

# 数值分析第一次大作业 实验报告

2017011010 杜澍滢 自 71

## 需求分析

本次大作业包含必做和选做两项任务。

必做任务要求对给定图像（清华二校门）分别以最近邻、双线性和双三次插值方法实现旋转扭曲和畸变校正两种变换。其中旋转扭曲要求可以调整旋转角度和半径两个参数；畸变校正要求实现桶形和枕形两种畸变，同时可以调整半径。

选做任务需两张人脸图片，下面分别称为“待修改人脸”和“目标人脸”。分别给出两张脸上的 68 个关键点，要求利用 TPS 网格变形把待修改人脸的关键点特征变成目标人脸的特征。注意此处 TPS 变形后得到的只是一个坐标映射关系，仍需要通过插值得到最终的“修改后人脸”，这里可以利用必做部分的函数。

## 必做任务

### 方案设计

必做的几个关键步骤如下：

- （1）读入图片，以其中心为原点换算各像素点坐标
  - （2）选择变形方式（旋转扭曲或畸变校正）
  - （3）选择参数（旋转扭曲需选择旋转方向、角度、半径；畸变校正需选择桶形/枕形、半径）得到原图与变换后图片的坐标映射
  - （4）选择插值方法（最近邻/双线性/双三次）并重新换算坐标
- 算法核心为（3）（4），下面说明这两部分的原理。

## 方案基本原理

### 旋转扭曲

输入：新图像的整数型坐标(x',y')

输出：对应的原图像浮点型坐标(x,y)

1. 计算(x',y')到图像中心点的距离 Distance，选择最大旋转角度 $a_{max}$ 和扭曲旋转的半径 Radius;

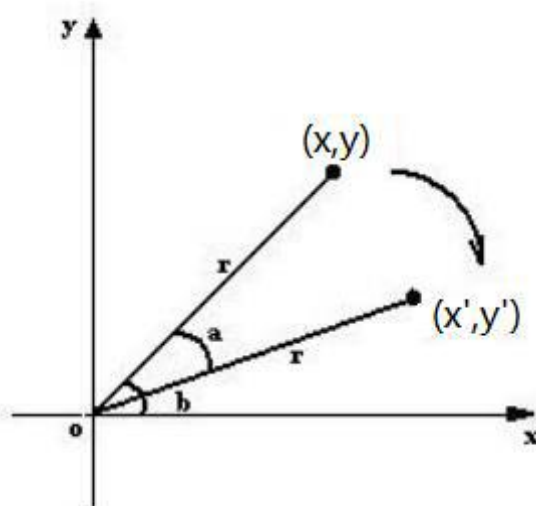
2. 坐标(x',y')旋转的角度为

$$a = a_{max} \times \frac{\text{Radius} - \text{Distance}}{\text{Radius}}$$

3. 计算浮点型坐标(x,y)，有

$$\begin{cases} x = x' \cdot \cos a - y' \cdot \sin a \\ y = x' \cdot \sin a + y' \cdot \cos a \end{cases}$$

4. 输出(x,y);



### 畸变校正

计算浮点坐标 (x, y)，对于桶形畸变有：

$$x = \left[ \frac{\text{Radius}}{\text{Distance}} \cdot \arcsin \left( \frac{\text{Distance}}{\text{Radius}} \right) \right] x'$$

