数字图像处理作业报告六

学号：71194506019 姓名：姜志刚 专业：计算机技术

题目

提取一副彩色图像中红色，用HIS模型处理。

RGB2HSI

公式：
求Theta

$$\theta = \cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{1/2}}\right\}$$

根据b g 大小确定H：

$$H = \begin{cases} \theta & \mathrm{B} <= \mathrm{G} \\ 360 - \theta & \mathrm{B} > \mathrm{G} \end{cases}$$

$$s = 1 - \frac{3}{(R+G+B)}[min(R,G,B)]$$

$$I = \frac{1}{3}(R+G+B)$$

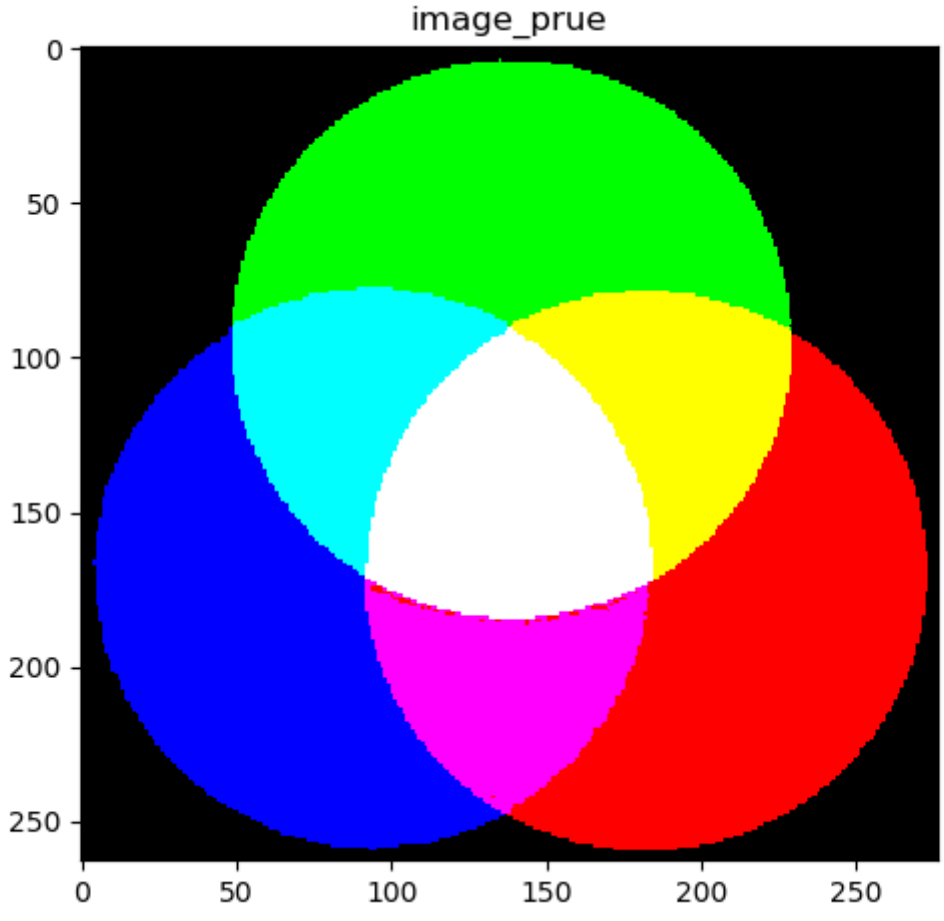
下面函数实现RGB转HSI模型，并对$350 \leq H \leq 360$ 且 $0 \leq H \leq 10$保存，以达到提取红色的目的：

```
def bgr_2_hsi(image):
    """
    :param image: RGB model image
    :return: HSI model image and slicing image in H
    """
    out = np.copy(image)
    out_slicing = np.zeros(image.shape, np.uint8)

    for x in range(image.shape[0]):
        # print(str(int(x / image.shape[0] * 100)) + "%")
        for y in range(image.shape[1]):
            b, g, r = image[x][y]
            b, g, r = int(b), int(g), int(r)
            i_s = np.sum([b, g, r])
            i = i_s / 3

            # i == 0, s and h is no sense
            if i_s == 0:
                i = 0
                s = 0
                h = 0
                out[x][y] = h, s, i
                continue

            s = (1 - (3 * np.min([b, g, r])) / i_s) * 255

            # s == 0 h is no sense
            if s == 0:
```

