

توضیح تراجع:

: HSI ~ RGB

ایجاد اسکنر رنگی می بیند:

Converting Colors from RGB to HSI

Given an image in RGB color format, the H component of each RGB pixel is obtained using the equation

$$H = \begin{cases} \theta & \text{if } B \leq G \\ 360 - \theta & \text{if } B > G \end{cases} \quad (6-16)$$

with[†]

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G)+(G-B)]}{\sqrt{[(R-G)^2 + (R-B)(G-B)]^{1/2}}} \right\} \quad (6-17)$$

The saturation component is given by

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)] \quad (6-18)$$

همینه مول حاصل مکرر θ صفر شود (کمترین مقدار)
صفر شود) $\theta \approx 0$ را دست سازی کنید

تابع تبدیل RGB ~ HSI

از طبق نظریه این تبدیل استفاده شاند:

```
def hsi_to_rgb(hsi):
    # input hsi:
    # hue range = [0, 360]
    # saturation range = [0, 1]
    # intensity range = [0, 256]
```

RGB sector ($0^\circ \leq H < 120^\circ$): When H is in this sector, the RGB components are given by the equations

$$B = I(1 - S) \quad (6-20)$$

$$R = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (6-21)$$

and

$$G = 3I - (R + B) \quad (6-22)$$

GB sector ($120^\circ \leq H < 240^\circ$): If the given value of H is in this sector, we first subtract 120° from it:

$$H = H - 120^\circ \quad (6-23)$$

Then, the RGB components are

$$R = I(1 - S) \quad (6-24)$$

$$G = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (6-25)$$

and

$$B = 3I - (R + G) \quad (6-26)$$

BR sector ($240^\circ \leq H \leq 360^\circ$): Finally, if H is in this range, we subtract 240° from it:

$$H = H - 240^\circ \quad (6-27)$$

Then, the RGB components are

$$G = I(1 - S) \quad (6-28)$$

$$B = I \left[1 + \frac{S \cos H}{\cos(60^\circ - H)} \right] \quad (6-29)$$

and

$$R = 3I - (G + B) \quad (6-30)$$

همینه خروجی این روابط مول حاصل مکرر θ را دست سازی کنید

```
def rgb_histogram(img_arr, title = "RGB Histogram"):
    r_map, g_map, b_map = rgb_map(img_arr)

    clw = 2
    plt.figure(figsize=(20, 11))
    plt.plot(r_map, color='r', label = 'Red', linewidth=clw)
    plt.plot(g_map, color='g', label = 'Green', linewidth=clw)
    plt.plot(b_map, color='b', label = 'Blue', linewidth=clw)

    plt.title(title)
    plt.ylabel("Count")
```

ایجاد اسکنر RGB histogram

```
def hsi_histogram(hsi):
    plt.figure(figsize=(20, 20))
    plt.subplot(2, 2, 1)
    plt.subplot(221),plt.hist(hsi[:, :, 0].flatten(), 75, facecolor='blue'), plt.title("Hue")
    plt.subplot(222),plt.hist(hsi[:, :, 1].flatten(), 75, facecolor='green'), plt.title("Saturation")
    plt.subplot(223),plt.hist(hsi[:, :, 2].flatten(), 255, facecolor='red'), plt.title("Intensity")
    plt.subplot(224),plt.imshow(hsi_to_rgb(hsi)), plt.title('Original Image'), plt.axis('off')
```

