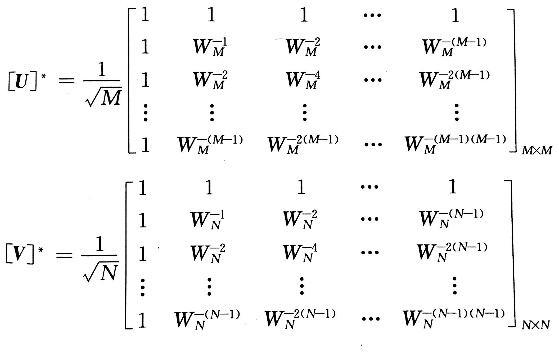
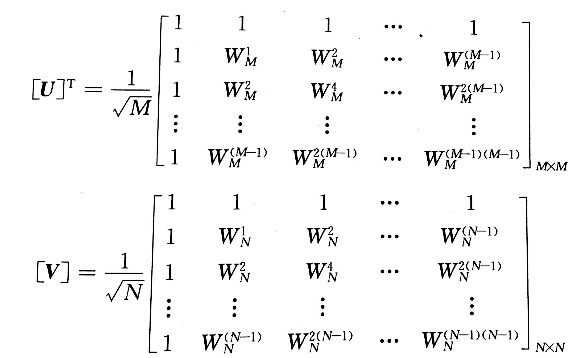
1. 代码设计思路

二维离散傅里叶变换及其反变换函数(I)DFT2根据课本中所介绍算法进行设计。首先读取灰度图像，读取该灰度矩阵规模M、N。随后根据课本中介绍的算法构造M\*M的U及N\*N的V矩阵，矩阵元素由下式给出。



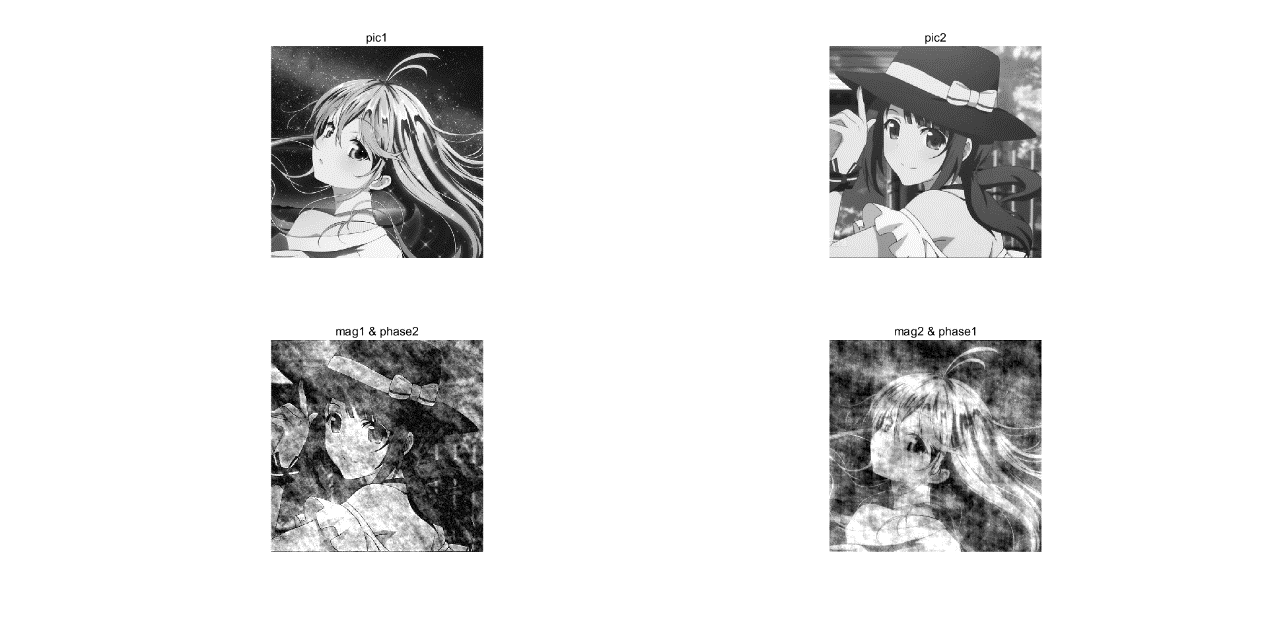
随后根据将三个矩阵相乘，即可得到原灰度图像的二维离散傅里叶变换。IDFT的实现方式与以上类似，。

我们同时编写了测试代码以完成题目要求。首先从工作文件夹中读取图像，之后将两张图片裁剪至相同尺寸以便于进行相位和幅度互换。根据要求采用rgb2gray()函数将两张图片转换为灰度图像。采用自己编写的DFT2()对灰度图像做二维离散傅里叶变换。之后采用MATLAB的abs()和angle()函数提取出幅度谱A及相位谱φ。根据下式

交换两张图片的幅度谱及相位谱。然后采用自己编写的 IDFT2()得到结果图像。利用subplot()函数展示结果。

1. 实验结果

由结果图像可以看出，由图一的幅度和图二的相位构成的结果图像保持了图二的轮廓，而图二的轮廓和图一的相位构成的图像则保持了图一的轮廓。这一结果可以说明幅度谱保持了图像更多的轮廓信息，相位谱相比幅度谱更为重要。



1. 学习体会

本实验采用自己编写的DFT2和IDFT2函数实现了二维离散傅里叶（反）变换。测试前先用MATLAB中fft2()和ifft2()函数完成幅度谱和相位谱互换的功能。之后自己编写函数以取代MATLAB内建函数。测试过程中发现课本中正反变换均需要乘以系数，而MATLAB内建函数fft2()没有乘以系数，而反变换则乘以了。尽管最终结果相同，但给调试过程带来了一点困难。