

数字图像处理实验二实验报告

白晋斌
171860607
810594956@qq.com

Table of Contents

<i>assignment1</i>	2
<i>assignment2</i>	3
<i>assignment3</i>	9
<i>assignment4</i>	10
主界面:.....	10
主要功能包括:.....	11
读取图片:.....	11
保存图片:.....	11
复古滤镜:.....	12
磨皮去皱:.....	13
肤色调节:.....	14
美白补光:.....	15
肤色匹配:.....	16

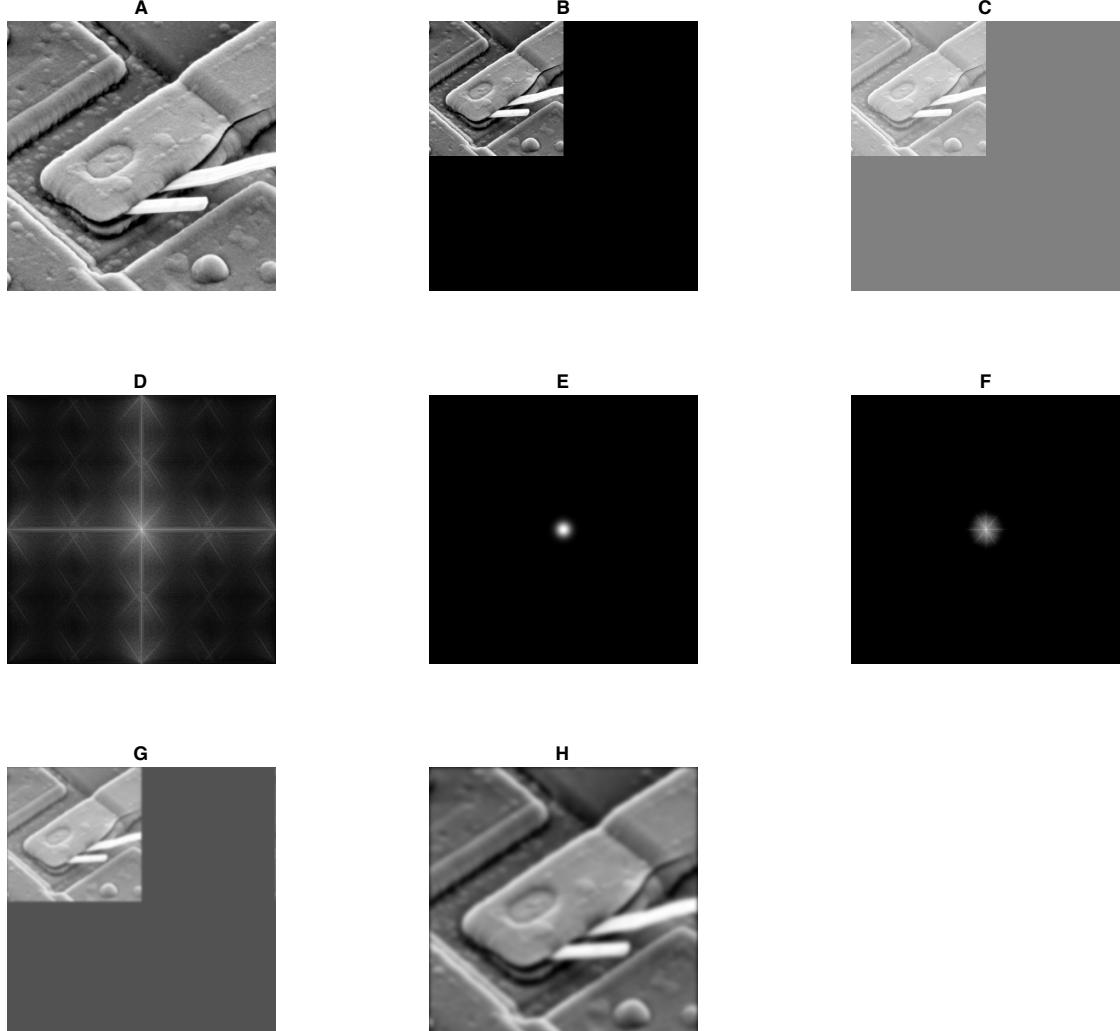
assignment1

根据教材 166 页的相关内容,我们可以得到代码编写的思路:

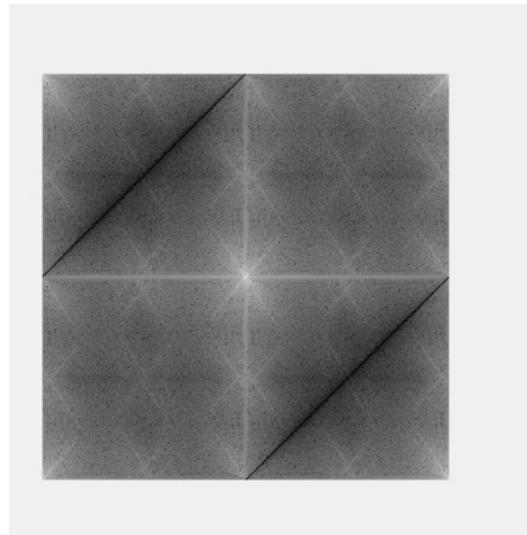
- a. 给定一幅大小为 $M \times N$ 的输入图像 $f(x,y)$, 得到填充参数 $P = 2M, Q = 2N$
- b. 对 $f(x,y)$ 添加必要数量的 0, 形成大小为 $P \times Q$ 的填充后的图像 $fp(x,y)$
- c. 用 $(-1)^{x+y}$ 乘 $fp(x,y)$ 移到其交换的中心
- d. 计算来自步骤 3 的图像的 DFT, 得到 $F(u,v)$
- e. 生成一个实对称的滤波函数 $H(u,v)$, 其大小为 $P \times Q$, 中心在 $(P/2, Q/2)$ 处
- f. 用阵列相乘形成乘积 $G(u,v) = H(u,v)F(u,v)$
- g. 得到处理后的图像
- h. 通过从 $gp(x,y)$ 的左上象限提取 $M \times N$ 区域, 得到最终处理结果 $g(x,y)$

其中, 第 e 步的滤波函数, 我们按照课本所述选择高斯低通滤波函数($D_0=40$).

