

**《数字图像处理》实验指导书**

**吕巨建**

**计算机科学学院**

### 实验一 图像基本运算

1. **实验目的**
2. 熟悉PYTHON数字图像的主要操作命令；
3. 学会数字图像的读取、写入、查询和显示；
4. 掌握数字图像的基本运算。
5. **实验内容**
6. PYTHON**数字图像的主要操作命令**
7. cv2.imread 图像文件的读取

**代码：**

**import cv2**

I=cv2.imread('cameraman.png')

cv2,imshow(I)

**cv2.waitKey(0)**

**结果：**



1. cv2.imwrite 图像文件的写入

**代码：**

**import cv2**

img=cv2.imread('cameraman.png')

cv2.imwrite(‘cameraman.png’,img)

**结果：**

在工作文件下面会生成一个名为cameraman的png格式的图像文件。



1. os.stat 查询图像文件信息

**代码：**

import os

fileinfo = os.stat("cameraman.png ")

print("索引号：",fileinfo.st\_ino)

print("被连接数目：",fileinfo.st\_dev)

print("文件大小:",fileinfo.st\_size,"字节")

print("最后一次访问时间:",fileinfo.st\_atime)

print("最后一次修改时间:",fileinfo.st\_mtime)

print("最后一次状态变化的时间：",fileinfo.st\_ctime)

**结果：**

索引号： 21673573207544768

被连接数目： 3404562452

文件大小: 196662 字节

最后一次访问时间: 1632807067.8367934

最后一次修改时间: 1632272070.4424326

最后一次状态变化的时间： 1632807067.8357852

1. cv2.imshow 显示图像

**代码：**

**import cv2**

I=cv2.imread('cameraman.png')

cv2,imshow(I)

**cv2.waitKey(0)**

**结果：**

****

1. 一个图像窗口中显示多幅图像

**代码：**

import cv2

import matplotlib.pyplot as plt

img3=cv2.imread('a.jpg')

gray=cv.cvtColor(img3,cv.COLOR\_RGB2GRAY) #灰度化

fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)

axes[0].imshow(img3[:,:,::-1])

axes[0].set\_title("orign pic")

axes[1].imshow(gray[:-1])

axes[1].set\_title("grey")

plt.show()

**结果：**



1. cv2.add 两幅图像相加

**代码：**

import cv2

Img1=cv2.imread('cameraman.bmp');

img3 = cv2.add(Img1, Img1) # cv中的加法

cv2.imshow('origin',Img1)

cv2.imshow('imgname',img3)

cv2.waitKey(0)

**结果：**



1. subtract 将一幅图像从另一幅图像中减去

**代码：**

import cv2

Img1=cv2.imread('cameraman.bmp');

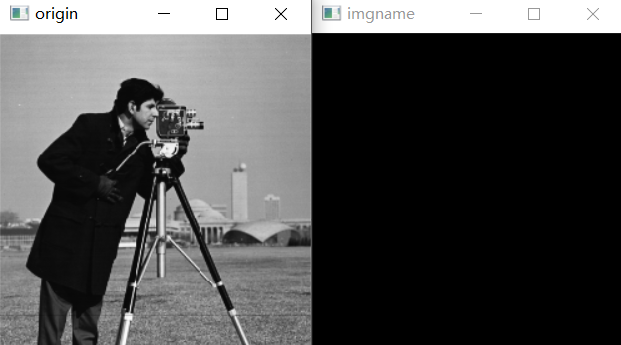
img3 = cv2.subtract(Img1, Img1) # cv中的加法

cv2.imshow('origin',Img1)

cv2.imshow('imgname',img3)

cv2.waitKey(0)

**结果：**



1. resize 图像的缩放

**代码：**

I=cv2.imread('cameraman.tif');

# 2.1 绝对尺寸

rows,cols = img1.shape[:2]

res = cv.resize(img1,(2\*cols,2\*rows),interpolation=cv.INTER\_CUBIC)

cv2.imshow(‘imname’,res)

cv2.waitKey(0)

