

# 多媒体技术pj2

---

## Part1. 获取图片RGB值

---

描述：给定一张图片，获取点(x,y)，以及(x,y)周围8个点的R、G、B值

要求：允许用户指定图片的路径、坐标；做好边界检查

- 测试结果
  - satomi.jpg, (100,100)  
image:satomi.jpg  
x:100  
y:100  
(99, 99): (71, 113, 77)  
(99, 100): (73, 115, 79)  
(99, 101): (75, 117, 81)  
(100, 99): (73, 115, 79)  
(100, 100): (75, 117, 81)  
(100, 101): (77, 119, 83)  
(101, 99): (75, 117, 81)  
(101, 100): (77, 119, 83)  
(101, 101): (79, 121, 85)

## Part2. 灰度图像

---

描述：使用matlab实现rgb2gray的功能，并能计算出转换后灰度图片的方差。

要求：可以使用matlab自带图片，也可自选图片，自选图请放在项目当前目录；输出的文件保存到当前目录即可

- 灰度图像转换
  - 原图像为satomi.jpg，灰度图像储存为my\_gray\_satomi.jpg
  - 原图像和转换后的图像如下：

原图像



灰度图像



- 灰度图像的方差
  - 打印计算结果

```
>> test_2
```

```
variance of grayscale image: 1986.324780
```

## Part3. 灰度图对比度增强

描述：分别使用灰度拉伸和直方图均衡化，增强给定图的对比度。

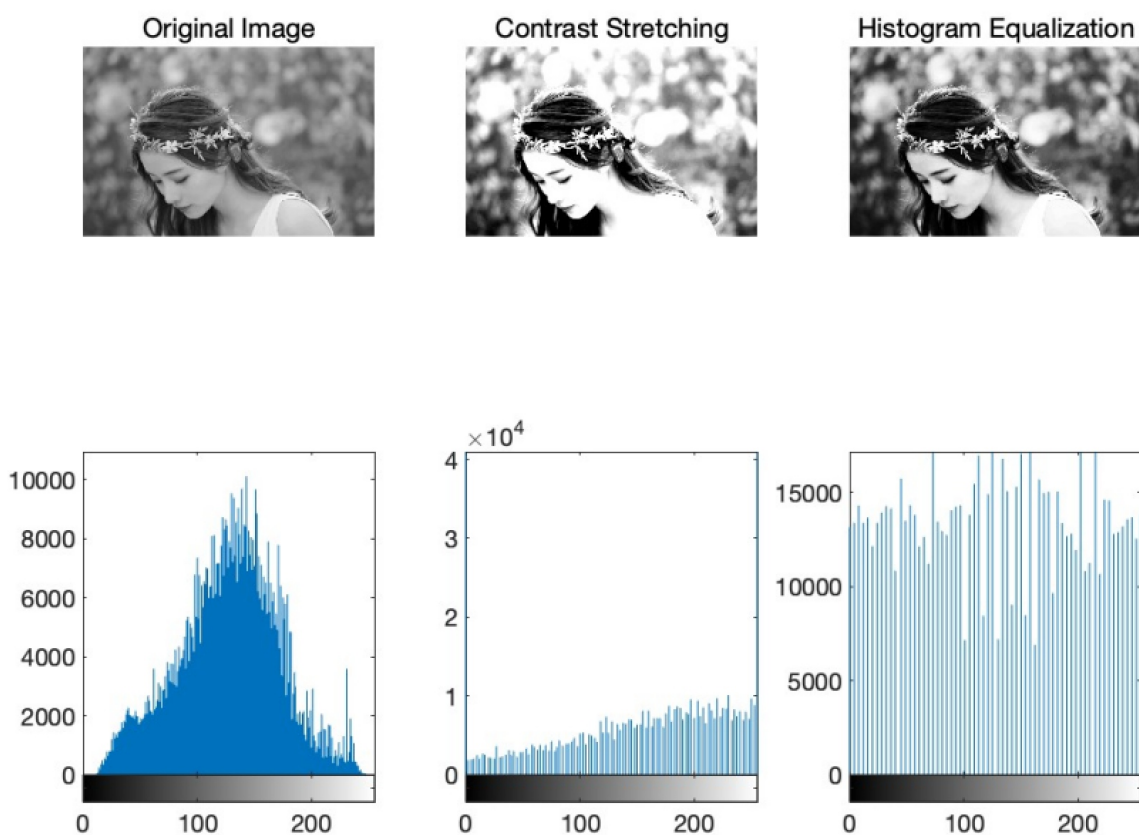
要求：需给出增强前后对比图，以及灰度值分布图。通过结果，谈谈你对灰度拉伸和直方图均衡化的理解。

- 灰度拉伸和直方图均衡化都是增强图像对比度的图像处理技术。
- 灰度拉伸
  - 灰度拉伸通过拉伸图像的灰度值范围，将原始图像中的灰度值映射到一个更广泛的范围内，从而增加图像中灰度值的动态范围，使图像中的细节更为突出。
  - 在test\_3.m中，将[0.2,0.6]之间的值映射到[0,1]之间，[0,0.2]之间的映射到0，[0.6,1]之间映射到1，因此，部分原图像中颜色相近的部分被填充为0/255的色块

```
J = imadjust(I, [0.2, 0.6], [0, 1]);
```

