

数值分析与算法

# 第一次大作业实验报告

新雅 62/CDIE6 2016013327 项雨桐

2019 年 11 月 12 日

# 目录

<b>1 实验分析</b>	<b>1</b>
1.1 实验要求	1
1.2 需求分析	1
<b>2 方案设计与基本原理</b>	<b>1</b>
2.1 坐标变换	1
2.1.1 旋转扭曲坐标变换	1
2.1.2 图像畸变坐标变换	2
2.2 插值算法	2
2.2.1 最近邻插值	3
2.2.2 双线性插值	3
2.2.3 双三次插值	3
<b>3 误差分析</b>	<b>4</b>
3.1 模型误差	4
3.1.1 图像采样定理	4
3.1.2 误差分析	5
3.1.3 总结	5
3.2 方法误差	5
3.2.1 最近邻插值	5
3.2.2 双线性插值	8
3.2.3 双三次插值	9
3.3 舍入误差	10
3.3.1 量化误差	10
3.3.2 浮点数存储/计算误差	10
3.3.3 截断误差	10
3.3.4 旋转坐标的计算误差	10
3.3.5 畸变坐标的计算误差	11
3.3.6 最近邻插值的计算误差	12
3.3.7 双线性插值的计算误差	13
3.3.8 双三次插值的计算误差	13

<b>4 实验总结</b>	<b>15</b>
4.1 利用插值法进行图像变形的一般过程 . . . . .	15
4.2 线性插值方法的特性 . . . . .	15

# 1 实验分析

## 1.1 实验要求

分别利用最近邻、双线性、双三次插值方法, 完成图像的旋转扭曲与畸变操作

## 1.2 需求分析

图像操作的实现主要分为两个部分:

1. 图像坐标变换: 坐标变换是图像变换的核心操作。图像变换的基本范式是规定原图像 (srcImage) 和目标图像 (dstImage) 坐标间的函数关系, 并通过数值计算求取对应的坐标值。
2. 像素插值计算: 通过 (srcImage) 对应坐标的像素值, 利用一定插值计算的方法获得 (dstImage) 的对应坐标的像素值。对于 RGB 图像, 可以对每个颜色通道分别插值后再组合获得新图像。

# 2 方案设计与基本原理

编写完成旋转扭曲和畸变坐标变换的算法, 通过 (dstImage) 获得相应的 (srcImage) 坐标, 通过对 (srcImage) 坐标进行插值计算, 获得对应 (dstImage) 坐标的像素值。

## 2.1 坐标变换

### 2.1.1 旋转扭曲坐标变换

旋转扭曲坐标变换需要设置两个参数: 最大旋转角  $\alpha_{\max}$  和旋转半径  $radius$ 。旋转中心为图像中央。

设 (dstImage) 的某点坐标为  $(x', y')$ , 则可得旋转角为

$$\alpha = \alpha_{\max} \cdot \frac{radius - distance}{radius}$$

其中  $distance$  是  $(x', y')$  到图像中心的欧式距离。

则对应的 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = x' \cos \alpha - y' \sin \alpha \\ y = x' \sin \alpha + y' \cos \alpha \end{cases} \quad (1)$$

本次实验中选取的旋转角分别为  $\frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}, \pi$ ；由于图像为正方形  $512 \times 512$ ，因此选取的旋转半径为图像尺寸的一半（像素是离散的，中心点选为  $(255.5, 255.5)$ ），因此半径  $radius = 255$ ）

### 2.1.2 图像畸变坐标变换

图像畸变坐标变换需要设置两个参数：正/负畸变和畸变半径  $radius$ ，畸变中心为图像中央。

设 (dstImage) 的某点坐标为  $(x', y')$ ，则可得正畸变下 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = [\frac{radius}{distance} \arcsin(\frac{distance}{radius})] \cdot x' \\ y = [\frac{radius}{distance} \arcsin(\frac{distance}{radius})] \cdot y' \end{cases} \quad (2)$$

负畸变下 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = [\frac{radius}{distance} \sin(\frac{distance}{radius})] \cdot x' \\ y = [\frac{radius}{distance} \sin(\frac{distance}{radius})] \cdot y' \end{cases} \quad (3)$$

