

深圳大学研究生课程论文

题目 医学图像处理与分析 成绩

专业 生物医学工程 课程名称、代码 151022020007

年级 研究生一年级 姓名 李婉君

学 号 2140220212 时间 2015 年 1 月

任课教师 汪天富

一、绪论

读入一幅灰度图象，编写程序显示图象中任一像素点的灰度值。

二、数字图像基础

编程实现 RGB 彩色空间到 HSV 彩色空间的转换，并通过调整 H、S、V 的值实现图像亮度、色度和色饱和度的调节。

三、图像增强技术

1. 灰度线性变换、分段线性变换
2. 灰度直方图均衡、规格化
3. 图象卷积：各种模板实现低通、高通滤波
4. 图象中值滤波、最大值滤波、最小值滤波、模板选择滤波
5. 伪彩色图象增强

四、图像运算

1. 图像的加、减、乘、除和逻辑运算。
2. 图像的镜像变换、图像的缩放（灰度插值）变换

五、图像正交变换

编程实现图像的 FFT 变换和反变换

六、频域图像增强技术

1. 理想低通滤波、巴特沃斯低通滤波
2. 理想高通滤波、巴特沃斯高通滤波
3. 同态滤波

七、数学形态学

1. 实现基本的形态学运算：腐蚀、膨胀、开、闭，交互式输入不同类型的结构元观察效果
2. 利用数学形态学运算实现图像的骨架化、细化

八、图像恢复

用逆滤波方法和维纳滤波方法恢复运动模糊图像（加噪声和不加噪声）

九、灰度图像分割

1. 阈值化算法：手动阈值、自动阈值（大津阈值）
2. 区域生长
3. 区域分裂合并

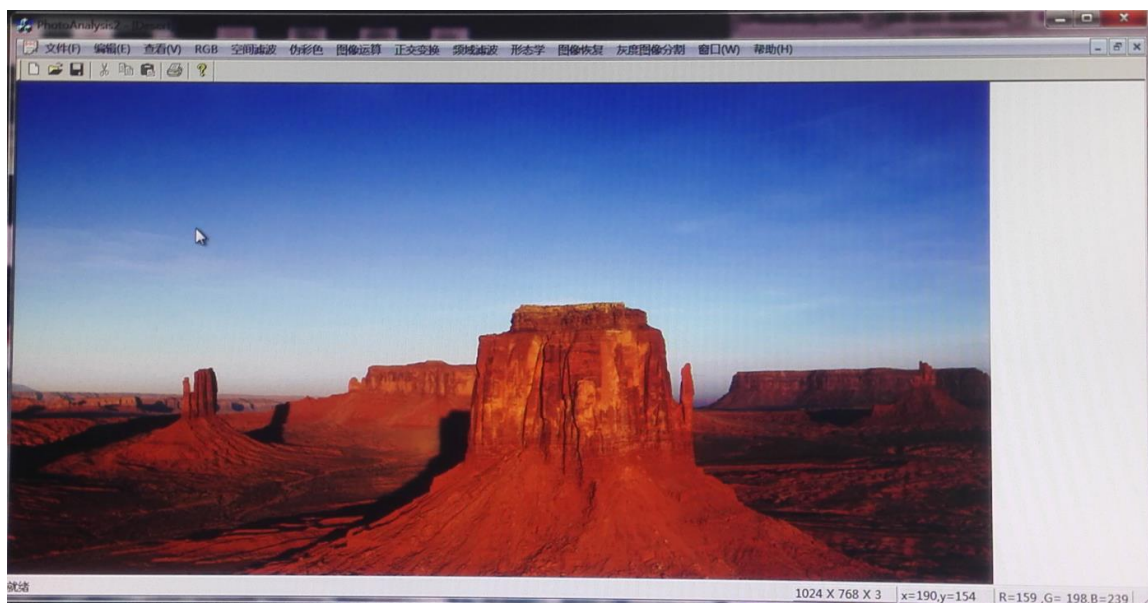
一、绪论

实验目的：读入一幅灰度图像，编写程序显示图像中任一像素点的灰度值。

实验原理：本次实验共考虑两种图像，灰度图像与彩色图像。其中灰度图像有 8 位和 24 位，其 RGB 是相等的。彩色图像 RGB 不相等。

算法解释：打开图像部分是用 `fileopen` 函数实现的，在 `fileopen` 中通过对话框获得图像路径，得到图像信息，在 `ondraw` 函数中显示出来。读取 RGB 是通过 `mousemove` 函数获得鼠标位置，在小框中显示出 RGB，并在状态栏显示坐标。

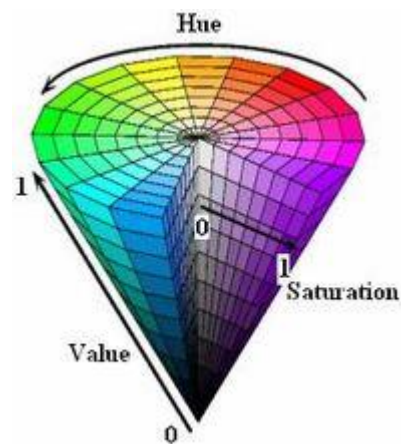
结果分析：（显示位于右下角）



二、数字图像基础

实验目的：编程实现 RGB 彩色空间到 HSV 彩色空间的转换，并通过调整 H、S、V 的值实现图像亮度、色度和色饱和度的调节。

实验原理：HSV(Hue, Saturation, Value)是根据颜色的直观特性创建的一种颜色空间, 也称六角锥体模型(Hexcone Model)。这个模型中颜色的参数分别是：色调（H），饱和度（S），亮度（V）。色调 H，用角度度量，取值范围为 $0^{\circ} \sim 360^{\circ}$ ，从红色开始按逆时针方向计算，红色为 0° ，绿色为 120° ，蓝色为 240° 。它们的补色是：黄色为 60° ，青色为 180° ，品红为 300° 。饱和度 S，取值范围为



0.0~1.0, 值越大, 颜色越饱和。亮度 V, 取值范围为 0(黑色)~255(白色)。

设 $m = \max(r, g, b)$, $n = \min(r, g, b)$, 其中 r, g, b 分别是归一化的 RGB 颜色空间中的值 h 值域为 $[0, 360]$, v, s 值域为 $[0, 1]$ 。

(1) 从 RGB 颜色坐标转换到 HSV 颜色坐标

$$v = m$$

$$s = \begin{cases} (m-n)/m, & m \neq 0 \\ 0, & m = 0 \end{cases}$$

$$h = \begin{cases} 0, & s = 0 \\ 60(g-b)/(m-n), & r = m \& g \geq b \\ 360 + 60(b-r)/(m-n), & r = m \& g < b \\ 120 + 60(b-r)/(m-n), & g = m \\ 240 + 60(r-g)/(m-n), & b = m \end{cases}$$

