

Digital
Image
Process
Report #9



Color Space
- HSI/HSV

로봇기계공학과
21721160
최병희

Lab 7-1 Example: RGB space

- 문제 - 이미지를 Load하여 RGB 영역 분리

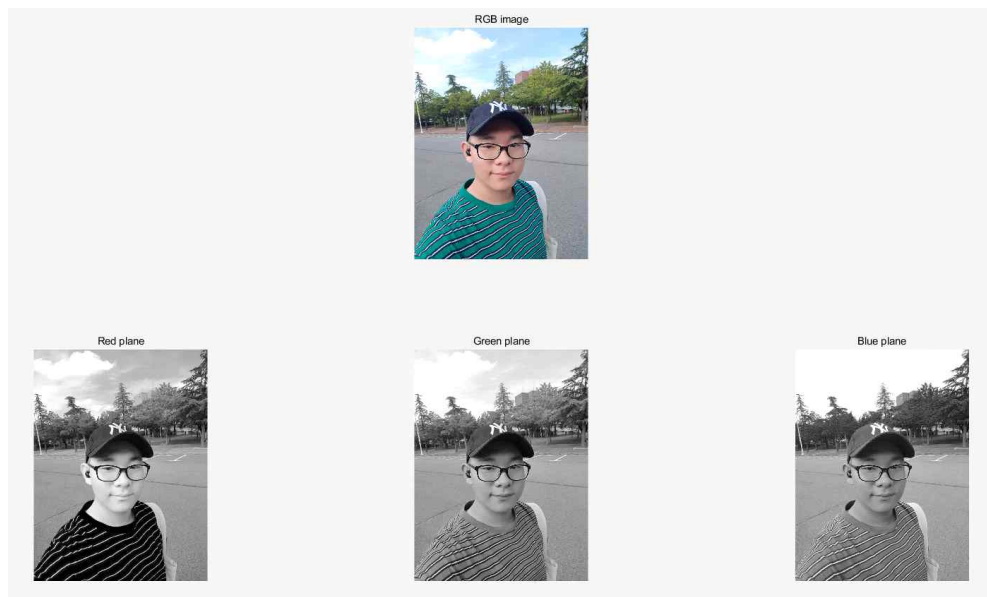
이번 과제의 목적은 Color 이미지의 Red, Green, Blue 영역을 분리하는 것입니다. 기본적으로 이미지는 빛의 삼원색인 적녹청으로 되어 있기 때문에 이미지 처리를 위하여 색 영역을 분리하는 경우에 사용합니다.

- Source code

DIP_9_Color_Space.m

```
clc;clear all;close all;  
img = imread('profile.jpg');  
  
%% Lab 7-1 Example: RGB space  
figure(1); subplot(2,3,2); imshow(img); title('RGB image');  
R = img(:,:,1);  
G = img(:,:,2);  
B = img(:,:,3);  
subplot(2,3,4); imshow(R); title('Red plane');  
subplot(2,3,5); imshow(G); title('Green plane');  
subplot(2,3,6); imshow(B); title('Blue plane');
```

- 결과 비교 및 배운 점



이미지를 Red 영역, Green 영역, Blue 영역으로 분리한 결과입니다. 처음에는 적, 녹, 청 영역으로 분리한 이미지의 색이 왜 grayscale 한 듯한 색인지 의아했으나, 검색 결과 imshow 함수는 이미지를 input된 값으로 나타내므로, 출력되는 이미지가 grayscale된 것처럼 나타나는 것이었습니다.