# 2.2 相对尺寸

res1 = cv.resize(img1,None,fx=0.5,fy=0.5)

cv2.imshow(‘imname’,res1)

cv2.waitKey(0)

**结果：**



1. warpAffine图像的旋转

**代码：**

import numpy as np

import cv2 as cv

import matplotlib.pyplot as plt

# 1 读取图像

img = cv.imread("'cameraman.tif'")

# 2 图像旋转

rows,cols = img.shape[:2]

# 2.1 生成旋转矩阵          旋转中心点坐标  角度 缩放比例

M = cv.getRotationMatrix2D((cols/2,rows/2),90,1)

# 2.2 进行旋转变换

dst = cv.warpAffine(img,M,(cols,rows))

# 3 图像展示

fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)

axes[0].imshow(img[:,:,::-1])

axes[0].set\_title("原图")

axes[1].imshow(dst[:,:,::-1])

axes[1].set\_title("旋转后结果")

plt.show()**结果：**



1. 图像的剪切

**代码：**

img=cv.imread('cameraman.tif ')

cat = img[50:60,45:55] #截取矩阵的w：50~60，h：45~55

cv2.imgshow(‘imgname’,img)

cv2.waitKey(0)

**结果：**



1. **学生作业**
2. 从文件中读入一幅图像，将其放大到1.5倍；

import cv2

I = cv2.imread('cameraman.png')

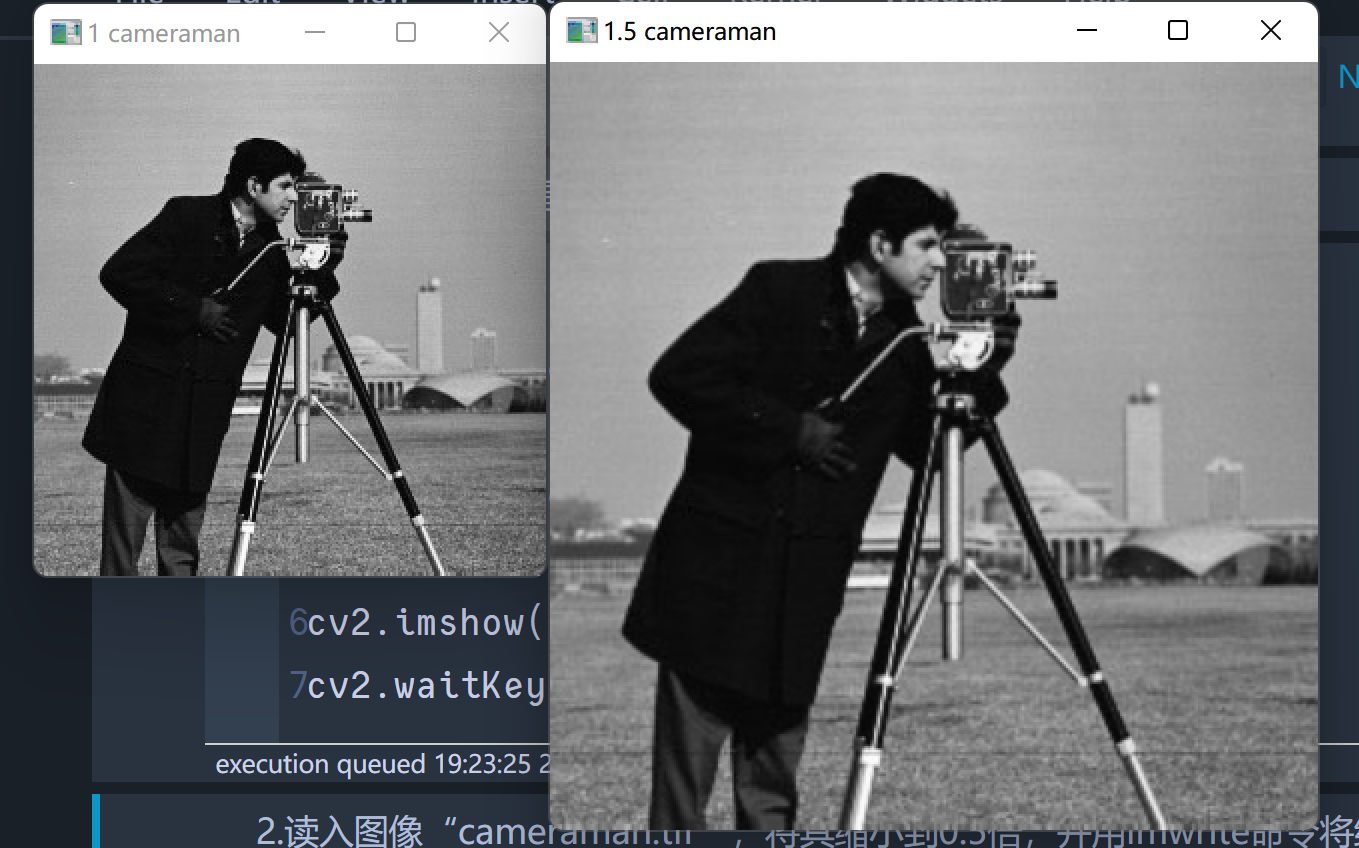
rows,cols = I.shape[:2]

