

浙江大学实验报告

姓名： 林炬乙 学号： 3180103721

课程名称： 数字图像处理 任课老师： 项志宇

实验名称： 恢复运动模糊图像 实验日期： 2021/4/27

1 实验目的和要求

（分点简要说明本次实验需要进行的工作和最终的目的）

试生成一幅带噪声的运动模糊图像，并采用逆滤波或者维纳滤波进行恢复。

2 实验原理

运动模糊

运动模糊，是在拍摄设备快门打开的时间内，物体在成像平面上的投影发生平移或旋转，使接受的影像彼此发生重叠。

对于真实图像，模糊图像自然是原始图像的叠加，但对于数字图像，由于像素信息由数值表示，不能简单地将相应像素值相加，而是将像素信息缩小后相加，否则会使亮度成倍增加，使图像严重失真。

例：匀速直线运动的模糊恢复

- 若景物与照相机作相对运动，快门时间为 T ，则成像是 T 时间内曝光累积的结果：

逆滤波

解决病态问题的一种方法是将限制滤波的频率，将频率限制在原点附近，降低遇到零值的概率。

- 直接逆滤波和在原点附近滤波的效果比较（巴特沃斯低通滤波半径分别为 40,70,85）
- 半径越大越接近于直接逆滤波

当退化函数 $H(u,v)$ 是零或者非常小的值，则 $N(u,v)/H(u,v)$ 很容易支配整个式子。

维纳滤波

在数学应用上，对于运动引起的图像模糊，最简单的方法是直接做逆滤波，但是逆滤波对加性噪声特别敏感，使得恢复的图像几乎不可用。最小均方差（维纳）滤波用来去除含有噪声的模糊图像，其目标是找到未污染图像的一个估计，使它们之间的均方差最小，可以去除噪声，同时清晰化模糊图像。

最小均方误差（维纳，Wiener）滤波

- 是一种频域方法，考虑了既有退化又有噪声的情况
- 可用原图 $f(x,y)$ 与恢复图 间的均方误差 e^2 最小为准则来恢复图像
- 找一点扩展函数，它与图像卷积的结果为恢复图像

——Wiener 滤波，也叫最小均方误差滤波

导出满足要求的转移函数，它对噪声放大自动有抑制作用

维纳滤波要求未退化图像和噪声的功率谱都是已知的；未知时用功率谱比的常数估计，未必总能得到合适解；

3 实验内容

（分点阐述实验步骤）

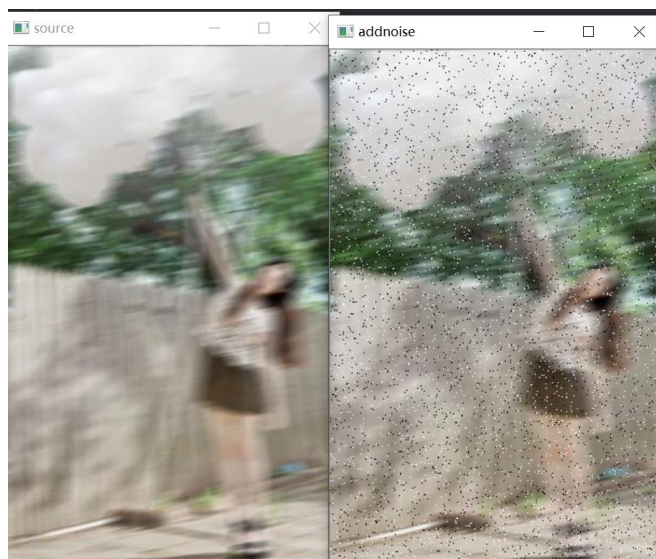
1. 生成运动模糊

```
blurred = im2double(imread('ShaonvBlur.jpg'));
```

```
% [hei,wid,~] = size(blurred);
```