选用教材图片, 进行滤波, 得:



实验过程中可能会生成如图所示的图 D



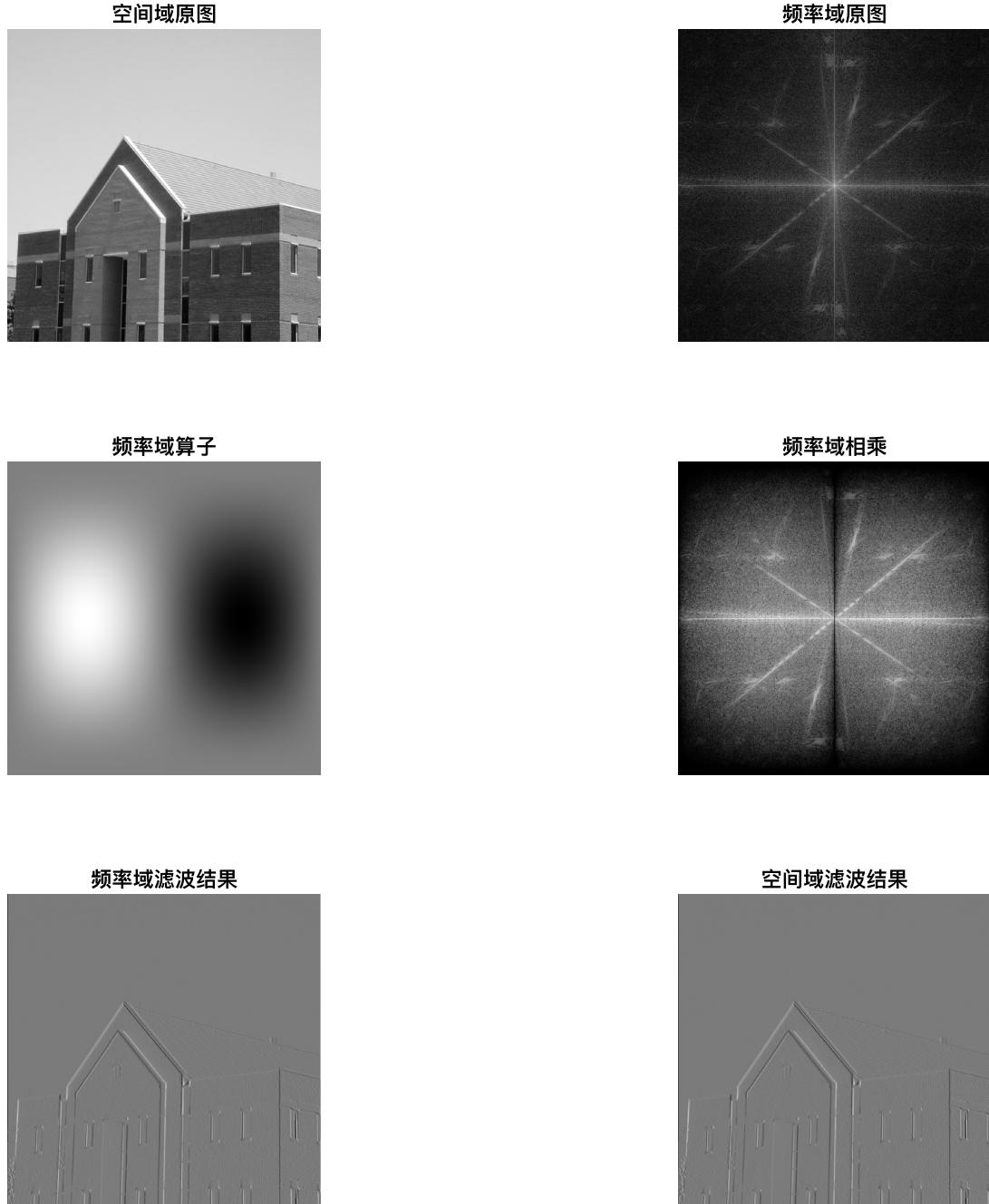
主要原因在于 c 步,如果数据是整数,乘-1 可能为 0 而不是负数,在读取文件时添加
`f=mat2gray(f);`即可解决问题.
所以函数代码均为自行实现.

assignment2

根据教材第 168 页的相关内容,我们可以得到代码编写的思路:

1. 空间域滤波
 - a. 读取原图
 - b. 扩充原图(分为零填充、padding 等方式,我们均进行了实现)
 - c. 将算子翻转 180 度
 - d. 利用算子对扩充后的原图进行卷积(分为 same、full、valid,我们以 same 为例,其余方式修改循环变量即可)
 - e. 作图
2. 频率域滤波
 - a. 读取原图(X,Y)
 - b. 读取算子(A,B),扩充原图与算子至(M,N),其中 $M \geq X+A-1, N \geq Y+B-1$, 其中 M 与 N 均为偶数(值得注意的是,Sobel 算子为奇对称,需要放置在扩充后的中央使得填充后依旧为奇对称)
 - c. 生成算子填充图的频率域 H(具体包括乘 $(-1)^{x+y}$,计算 DFT,实部置 0,乘 $(-1)^{x+y}$)
 - d. 生成原图扩充图的频率域 F(具体包括乘 $(-1)^{x+y}$,计算 DFT)
 - e. 用阵列相乘形成乘积 $G(u,v) = H(u,v)F(u,v)$, 乘 $(-1)^{x+y}$
 - f. 作图

选用教材图片,进行滤波,得:



此外,为验证验证空间域滤波和频率域滤波的等价性,我们添加如下代码:

```
d=abs(mapminmax(finalsf)-mapminmax(finalff));
max(d(:))
min(d(:))
```

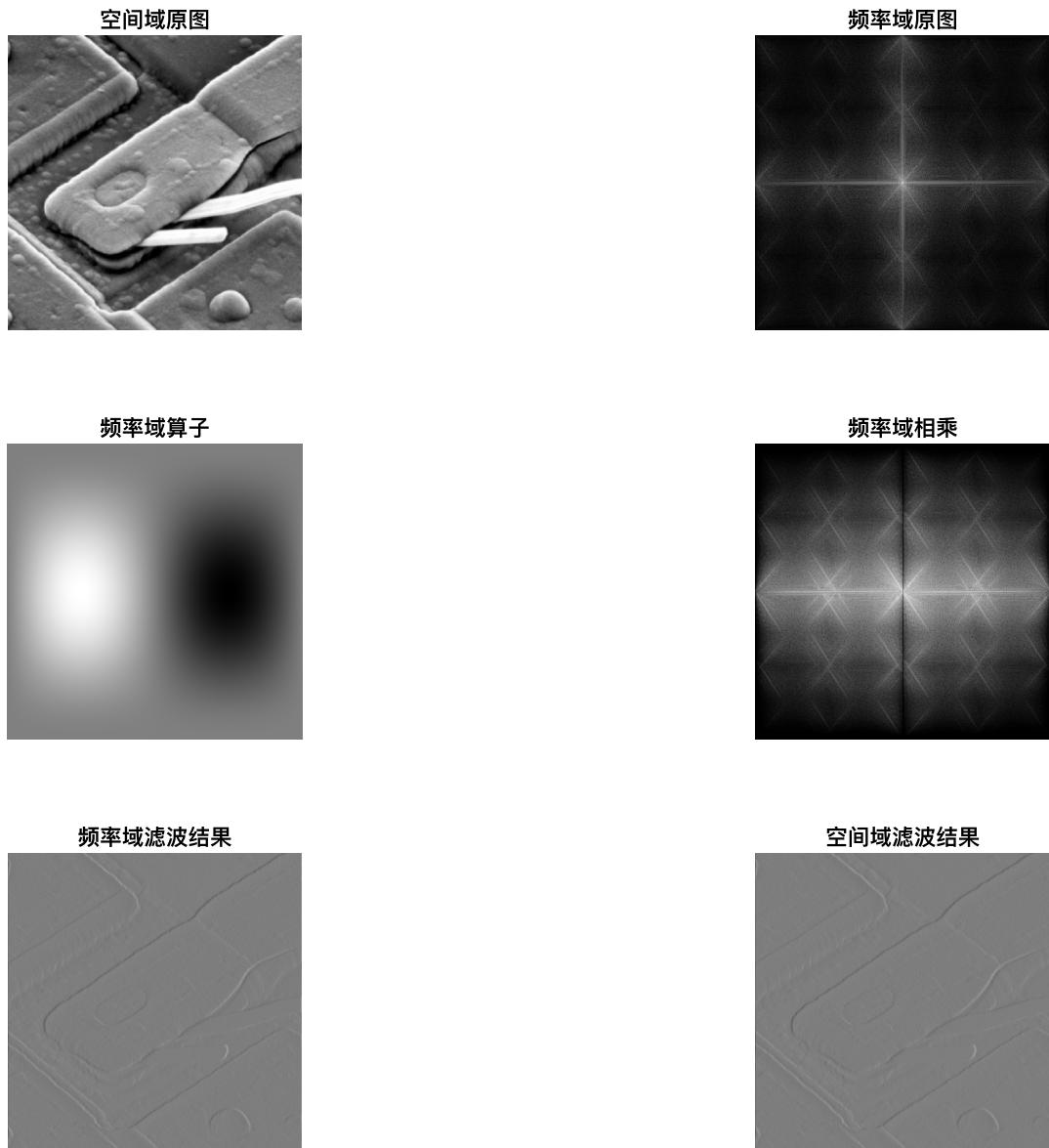
其中 `finalsf` 和 `finalff` 分别是频率域滤波结果和空间域滤波结果,实验输出为
`Ans= 2.664535259100376e-15`

