傅立叶分析和小波分析实验报告

课程名称：傅立叶分析和小波分析

实验项目名称：数字图像处理 实验时间：2022.4.21

班级； 信计1901 姓名： 唐川淇 学号： 1131190111

**实验目的：**

* 二维信号的小波分析处理方式

**实 验 环 境:**

Matlab

**实验步骤：**

学习并实现二维信号的小波分析处理，了解相关其他方法

参考链接：https://blog.csdn.net/qq\_23023937/article/details/109158715

**实 验 心 得：**

1. 预处理

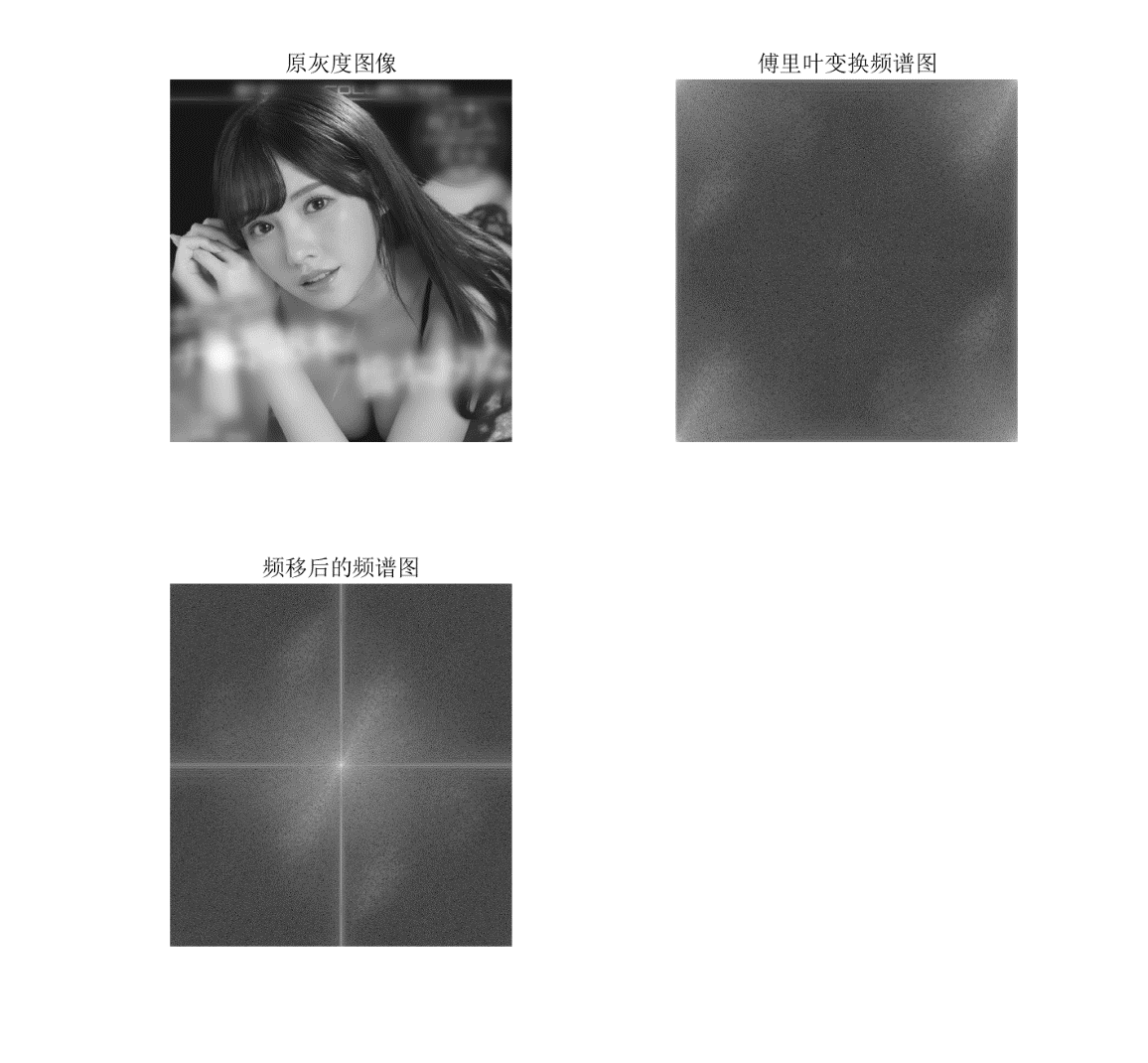
**选取桥本有菜图片，做预处理。**

|  |
| --- |
| **读取图像、转为灰度图、直方图** |
| **clear ;**  **close all;**  **color\_pic=imread('JBYC.png'); %读取图像**  **figure('name','图像读取');**  **subplot(2,2,1);**  **imshow(color\_pic); %显示图像**  **title('原彩色图像');**  **gray\_pic=rgb2gray(color\_pic); %将彩色图转换成灰度图**  **subplot(2,2,2);**  **imshow(gray\_pic);**  **title('灰度图像');**  **subplot(2,2,3);**  **imhist(gray\_pic); %查看灰度直方图，默认n=256,256个长度为1的灰度空间**  **title('灰度直方图256等级');**  **subplot(2,2,4);**  **imhist(gray\_pic,64);%n=64，64个长度为4的灰度空间**  **title('灰度直方图64等级');** |

