上海理工大学

光电信息与计算机工程学院

实验报告

实验名称 实验一

课程名称 数字图像处理

姓 名	张翼飞	学 号	1712480237
日 期	2019.11.8	地 点	机房 510
成 绩		教 师	

备注：

1. 实验内容

    1、 读入彩色图像，并显示原图、灰度图、二值图及其标题

    2、 分别对两幅不同大小的图像进行加、减、乘、除运算，并显示四个运算后的结果图像、

    3、 对原灰度图像进行灰度线性变换，分别是变亮、变暗、负片效果，对原灰度图像进行对数和幂指数灰度运算，并显示结果。

    4、 给定一副图像，分别计算其灰度直方图，并进行直方图均衡化，并显示原图、结果图像、原直方图及结果直方图，

    5、 对一副图像进行缩小、放大、旋转、翻转的几何运算

    6、 给定一副图像，计算并显示其傅里叶变化后的频谱图，并计算旋转后图像的频谱图。

    7、 给定一副图像，添加椒盐噪声，然后分别利用均值滤波、平滑滤波进行去噪。

    8、 给定一副图像分别使用索贝尔算子、拉普拉斯算子计算边缘。

2. 实验环境（软件、硬件条件）

Dell inspiron 7577 ，处理器：intel（R）Core（TM）i7-7700HQ CPU@2.8GHz 2.81GHz ，内存 16GB

Win10，matlabR2017b(9.3.0.713579),64-bit

3. 实验方法

Matlab 处理图像的方法：

读入图像，imread（图像）；显示图像，imshow（图像）；将彩色图像信息装换成灰度图像 rgb2gray（图像）；显示二值图像 im2bw（）；

图像相加 imadd（），图像相加可以用于带噪图像多帧叠加平均降噪；

图像相减 imsubtract（），增强图像间的差别，在医学造影方面有重要作用，可用于观察图像变化，用于目标识别，目标跟踪；

图像相乘 immultiply（），图像相除 imdivide（），图像的乘除操作主要用于校正阴影；

灰度线性变化，imadjust（图像，【】,【】）将前一区间的像素点映射到后一区间，道道线性变化,灰度线性变化通过建立灰度映射来调整资源图像的灰度从而达到图像增强的作用；

可以通过 log，幂次对图像进行对数、幂指变换，对数变化将输入中范围窄的低灰度值映射为输出中较宽范围的灰度值,使用非线性的变化使图像的灰度能够真实反映图像的物理特性，增强对比度和扩展对比度，显示和标定轮廓线；

显示灰度直方图 imhist（）；直方图均衡化 histeq（），灰度直方图可以较好地显示原图像灰度的分布情况，图像均衡化后能增加图像的全局对比度；

对图像放大、缩小 imresize（）；图像旋转 imrotate（）；对图像进行上下翻转 flipud（），左右翻转 fliplr（）；

傅里叶变换，表示能将满足一定条件的某个函数表示成三角函数（正弦和/或余弦函数）或者它们的积分的线性组合。

f(t)是 t 的周期函数，如果 t 满足狄里赫莱条件：在一个以 2T 为周期内 f(X)连续或只有有限个第一类间断点，附 f（x）单调或可划分成有限个单调区间，则 F（x）以 2T 为周期的傅里叶级数收敛，和函数 S（x）也是以 2T 为周期的周期函数，且在这些间断点上，函数是有限值；在一个周期内具有有限个极值点；绝对可积。则有下图①式成立。称为积分运算 f(t)的傅立叶变换，

②式的积分运算叫做 F(ω)的傅立叶逆变换。F(ω)叫做 f(t)的像函数，f(t)叫做 F(ω)的像原函数。F(ω)是 f(t)的像。f(t)是 F(ω)原像。

①傅立叶变换

$$F(\omega)=\mathcal{F}[f(t)]=\int_{-\infty}^{\infty} f(t)e^{-iwt}dt$$

②傅立叶逆变换

$$f(t)=\mathcal{F}^{-1}[F(\omega)]=\frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega)e^{iwt}d\omega$$

线性性质、

傅里叶变换的线性，是指两函数的线性组合的傅里叶变换，等于这两个函数分别做傅里叶变换后再进行线性组合的结果。具体而言，假设函数 $f(x)$和 $g(x)$的傅里叶变换 $\mathcal{F}[f]$ 和 $\mathcal{F}[g]$都存在， α和 β为任意常系数，则有

$$\mathcal{F}[\alpha f+\beta g]=\alpha\mathcal{F}[f]+\beta\mathcal{F}[g]$$

尺度变换性质

若函数 $f(x)$的傅里叶变换为 $F(\omega)$，则对任意的非零实数 a，函数 $f_a(x)=f(ax)$ 的傅里叶变换 $F_a(\omega)$存在，且等于$F(a\omega)=\frac{1}{|a|}F\left(\frac{\omega}{a}\right)$对于 $a>0$的情形，上式表明，若将 $f(x)$的图像沿横轴方向压缩 a倍，则其傅里叶变换的图像将沿横轴方向展宽 a倍，同时高度变为原来的 $1/a$。对于 $a<0$的情形，还会使得傅里叶变换的图像关于纵轴做镜像对称。