`Ans= 0`

因为做了归一化处理,所以实际上最大差的 `2e-15` 的误差纯属计算误差,因此可以证明空间域滤波和频率域滤波是等价的.

实验过程中某些图直接进行 `imshow` 效果不佳,这时候需要进行 `log(abs+1)` 处理,此外,对于复数,依旧可以直接用 `abs` 进行取模.

原图和算子在频域中的图片已放入 asset/image 文件夹.
所有函数代码均为自行实现.
此外,分别应用 436.tif 和 woman.png 进行测试,结果如下:



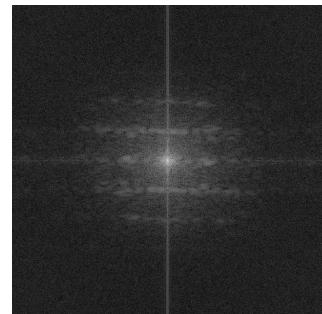
Ans= 2.664535259100376e-15

Ans= 0

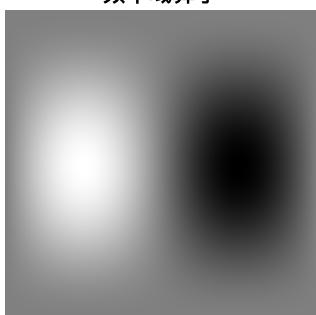
空间域原图



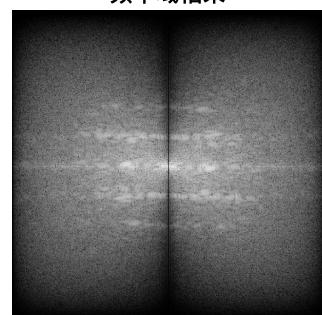
频率域原图



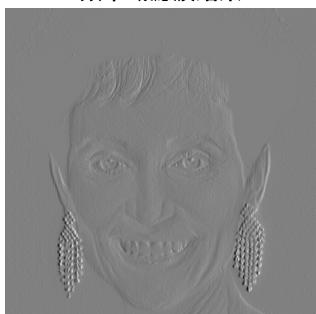
频率域算子



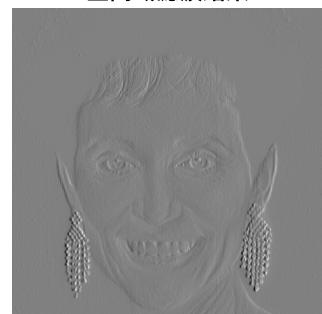
频率域相乘



频率域滤波结果



空间域滤波结果



Ans= 1.332267629550188e-14

Ans= 0

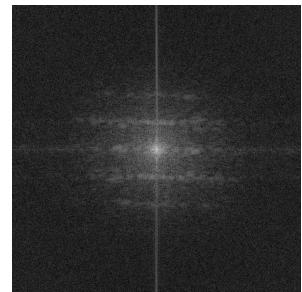
对归一化的结果做差得到的答案也证明了频率域滤波结果和空间域滤波结果等价对于不同图片均成立。

此外,我们尝试对角线形式的 Laplacian 算子([1,1,1;1,-8,1;1,1,1])和 Prewitt 算子,实验结果如下:

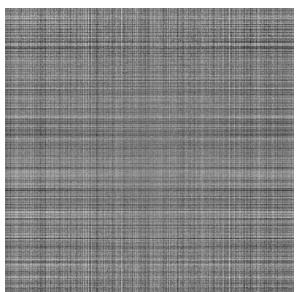
空间域原图



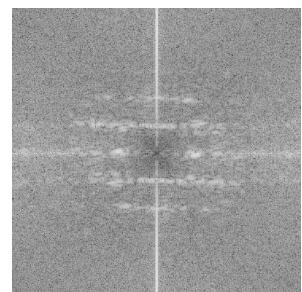
频率域原图



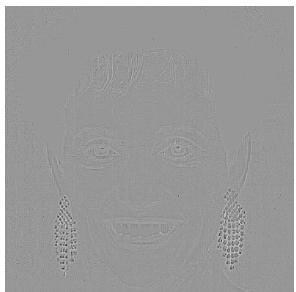
频率域算子



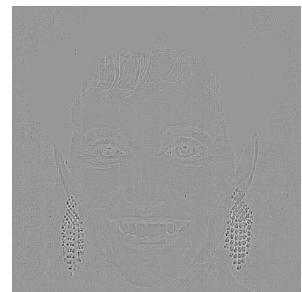
频率域相乘



频率域滤波结果



空间域滤波结果



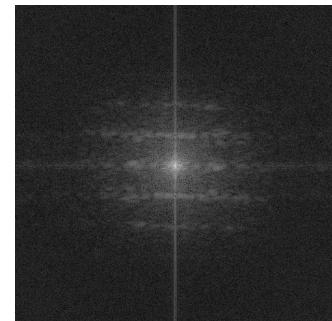
Ans= 1.443289932012704e-14

Ans= 0

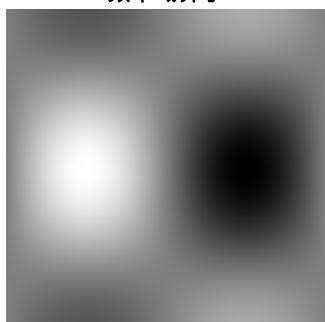
空间域原图



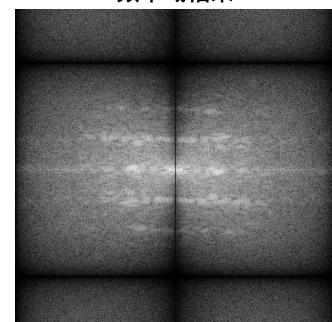
频率域原图



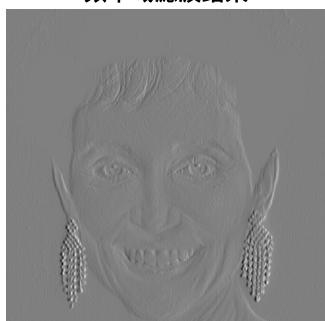
频率域算子



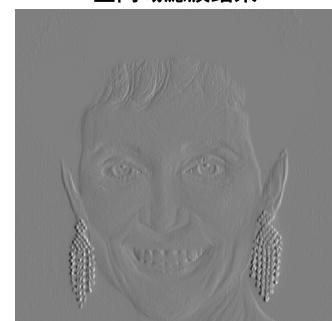
频率域相乘



频率域滤波结果



空间域滤波结果



Ans= 1.443289932012704e-14

Ans= 0

对归一化的结果做差得到的答案也证明了频率域滤波结果和空间域滤波结果等价对于不同算子均成立。

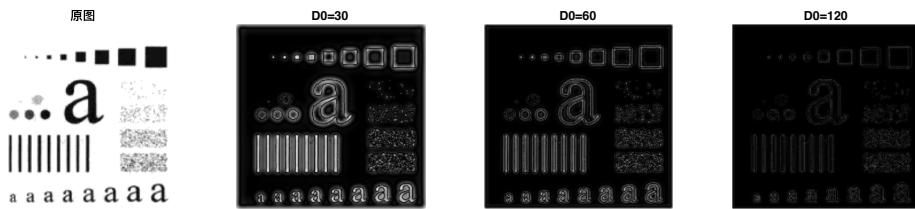
(此外我们还发现, Laplace 算子对边缘检测的效果并不好, 这是由于 Laplace 算子对孤立象素的响应要比对边缘或线的响应要更强烈, 因此只适用于无噪声图象. 存在噪声情况下, 使用 Laplacian 算子检测边缘之前需要先进行低通滤波. 所以, 通常的分割算法都是把 Laplacian 算子和平滑算子结合起来生成一个新的模板.)

assignment3

根据教材 176-178 页的相关内容,我们可以轻松修改 assignment1 的代码并进行封装,完成实验.

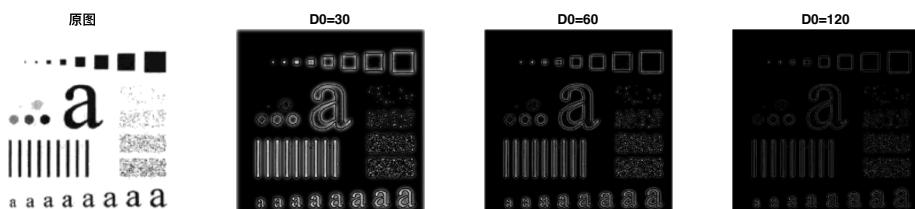
所有函数代码均为自行实现.

输出结果(默认巴特沃斯高通滤波器 $n=2$)如下:



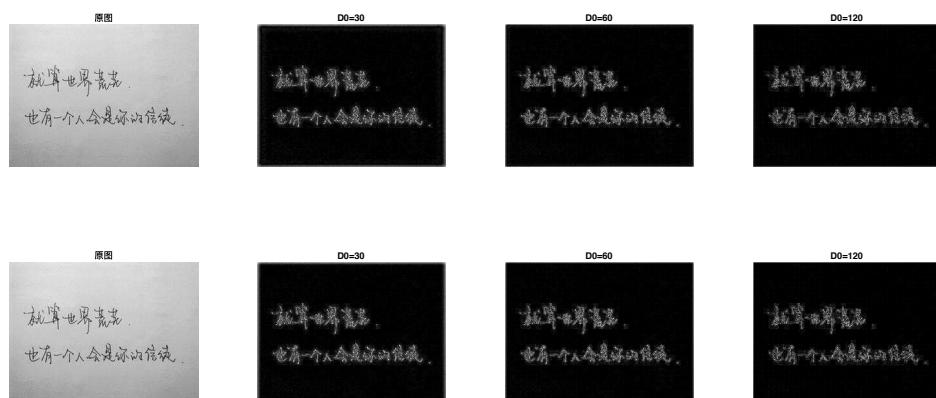
从上图观察可以得到,当 D_0 较小时,振铃现象相当严重,以至于产生了失真,物体的边界被加粗.而随着 D_0 的增大,振铃现象得到缓解,边缘失真也得到改善,部分较小物体开始出现.当 D_0 足够大时,边缘更清晰,失真更小,部分较小的物体已十分清晰.

此外,我们的代码中还实现了巴特沃斯高通滤波器和高斯高通滤波器的对比(巴特沃斯高通滤波器为上图,高斯高通滤波器为下图)



对比可以发现,高斯高通滤波器的结果更清晰一些,曲线更平滑,白色边缘宽度稍微窄一点.而二者对于小物体的识别能力差不多.

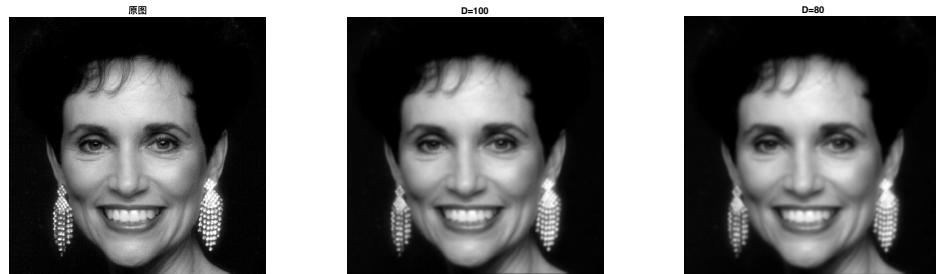
此外,选用我们自己的图片,分别使用巴特沃斯高通滤波器(上)和高斯高通滤波器(下).



经观察,可以得到类似结论.

