

计算机图形学

上机作业七

刘国瑞 PB21000145 id: 64

2023 年 5 月 2 日

问题描述

实现 ARAP 算法，实现找到原像和目标像之间的过渡图像。需要对网格三角形变化的 jacobi 矩阵进行插值，并保证尽可能满足变换要求。bonus 为解决算法中的旋转一致性问题。

实验原理

ARAP

原像到目标图像时，网格三角形之间发生的是仿射变换，其系数矩阵满足：

$$A(x_1 - x_2) = y_1 - y_2; A(x_1 - x_3) = y_1 - y_3$$

带入 x, y 即可解出系数矩阵 A 。

为了对 jacobi 矩阵插值，先将 A 分解成旋转矩阵和放缩矩阵：

$$A = U\Sigma V^T = (UV^T)(V\Sigma V^T) = RQ$$

对 R 和 Q 插值得到 Rt 和 Qt ：

$$\arg Rt = t * \arg R; \quad Qt = (1 - t)I + tQ$$

再取 $At = RtQt$ 即可

而在过程中图像三角网格必须拼接在一起，不能完全满足 jacobi 矩阵 At 的变化，所以需要使用最小二乘法。

bonus：旋转一致性

解决旋转一致性问题，我们使用修改 ARAP 求出的旋转矩阵对应角度的方法。

1. 具体的思路是让每个相邻三角形的旋转角度尽可能一致，即 $-\pi < \theta_i - \theta_j < \pi$
2. 算法用了 find 函数和 sum 函数的高级使用方法，可以实现一行找到相邻三角形。步骤如下：
首先取 t 中第一个三角形，作为队列的第一个元素。找到与之相邻的三角形，对这些三角形都改变一下角度，即在差值大于 180 度的时候减去 2π ，小于就加上

3. 使用 BFS 算法（应该是这个，分不太清 DFS 和 BFS）：修改完后，将这些处理过的三角形标记，并排到队列中。每次处理队列中的第一个元素，同样进行上述步骤，将相邻三角形的旋转角进行操作。并将处理过的元素去掉。

实验结果

不解决旋转一致性得到的结果如下：

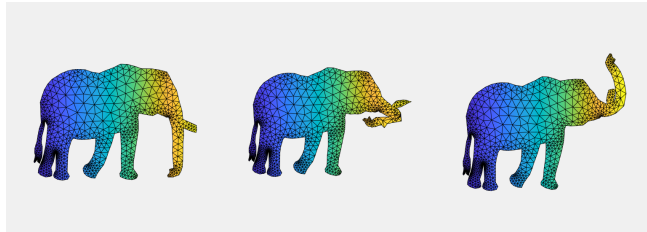


图 1.

解决了之后得到的结果：

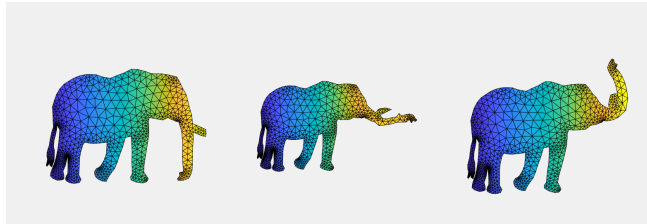


图 2.

结果分析

可见不解决一致性时象鼻会在中段扭曲，直接截断到了后面，处理了旋转一致性之后这个问题得到了较好的解决。

代码说明

bonus 部分见 `changetheta.m` 函数，算法核心即插值并求方程组得到结果见 `ARAP interp.m`

致谢

感谢助教的辛苦付出！