**实验二：数字图像频域增强实验**

**作业一：彩色图像的频域滤波器**

问题1：

MATLAB有对空间域进行滤波的fspecial和imfilter函数，没有对频域进行滤波的函数。

\*（查资料有查到lpfilter函数可以对频域进行滤波，但是Mathworks似乎在早些的某个MATLAB版本中将该函数移除，原因未知）

**算法原理：**由于是对彩色图像进行滤波，所以在此处的算法里仍需要将图像的RGB通道分开。

第一步：拆分出彩色图像的R, G, B通道；

第二步：对每个通道进行傅里叶变换，将其从空域转为频域。

第三步：对每个通道的频域分别进行高斯滤波操作。对整个图像进行加权平均，用一个模板扫描图像中的每一个像素，每一个像素点的值都由其本身和领域内的其他像素值加权平均后得到。距图像中间位置越近的点，其权重越大。

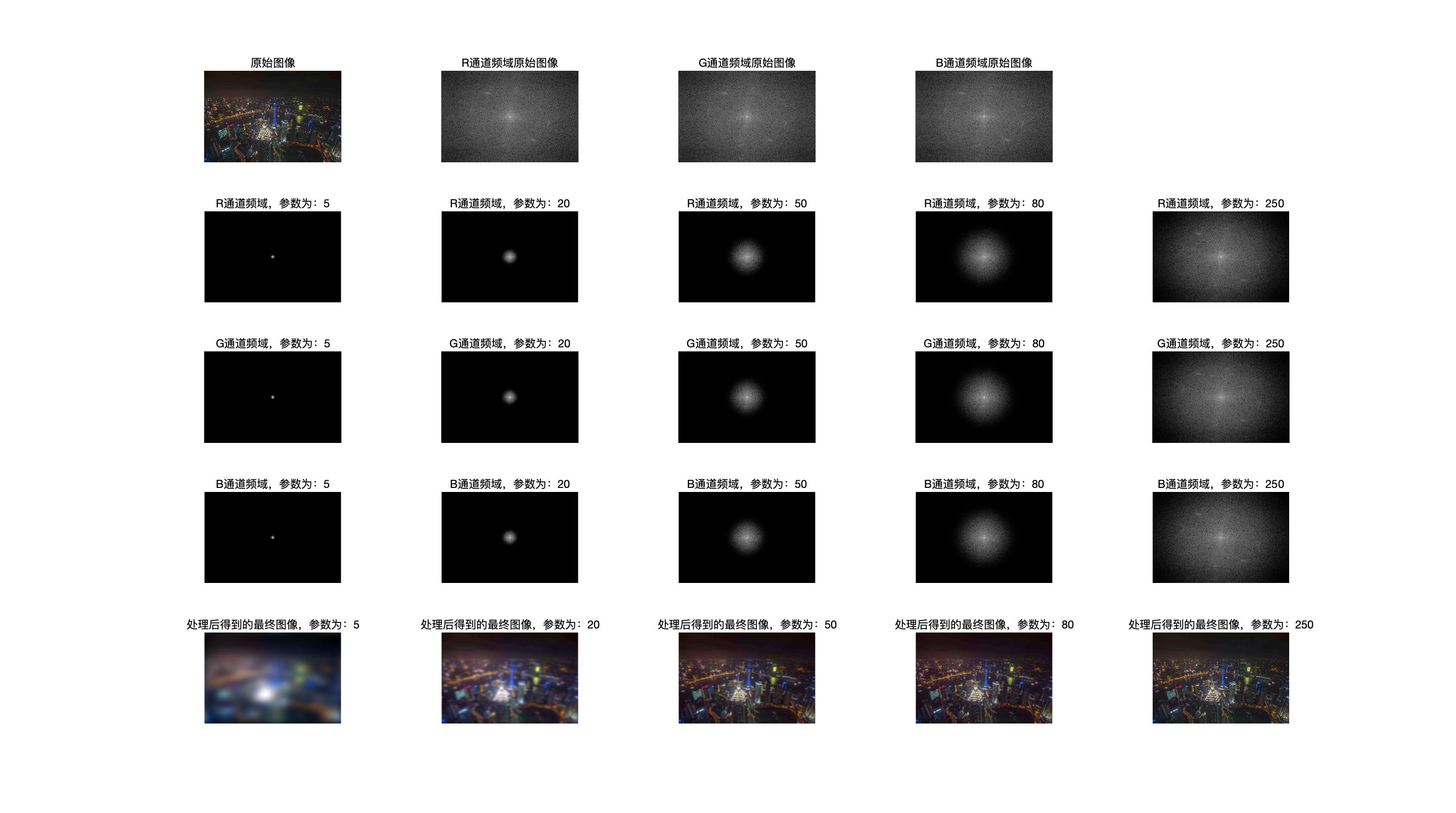
第四步：对每个通道的频域进反傅里叶变换，将其从频域转为空域。

第五步：把转换后得到的空域进行合成，得到滤波后的图像。

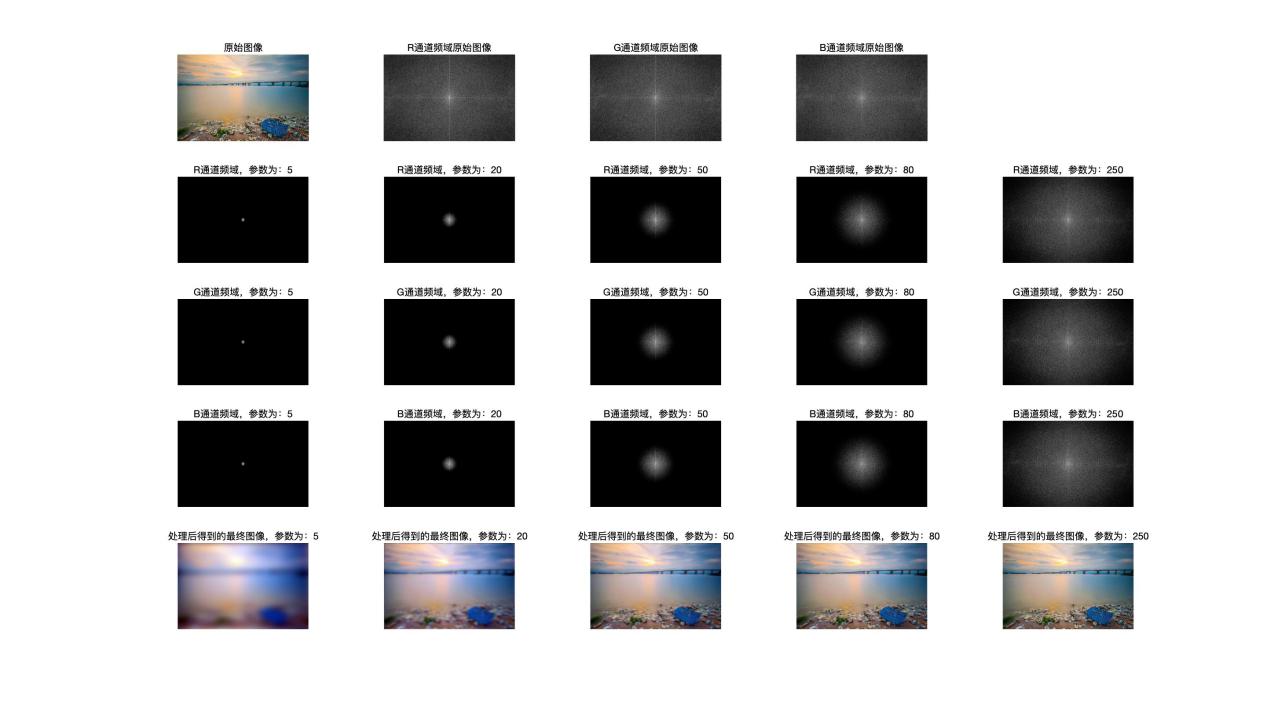
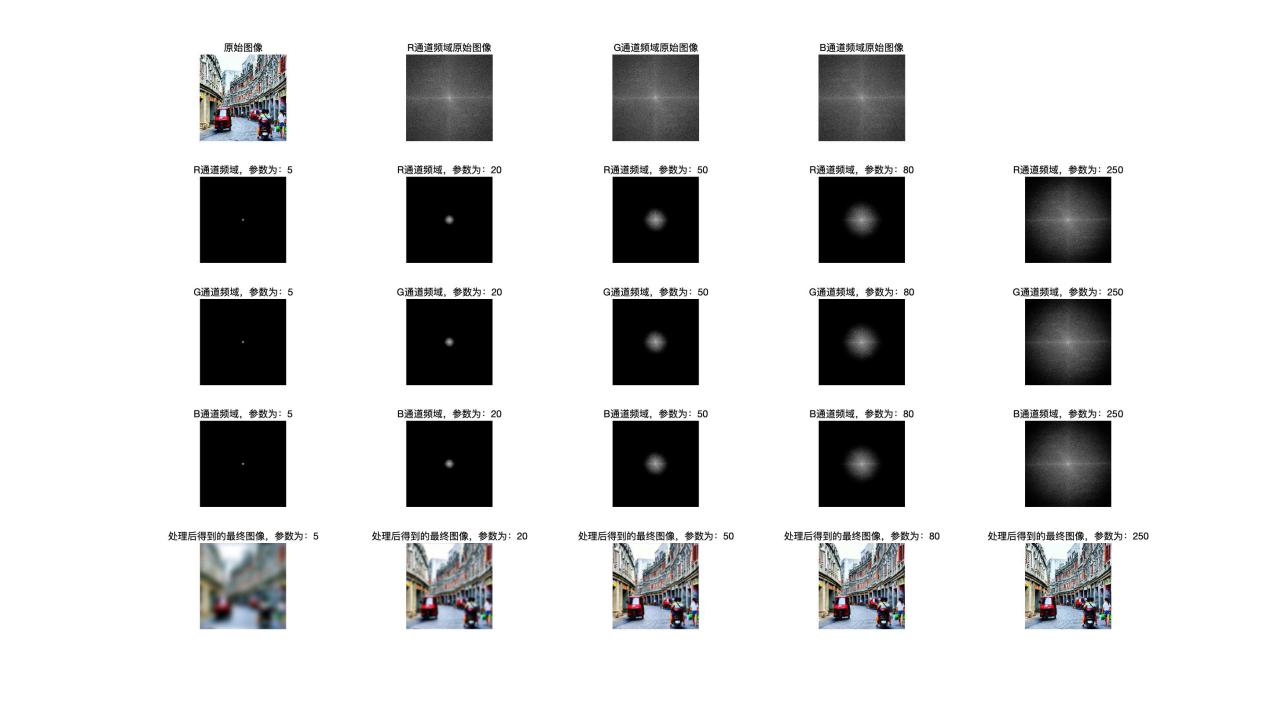
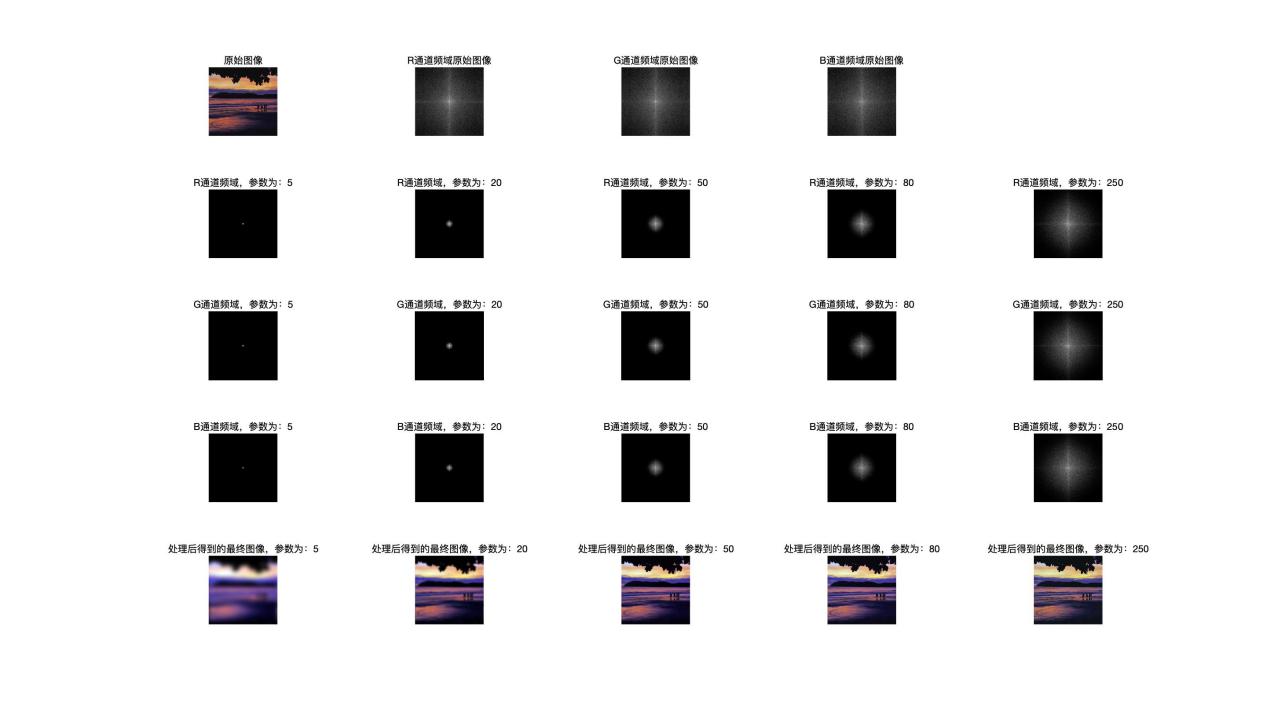
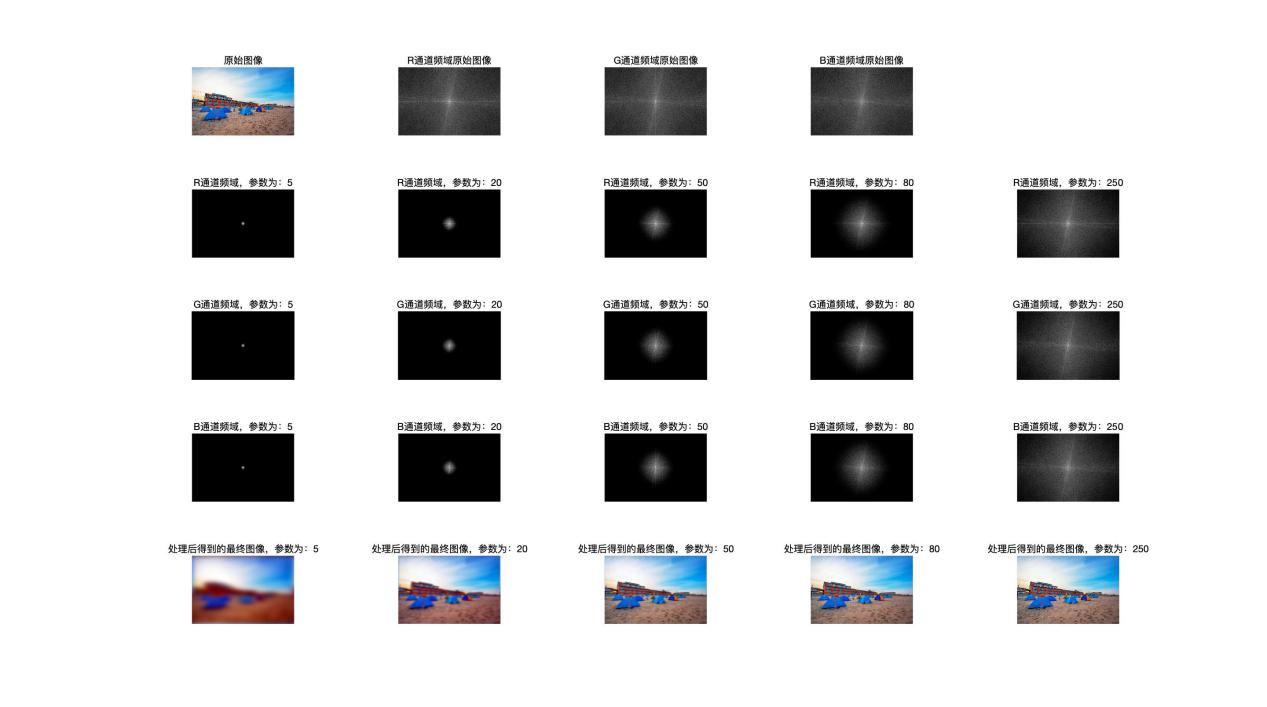
IMG_256

高斯算子

**实验效果展示及分析：**

****

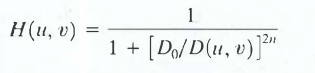
参数越小，频域滤波后所剩下的就越少，图片也就更加模糊，从数值上来看就越平滑，“中间点”失去细节。