(2) 从 HSV 颜色坐标到 RGB 颜色坐标的转换

$$\text{if } (h = 360), h = 0$$

$$i = \text{floor}(h/60), f = h/60 - i$$

其中 floor 是取整运算, i 为不大于其的最大整数。

$$p = v * (1 - s)$$

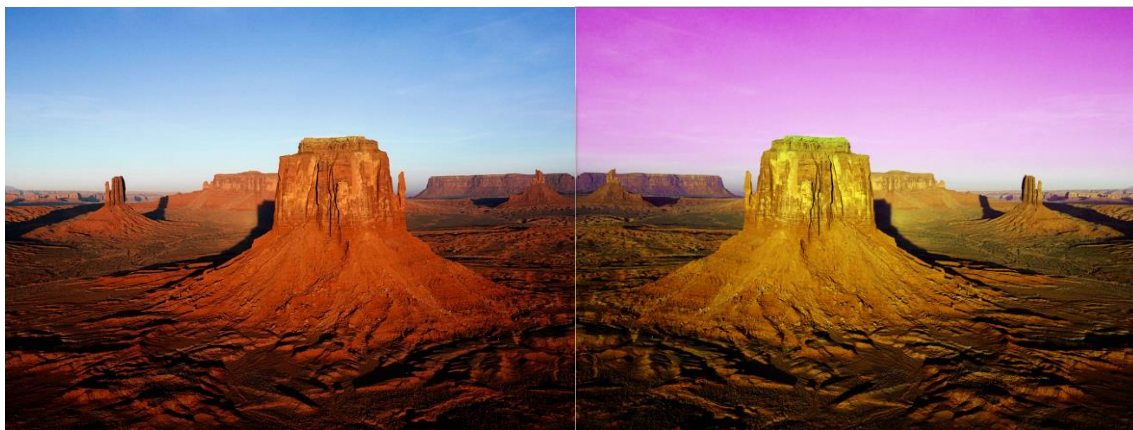
$$q = v * (1 - s * f)$$

$$t = v * (1 - s * (1 - f))$$

$$\begin{cases} r = v, g = t, b = p, & i = 0 \\ r = q, g = v, b = p, & i = 1 \\ r = p, g = v, b = t, & i = 2 \\ r = p, g = q, b = v, & i = 3 \\ r = t, g = p, b = v, & i = 4 \\ r = v, g = p, b = q, & i = 5 \end{cases}$$

算法解释: 设置弹出对话框, 调节拉杆的形式进行 H, S, V 调节, 释放拉杆即在图像中显示变化。

结果分析: 调节 H, 不改变 S 和 V 时, 亮度不变, 色彩的种类不变, H 从 0 度到 360 度变化, 可以实现由红色变成橙色、黄色、绿色、青色、蓝色、紫色或者洋红色。如下对比图。



修改 S 饱和度。使颜色更加饱和。修改亮度，可以使图像更加明亮。如下图。



三、图像增强技术

实验目的：1. 灰度线性变换、分段线性变换

2. 灰度直方图均衡、规格化

3. 图象卷积：各种模板实现低通、高通滤波

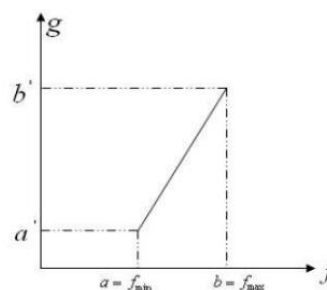
4. 图象中值滤波、最大值滤波、最小值滤波、模板选择滤波

5. 伪彩色图象增强

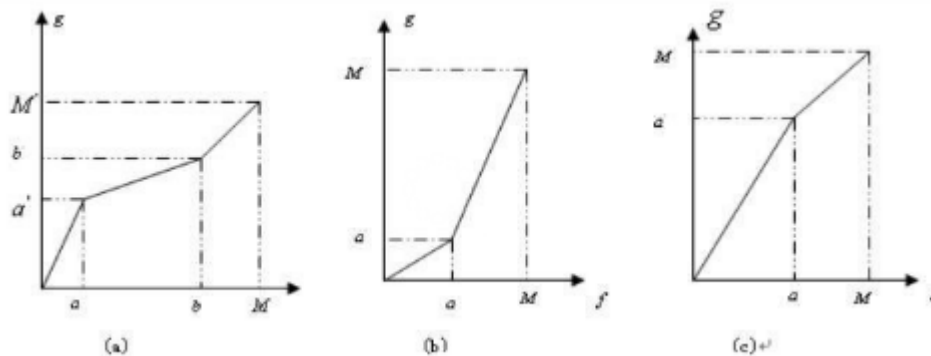
实验原理：

1. 灰度线性变换：通过修改 $p(r)$ 达到增强图像的目的。使格点灰度按照 $S=T(r)=k*r+b$ 进行变换。

分段线性变换：线性拉伸将原始输入图像中的灰度值不加区别的扩展。在实际应用中，为了突出图像中感兴趣的研究对象，常要求局部拉伸某一范围的灰度值，或对不同范围的灰



度值进行不同的拉伸变换，即分段线性拉伸。分段线性拉伸是仅将某一范围的灰度值进行拉伸，其余范围实际上是被压缩。



$$g(x,y) = \begin{cases} \frac{a'}{a} \cdot f(x,y), & 0 \leq f(x,y) < a \\ \frac{b'-a'}{b-a} \cdot (f(x,y)-a) + a', & a \leq f(x,y) < b \\ \frac{M'-a'}{M-a} \cdot (f(x,y)-b) + b', & b \leq f(x,y) < M \end{cases}$$

2. 灰度直方图均衡、规格化

直方图均衡：直方图均衡化的目的是将原始图像的直方图变为均衡分布的形式，即将一已知灰度概率密度分布的图像，经过某种变换变成一幅具有均匀灰度概率密度分布的新图像，从而改善图像的灰度层次。它的基本思想是把原始图像的直方图变换成均匀分布的形式，这样就增加了像素灰度值的动态范围，从而达到了增强图像整体对比度的效果。

直方图规格化：（1）参考图像的直方图，通过变换，使两幅图像的亮度变化规律尽可能地接近；（2）特定函数形式的直方图，通过变换，使变换后的图像亮度变化规律可能地服从这种函数的分布。直方图规定化的原理：对两个直方图都做均衡化，变成相同的归一化的均匀直方图，以此均匀直方图为媒介，再对参考图像做均衡化的逆运算。

3. 图象卷积：各种模板实现低通、高通滤波

线性滤波器是线性系统和频域滤波概念在空域的自然延伸。其特征是结果像素值的计算由下列公式定义： $R = w_1z_1 + w_2z_2 + \dots + w_nz_n$

其中： $w_i \quad i = 1, 2, \dots, n$ 是模板的系数， $z_i \quad i = 1, 2, \dots, n$ 是被计算像素及其邻域像素的值。