对称性

若函数 $f(x)$的傅里叶变换为 $F(\omega)$，则存在

$$\mathcal{F}(F(x))=2\pi f(-\omega)$$

加入椒盐噪声 imnoise（），椒盐噪声也称为脉冲噪声，是图像中经常见到的一种，它是噪声，一种随机出现的白点或者黑点，可能是亮的区域有黑色像素或是在暗的区域有白色像素（或是两者皆有)。盐和胡椒噪声的成因可能是影像讯号受到突如其来的强烈干扰而产生、类比数位转换器或位元传输错误等；

均值滤波，也称为线性滤波，其采用的主要方法为邻域平均法。线性滤波的基本原理是用均值代替原图像中的各个像素值，即对待处理的当前像素点（x，y），选择一个模板，该模板由其近邻的若干像素组成，求模板中所有像素的均值，再把该均值赋予当前像素点（x，y），作为处理后图像在该点上的灰度 g（x，y），即 g（x，y）=∑f（x，y）/m m 为该模板中包含当前像素在内的像素总个数。
不足：存在着固有的缺陷，即它不能很好地保护图像细节，在图像去噪的同时也破坏了图像的细节部分，从而使图像变得模糊，不能很好地去除噪声点。

平滑滤波，低频增强的空间域滤波技术。它的目的有两类：一类是模糊；另一类是消除噪音。空间域的平滑滤波一般采用简单平均法进行，就是求邻近像元点的平均亮度值。邻域的大小与平滑的效果直接相关，邻域越大平滑的效果越好，但邻域过大，平滑会使边缘信息损失的越大，从而使输出的图像变得模糊，因此需合理选择邻域的大小；

sobel 算子，在边缘检测中，常用的一种模板是 Sobel 算子。Sobel 算子有两个，一个是检测水平边缘的 ；另一个是检测垂直边缘的 。与 Prewitt 算子相比，Sobel 算子对于象素的位置的影响做了加权，可以降低边缘模糊程度，因此效果更好。
Sobel 算子另一种形式是各向同性 Sobel(Isotropic Sobel)算子，也有两个，一个是检测水平边缘的 ，另一个是检测垂直边缘的 。各向同性 Sobel 算子和普通 Sobel 算子相比，它的位置加权系数更为准确，在检测不同方向的边沿时梯度的幅度一致。将 Sobel 算子矩阵中的所有 2 改为根号 2，就能得到各向同性 Sobel 的矩阵。

由于 Sobel 算子是滤波算子的形式，用于提取边缘，可以利用快速卷积函数， 简单有效，因此应用广泛。美中不足的是，Sobel 算子并没有将图像的主体与背景严格地区分开来，换言之就是 Sobel 算子没有基于图像灰度进行处理，由于 Sobel 算子没有严格地模拟人的视觉生理特征，所以提取的图像轮廓有时并不能令人满意。 在观测一幅图像的时候，我们往往首先注意的是图像与背景不同的部分，正是这个部分将主体突出显示，基于该理论，我们给出了下面阈值化轮廓提取算法，该算法已在数学上证明当像素点满足正态分布时所求解是最优的。

索贝尔算子边缘算子所采用的算法是先进行加权平均，然后进行微分运算，算子的计算方法如下：

$$\Delta_x f(x,y) = [f(x-1,y+1) + 2f(x,y+1) + f(x+1,y+1)] - [f(x-1,y-1) + 2f(x,y-1) + f(x+1,y-1)]$$
$$\Delta_y f(x,y) = [f(x-1,y-1) + 2f(x-1,y) + f(x-1,y+1)] - [f(x+1,y-1) + 2f(x+1,y) + f(x+1,y+1)]$$

Sobel 算子垂直方向和水平方向的模板，前者可以检测出图像中的水平方向的边缘，后者则可以检测图像中垂直方向的边缘。实际应用中，每个像素点取两个模板卷积的最大值作为该像素点的输出值，运算结果是一副边缘图像。

拉普拉斯算子、

拉普拉斯算子是 *n* 维欧几里得空间中的一个二阶微分算子，定义为梯度（▽f）的散度（▽·f）。因此如果 *f* 是二阶可微的实函数，则 *f* 的拉普拉斯算子定义为：

$$\Delta f = \nabla^2 f = \nabla . \nabla f,$$

f 的拉普拉斯算子也是笛卡尔坐标系 *xi* 中的所有*非混合*二阶偏导数：

$$\Delta f = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2}.$$

作为一个二阶微分算子，拉普拉斯算子把 *C* 函数映射到 *C* 函数，对于 *k*≥2 时成立。算子 Δ :*C*(**R**) →*C*(**R**)，或更一般地，定义了一个算子 Δ :*C*(Ω) →*C*(Ω)，对于任何开集 Ω 时成立。

