 cvtColor 函数将图像灰度化，然后给定相应的参数，声明 HOGDescriptor 对象 hog，采用 hog.compute 函数即可计算 HOG 特征，最后利用 flatten 函数将 HOG 特征数据变成一维的。

对数据集提取出特征后，由于特征间可能存在的多重共线性使特征冗余，故需要进行 PCA 降维以减少特征属性的个数，确保特征属性之间相互独立，减小算法的开销。

设有 m 条 n 维数据，则 PCA 降维的主要过程如下：

- 1) 将原始数据按列组成 n 行 m 列矩阵 X
- 2) 将 X 的每一行（代表一个属性字段）进行零均值化，即减去这一行的均值
- 3) 求出协方差矩阵
- 4) 求出协方差矩阵的特征值及对应的特征向量 r
- 5) 将特征向量按对应特征值大小从上到下按行排列成矩阵，取前 k 行组成矩阵 P ，即为降维到 k 维后的数据

这里从 `sklearn` 库中导入有关 PCA 降维有关的方法直接调用即可实现对 HOG 特征的降维。

4.2 颜色特征

图像的颜色特征是一种全局特征，描述了图像或图像区域所对应的景物的表面性质。一般颜色特征是基于像素点的特征，此时所有属于图像或图像区域的像素都有各自的贡献。

这里的颜色特征利用 `cv2` 库中的 `split` 函数进行 RGB 通道的拆分。分别利用 `numpy` 库中对应的函数计算 R、G、B 的均值、标准差和三阶颜色矩，最后利用 `log1p` 函数对数据进行平滑处理。以最终得到的结果作为颜色特征。

其中三阶颜色矩的公式为：

$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{i,j} - \mu_i)^3 \right)^{\frac{1}{3}} \quad (4)$$

4.3 SIFT

SIFT，即尺度不变特征变换（Scale-invariant feature transform, SIFT），是用于图像处理领域的一种描述。这种描述具有尺度不变性，可在图像中检测出关键点，是一种局部特征描述子。SIFT 特征基于物体上的一些局部外观的兴趣点，与影像的大小和旋转无关，对于光线、噪声、微视角改变的容忍度也相当高。

SIFT 特征检测主要包括以下 4 个基本步骤：

1) 尺度空间极值检测：搜索所有尺度上的图像位置。通过高斯微分函数来识别潜在的对于尺度和旋转不变的兴趣点。

2) 关键点定位：在每个候选的位置上，通过一个拟合精细的模型来确定位置和尺度。关键点的选择依据于它们的稳定程度。

3) 方向确定:基于图像局部的梯度方向,分配给每个关键点位置一个或多个方向。所有后面的对图像数据的操作都相对于关键点的方向、尺度和位置进行变换,从而提供对于这些变换的不变性。

4) 关键点描述:在每个关键点周围的邻域内,在选定的尺度上测量图像局部的梯度。这些梯度被变换成一种表示,这种表示允许比较大的局部形状的变形和光照变化。

在本实验中,调用 `cv2` 库中的 `SIFT_create` 函数实例化 `sift` 对象,继而调用 `sift.detect` 函数可以找出关键点,最后通过 `sift.compute` 函数计算关键点对应的 `sift` 特征向量。

5 模型训练

5.1 模型选择

5.1.1 词袋模型

将所有词语装进一个袋子里,不考虑其词法和语序的问题,即每个词语都是独立的。词袋模型被广泛应用在文件分类,词出现的频率可以用来当作训练分类器的特征。在本实验中,把图像的特征当做单词,把图像“文字化”之后,有助于大规模的图像检索。

在本实验中,首先进行 SIFT 特征提取,提取出特征数据;接着把处理好的特征数据全部合并,利用聚类把特征词分为若干类,此若干类的数目由自己设定,每一类相当于一个视觉词汇;最后利用视觉词袋量化图像特征,每一张图像由很多视觉词汇组成,利用统计的词频直方图,可以表示图像属于哪一类。

5.1.2 K-means

K-means 算法是非常经典的聚类算法,算法的思想很简单,对于给定的样本集,按照样本之间的距离大小,将样本集划分为 K 个簇。让簇内的点尽量紧密的连在一起,而让簇间的距离尽量的大。算法的基本流程如下:

- 1) (随机) 选择 K 个聚类的初始中心;

2) 对任意一个样本点, 求其到 K 个聚类中心的距离, 将样本点归类到距离最小的中心的聚类, 如此迭代 n 次;

3) 每次迭代过程中, 利用均值等方法更新各个聚类的中心点(质心);