res = cv2.resize(I,None,fx=1.5,fy=1.5)

cv2.imshow('1 cameraman',I)

cv2.imshow('1.5 cameraman',res)

cv2.waitKey(0)



1. 读入图像“cameraman.tif”，将其缩小到0.5倍，并用imwrite命令将结果以文件名“cameraman”的 bmp格式文件写入硬盘；

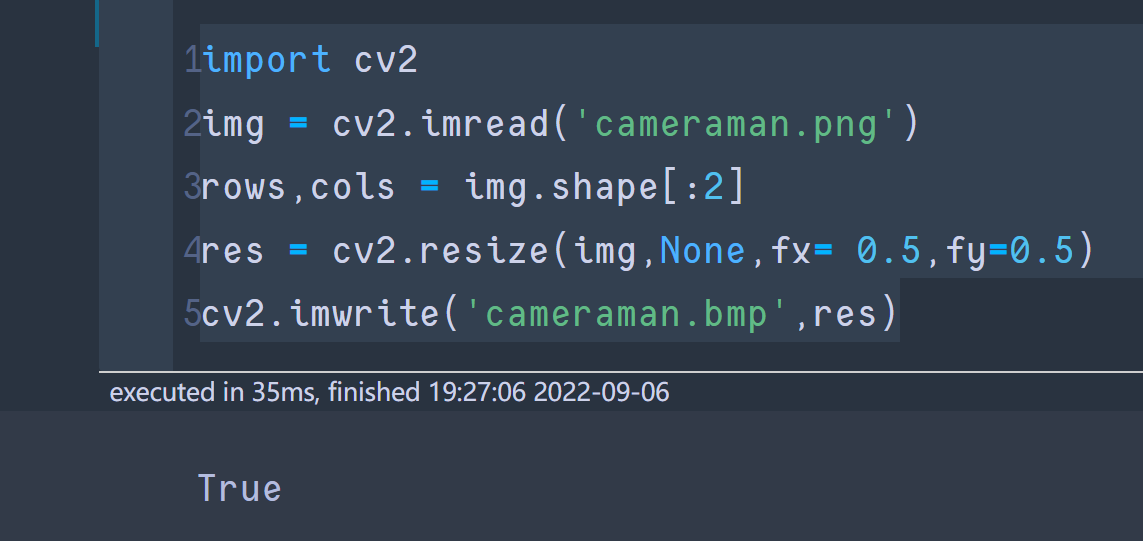
import cv2

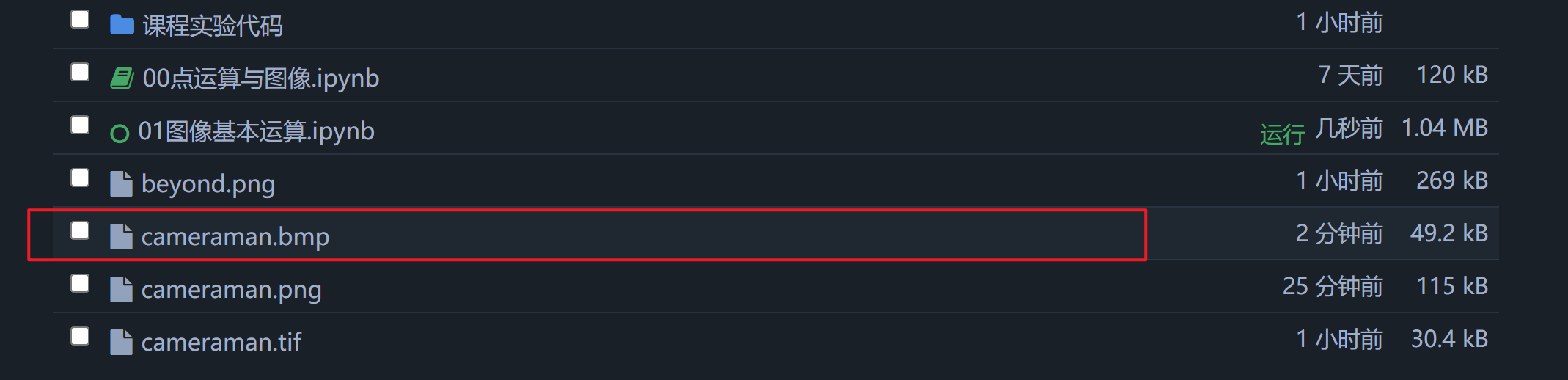
img = cv2.imread('cameraman.png')

rows,cols = img.shape[:2]

res = cv2.resize(img,None,fx= 0.5,fy=0.5)

cv2.imwrite('cameraman.bmp',res)





1. 读入一幅图像（自选）， 分别将其顺时针旋转30度，逆时针旋转30度，并将原图和旋转后的图显示在一个图形框（总共3幅图）。

import cv2

import matplotlib.pyplot as plt

img = cv2.imread('beyond.png')

rows,cols = img.shape[:2]

# 生成逆旋转矩阵

img\_ccw = cv2.getRotationMatrix2D((cols/2,rows/2),30,1)

# 生成顺旋转矩阵

img\_cw = cv2.getRotationMatrix2D((cols/2,rows/2),-30,1)

# 旋转变换

ccw = cv2.warpAffine(img,img\_ccw,(cols,rows))

cw = cv2.warpAffine(img,img\_cw,(cols,rows))

# 3 图像展示

fig,axes=plt.subplots(nrows=1,ncols=3,figsize=(10,8),dpi=100)

axes[0].imshow(img[:,:,::-1])