4. 图象中值滤波、最大值滤波、最小值滤波、模板选择滤波

使用模板进行结果像素值的计算，结果值直接取决于像素邻域的值，而不使用乘积和的计算 $R = f(z_1, z_2, \dots, z_n)$

5. 伪彩色图象增强

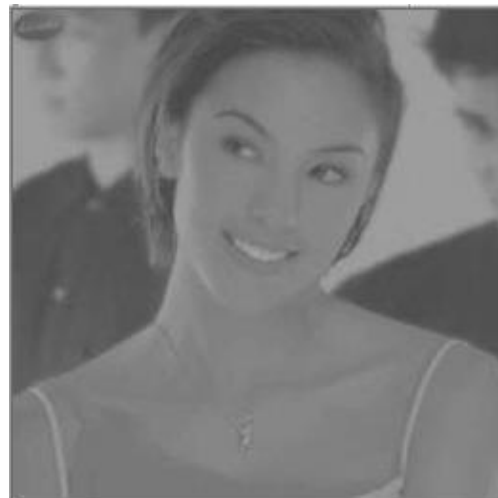
由于人眼对灰度图像的分辨是很有限的，但对颜色的分辨是非常丰富的，固可以用为彩色来使图像达到增强的目的。彩色到灰度使用 $g=R*0.29+G*0.57+B*0.14$ 的公式实现的。灰度到彩色是将灰度分段，每一段各自变换的。

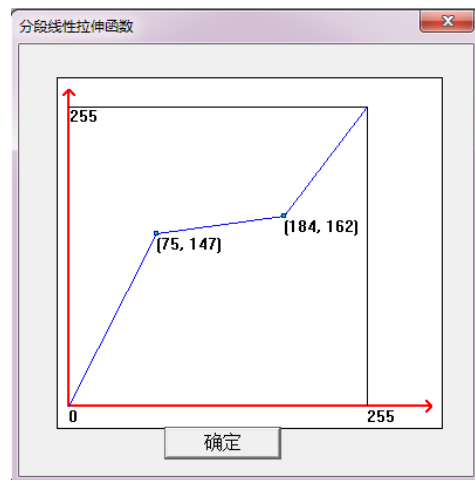
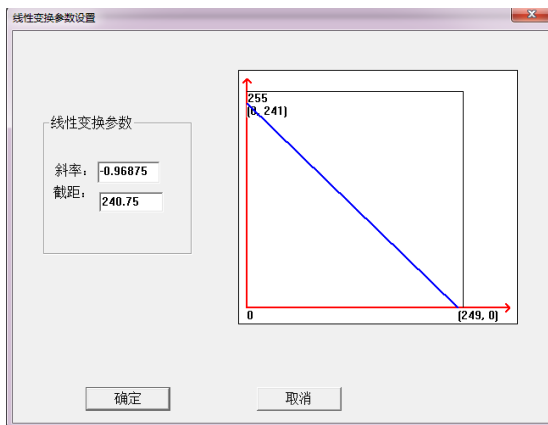
算法解释：

1. 灰度线性变换是以对话框形式，可以在对话框上绘图，得到需要进行变换的线性系数以及截距，放开鼠标即有显示。分段线性变换操作同上，只是有预先设置线段，有两个可操作的点。
2. 直方图均衡点击即有显示，是使图像灰度分布更加均匀的。直方图匹配与灰度拉伸操作方法类似。
3. 锐化空间滤波可通过对话框设置模板系数。
4. 图象中值滤波、最大值滤波、最小值滤波、均值滤波可通过对话框设置模板大小。
5. 伪彩色做了灰度到彩色以及彩色到灰度的变换。