其中  $distance$  是  $(x', y')$  到图像中心的欧式距离。

本次实验中选取的畸变半径分别为  $\frac{3}{4}radius$  和  $radius$ ，其中  $radius$  是旋转变换中使用的半径。

## 2.2 插值算法

设  $(x', y')$  经坐标变换得到的 (srcImage) 浮点型坐标为  $(x, y)$ ，令

$$\begin{cases} x = i + u \\ y = j + v \end{cases} \quad (4)$$

其中  $i = \text{floor}(x)$ ,  $j = \text{floor}(y)$ ，因此  $u, v \in [0, 1]$

设  $f(m, n)$  代表 (srcImage) 在坐标  $(m, n)$  处的像素值（单通道），则 (dstImage) 在 (srcImage) 对应坐标的像素值表示为  $f(x, y)$

### 2.2.1 最近邻插值

$$f(x, y) = \begin{cases} f(i, j) & u \leq 0.5, v \leq 0.5 \\ f(i+1, j) & u > 0.5, v \leq 0.5 \\ f(i, j+1) & u \leq 0.5, v > 0.5 \\ f(i+1, j+1) & u > 0.5, v > 0.5 \end{cases} \quad (5)$$

### 2.2.2 双线性插值

$$f(x, y) = \begin{bmatrix} 1-u & u \end{bmatrix} \begin{bmatrix} f(i, j) & f(i, j+1) \\ f(i+1, j) & f(i+1, j+1) \end{bmatrix} \begin{bmatrix} 1-v \\ v \end{bmatrix} \quad (6)$$

### 2.2.3 双三次插值

$$f(x, y) = ABC$$

其中

$$A = \begin{bmatrix} S(1+u) & S(u) & S(1-u) & S(2-u) \end{bmatrix} \quad (7)$$

$$B = \begin{bmatrix} f(i-1, j-1) & f(i-1, j) & f(i-1, j+1) & f(i-1, j+2) \\ f(i, j-1) & f(i, j) & f(i, j+1) & f(i, j+2) \\ f(i+1, j-1) & f(i+1, j) & f(i+1, j+1) & f(i+1, j+2) \\ f(i+2, j-1) & f(i+2, j) & f(i+2, j+1) & f(i+2, j+2) \end{bmatrix} \quad (8)$$

$$C = \begin{bmatrix} S(1+v) \\ S(v) \\ S(1-v) \\ S(2-v) \end{bmatrix} \quad (9)$$

插值基函数：

$$S(x) = \begin{cases} 1 - (a+3)|x|^2 + (a+2)|x|^3 & |x| \leq 1 \\ -4a + 8a|x| - 5a|x|^2 + a|x|^3 & 1 < |x| \leq 2 \\ 0 & otherwise \end{cases} \quad (10)$$

本实验中取  $a = -1$

### 3 误差分析

本实验的误差主要包括以下四种：

- 观测误差（采样误差）：图像采集设备的光学、机械、电学特性给采样数据带来的误差
- 模型误差：由采样值重建原模拟图像与实际图像函数的误差
- 方法误差：用插值函数重建图像与理想值的误差
- 舍入误差：像素值的量化误差、浮点数据的存储与计算带来的误差等

本实验的观测误差已不可考，下面分析其他三种误差：

#### 3.1 模型误差

分析图像重建带来的误差，先引入图像采样与重建的概念。

##### 3.1.1 图像采样定理

设图像在频域内是有限带宽的，也即：

$$\mathcal{F}(\Omega_x, \Omega_y) = 0$$

其中  $\|\Omega_x\| > \frac{\pi}{\Delta x}, \|\Omega_y\| > \frac{\pi}{\Delta y}$ ， $\Delta x$ 、 $\Delta y$  是采样间隔，则利用采样值可以精确重建原图像。设原图像坐标为  $(x, y)$ ，对应的像素值为  $f(x, y)$ ，则：

$$f(x, y) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} f(m\Delta x, n\Delta y) \cdot \text{sinc}(x - m\Delta x) \cdot \text{sinc}(y - n\Delta y)$$

其中  $\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$

在本实验中，我们取  $\Delta x = \Delta y = 1$ 。

### 3.1.2 误差分析

若图像在频域内有限带宽，由 IFT 的知识可知图像在空域内是无限延展的；反之，对任意有限范围内的图像，其频域内都有无限的高频成分。但由于人眼对图像的高频成分不敏感，所以由频率截断（低通滤波）带来的误差可以忽略不计，也即认为图像采样定理一般是成立的。这样，在图像函数满足 Dirichlet 条件的前提下，利用  $\text{sinc}(x)$  作为插值函数的图像重建函数与原图像的函数表达严格相等。