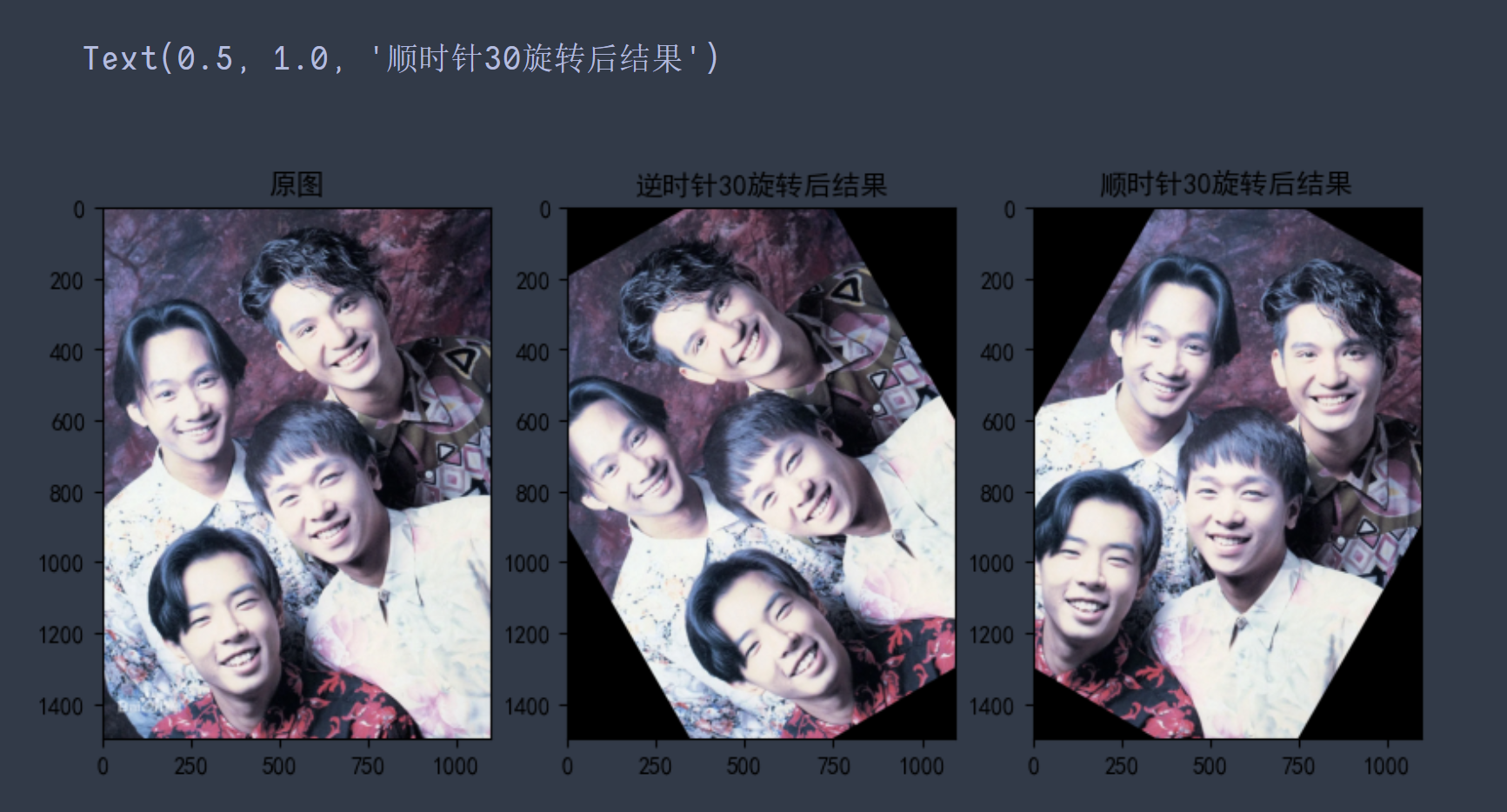
axes[0].set\_title("原图")

axes[1].imshow(ccw[:,:,::-1])

axes[1].set\_title("逆时针30旋转后结果")

axes[2].imshow(cw[:,:,::-1])

axes[2].set\_title("顺时针30旋转后结果")



1. **小结**
2. 用Python独立编程完成实验和作业内容并给出运行结果。
3. 实验报告书写格式要求规范，内容包括：（1）实验目的；（2）实验内容；（3）程序清单；（4）运行结果；（5）总结（实验结果分析、心得与体会等）。

### 实验二 图像变换

1. **实验目的**
2. 熟悉图像从空间到频域的变换；
3. 掌握图像的傅里叶变换及其性质；
4. 掌握图像的离散余弦变换及其性质。
5. **实验内容**
6. **图像的傅里叶变换**