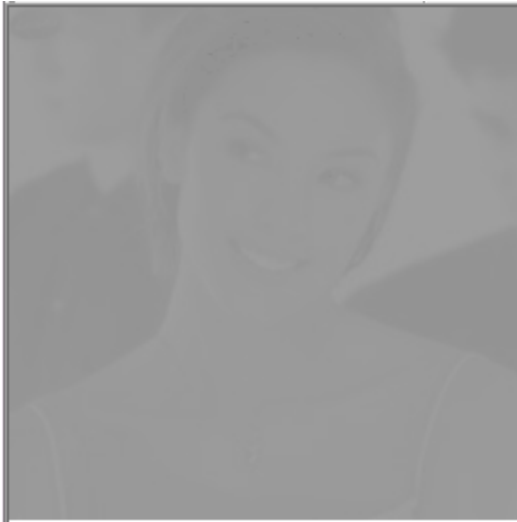
结果分析：

1. 灰度线性变换：

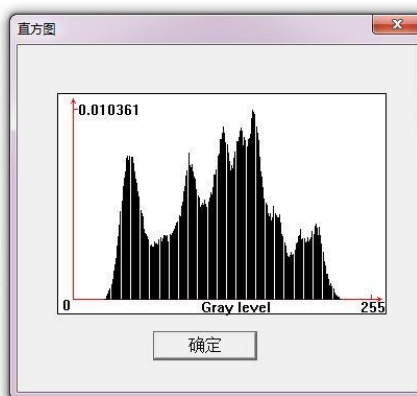


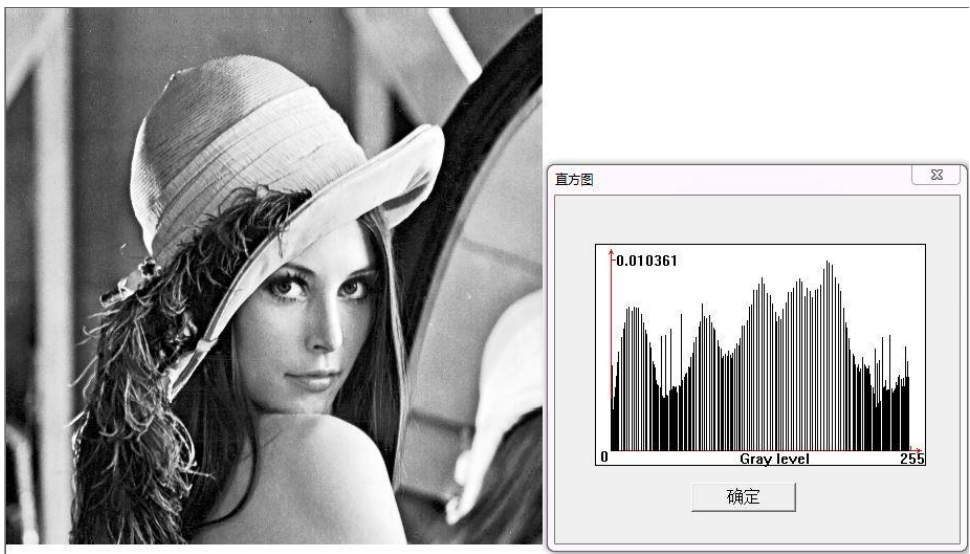


分段拉伸变换：将图片灰度集中分布的灰度值压缩，导致细节大量丢失。



2. 直方图均衡，直方图规定化

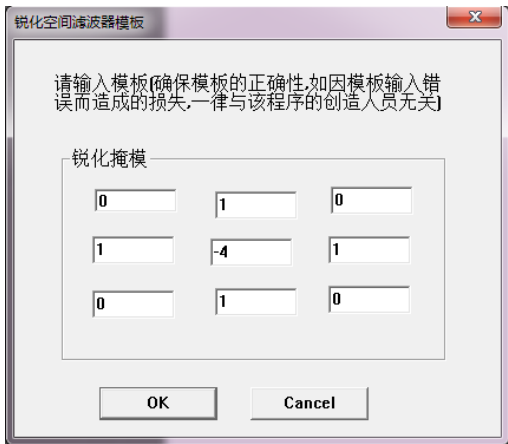




均衡后的图像及直方图。可以看出灰度分布更加均匀不集中了。

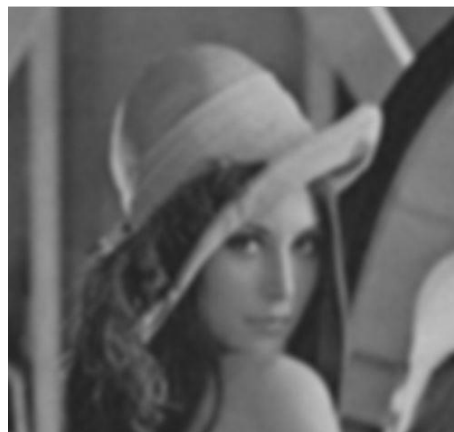
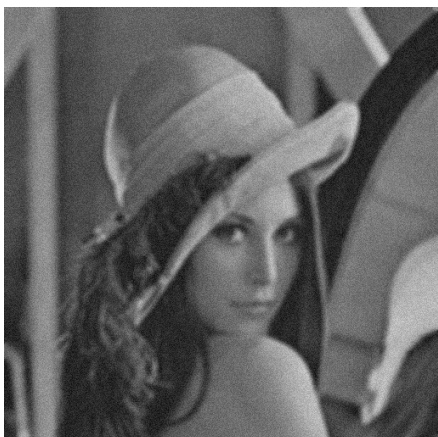
直方图规定是对两个直方图作均衡，弥补了均衡不能交互处理的缺憾。

3. 锐化空间滤波



左图为所用的模板。可以看出边缘被比较好的保留下来，而细节大量丢失。

4. 均值滤波



左图为有运动模糊的图片，右图为使用 9*9 均值滤波模板后的图像。消除了运动噪声，但是使图片变得模糊，丢失了很多细节。



最大值滤波，最小值滤波

左为最大值滤波，右为最小值滤波（5*5 模板）。最大值滤波是图像亮度提升，最小值滤波是图像变暗。

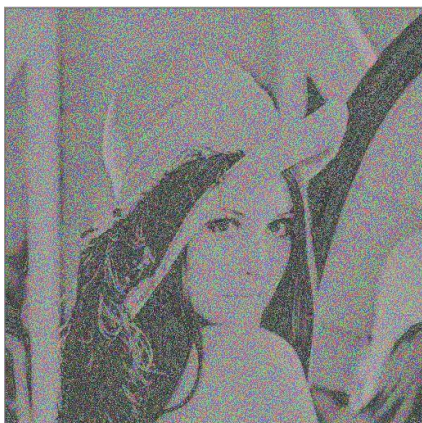
中值滤波



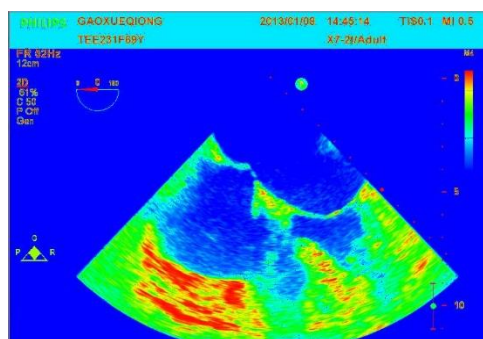
很好地消除了椒盐噪声。适合去除斑点噪声，且根据斑点的大小

选择合适的滤波模板也很重要，防止图像过度模糊。

5. 伪彩色图像处理



转换为 24 位灰度图没有问题，8 位会显示不清楚。



灰度转彩色之后，可以明显地看出有很多细节被显现出来了。

四、图像运算

实验目的：1. 图像的加、减、乘、除和逻辑运算。

2. 图像的镜像变换、图像的缩放（灰度插值）变换。

实验原理:

1. 加法运算是对两幅图像的灰度值进行线性叠加，线性系数之和为 1。实现二次曝光。

图像相减是常用的图像处理方法，用于检测变化及运动物体。图像相减可用于去除一幅图像中所不需要的加性图案，加性图案可能是缓慢变化的背景阴影、周期性的噪声或在图像上每一像素处均已知的附加污染等。在计算用于确定物体边界位置的梯度时，也要用到图像相减。在可控制的条件下，如工业视觉环境下，这种称之为差分方法的简单处理与阈值化处理往往是建立机器视觉系统最有效的方法之一。对于不可控制的环境，如室外场景，在利用差分方法时需要考虑背景的更新机制，消除（补偿）因天气、光照等因素造成的影响。

乘法运算可以用来实现掩膜（mask）处理，即屏蔽掉图像的某些部分。此外由于时域的卷积和相关运算与频域的乘积运算对应，因此乘法运算有时也被用来作为一种技巧来实现卷积或相关处理。

除法运算可用于校正成像设备的非线性影响，在特殊形态的图像（如断层扫描等医学图像）处理中用到。在光谱分析中用于得到比率图像。

求补，即取反，补图像与原图像灰度之和为 255。

异或运算，将两幅图像的灰度值进行异或。

或运算，是将灰度值求或，求并集。

与运算，是灰度值求与，得交集。

2. 几何运算，镜像变换是将数据区反过来重新输出，按 Y 或 X，放缩变换，需要改变文件头，几何运算中灰度级插值是必不可少的组成部分，因为图像一般用整数位置处的像素来定义，而几何变换中， $g(x, y)$ 的灰度值一般由处在非整数坐标上的 $f(x, y)$ 的值来确定，即 g 中的一个像素一般对应于 f 中的几个像素之间的位置。反过来看也是一样，即 f 中的一个像素往往被映射到 g 中的几个像素之间的位置。最简单的插值方法是最近邻插值，即选择离它所映射到的位置最近的输入像素的灰度值为插值结果。复杂一点的方法是双线性插值。