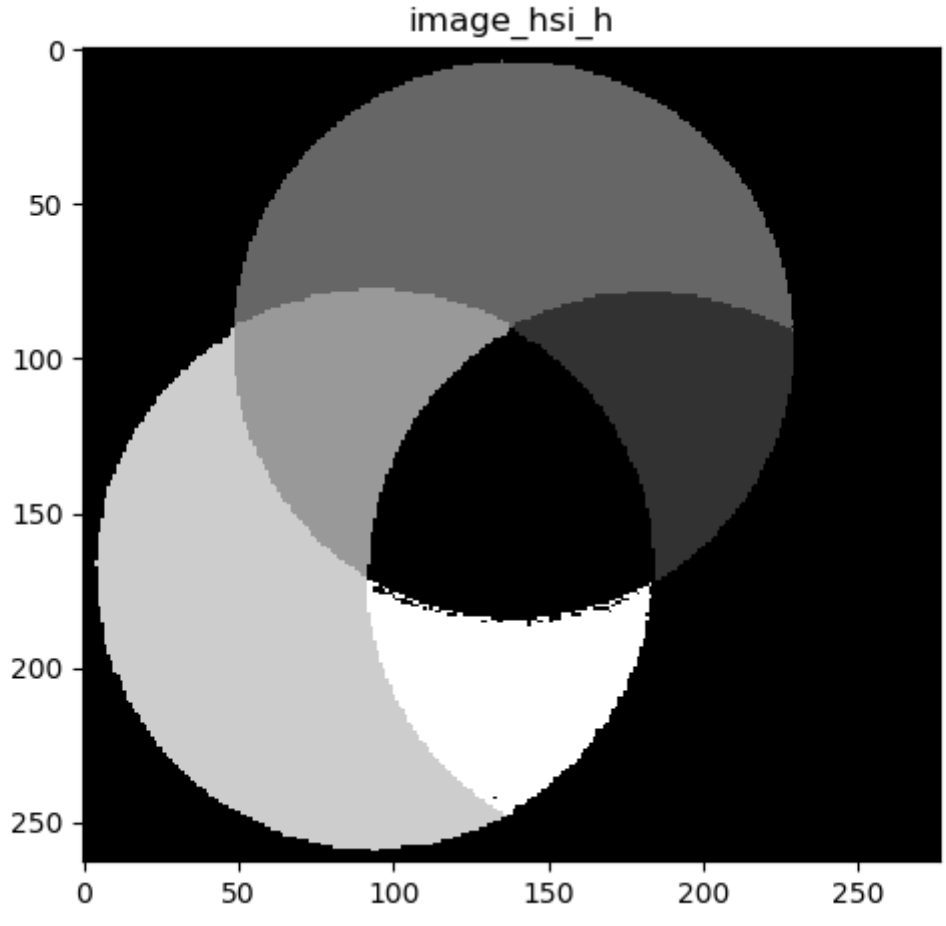
```
                h = 0
                out[x][y] = h, s, i
                continue

            thea = np.arccos((2 * r - g - b) / (2 * np.sqrt((r - g) ** 2 + (r - b) * (g - b))))
            if g >= b:
                h1 = thea
            else:
                h1 = np.pi * 2 - thea
            h1 = np.rad2deg(h1)
            # slicing
            if (int(h1) in range(0, 11) or int(h1) in range(350, 361) ) and s/255 > 0.1:
                print(int(h1))
                out_slicing[x][y] = image[x][y]

            h = h1 / 360 * 255
            out[x][y] = h, s, i

    return out, out_slicing
```