### 3.1.3 总结

在忽略低通滤波带来的截断误差、图像函数满足 Dirichlet 条件的前提下，图像函数

$$f(x, y) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} f(m, n) \cdot \text{sinc}(x - m) \cdot \text{sinc}(y - n)$$

不存在模型误差。

## 3.2 方法误差

方法误差：

$$R(x, y) = f(x, y) - L(x, y)$$

其中  $L(x, y)$  是插值函数。

令

$$\begin{cases} x = i + u & i = \text{floor}(x) \\ y = j + v & j = \text{floor}(y) \end{cases} \quad (11)$$

### 3.2.1 最近邻插值

插值函数：

$$\begin{aligned} L(x, y) = & f(i, j) \cdot \mathbb{1}_{\left| \begin{smallmatrix} u \leq 0.5, v \leq 0.5 \end{smallmatrix} \right|} + f(i + 1, j) \cdot \mathbb{1}_{\left| \begin{smallmatrix} u > 0.5, v \leq 0.5 \end{smallmatrix} \right|} \\ & + f(i, j + 1) \cdot \mathbb{1}_{\left| \begin{smallmatrix} u \leq 0.5, v > 0.5 \end{smallmatrix} \right|} + f(i + 1, j + 1) \cdot \mathbb{1}_{\left| \begin{smallmatrix} u > 0.5, v > 0.5 \end{smallmatrix} \right|} \end{aligned} \quad (12)$$



误差函数：

$$\begin{aligned}
R(x, y) &= f(i, j) \cdot [\text{sinc}(u) \cdot \text{sinc}(v) - \mathbb{1} \Big|_{u \leq 0.5, v \leq 0.5}] \\
&+ f(i+1, j) \cdot [\text{sinc}(u-1) \cdot \text{sinc}(v) - \mathbb{1} \Big|_{u > 0.5, v \leq 0.5}] \\
&+ f(i, j+1) \cdot [\text{sinc}(u) \cdot \text{sinc}(v-1) - \mathbb{1} \Big|_{u \leq 0.5, v > 0.5}] \\
&+ f(i+1, j+1) \cdot [\text{sinc}(u-1) \cdot \text{sinc}(v-1) - \mathbb{1} \Big|_{u > 0.5, v > 0.5}] \\
&+ \sum_{\substack{m=-\infty \\ \|x-m\| > 1}}^{+\infty} \sum_{\substack{n=-\infty \\ \|y-n\| > 1}}^{+\infty} f(m, n) \cdot \text{sinc}(x-m) \cdot \text{sinc}(y-n) \\
&= \text{Int}(u, v) + \text{Res}(x, y)
\end{aligned} \tag{13}$$

$g(s, t) = \text{sinc}(s) \cdot \text{sinc}(t)$  在  $[0, 1]^2$  上的图像如下所示：

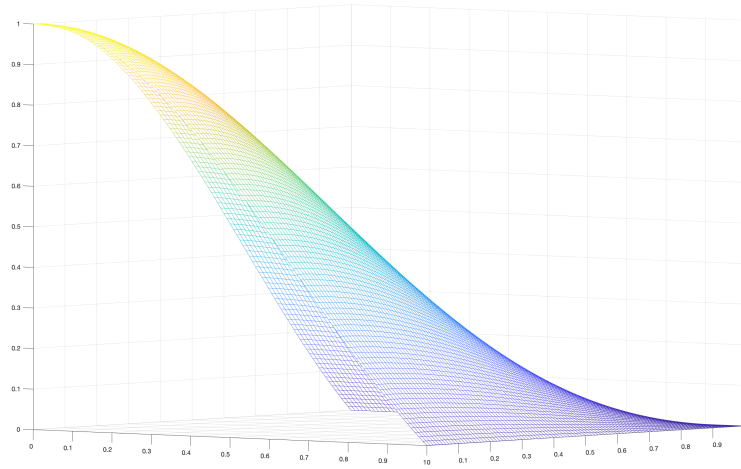


图 3.1:  $\text{sinc}(s) \cdot \text{sinc}(t)$  图像

因此当  $s = t = 0.5$  时  $g(s, t) - \mathbb{1} \Big|_{s < 0.5, v < 0.5}$  取最大值为  $\frac{4}{\pi^2}$

