

实验 3-A. 变换域图像处理

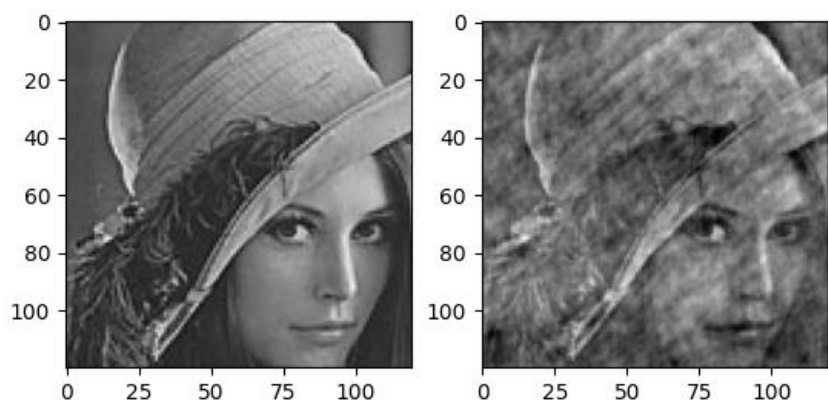
仔细阅读钉钉群中上传的代码 DFT-IDFT.ipynb、ft_merge.ipynb 和 low-high-filter.ipynb，编写程序实现：

1. 打开 Lenna.jpg，取绿色分量图像左上角为(100,100)，大小为 64*64 的一个正方形区域，作傅立叶变换，且计算频谱，屏幕打印输出 $0 \leq u, v \leq 10$ 范围内的频谱值。傅立叶变换和频谱计算的参考函数如下：

```
9 # cv2.dft(im, flags)
10 # im--要求是灰度图像，且需要转换为 np.float32格式，flags: 转换的标识符，取值如下:
11 # DFT_COMPLEX_OUTPUT -- 输出的是一个2 channel的矩阵，第一个channel是变换结果的实数部分，第二个channel为虚数部分
12 # DFT_REAL_OUTPUT -- 只输出实数部分 ....
13 |
14 dft_ori = cv2.dft(np.float32(im), flags=cv2.DFT_COMPLEX_OUTPUT)
```

```
21 # 计算幅度值, magnitude()
22 img_b = cv2.magnitude(img_dft[:, :, 0], img_dft[:, :, 1])
```

2. 打开 Lenna.jpg，取红色分量图像作二维 DFT 变换，将频率坐标的原点移到图像中心位置，设置边长分别为 12、24、32 的正方形 mask，作低通滤波，观察不同 mask 对逆变换图像的影响。
3. 打开 Lenna.jpg，取红色分量图像左上角为(100,100)，大小为 8*8 的一个正方形区域，作傅立叶变换，且计算频谱，根据频谱的中心对称特征，找出分别与位置 (2, 3), (3, 2), (3, 5) 频谱值相等的坐标位置。
4. 编写代码由 Lenna 面部图像(LennaFace.png)的相位谱与 Flower 局部图像(Flower_1x1.png)的幅度谱混合产生合成图像(结果可参见如下的合成图像)。

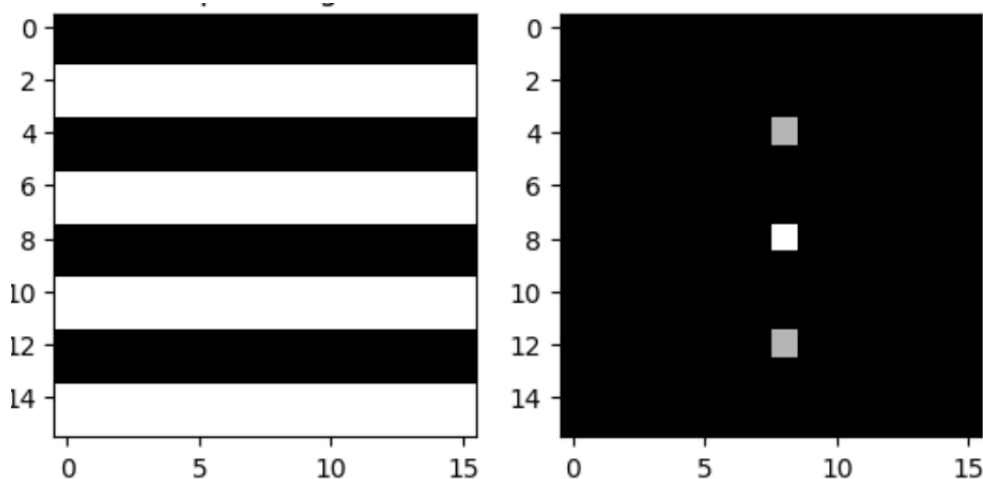


实验 3-B

1. 打开 LennaFace.png，对其作傅里叶变换且将频率坐标原点移到图像平面中心，计算幅度谱；对 LennaFace.png 图像作直方图均衡化处理，作傅里叶变换且将频率坐标原点移到图像平面中心，计算幅度谱。比较上述两个幅度谱的分布有何特征？（可显示结合打印输出比较）

实验 3-C

1. 使用 numpy 包生成如下左图图像（16*16 大小，高度方向隔 2 个像素分布 2 像素宽度的水平条带，白色对应 255 灰度，黑色对应 0），编写代码验证该图像经傅里叶变换后的频谱如右图所示（即非零频谱只分布于与 Y 轴对应的频率轴），请思考原因。



2. 若需要频谱为上图频谱绕中心顺时针旋转 90 度的形式（参见下图），则对应的图像应该是怎样的图像？请思考原因。