函数的拉普拉斯算子也是该函数的黑塞矩阵的迹

$$(x+a)^n = \sum_{k=0}^{n} \binom{n}{k} x^k a^{n-k}$$

:

$$\Delta f = tr(H(f)).$$

另外，满足▽·▽f=0 的函数 f, 称为调和函数。

二维空间、

其中 **x** 与 **y** 代表 x-y 平面上的笛卡尔坐标：

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

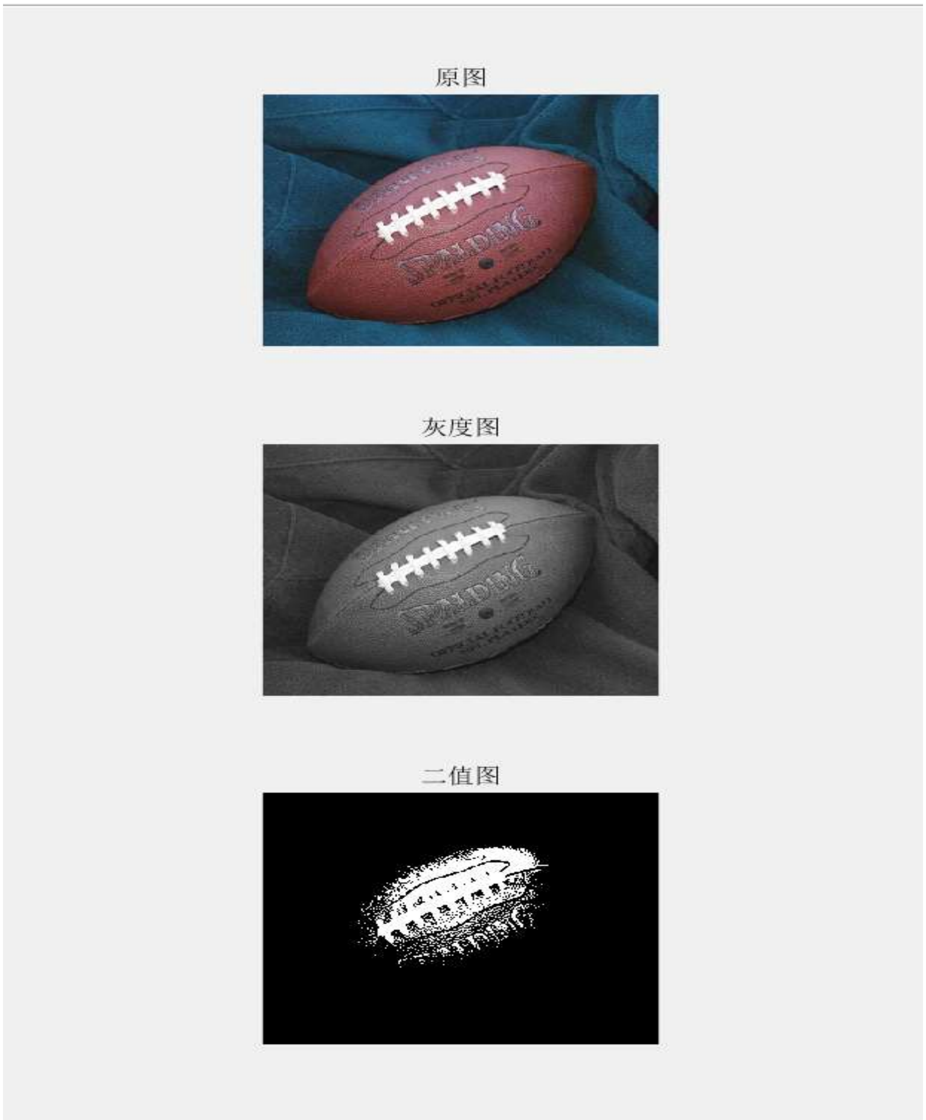
另外极坐标的表示法为：

$$\Delta f = \frac{1}{r} \frac{\partial}{\partial r}\left(r \frac{\partial f}{\partial r}\right) + \frac{1}{r^2} \frac{\partial^2 f}{\partial \theta^2}$$