- 原图像的灰度值大多分布在50-200，导致图像对比度相对较低；而进行灰度拉伸后，灰度值更均匀地分布在0-255之间，直方图不再有明显的“峰值”，使得原图像中较亮和较暗的区域在新图像中更加明显，增强了图像的对比度
- 直方图均衡化
  - 直方图均衡化通过累积分布函数（CDF，Cumulative Distribution Function）的变换，重新分布图像的灰度级别，将原始图像中的像素灰度值映射到一个新的灰度级别，使得整个灰度范围内的像素值得到更加均匀的分布。这有助于拉伸图像的动态范围，使细节更为突出，从而改善图像的视觉质量。

- 进行均衡化后，图像的灰度范围在0-255之间相对均匀地分布，使得图像的细节更明显
- 直方图均衡化不需要调整参数，会根据输入图像的实际分布进行映射，因此具有良好的适应性
- 但是，由于直方图均衡化需要计算累计分布函数和进行灰度映射，计算复杂度相对较高
- 从下面的例子中看
  - 灰度映射得到的图像对比度更强，使部分细节更加突出，但由于偏亮和偏暗的像素全部被处理成0和255，也丢失了一部分细节
  - 直方图均衡化得到的图像则保留了相对更多的细节
  - 仅从本变换对比来看，直方图均衡化的效果相对好些



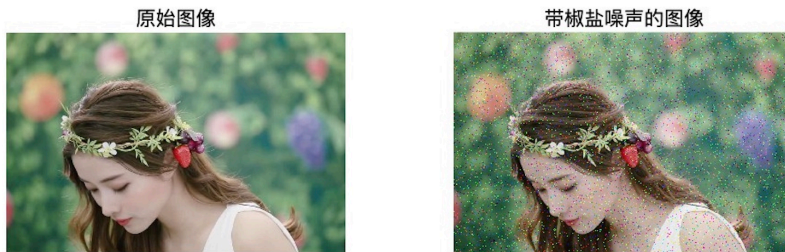
## Part4. 图像去噪

描述：对给定图片加椒盐噪声，并至少选择两个你了解的去噪算法去除椒盐噪声

要求：给出去噪后的结果对比图，通过算法分析造成这种差异的原因

- 加噪声

- 椒盐噪声
  - 图像中随机分布的黑白像素点，就像椒盐一样
  - 椒盐噪声在图像中呈现出随机的、不规则的分布，使图像中的一些像素值变成最低或最高灰度值
  - 噪声点通常是孤立的，不与周围像素的灰度值相关联
- 添加密度为0.04的椒盐噪声



- 去噪

- 中值滤波去噪
  - 原理：对目标像素及周边像素取中值后再填充目标像素
  - 适用于非常强烈的噪声情况，因为异常值对中值影响相对较小，对于椒盐噪声有良好的去噪效果，能够有效地去除异常值；下图去噪后基本看不出来椒盐颗粒
  - 但在去噪的同时可能引入一些模糊，下图去噪后明显比原图对比度下降；此外，对于一些其他类型的噪声，中值滤波可能不如其他滤波方法效果好



- 均值滤波去噪
  - 原理：对目标像素及周边像素取均值后再填充目标像素，通过平滑图像来消除噪声。
  - 优点：计算简单，适用于轻度噪声和快速去噪；且在一些平滑场景中效果良好。
  - 缺点：但是在偏强烈的噪声场景（如椒盐噪声）中，噪声对均值影响较大，可能导致图像模糊，且对细节的保留不如中值滤波。下图去噪后仍然可以看到较为明显的“颗粒”，效果不理想



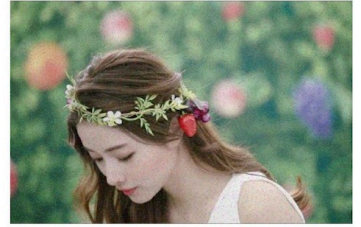
原始图像



加椒盐噪声的图像



均值滤波去噪后的图像



#### ○ 高斯滤波去噪

- 原理：应用高斯核函数对图像进行卷积，从而实现对图像中各个像素值的加权平均，通过权衡周围像素的贡献，减少噪声对图像的影响。高斯滤波更注重平滑，而不是直接去除极端值。
- 优点：能够保留图像的整体细节，适用于轻度噪声和需要平滑图像的情况。
- 缺点：但对于椒盐噪声等强烈噪声的去除效果相对较差，且对于边缘细节的保留不如其他算法；下图去噪后仍然可以看到较为明显的椒盐颗粒

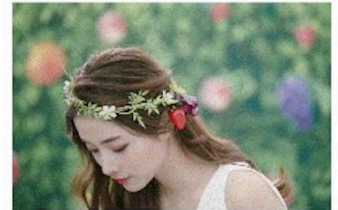
原始图像



带椒盐噪声的图像



高斯滤波去噪后的图像



- 总之，由于椒盐噪声是孤立的，并且异常值对均值影响较大，并且均值滤波和高斯滤波都更注重平滑而不是直接去除极端值（比如椒盐噪声），因此对于上述三种去噪算法，中值去噪的效果相对更好
- 因此选择去噪算法时需要先分析噪声的特点，选择更合适的方法
- 此外，也需要明白，即便使用去噪算法，也很难达到原图像的清晰度