****

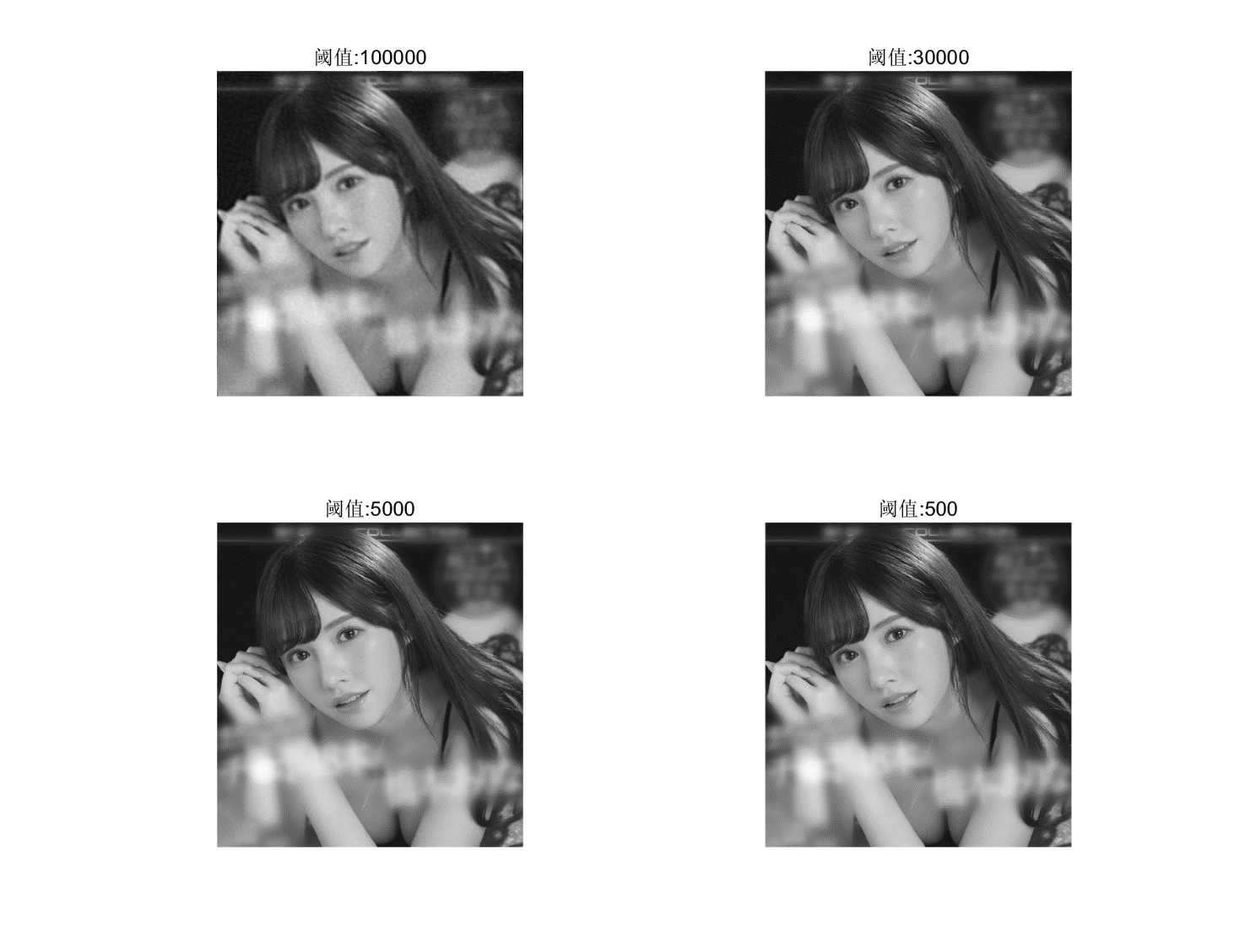
1. 傅里叶变换

|  |
| --- |
| **傅里叶变换** |
| **clear; %清除变量**  **close all; %关闭生成的画图窗口**  **color\_pic=imread('JBYC.png'); %读取图像**  **gray\_pic=rgb2gray(color\_pic); %将彩色图转换成灰度图**  **figure('name','傅里叶变换'); %傅里叶变换**  **subplot(2,2,1);**  **imshow(gray\_pic);**  **title('原灰度图像');**  **Fourier=fft2(gray\_pic); %对灰度图像进行傅里叶正变换**  **log\_Fourier=log(abs(Fourier)+1); %取模并进行缩放，调高频谱图像的低灰度值而对高灰度值仅可能减小**  **subplot(2,2,2);**  **imshow(log\_Fourier,[]); %未进行频谱搬移时的频谱图**  **title('傅里叶变换频谱图');**  **Fourier\_shift=fftshift(Fourier); %将频谱图中零频率成分移动至频谱图中心**  **log\_Fourier\_shift=log(abs(Fourier\_shift)+1); %取模并进行缩放，对于（0，1）之间的x值经过取对数后会变成负值，而log（x+1）则将所有的x值映射到正数范围内**  **subplot(2,2,3);**  **imshow(log\_Fourier\_shift,[]);**  **title('频移后的频谱图');** |