结果

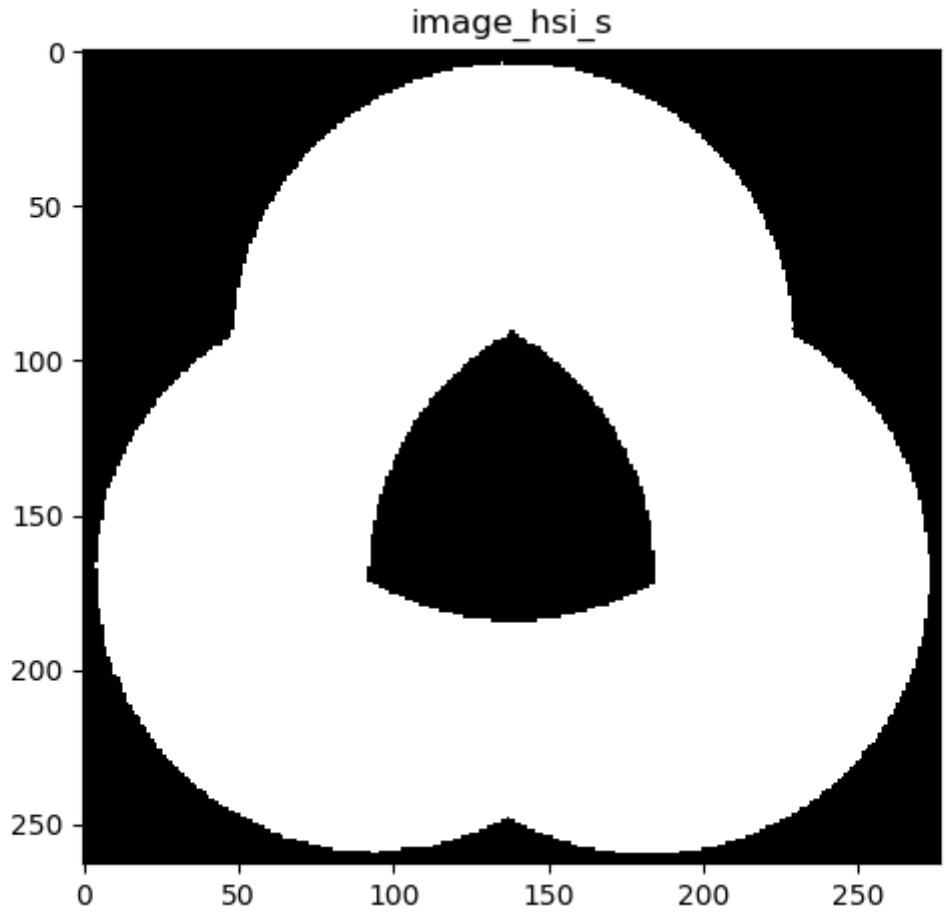
原图：



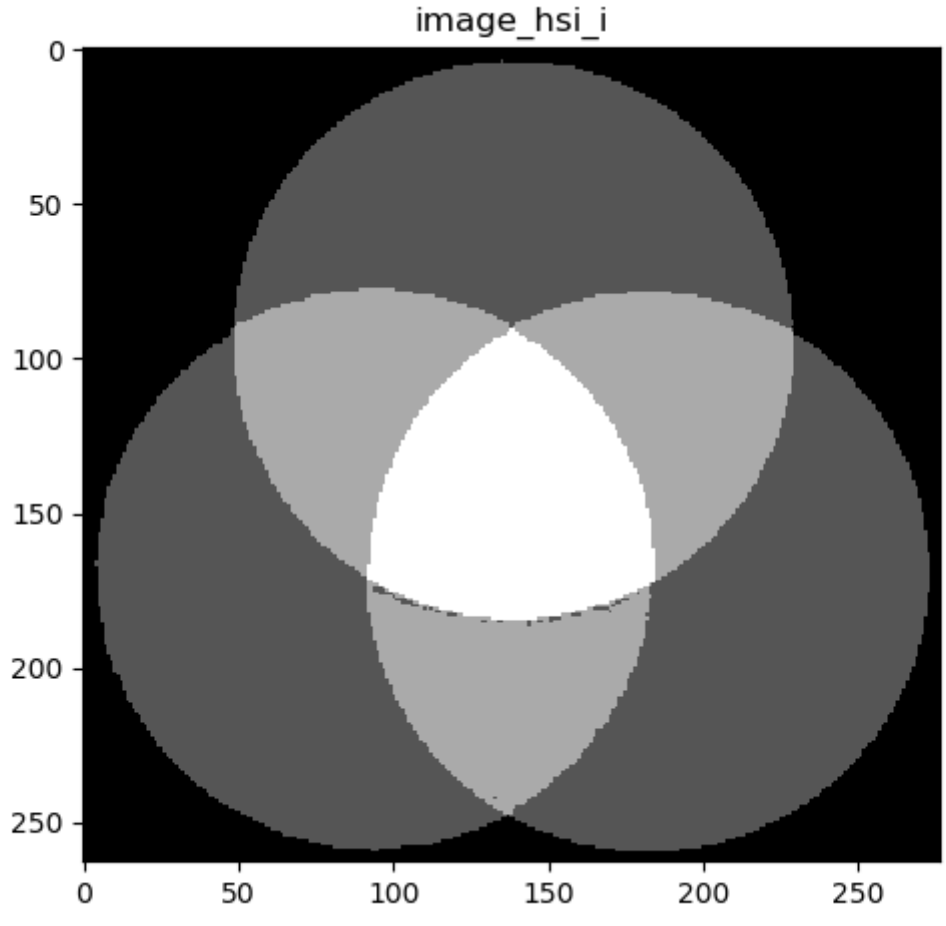
HSI模型中色调H分量：



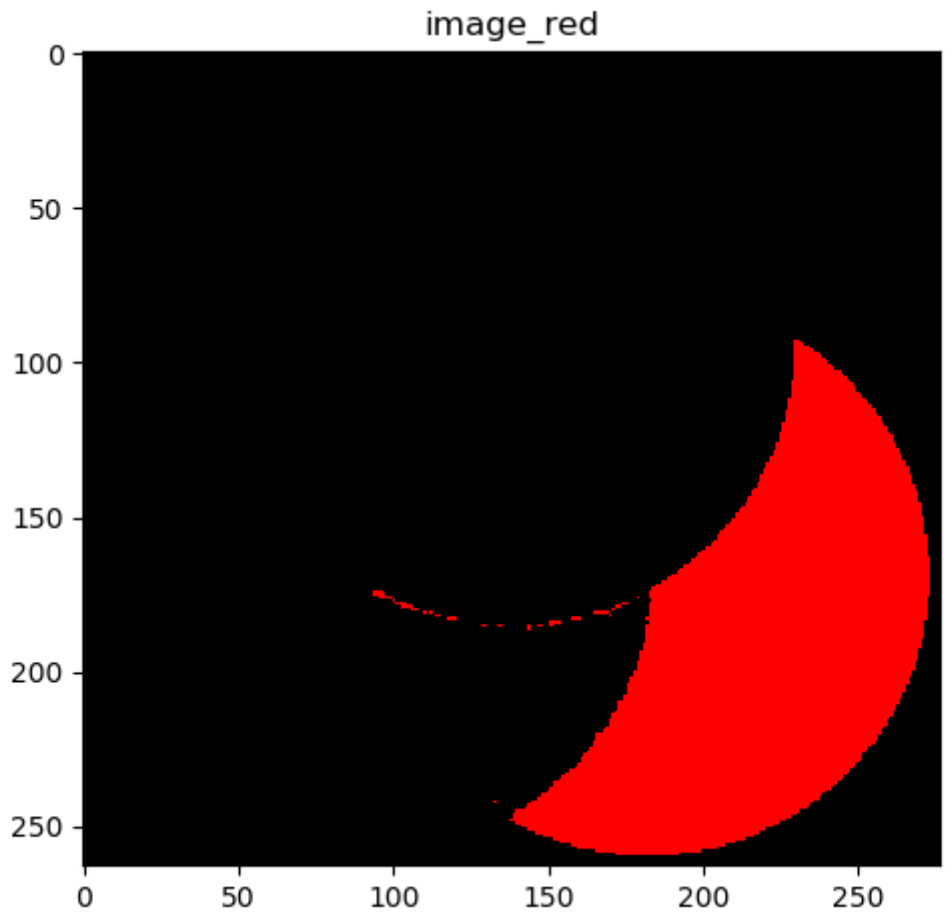
HSI模型中饱和度S分量：



HSI模型中亮度I分量：



抽取的结果：



有噪声是因为图片质量问题

补充1

RGB模型进行分层

```
def color_slicing(image, center, w):
    """
    :param image:
    :param center: b, g, r ib range 0 ~ 255
    :param w: width
    :return:
    """