则

$$\begin{aligned}
\text{Int}(u, v) &\leq \frac{4}{\pi^2} \cdot \max\{f(i : i+1, j : j+1)\} \\
&+ \left(\frac{4}{\pi^2} - 1\right) \cdot \left(\sum_{k=i}^{i+1} \sum_{l=j}^{j+1} f(k, l) - \max\{f(i : i+1, j : j+1)\}\right) \\
&\approx 0.41 \cdot \max\{f(i : i+1, j : j+1)\} \\
&- 0.59 \cdot \left(\sum_{k=i}^{i+1} \sum_{l=j}^{j+1} f(k, l) - \max\{f(i : i+1, j : j+1)\}\right)
\end{aligned} \tag{14}$$

对于  $\text{sinc}(x-m) \cdot \text{sinc}(y-n)$ , 当  $\|x-m\| > 1, \|y-n\| > 1$  时, 由于  $x, y$  独立, 因此

$$\text{sinc}(x-m) \cdot \text{sinc}(y-n) \leq \text{sinc}^2(w_m)$$

其中  $w_{max}$  是  $\text{sinc}^2(w)$  的非零极值点, 满足超越方程  $\tan(w_{max}) = w_{max}$ ,  $\|w_{max}\| > 1$ , 分析可知点列  $w_{max}$  的正数部分第一项为  $w_1 = 1.43$ , 当  $n \rightarrow \infty$  时一致收敛于  $n+0.5$ , 因此  $n+0.4 < w_n < n+0.5$ 。

故:

$$\begin{aligned}
\text{Res}(x, y) &\leq \max f(m, n) \cdot \sum_{\substack{m=-\infty \\ \|x-m\| > 1}}^{+\infty} \sum_{\substack{n=-\infty \\ \|y-n\| > 1}}^{+\infty} \text{sinc}(x-m) \cdot \text{sinc}(y-n) \\
&\leq \max f(m, n) \cdot \sum_{w_{max} \in (-\infty, +\infty) \setminus \{0\}} \text{sinc}^2(w_{max}) \\
&\leq \max f(m, n) \cdot \sum_{w_{max} \in (-\infty, +\infty) \setminus \{0\}} \left[ \frac{\sin(\pi w_{max})}{\pi w_{max}} \right]^2 \\
&= \max f(m, n) \cdot \frac{2}{\pi^2} \sum_{w_{max} \in (0, +\infty)} \left[ \frac{\sin(\pi w_{max})}{w_{max}} \right]^2 \\
&\leq \max f(m, n) \cdot \frac{2}{\pi^2} \sum_{w_{max} \in (0, +\infty)} \left( \frac{1}{w_{max}} \right)^2 \\
&\leq \max f(m, n) \cdot \frac{2}{\pi^2} \sum_{n=1}^{+\infty} \left( \frac{1}{n+0.4} \right)^2 \\
&\leq \max f(m, n) \cdot \frac{2}{\pi^2} \cdot 1.03
\end{aligned}$$

因此

$$\begin{aligned}
R(x, y) &\leq \frac{4}{\pi^2} \cdot \max\{f(i : i + 1, j : j + 1)\} \\
&+ \left(\frac{4}{\pi^2} - 1\right) \cdot \left(\sum_{k=i}^{i+1} \sum_{l=j}^{j+1} f(k, l) - \max\{f(i : i + 1, j : j + 1)\}\right) \\
&+ \frac{2}{\pi^2} \sum_{n=1}^{+\infty} \left(\frac{1}{n + 0.4}\right)^2 \cdot \max f(m, n) \\
&\approx 0.41 \cdot \max\{f(i : i + 1, j : j + 1)\} \\
&- 0.59 \cdot \left(\sum_{k=i}^{i+1} \sum_{l=j}^{j+1} f(k, l) - \max\{f(i : i + 1, j : j + 1)\}\right) \\
&+ 0.21 \cdot \max_{\substack{m \in \mathbb{Z} \setminus \{i, i+1\} \\ n \in \mathbb{Z} \setminus \{j, j+1\}}} f(m, n)
\end{aligned} \tag{15}$$

是相关像素点值的线性组合

### 3.2.2 双线性插值

插值函数：

$$\begin{aligned}
L(x, y) &= f(i, j) \cdot (1 - u)(1 - v) + f(i + 1, j) \cdot u(1 - v) \\
&+ f(i, j + 1) \cdot (1 - u)v + f(i + 1, j + 1) \cdot uv
\end{aligned} \tag{16}$$