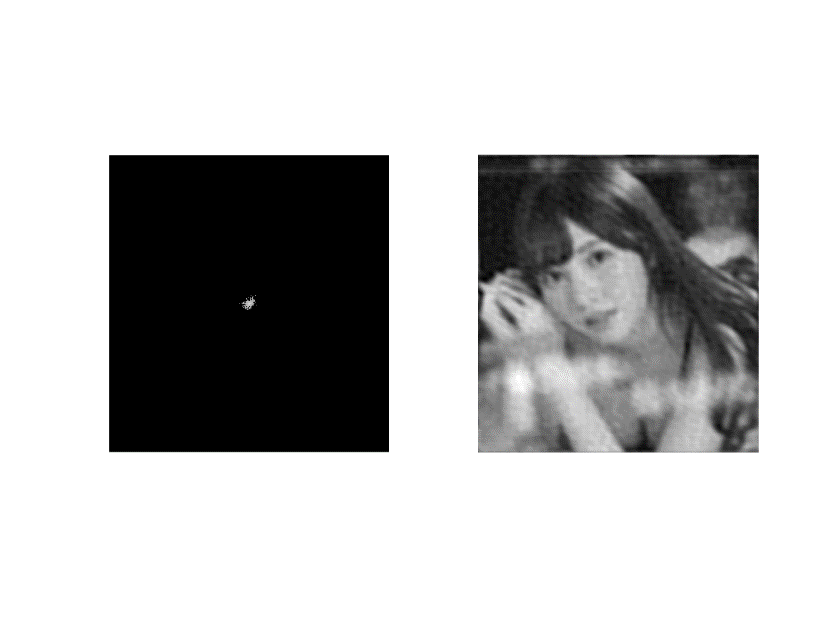


1. 傅里叶逆变换

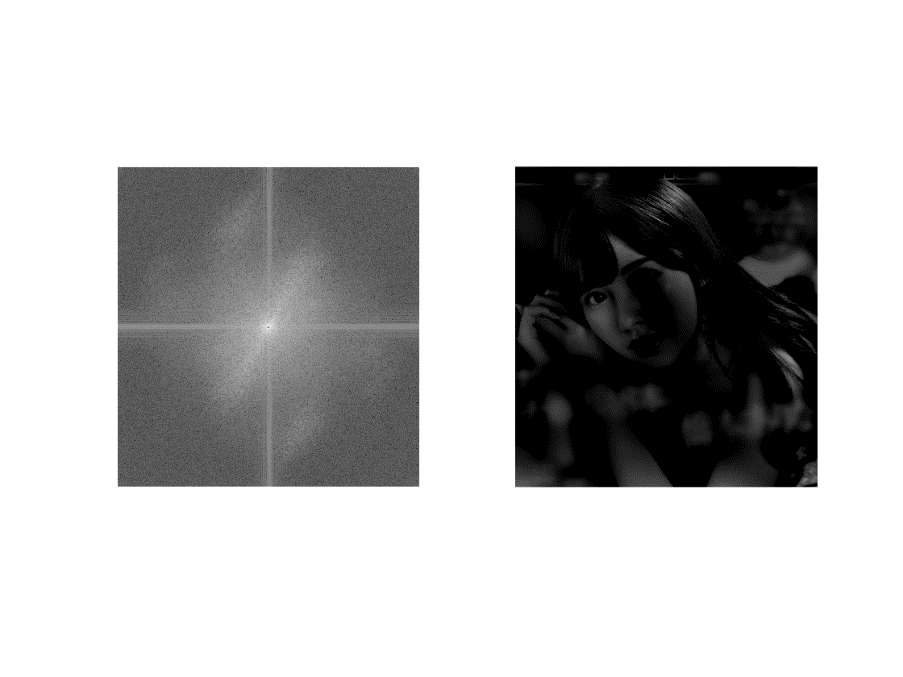
|  |
| --- |
| **傅里叶逆变换** |
| **clear;**  **close all;**  **color\_pic=imread('JBYC.png'); %读取图像**  **gray\_pic=rgb2gray(color\_pic); %将彩色图转换成灰度图**  **threshold=[100000,30000,5000,500]; %设置不同阈值 (高频部分能量低)**  **figure('name','傅里叶逆变换图像');**  **for i=1:4**  **Fourier=fft2(gray\_pic); %对灰度图像进行傅里叶正变换**  **Fourier\_shift=fftshift(Fourier); %将频谱图中零频率成分移动至频谱图中心**  **h\_Fourier\_shift=abs(Fourier\_shift);% 取傅里叶变换后幅度模值，使灰度值为正数**  **Fourier\_shift(h\_Fourier\_shift<threshold(i))=0; %取阈值消除部分高频**  **IFourier=real(ifft2(ifftshift(Fourier\_shift))); %傅里叶逆变换，要记得先把频移的频谱频移回去**  **ret=uint8(IFourier); %将灰度级映射到0-255上**  **subplot(2,2,i);**  **imshow(ret);**  **str=num2str(threshold(i));**  **title(['阈值:',str]);**  **end** |



**低频对应着整体信息，例如轮廓；而高频对应着细节，例如头发的线条感。去除图像的高频信息可以达到图像压缩的效果。**

****

**现在我们只保留图中的中间部分，也就是图左侧所示，你可以看到只有中间一个亮点（对应着图像的低频信息），其他的把它“滤”掉了。对图左进行**[**傅立叶变换**](https://www.zhihu.com/search?q=%E5%82%85%E7%AB%8B%E5%8F%B6%E5%8F%98%E6%8D%A2&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A1473209132%7D)**，得到图右。我们大概可以看出来这是一个人的头部，但是丢失了细节。**

