

# Computer Vision

## COLOR SPACE

Prof. Dr. Eng. Fitri Utaminingrum, ST, MT

# OUTLINE

## COLOR SPACE

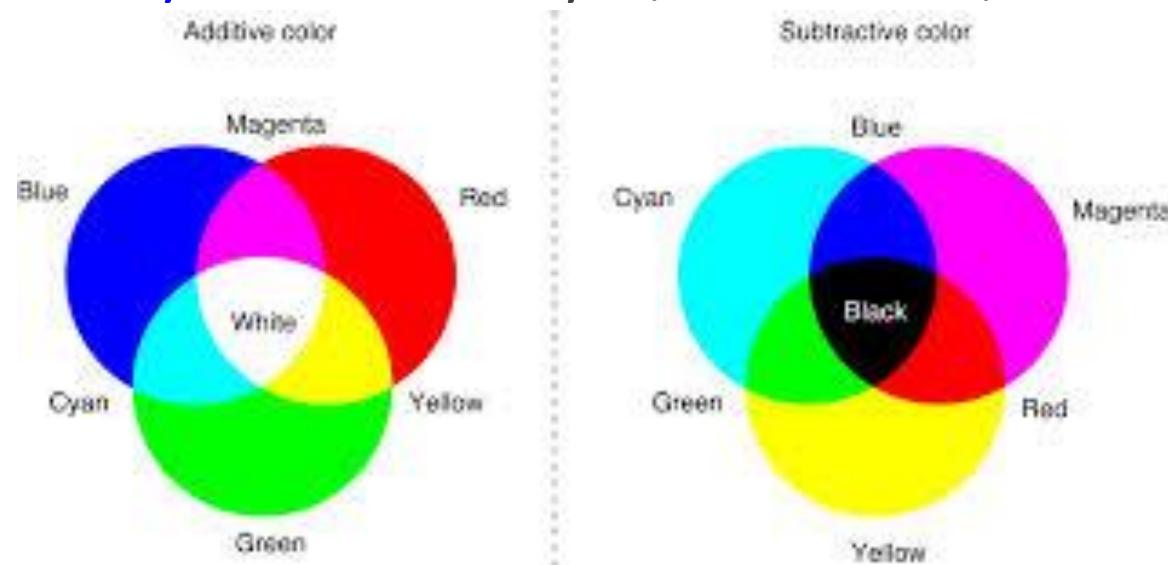
- ❑ Macam-macam ruang warna dan konversinya
- ❑ Macam-macam kompresi citra digital.

# FAKTOR YANG MEMPENGARUHI PENGLIHATAN

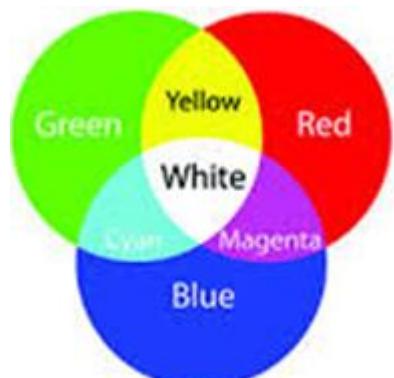
- Cahaya : spektrum energi yang menerangi permukaan objek
- Reflectance/Pantulan : perbandingan antara cahaya yang dipantulkan dengan cahaya yang datang
- Specularity : specular tinggi (mengkilap) vs. permukaan kasar
- Jarak : jarak dengan sumber cahaya
- Sudut : sudut antara permukaan objek dan sumber cahaya
- Sensitivity : Seberapa sensitif sensor yang digunakan

# Primary and secondary colors

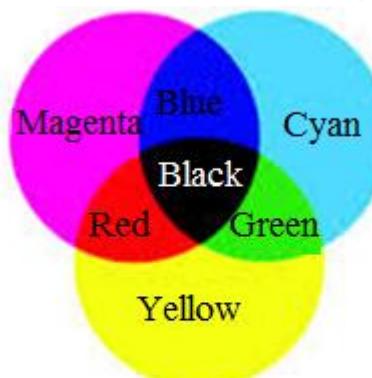
- ▶ Pada 1931, CIE (International Commission on Illumination) mendefinisikan nilai spesifikasi panjang gelombang untuk warna primer:
  - ▶  $B = 435.8 \text{ nm}$ ,  $G = 546.1 \text{ nm}$ ,  $R = 700 \text{ nm}$
- ▶ Secondary colors:  $G+B=\text{Cyan}$ ,  $R+G=\text{Yellow}$ ,  $R+B=\text{Magenta}$