误差函数：

$$\begin{aligned}
R(x, y) &= f(i, j) \cdot [\text{sinc}(u) \cdot \text{sinc}(v) - (1 - u)(1 - v)] \\
&+ f(i + 1, j) \cdot [\text{sinc}(u - 1) \cdot \text{sinc}(v) - u(1 - v)] \\
&+ f(i, j + 1) \cdot [\text{sinc}(u) \cdot \text{sinc}(v - 1) - (1 - u)v] \\
&+ f(i + 1, j + 1) \cdot [\text{sinc}(u - 1) \cdot \text{sinc}(v - 1) - uv] \\
&+ \sum_{m=-\infty}^{+\infty} \sum_{\substack{n=-\infty \\ \|x-m\|>1 \\ \|y-n\|>1}}^{+\infty} f(m, n) \cdot \text{sinc}(x - m) \cdot \text{sinc}(y - n) \\
&= \text{Int}(u, v) + \text{Res}(x, y)
\end{aligned} \tag{17}$$

定义  $g(s, t) = \text{sinc}(s) \cdot \text{sinc}(t) - (1 - s)(1 - t)$ ,  $(s, t) \in [0, 1]^2$   
 $g(s, t)$  在  $[0, 1]^2$  上的图像如下所示：

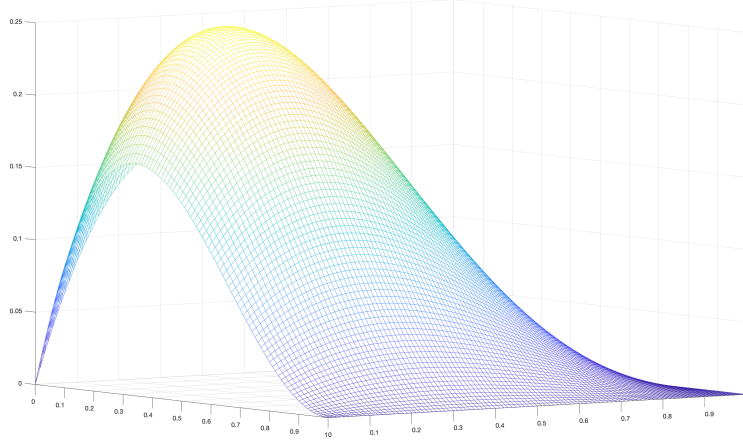


图 3.2:  $g(s, t)$  图像

$(s, t) \approx (0.27, 0.27)$  时,  $\max g(s, t) \approx 0.25$

$\text{Res}(x, y)$  的分析同最近邻插值, 因此:

$$R(x, y) \leq 0.25 * \max\{f(i : i + 1, j : j + 1)\} + 0.21 \cdot \max_{\substack{m \in \mathbb{Z} \setminus \{i, i+1\} \\ n \in \mathbb{Z} \setminus \{j, j+1\}}} f(m, n) \quad (18)$$

也是相关像素点值的线性组合

### 3.2.3 双三次插值

双三次插值的分析方法和前两者类似, 因公式过于繁琐不再赘述。另一种比较好的估计方法是:

$$\begin{aligned} |R_x(x, y)| &\leq \frac{1}{4!} \max \left| \frac{\partial^4 f}{\partial x^4} \right| |(1+u)u(1-u)(2-u)| = \frac{3}{128} \max \left| \frac{\partial^4 f}{\partial x^4} \right| \\ |R_y(x, y)| &\leq \frac{1}{4!} \max \left| \frac{\partial^4 f}{\partial y^4} \right| |(1+v)v(1-v)(2-v)| = \frac{3}{128} \max \left| \frac{\partial^4 f}{\partial y^4} \right| \end{aligned} \quad (19)$$

也即

$$|R(x, y)| \leq |R_x(x, y)| + |R_y(x, y)| = \frac{3}{128} \left( \max \left| \frac{\partial^4 f}{\partial x^4} \right| + \max \left| \frac{\partial^4 f}{\partial y^4} \right| \right) \quad (20)$$

### 3.3 舍入误差

舍入误差的来源如下：

#### 3.3.1 量化误差

量化误差主要是指像素值在量化为离散值时产生的误差。由于像素值的量化是向下取整，所以绝对误差为 1，相对误差为  $1/256$ 。

另外，在双三次插值出现 overshooting 和 undershooting 现象时采用 clamp 函数修正也会带来误差。

#### 3.3.2 浮点数存储/计算误差

- 程序中的 (dstImage) 坐标值和 (srcImage) 中心点坐标值、旋转半径不存在误差<sup>1</sup>
- 利用坐标计算的距离 distance 和最大旋转角  $\alpha_{\max}$  在存储时存在截断误差
- 利用二者计算得到变换后 (srcImage) 对应的浮点坐标  $(x, y)$ 、插值计算中得到的插值基函数、像素值存在计算带来的误差累积和本身的存储误差

