山东大学 计算机科学与技术 学院

计算机视觉 课程实验报告

|  |  |  |
| --- | --- | --- |
| 学号：201918130223 | 姓名： 刘庆典 |  |
| 实验题目：图像结构 | | |
| 实验过程中遇到和解决的问题：  （记录实验过程中遇到的问题，以及解决过程和实验结果。可以适当配以关键代码辅助说明，但不要大段贴代码。）  5.1连通域：  实现一个8连通的快速连通域算法，并基于该算法对测试图像horse\_mask进行处理：计算白色连通区域的个数；删除较小的白色连通区域，只保留最大的一个。八连通区域即是以某像素为中心的九宫格内的其他所有像素，如果两个像素区域的这八个方位有任一方位相连，这两个区域就是连通的，即视为同一个区域。这里我采取的是种子填充算法，即seedFilling，选取一个前景像素点作为种子，然后根据连通区域像素值相同的原则将与种子相邻的前景像素合并到一个像素集合中，最后得到的像素集合则为一个连通区域。对于这八个相邻的像素进行像素值判断，当为白色且未被标记过时将其压入栈，如下：    定义栈nearPixel存放与种子相邻的所有前景像素，mask数组存放每一个未知是否被标记过，若标记则置一，这样就避免重复扫描，加快了运算效率。  实现快速连通算法前需要将图像二值化，其中将阈值设为了127，如下：    关于删除面积较小的白色连通区域，首先需要找出图像中连通域面积最大的一个，所以先搜索图像的连通区域轮廓，遍历每一个连通区域，计算它们的面积 并实时比较得出一个最大值。    搜索完毕后就将最大值设为阈值以便之后删除。然后进行第二遍搜索，搜索图像的连通区轮廓，然后遍历每一个连通区域，计算它们的面积面积并基于阈值删除小于它的所有连通区域，删除即是将该连通区域内的所有像素置0。    5.2距离变换：了解opencv的距离变换函数distanceTransform，使用合适的图像进行测试，并将距离场可视化输出。distanceTransform函数用于计算二值图像中每一个非零点像素与其最近的零点像素之间的距离。输出的是保存每一个非零点与最近零点的距离信息，图像上越亮的点，代表了离零点的距离越远。该函数的原始图像应为二值图像，图像的二值化使得数据量大为减少，更能突显出目标的轮廓，所以需要将待处理图像先转化为灰度图像，再转化为二值图像再调用该函数。    CvtColor()函数负责将图像转换为灰度图，threshold()函数则负责将灰度图像转换为二值图， | | |
| 结果分析与体会：  5.1连通域：  使用二值化之后的图像进行连通域计算得到的白色连通域数量为：  若直接使用图像实现连通域算法得到的白色连通域数量为：  原图与二值化后的图像对比：      只保留最大的白色连通区域：    最大连通区域面积为：  5.2距离变换：  distanceTranform()函数中距离类型有如下几种：    本实验中选用的是DIST\_L2，即欧氏距离  thresh()函数中阈值类型有如下几种：    实验中我采用的是THRESH\_BINARY，即大于阈值的均设为最大值255，其余的均设为0  阈值设为85时：    阈值设为127时：    阈值设为180时： | | |