assignment4

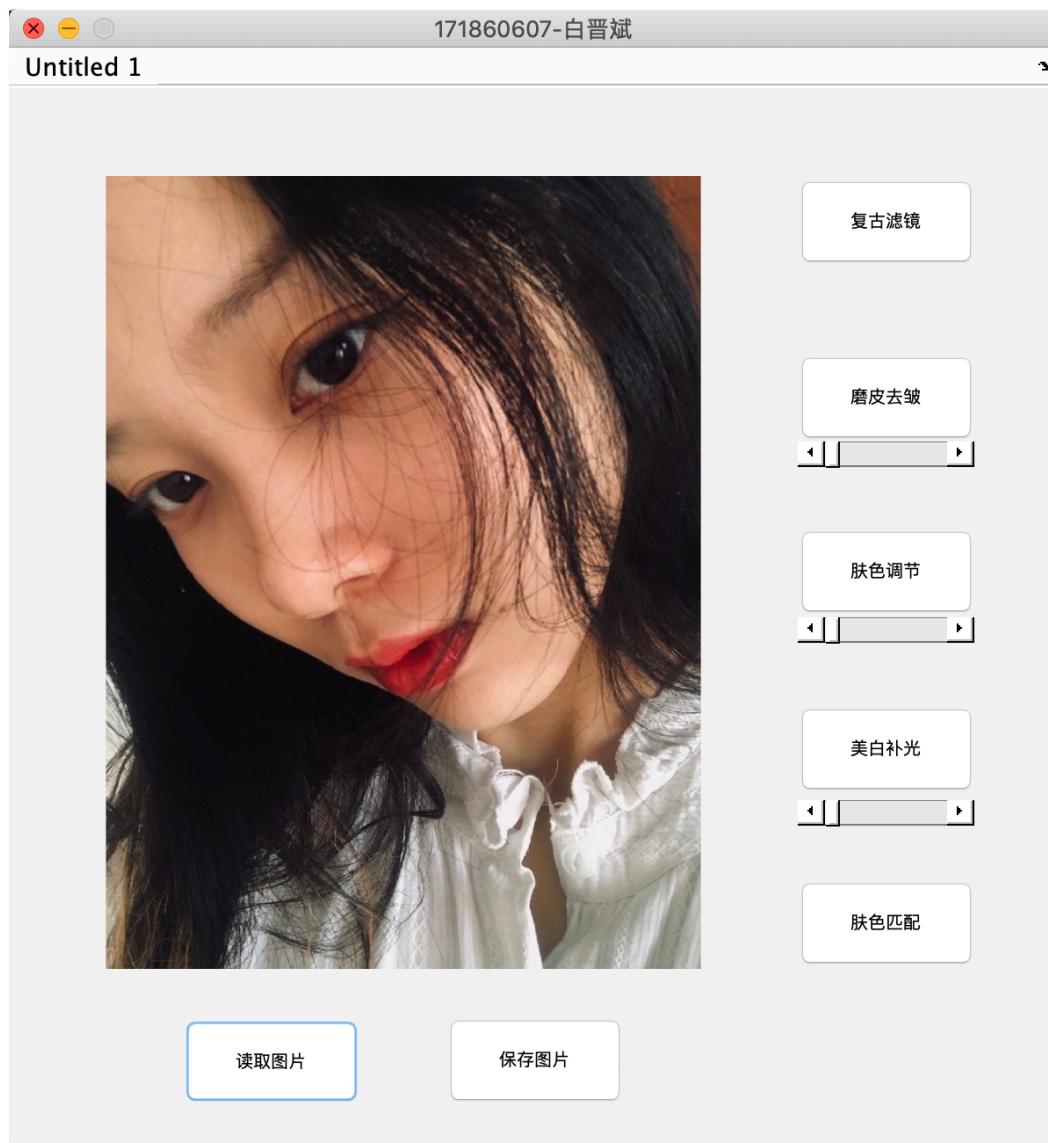
首先展示对课本图片进行取出皱纹的实验结果.这里可以发现 D=100 皱纹不太明显,D=80 皱纹完全不见.



接下来展示我们的美颜软件.

本软件未使用高级库函数, 所以函数代码均为自行实现.

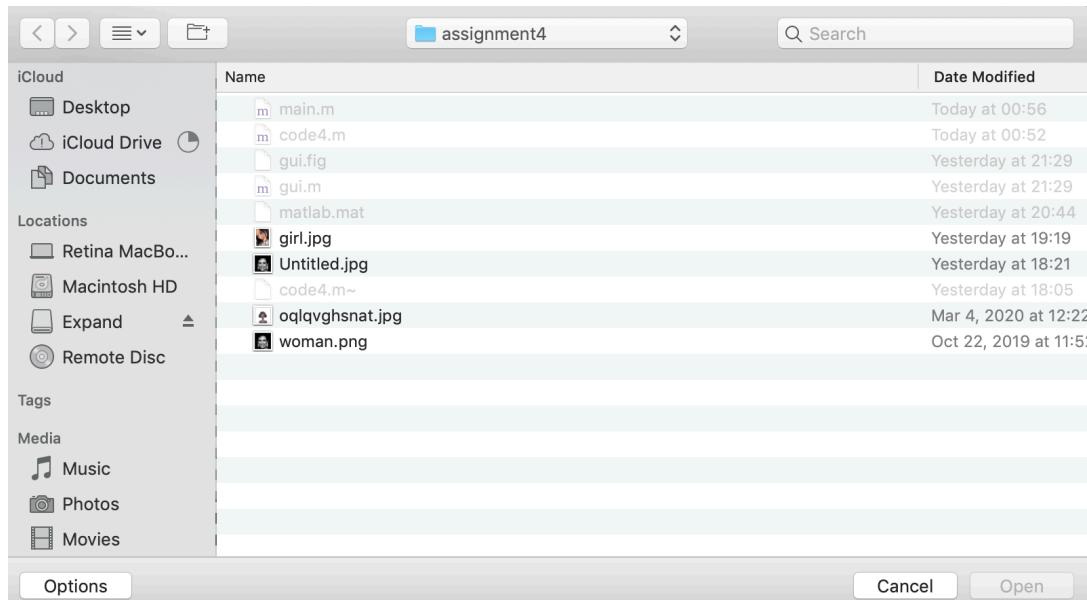
主界面:



主要功能包括:

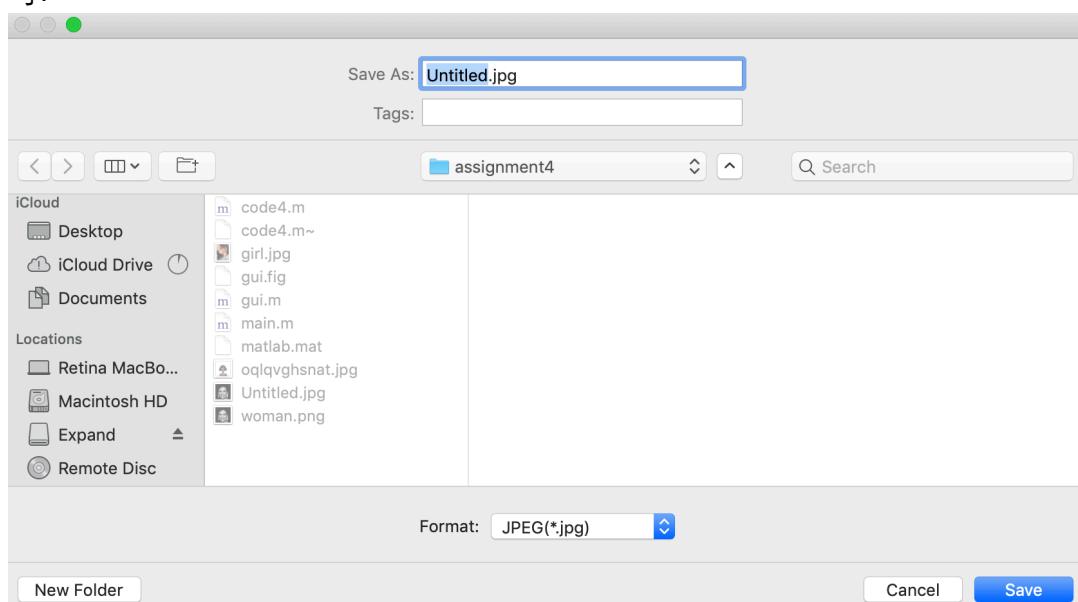
读取图片:

点击后调用系统文件夹,直接选择对应文件即可.选择后图片将展示在主界面.