****

问题2：

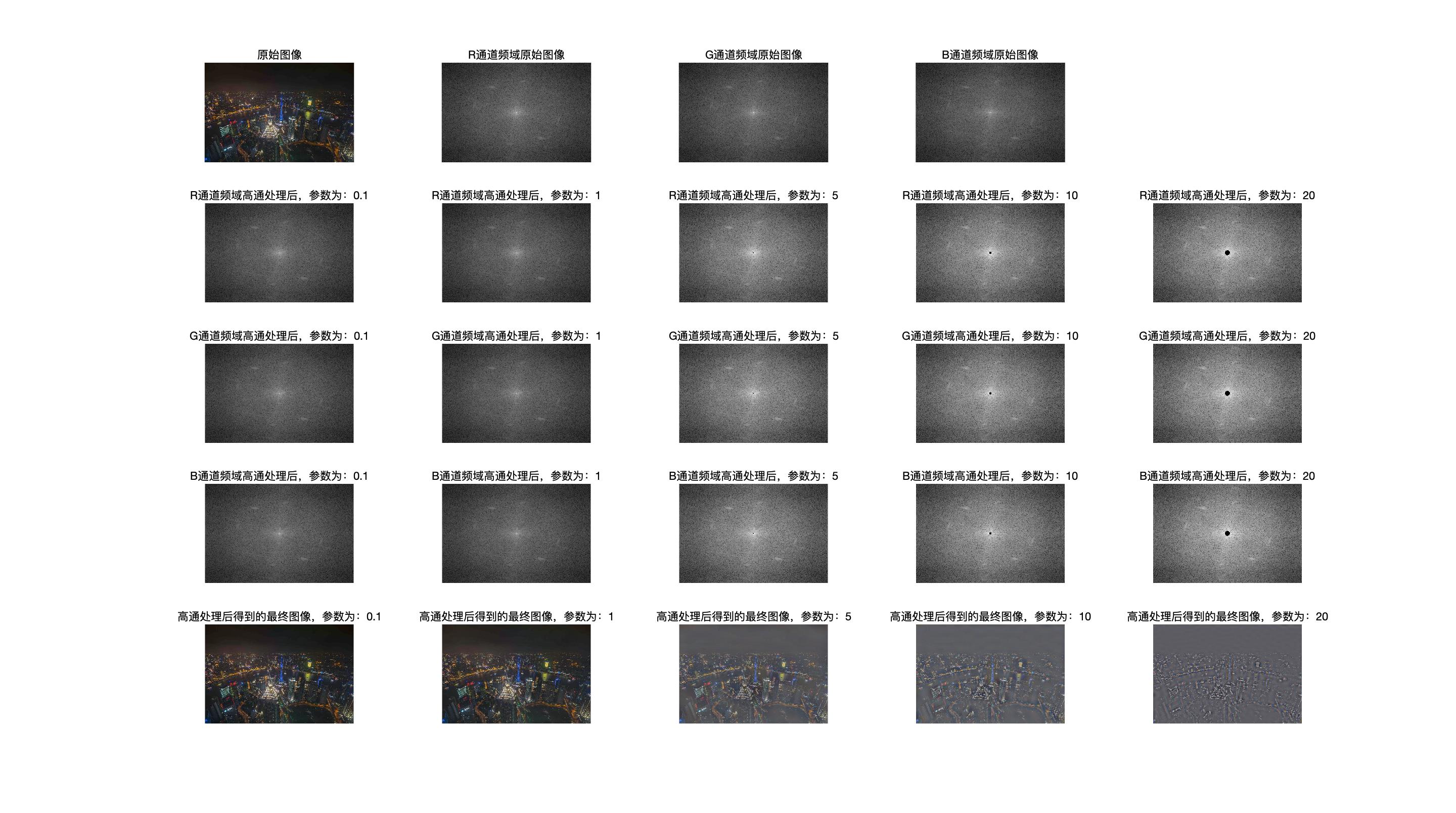
**算法原理：**

此处我选择的是Butterworth高通滤波，基本原理同问题1，只是在进行滤波操作时将算子从高斯算子改为了Butterworth高通算子。