    out = np.zeros(image.shape, np.uint8)
    for x in range(image.shape[0]):
        for y in range(image.shape[1]):
            r_b, r_g, r_r = center
            a_b, a_g, a_r = image[x][y]
```

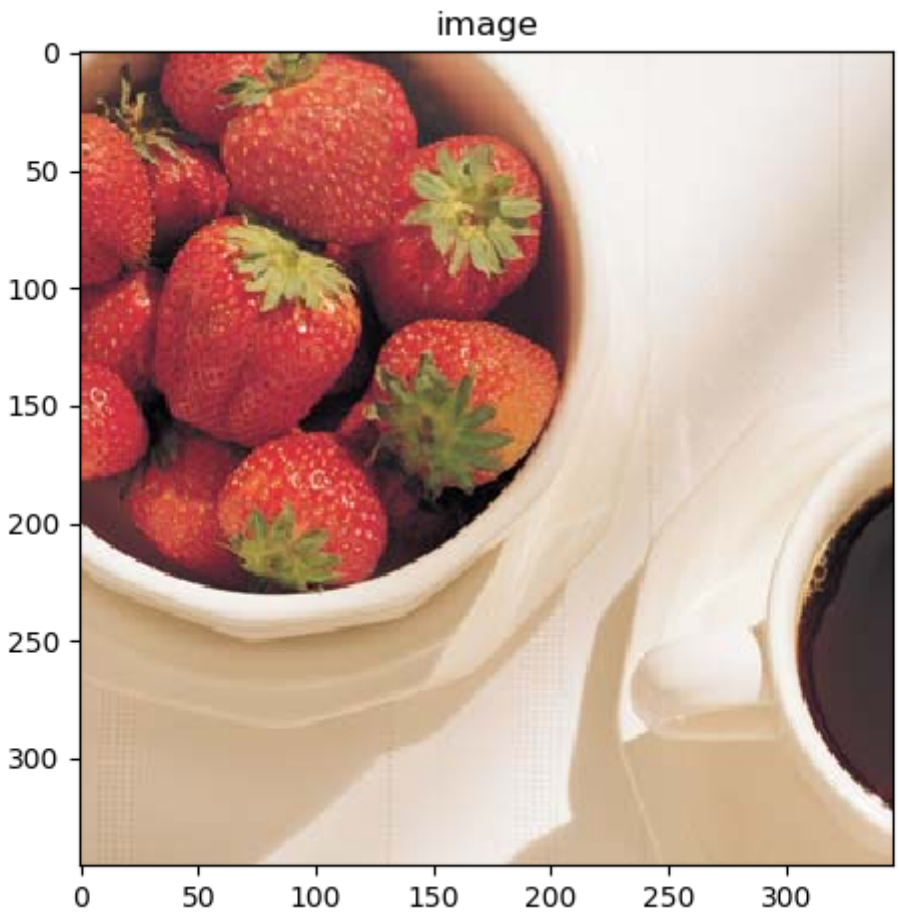
```
            if abs(r_b - a_b) < w/2 and abs(r_g - a_g) < w/2 and abs(r_r - a_r) < w/2:
                out[x][y] = image[x][y]
    return out
```