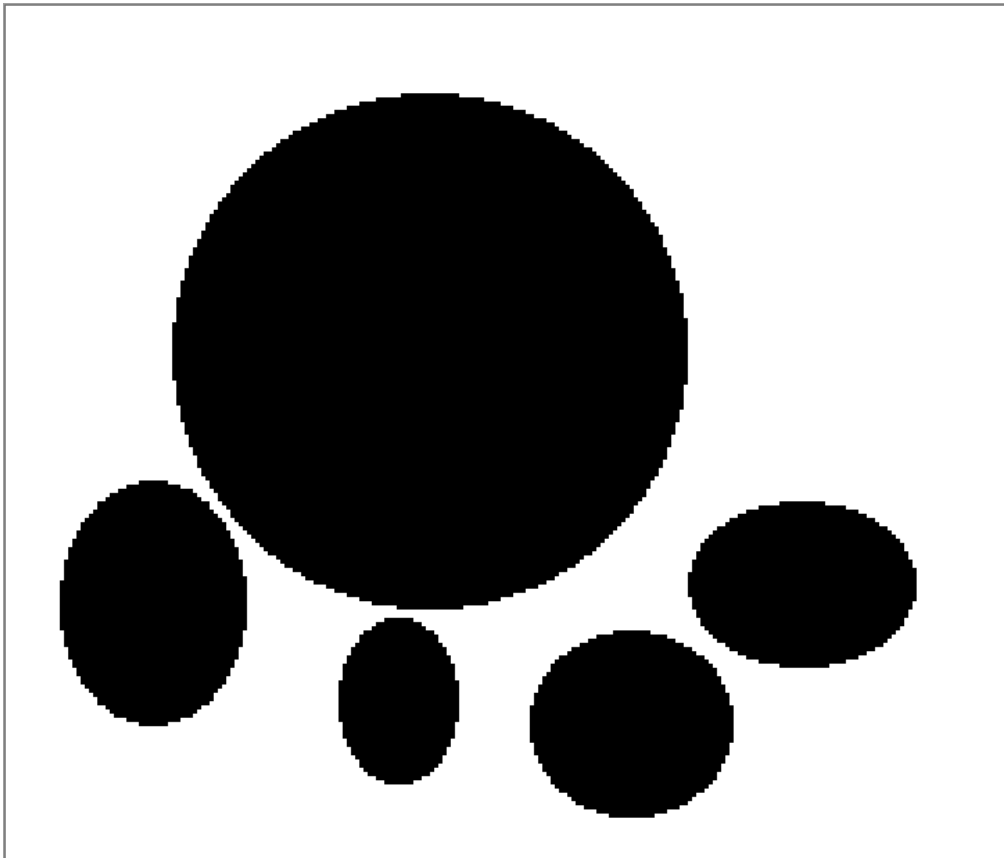
算法解释：几何运算中镜像可以直接选择显示，放缩，使用下拉菜单选择放缩的方法，可选近邻，双线性和立方三种，需要输入目标高宽，不可连续使用。

结果分析： 仅加法支持彩色图像，其余请用灰度图像。

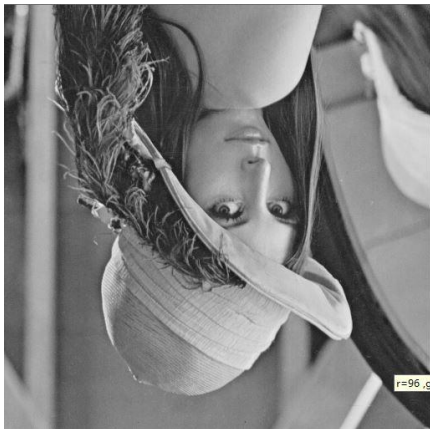
将 1024*1024 的图像用双线性插值的方法缩小为 100*100 的图像



放大：放大三倍后，边缘呈锯齿状。



镜像变换



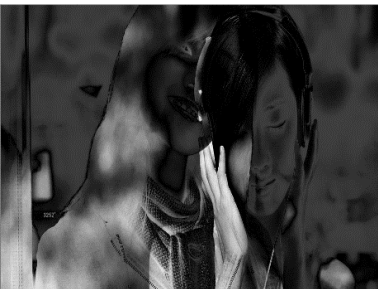
图像相加：第一幅为结果图。



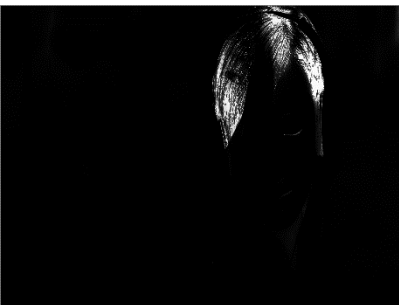
图像相减：增加被减图像低灰度的可见性。



图像相乘：结果为上面第 1, 3 幅图像相乘的结果。较亮的是两幅图共同亮的地方。



图像相除：



五、图像正交变换

实验目的：编程实现图像的 FFT 变换和反变换

实验原理：假设输入的影像 $s[n, m]$ 水平方向长度是 N ，垂直方向长度是 M

二维离散傅立叶变换定义为

$$S(j\omega_1, j\omega_2) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} s[n, m] e^{-j(\omega_1 n + \omega_2 m)}$$

二维离散傅立叶逆变换定义为

$$s[n, m] = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(j\omega_1, j\omega_2) e^{j(\omega_1 n + \omega_2 m)} d\omega_1 d\omega_2$$

同样为了方便，我们可将上述两式改为向量形式

其中， $\vec{n} = (n, m)^T$ 。

要求 N 为 2 的幂，即 $N=2^M$ ，M 为正整数。将输入时间序列按奇、偶抽取

$$S(j\vec{\omega}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} s(\vec{n}) e^{j\vec{\omega}^T \vec{n}} d\vec{n},$$

$$s(\vec{n}) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) e^{-j\vec{\omega}^T \vec{n}}$$

$$f(x) = \begin{cases} f_1(x) + F_2(u) + w_N^u F_2(u - \frac{N}{2}) & 0 \leq u \leq \frac{N}{2} - 1 \\ F_1(u - \frac{N}{2}) - w_N^u F_2(u - \frac{N}{2}) & \frac{N}{2} \leq u \leq N - 1 \end{cases}$$

算法解释：利用 FFT 蝶形流程图算法在计算机上实现快速付里叶变换必须解决如下问题：

迭代次数 r 的确定、对偶节点的计算、加权系数 的计算、重新排序问题。

(1) 迭代次数 r 的确定

由蝶形流程图可见，迭代次数与 N 有关。r 值可由下式确定 $r = \log_2 N$ ，式中 N 是变换序列的长度，对于基 2 的蝶形程图，N 是 2 的整数次幂。例如，序列长度为 8 则要三次迭代，序列长度为 16 时就要 4 次迭代。