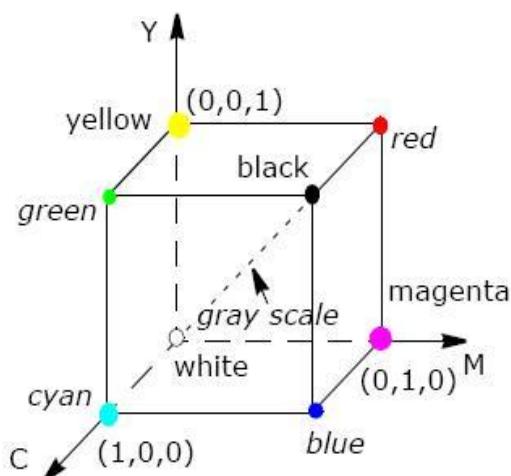
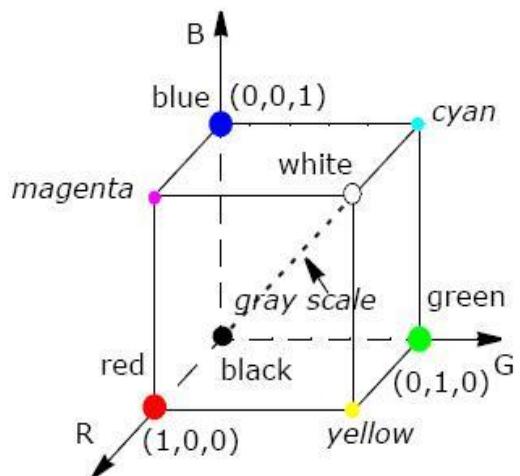
# Hubungan CMY dan RGB