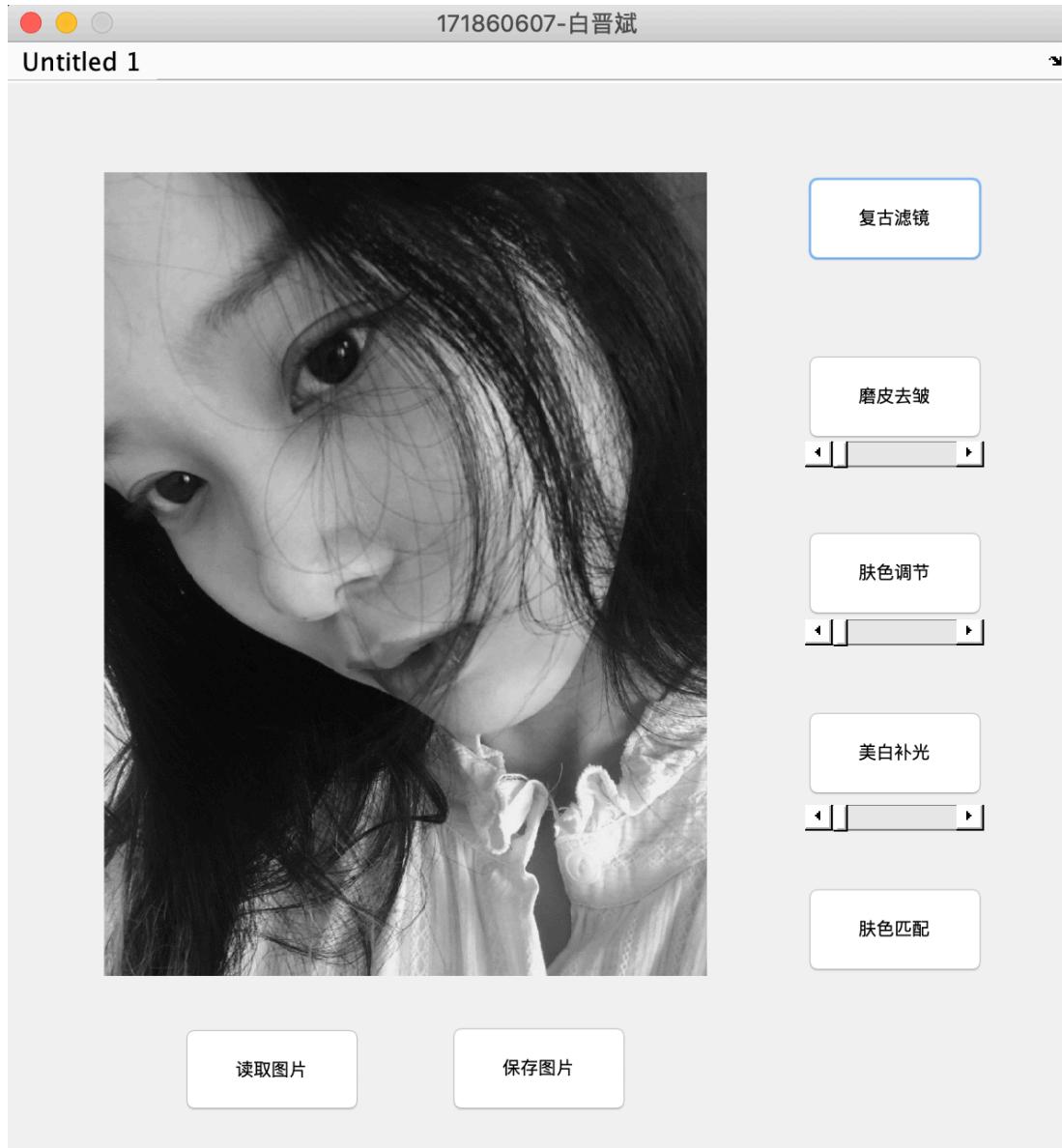
保存图片:

点击后调用系统文件夹,输入另存为的文件格式,文件名,在对应地址选择保存即可.