(2) 对偶节点的计算

在流程图中把标有 $x_i(k)$ 的点称为节点。其中下标 i 为列数，也就是第几次迭代，例如， $x_1(k)$ 则说明它是第一次迭代的结果。k 代表流程图中的行数，也就是序列的序号数。其中每节点的值均是用前一 节点对计算得来的。例如， $x_1(0)$ 和 $x_1(4)$ 均是 $x(0)$ 和 $x(4)$ 计算得来的。在蝶形流程图中，把具有相同来源的一对节点叫做对偶节点。如： $x_1(0)$ 和 $x_1(4)$ 就是一对对偶节点，因为它们均来源于 $x(0)$ 和 $x(4)$ 。对偶节点的计算也就是求出在每次迭代小对偶节点的间隔。由流程图可见，第一次迭代的间距为计 $N/2$ ，第二次迭代的节距为 $N/4$ ，第三次迭代的节距为人 $N/23$ 等等。由以上分析可得到对偶节点的计算方法。如果某一节点为 $x_i(k)$ ，那么它的对偶节点为

$$x_i(k - \frac{N}{2^i})$$

(3) 加权系数 W_N^p 的计算

W_N^p 的计算主要是确定 p 值。 p 值可用下述方法求得：把 k 值写成 r 位的二进制数 (k 序列的序号数， r 是迭代次数)；把这个二进制数右移 $r-1$ 位，并把左边的空位补零 (结果仍为 r 位)；把这个右移后的二进制数倒序；把倒序后的二进制数转换维十进制数就得到 p 值。

(4) 重新排序

由蝶形流程图可见，如果序列 $x(n)$ 是按顺序排列的，经过蝶式运算后，其变换序列是倒序排列的；反之，如果 $x(n)$ 是倒序排列的，那么输出就是顺序排列的。因此，为了便于输出使用，最好加入倒序程序以保证 $x(n)$ 与它的变换系数 $X(m)$ 的对应关系。

六、频域图像增强技术

实验目的：1. 理想低通滤波、巴特沃斯低通滤波

2. 理想高通滤波、巴特沃斯高通滤波

3. 同态滤波

实验原理：频域滤波的一般流程

$$f(x, y) \xrightarrow{\text{变换}} F(u, v) \xrightarrow{\text{乘} H(u, v)} G(u, v) \xrightarrow{\text{反变换}} g(x, y)$$

$$G(u, v) = H(u, v)F(u, v)$$

所以，选择合适的滤波器函数 $H(u, v)$ 是关键。

(1) 理想低通滤波器

$$H(u, v) = \begin{cases} 1, D(u, v) \leq D_0 \\ 0, D(u, v) > D_0 \end{cases}$$

其中 D_0 为截止频率，是一个非负整数， $D(u, v)$ 是从点 (u, v) 到频率平面原点的距离。

理想高通滤波器

$$H(u, v) = \begin{cases} 0, D(u, v) \leq D_0 \\ 1, D(u, v) > D_0 \end{cases}$$

其中 D_0 为截止频率，是一个非负整数， $D(u, v)$ 是从点 (u, v) 到频率平面原点的距离。

(2) 巴特沃斯低通滤波、巴特沃斯高通滤波

$$H(u, v) = \frac{1}{1 + [D(u, v) / D_0]^{2n}}$$

(3) 同态滤波

同态滤波是一种在频域中同时将图象亮度范围进行压缩和将图象对比度进行增强的方法。

一个图像 $f(x, y)$ 可以根据它的照度和反射分量的乘积来表示

$$f(x, y) = i(x, y)r(x, y), \text{ 其中: } i(x, y) \text{ 为照度函数, } r(x, y) \text{ 反射分量函数}$$

算法解释：(1)理想，输入截止频率，点击确定查看结果。

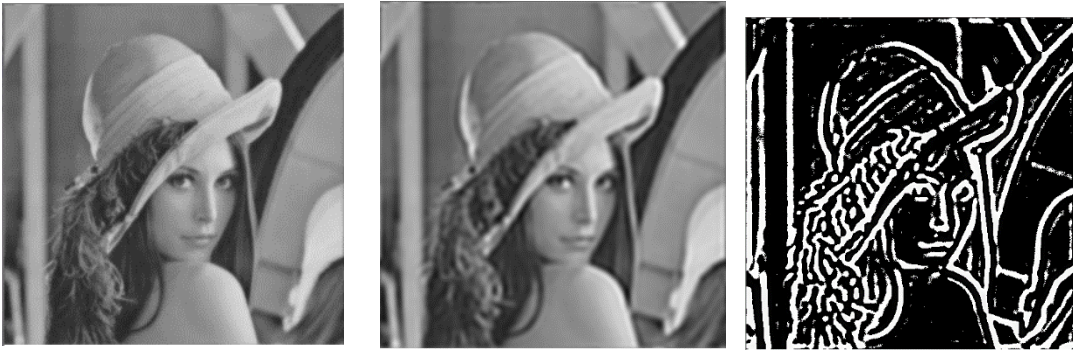
(2) 输入截止频率和阶数，点击确定返回结果。

(3) 输入 γ_H ， γ_L 以及截止频率，点击确定查看结果。

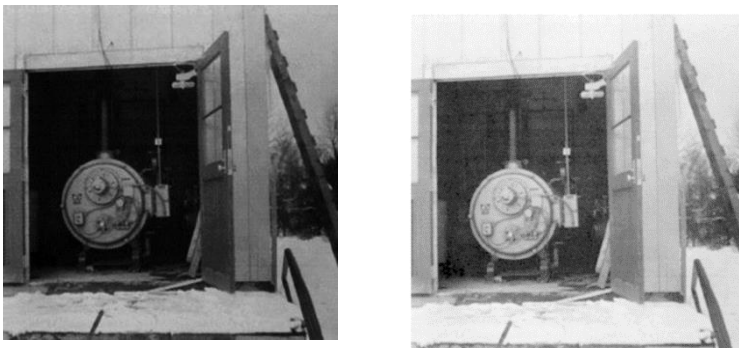
结果分析：理想低通滤波器和理想高通滤波器均有振铃性质。这与其 IFT 有 Sinc 函数形状有关。图示分别为原图，50Hz 理想低通，70Hz 理想高通。



巴特沃斯滤波器效果较理想的要好很多，在低频和高频之间没有激变，平滑很多，故没有可见的振铃现象。一阶中没有体现，但是高阶中会很明显。选择滤波器时要平衡选择。



同态滤波，降低照射分量的影响，使低灰度变得更加可见。增强高频，故边缘更加明显，改进图像。



七、数学形态学

实验目的：1. 实现基本的形态学运算：腐蚀、膨胀、开、闭，交互式输入不同类型的结构元观察效果

2. 利用数学形态学运算实现图像的骨架化、细化

实验原理：形态学图象处理以几何学为基础，着重研究图象的几何结构，这种结构可以表示分析对象的宏观性质，如在分析一个工具或印刷字符的形状时，研究的就是其宏观结构形态；也可以是微观性质，如在分析颗粒分布或由小的基元产生的纹理时，研究的便是微观结构形态。形态学研究图象几何结构的基本思想是利用一个结构元素(structure element)去探测一个图象，看是否能够将这个结构元素很好地填放在图象的内部，同时验证填放结构元素的方法是否有效。

