

****

生物医学图像处理课程

**编程练习题**

学院： 计算机学院 专业：计算机体系结构

姓名： 余 乔 班级： 博1901班

学号： D201980953 时间： 2019年10月

# 图像旋转

## 题目描述

## 基本思路

## 算法描述

## 代码实现

## 结果展示

# 直方图规定化

## 题目描述

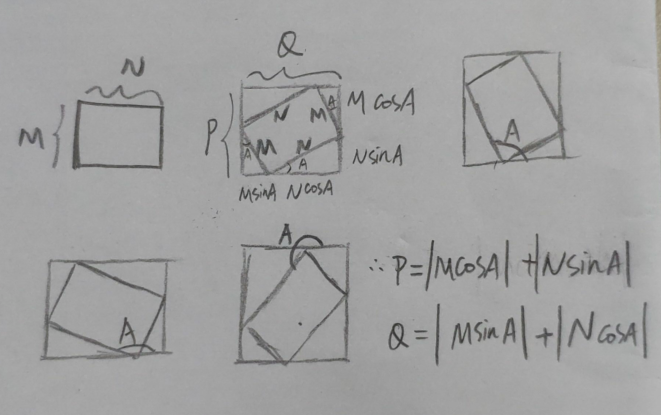
设一幅大小为M×N的灰度图像I中，现要将其逆时针旋转 A度，得到图像J，请写出J的生成算法（要求使用近邻插值）。

## 基本思路

1. 首先求出旋转后图像J的行数P和列数Q。
2. 对J中的每个像素的坐标 (x1,y1)，求出与其对应的原图I中像素的坐标(x,y)
3. 用最邻近插值，求出最接近于坐标(x,y)的整数坐标(x0,y0)
4. 将J (x1,y1)的灰度值赋为I(x0,y0)的灰度值。