调用：

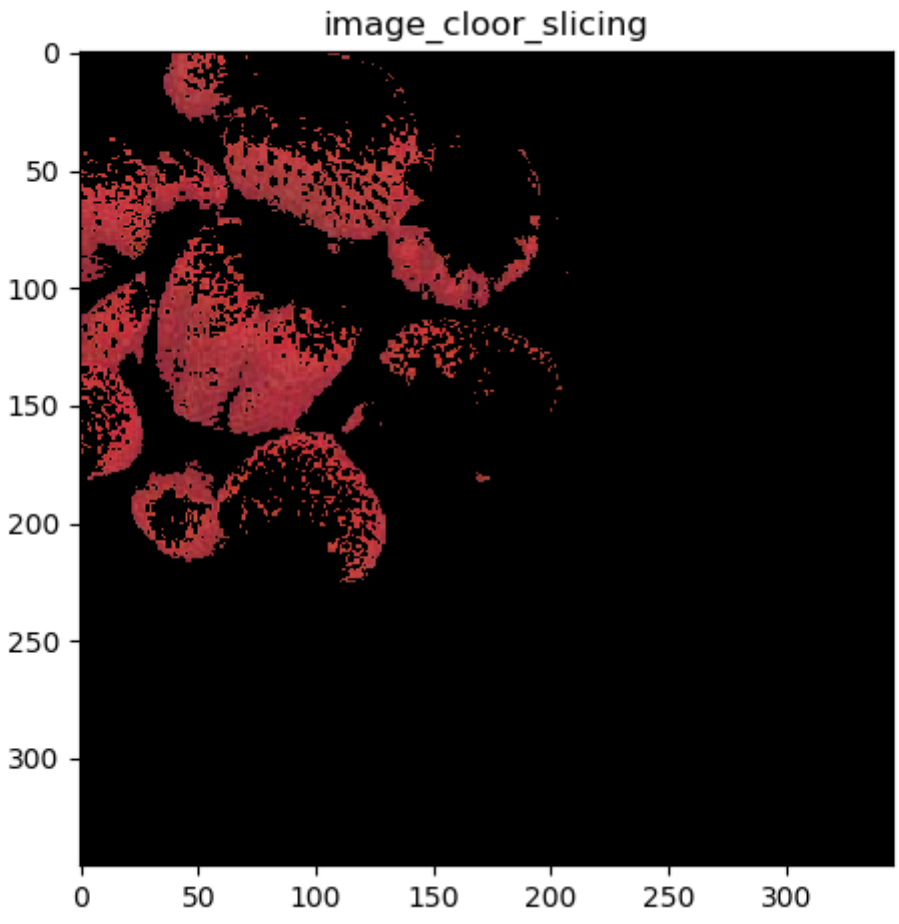
```
image_cloor_slicing = color_slicing(image, (0.1922 *255, 0.1608 *255, 0.6863 * 255), 0.2549*255)
```

