

实验报告

实验题目

图像处理中的傅里叶变换与插值操作

实验目的

- 实现图像的顺时针旋转和基于最近邻与双线性插值的图像放大。
- 进行图像的傅里叶变换，提取频谱信息。

实验原理

1. 傅里叶变换

傅里叶变换是一种数学工具，用于将信号转换到频率域。它表明任何信号都可以表示为一系列正弦波的叠加，揭示信号的频率特性。

2. 插值方法

插值是在图像缩放过程中，根据已有像素值估算新像素值的过程。通过最近邻插值和双线性插值两种方法对图像进行放大处理。

实验内容

1. 图像顺时针旋转 30 度

```

% 设定旋转角度
theta = 30; % 旋转角度
theta_rad = deg2rad(theta); % 转换为弧度

% 计算旋转矩阵
R = [cos(theta_rad) -sin(theta_rad); sin(theta_rad) cos(theta_rad)];

% 计算旋转后的图像大小
new_rows = round(abs(rows * cos(theta_rad)) + abs(cols * sin(theta_rad)));
new_cols = round(abs(cols * cos(theta_rad)) + abs(rows * sin(theta_rad)));

% 创建新图像
rotatedImg = zeros(new_rows, new_cols, 'uint8');

% 计算偏移中心
offsetX = new_cols / 2;
offsetY = new_rows / 2;

% 旋转图像
for x = 1:cols
    for y = 1:rows
        % 原图像的像素坐标
        originalCoords = [x - cols / 2; y - rows / 2];

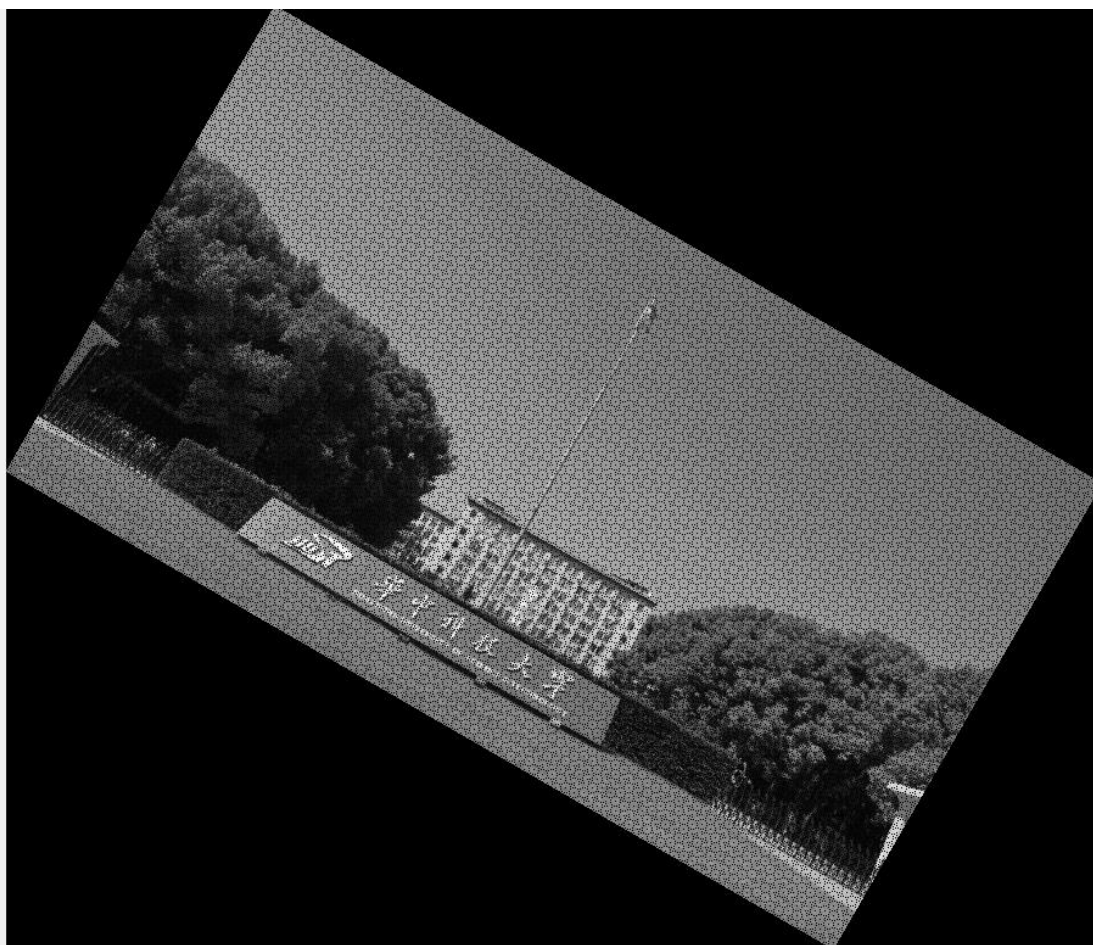
        % 应用旋转
        newCoords = R * originalCoords;

        % 计算新坐标
        newX = round(newCoords(1) + offsetX);
        newY = round(newCoords(2) + offsetY);

        % 将像素值放入新图像
        if newX > 0 && newX <= new_cols && newY > 0 && newY <= new_rows
            rotatedImg(newY, newX) = Y(y, x);
        end
    end
end
figure;
imshow(rotatedImg);