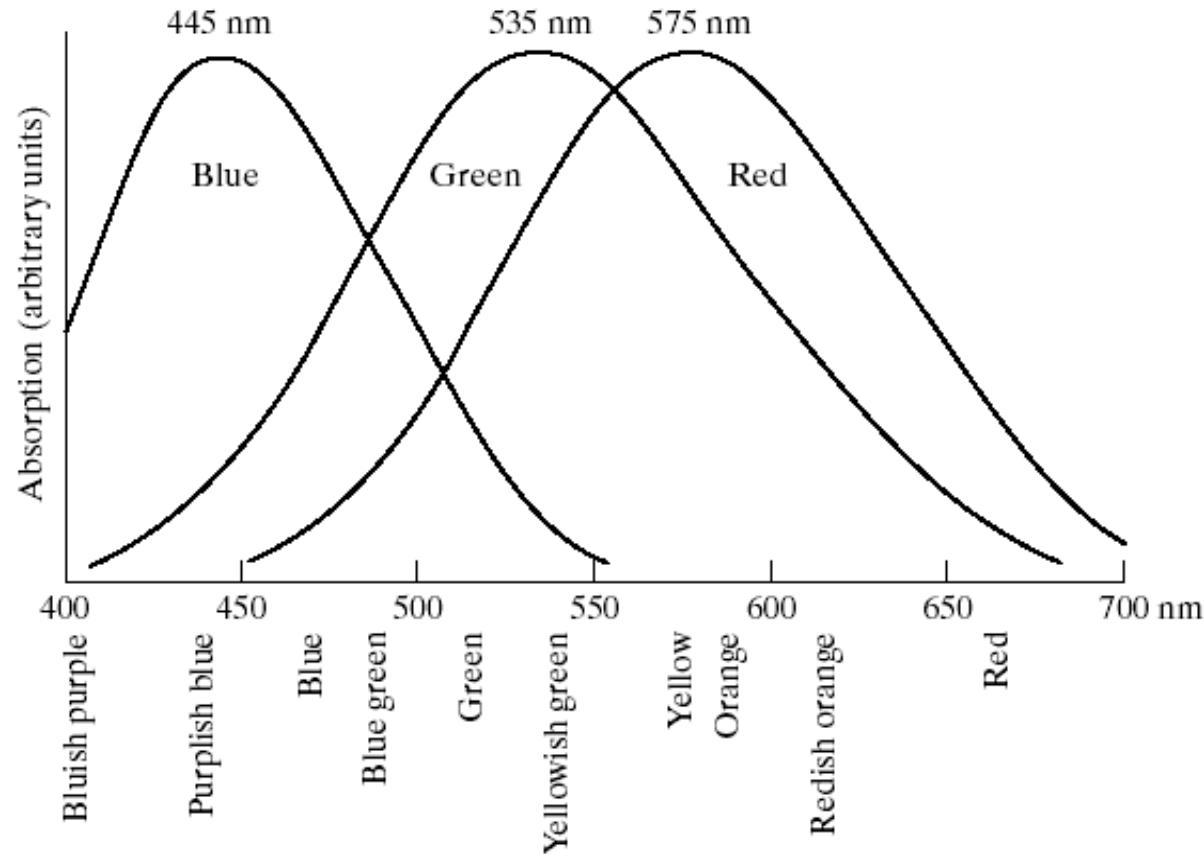
RGB



CMY



# Panjang Gelombang RGB



**FIGURE 6.3** Absorption of light by the red, green, and blue cones in the human eye as a function of wavelength.

- ▶ Color model, color space, color system
    - ▶ Spesifikasi warna dalam cara standar
    - ▶ Sebuah coordinate system yang setiap warna direpresentasikan oleh single point
  
  - ▶ RGB model
  - ▶ CYM model
  - ▶ CYMK model
  - ▶ HSI model
  - ▶  $YC_bC_r$
  - ▶ YUV dan YIQ
- { Suitable for hardware or applications  
- match the human description

# Color models

- **RGB** adalah sistem aditif (menambahkan warna-warna menjadi hitam) yang digunakan pada display/monitor.
- **CMY** adalah sistem subtractive yang digunakan untuk pencetakan.
- **HIS** dan **HSL** adalah ruang perceptual yang cukup bagus untuk keperluan seni, psikologi dan pengenalan.
- **HSV** dan **YCbCr** adalah baik digunakan untuk deteksi kulit
- **YUV** dan **YIQ** digunakan untuk TV (baik untuk kompresi).

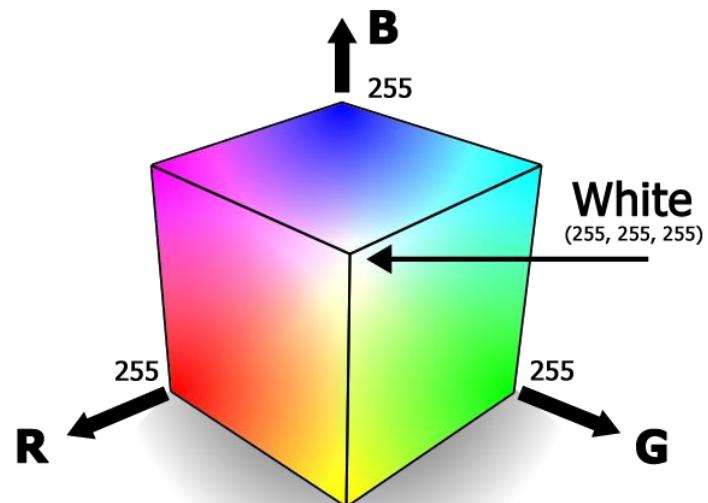
# Citra Berwarna (RGB)

- ✓ Secara *default* kamera akan menghasilkan gambar yang mempunyai ruang warna RGB yang artinya *Red*, *Green*, dan *Blue*
- ✓ Setiap komponen warna pada RGB mengandung 8 bit yang nilainya berkisar antara 0-255
- ✓ Kemungkinan pilihan warna yang dapat disajikan adalah  $255 \times 255 \times 255 = 16.581.375$  warna

# Citra Berwarna (RGB)

*Contoh warna dan nilai penyusun warna*

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255
Kuning	0	255	255



# Color palette dan RGB yang dinormalisasi

Intensitas

$$I = (R+G+B) / 3$$

Normalized red

$$r = R/(R+G+B)$$

Normalized green

$$g = G/(R+G+B)$$

Normalized blue

$$b = B/(R+G+B)$$

Pada representasi yang sudah dinormalisasi,  $b = 1 - r - g$ ,  
Sehingga kita hanya perlu Melihat r dan g untuk menentukan warna.