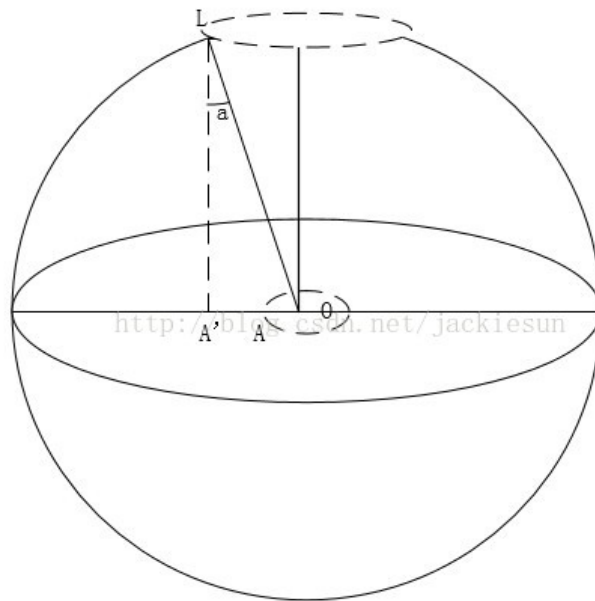
$$y = \left[ \frac{\text{Radius}}{\text{Distance}} \cdot \arcsin \left( \frac{\text{Distance}}{\text{Radius}} \right) \right] y'$$

对于枕形畸变有

$$x = \left\{ 1 / \left[ \frac{\text{Radius}}{\text{Distance}} \cdot \arcsin \left( \frac{\text{Distance}}{\text{Radius}} \right) \right] \right\} x'$$

$$y = \left\{ 1 / \left[ \frac{Radius}{Distance} \cdot \arcsin \left( \frac{Distance}{Radius} \right) \right] \right\} y'$$

输出 (x, y)



## 插值方法

### (1) 最近邻插值

找到原图中离变换后得到的浮点坐标最近的像素点并将其像素值赋给该浮点坐标对应的新图中的像素点。

### (2) 双线性插值

在原图中找到变换后得到的浮点坐标周围的四个像素点，根据以下公式计算出一像素值并将其赋给该浮点坐标对应的新图中的像素点。

$$f(x, y) = \begin{bmatrix} 1-u \\ u \end{bmatrix}^T \begin{bmatrix} f(i, j) & f(i, j+1) \\ f(i+1, j) & f(i+1, j+1) \end{bmatrix} \begin{bmatrix} 1-v \\ v \end{bmatrix}$$

### (3) 双三次插值

在原图中找到变换后得到的浮点坐标周围的十六个像素点，根据以下公式计算出一像素值并将其赋给该浮点坐标对应的新图中的像素点。

$$f(i+u, j+v) = ABC^T$$

$$A = \begin{bmatrix} S(u+1) & S(u) & S(u-1) & S(u-2) \end{bmatrix}$$

$$C = \begin{bmatrix} S(v+1) & S(v) & S(v-1) & S(v-2) \end{bmatrix}$$

$$B = f(i-1:i+2, j-1:j+2)$$

$$S(x) = \begin{cases} 1 - 2|x|^2 + |x|^3 & |x| \leq 1 \\ 4 - 8|x| + 5|x|^2 - |x|^3 & 1 < |x| < 2 \\ 0 & \text{otherwise} \end{cases}$$