腐蚀：腐蚀是数学形态学最基本的运算，它的实现是基于填充结构元素的概念。利用结构元素填充的过程，取决于一个基本的欧氏空间运算—平移。将一个集合 A 平移距离 x 可以表示为 $A+x$ ，其定义为： $A+x=\{a+x:a\in A\}$

从几何上看 $A+x$ 表示 A 沿矢量 x 平移了一段距离，探测的目的就是标记出图象内部那些可以将结构元素填入的(平移)位置。

集合 A 被集合 B 腐蚀，定义为 $A\ominus B=\{x:B+x\subset A\}$

腐蚀由将 B 平移 x 但仍包含在 A 内的所有点 x 组成，如果将 B 看做为模板，那么腐蚀则由在平移模板的过程中，所有可以填入 A 内部的模板的原点组成。

如果原点在结构元素的内部，那么腐蚀具有收缩输入图象的作用，腐蚀后的图象为输入图象的一个子集，如果原点在结构元素的外部，那么，腐蚀后的图象则可能不在输入图象的内部。

腐蚀除了可以用填充形式表示外，还有一个更重要的表达形式 $A\ominus B=\{A-b:b\in B\}$

这里腐蚀可以通过将输入图象平移 $-b$ (b 属于结构元素)，并计算所有平移的交集而得到。

从图象处理的观点看，腐蚀的填充定义具有非常重要的含意，而上式则无论对计算还是理论分析都十分重要。

膨胀：二值数学形态学的第二个基本运算是膨胀。膨胀是腐蚀运算的对偶运算(逆运算)，可以通过对补集的腐蚀来定义，其定义为： $A\oplus B=[A^c\ominus(-B)]^c$ 其中， A^c 表示 A 的补集。为了利用 B 膨胀 A ，可将 B 相对原点旋转 180。得到 $-B$ ，再利用 $-B$ 对 A^c 进行腐蚀。腐蚀结果的补集，便是所求的结果。因为膨胀是利用结构元素对图象补集进行填充，因而它表示对图象外部作滤波处理。而腐蚀则表示对图象内部作滤波处理。在书写时，按照图象处理的习惯，总是先写出输入图象，然后写出结构元素。如果结构元素为一个圆盘，那么膨胀可填充图象中的小孔(相对于结构元素而言比较小的孔洞)，及在图象边缘处的小凹陷部分。而腐蚀可以消除图象中小的成分，并将图象缩小，从而使其补集扩大。

开运算：先腐蚀后膨胀的过程称为开运算，用来消除小物体、在纤细点处分离物体、平滑较大物体的边界的同时并不明显改变其面积 $A\circ B=(A\ominus B)\oplus B$

等价于 $A \circ B = \cup \{B+x: B+x \subset A\}$ 开运算可以通过计算所有可以填入图象内部的结构元素平移的并求得，即对每一个可填入位置作标记，计算结构元素平移到每一个标记位置时的并，便可得到开运算结果，这正是先作腐蚀，然后作膨胀运算的结果。

闭运算：先膨胀后腐蚀的过程称为闭运算。用来填充物体内部细小空洞、连接邻近物体、平滑其边界的同时并不明显改变其面积。 $A \bullet B = [A \oplus (-B)] \ominus (-B)$ 通常，由于噪声的影响，图象在阈值化后所得到边界往往是很不平滑的，物体区域具有一些噪声孔，背景区域上散布着一些小的噪声物体。连续的开和闭运算可以有效地改善这种情况。有时需要经过多次腐蚀之后再加上相同次数的膨胀，才可以产生比较好的效果。

骨架化：骨架化可以通过腐蚀和开操作来表达，即骨架可以表示为 $S(A) = \bigcup_{i=1}^n S_i(A)$

其中， $S(A) = (A \ominus k B) - (A \ominus k B) \circ B$ ，式中，B 是一个结构元，而 $A \ominus k B$ 表示对 A 的连续 k 次腐蚀。

细化：许多数学形态学算法都依赖于击中击不中变换，其中图象细化便是一种最常见的使用击中击不中变换的形态学算法。对于结构对 $B = (E, F)$ ，利用 B 细化 S 定义为：

$S \otimes B = S - S * B$ ，即 B 细化 S 为 $S * B$ 与 S 的差集。更一般地，利用结构对序列 B^1, B^2, \dots, B^k ，迭代地产生输出序列：

$S^1 = S \otimes B^1, \dots, S^k = S^{k-1} \otimes B^k$

或 $\{S^i\} = S \otimes \{B^i\} = \left(\dots \left((S \otimes B^1) \otimes B^2 \right) \dots \otimes B^k \right)$ 随着迭代的进行，得到的集合也不断细化。假设输入集合是有限的，最终将得到一个细化的图象。结构对的选择仅受结构元素不相交的限制。事实上，每一个 B^i 都可以是相同的结构对，即在不断重复的迭代细化过程使用同一个结构对。在实际应用中，通常选择一组结构元素对，迭代过程不断在这些结构对中循环，当一个完整的循环结束时，如果所得结果不再变化，则终止迭代过程。

算法解释：在下拉菜单中选择要进行的操作，然后点击选择结构元，（预览没有效果。。）点击 OK，对话框不会消失，会产生效果。

结果分析：左图为经过多次膨胀后的图像，右图为原图，可见，膨胀可以使图像之间的缝隙减小，横正向的结构元可以使横向变化更快，纵正向的结构元可以使纵向变化快。



以下为 lena 图像的膨胀和腐蚀对比图，腐蚀会丢失一些细线类的细节，膨胀会丢失细的缝隙。



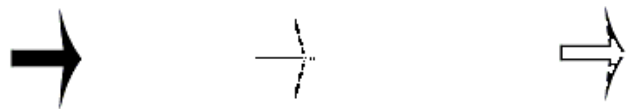
以下是对 lena 图像进行开运算和闭运算，可见，[开运算](#)可以消除细小的物体、在纤细点处分离物体、平滑较大物体的边界时不明显的改变其面积，[闭运算](#)具有填充物内细小的空洞、连接邻近物体、在不明显改变物体面积的情况下平滑其边界的作用。