```

代码如下，实验效果：



2. 图像插值放大

代码如下：

```
% 最近邻插值放大 2 倍 largeImg_nearest2x = imresize(grayImg, 2, 'nearest'); % 最近邻插值放大 4 倍
largeImg_nearest4x = imresize(grayImg, 4, 'nearest'); % 双线性插值放大 2 倍 largeImg_bilinear2x =
imresize(grayImg, 2, 'bilinear'); % 双线性插值放大 4 倍 largeImg_bilinear4x = imresize(grayImg, 4,
'bilinear'); % 显示结果 figure; subplot(2, 2, 1); imshow(largeImg_nearest2x); title('Nearest
Neighbor (2x)'); subplot(2, 2, 2); imshow(largeImg_nearest4x); title('Nearest Neighbor (4x)');
subplot(2, 2, 3); imshow(largeImg_bilinear2x); title('Bilinear (2x)'); subplot(2, 2, 4);
imshow(largeImg_bilinear4x); title('Bilinear (4x)');
```

结果如下：



最近邻插值法 2 倍 1



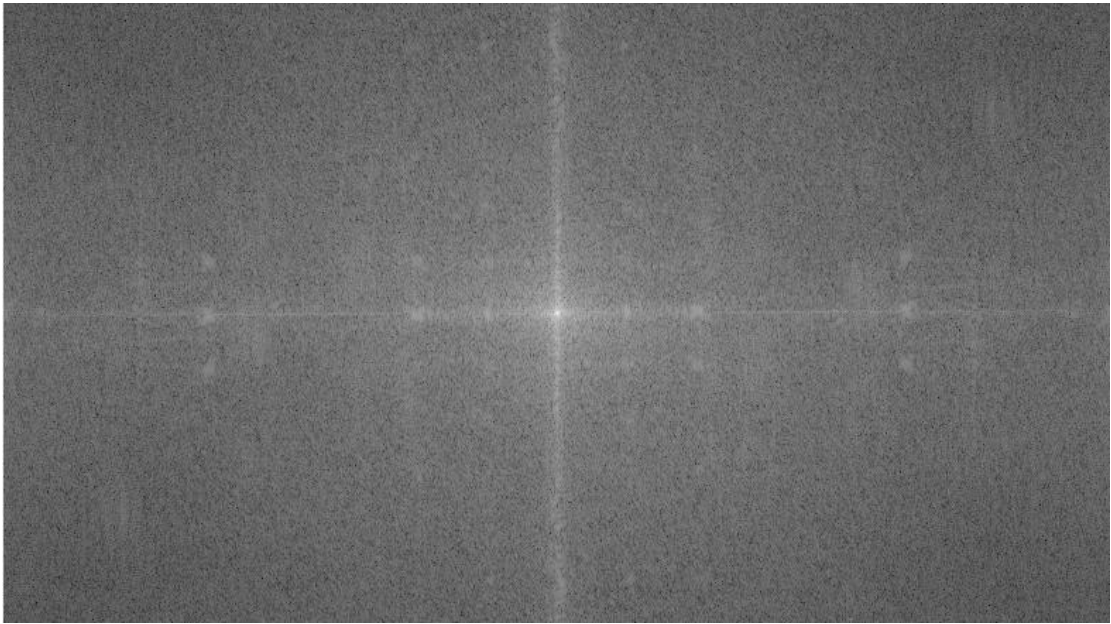
双线性插值法 2 倍 1

3. 图像傅里叶变换

执行图像的傅里叶变换，并提取其幅度谱，代码如下：

```
% 读取并转换为灰度图像 img = imread('your_image.jpg'); grayImg = rgb2gray(img); % 进行傅里叶变换 F
= fft2(double(grayImg)); % 进行 2D 傅里叶变换 F_shifted = fftshift(F); % 将频谱移到中心 % 计算幅度谱
magnitudeSpectrum = log(1 + abs(F_shifted)); % 显示傅里叶变换的结果 figure;
imshow(magnitudeSpectrum, []); title('Magnitude Spectrum');
```


结果如下：



幅度谱分布 1

实验结果评价

- 插值放大的结果展示了不同方法的效果，最近邻插值显得较为模糊，而双线性插值提供了更平滑的结果。
- 傅里叶变换的幅度谱清楚地展示了图像在频率域中的分布特性。

结论

通过本次实验，深入理解了傅里叶变换及其在图像处理中的应用，同时掌握了图像旋转和插值放大的基本方法。这些技术在实际应用中具有重要意义，能够帮助我们进行更精细的图像分析与处理。