4) 对 K 个聚类中心, 利用 2, 3 步迭代更新后, 如果位置点变化很小(可以设置阈值), 则认为达到稳定状态, 迭代结束, 对不同的聚类块和聚类中心可选择不同的颜色标注。

在本实验中, 该聚类思想被用在上述词袋模型中对特征数据进行聚类, 类别的数目即聚类中心的数目。由词袋模型结合图像的特征数据及聚类算法进行分类的思想在 `cv2` 库中已有 `BOWKMeansTrainer` 类将其实现, 借助该工具及上文提取的 SIFT 特征可以实现分类过程。实际使用中, 还需借助 `cv2` 库中的 `FlannBasedMatcher` 和 `BOWImgDescriptorExtractor` 进行特征点的计算与匹配, 再利用 `setVocabulary` 函数完成词汇表的创建。

5.1.3 支持向量机

支持向量机(svm)是一组用于分类、回归和异常值检测的监督学习方法, 其基本模型定义为特征空间上的间隔最大的线性分类器。

支持向量机的优点有: 在高维空间中有效; 在维数大于样本数的情况下仍然有效; 在决策函数中使用训练点的子集(称为支持向量), 因此它也是高效的内存; 多功能: 可以为决策函数指定不同的内核函数。

在本实验中, 采用 `sklearn` 库中已有的 `svc` 模块, 给定输入特征和 `svc` 核函数的类型等参数拟合模型。

5.2 模型训练

模型的输入对图像的 HOG 特征、颜色特征和由 SIFT 特征形成的词汇表进行了综合, 作为训练数据的属性。选用支持向量机模型进行多分类。

采用交叉验证的方式, 训练后得到的模型在验证集上进行验证, 根据得分调整参数, 参数包括提取特征时的有关参数和支持向量机模型的有关参数。其中交叉验证利用了 `sklearn` 库中的 `shuffle` 和 `train_test_split` 模块, 对训练集顺序进行随机打乱和训练集与验证集的划分, 提高模型的泛化性能, 避免过拟合。

6 特征与模型的选择和超参数的调整

6.1 特征的选择

因为对于本次的实验，植物的图片有多种的特征可以选择。我在实验开始阶段先对图像的绿色部分经过掩膜处理、屏蔽了环境背景的影响并提取出叶片部分后，先获取了相对比较简单的叶片的形状的特征，即叶片部分的周长、面积、周长面积比，使用一个简单的 SVM 模型来训练，经过简单的调参尝试后，准确率只能达到 36%左右。

第一个特征因为只是图片中的叶片的大小形状的特征，因此非常容易受到拍摄角度、拍摄大小和叶片大小的影响，准确率比较低也是在情理之中。不过第一个特征作为第一次的尝试使我们对于本次实验的总体的过程：特征的提取和选择、模型的训练、参数的简单调整、对待预测图片进行预测、保存预测结果并提交获取分数的步骤有了初步的了解。同时，还可以在后续的尝试中感受到我们可以从更加高级的特征和更加复杂的模型中获取多大的收益。

在第一个简单的特征尝试过后，我们小组成员分别尝试开始提取更多的特征。我们分别尝试了 HOG 特征、颜色特征、LBP 特征和 sitf + bag of words 特征：

6.1.1、HOG 特征。

我首先尝试的是手动提取 HOG 特征，仅使用了 `cv2.Sobel()` 函数来计算图片的方向梯度，然后自己手动对划分的各个 cell、各个 block 的梯度特征进行整合来获取整个图片的特征向量。然后开始尝试使用多个模型来训练。在尝试了使用 KNN、SVM、XGboost 三个模型后，KNN 训练出的模型在测试集准确率只能达到 40%左右，而 SVM 和 XGBoost 训练出的模型可以达到 50%左右。SVM 和 XGBoost 的效果要好于 KNN。

但是，我认为 50%的准确率完全低于我的预期。我认为 HOG 特征应该有更好的效果。于是我尝试了使用 opencv 的库中的提取 HOG 特征的函数 `cv2.HOGDescriptor()` 来提取，相对于我手写的提取 HOG 的函数，opencv 库中的可以提取出更多方向的梯度，并且可以更加方便地划分 win、block、cell 的大小和步长。再次训练模型后可以达到 65%左右的准确率。

6.1.2、LBP 特征。

我们使用了 `skimage` 中的 `local_binary_pattern` 函数来提取 LBP 特征，但是它并不能直接提取出特征向量，而是获得了处理过的图片。我们又提取了经过 LBP 特征处理后的图片的颜色直方图。经过训练后准确率只能达到 40% 多。