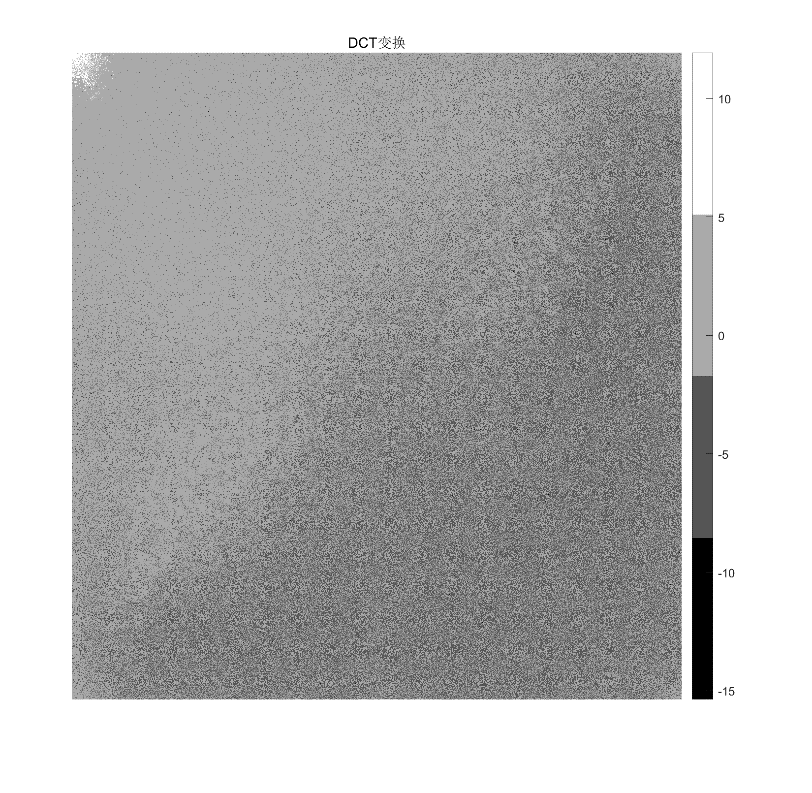
****

**现在我们保留图1中除去中间的部分，也就是图左侧所示，你可以看到中间被挖去了，剩下挖去中间的外面部分（对应着图像的高频信息）。对图左进行傅立叶变换，得到图右。我们现在看到这幅图的细节都在（比如头发的质感），但是丢失了一些整体信息。**

1. 离散余弦变换(DCT)

|  |
| --- |
| **离散余弦变换(DCT)** |
| **clear;**  **close all;**  **color\_pic=imread('JBYC.png'); %读取图像**  **gray\_pic=rgb2gray(color\_pic); %将彩色图转换成灰度图**  **figure('name','DCT变换');**  **DCT=dct2(gray\_pic); %DCT变换**  **h\_DCT=abs(DCT); %DCT变换后的幅度**  **log\_DCT=log(h\_DCT); %取模并进行缩放，调高频谱图像的低灰度值而对高灰度值仅可能减小**  **imshow(log\_DCT,[]);**  **title('DCT变换');**  **colormap(gray(4)); %重新设置灰度级为4，便于查看DCT变换后的频谱图特点**  **colorbar; %显示颜色栏** |