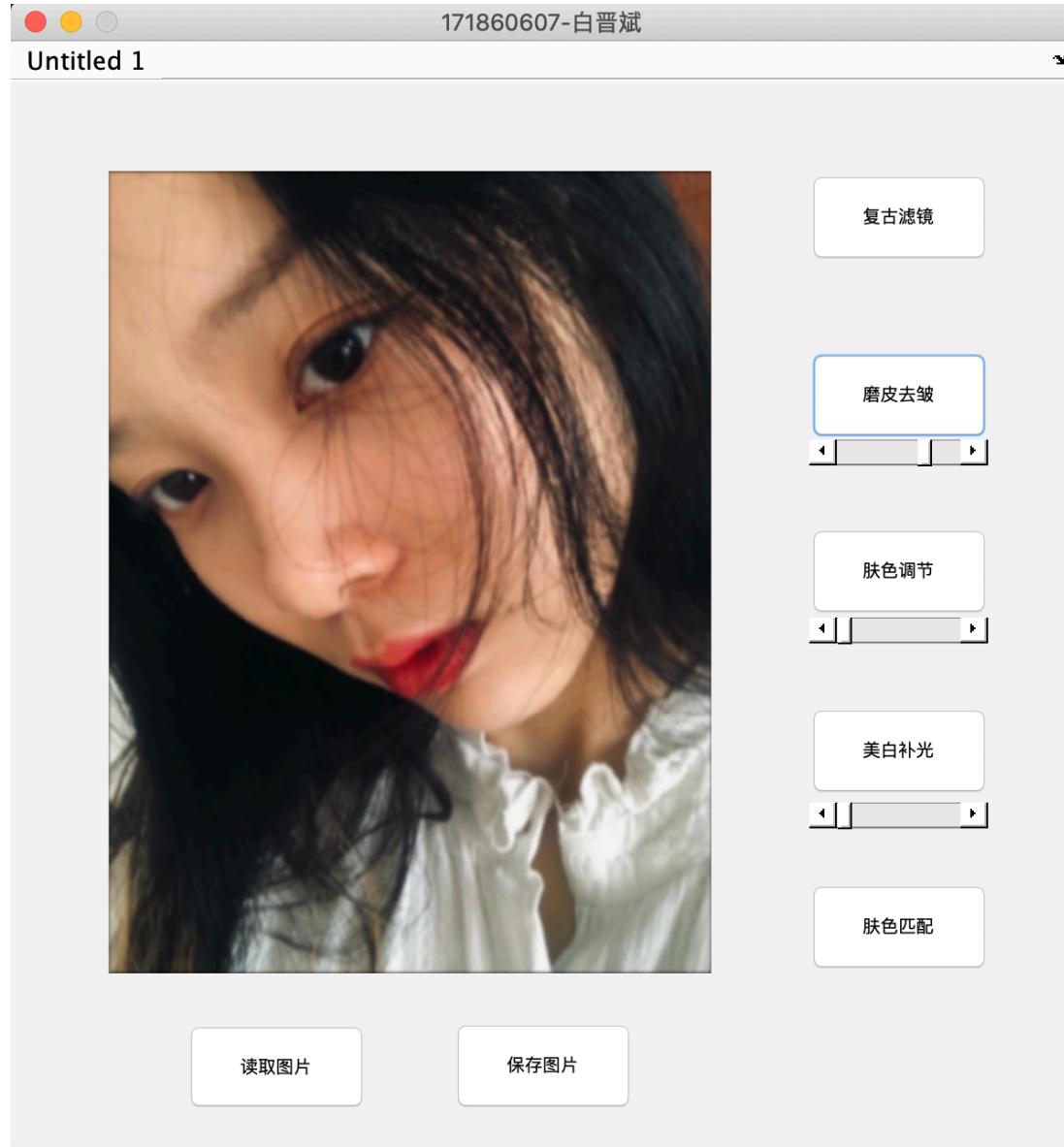
复古滤镜：

复古风~点击后将照片格式转换为灰度图,重新回到黑白照片的年代~



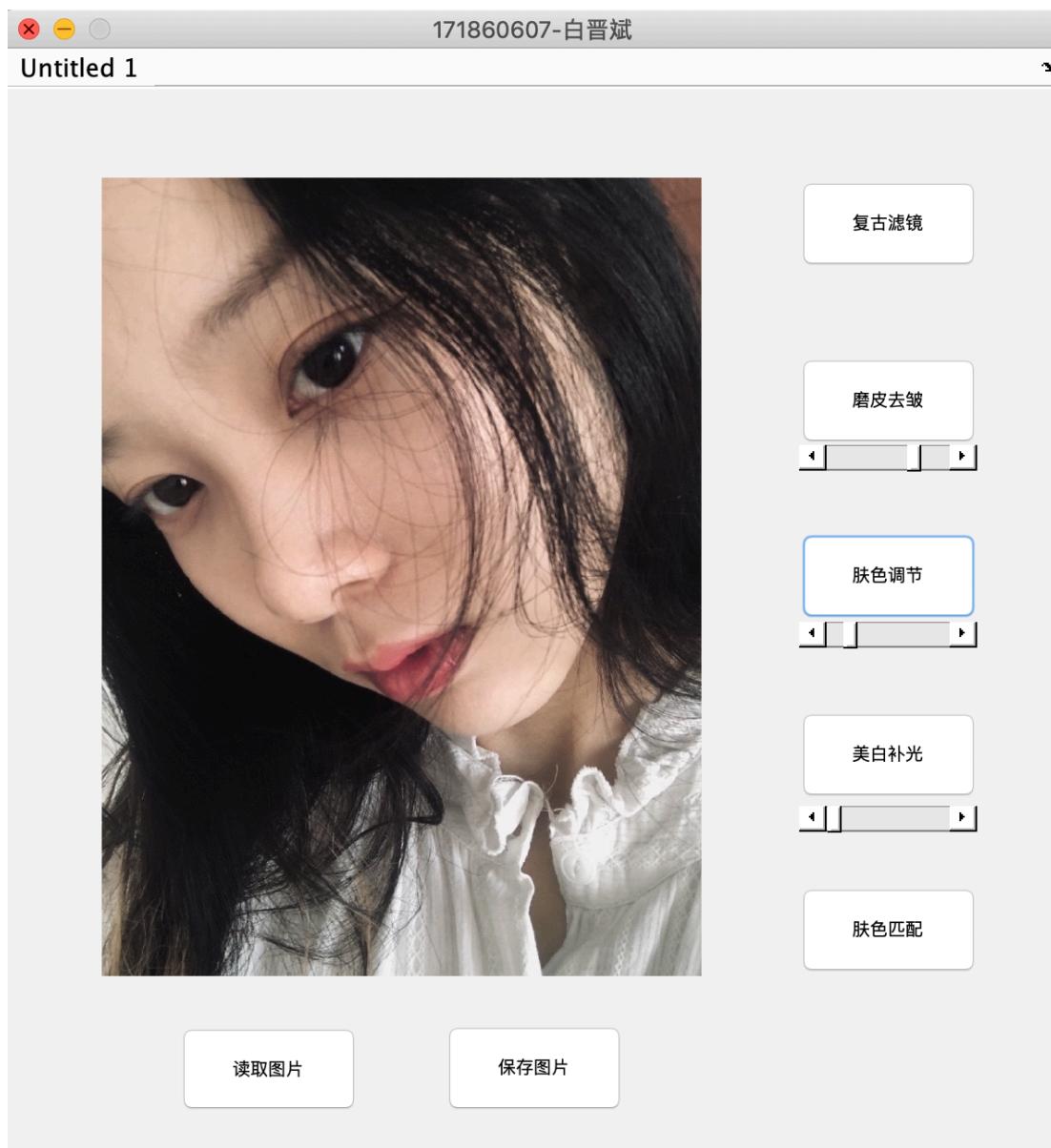
磨皮去皱:

再也不用担心脸上有雀斑啦~磨皮去皱,轻松免去化妆的困扰,调节滑动杆可调节磨皮去皱的力度,左边为力度最大,右边为力度最小(可以反过来这并不重要),中间偏右即可轻松抹去脸上雀斑~(每次拉完以后要点击磨皮去皱哦)