**代码：**

import cv2

import numpy as np

import matplotlib.pyplot as plt

# 读取图片 灰度图

img = cv2.imread('./img/person.png',0)

img32=np.float32(img)

# 傅里叶变换

f = np.fft.fft2(img)

fshift = np.fft.fftshift(f)

print('f',f.shape)

print('fshift',fshift.shape)

"""

(240, 235, 2)

(240, 235, 2)

为什么这里多了两个维度，因为傅里叶变换后的结果有 "实数和虚数"

"""

#取绝对值：将复数变化成实数

#取对数的目的为了将数据变化到较小的范围（比如0-255）

s1 = np.log(np.abs(f))

s2 = np.log(np.abs(fshift))

fig,axes=plt.subplots(nrows=1,ncols=3,figsize=(10,8),dpi=100)

axes[0].imshow(img,'gray')

axes[0].set\_title("orign pic")

axes[1].imshow(s1,'gray')

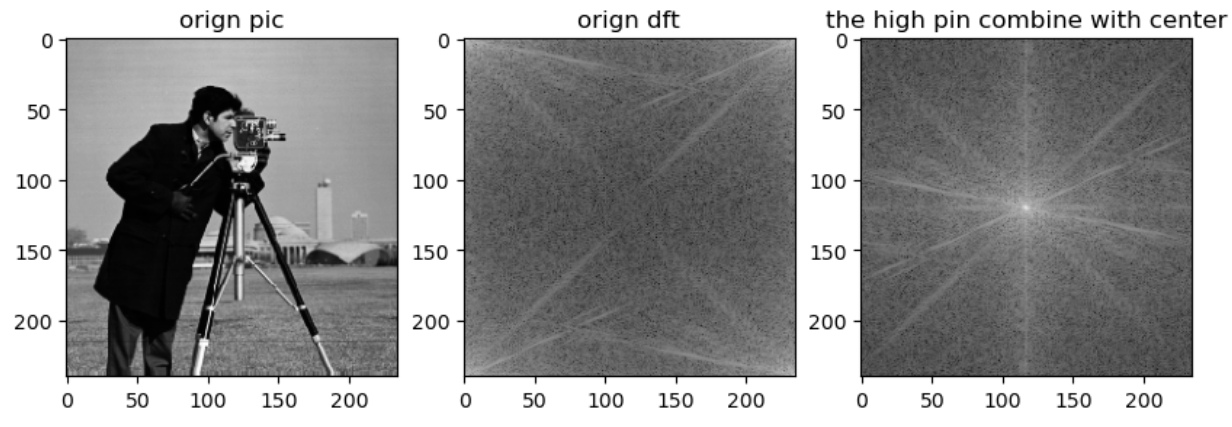
axes[1].set\_title("orign dft")

axes[2].imshow(s2,'gray')

axes[2].set\_title("the high pin combine with center")

plt.show()

**结果：**



1. **图像傅里叶变换平移性质**

**代码：**

import cv2

import numpy as np

import matplotlib.pyplot as plt

img1 = cv2.imread('./img/test01.jpg',0)

rows,cols = img1.shape[:2]

# 原图傅里叶变换

f = np.fft.fft2(img1)

fshift = np.fft.fftshift(f)

s0 = np.log(np.abs(fshift))

M1=np.float32([[1,0,80],[0,1,0]])# 平移矩阵

img\_row = cv2.warpAffine(img1,M1,(cols,rows))