**从DCT频谱图易看出，低频部分(图像轮廓)能量集中在左上角，因此可进行图像压缩。**



1. DCT逆变换

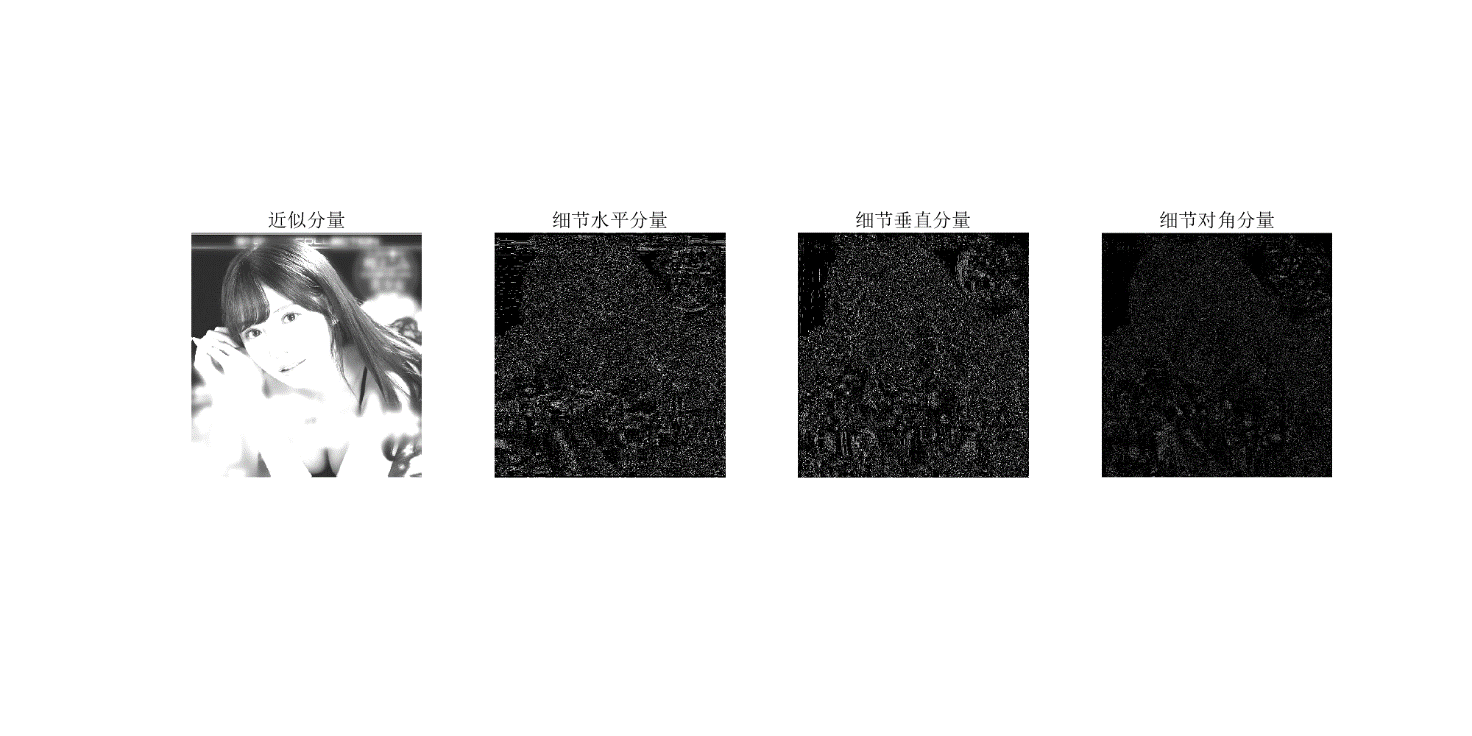
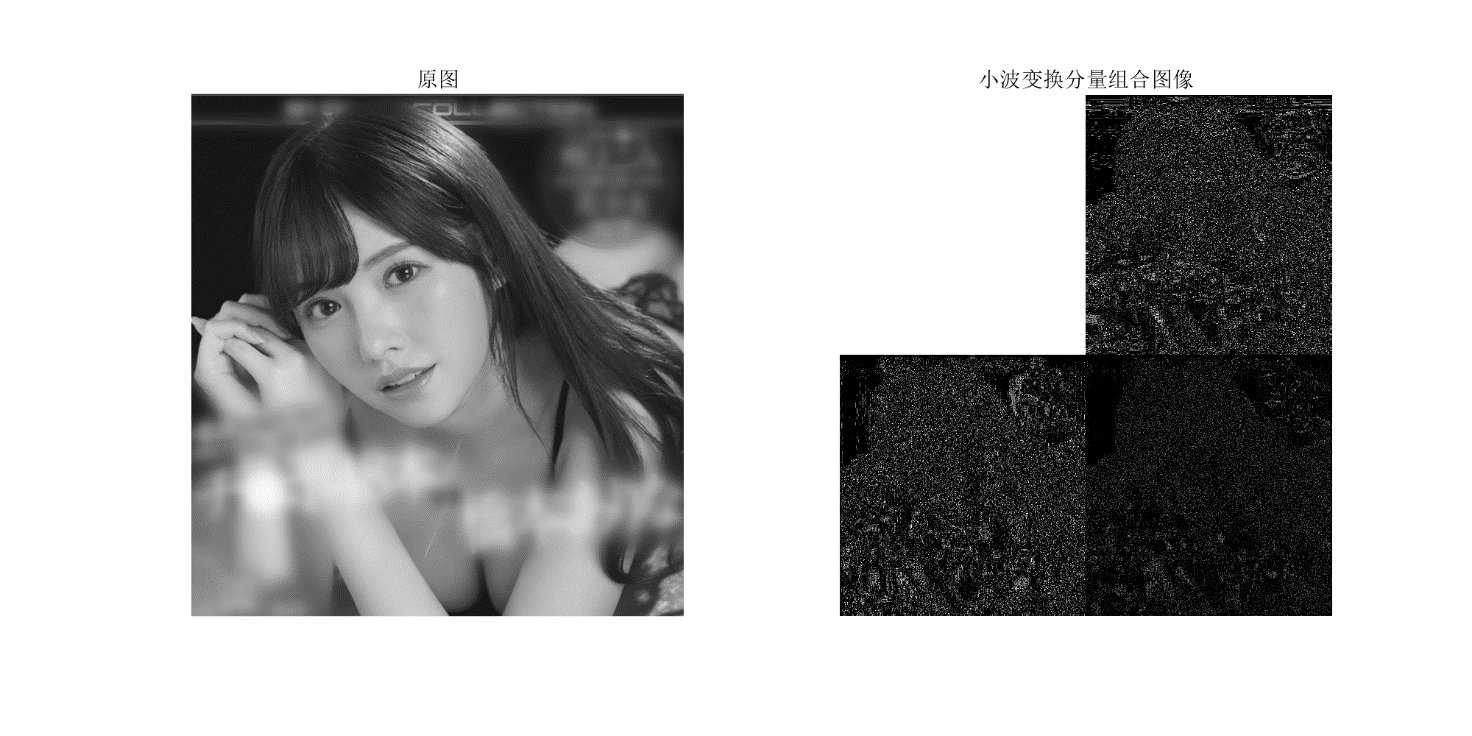
|  |
| --- |
| **DCT逆变换** |
| **close all;**  **color\_pic=imread('JBYC.png'); %读取图像**  **gray\_pic=rgb2gray(color\_pic); %将彩色图转换成灰度图**  **threshold=[200,100,10,1]; %设置不同阈值 (高频部分能量低)**  **for i=1:4**  **DCT=dct2(gray\_pic); %DCT正变换**  **h\_DCT=abs(DCT); %DCT变换后的幅度**  **DCT(h\_DCT<threshold(i))=0; %DCT变换后取模得出的幅度值若小于20则至0**  **IDCT=uint8(idct2(DCT)); %对图像进行DCT逆变换，并将灰度级映射到0-255上**  **subplot(2,2,i);**  **imshow(IDCT);**  **str=num2str(threshold(i));**  **title(['阈值:',str]);**  **end** |