# Pixel depth

- ▶ Pixel depth: nilai bit yang digunakan untuk merepresentasikan setiap pixel dalam RGB space
- ▶ Full-color image: 24-bit RGB color image
  - ▶  $(R, G, B) = (8 \text{ bits}, 8 \text{ bits}, 8 \text{ bits})$

## CMY/CMYK

- CMY umumnya digunakan oleh Printer warna.
- Sehingga diperlukan konversi dari RGB ke CMY/CMYK



- CMY color space terdiri dari cyan, magenta, dan yellow.
- CMY adalah kebalikan/komplemen dari RGB color space.

# Hubungan CMY dan RGB

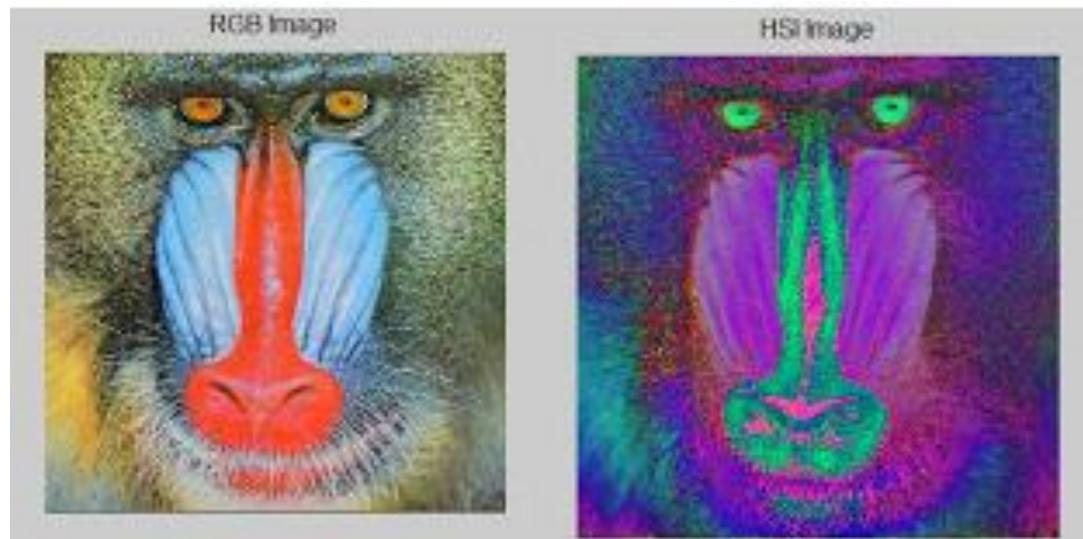
- Rubah range RGB (0 - 255) → (0 - 1) dengan cara membagi dengan 255
  - RGB ke CMY
    - $C = 1.0 - R$
    - $M = 1.0 - G$
    - $Y = 1.0 - B$
  - CMY ke RGB
    - $R = 1.0 - C$
    - $G = 1.0 - M$
    - $B = 1.0 - Y$

# Konversi CMY ke CMYK

- Konveri dari CMY ke **CMYK** adalah dengan menambahkan komponen hitam (K) ke CMY.
  - **K** = min(C, M, Y)
  - **C** = C - K
  - **M** = M - K
  - **Y** = Y - K