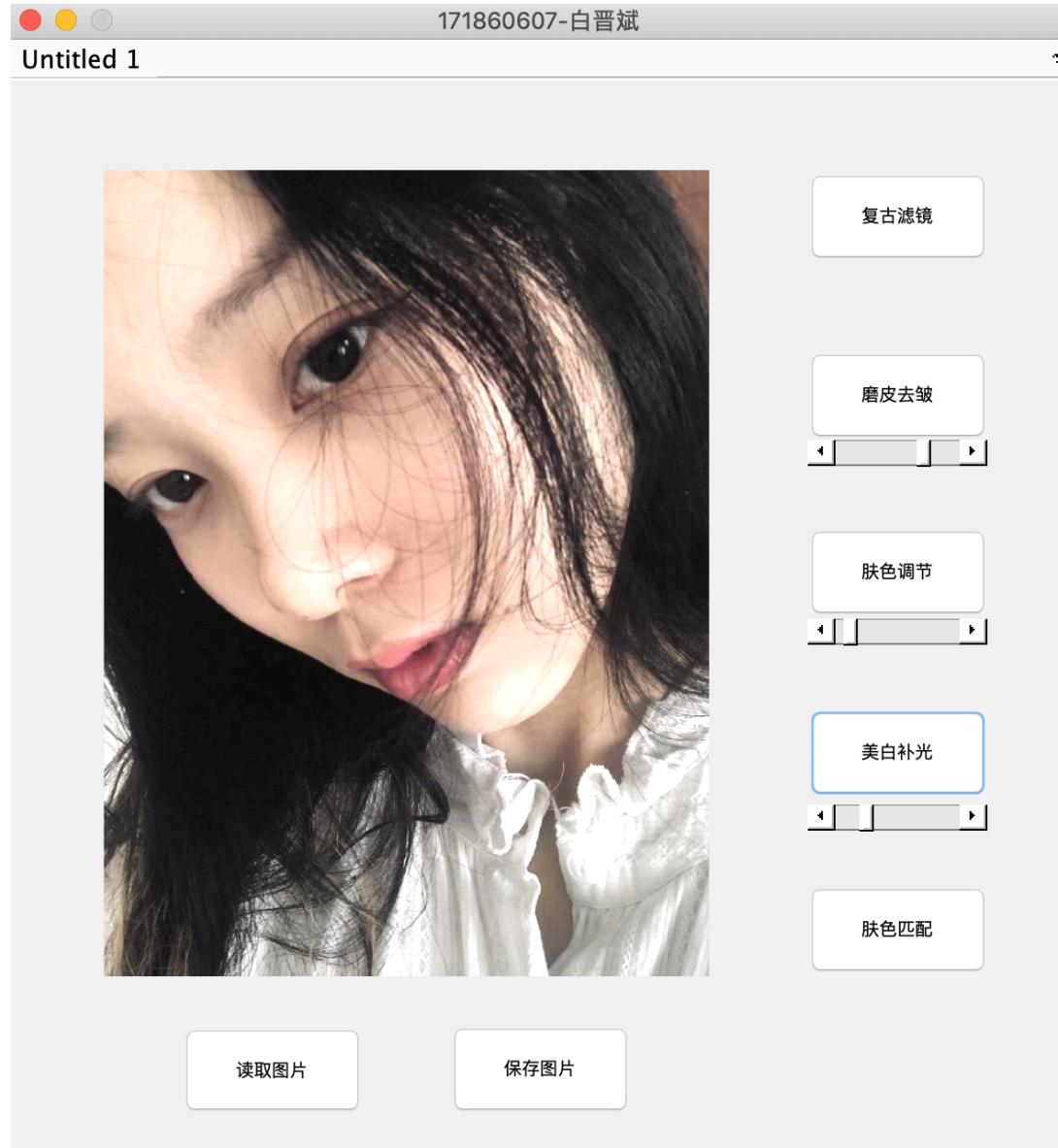
肤色调节：

感觉脸色苍白?别担心,滑动杆往右拉,轻松面色红润.感觉脸色发黑?也别担心,滑动杆往左拉,轻松美白肌肤.(每次拉完以后要点击肤色调节哦) 效果可以对比上图和下图.



美白补光:

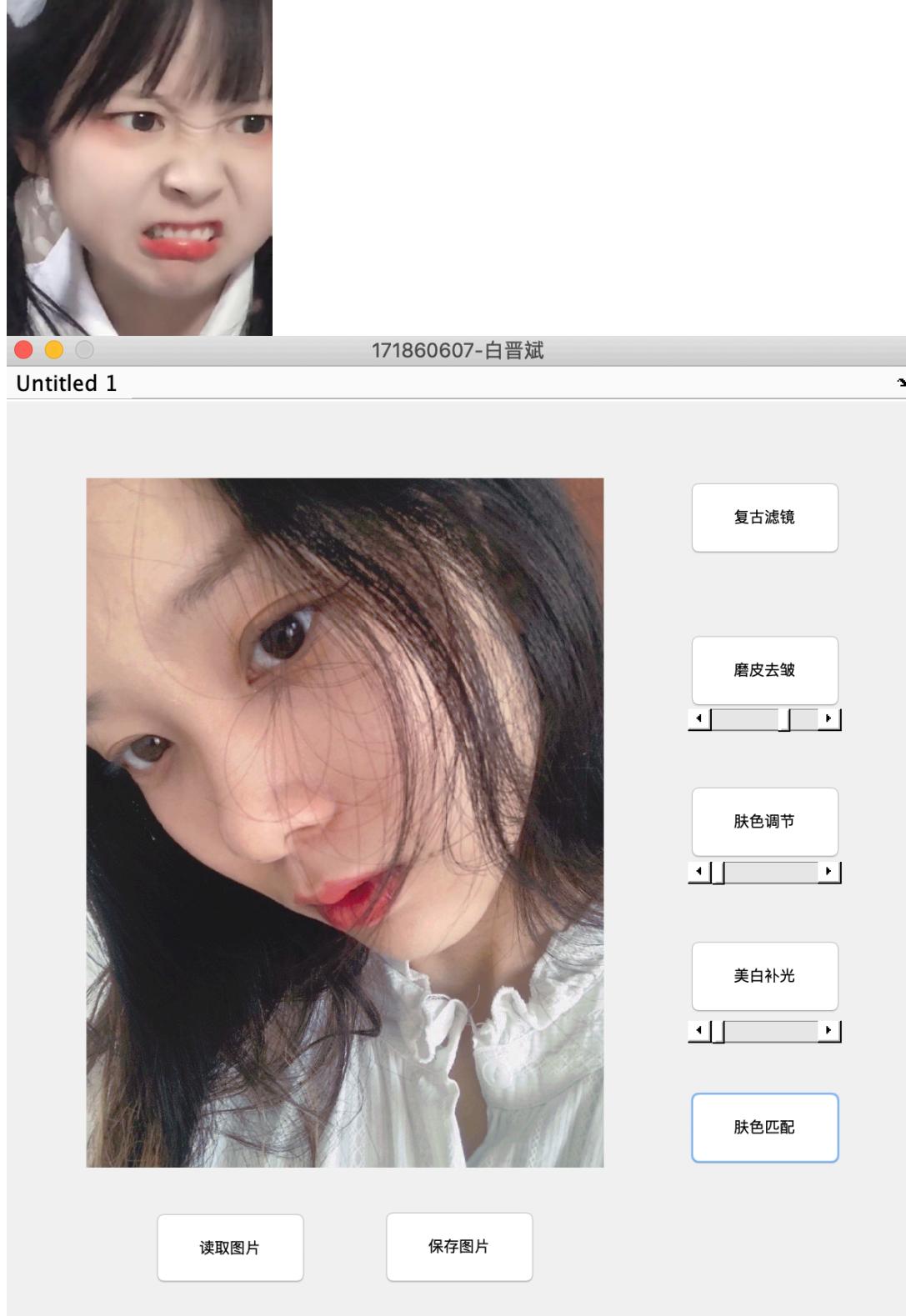
光线不好?皮肤发暗?都别担心,一键美白,一键补光,稍微把滑动杆往右拉即可实现增光效果~(每次拉完以后要点击一键美白哦)效果可以对比上图和下图.(示例选用肤色调节后的保存图 tmp.jpg 进行美白补光)



肤色匹配：

羡慕某个小姐姐的肤色?想要让自己的肤色变得和她一样?手动调节难度太高?肤色匹配拯救你!本模块的思路来源于作业一的直方图匹配.

点击肤色匹配,选择要匹配的小姐姐的照片,即可完成肤色匹配.(示例为原图(girl.jpg)匹配下左图(lassock.jpg)).



考虑到不同用户使用平台不同,使用平台的版本亦有所不同,如本人习惯使用 macOS,其他系统包括 Windows,Ubuntu 等,故这里不作打包,提供 gui.fig 和 gui.m 文件供助教测试,使用方式非常简单,使用 Matlab 打开 gui.fig 文件所在目录,双击 gui.fig 即可运行.如有任何问题可联系 QQ810594956 或 QQ 邮箱 810594956@.com.