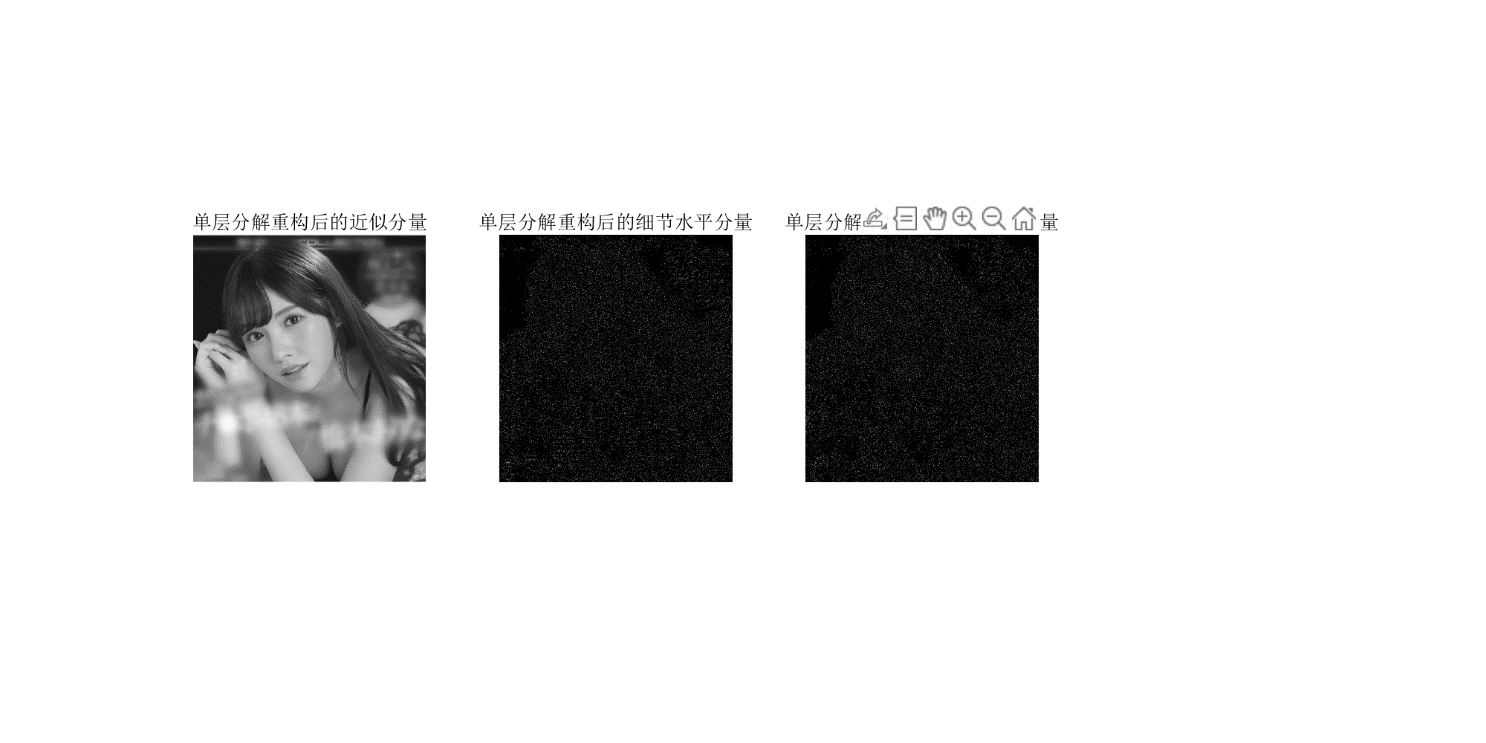
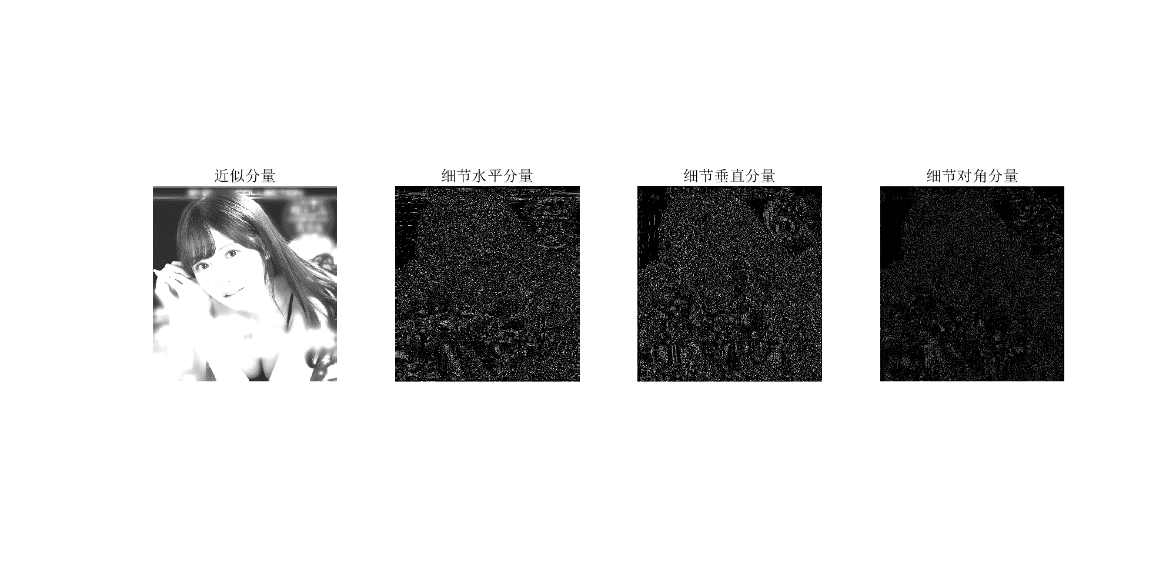
**因为高频部分能量较低，即DCT变换后的高频部分幅度值较低，当设置的阈值越小时，保留了更多低频部分，即轮廓部分保留下来，图像也就恢复的越接近原图。**