**# 傅里叶变换 img\_row**

**f\_row = np.fft.fft2(img\_row)**

**fshift\_row = np.fft.fftshift(f\_row)**

**s1 = np.log(np.abs(fshift\_row))**

**M2=np.float32([[1,0,0],[0,1,80]])# 平移矩阵**

**img\_col = cv2.warpAffine(img1,M2,(cols,rows))**

**# 傅里叶变换 img\_col**

**f\_col = np.fft.fft2(img\_col)**

**fshift\_col = np.fft.fftshift(f\_col)**

**s2 = np.log(np.abs(fshift\_col))**

**fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)**

**axes[0].imshow(img1,'gray')**

**axes[0].set\_title("orign")**

**axes[1].imshow(s0,'gray')**

**axes[1].set\_title("orign dft")**

**plt.show()**

**fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)**

**axes[0].imshow(img\_row,'gray')**

**axes[0].set\_title("row")**

**axes[1].imshow(s1,'gray')**

**axes[1].set\_title("row dft")**

**plt.show()**

**fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)**

**axes[0].imshow(img\_col,'gray')**

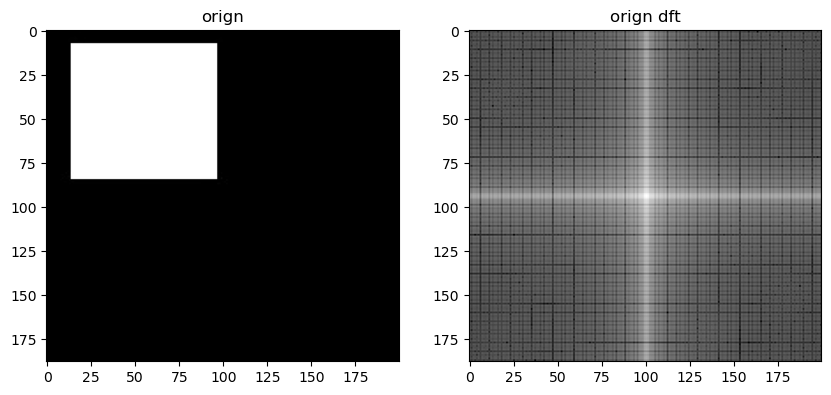
**axes[0].set\_title("col")**

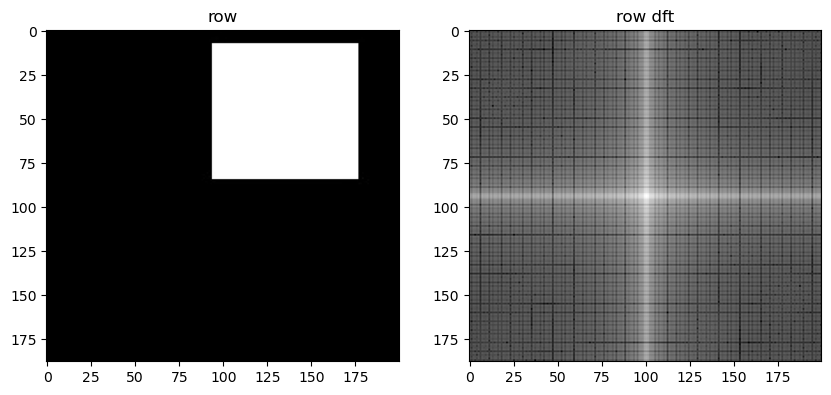
**axes[1].imshow(s2,'gray')**

**axes[1].set\_title("col dft")**

**plt.show()**

**结果：**

****

****

1. **图像傅里叶变换旋转性质**

**代码：**

import numpy as np

import cv2

import matplotlib.pyplot as plt

img\_orign=cv2.imread('./img/test01.jpg',0)

rows,cols=img\_orign.shape[:2]

# 2.1生成旋转矩阵   旋转中心点坐标  角度 缩放比例

M1=cv2.getRotationMatrix2D((cols/2,rows/2),45,1)

# 2.2 进行旋转变换

img\_circle=cv2.warpAffine(img\_orign,M1,(cols,rows))

# 傅里叶变换 orign

f\_orign = np.fft.fft2(img\_orign)

fshift\_orign = np.fft.fftshift(f\_orign)

s\_orign = np.log(np.abs(fshift\_orign))