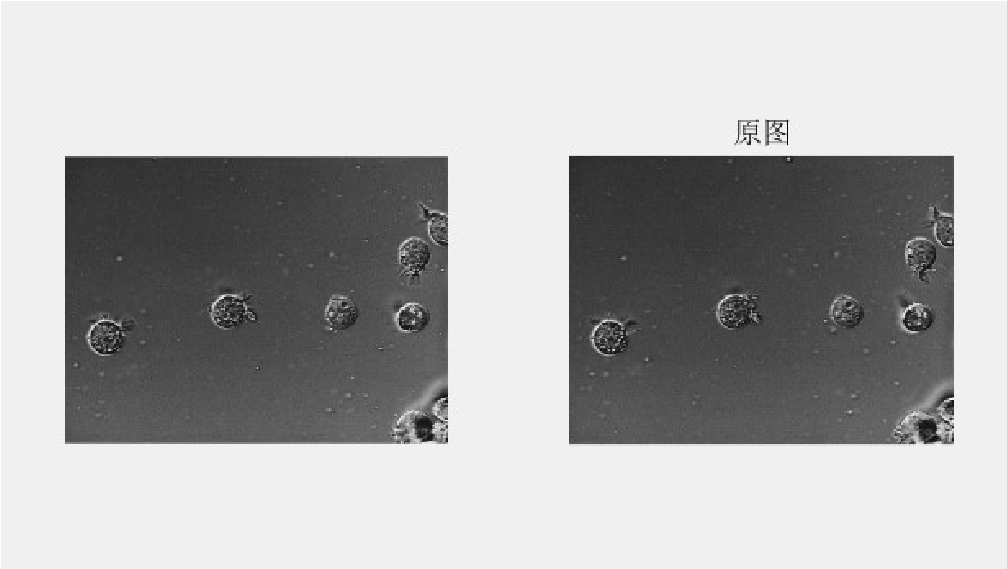
$$= \frac{1}{r} \frac{\partial f}{\partial r} + \frac{\partial^2 f}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2 f}{\partial \theta^2}$$

