

基于 R 树的图像检索技术研究

周伯威

2012012221

Class 21

School of Software,

Tsinghua Univ.

zbwww@gmail.com

王华清

2012013317

Class 22

School of Software,

Tsinghua Univ.

paddywhq@163.com

种璐瑶

2012013333

Class 22

School of Software,

Tsinghua Univ.

lingboxunmeng@163.com

摘要

图像内容识别与检索是当今计算机科学的重要研究领域之一。为了实现高效的检索,常常从一张图片中提取能够代表图片的特征向量,在图片数据集的众多特征向量中寻找与之距离相近的向量即完成检索。本文将探讨如何从给定图片中提取特征向量,并使用 R 树结构实现高效的索引。

关键词

图像特征 颜色直方图 HSV 颜色模型 基于颜色特征的图像检索 R 树

1. 介绍

目前图像的检索主要有基于颜色特征、纹理特征、内容特征三种方式。前两种具有计算快速、易于实现的优点,但丢失了图像的空间特征。基于内容的图像检索能够很好地检索具有明显空间特征的图像,但不易实现。本文中的方法将主要采取第一种,即提取图像的颜色特征,并尝试提取了图像的内容特征作为补充。

检索图像时,利用 R 树存储提取出的特征向量可以提高查找效率。本文给出了实现在 R 树的基础上实现 K 近邻查找的一种方法,并将其应用于图像的检索。

2. 图像特征提取

2.1 图像的颜色特征

2.1.1 HSV 颜色模型

计算机中常使用 RGB 颜色模型来存储图像的颜色信息,用数值表示各个通道中颜色数量,将 R、G、B 三个通道等分为 256 份,各用一字节存储。其优点是颜色分布均匀,可以用较少的空间存储较多的颜色信息。但是 RGB 颜色模型不能很好的与人的主观感觉对应,其主要表现为相距不远的两个 RGB 颜色在人眼中可能差别很大。对于图像的识别与检索,常常采用一种更好的颜色模型——HSV 颜色模型。

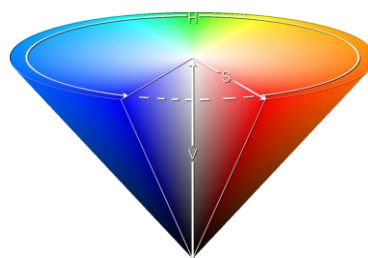
RGB 与 HSV 空间的转换方式如下:

$$h = \begin{cases} 0^\circ & \text{if } \max = \min \\ 60^\circ \times \frac{g-b}{\max-\min} + 0^\circ, & \text{if } \max = r \text{ and } g \geq b \\ 60^\circ \times \frac{g-b}{\max-\min} + 360^\circ, & \text{if } \max = r \text{ and } g < b \\ 60^\circ \times \frac{b-r}{\max-\min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max-\min} + 240^\circ, & \text{if } \max = b \end{cases}$$
$$s = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max-\min}{\max} = 1 - \frac{\min}{\max}, & \text{otherwise} \end{cases}$$

$v = \max$

其中, max 与 min 分别代表 r、g、b 中的最大、最小值。

HSV 颜色空间常用一个圆锥体表示[1]:



将 H、S、V 的柱坐标转换为直角坐标[2]后便可准确地使用欧氏距离度量两个颜色间的差异。

2.1.2 颜色直方图

图像颜色特征提取方法主要有颜色直方图法、颜色矩法、主颜色法等。本文所采用的颜色直方图法是一种简单并有效的方法。其实现如下:

根据需要将 H、S、V 三个维度各自等分为若干份,逐一扫描一张图像中的各个像素点,统计各个维度中出现的像素个数,最终的结果将作为特征向量。

2.1.3 颜色直方图的改进

针对颜色直方图法中可能出现的问题，我们在最终的程序中设置了若干条件编译开关，通过控制相关参数来获得更好的提取结果。优化方式有以下几点：

- a. 原图片中可能包含噪声或是其他干扰因素，而对于人眼，这些干扰因素会被忽略，因此将原图片进行小尺度的高斯模糊(高斯模糊核取 $5\text{px} \times 5\text{px}$)，以过滤掉图像中干扰颜色分类的噪声像素。
- b. 因图像的主体常在中央区域，将图像在空间上分为中央与边缘区域，在计算像素数时予以不同的权重。

进行模糊、分块后的图像示意图如下：



2.2 图像的内容特征

本文亦尝试了对图像内容特征的提取作为补充。主要思路是识别其两种主体颜色并将图像二值化，再根据二值化后的图像提取颜色的空间特征。其步骤如下：

将图像做高斯模糊处理，消减噪声。以 $24 \times 20 \times 20$ 的维度等分 HSV 空间，统计各个维度出现的像素数，即颜色直方图。对上述颜色直方图进行类似高斯模糊的处理，将各个维度数值“平滑化”，使某一维度的数值不仅与自身有关，也与邻近维度的像素数有关。找到所有维度中数值最高者，即众数，用来表示图像的第一特征色。将第一特征色附近一定区域的维度数值置零，再次统计众数，用来表示图像的第二特征色。逐点扫描图像，依据每个像素与两特征色在上述 HSV 直角坐标下的距离，将图像二值化。统计图像两个区域中的颜色，求其在 HSV 直角坐标下的均值，将之作为新的特征色。对新的特征色将图像重新二值化。对二值化后的图像的两个区域分别求若干统计量，如方差、空间离散程度等。计算两部分接近中央的程度，分别定义中央特征色与边缘特征色。