结果

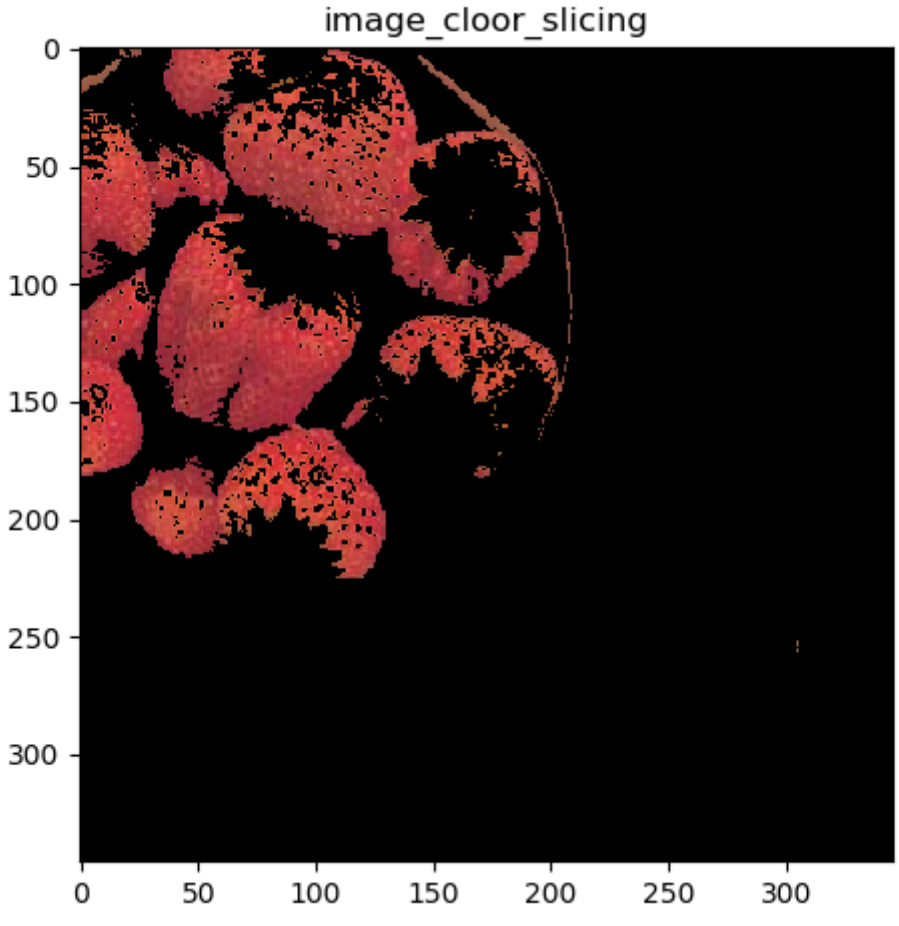
原图：



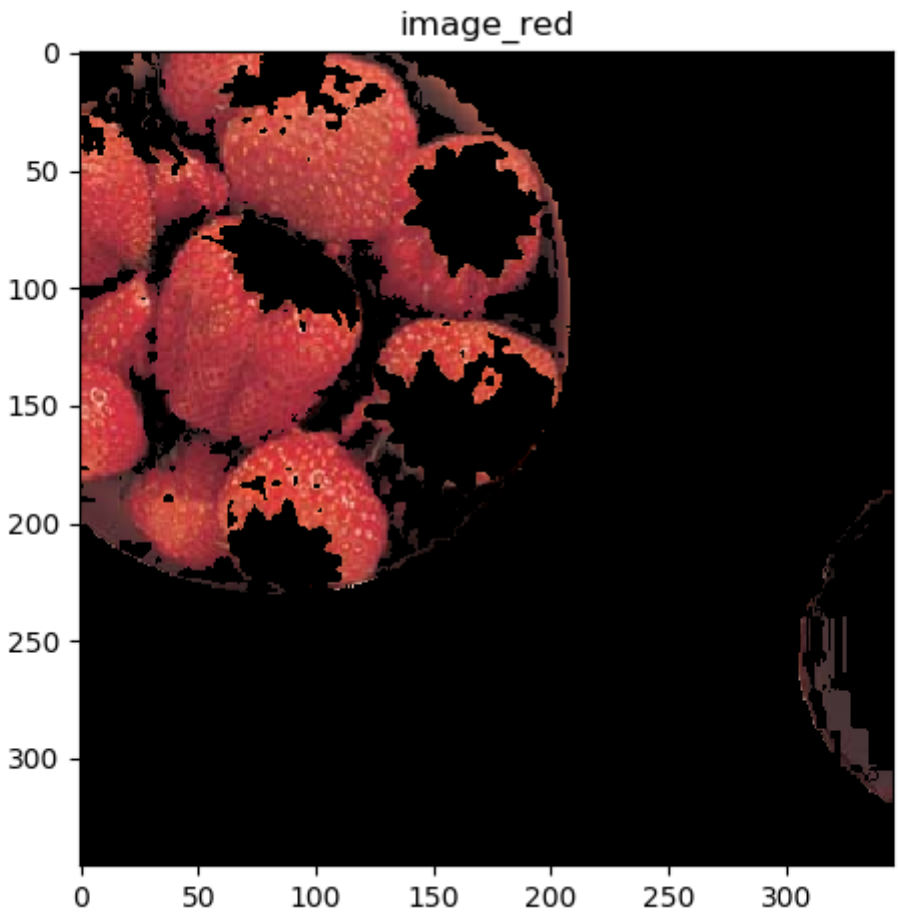
在宽度为W = 0.2549,中心在(0.6863,0.1608,0.1922)的RGB立方体中检测红色的彩色分层变换：