ایجاد 4x2 ماتریس HSI

Intensity, Saturation, Hue

(bins) کسر سازی سنتی، ایجاد HSI مول حاصل مکرر

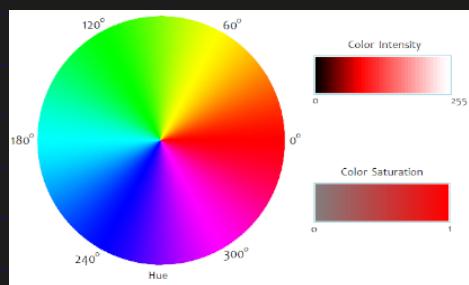
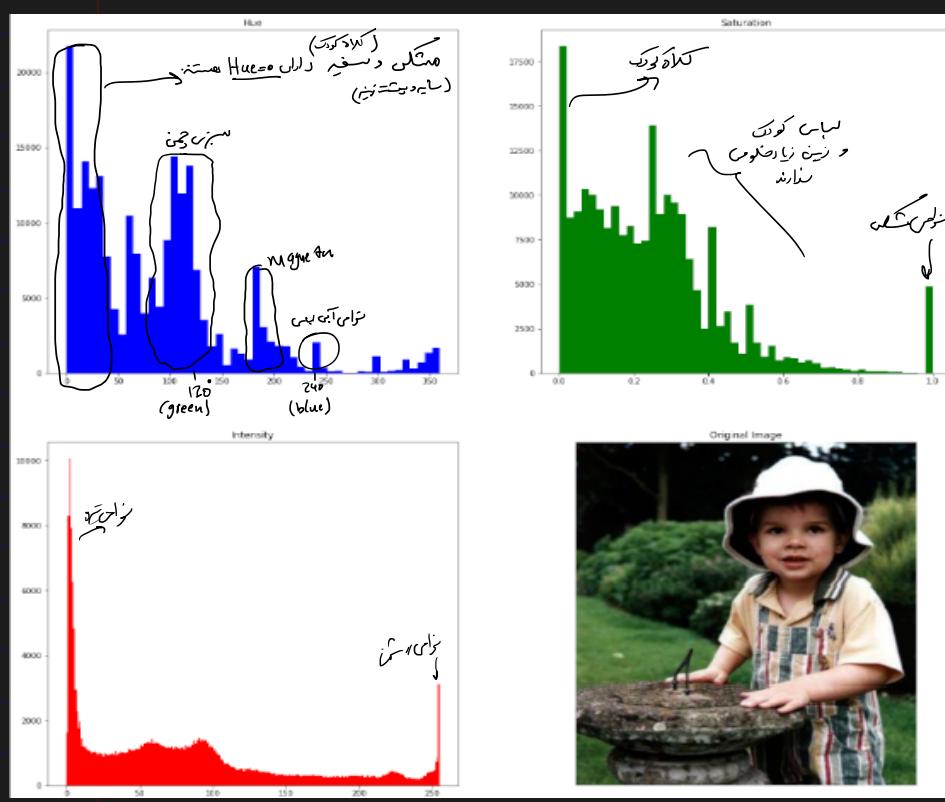
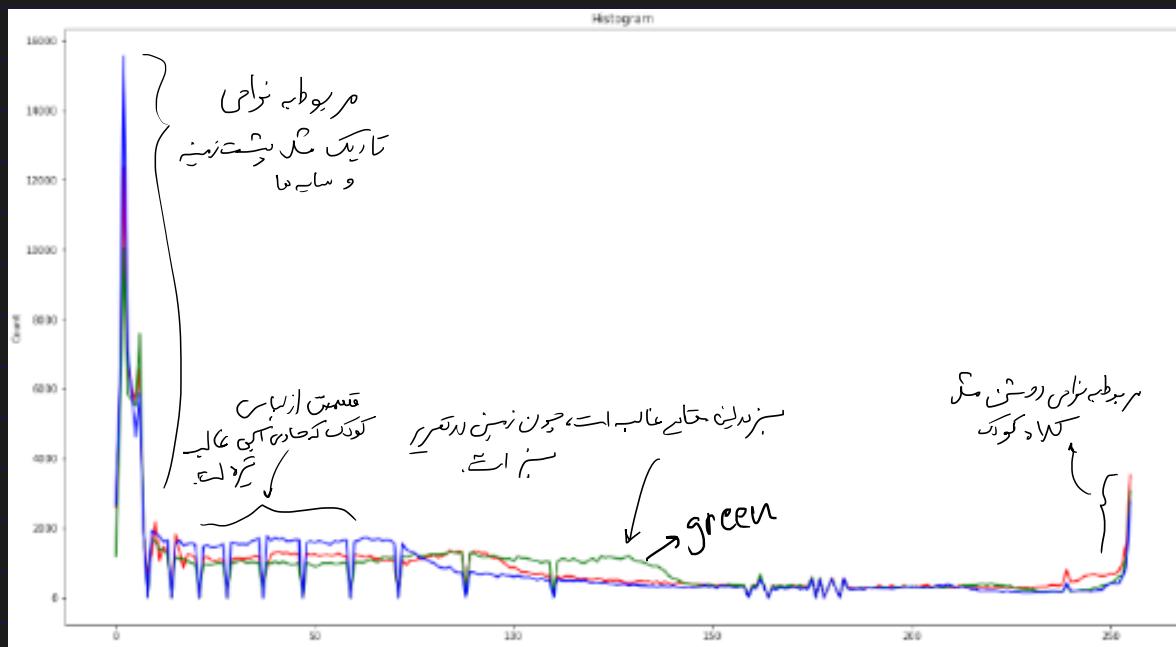
```
def display_hsi(hsi):
    plt.figure(figsize=(20, 20))
    plt.subplot(2, 2, 1)
    plt.subplot(221),plt.imshow(hsi[:, :, 0].astype('uint8'), cmap='gray'), plt.title("Hue"), plt.axis('off')
    plt.subplot(222),plt.imshow(hsi[:, :, 1], cmap='gray'), plt.title("Saturation"), plt.axis('off')
    plt.subplot(223),plt.imshow(hsi[:, :, 2].astype('uint8'), cmap='gray'), plt.title("Intensity"), plt.axis('off')
    plt.subplot(224),plt.imshow(hsi_to_rgb(hsi)), plt.title('Original Image'), plt.axis('off')
```