# 傅里叶变换 旋转

f\_circle = np.fft.fft2(img\_circle)

fshift\_circle = np.fft.fftshift(f\_circle)

s\_circle = np.log(np.abs(fshift\_circle))

fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)

axes[0].imshow(img\_orign,'gray')

axes[0].set\_title("orign pic")

axes[1].imshow(s\_orign,'gray')

axes[1].set\_title("dft orign")

plt.show()

fig,axes=plt.subplots(nrows=1,ncols=2,figsize=(10,8),dpi=100)

axes[0].imshow(img\_circle,'gray')

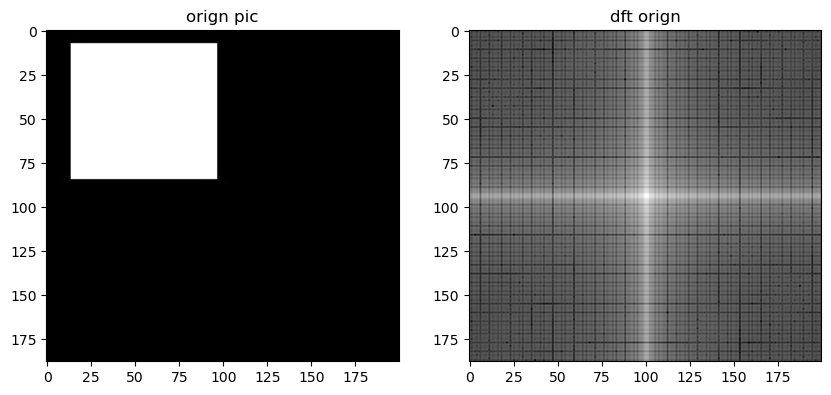
axes[0].set\_title("orign pic")

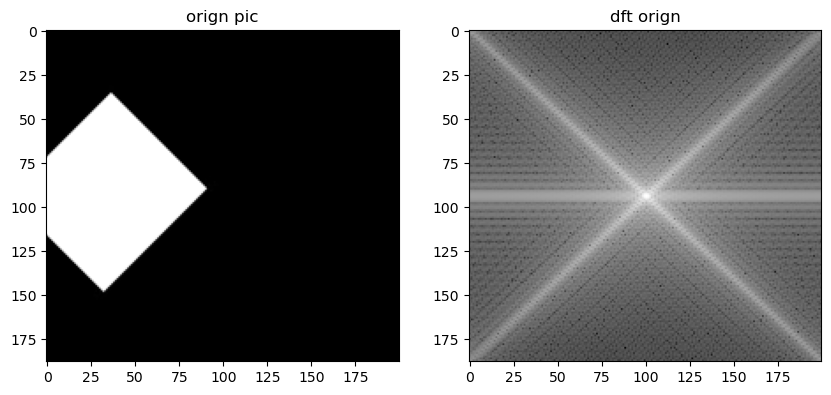
axes[1].imshow(s\_circle,'gray')

axes[1].set\_title("dft orign")

plt.show()

**结果：**

****

****

1. **作业内容**
2. 读取一张自选灰度图像，绘制原始图像及原始图像的傅里叶变换频谱图。
3. 参考实验内容2，创建一个小的白色圆形图像，对其进行X轴和Y轴上的平移，绘制原始图像和平移图像及3图像的傅里叶变换频谱图，进行观察并分析。
4. 参考实验内容3，创建一个白色正方形图像，对其进行45度旋转，绘制原始图像和旋转图像及2图像的傅里叶变换频谱图，进行观察并分析。
5. 读取一张自选灰度图像，绘制原始图像、余弦变换系数及余弦反变换重构图像。
6. **实验要求**
7. 用Python独立编程完成实验和作业内容并给出运行结果。
8. 实验报告书写格式要求规范，内容包括：（1）实验目的；（2）实验内容；（3）程序清单；（4）运行结果；（5）总结（实验结果分析、心得与体会等）。