#### 3.3.3 截断误差

float 格式利用 23 个比特位表示小数部分，因此所有 float 类型变量的截断误差是  $(\frac{1}{2})^{-24}$ 。

#### 3.3.4 旋转坐标的计算误差

由

$$\|\Delta A\| \leq \max \left\| \frac{\partial f}{\partial x} \right\| \cdot \|\Delta x\| + \max \left\| \frac{\partial f}{\partial y} \right\| \cdot \|\Delta y\|$$

和

$$\alpha = \alpha_{\max} \cdot \frac{radius - distance}{radius}$$
$$\Delta distance = \Delta \alpha_{\max} = \left(\frac{1}{2}\right)^{-24}$$

---

<sup>1</sup>中心点坐标值十进制为 255.5、旋转半径 0.75 倍 radius 为 191.25，在二进制用 float 存储时可以准确表示

知：

$$\begin{aligned}\frac{\partial}{\partial \alpha_{max}} &= \frac{radius - distance}{radius} \\ \frac{\partial}{\partial (distance)} &= -\frac{\alpha_{max}}{radius}\end{aligned}$$

因此：

$$\|\Delta\alpha\| \leq \max \left\| \frac{radius - distance}{radius} \right\| \cdot \left(\frac{1}{2}\right)^{-24} + \max \left\| \frac{\alpha_{max}}{radius} \right\| \cdot \left(\frac{1}{2}\right)^{-24}$$

对应的 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = x' \cos \alpha - y' \sin \alpha \\ y = x' \sin \alpha + y' \cos \alpha \end{cases} \quad (21)$$

则：

$$\begin{aligned}\frac{\partial x}{\partial \alpha} &= -x' \sin \alpha - y' \cos \alpha \\ \frac{\partial y}{\partial \alpha} &= x' \cos \alpha - y' \sin \alpha\end{aligned}$$

因此：

$$\begin{aligned}\|\Delta x\| &\leq \max \| -x' \sin \alpha - y' \cos \alpha \| \cdot \|\Delta\alpha\| \\ \|\Delta y\| &\leq \max \| x' \sin \alpha + y' \cos \alpha \| \cdot \|\Delta\alpha\|\end{aligned}$$

### 3.3.5 畸变坐标的计算误差

正畸变下 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = \left[ \frac{radius}{distance} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot x' \\ y = \left[ \frac{radius}{distance} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot y' \end{cases} \quad (22)$$

则

$$\begin{aligned}\frac{\partial x}{\partial distance} &= \left[ \frac{1}{distance \sqrt{1 - \left( \frac{distance}{radius} \right)^2}} - \frac{radius}{distance^2} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot x' \\ \frac{\partial y}{\partial distance} &= \left[ \frac{1}{distance \sqrt{1 - \left( \frac{distance}{radius} \right)^2}} - \frac{radius}{distance^2} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot y'\end{aligned}$$

因此：

$$\begin{aligned}\|\Delta x\| &\leq \max \left\| \left[ \frac{1}{distance \sqrt{1 - (\frac{distance}{radius})^2}} - \frac{radius}{distance^2} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot x' \right\| \cdot \left( \frac{1}{2} \right)^{-24} \\ \|\Delta y\| &\leq \max \left\| \left[ \frac{1}{distance \sqrt{1 - (\frac{distance}{radius})^2}} - \frac{radius}{distance^2} \arcsin \left( \frac{distance}{radius} \right) \right] \cdot y' \right\| \cdot \left( \frac{1}{2} \right)^{-24}\end{aligned}$$

负畸变下 (srcImage) 的坐标  $(x, y)$  满足：

$$\begin{cases} x = \left[ \frac{radius}{distance} \sin \left( \frac{distance}{radius} \right) \right] \cdot x' \\ y = \left[ \frac{radius}{distance} \sin \left( \frac{distance}{radius} \right) \right] \cdot y' \end{cases} \quad (23)$$

则

$$\begin{aligned}\frac{\partial x}{\partial distance} &= \left[ \frac{1}{distance} \cos \frac{distance}{radius} - \frac{radius}{distance^2} \sin \left( \frac{distance}{radius} \right) \right] \cdot x' \\ \frac{\partial y}{\partial distance} &= \left[ \frac{1}{distance} \cos \frac{distance}{radius} - \frac{radius}{distance^2} \sin \left( \frac{distance}{radius} \right) \right] \cdot y'\end{aligned}$$