## 功能展示



### 必做任务



变换类型 ☐ 旋转扭曲 ☒ 畸变校正

旋转方向 ☐ 顺时针 ☐ 逆时针

畸变类型 ☒ 桶形畸变 ☐ 枕形畸变

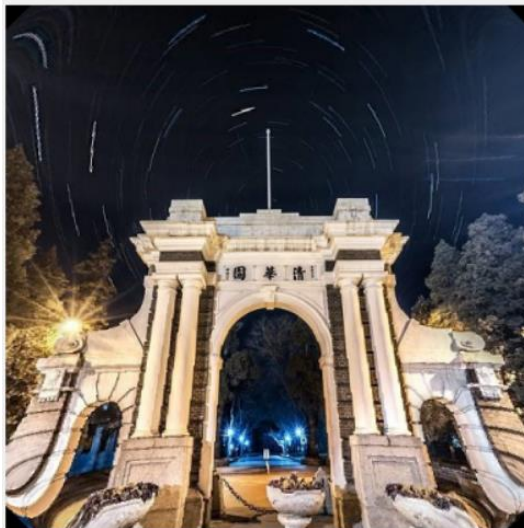
旋转角度 < > 半径 < > 408

☐ 最近邻插值 ☐ 双线性插值 ☒ 双三次插值

选择图片

变换图片

### 必做任务



变换类型 ☐ 旋转扭曲 ☒ 畸变校正

旋转方向 ☐ 顺时针 ☐ 逆时针

畸变类型 ☐ 桶形畸变 ☒ 枕形畸变

旋转角度 < > 半径 < > 345

☒ 最近邻插值 ☐ 双线性插值 ☐ 双三次插值

选择图片

变换图片

## 选做任务

### 方案设计

选做的几个关键步骤如下：

- (1) 分别读入待修改人脸（及其关键点数据）和目标人脸（及其关键点数据）
- (2) 实现 TPS 变形，得到修改后人脸与待修改人脸间的图像坐标映射
- (3) 选择插值方法（最近邻/双线性/双三次）得到结果

算法核心为（2），下面说明这一部分的原理。

### 方案基本原理

薄板样条（Thin plate spline, TPS）是一种常见的插值模型，目标是寻找一个通过所有控制点的光滑曲面 $f(x, y)$ ，使得能量函数最 $I_f$ 小。

$$I_f = \iint_{R^2} \left( \left( \frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 f}{\partial y^2} \right)^2 \right) dx dy$$

给定  $n$  个控制点（目标人脸图） $P_1 = (x_1, y_1), \dots, P_n = (x_n, y_n)$ ，记

$$K = \begin{bmatrix} 0 & U(r_{12}) & \dots & U(r_{1n}) \\ U(r_{21}) & 0 & \dots & U(r_{2n}) \\ \dots & \dots & \dots & \dots \\ U(r_{n1}) & U(r_{n2}) & \dots & 0 \end{bmatrix} \quad P = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ \dots & \dots & \dots \\ 1 & x_n & y_n \end{bmatrix} \quad L = \begin{bmatrix} K & P \\ P^T & \mathbf{0} \end{bmatrix}$$

假设目标点（待修改人脸图）为 $\tilde{P}_1 = (x'_1, y'_1), \dots, \tilde{P}_n = (x'_n, y'_n)$ ，记

$$V = \begin{bmatrix} x'_1 & x'_2 & \dots & x'_n \\ y'_1 & y'_2 & \dots & y'_n \end{bmatrix} \quad Y = \left( V \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)^T$$

则 $f(x, y) = [f_x(x, y), f_y(x, y)]^T = \mathbf{a}_1 + \mathbf{a}_x x + \mathbf{a}_y y + \sum_{i=1}^n \mathbf{w}_i U(|P_i - (x, y)|)$

其中 $\mathbf{a}_1, \mathbf{a}_x, \mathbf{a}_y, \mathbf{w}$ 为线性方程组 $L[\mathbf{w}_1, \dots, \mathbf{w}_n, \mathbf{a}_1, \mathbf{a}_x, \mathbf{a}_y]^T = Y$ 的解， $U(r) = \begin{cases} r^2 \log(r^2), & r \neq 0 \\ 0, & r = 0 \end{cases}$

因此（2）需要完成的即

① 构造矩阵  $L$  和  $Y$ ;

② 求解 $[\mathbf{w}_1, \dots, \mathbf{w}_n, \mathbf{a}_1, \mathbf{a}_x, \mathbf{a}_y]^T$ ;

③ 计算出 $f(x, y)$ .

## 功能展示



## 误差分析

### 观测误差

因为图片 RGB 为 unit8 类型，所以观测误差最大为 $\frac{1}{2}$ .

