

数字图像

8 图像形态学

- 腐蚀 & 膨胀
 - 当目标图像不变，但所给的结构元素的形状发生变化时，或者结构元素的形状不变，而其原点位置改变时，膨胀运算的结果会改变
 - 对目标图像的膨胀运算，相当于对图像比较的腐蚀运算操作；对目标图像的腐蚀运算，相当于对图像背景的膨胀运算
- 开运算
 - 先腐蚀再膨胀
 - 使得物体的轮廓线更加光滑，磨光图像内边界
- 闭运算
 - 先膨胀再腐蚀
 - 磨光图像外边界
- 开闭运算互为对偶

7 灰度图像分割

- 阈值分割分类
 - 全局阈值
 - 局部阈值
 - 动态阈值
- 通过交互方式获得阈值
 - $i f|f(x, y) - f(x_0, y_0)| <= R \text{ set } 255, \text{ else set } 0$
- 通过直方图获得阈值
 - 基本思想
 - 边界上的点灰度值出现次数少
 - 极小点阈值
 - 通过寻找直方图的极小点确定分割阈值
 - 最优阈值
 - 减小分割误差
 - 通过背景和目标的灰度概率分布来确定最佳阈值
 - 取值方法
 - 取直方图谷底（最小值）的灰度值为阈值T
 - 缺点
 - 会受到噪声的干扰，导致最小值不是预期的阈值
 - 改进
 - 取两个峰值之间的某个固定位置（一般情况下，比选谷底更可靠）
 - 对直方图进行平滑处理
- 迭代阈值选择
 - 1. 开始选择一个初始估计值T
 - 2. 使用T分割图像，产生两组像素G1 & G2
 - 3. 计算G1和G2范围内的像素的平均值u1和u2
 - 4. 计算一个新的阈值T=1/2(u1+u2)
 - 重复2-4，直到T的变化比预先设定的小为止
- 最佳阈值分割
 - 使得误判概率最小
 - 若 $\sigma_b = \sigma_a, t = \frac{1}{2}(u_a + u_b) + \frac{\sigma^2}{u_b - u_a} \ln(\frac{1 - \theta}{\theta})$
 - 若 $\sigma_b = \sigma_a$, 且 $\theta = 1/2, t = 1/2(u_b + u_a)$
 - 若图像的目标物和背景像素灰度级呈正态分布
 - 且偏差相等
 - 像素总数也相等
 - 那么这个最佳分割阈值就是目标物和背景像素灰度级两个均值的平均
 - 对于复杂图像，如果仅用单一阈值不能给出良好的分割结果，解决办法是两个阈值或者多个阈值
- OTSU法
 - 使得类间方差最大
 - 简单，快速
 - 步骤
 - 1. 计算图像的归一化直方图pi
 - 2. 遍历所有灰度级作为阈值T，灰度级处于[0, T]的像素分为C0，处于[T + 1, L - 1]的像素分为C1
 - 3. 依次计算 $u_0, u_1, w_0, w_1, \sigma_B^2$
 - 记录下最大类间方差对应的T*
 - 5. 若不唯一，取多个T*的均值
 - 用T*作为最优阈值进行分割
 - 公式： $H = - \sum_i p(x_i) \log p(x_i)$
- 一维熵阈值分割
 - 即选取一个阈值，使得分出这两个部分的一阶灰度统计的信息量最大
 - 当熵函数最大时对应的灰度值即为最佳阈值
- 面向区域的图像分割
 - 目标
 - 将区域R划分为若干个子区域，使得它们满足
 - 1. 完备性 2. 连通性 3. 独立性 4. 单一性 5. 互斥性
 - 像素集合的区域生长
 - 区域的分裂与合并

9 彩色图像分割

- 分水岭算法
 - 任何灰度级图像都可以被看做是一个地形图
 - 当处在不同汇聚盆地中的水将要聚合在一起的时候，修建打吧将阻止聚合。最后得到的水坝边界就是分水岭的分割线
 - 步骤
 - 获取局部极小
 - 统计连通分量
 - 以最小的局部极小为基准提升高度
 - 如果出现某两个连通分量合并，则记录分界线
 - 逐步提升高度，直至整个图像合并一个连通分量
 - 记录的所有的分界线即为分割边界
 - 缺点
 - 由于噪声或者局部不规则而引起“过度分割”
 - 没有自顶向下的信息
 - 解决办法
 - 分水岭之前平滑去噪
 - 标记约束分水岭
 - 从先前已定好的区域开始浸水，防止过度分割
 - 分级分割
 - 通过分水岭算法，得到一张初始的分割图片；以这些相对高度为基础，再次使用分水岭算法
- Meanshift
 - 从聚类的角度理解
 - 用概率密度估计函数的极大值点来聚类
 - 假设
 - 不同类的数据点符合不同的概率密度分布
 - 局部密度最大值被认为是类心，收敛到相同局部最大值的点被认为是同一类
 - 迭代
 - 随机找一个起点，通过迭代的方式，一步步向概率密度极大值移动
 - Mean shift
 - 均值偏移
 - 通过计算窗口内的均值向量，更新爬山的方向；通过确定各类的中心，完成聚类
- Ncuts
 - 图论中，去掉所有能使一张网络流图不再连通的边集称为图的割
 - 图割
 - 如果一个割，它的边所有权值之和最小，那么这个割被称为最小割，也就是图割的结果
 - 分离度量
 - 最小割将得到一个最大分离度的划分，但是容易产生孤立点
 - 因为单个像素组成的点集往往与其他点集相邻边少，其权和也最小
 - 归一化图割
 - 看成图的谱聚类问题
 - 方法
 - 转换为对上述标准形式求特征值的问题：最小特征值是0
 - 用第二小的特征值及特征向量，对图进行划分
 - 判断是否还要继续细分下去（用递归）
 - 拉普拉斯矩阵
 - 公式： $L = D - W$
 - D是图的度矩阵（加权），为对角阵
 - 性质
 - 对称半正定矩阵
 - 最小特征值为0，对应的特征向量为1
 - 其余n个实特征值均为负
 - 优点
 - 能够有效集成图像像素颜色，纹理等特征信息及空间位置等信息
 - 具有良好的平衡划分性能，分割性能较好
 - 分割类别个数事先指定，利于场景划分
 - 缺点
 - 权值矩阵维数为图像中像素的个数，特征系数维数太高，无法处理大尺度图像
 - 计算复杂度高，不利于实时应用
 - Ncuts算法改进
 - 对每一个设定阈值T计算Ncut(A,B)最小的Ncut，对应的阈值T为最佳阈值
 - W矩阵基于图像灰度级构建，而不是像素数量，以减少单独大大减少计算时间
 - 先用Mean shift进行初始区域分割，再采用Ncut对区域节点进行分组，完成场景的划分
 - 集成Mean Shift算法和Ncuts算法

$$Minimize Ncut(A, B) = \frac{cut(A, B)}{asso(A, V)} + \frac{cut(A, B)}{asso(B, V)}$$