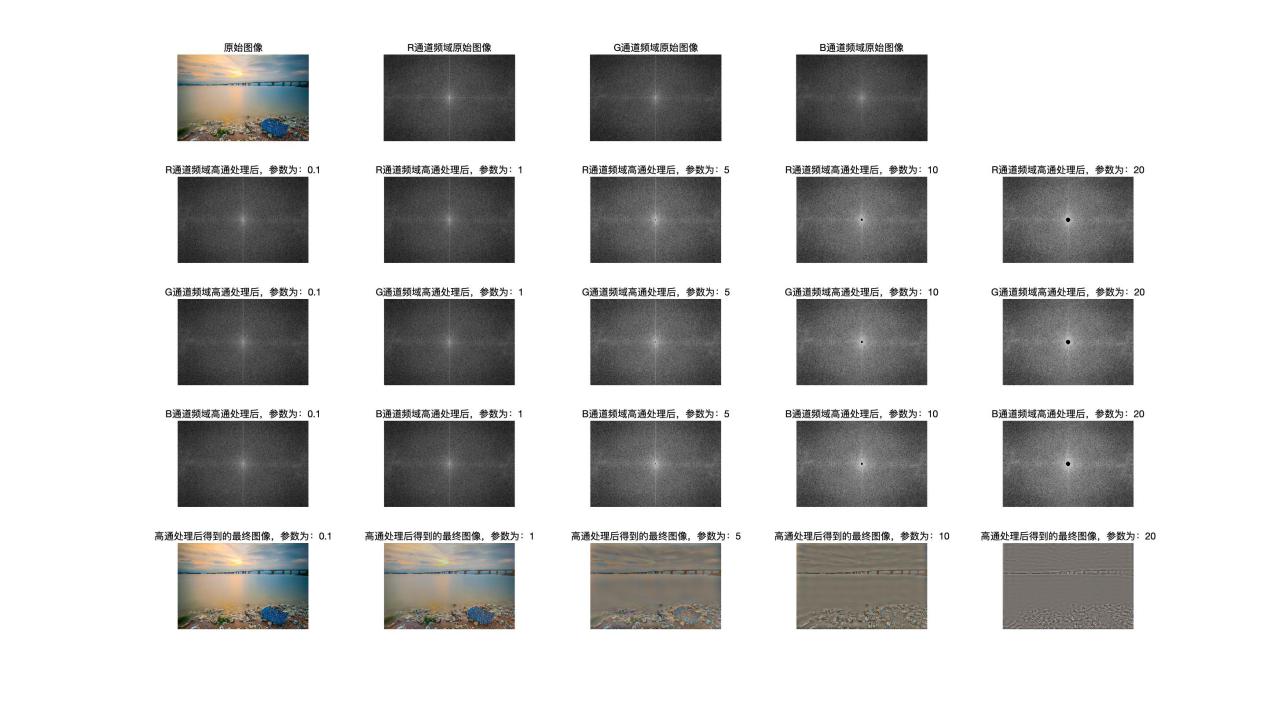
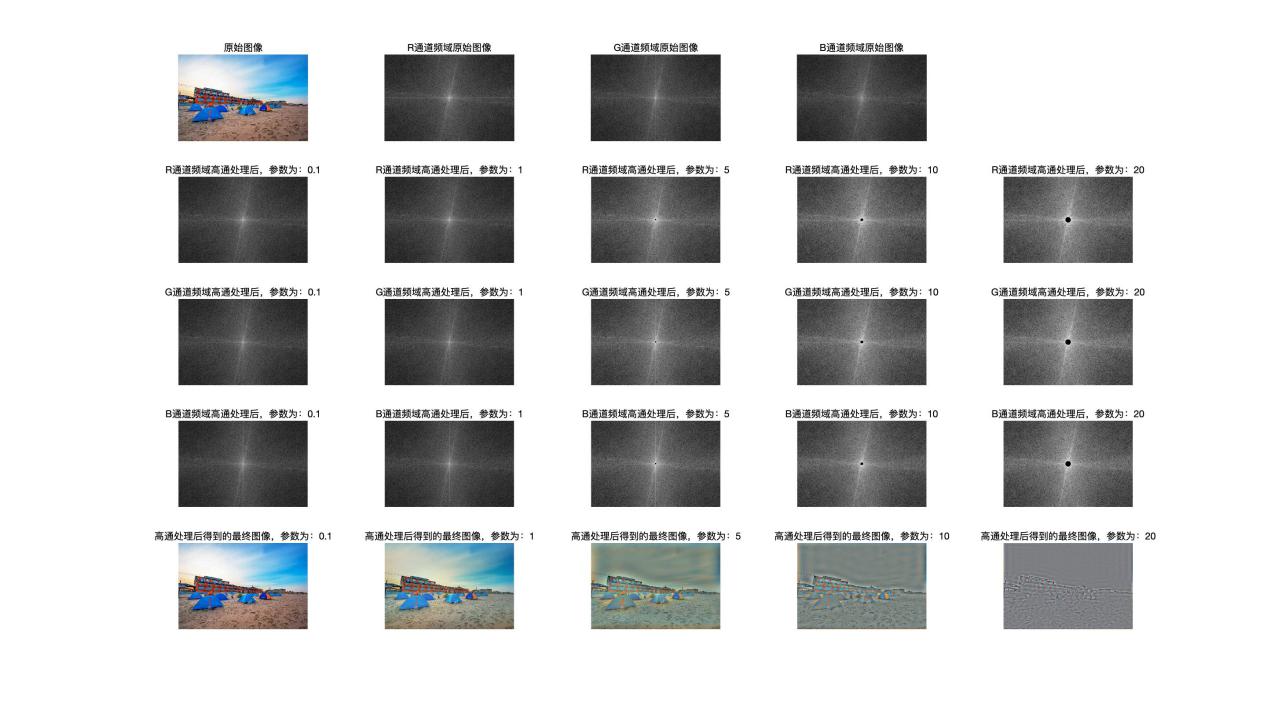
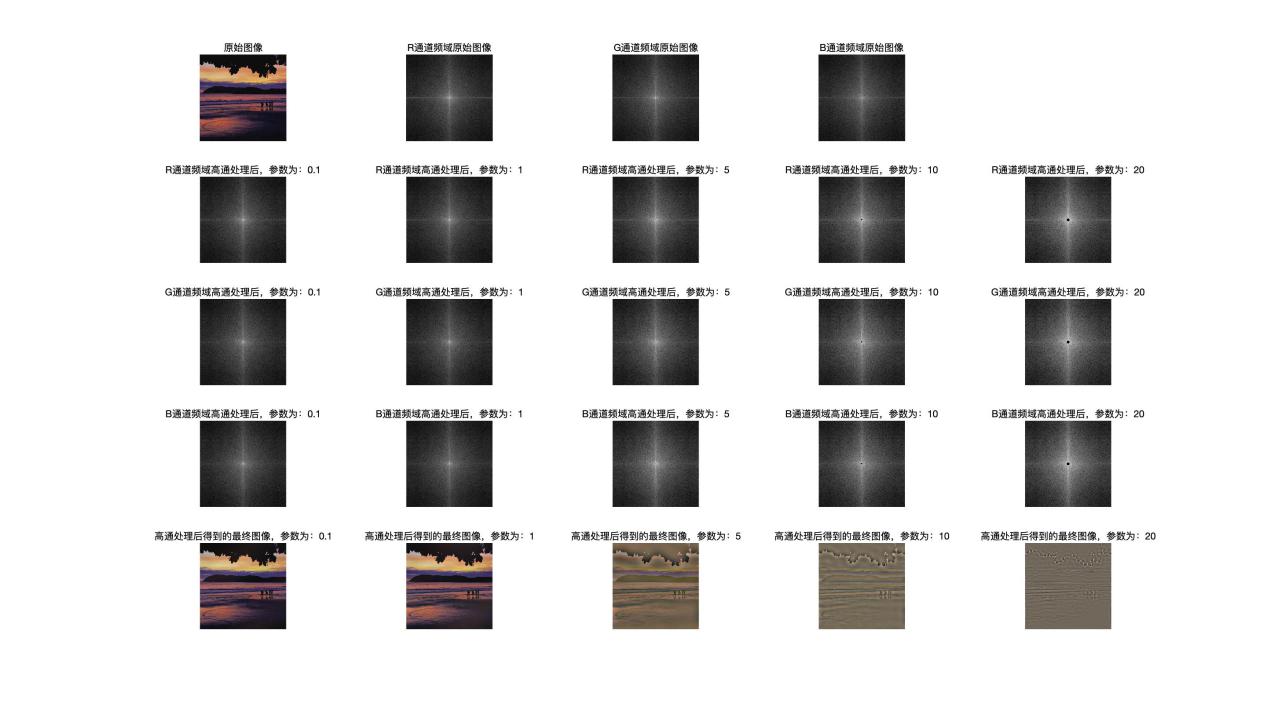
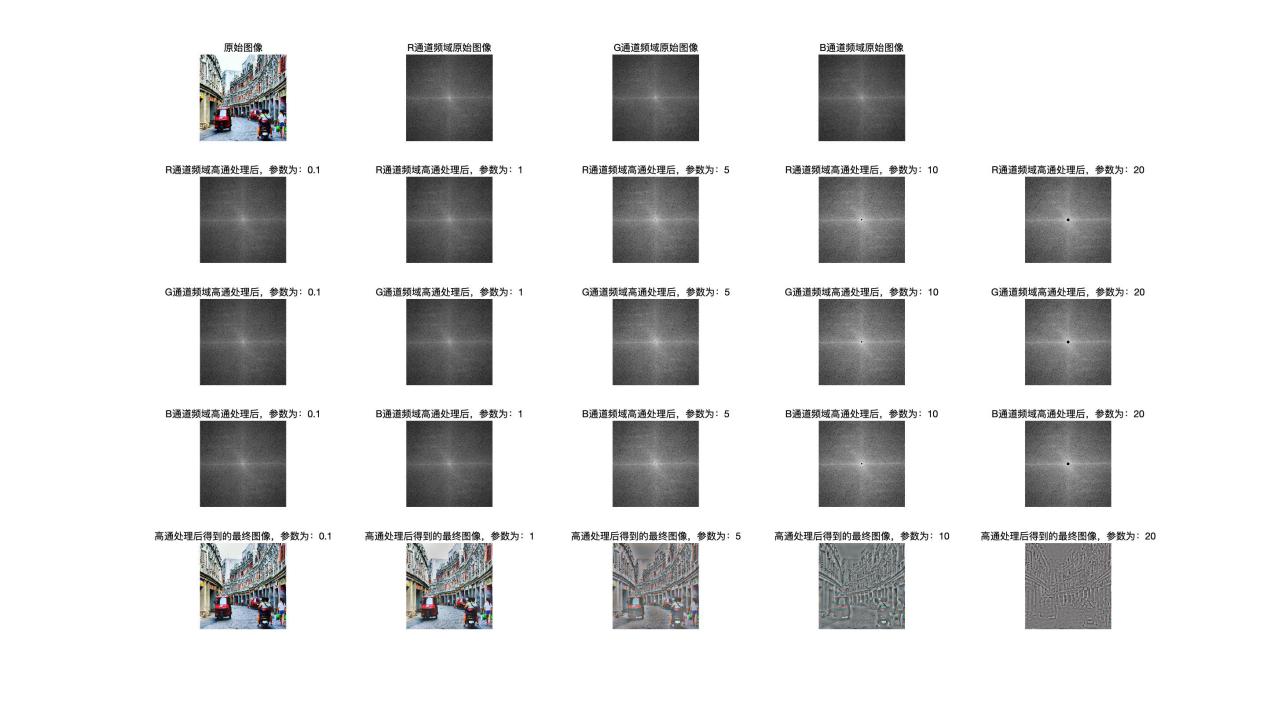