### 求J的行列数P，Q

其中，行列数PQ的求法如下：



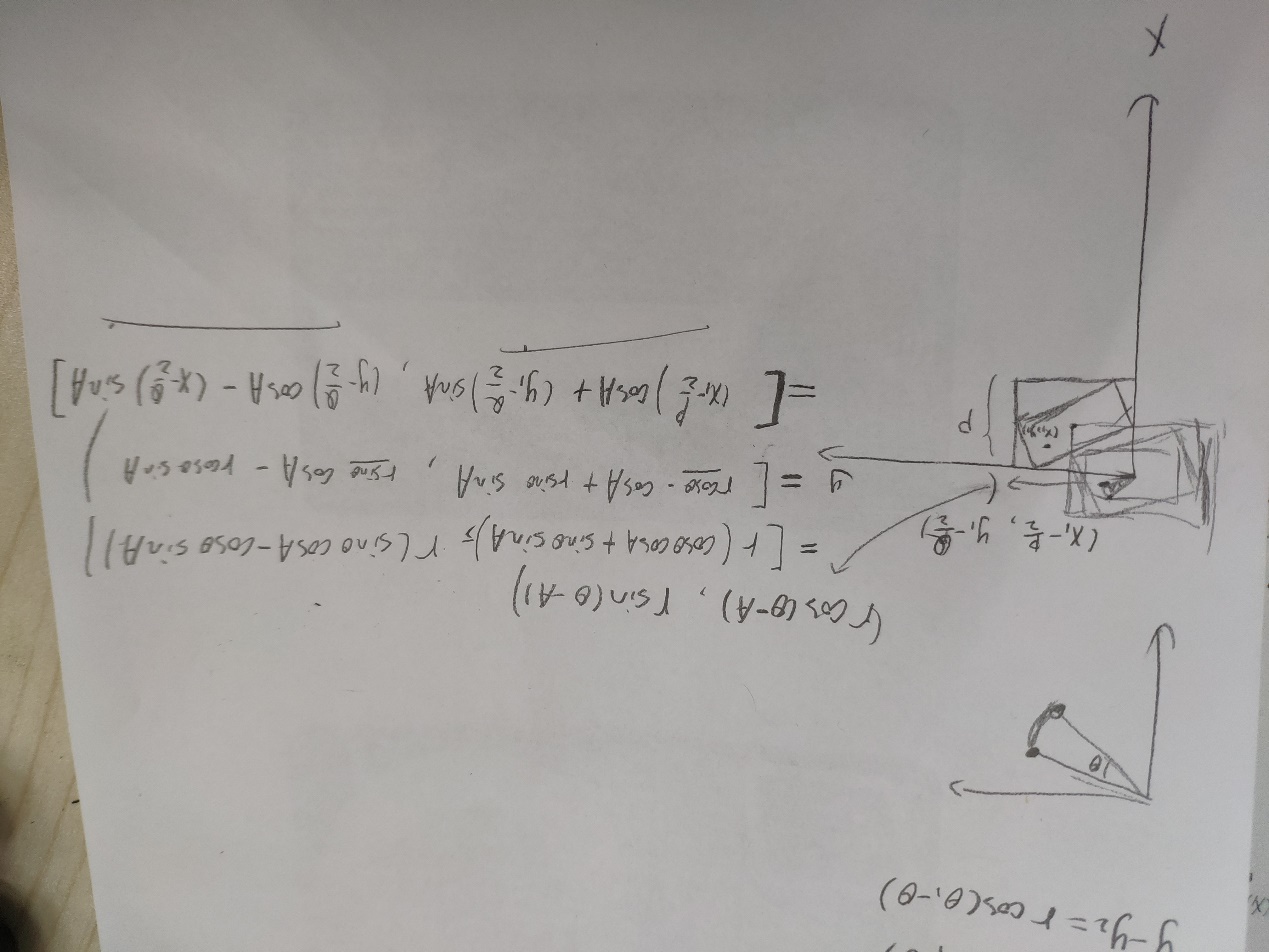
### 求J(x1,y1)对应点I(x,y)的坐标

J(x1,y1)对应点I(x,y)的坐标求法如下：

1. 将J的中心点J(P/2,Q/2)移动到(0,0)处
2. 将整个图像顺时针旋转A度
3. 最后将图像的(0,0)平移到(M/2,N/2)。

此时，原来在(x1,y1)的像素，就被变换到了(x,y)处.