因此：

$$\begin{aligned}\|\Delta x\| &\leq \max \left\| \left[ \frac{1}{distance} \cos \frac{distance}{radius} - \frac{radius}{distance^2} \sin \left( \frac{distance}{radius} \right) \right] \cdot x' \right\| \cdot \left( \frac{1}{2} \right)^{-24} \\ \|\Delta y\| &\leq \max \left\| \left[ \frac{1}{distance} \cos \frac{distance}{radius} - \frac{radius}{distance^2} \sin \left( \frac{distance}{radius} \right) \right] \cdot y' \right\| \cdot \left( \frac{1}{2} \right)^{-24}\end{aligned}$$

### 3.3.6 最近邻插值的计算误差

最近邻插值算法：

$$f(x, y) = \begin{cases} f(i, j) & u \leq 0.5, v \leq 0.5 \\ f(i + 1, j) & u > 0.5, v \leq 0.5 \\ f(i, j + 1) & u \leq 0.5, v > 0.5 \\ f(i + 1, j + 1) & u > 0.5, v > 0.5 \end{cases} \quad (24)$$

其中：

$$\begin{cases} x = i + u \\ y = j + v \end{cases} \quad (25)$$

$i = \text{floor}(x)$ ,  $j = \text{floor}(y)$ , 因此  $\Delta u = \Delta x$ ,  $\Delta v = \Delta y$

除去像素值本身的量化误差，由上述公式可知，算法引入的误差即为  $\Delta u = \Delta x$  和  $\Delta v = \Delta y$

### 3.3.7 双线性插值的计算误差

双线性插值算法：

$$\begin{aligned} f(x, y) = & f(i, j) \cdot (1 - u)(1 - v) + f(i + 1, j) \cdot u(1 - v) \\ & + f(i, j + 1) \cdot (1 - u)v + f(i + 1, j + 1) \cdot uv \end{aligned} \quad (26)$$

$$\begin{cases} x = i + u \\ y = j + v \end{cases} \quad (27)$$

$i = \text{floor}(x)$ ,  $j = \text{floor}(y)$ , 因此  $\Delta u = \Delta x$ ,  $\Delta v = \Delta y$

除去像素值本身的量化误差, 由上述公式可知：

$$\begin{aligned} \frac{\partial f}{\partial u} &= (1 - v)[f(i + 1, j) - f(i, j)] + v[f(i + 1, j + 1) - f(i, j + 1)] \\ \frac{\partial f}{\partial v} &= (1 - u)[f(i, j + 1) - f(i, j)] + u[f(i + 1, j + 1) - f(i + 1, j)] \end{aligned}$$

因此：

$$\begin{aligned} \|\Delta f\| \leq & \max_{(u, v) \in [0, 1]^2} \|(1 - v)[f(i + 1, j) - f(i, j)] + v[f(i + 1, j + 1) - f(i, j + 1)]\| \cdot \|\Delta x\| \\ & + \max_{(u, v) \in [0, 1]^2} \|(1 - u)[f(i, j + 1) - f(i, j)] + u[f(i + 1, j + 1) - f(i + 1, j)]\| \cdot \|\Delta y\| \end{aligned} \quad (28)$$

### 3.3.8 双三次插值的计算误差

插值基函数：

$$S(x) = \begin{cases} 1 - 2|x|^2 + |x|^3 & |x| \leq 1 \\ 4 - 8|x| + 5|x|^2 - |x|^3 & 1 < |x| \leq 2 \\ 0 & otherwise \end{cases} \quad (29)$$

插值函数：

$$f(x, y) = ABC$$

其中

$$A = \begin{bmatrix} S(1 + u) & S(u) & S(1 - u) & S(2 - u) \end{bmatrix} \quad (30)$$



$$B = \begin{bmatrix} f(i-1, j-1) & f(i-1, j) & f(i-1, j+1) & f(i-1, j+2) \\ f(i, j-1) & f(i, j) & f(i, j+1) & f(i, j+2) \\ f(i+1, j-1) & f(i+1, j) & f(i+1, j+1) & f(i+1, j+2) \\ f(i+2, j-1) & f(i+2, j) & f(i+2, j+1) & f(i+2, j+2) \end{bmatrix} \quad (31)$$

$$C = \begin{bmatrix} S(1+v) \\ S(v) \\ S(1-v) \\ S(2-v) \end{bmatrix} \quad (32)$$