1. 小波变换与重构

|  |
| --- |
| **小波变换** |
| **close all; %关闭当前所有图形窗口**  **clear all; %清楚工作空间变量**  **clc; %清除命令行数据**  **X=imread('JBYC.png'); %读取图像**  **X=rgb2gray(X); %转换为灰度图像**  **[ca1,chd1,cvd1,cdd1] = dwt2(X,'bior3.7'); %进行单层小波分解**  **set(0,'defaultFigurePosition',[100,100,1000,500]); %修改图形图像位置的默认设置**  **set(0,'defaultFigureColor',[1 1 1])**  **figure %显示小波变换各个分量**  **subplot(141);**  **imshow(uint8(ca1)); %近似系数分量**  **title('近似分量')**  **subplot(1,4,2);**  **imshow(chd1); %细节系数的水平分量**  **title('细节水平分量')**  **subplot(1,4,3);**  **imshow(cvd1); %细节系数的垂直分量**  **title('细节垂直分量')**  **subplot(1,4,4);**  **imshow(cdd1); %细节系数的对角分量**  **title('细节对角分量')**  **figure**  **subplot(121),imshow(X); %显示原图**  **title('原图')**  **subplot(122),imshow([ca1,chd1;cvd1,cdd1]); %显示小波变换分量组合图像**  **title('小波变换分量组合图像')** |



|  |
| --- |
| **重构** |
| **close all; %关闭当前所有图形窗口**  **clear all; %清楚工作空间变量**  **clc; %清除命令行数据**  **X=imread('JBYC.png'); %读取图像**  **X=rgb2gray(X); %转换为灰度图像**  **[c,s] = wavedec2(X,2,'db4'); %对图像进行小波2层分解**  **siz = s(size(s,1),:); %提取第2层小波分解系数矩阵大小**  **ca2 = appcoef2(c,s,'db4',2); %提取第1层小波分解的近似系数**  **chd2 = detcoef2('h',c,s,2); %提取第1层小波分解的细节系数水平分量**  **cvd2 = detcoef2('v',c,s,2); %提取第1层小波分解的细节系数垂直分量**  **cdd2 = detcoef2('d',c,s,2); %提取第1层小波分解的细节系数对角分量**  **a2 = upcoef2('a',ca2,'db4',2,siz); %提取第1层小波分解的细节系数水平分量**  **hd2 = upcoef2('h',chd2,'db4',2,siz); %利用函数upcoef2()实现图像多层小波重构**  **vd2 = upcoef2('v',cvd2,'db4',2,siz);**  **dd2 = upcoef2('d',cdd2,'db4',2,siz);**  **A1=a2+hd2+vd2+dd2; %重构近似图像**  **[ca1,ch1,cv1,cd1] = dwt2(X,'db4'); %对图像进行小波单层图像分解**  **a1 = upcoef2('a',ca1,'db4',1,siz); %利用函数upcoef2()提取1层小波分解系数进行重构**  **hd1 = upcoef2('h',cd1,'db4',1,siz);**  **vd1 = upcoef2('v',cv1,'db4',1,siz);**  **dd1 = upcoef2('d',cd1,'db4',1,siz);**  **A0=a1+hd1+vd1+dd1; %重构近似图像**  **set(0,'defaultFigurePosition',[100,100,1000,500]); %修改图形图像的默认设置**  **set(0,'defaultFigureColor',[1 1 1]) %修改图形背景颜色的设置**  **figure %显示2层分解重构后的分量**  **subplot(141),imshow(uint8(a2));**  **title('2层分解重构后的近似分量')**  **subplot(142),imshow(hd2);**  **title('2层分解重构后的细节水平分量')**  **subplot(143),imshow(vd2);**  **title('2层分解重构后的细节垂直分量')**  **subplot(143),imshow(dd2);**  **title('2层分解重构后的细节对角分量')**  **figure %显示单层分解重构后的分量**  **subplot(141),imshow(uint8(a1));**  **title('单层分解重构后的近似分量')**  **subplot(142),imshow(hd1);**  **title('单层分解重构后的细节水平分量')**  **subplot(143),imshow(vd1);**  **title('单层分解重构后的细节垂直分量')**  **subplot(143),imshow(dd1);**  **title('单层分解重构后的细节对角分量')**  **figure**  **subplot(131),imshow(X); %显示原图像**  **title('原图像')**  **subplot(132),imshow(uint8(A1)); %显示2层分解重构后的分量**  **title('2层分解重构后的近似图像')**  **subplot(133),imshow(uint8(A0)); %显示单层分解重构后的分量**  **title('单层分解重构后的近似图像')** |



