

HITSZ ASC22 校内练习题

Easton Man

November 2021

1 SLIC 算法简介

超像素算法就是将图像中的像素依据某种相似性进行聚类，形成一个大“像素”，这个大“像素”可以作为其他图像处理算法的基础。在众多的超像素算法中，比较常用的是 SLIC (simple linear iterative clustering)，它是 Radhakrishna Achanta 等人于 2010 年提出的一种简单高效的超像素算法，它的计算复杂度为 $O(N)$ ，其中 N 为图像像素点个数。

SLIC 算法的基本思想是，首先将图像从 RGB 颜色空间转换到 CIE-Lab 颜色空间，并把每个像素的 (L, a, b) 颜色值和 (x, y) 坐标值组成一个 5 维的特征向量 $V[L, a, b, x, y]$ ，然后，根据给定的网格步长 S ，初始化聚类中心 $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ ，之后在每个聚类中心 C_k 的邻域 ($2S \times 2S$)，计算邻域内各像素与该 C_k 点的相似性度量，从而对邻域内的像素点进行聚类，之后迭代更新聚类中心，直至满足收敛条件。

SLIC 算法的具体流程如下：

Algorithm 1 SLIC superpixel segmentation

/ Initialization */*

Initialize cluster centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ by sampling pixels at regular grid steps S .

Move cluster centers to the lowest gradient position in a 3×3 neighborhood.

Set label $l(i) = -1$ for each pixel i . Set distance $d(i) = \infty$ for each pixel i .

while $E > threshold$ **do**

/ Assignment */*

for each cluster center C_k **do**

for each pixel i **in a** $2S \times 2S$ **region around** C_k **do**

 Compute the distance D between C_k and i .

if $D < d(i)$ **then**

 set $d(i) = D$

 set $l(i) = k$

end

end

end

/ Update */*

 Compute new cluster centers.

 Compute residual error E .

end

2 题目说明

1. 源代码包括以下部分：

- main.cpp：入口和计时代码
- src/：SLIC 源代码
- include/：SLIC 头文件
- data/：数据文件
- Makefile：GNU Makefile

2. 程序运行方法：

- (a) 编译：make
- (b) 运行：make run

3. 评价指标：所有测试用例加速比的平均值

4. 修改范围：

- 允许修改 src/
 - 允许修改 include/
 - 允许新增文件
 - 允许修改 Makefile（包括编译选项和编译器）
 - 禁止修改 main.cpp 或任何计时代码
 - 禁止修改 data/中的测试样例
5. 可以修改算法的计算量，也可以修改计算顺序，但是优化的算法必须和原有算法等价，以通过验证为准
6. 正确性判断以基准验证文件 check.ppm 为准，差异点数为 0 则通过，其余不通过
7. 可以针对特定的数据进行优化，但必须保证所有情况下的正确性，后续可能加入更多不同的测试数据

3 结果要求

1. 最终源代码需要保证能够通过 make 和 make run 编译和运行，添加文件需要编写好对应的 Makefile
2. 除源代码外还需要编写一个优化记录，描述每处优化的思路 and 效果

4 参考文献

[SLIC Superpixels Compared to State-of-the-art Superpixel Methods](#)