Butterworth高通算子

**结果展示及分析**：



高通的参数越大，图像内物体的边缘被模糊掉的就越严重，参数过大时图像会异常发灰，证明过多的边缘细节被抹除。



**作业二：灰度和彩色图像的离散余弦变换**

**算法原理：**

第一步：将彩色图像的RGB通道拆开，灰度图像跳过此步。

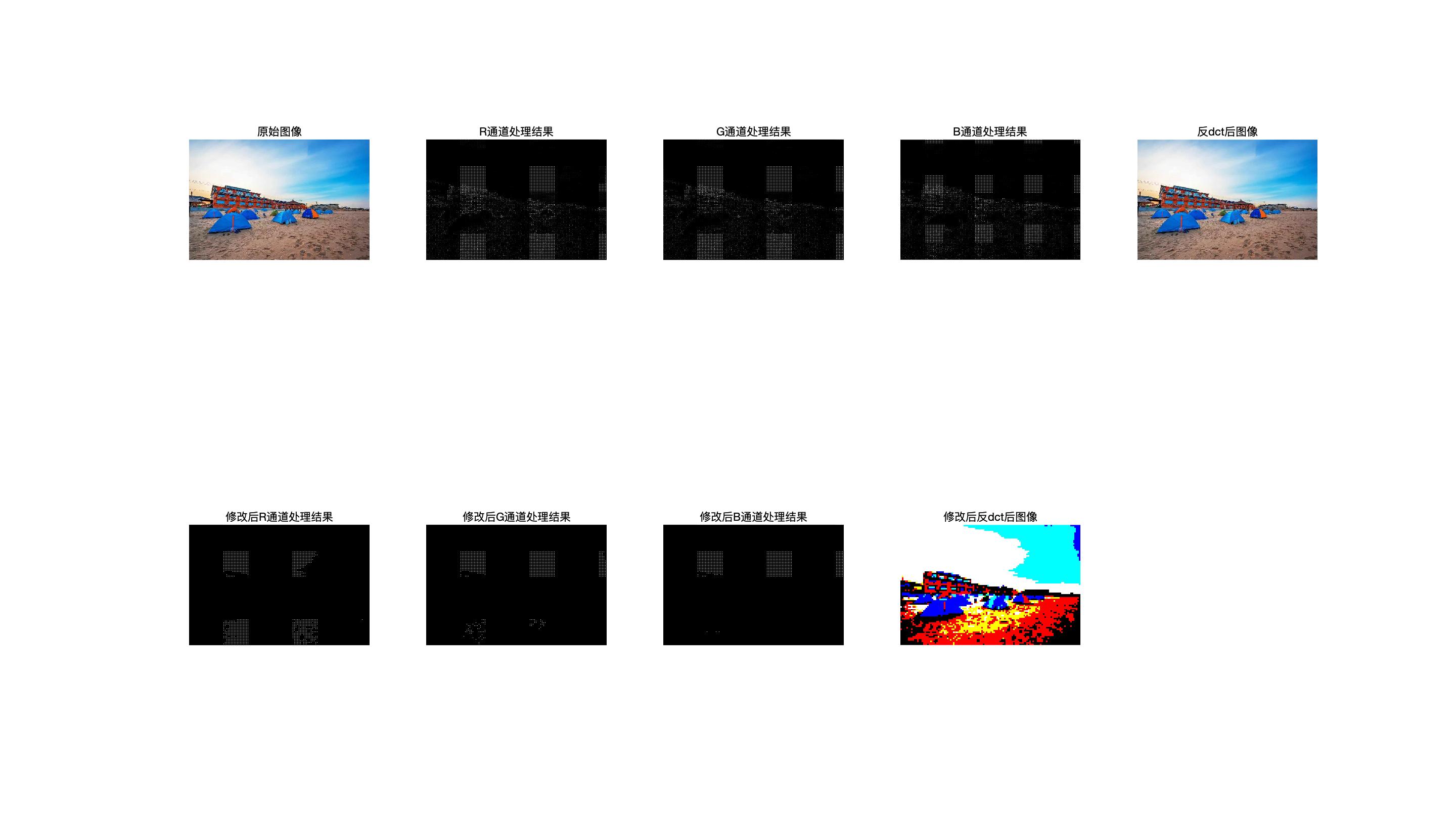
第二步：将三个通道的灰度值分别进行8x8的分割和处理\*，并再反DCT后合并获得一张测试图。