### 舍入误差

运算中间过程使用的数据类型为 C#的 double (15 位有效数字)，舍入误差

很小，基本可以忽略。在插值算法中对像素点赋值时将 double 转换为 unit8 类型，舍入误差最大为  $\frac{1}{2}$ 。

## 方法误差

### (1) 最近邻插值

利用 Math.Round ( ) 处理最近邻插值的非整点坐标，则有

$$\begin{cases} |\Delta x| \leq \frac{1}{2} \\ |\Delta y| \leq \frac{1}{2} \end{cases}$$

故截断误差上限为

$$|R(x, y)| \leq \max\left(\left|\frac{\partial f}{\partial x}\right|\right) |\Delta x| + \max\left(\left|\frac{\partial f}{\partial y}\right|\right) |\Delta y| \leq \frac{1}{2} \left[ \max\left(\left|\frac{\partial f}{\partial x}\right|\right) + \max\left(\left|\frac{\partial f}{\partial y}\right|\right) \right]$$

由于数字图像是离散信号，需将微分替换为差分，即

$$|R(x, y)| \leq \frac{1}{2} \left[ \max(|\Delta_x f|) + \max(|\Delta_y f|) \right]$$

### (2) 双线性插值

先分别计算各方向上的误差上限，再计算总误差上限。

对于  $x$  分量，需要计算  $y = j$  和  $y = j + 1$  两个方向的误差，有：

$$|R_x(x, j)| \leq \frac{1}{2!} \max\left(|\Delta_x^{(2)} f|\right) \cdot \max(|u(1-u)|) = \frac{1}{8} \max\left(|\Delta_x^{(2)} f|\right)$$

$$|R_x(x, j+1)| \leq \frac{1}{8} \max\left(|\Delta_x^{(2)} f|\right)$$

$$|R_x(x, y)| \leq (1-v) \cdot |R_x(x, j)| + v \cdot |R_x(x, j+1)| = \frac{1}{8} \max\left(|\Delta_x^{(2)} f|\right)$$

对于  $y$  分量，同理有

$$|R_y(x, y)| \leq \frac{1}{8} \max\left(|\Delta_y^{(2)} f|\right)$$

综上有

$$|R(x, y)| \leq \frac{1}{8} \left[ \max\left(|\Delta_x^{(2)} f|\right) + \max\left(|\Delta_y^{(2)} f|\right) \right]$$

### (3) 双三次插值

仿照双线性的分析方法有

$$|R_x(x, j+k-1)| \leq \frac{3}{128} \max\left(|\Delta_x^{(4)} f|\right), k = 0, 1, 2, 3$$



$$|R_y(j+k-1, y)| \leq \frac{3}{128} \max(|\Delta_y^{(4)} f|), k = 0, 1, 2, 3$$

$$|R_x(x, y)| \leq \frac{3}{128} \max(|\Delta_x^{(4)} f|)$$

$$|R_y(x, y)| \leq \frac{3}{128} \max(|\Delta_y^{(4)} f|)$$

$$|R(x, y)| \leq \frac{3}{128} [\max(|\Delta_x^{(4)} f|) + \max(|\Delta_y^{(4)} f|)]$$

## 总结

### 三种插值方式效果对比

根据理论知识，最近邻插值速度最快，但可能导致图像过渡不够好，并不常用；双线性插值速度也较快，但因为带有低通滤波特性可能会模糊图像；双三次插值速度较慢但精度最高。下面通过对比必做任务中的一组结果来对比三种方法的效果。



最近邻



双线性



双三次

变形方式为旋转扭曲，方向为顺时针，角度  $250^\circ$ ，半径 256。结果如上所示，对比可见和理论知识吻合。

## 反思与不足

通过本次大作业我熟悉了 C#并对它在制作图形界面方面的灵活性有了直观的认识,也对用到的图片变形方式和上面三种插值方式的原理有了更深刻的理解。由于我做必做任务时没有考虑选做,故将必做操作图的像素信息直接作为全局变量写进了函数中,做选做时不能直接调用,为了减少代码改动量重写了专用于选做部分的插值算法,其核心内容和必做相同,这一部分的可移植性不好。