% 以上是读入自己的模糊函数，好像行不通，他这个好像要知道模糊函数然后求解

生成的模糊图像如图，我把它保存为文件，然后加噪声。一开始加 3000 个椒 3000 个盐噪声点



2. 维纳滤波恢复

我把两个三维相减，计算噪声的方差。

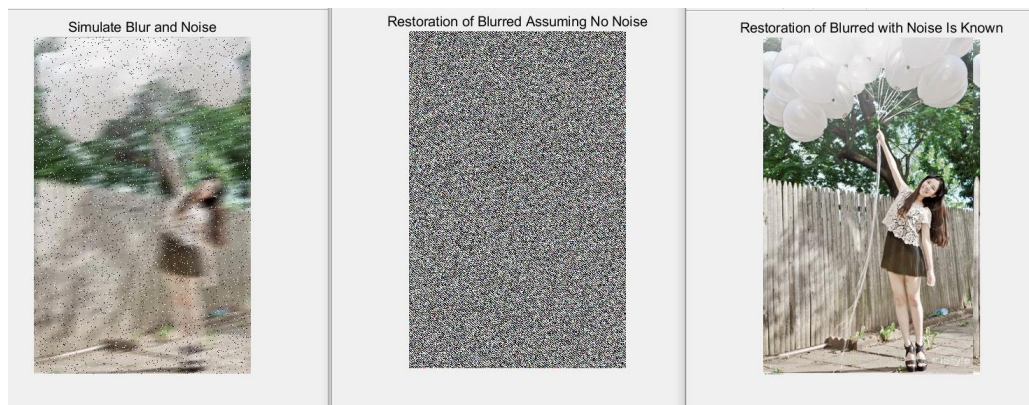


这样似乎效果一般，图像还是很模糊，因为两个相减变暗了很多。

为什么会分析不出来噪声呢？

我们需要噪声信号比

频率域的一次尝试：如果知道噪声，那么试试更大的噪声也可以。频率域滤波效果非常好。



遇到的错误:

问题 0.

OpenCV(3.4.1) Error: Assertion failed ($(_src.type() == (((0) \& ((1 < 3) - 1)) + (((1)-1) < 3)))$) in `cv::equalizeHist`, file C:\build\master_winpack-build-win64-vc14\opencv\modules\imgproc\src\histogram.cpp, line 3914

原因：这是因为做直方图均衡需要你的灰度值是整数，将输入变量加上 `np.uint8()` 转换为整数就可以了；

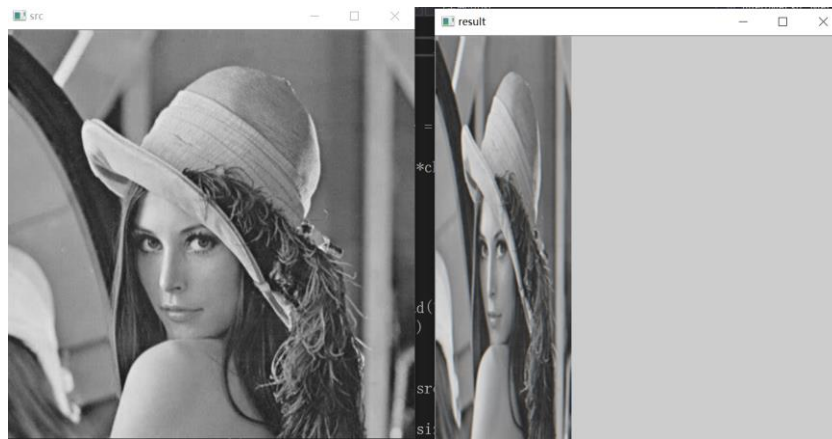
也有可能是输入的图是三通道的，要注意做直方图均衡需要输入单通道的图；如果想做 RGB 或者其他三通道的图，可以转为灰度图，或者取各个通道 `im[:, :, 0]` 依次做，再 `merge` 到一起

一种可能是数组越界，`i` 和 `j` 不对，`i` 是横坐标，是 `<col`，`j` 是纵坐标是 `<row`。

你可以加参数，`imread("Lina.jpg", 参数)`，直接读入灰度图处理比较方便，不要用默认的 `rgb` 三个通道。
`image = cv2.imread('images/1.jpg', cv2.IMREAD_COLOR)` 每个点的值是 `vec3b` 类型
`image = cv2.imread('images/1.jpg', cv2.IMREAD_GRAYSCALE)` 每个点的值是 `uchar` 类型

正确解决：因为我没有初始化 `dst`，就直接给它赋值了。

问题 1：不管怎么修改运动模糊代码，运动模糊后图像变小了。



原因:

Mat srcImg = imread("Lina.jpg"); 之前是读入灰度图，就会出问题，把 IMREAD_GRAYSCALE 去掉就不会变小了。

问题 2: 没有多个窗口, openCV 同一个函数多次调用 imshow 怎么不覆盖

原因: 窗口名都是同一个就会覆盖，应该把窗口名作为变量，我们把窗口名传入进去。

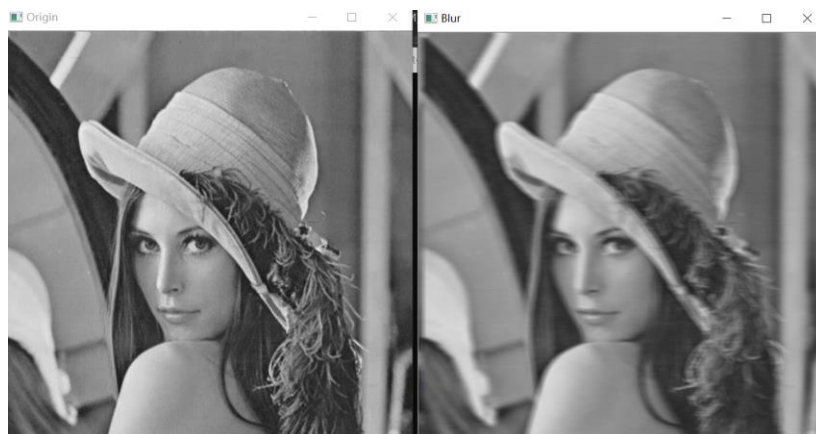
问题3: 怎么把行和列变成偶数

解决方法:

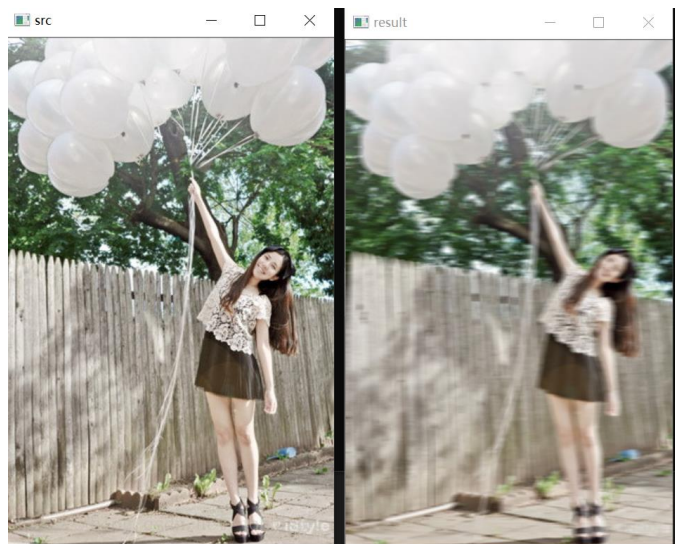
&上-2，其实是为了把行和列变成偶数 -2的二进制是11111111.....10 最后一位是0，就是把最后一位变成0。