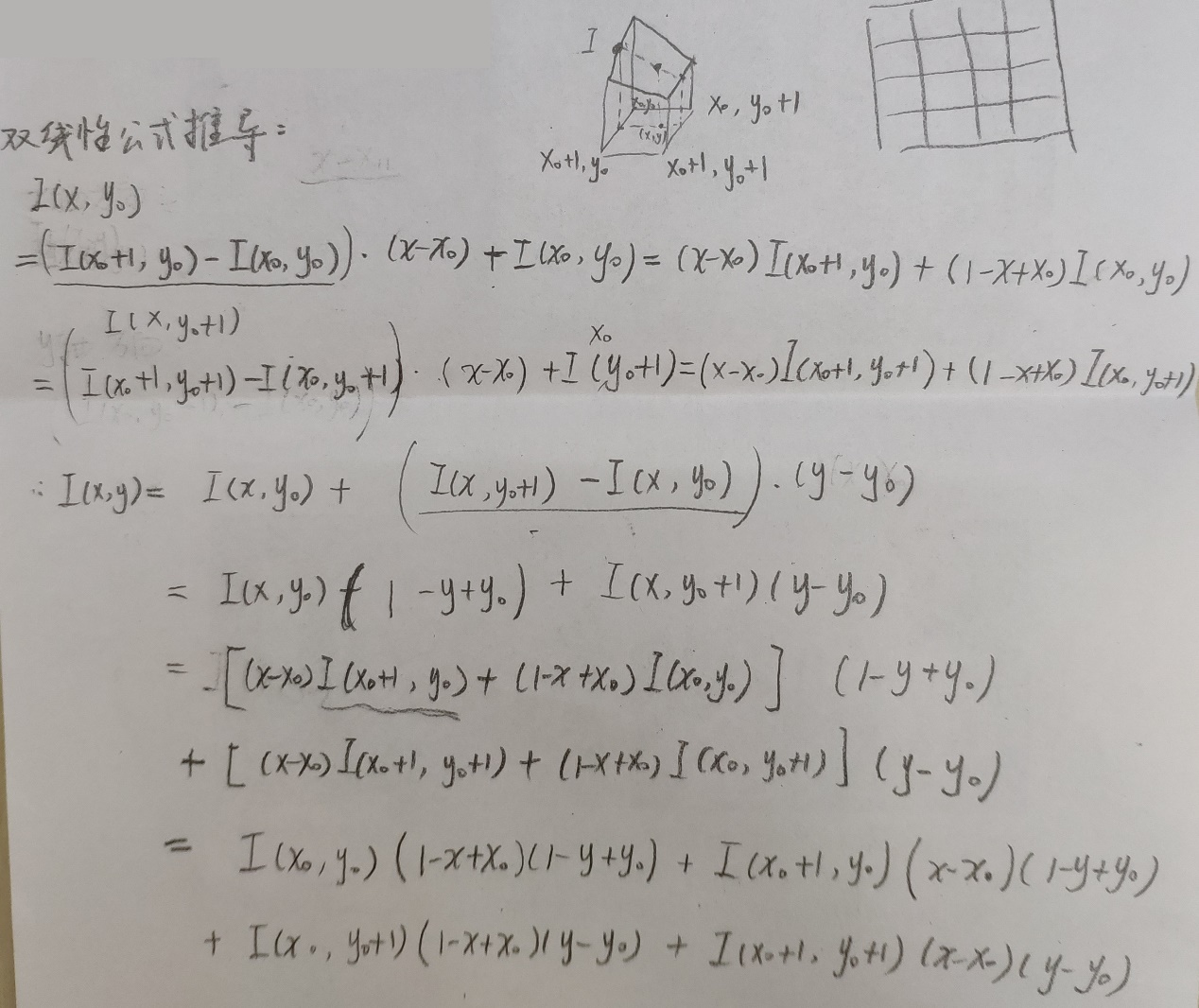
下图要换



### 最邻近插值&双线性插值

最邻近插值中，直接x0,y0分别直接取x,y四舍五入得到的整数。

如果使用双线性插值，则方法如下：



## 算法描述

P = |M cosA | + |N sinA|

Q = |M sinA | + |N cosA|

for x1 = 1:P

for y1 = 1:Q

x = (x1-P\*0.5) \* cos(A) + (y1-Q\*0.5)\* sin(A) + M\*0.5;

y = -(x1-P\*0.5) \* sin(A) + (y1-Q\*0.5)\* cos(A) + N\*0.5;

x0 = 最接近x的整数

y0 = 最接近y的整数

如果I(x0,y0)的坐标x0不在0~M范围内或y0不在0~N范围内，则对应的J(x1,y1) = 0

否则J(x1,y1) = I(x0,y0)

## 代码实现

function [J] = rotateYQ(I,A)

%rotate 设一幅大小为M×N的灰度图像I中，现要将其逆时针旋转 A度（角度制），得到图像J，请写出J的生成算法（要求使用近邻插值）。

% 设I(x,y)经旋转后对应的点为J(x1,y1).(x0,y0)是最接近(x,y)的整数对

% 求J的行列数P/Q

[M, N ] = size(I);

P = round( abs(M \* cosd(A)) + abs(N \* sind(A)));

Q = round( abs(M \* sind(A)) +abs( N \* cosd(A)));

J = zeros(P,Q);

for x1 = 1:P

for y1 = 1:Q

% 求J(x1,y1)在原图I中对应的像素坐标(x,y)

x = (x1-P\*0.5) \* cosd(A) + (y1-Q\*0.5)\* sind(A) + M\*0.5;

y = -(x1-P\*0.5) \* sind(A) + (y1-Q\*0.5)\* cosd(A) + N\*0.5;

% 最邻近插值

x0 = round(x);

y0 = round(y);

% 给J赋值

if x0 < 1 || y0 < 1 || x0 > M || y0 > N

J(x1,y1) = 0;

else

J(x1,y1) = I(x0,y0);

end

end

end

% 显示结果

imshow(uint8(I));title('原图I');

figure(2),imshow(uint8(J));title('旋转后的图像J');

end

## 结果展示

# 频域滤波

## 题目描述

基本思路

算法描述

代码实现

结果展示

# 连通区域标记

## 题目描述

基本思路

算法描述

代码实现

结果展示

# 边界追踪

## 题目描述

基本思路

算法描述

代码实现

结果展示

# 附加题：图中有多少粒米？

## 题目描述

基本思路

算法描述

代码实现

结果展示