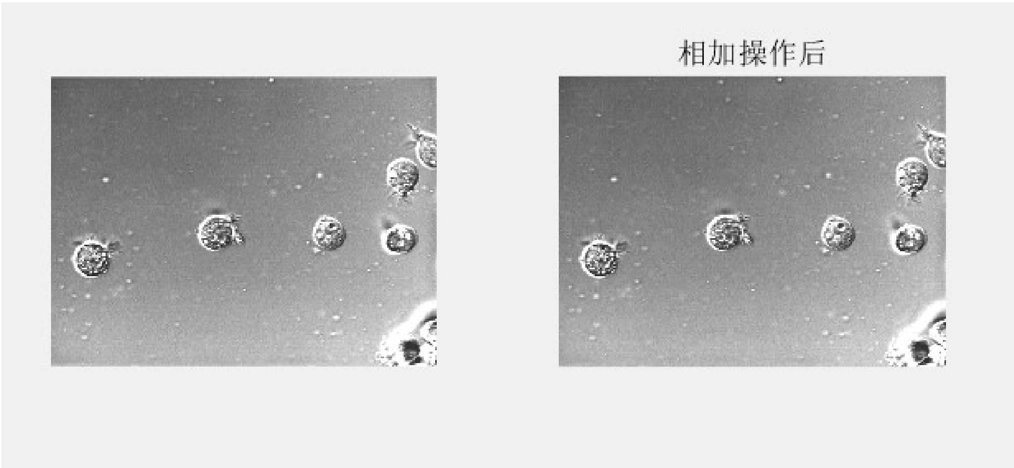
4. 实验分析

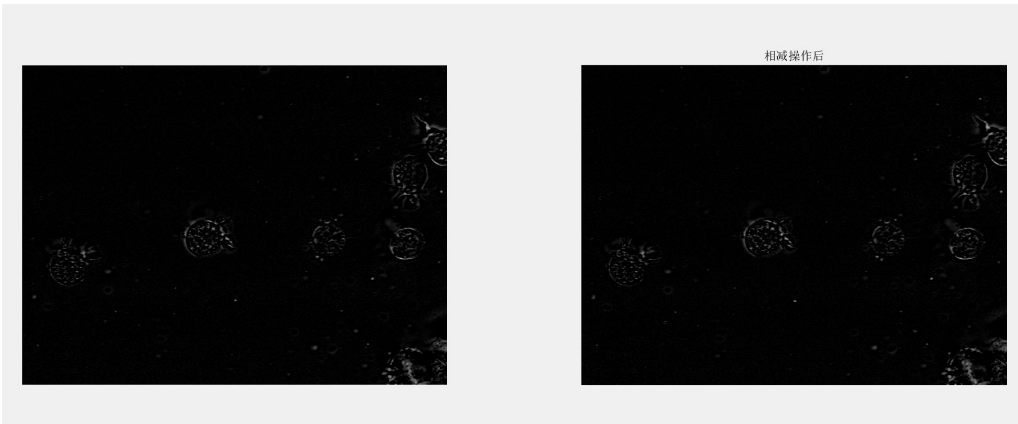
1) 彩色图像显示原图、灰度图、二值图。

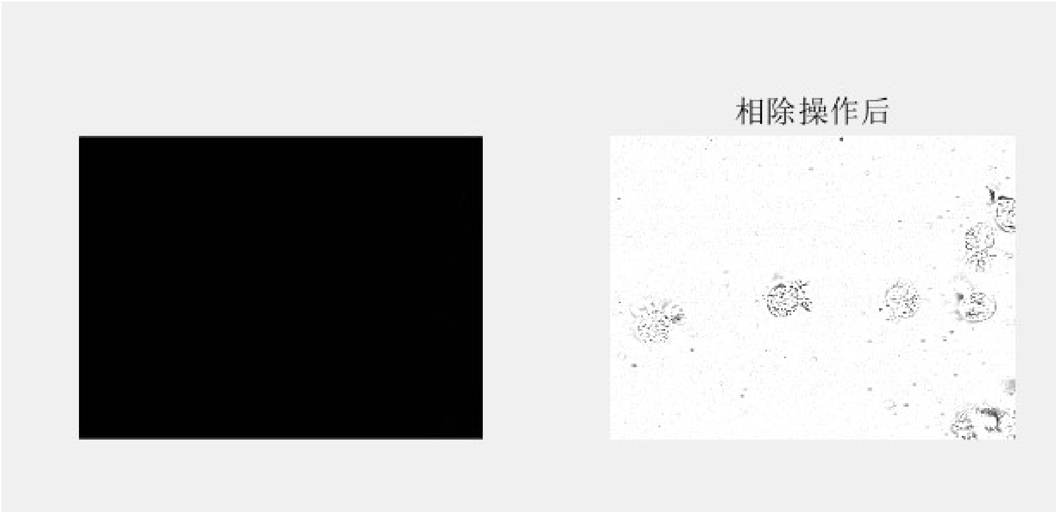


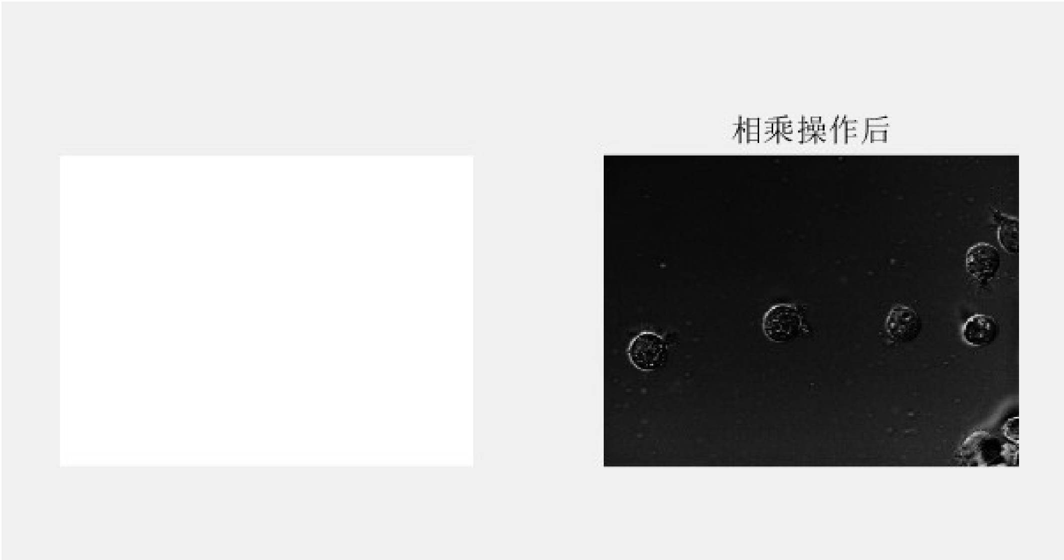