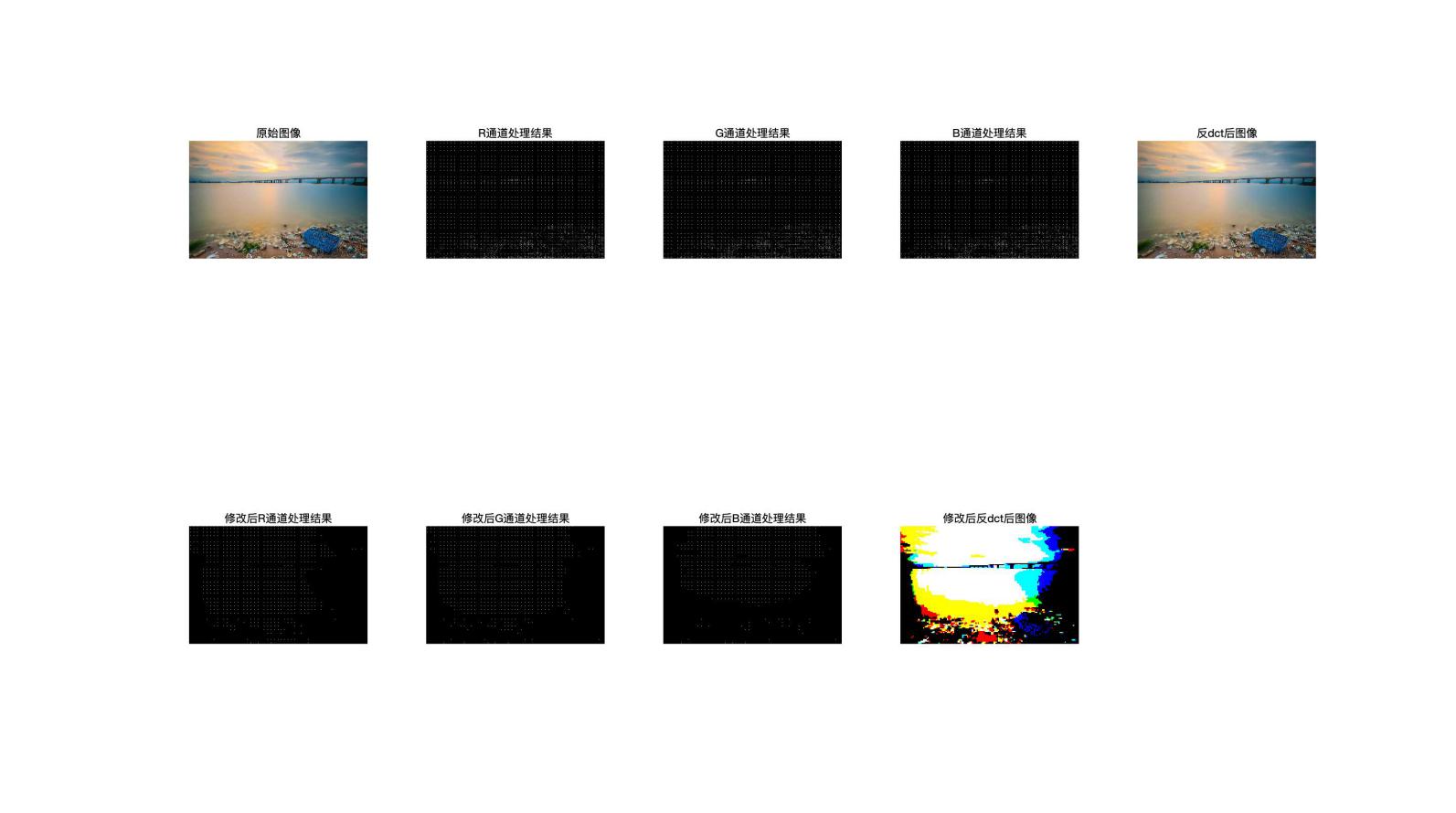
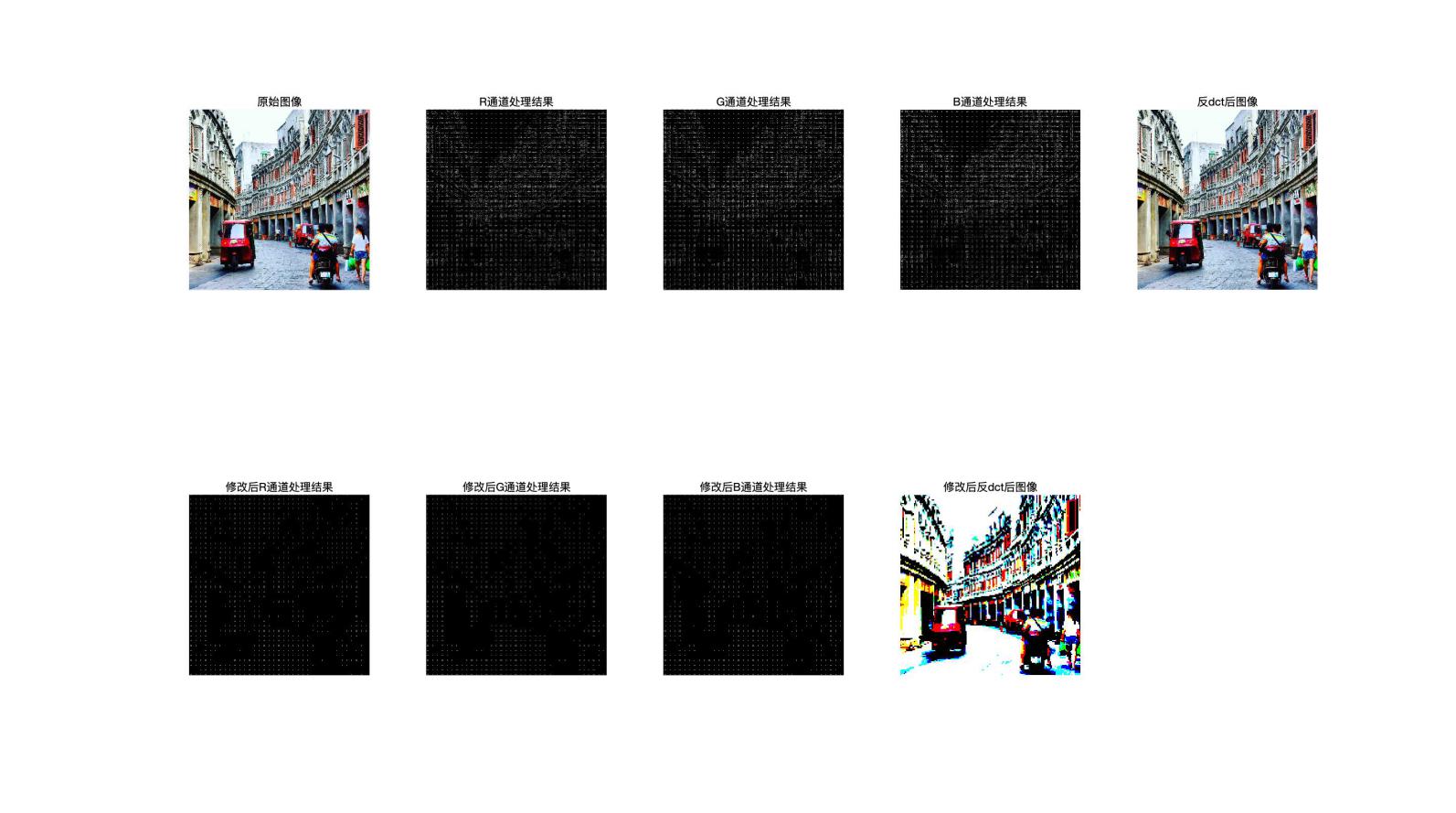
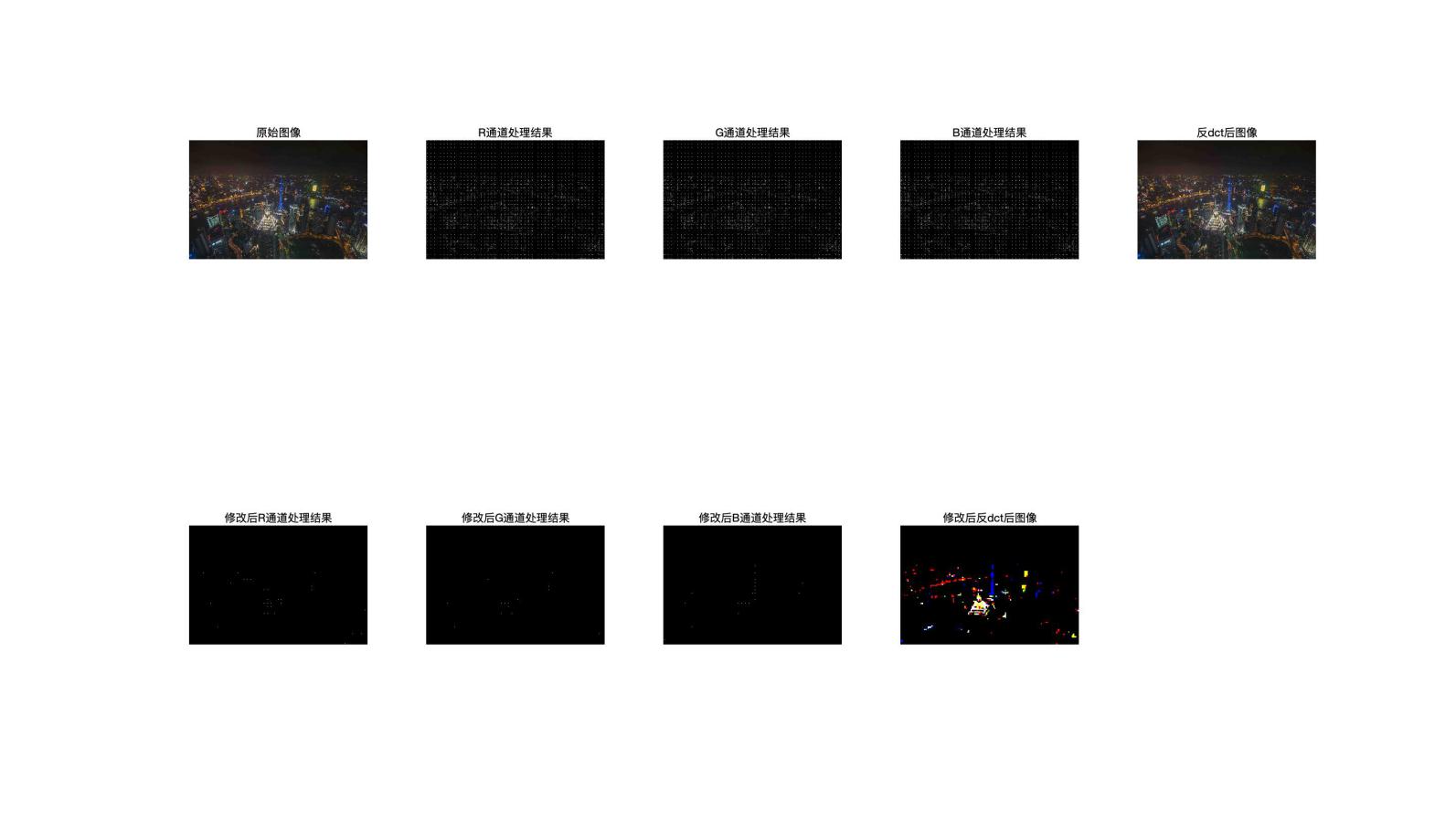
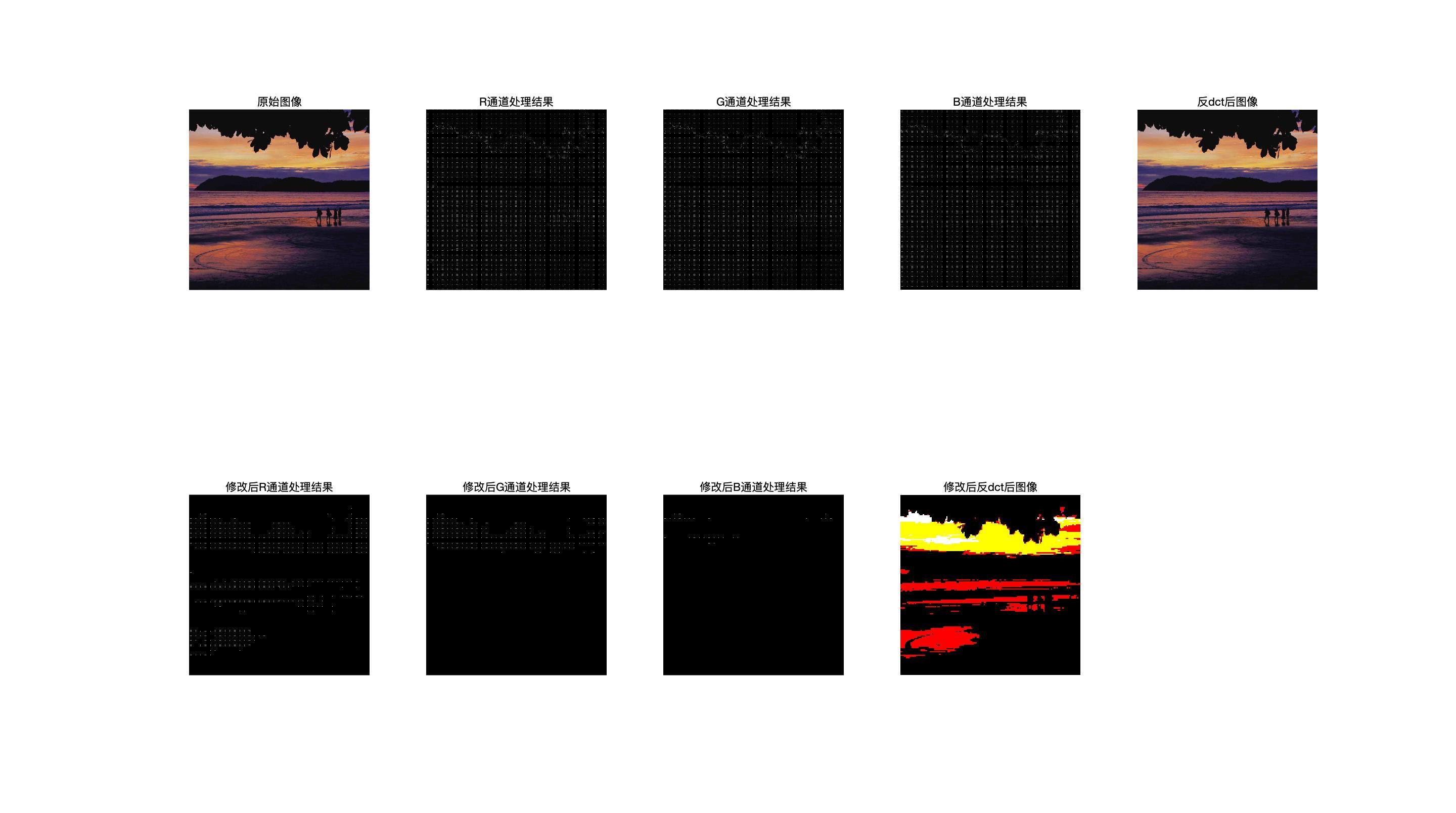
第三步：分别遍历三个通道经过DCT处理后的矩阵，将其中的值除以8取整之后再乘以8 。