1. 小波去噪

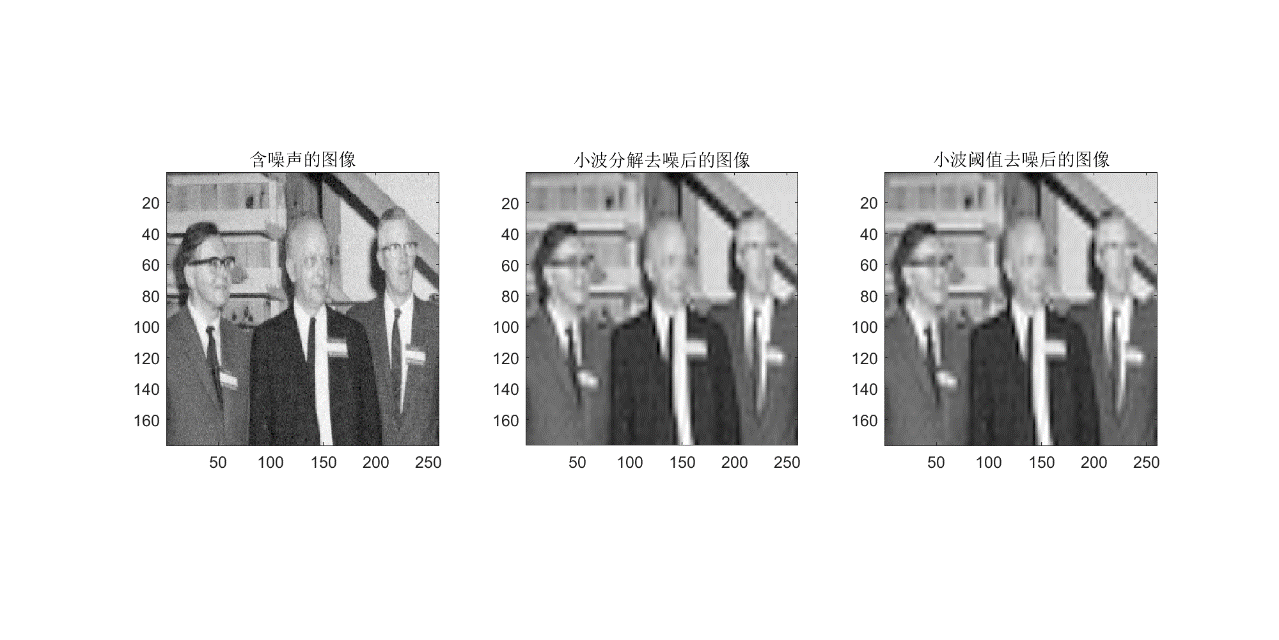
**目前，小波去噪的方法大概可以分为三大类[1]:**

**小波萎缩法是目前研究最为广泛的方法, 小波 萎缩法又分成如下两类: 第 1 类是阈值萎缩，第2类是比例萎缩。**

**投影方法的原理就在于将带噪信号以一种迭代的方式, 投影到逐步缩小的空间, 由于最后的空间能更好地体现原信号的特点, 所以投影法也能够有效地区分噪声和信号. 投影方法有M atch ing Pu rsu its法 和 M CD (M u ltip le Com pact Dom ain ) 或 POCS(P ro jection O n to Convex Set ) 法 两 类.**

**相关方法主要是基于信号在各层相应位置上的小波系数之间往往具有很强的相关性, 而噪声的小波系数则具有弱相关或不相关的特点来进行去噪的。**

|  |
| --- |
| **小波去噪** |
| **close all; %关闭当前所有图形窗口**  **clear all; %清楚工作空间变量**  **clc; %清除命令行数据**  **load gatlin2; %装载并显示原始图像**  **init=2055615866;**  **randn('seed',init) %随机值**  **XX=X+2\*randn(size(X)); %添加随机噪声**  **[c,l]=wavedec2(XX,2,'sym4'); %对图像进行清噪处理，用sym4小波函数对x进行两层分解**  **a2=wrcoef2('a',c,l,'sym4',2); %重构第二层图像的近似系数**  **n=[1,2]; %设置尺度向量**  **p=[10.28,24.08]; %设置阈值向量**  **nc=wthcoef2('t',c,l,n,p,'s'); %对高频小波系数进行阈值处理**  **mc=wthcoef2('t',nc,l,n,p,'s'); %再次对高频小波系数进行阈值处理**  **X2=waverec2(mc,l,'sym4'); %图像的二维小波重构**  **set(0,'defaultFigurePosition',[100,100,1000,500]); %修改图形图像的默认设置**  **set(0,'defaultFigureColor',[1 1 1]) %修改图形背景颜色的设置**  **figure**  **colormap(map)**  **subplot(131),image(XX),axis square; %显示含噪声的图像**  **title('含噪声的图像')**  **subplot(132),image(a2),axis square; %显示小波分解去噪后的图像**  **title('小波分解去噪后的图像')**  **subplot(133),image(X2),axis square; %小波阈值去噪后的图像**  **title('小波阈值去噪后的图像')**  **ps=sum(sum((X-mean(mean(X))).^2)); %计算信噪比**  **pn=sum(sum((a2-X).^2));**  **disp('利用小波二层分解去噪的信噪比')**  **snr1=10\*log10(ps/pn)**  **disp('利用小波阈值去噪的信噪比')**  **pn1=sum(sum((X2-X).^2));**  **snr2=10\*log10(ps/pn1)** |

****

1. 心得体会

**小波具备良好的时频信息，在图像去噪中可以得到很好的应用，具体来说，小波去噪的优势主要在以下几个方面[2]：（1）低熵性，小波系数稀疏分布，使得图片变换后熵降低。（2）多分辨率，由于采取了多分辨率的方法，所以可以很好的刻画信号的非平稳特征。（3）去相关性，小波变换可以对信号进行去相关。（4）选基灵活，可以根据不同的研究对象选择不同的小波母函数。**

**附 录：**

**[1]** **谢杰成, 张大力, 徐文立. 小波图象去噪综述[J]. 中国图象图形学报, 2002, 7(3): 209-217.**

**[2] Vidakovic B, Lozoya C B. On time-dependent wavelet denoising[J]. IEEE Transactions on Signal Processing, 1998, 46(9): 2549-2554.**

备注：以上各项空白处若填写不够，可自行扩展