6.1.3、sift+bag of words 特征。

我们使用了 `cv2` 库中的 `SIFT_create` 函数实例化 `sift` 对象，继而调用 `sift.detect` 函数可以找出关键点，最后通过 `sift.compute` 函数计算关键点对应的 `sift` 特征向量。但是不同图片的特征向量的维度都是不同的，然后我们又使用了 bag of words 模型，把处理好的特征数据全部合并，利用聚类把特征词分为若干类，此若干类的数目由自己设定，每一类相当于一个视觉词汇；最后利用视觉词袋量化图像特征，每一张图像由很多视觉词汇组成，利用统计的词频直方图，可以表示图像属于哪一类。最终在 SVM 上面训练，准确率可以达到 76%。

6.1.4、颜色特征。

我们提取了图片的颜色分布直方图，但是训练后的模型的准确率很差。然后我们便尝试了分离了图片的 RGB 通道，再分别计算图片的一阶矩、二阶矩、三阶矩，由于数值较大我们还对其做了取 \log 的处理。经过训练后准确率可以达到 70%

以上是我们的全部尝试的特征及其单个特征模型的训练效果。

6.2 特征的融合

在考虑了所有单一特征模型的效果后，我们选择了效果最好的三个：HOG 特征、颜色特征和 sift+bag of words 特征进行特征融合。

首先我们尝试了一个简单的多数投票的模型融合的方法。对于 HOG、颜色、sift 三个单一特征的模型进行投票，得到最终的预测结果。该模型的准确率在 78% 左右。

接下来我们开始尝试对特征融合。因为颜色特征和 sift 特征的特征向量的维度比较小，分别为 9 个特征和 100 个特征，因此我们先将这两个进行了特征融合并看一下效果如何。我们直接将颜色特征向量和 sift 特征向量连接到了一起进行了训练。最终最好的效果是在 svm 上训练得到的，准确率达到 84%。

然后将 HOG 特征也融入进去，但是因为 HOG 特征的效果相对来说要弱于颜色特征和 sift 特征，并且 HOG 的特征向量比较大，有 4000 维左右。如果直接融合的话 HOG 特征会占主导效果，会导致过拟合并且影响 sift 特征和颜色特征的效果。因此我们对 HOG 特征进行了 PCA 降维，把特征维数降为 20 维左右，然后将三个特征的特征向量拼接到一起再训练，训练结果的准确率可以达到 87%。

6.3 超参数的调整

然后我们进行了最终模型的超参数的调整。此时我们需要调整的超参数分别有：HOG 特征的 win、block、cell 窗口的大小和步长、bag of words 词袋模型的词向量的个数、最终模型的选择。

对于 HOG 特征参数，因为我们需要对 HOG 特征进行降维到 20 维，因此我在提取 HOG 特征的时候就尽量地使得特征向量维度比较小。首先我尝试了图片的尺寸设为了 (128, 128)，并将 win 窗口的尺寸设为了和图片相同 (128, 128)，而 block 的尺寸为 (32, 32)，block 的步长为 (16, 16)，但是训练得到的特征向量比较多，效果较差。接着我将图片的尺寸设为了 (64, 64)，并将 win 窗口的尺寸设为了和图片相同 (64, 64)，然后不断尝试调整 block 的尺寸和步长。在调参的过程中，我发现如果增大 block 的步长和尺寸，特征向量的维度较少但是准确率会下降。如果调小 block 的步长，会提取出更加细小部分的特征，使准确率有所提升，但是很容易导致过拟合，并且过多的特征在降维后效果会下降很多。最终 block 的最佳尺寸为 (32, 32)，步长为 (16, 16)。兼顾了准确率和特征的维数不会过多，这样在融合模型中的效果也是相对较好的。

在 sift 特征中，需要调整的参数是词袋词向量的多少。太小的话无法保存足够的特征向量，会导致准确率较低，但是太多的话有可能会过拟合，并且过多的话会导致提取和训练的过程很慢。我在尝试把大小由 50 调整为 80 后，准确率有了较大的提升，再调整到 100 后，准确率提升较少，但是训练的时间增加了很多。考虑到再增大也很难有更高的提升效果并且会有更加漫长的提取时间，于是我们最终的词袋的大小为 100。

最终模型的选择，我们尝试了 SGD、XGBoost、SVM，在第一轮的尝试中，线性核的 SVM 和 XGboost 效果较好，并且线性核的 SVM 准确率要略高于 XGBoost。线性核的 SVM 模型准确率可以达到 87%，而 XGBoost 则在 86%。