则

$$\begin{aligned} \frac{\partial f}{\partial u} &= \frac{\partial A}{\partial u} BC \\ \frac{\partial f}{\partial v} &= AB \frac{\partial C}{\partial v} \end{aligned} \quad (33)$$

其中

$$\frac{\partial A}{\partial u} = \begin{bmatrix} \frac{\partial S(1+u)}{\partial u} & \frac{\partial S(u)}{\partial u} & \frac{\partial S(1-u)}{\partial u} & \frac{\partial S(2-u)}{\partial u} \end{bmatrix} \quad (34)$$

$$\frac{\partial C}{\partial v} = \begin{bmatrix} \frac{\partial S(1+v)}{\partial v} \\ \frac{\partial S(v)}{\partial v} \\ \frac{\partial S(1-v)}{\partial v} \\ \frac{\partial S(2-v)}{\partial v} \end{bmatrix} \quad (35)$$

对  $t \in [0, 1]$

$$\begin{aligned} \frac{\partial S(1+t)}{\partial t} &= -1 + 4t - 3t^2 \\ \frac{\partial S(t)}{\partial t} &= -4t + 3t^2 \\ \frac{\partial S(1-t)}{\partial t} &= 1 + 2t - 3t^2 \\ \frac{\partial S(2-t)}{\partial t} &= -24 + 22t - 3t^2 \end{aligned}$$

因此：

$$\|\Delta f\| \leq \max_{(u,v) \in [0,1]^2} \left\| \frac{\partial A}{\partial u} BC \right\| \cdot \|\Delta x\| + \max_{(u,v) \in [0,1]^2} \left\| AB \frac{\partial C}{\partial v} \right\| \cdot \|\Delta y\| \quad (36)$$

## 4 实验总结

### 4.1 利用插值法进行图像变形的一般过程

1. 图片预处理，一般包括通道分离等
2. 确定坐标变换函数，由目标图像对应整型坐标得到原图像浮点坐标
3. 利用插值算法，从原图像整型坐标的像素值获得浮点坐标的像素值，也即目标图像对应点的像素值
4. 图片终处理，例如通道合成，偏差修正等

### 4.2 线性插值方法的特性

线性插值方法表现在预测点的像素值是原图像像素值的线性组合。不同的线性组合系数、原图像像素值的选取会得到不同的线性插值方法。

最近邻插值是一种零次插值，表现在只选取了原图像的 1 个采样点来预测对应点的像素；双线性插值是两个方向上的一次插值，因此利用了最近的 4 个点的像素，插值基函数是线性的；双三次插值是两个方向上的三次插值，利用了附近 16 个点的像素值，插值基函数为 3 次，可以通过选取不同的参数来调节。

插值基函数的特性<sup>2</sup>：

- $S(0) = 1$
- $S(n) = 0, n \in \mathbb{Z} \setminus \{0\}$
- $S(x) = 0$ , 当  $x$  超过插值范围时

本实验中的插值基函数是对理想插值函数  $\text{sinc}(x)$  的逼近

这三种线性插值均采用多项式函数作为插值基函数，如果图像函数  $f(x, y)$  满足一定的光滑特性，则可以证明这些经插值后的图像一致收敛于原图像函数；且多项式阶次越高、选取的采样点越多，方法误差就越小，表现在图像上则是人工痕迹例如锯齿等现象越不明显。因此，从获得的图像中可以看出，双三次插值得到的图像最自然，双线性次之，最近邻最差。但多项式阶次和采样点数目的增长带来的是计算复杂度的增加，因此在实际

---

<sup>2</sup>[https://dailc.github.io/2017/11/01/imageprocess\\_bicubicinterpolation.html](https://dailc.github.io/2017/11/01/imageprocess_bicubicinterpolation.html)

应用中，应当根据所需的性能与计算开销之间进行平衡，合理选取插值方案。

从分析中也可以看出，线性插值方法的误差对于各采样点像素值也是线性的。

另外还需指出的是，双三次插值有可能造成 overshooting 或 undershooting 现象，也即某像素预测值超出了 0~255 的范围<sup>3</sup>。表现在图像上就是产生许多纯度高的彩色噪点<sup>4</sup>。这时需要利用 clamp 函数来将插值结果限制为 0~255 的范围。

---

<sup>3</sup><https://github.com/pytorch/pytorch/issues/21044>

<sup>4</sup>其实是 undershooting 产生的负值在强制转换为 uchar 类型时变为 overshooting