第四步：将上一步处理得到的三个通道合并为新的图片。

\*：分割处理时可以使用blkproc函数进行处理，会更加便捷。

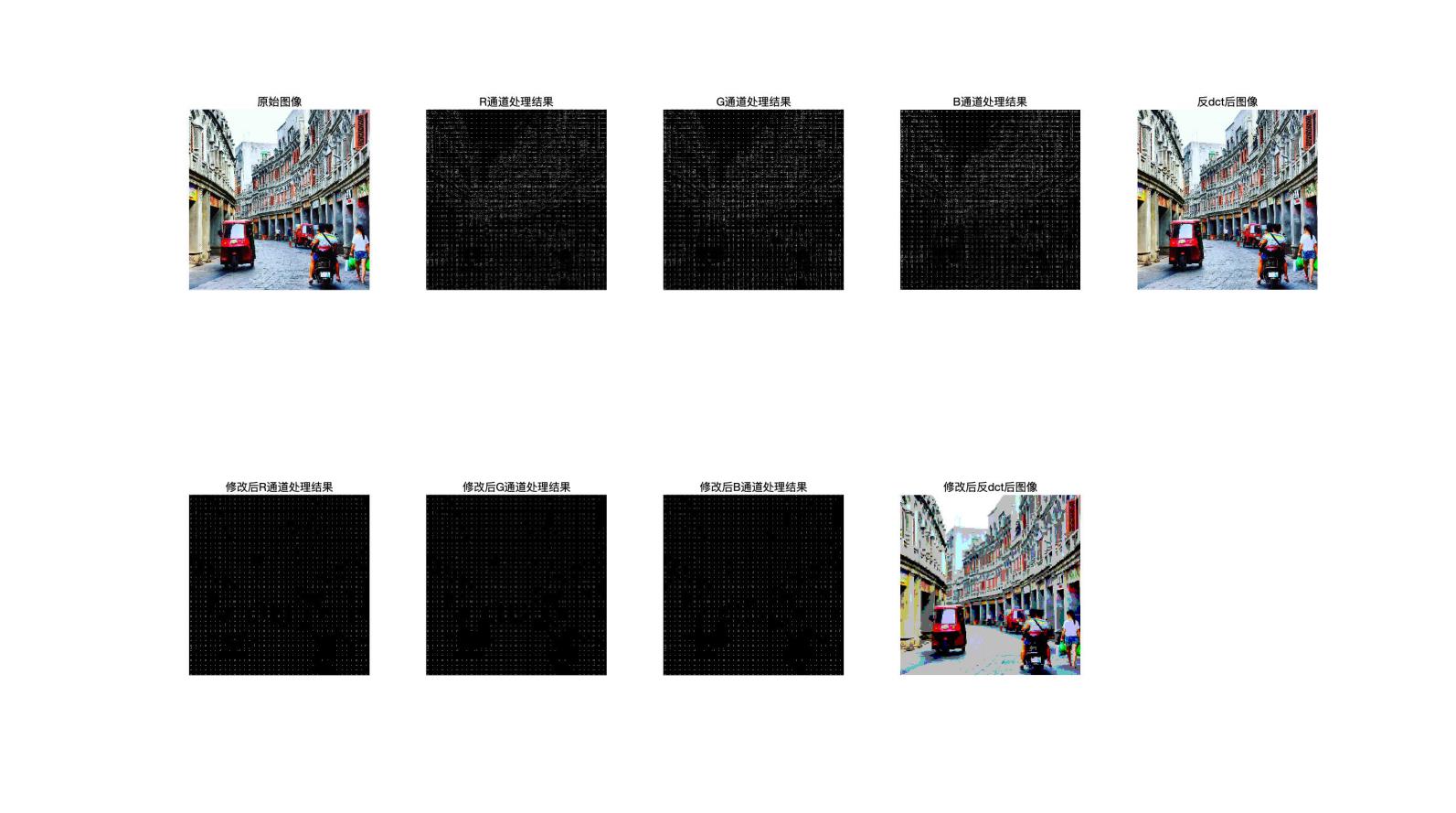
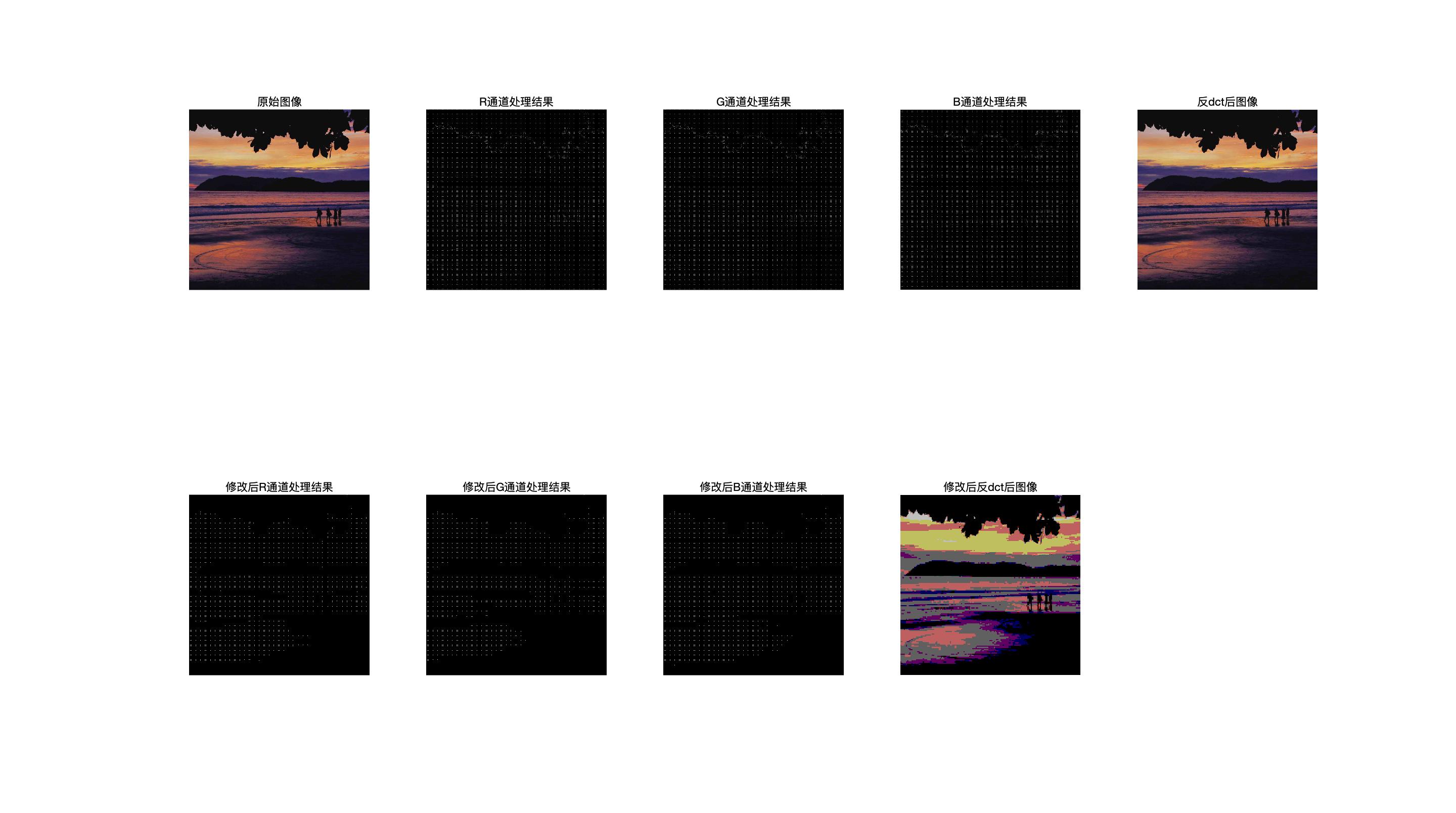
**结果展示及分析：**