下图为特征色抽取结果：



本方法能够有效地表示主次特征明显的图像，对主次不明显的图像效果一般。

2.3 不同特征的查找准确率

对于不同的特征，查找数据量 5613 张图准确率结果如下：

| 特征 | 准确率 |
|----------------|----------|
| 颜色直方图 9 维 (提供) | 27.1513% |
| HSV20 维 | 33.1676% |
| HSV9 维 | 26.6132% |
| HSV75 维 | 32.3196% |
| RGB9 维 | 24.9011% |

从中可以看出，维数相同时，HSV 提取特征准确率最高，RGB 相对较低。而同样为 HSV，提高维度能适当提高准确率。但维度过高时，准确率与提取的特征有一定关系，并不是越高越好。

3. R 树

3.1 利用 R 树存储图像特征向量

R 树多维的存储结构可以很好地解决图像特征向量的存储。对于一组 N 维的特征向量，可以建一颗 N 维的 R 树，R 树的每一维对应向量的每一维。这样对于每个图像特征向量，可以将其看做一个 N 维空间中的点。而找与目标图像相近的 K 张图像问题可以转化成 N 维空间中的 K 近邻查找问题。

3.2 基于 R 树的 K 近邻查找

考虑到 R 树的 MBR 可能重叠，所以在进行 K 近邻查找时必须搜索每一个可能的结点。因此采用深度优先搜索的方式较为合理。为了提高搜索效率，可以利用 R 树存储的特性作如下的处理：

建立一个空集 S，用于存储已经找到的点。搜索时，首先直接将深度优先搜索到的前 K 个点存入 S 中。进行后面的搜索时，对于每个节点，若其对应的 MBR 与目标点的最小距离大于 S 中点与目标点的最大距离，则该 MBR 中的任何点均不为 K 近邻点，所以可以对该节点做剪枝优化。这样就提升了搜索效率。

我们可以把点的情况推广到矩形区域。定义两个矩形区域的最小距离为两个区域中最近两点的距离，最大距离为最远两

点的距离。则对于每个节点，若其对应的 MBR 与目标矩形的最小距离大于 S 中矩形与目标矩形的最小距离的最大值，则该 MBR 中的任何点均不为 K 近邻矩形，所以可以对该节点做剪枝优化。

3.3 计算两个矩形区域的最小 L2 距离

计算 A 与 B 的最小 L2 距离时可以用如下方法：对于每一维，计算 A 与 B 在该维中最小距离差。A 与 B 的最小 L2 距离即为这些差的平方和再开根号。

3.4 计算两个矩形区域的最大 L2 距离

计算 A 与 B 的最大 L2 距离时可以用如下方法：对于每一维，计算 A 与 B 在该维中最大距离差。A 与 B 的最大 L2 距离即为这些差的平方和再开根号。

3.5 维度和数据量对 R 树性能的影响

通过更改维度（4、8、12、16、20）和数据量（1000、2000、3000、4000、5000），研究进行 10 近邻查找时对 R 树的结点访问次数如下：

表1. 不同维度和数据量下 10 近邻查找访问 R 树结点次数

| | 1000 | 2000 | 3000 | 4000 | 5000 |
|----|---------|---------|---------|---------|---------|
| 4 | 242.784 | 402.777 | 464.807 | 472.006 | 530.069 |
| 8 | 398.537 | 569.591 | 638.802 | 713.029 | 807.990 |
| 12 | 551.050 | 882.673 | 1023.94 | 1168.25 | 1347.40 |
| 16 | 713.607 | 1179.92 | 1455.48 | 1703.84 | 2005.46 |
| 20 | 865.072 | 1491.89 | 1948.23 | 2365.82 | 2775.32 |

我们可以看出：当数据量一定时，维度越高，访问 R 树结点次数越多；当维度一定时，数据量越大，访问 R 树结点次数越多。

3.6 讨论对 R 树结点分裂次数的影响因素

R 树结点分裂出现在 R 树的插入操作中。影响 R 树的分裂次数因素主要为查找新插入结点位置的算法。考虑如下情况：若每次插入都在 R 树最左侧子树中，则分裂次数最大。这样导致 R 树其他部分始终为 R 树结点分支下界 m。

为了减少分裂次数，可以查找出 R 树空余较多的子树进行插入。但此操作会导致 R 树 MBR 的重叠率增加，进而影响 R 树的查询效率。故从查找效率出发，维持原分裂方法。

3.7 讨论对 R 树 MBR 重叠的影响因素

R 树 MBR 的重叠区域也与插入算法有关。插入一个数据时，由于每次找的分支均为最优，故能使 MBR 重叠区域较少。

而分离节点的算法也对 MBR 的重叠区域有较大影响。对于现采用的方法为：选取两个矩形使得它们在一个矩形中时浪费空间最多。再每次将并入时增加面积最小的块并入，直到有一块中数量达到结点分支下界 m+1。

暴力搜索复杂度过高不考虑。现在考虑选取的最开始两块矩形的选取，现用方法可能导致左上右下存在两块大矩形被并入同一结点。而在选取时，可以给出所有结点后寻求一条分割，从而解决该问题。

4. 参考文献及引用

[1] 图片引用自http://zh.wikipedia.org/wiki/File:HSV_cone.png
[2] 王涛,胡世民,孙家广.基于颜色-空间特征的图像检索.软件学报,Vol.13,No.10.