在宽度为W = 0.4549,中心在(0.7863,0.1608,0.1922)的RGB立方体中检测红色的彩色分层变换：



使用HSI分层：



使用HSI模型相比较于RGB模型，可以更直观的提取颜色

补充2

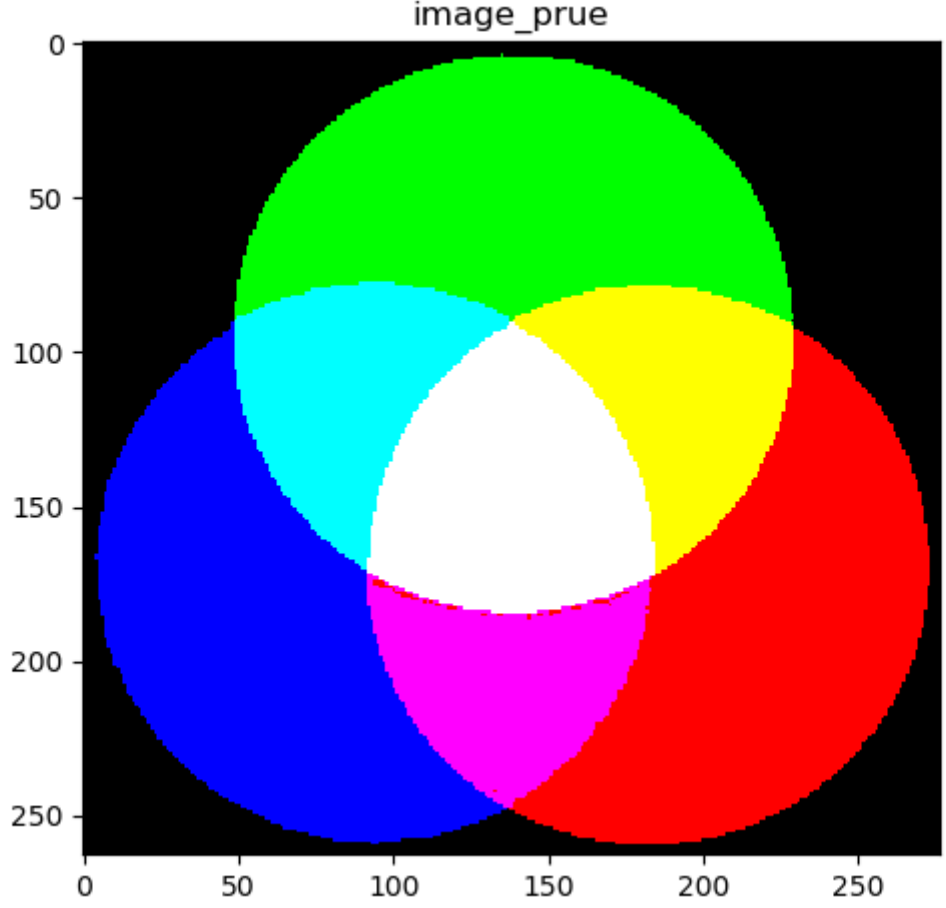
HSI2RGB

```python
def hsi_2_bgr(image):
    out = np.copy(image)

    for x in range(image.shape[0]):
        for y in range(image.shape[1]):
            h, s, i = image[x][y]
            h, s, i = h / 255 * 360, s / 255, i / 255
            b, g, r = 0, 0, 0
            # not use float in range(int, int) :(
            if h >= 0 and  h < 120:  # RG
                b = i * (1 - s)
                r = i * (1 + (s * math.cos(math.radians(h)) / math.cos(math.radians(60 - h))))
                g = 3 * i - (b + r)
```