2)　图像间加减乘除，原类型图在左，转换成 double 类型的图在右。





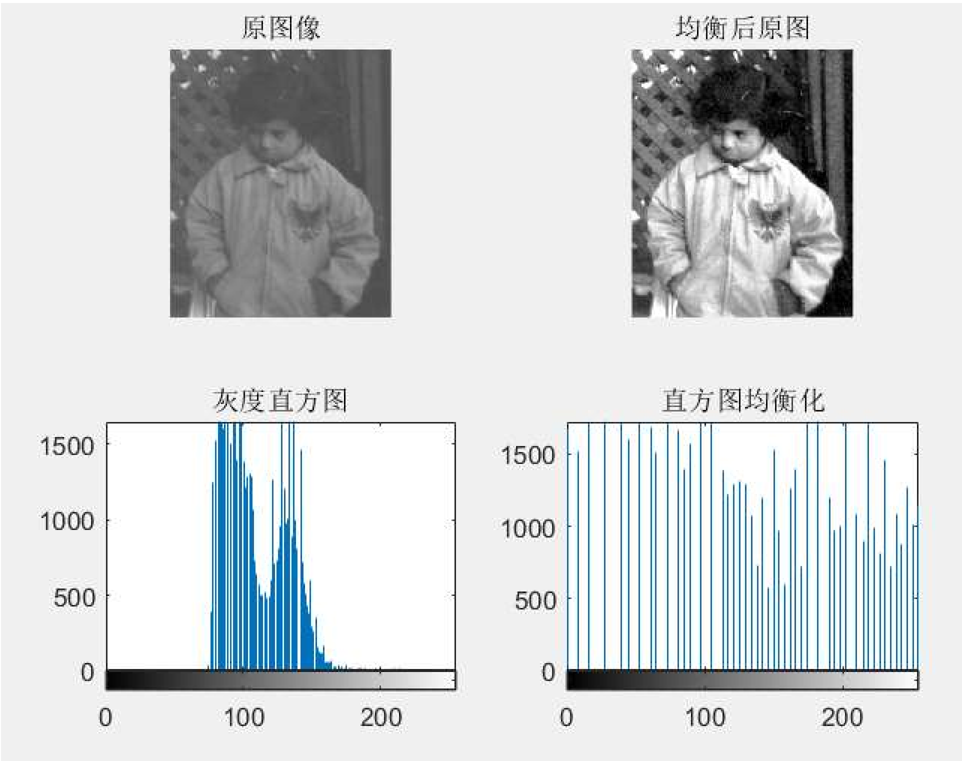



相除操作后


相乘操作后

3) 线性变换和非线性变换

4) 灰度直方图，直方图均衡化



5)　缩小放大旋转翻转

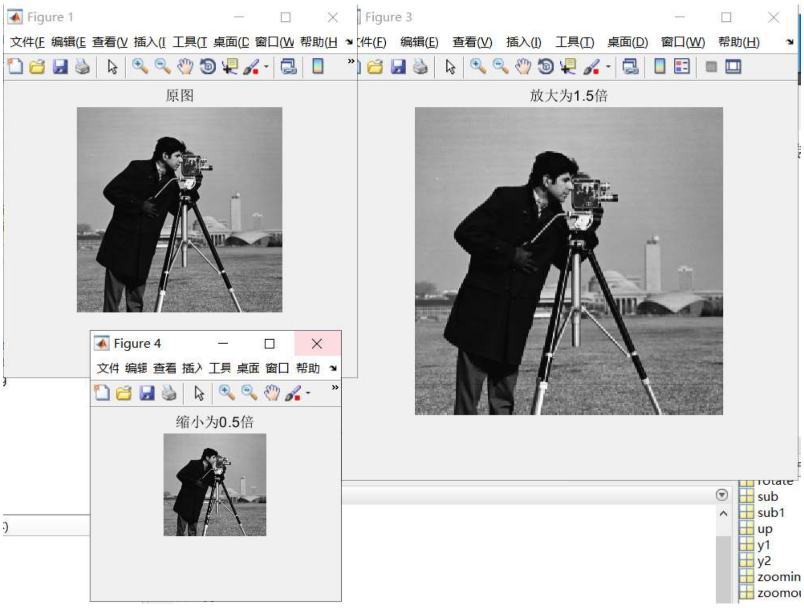
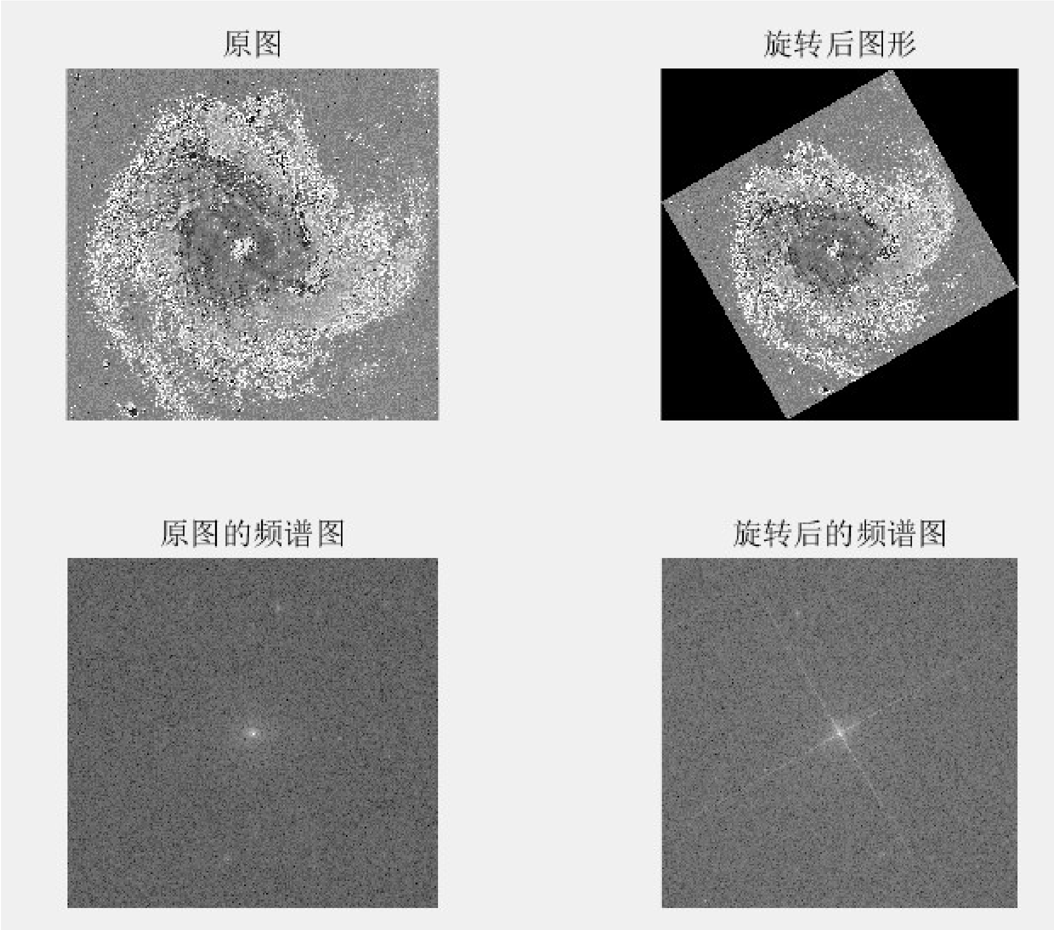


