

CVDL第二次作业报告

李浩辰 1500012746

2018.5

1 简介

本次作业的目的是实现一种图形分割的算法，这里实现的算法是graph cut，在下一节会说明实现的方式。

2 实现

使用了python中的opencv模块，读入文件a.jpg，将其缩放为 512×512 像素，这里缩放的原因是为了尽快的看到结果，在图像大小上升的时候算法运行时间较长。之后由用户指定物体和背景像素（必须）。

之后按照网络流建图，源点向所有像素连边，如果指定为物体，则权值为inf，若为背景，则为0，否则为像素与物体颜色的相似度倒数。所有像素也向汇点连边，权值类似于源点连边。像素之间只在相邻四向的像素相连，由于距离都相同，所以权值也为颜色相似度的倒数。

建图之后运行dinic算法，求出最大流，按照图中满流的边将图片分为两部分，即为物体和背景。

实验结果表现为img.png，obj.png，bg.png。

3 使用方法

python3 main.py，使用鼠标左键刷物体，m键切换，再次使用鼠标左键选取背景。选取完成后按esc开始算法。可以在命令行观察处理进度。

4 思考与总结

算法存在一些已知的问题。首先，与其他的图像分割算法相比，graph cut获得的结果存在许多的噪点，这些噪点来源于最大流算法，根据边权设置的不同噪点的数量会有所差别。这个噪点的解决方法有两个，第一是小心的设置边权，调整参数，使得最大流的结果更接近预期。第二是对于处理完的图片进行其他操作，例如膨胀收缩等，直接处理可能的噪点。

另一个存在的问题是物体像素的选取和背景像素的选取。此处我的做法是取了物体和背景像素的平均值作为基准，在源汇点连边的时候相似度计算来源于这两个均值。此处可能有更好的做法，例如考虑该像素点和用户指定的物体像素的最近距离等因素，可以将其考虑在内。

此外，在物体的像素只取一个均值可能也并不是好的选择，例如在测试的过程之中，如果指定物体的时候不把花蕊指定进去，那么就会被划分为背景。可能在边权的部分需要更多考虑。