Lab 7-2 Example: HSI space

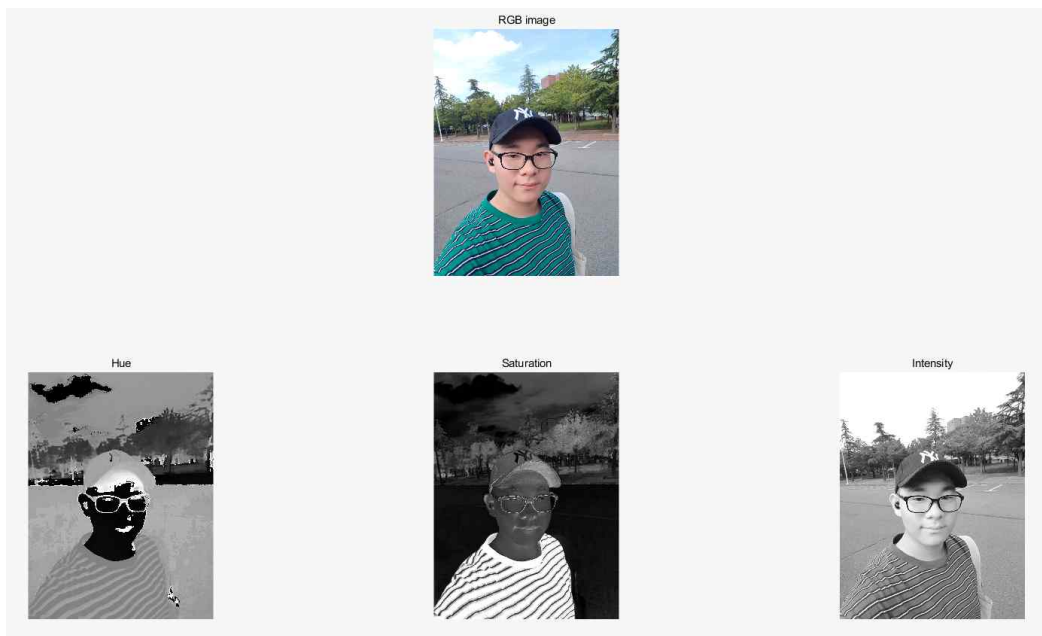
- 문제 - 이미지를 Load하여 HSI 영역 분리
이번 과제의 목적은 내장함수를 사용하여 이미지의 HSI 영역으로 분리하는 것입니다.
RGB plane 분리와 마찬가지로 이미지 처리를 위해 영역을 분리하는 데 사용합니다.
- Source code

```
DIP_9_Color_Space.m

clc;clear all;close all;
img = imread('profile.jpg');

%% Lab 7-2 Example: HSI space
hsi = rgb2hsv(img);
figure(2); subplot(2,3,2); imshow(img); title('RGB image');
H = hsi(:,:,1);
S = hsi(:,:,2);
I = hsi(:,:,3);
subplot(2,3,4); imshow(H); title('Hue');
subplot(2,3,5); imshow(S); title('Saturation');
subplot(2,3,6); imshow(I); title('Intensity');
```

- 결과 비교 및 배운 점



이미지를 Hue 영역, Saturation 영역, Intensity 영역으로 분리한 결과입니다. 각각 밝기, 백색광의 상대적인 양, 그리고 강도를 나타낸 값입니다.

Lab 7-3 MATLAB: RGB to HSI Color Model(Space)

- 문제 - 자신의 이미지로 RGB에서 HSI로 변환
이번 과제와 목적은 이미지의 RGB 영역을 분리하여 HSI로 변환하는 함수를 작성하는 것입니다. 강의자료의 공식과 검색한 공식을 참고하여 작성하였습니다.

$$I = \frac{1}{3}(R + G + B)$$
$$S = 1 - \frac{3}{(R + G + B)}[\min(R, G, B)]$$
$$H = \cos^{-1} \left[\frac{\frac{1}{2}[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right]$$