ایجاد 4x2 ماتریس HSI components

ایجاد 2x2 ماتریس HSI

مسئلہ ۱) (زیرج) $hs_{\text{to_hs}} - \text{ماد} \rightarrow \text{کدر صفحہ قبلی صفحہ راہ سفرم ای بہن عکس بہ صفتی } HSI \text{ کئے جائے۔}$

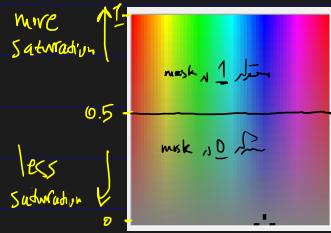
تکلیف نصیر کرام سا:



مسئلہ 2

بحث اول : Image segmentation in HSI

ارسال میں سے ایک saturation mask ہے کہ saturation کو 0.5 کم کر کر سطح ماسک برداشت کرے گا اسے سترے (آخر 0.5 cm میں)

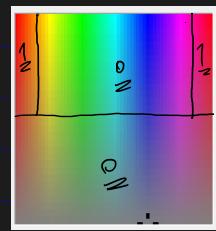
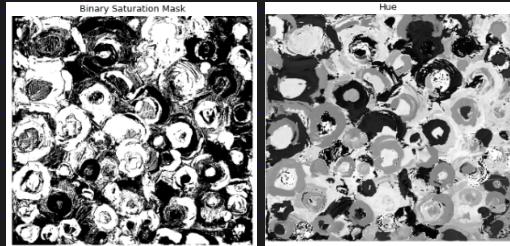


درسترنیوں کی saturation mask:

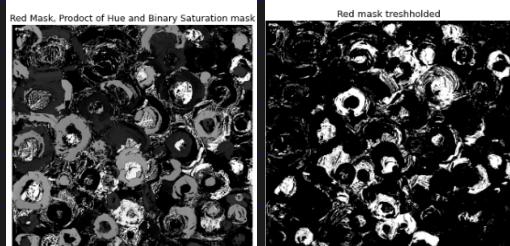
حالتاں میں سترے را دیا کر، HSI color palette کو درسترنیوں کی saturation mask کے لئے گزینہ کر دیا جائے گا

حاملہ سے Hue saturation mask کو دیکھ دیں جو با ضرور بلا فنکچر دو اور لیکن اسے ماں کے لئے

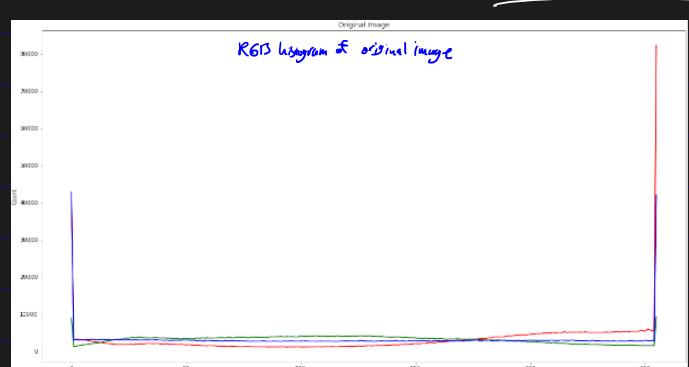
بر جکڑنے کے تردد، سوونہ Hue saturation mask $\{0-30^\circ\}$ $\{330-360^\circ\}$



