山东大学 计算机科学与技术 学院

数字图像处理 课程实验报告

|  |  |  |
| --- | --- | --- |
| 学号：201705130113 | 姓名：黄瑞哲 | 班级：计科17.3 |
| 实验题目：几何变换与变形 | | |
| 实验内容：  实验2.1：图像缩放  实现一个图像缩放函数，可以对输入图像进行任意倍数的缩放，要求：  1. 采用双线性插值进行重采样  2. X,Y方向的缩放倍数参函数参数的形式传入  3. 可以只考虑输入图像为3通道，8位深度的情况  4. 不能调用图像处理库的缩放函数来完成  实验2.2：图像变形  记[x’, y’]=f([x, y])为像素坐标的一个映射，实现f所表示的图像形变。其中f的逆映射为 | | |
| 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。）  1. 在第一个实验任务中，绘制的新图像的画布大小不对。仔细查阅文档后发现在设置size的时候行列写反了，导致新图像画布的尺寸有问题。  2. 在Lookup的时候系统会提示segment fault错误。检查发现由于原先在发现四个相邻的像素点是，xy坐标分别为下取整和下取整+1，导致在映射像素最后一个坐标点的时候，四个相邻的像素点中的右下角那个点超出了有效的像素点范围，因此在索引的时候发生了越界错误，导致segment fault。将发现四个相邻像素点的方法改成对xy坐标分别下取整和上取整后问题解决。 | | |
| 实验步骤和代码：  实验2.1：图像缩放  对于图像缩放来说我们课堂中学习到的有两种方式：第一种是将原图像中的每一个像素映射到新图像中，然后对于新图像中缺失的像素点通过插值来补全，这称为Projection；第二种是对于新图像中的每一个像素点求其在原图像中对应的点，而对应的点可能是非整数点，因此需要作插值求出其像素值来，这称为Lookup。  在该实验中，我采用的是Lookup的方式，这个方式相比于Projection来说，主要的差别在于lookup方便找到映射点的周围四个整数点，方便作差值运算。  该算法在实现的时候，首先根据X轴和Y轴的缩放比例计算出新图像的画布大小，也就是它的宽和高。然后对于图像中每一行每一列每一通道的像素点，计算该像素点在原图像中对应的坐标，根据这个坐标找到它上下左右四个相邻的整数像素点来，然后根据这四个整数点作双线性插值。  算法主要部分的代码如图所示    对于双线性插值来说又分为两种方式：  第一种是两次水平一次垂直，第二种是两次垂直一次水平    它们对应的代码分别为    因此只需要将这两段代码其中之一代替给缩放函数主循环中的bilinear即可。  实验2.2  实验任务中给出的变换公式坐标均为中心归一化坐标，在对我们的图像进行处理之前，需要将原有的图像坐标转换成中心归一化坐标。    将坐标中心归一化以后，我们开始考虑如何根据f的逆函数进行图像变形。  由于实验任务中给出的是f的逆函数，我们知道新图像坐标到原图像坐标的转换规则，因此我们借鉴图像缩放中Lookup的思想，即对于新图像中的每一个像素点，计算其在原图像中的坐标，然后在原图像中对该坐标进行二次插值运算，即可得到新图像该坐标的像素值来。 | | |
| 实验结果：  实验2.1结果，（左边为原图，右边为缩放，由于实验报告中图片大小比例问题，两次实验结果中的原图看起来大小可能不太一样，但保证结果真实准确）  缩小，系数均为0.6    放大，系数均为1.6    实验2.2结果，图像中部变得扭曲 | | |
| 实验总结：  在没有实验之前，我一直觉得Projection的做法可能更符合人们的直觉，但是在实验之后，我发现用代码实现Projection的时候，寻找插值所需要的四个像素点是非常困难的，而且如果不做线性插值的话会导致新图像中的某些像素点是空白的，这样生成的图片看起来就非常的奇怪，而反观Lookup的做法，它是对于新图像中的每一个点计算在原图像中的像素点，这样插值所需要的四个点是非常容易寻找的，分别将xy坐标上取整和下取整即可，而且这样可以保证新图像中的每一个像素点都有被计算过，不会出现空白的情况。  而且针对第二个实验任务，在已知映射函数f的逆函数的条件下，使用Lookup的思想也便于我们计算出新图像每一个像素点的值，而不需要根据逆函数求出这个映射函数的原函数来。 | | |