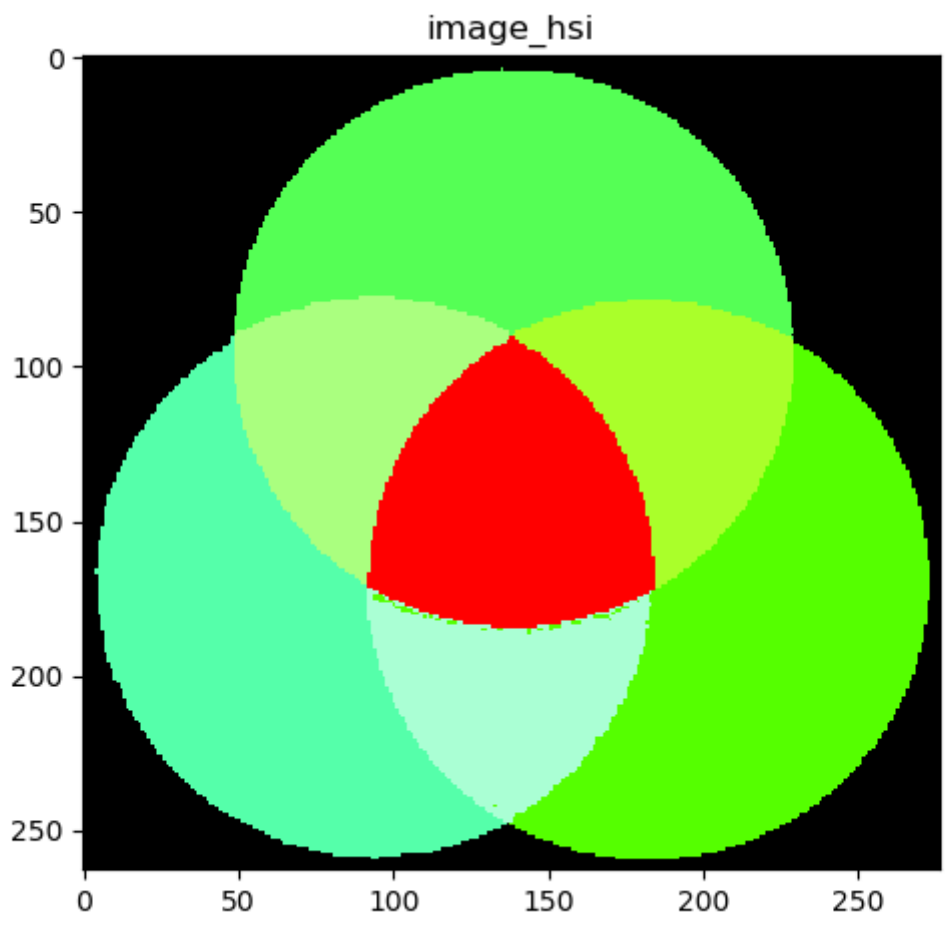
```
            elif h >= 120 and  h < 240:  # GB
                h -= 120
                r = i * (1 - s)
                g = i * (1 + (s * math.cos(math.radians(h)) / math.cos(math.radians(60 - h))))
                b = 3 * i - (r + g)
            elif h >= 240 and  h < 360:  # BR
                h -= 240
                g = i * (1 - s)
                b = i * (1 + (s * np.cos(math.radians(h)) / np.cos(math.radians(60 - h))))
                r = 3 * i - (g + b)

            out[x][y] = b * 255, g * 255, r * 255
    return out
```

结果

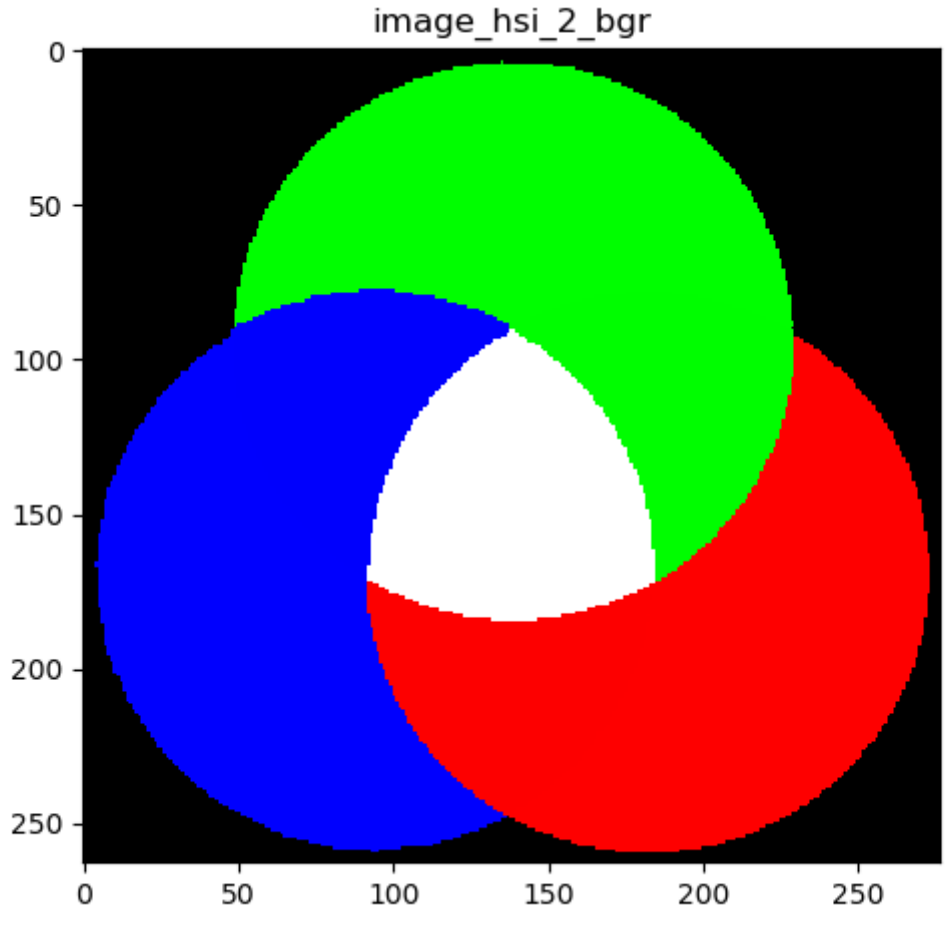
原图：



HSI：



再转回RGB：



黄、青、深红不知道为什么还原不了