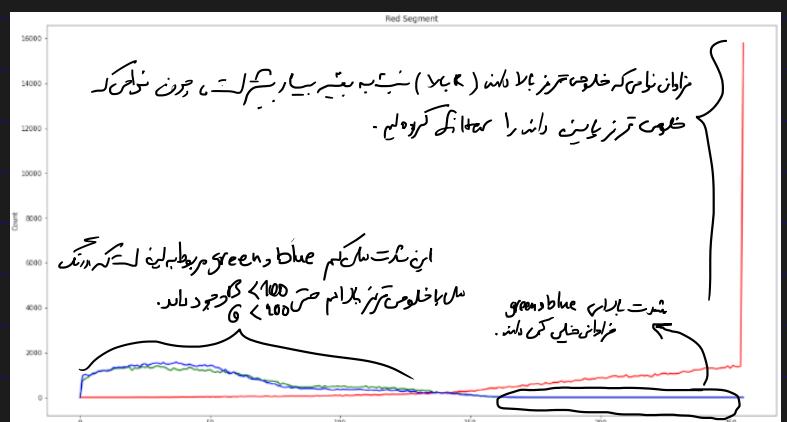
درسترنیوں کی HSI mask:



بامزہ لئے ماسک کو تعمیر کیا، تکمیل کر کر بے صراحت و
جس سے میکڑے تردد، سترے کو مانگنا (مکمل سکھن)



ھیکڑے تردد (red segment)، دت ترد ایسا ہے کہ نرم تریکریز ممکن تھا، دھیکڑے کو حفظ کے لئے جوں کو ک
تمدار لئے تاریخ بال لئے (ازاص تکمیل کی) دا حصہ
نہ ہے، لائیٹنگ کے تاریخے کے دھیکڑے کو سماں نہیں کس
کا نہ دھیکڑے تریکریز دیا دیا منہ تو نہ



بجئی (ر) : Color slicing in RGB

بالتقاضی رایج نیز، یک مکعب (و مناسی ۶۴x۶۴) از امارات نگ مردتل برای ساختن دلیل مکعب حکم mask ور فعالیتی داشتی که ۶۴x۶۴ را درست.

```
def rgb_color_slice(rgb_c, img, w):
    result = np.zeros((img.shape))
    n, m = (img.shape[0:2])

    for x in range(n):
        for y in range(m):
            flag = True
            for k in range(3):
                if np.abs(img[x][y][k] - rgb_c[k]) > w/2:
                    # not in the cube
                    flag = False
                    break

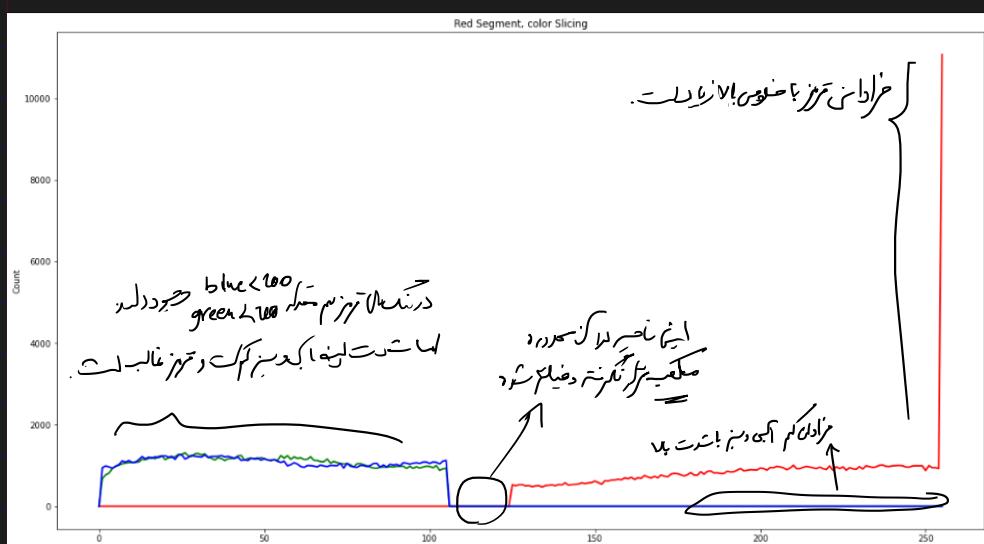
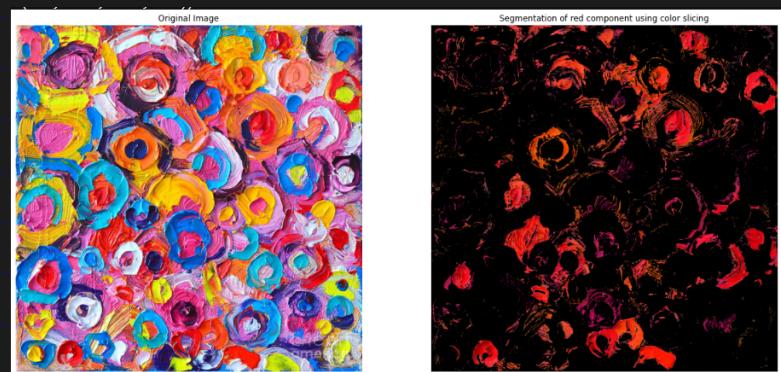
            if flag == True:
                # in cube
                for k in range(3):
                    result[x][y][k] = img[x][y][k]
            else:
                for k in range(3):
                    result[x][y][k] = 0

    return result
```