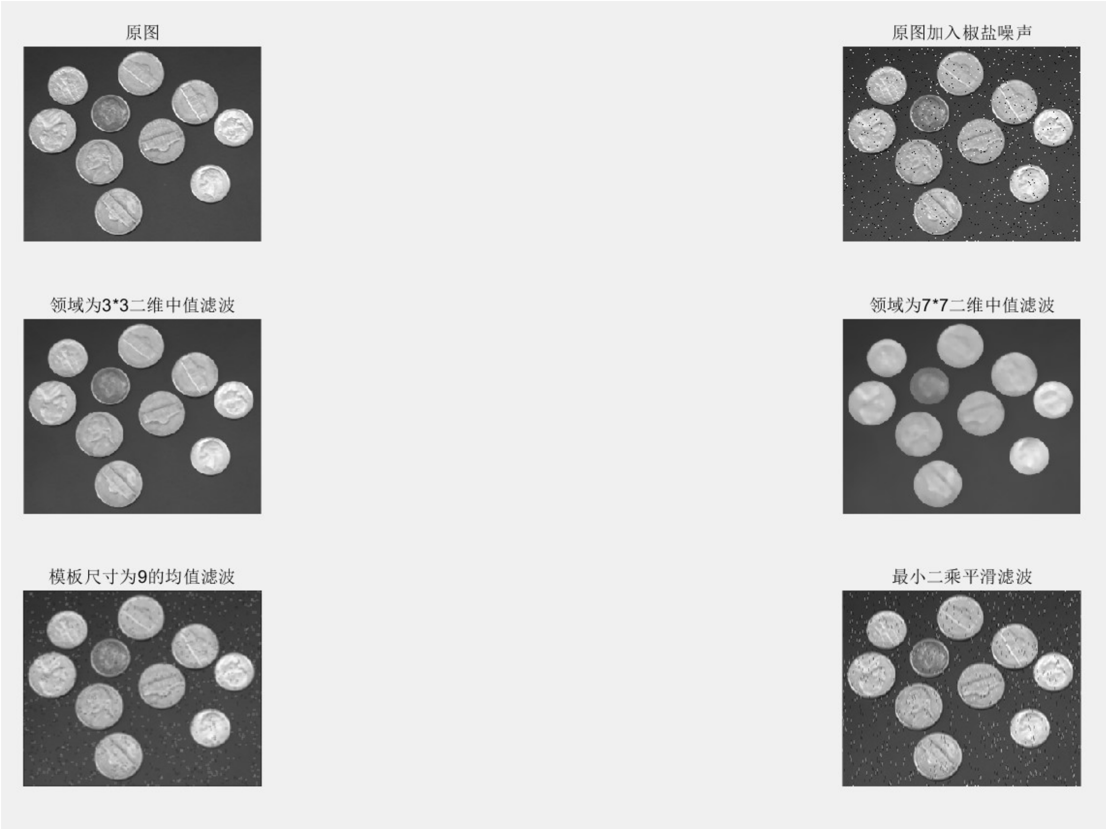
Figure 1
原图
Figure 3
放大为1.5倍
Figure 4
缩小为0.5倍
sub
sub1
up
y1
y2
zoomin