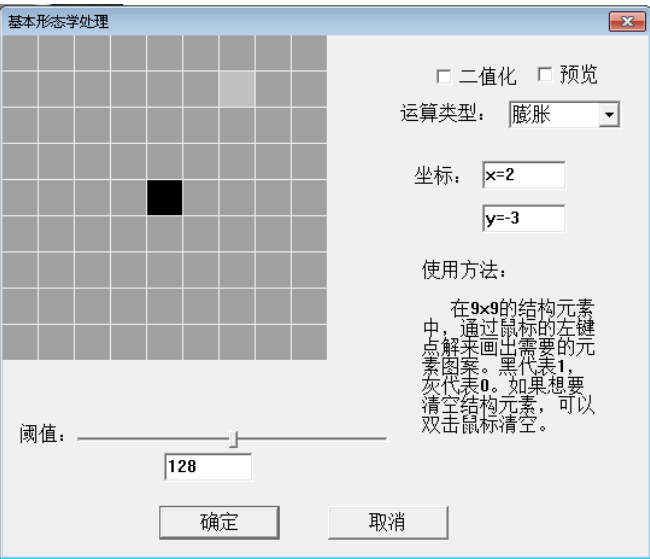
下图是操作对话框界面，可以交互研究结构元对图像的影响。

下图为箭头的原图，细化图和骨架化图，均为全方形结构元结果。由于计算方法的问题，图片太大程序很容易崩溃。

可见，点、线的骨架就是自身，细化过程中去掉很多信息，但仍然要保证形状不变。



八、图像恢复



实验目的：用逆滤波方法和维纳滤波方法恢复运动模糊图像（加噪声和不加噪声）

实验原理：图象在形成、传输和记录过程中，由于成像系统、传输介质和记录设备的不完善，都会使图象的质量退化。图象恢复是要将图象退化的过程加以估计,并补偿退化过程造成的失真,以便获得未经干扰退化的原始图象或原始图象的最优估值的一种图象质量改善方法。图像退化可以模型化为原图像先乘以一个退化函数再加上一个噪声函数（空域和频域都是这样只不过一个是卷积一个是乘积）

$$g(x,y) = h(x,y)*f(x,y) + n(x,y)$$
$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$

所以，图像退化有两种因素：图像噪声 $n(x,y)$ 和退化系统 $h(x,y)$ 。

噪声来源于图像获取过程和传输信道。包括常见高斯、瑞利、伽马、指数分布、均匀分布、椒盐噪声。基于此，有空间滤波和频域滤波两种方式来消除噪声。若 $n=0$ ，那么就是

$H(u, v)$ 使其退化。估计 $H(u, v)$ 有观察法、实验法、数学建模法三种。逆滤波有两种情况，已知无噪声，且退化函数已知，那很容易得到

$$\hat{F}(u, v) = \frac{F(u, v)}{H(u, v)} = \frac{G(u, v)}{H(u, v)}$$

若有噪声，那么噪声将决定图像复原的好坏。

维纳滤波，即最小均方误差滤波。误差度量 $e^2 = E\{(f - \hat{f})^2\}$ ，维纳滤波器的表达式近似为：

算法解释：点击实现。

$$\begin{aligned}\hat{F}(u, v) &= \left[\frac{H^*(u, v) S_f(u, v)}{S_f(u, v) |H(u, v)|^2 + S_\eta(u, v)} \right] G(u, v) = \left[\frac{H^*(u, v)}{|H(u, v)|^2 + \frac{S_\eta(u, v)}{S_f(u, v)}} \right] G(u, v) \\ &= \left[\frac{1}{H(u, v)} \times \frac{|H(u, v)|^2}{|H(u, v)|^2 + \frac{S_\eta(u, v)}{S_f(u, v)}} \right] G(u, v) \\ \hat{F}(u, v) &= \left[\frac{1}{H(u, v)} \times \frac{|H(u, v)|^2}{|H(u, v)|^2 + \frac{S_\eta(u, v)}{S_f(u, v)}} \right] G(u, v) \\ \hat{F}(u, v) &= \left[\frac{1}{H(u, v)} \times \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} \right] G(u, v)\end{aligned}$$

结果分析：以下为用逆滤波 (20Hz, T=70) 和维纳滤波 (20Hz, T=90) 结果对比。有噪声的情况下，维纳滤波（第三幅图）效果更好些。



九、灰度图像分割

实验目的：1. 阈值化算法：手动阈值、自动阈值（大津阈值）

2. 区域生长

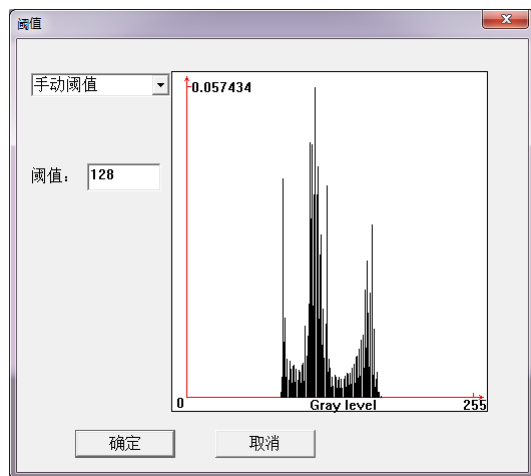
3. 区域分裂合并

实验原理：图像分割是为了将图像中的目标分为各个感兴趣的区域，与图像中的各种物体目标相对应。通过对分割结果的描述，可以理解图像中包含的信息。分割依据可以使灰度值，颜色特征或者图像的空间模式特性。分割要求区域内离散度最小，区域间离散度最大。灰度阈值分割法是最古老的分割技术，通过对灰度取阈值后得到的图像。只能应用于图像中组成感兴趣对象的灰度值是均匀的，并且和背景的灰度值不一样关键是怎样选择阈值，简便的方法是检查图像的直方图。单一阈值方法不适合于由许多不同纹理组成一块块区域的图像。双峰模式比较适合，取峰间谷值为阈值。根据类间距离最大准则，把直方图在某一阈值处分割成两组，当被分成的两组间方差为最大时，确定阈值。

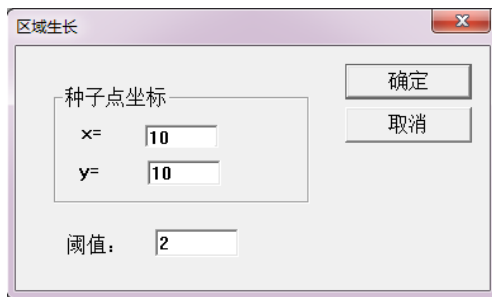
2. 区域生长算法：根据图像的不同应用选择一个或一组种子，它或者是最亮或最暗的点，或者是位于点簇中心的点；选择一个描述符（条件）；从该种子开始向外扩张，首先把种子像素加入集合，然后不断将与集合中各个像素连通、且满足描述符的像素加入集合；上一过程进行到不再有满足条件的新结点加入集合为止。

3. 分裂合并：对于图像中灰度级不同的区域，均分为四个子区域；如果相邻的子区域所有像素的灰度级相同，则将其合并；反复进行上两步操作，直至不再有新的分裂与合并为止。

算法解释：阈值分割：在对话框中选择算法种类，手动阈值还要输入阈值，右侧是直方图参考，大津阈值，选择点击 OK 后会更新阈值数据，看到结果。



区域生长：输入种子坐标及阈值，确定即可出结果。



结果分析：阈值分割：左为大津阈值，右为 50 手动阈值。



区域生长：没调出来

区域分裂合并：用四叉树分裂，选择 70%满足在灰度差内就将其合并。Lena 图。