$$s_i = \begin{cases} 0.5 & \text{if } |r_j - a_j| > \frac{W}{2} \\ r_j & \text{otherwise} \end{cases} \quad \text{any } 1 \leq j \leq n \quad i = 1, 2, \dots, n \quad (6-43)$$

نگ مردتل برای جهادیان خوزرا $\left\{ \begin{array}{l} R = 200 \\ G = 50 \\ B = 30 \end{array} \right.$
انتعاجی کرو، هلاکتیا مکعب را ۱۵۰ اختیار ساخت.

: Color slicing / ر



کلیل سیگریز :
(سایه سیل مبل نه سیگریز)
 $\left\{ \begin{array}{l} R < 200 \\ G < 50 \\ B = 0 \end{array} \right.$

حالتی

برای این داده از هم ریکر تنظیم کرده از صورت lena را داشت انتها میگیریم

: Gamma adjustment < بینج اول >

```
# gamma_adjustment(rgb, gamma, channel):
    result = np.copy(rgb).astype(np.uint64)
    result[:, :, channel] = result[:, :, channel] ** gamma
    min_channel = np.amin(result[:, :, channel])
    max_channel = np.amax(result[:, :, channel])
    result[:, :, channel] = (result[:, :, channel] - min_channel) / (max_channel - min_channel) * 255
    return result
```



این بخش تصویر را با وضوح مطلقاً (channel) که اصلح گذاشتیم در نظر میگیریم
لهم تصور را ب عنوان داده های پسین

Output = Input^(γ)

مقدار
تصویر
channel

عن Scale ، 255، 0 باشد بنابراین Output = γ × Input

$$\rightarrow \gamma = 2, \gamma = 0.5 \text{ یا } \gamma = 1$$

: Saturation adjustment < بینج دوم >

```
def saturation_adjustment(hsi, sat_adj):
    res = hsi.copy()
    res[:, :, 1] = res[:, :, 1] * (1 + sat_adj/100)
    np.clip(res[:, :, 1], 0, 1, out=res[:, :, 1])
    print(np.amax(res[:, :, 1]))
    return res
```

$$\text{output Sat} = \text{Input-Sat} \times \left(\frac{100 + \text{Sat-adj}}{100} \right)$$

در این بخش مقدار Saturation میخواهیم بین 0 تا 1 بیندازیم، به همینکه مقدار HSI و Saturation را بداند [اوه سترکت] بین سaturate و راس کردن است (scale)



جستجو : Hue Shifting

Hue shifting را در اینجا سه دسته می‌دانم: مجموعی جن مکانیکی
را نمایند (0, 360) نتیجه را بین 0 و 360 درجه می‌گیرند.

```
def hue_shifting(hsi, degree):
    res = hsi.copy()
    res[:, :, 0] = (res[:, :, 0] + degree) % 360
    print(np.amax(res[:, :, 0]), np.amin(res[:, :, 0]))
    return res
```

Original Image



Hue Shiftin 60



Original Image



Hue Shiftin 180



Original Image



Hue Shiftin 120



Original Image



Hue Shiftin 240



Original Image



Hue Shiftin 300



Original Image



Hue Shiftin 360