- Source code

DIP_9_Color_Space.m

```
clc;clear all;close all;
img = imread('profile.jpg');

%% Lab 7-3 MATLAB: RGB to HSI Color Model(Space)
hsi_myfunc = rgb2hsi(img);
H = hsi_myfunc(:,:,1);
S = hsi_myfunc(:,:,2);
I = hsi_myfunc(:,:,3);

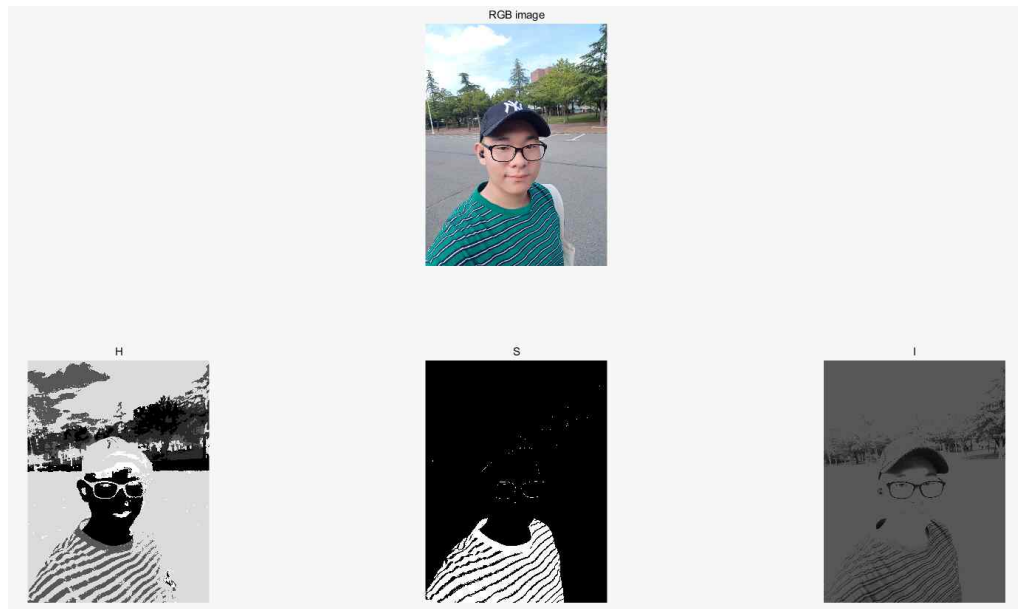
figure(3); subplot(2,3,2); imshow(img); title('RGB image');
subplot(2,3,4); imshow(uint8(H), []); title('H');
subplot(2,3,5); imshow(uint8(S), []); title('S');
subplot(2,3,6); imshow(uint8(I)); title('I');
```

rgb2hsi.m

```
function hsi = rgb2hsi(rgb)
R = rgb(:,:,1);
G = rgb(:,:,2);
B = rgb(:,:,3);
a = double(0.5*((R-G)+(R-B)));
b = sqrt(double((R-G).^2 + (R-B).*(G-B)));
value = a./b;
value(isnan(value)) = 0;
theta = acos(value);
H = theta.*(B<=G) + (2*pi-theta).*(B>G);

I = (R+G+B)/3;
num = double(min(min(R,G), B));
den = double(R+G+B);
S = 1 - 3.*num./den;
hsi = cat(3, H, S, I);
end
```

- 결과 비교 및 배운 점



제가 직접 작성한 rgb2hsi 함수의 결과입니다. Hue 영역의 결과는 비슷하게 나왔으나, Saturation 영역과 Intensity 영역의 결과는 rgb2hsv 함수와 다소 차이가 있는 것을 확인할 수 있습니다. Intensity의 결과값 차이는 cone mosaic의 분포를 적용한 결과의 차이라고 추측하고 있습니다. Saturation의 결과값 차이의 이유 음수에 대한 처리 문제라고 생각합니다.
$$-3 \frac{\min(R, G, B)}{R + G + B}$$
 식에 따르면, 결과값이 음수가 나올 수 있을 텐데, 이에 대한 처리 방법은 강의자료에서 제시하지 않았기에, 이에 대한 차이라고 생각합니다.