逆时针旋转30度
上下翻转
左右翻转

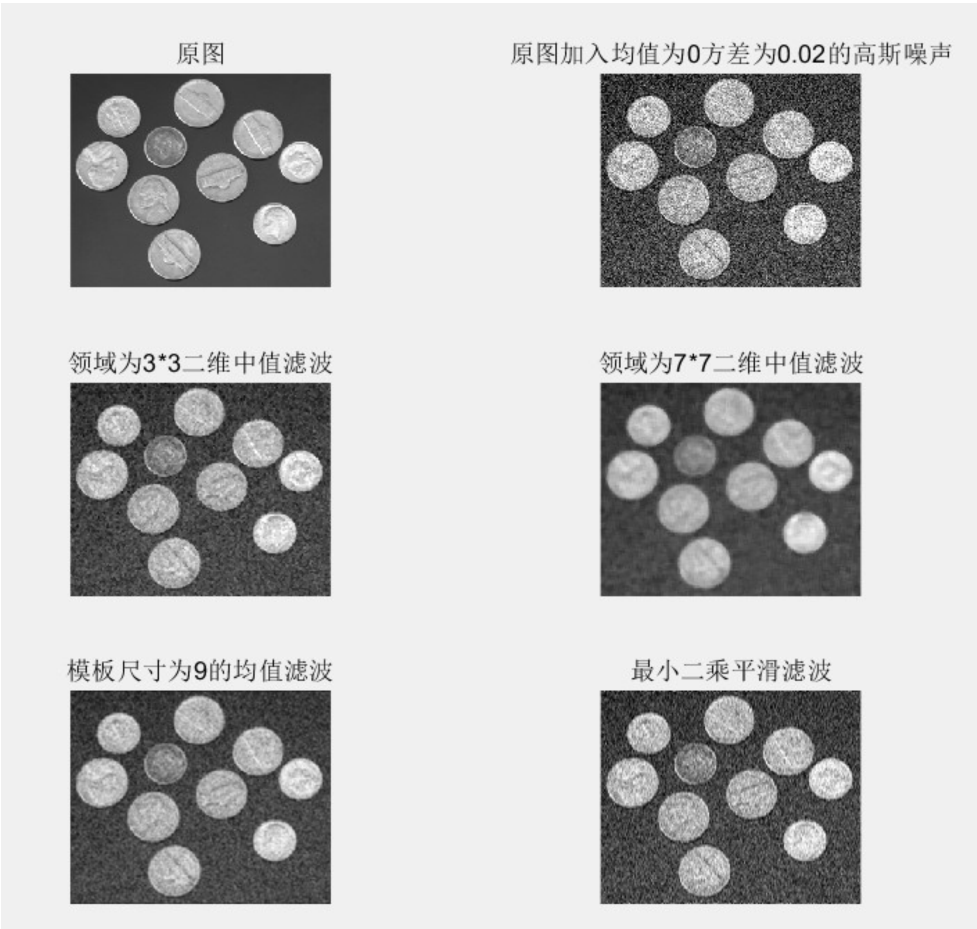
6) 傅里叶变换后的频谱图旋转后的频谱图



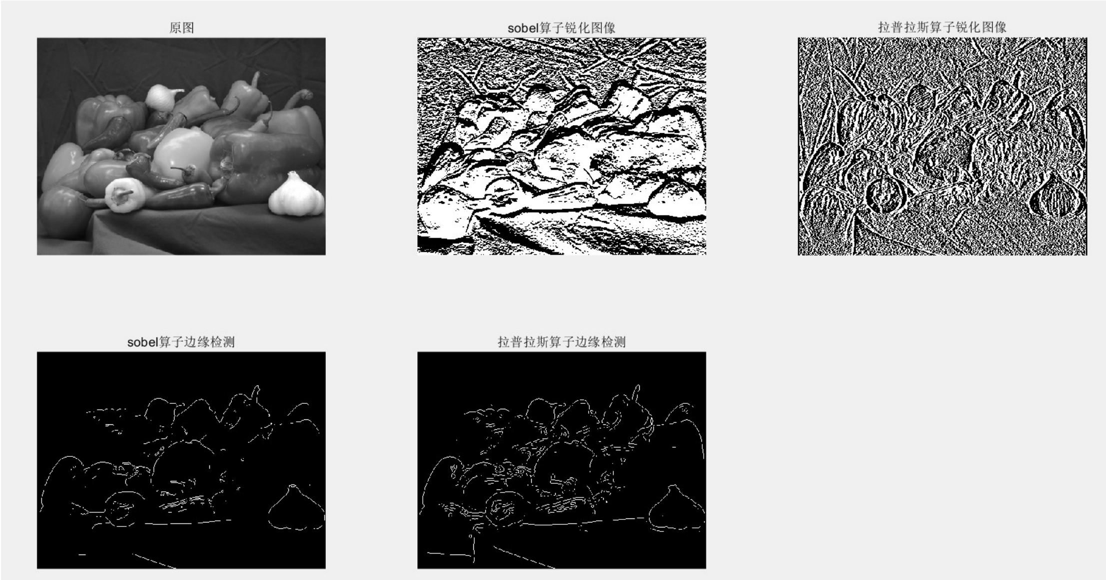
7) 加入椒盐噪声后的均值滤波、中值滤波和平滑滤波



加入高斯噪声后的滤波处理



8) Sobel 算子、拉普拉斯算子计算边缘及图像锐化



5. 实验结论

使用 matlab 读取彩色图像，了解到图像有三维信息，分别为横向像素点个数，纵向像素点个数和 rgb 三个通道，进行某些图像处理之前需要 rgb2gray 转换成二维的灰度图像。

学习使用了 matlab 对图像处理的基本操作，使用观察了图像的灰度直方图，均衡直方图，傅里叶变化后的频谱图，对图像分别使用了均值滤波和平滑滤波，效果都不是很好，滤波应该综合使用才能达到比较好的效果。使用索贝尔算子和拉普拉斯算子计算边缘，尝试了图像锐化和边缘检测，效果比较好。