4 实验结果和分析

（使用图片和文字叙述实验结果，并对这些结果进行适当分析）

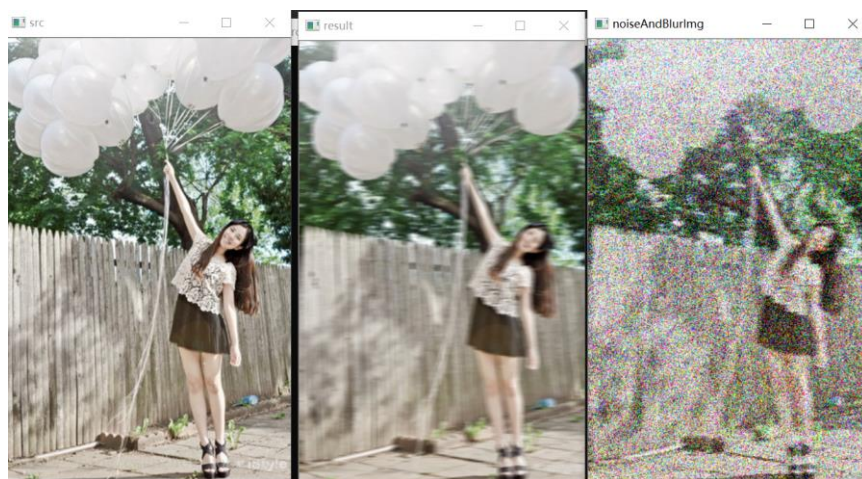


如图，我们可以实现 运动模糊.不过总觉得和别的模糊看起来不是很明显.所以我还试了另一个图



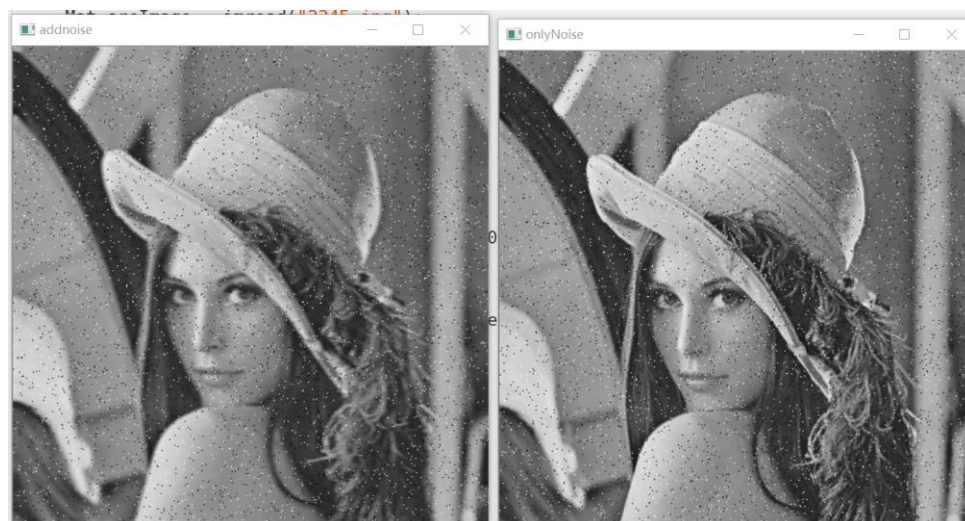
更有运动模糊的感觉.

加高斯白噪声

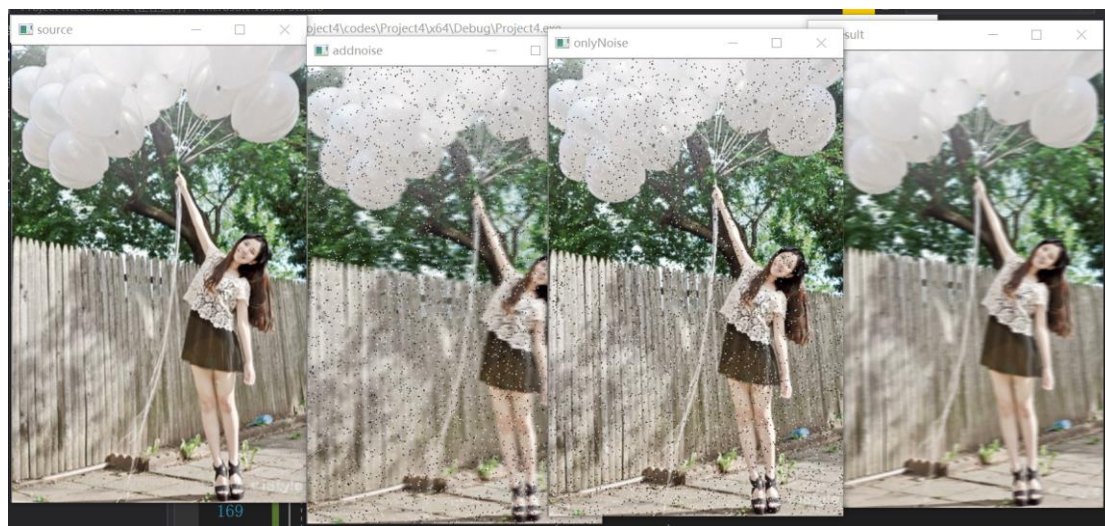


这个好像过密了, 恢

复效果很差.



左图模糊加噪声, 右图是原图加噪声.



噪声稍微少一些.

不过好像还是太大了, 我看要

```
% Simulate additive noise.
noise_mean = 0;
noise_var = 0.0001;
blurred_noisy = imnoise(blurred, 'gaussian', ...
                        noise_mean, noise_var);

subplot(2 3 4)
```

方差为 0.0001 时

才能成功滤波.而且需要知道模糊函数



可以看到给加 0.0001 方差的噪声,不考虑噪声时无法恢复都是乱点, 考虑噪声后成功恢复.

代码

```
Cpp 运动模糊代码void filter(Mat src, Mat dst, const float core[], int cx, int cy);
Cpp加入噪声代码Mat addSaltNoise(const Mat srcImage, int n)
cpp运动模糊代码,生成模板
int motionblurcore(float* core, int width, int height, int dir)
{
```

```
if (height != 1 || width % 2 != 1)
    return 0;

int count = width / 2 + 1;
if (dir == 0)
{
    for (int i = 0; i < count; i++)
        core[i] = 1.0 / count;
    for (int i = count; i < width; i++)
        core[i] = 0.0;
}
else {
    for (int i = 0; i < count - 1; i++)
        core[i] = 0.0;
    for (int i = count - 1; i < width; i++)
        core[i] = 1.0 / count;
}
return 1;
}
```