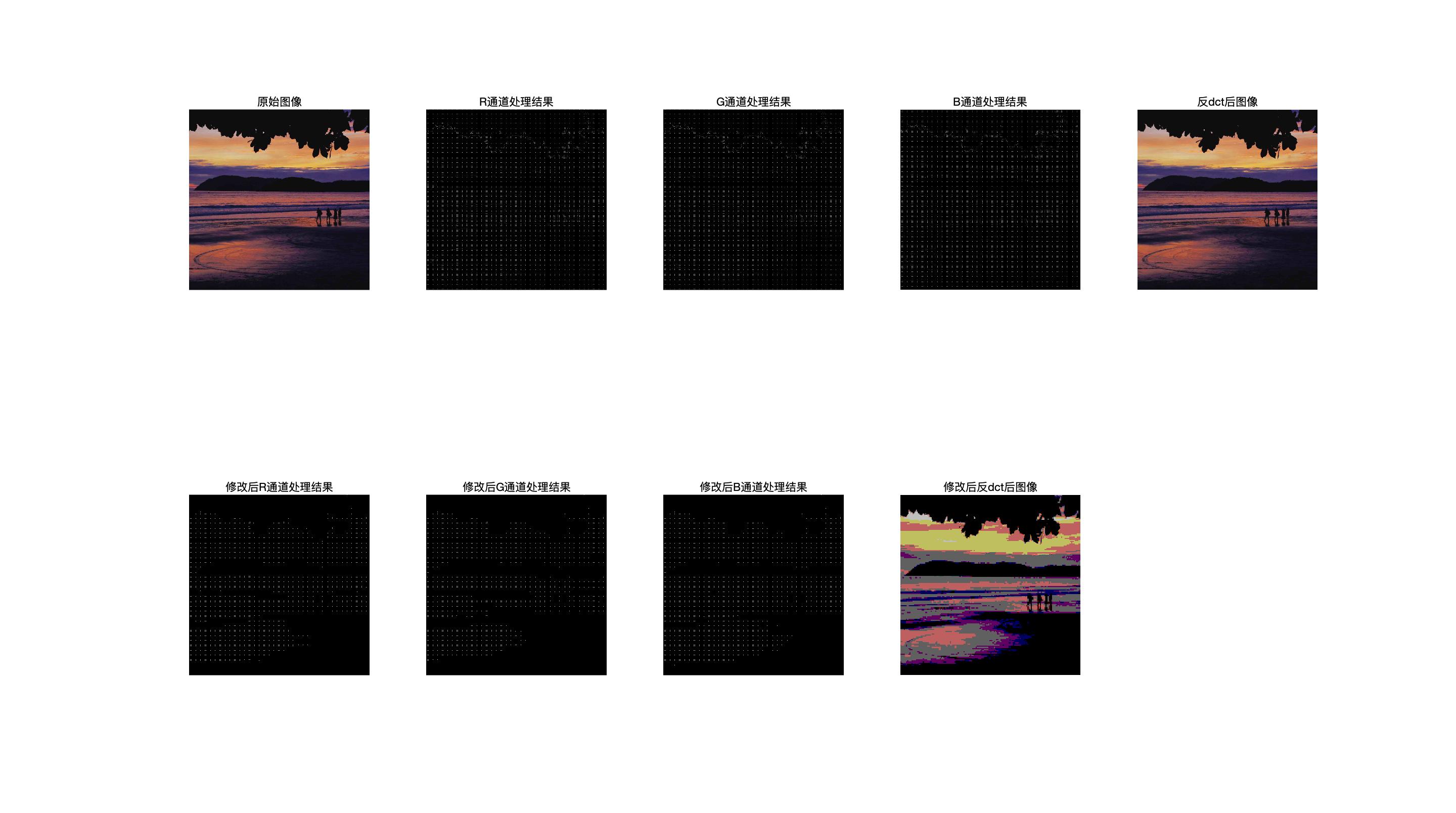
对DCT系数进行操作后生成的图片较原图失真，猜测可能是8过于大，尝试下调。

将8改为3

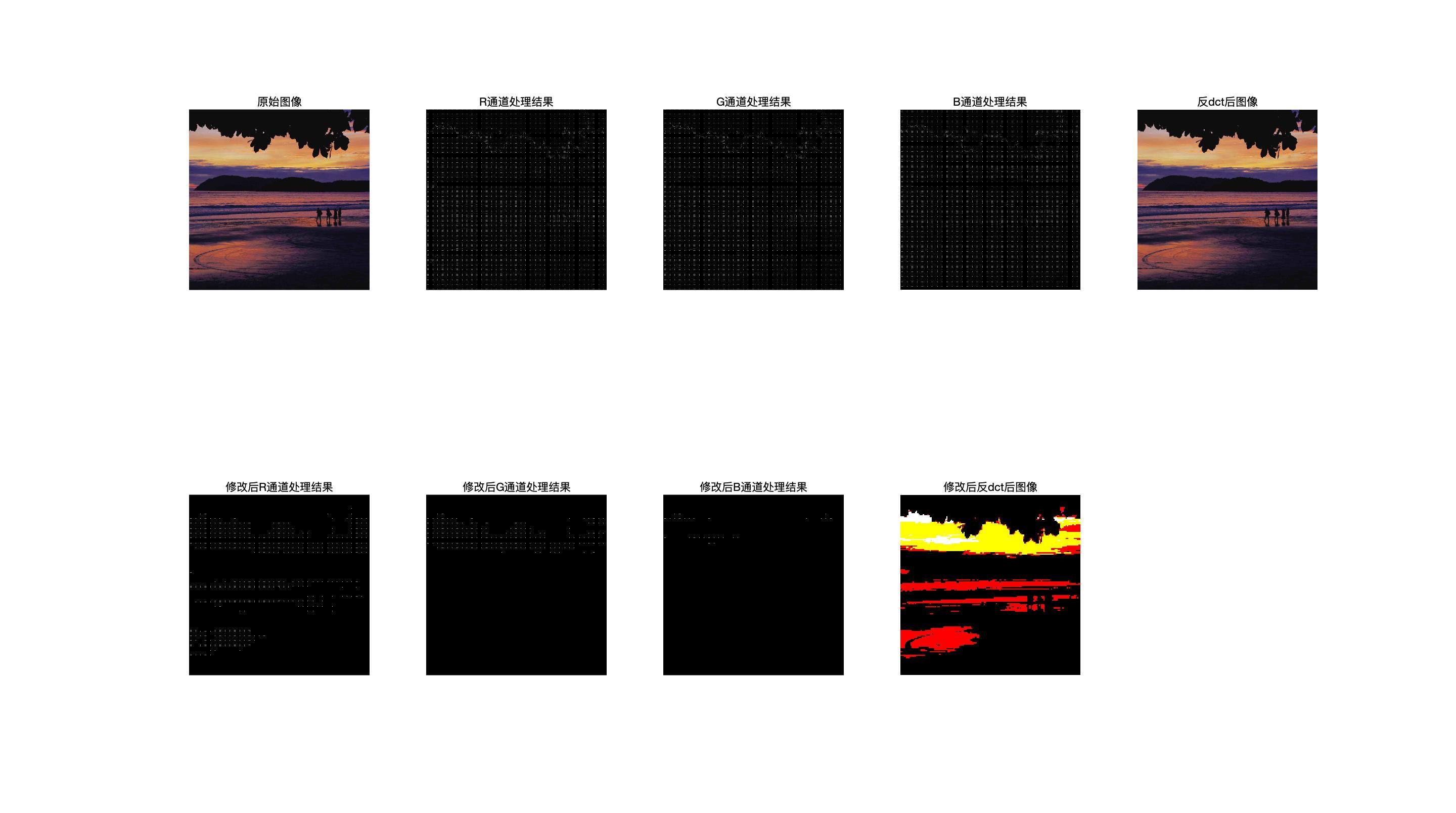


取这一组作为对比

DCT参数为3：



DCT参数为8：



参数为3时，最终图像的失真程度要比参数为8时小得多，证明上方的猜想是正确的。