接下来我们尝试了 rbf 和 poly 核的 SVM，需要调整的参数为惩罚系数 C，而 gamma 系数我们设置为 auto 即可以获得不错的效果。惩罚系数 C 越大，模型收敛速度越快且准确率越高，但是过大的惩罚系数会导致过拟合。相对于线性核，rbf 核和 poly 核的 svm 更不容易过拟合。在线性核的 svm 中，我把惩罚系数设

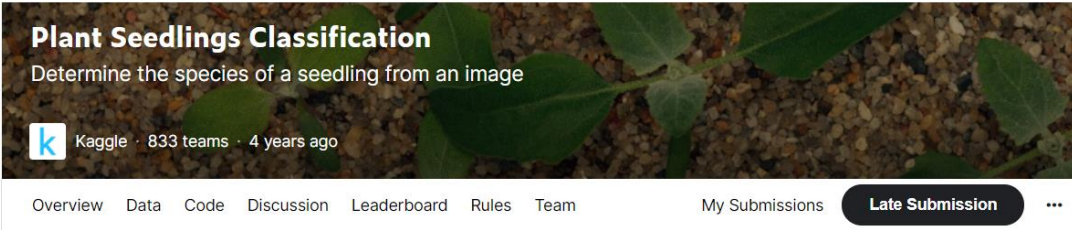
为 $C=50$ 左右效果已经是最好，再增大就会过拟合使得测试集准确率下降。但是在 `rbg` 和 `poly` 核中，经过多次的调整，我们的惩罚系数值分别为 $C=700$ 和 $C=800$ ，模型的拟合程度较好，`rbf` 核的 SVM 最好可以在 kaggle 获得 90% 的准确率，而 `poly` 核可以获得 89% 的准确率。

至此，我们的最终特征为降维为 16 维后的 HOG 特征、sift+bag of words 的 100 维的特征、9 维的颜色特征，总的特征向量共 125 维，使用的模型为 `rbf` 核的 SVM，最终的模型的预测结果提交到 kaggle 平台可以获得 0.9 的分数。

7 模型评价与分析

7.1 结果分析

模型训练完成后，在测试集上使用模型进行预测，将结果上传至 kaggle 平台，得到的分数如下图所示



Plant Seedlings Classification
Determine the species of a seedling from an image

Kaggle · 833 teams · 4 years ago

Overview Data Code Discussion Leaderboard Rules Team My Submissions **Late Submission** ...

Your most recent submission

Name	Submitted	Wait time	Execution time	Score
siftcolorHOGSVM.csv	just now	1 seconds	0 seconds	0.90428

Complete

[Jump to your position on the leaderboard](#)

F1-Score 得分为 0.904，说明模型的性能较好。

7.2 合理性与创新性

7.2.1 模型合理性

特征工程方面，选取图片数据处理中较常用的 HOG 特征、颜色特征和 SIFT 特征，且将这些特征进行了综合，能够充分利用图像的信息，减小单一特征下的误差；除此之外，HOG 特征进行了降维处理，缓解了维度过高及特征线性相关的

问题，削弱了噪声的影响。模型训练方面，主要采用了支持向量机模型，模型的理论性好，且分类的准确率高，适应性强。

7.2.2 模型创新性

在读取数据方面，将数据 dump 保存，使下次不需要重新读取，节省了时间开销。模型选择方面将在 NLP 领域广泛用于文本分类的词袋模型应用到图像分类领域，巧妙地利用图像的特征数据建立词汇表，从而采取类似的方法完成分类。

7.3 模型准确性

根据模型在测试集上运行的结果可见模型的准确性较高，但模型也有不足之处有待改进。

比如特征工程方面，由于颜色对图像或图像区域的方向、大小等变化不敏感，所以颜色特征不能很好地捕捉图像中对象的局部特征。除此之外，整个过程的计算较多，花费时间相对较长。因此，可以综合采用其他的图像处理技术，优化算法的执行过程，以获得更好的效果。

8 收获与体会

通过这次实验，我对机器学习完整的过程有了更深入的了解，我意识到机器学习问题除了模型的选择与训练过程，特征工程也同样重要。训练的数据并不是一开始就可以输入到模型直接使用的，就以这次的图片数据为例，我们还需要对图片进行适当的处理，并利用图像处理技术进行特征的选择与提取，最后找到合适的模型进行训练。除此之外，当模型复杂时，调参的过程也变得繁杂起来，所有的组成部分都可能对最终的结果产生影响，需要不断地分析与尝试