

无监督学习

ML09



礼欣

www.python123.org



基于聚类的“图像分割”实例编写

图像分割

图像分割：利用图像的灰度、颜色、纹理、形状等特征，把图像分成若干个互不重叠的区域，并使这些特征在同一区域内呈现相似性，在不同的区域之间存在明显的差异性。然后就可以将分割的图像中具有独特性质的区域提取出来用于不同的研究。

图像分割技术已在实际生活中得到广泛的应用。例如：在机车检验领域，可以应用到轮毂裂纹图像的分割，及时发现裂纹，保证行车安全；在生物医学工程方面，对肝脏CT图像进行分割，为临床治疗和病理学研究提供帮助。

图像分割

图像分割常用方法：

1. 阈值分割：对图像灰度值进行度量，设置不同类别的阈值，达到分割的目的。
2. 边缘分割：对图像边缘进行检测，即检测图像中灰度值发生跳变的地方，则为一块区域的边缘。
3. 直方图法：对图像的颜色建立直方图，而直方图的波峰波谷能够表示一块区域的颜色值的范围，来达到分割的目的。
4. 特定理论：基于聚类分析、小波变换等理论完成图像分割。

实例描述

目标：利用K-means聚类算法对图像像素点颜色进行聚类实现简单的图像分割

输出：同一聚类中的点使用相同颜色标记，不同聚类颜色不同

技术路线：`sklearn.cluster.KMeans`

实例数据

本实例中的数据可以是任意大小的图片，为了使效果更佳直观，可以采用区分度比较明显的图片。





“kmeans实现图片分割” 实例编写

实验过程

使用算法：Kmeans

实现步骤：

1. 建立工程并导入sklearn包
2. 加载图片并进行预处理
3. 加载Kmeans聚类算法
4. 对像素点进行聚类并输出

关于一些相关包的介绍：

- **PIL包**：因为本实验涉及图像的加载和创建，因此需要使用到PIL包，如未安装，请下载相关包自行安装。

实现步骤——1.建立工程并导入sklearn包

- 创建Kmeans.py文件
- 导入sklearn相关包
 - `import numpy as np`
 - `import PIL.Image as image` #加载PIL包，用于加载创建图片
 - `from sklearn.cluster import KMeans` #加载Kmeans算法

实现步骤——2.加载图片并进行预处理

- 加载训练数据

```
def loadData(filePath):  
    f = open(filePath, 'rb') #以二进制形式打开文件  
    data = []  
    img = image.open(f) #以列表形式返回图片像素值  
    m, n = img.size #获得图片的大小  
    for i in range(m): #将每个像素点RGB颜色处理到0-1  
        for j in range(n): #范围内并存放进data  
            x, y, z = img.getpixel((i, j))  
            data.append([x/256.0, y/256.0, z/256.0])  
    f.close()  
    return np.mat(data), m, n #以矩阵形式返回data，以及图片大小  
  
imgData, row, col = loadData('kmeans/bull.jpg') #加载数据
```

实现步骤——3.加载Kmeans聚类算法

- 加载Kmeans聚类算法
 - `km = KMeans(n_clusters=3)`
 - 其中`n_clusters`属性指定了聚类中心的个数为3

实现步骤——4：对像素点进行聚类并输出

- 依据聚类中心，对属于同一聚类的点使用同样的颜色进行标记

```
#聚类获得每个像素所属的类别
label = km.fit_predict(imgData)
label = label.reshape([row,col])
#创建一张新的灰度图保存聚类后的结果
pic_new = image.new("L", (row, col))
#根据所属类别向图片中添加灰度值
for i in range(row):
    for j in range(col):
        pic_new.putpixel((i,j),256 / (label[i][j]+1))
#以JPEG格式保存图片
pic_new.save("result-bull-4.jpg", "JPEG")
```

结果展示



原图



K=4



原图



K=3

实验分析

通过设置不同的 k 值，能够得到不同的聚类结果。同时， k 值的不确定也是Kmeans算法的一个缺点。往往为了达到好的实验结果，需要进行多次尝试才能够选取最优的 k 值。而像层次聚类的算法，就无需指定 k 值，只要给定限制条件，就能自动地得到类别数 k 。