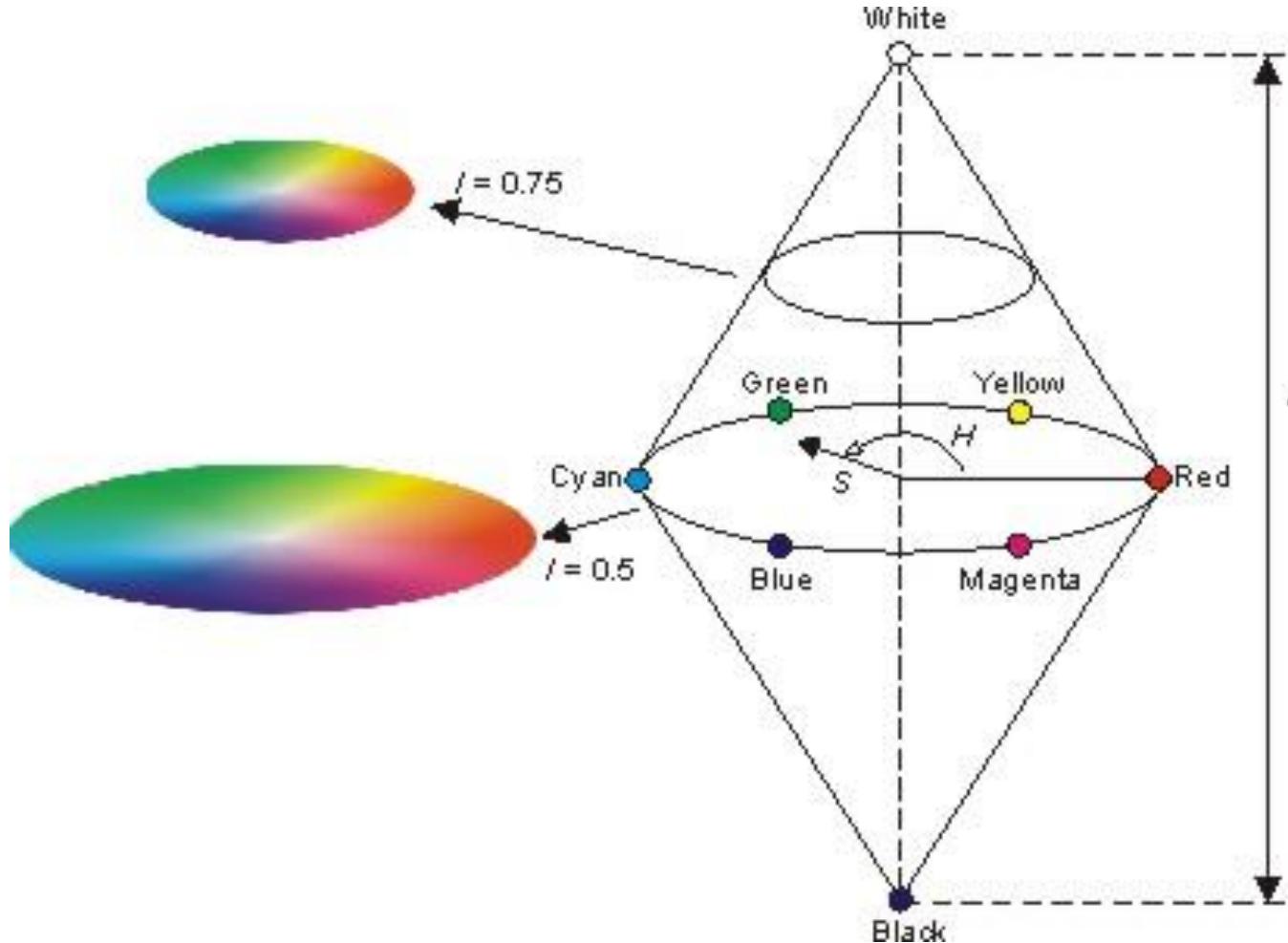
- HSI adalah hue, saturation, dan intensity yang digunakan dalam mendeskripsikan warna.
- Ketika menggunakan HSI color space, kita hanya butuh mengatur hue untuk mendapatkan warna yang diinginkan.



# Aplikasi HSI color model

- Machine vision menggunakan HSI color space dalam mengidentifikasi warna dari obyek yang berbeda
- Operasi-operasi tersebut sangat mudah dilakukan pada citra dengan menggunakan HSI color space.

# HSI dimodelkan dalam Koordinat silinder



## Konversi RGB ke HSI

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

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

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

- Jika B lebih besar dari G, maka  $H_1 = 2\pi - H'$
- Jika B Kurang dari G, maka  $H_1 = H'$
- Untuk mendapatkan  $H = H_1 / (2\pi)$

# Mengkonversi HSI ke RGB

Untuk mengkonversi HSI ke RGB tergantung pada posisi sector warna H

- Untuk RG sector ( $0^0 \leq H \leq 120^0$ ):

$$b = \frac{I}{3}(1 - S)$$

$$r = \frac{I}{3} \left[ I + \frac{S \cos(H)}{\cos(60^0 - H)} \right]$$

$$g = I - (r + b)$$

# Mengkonversi HSI ke RGB

- Untuk GB sector ( $120^\circ \leq H \leq 240^\circ$ ):

$$H = H - 120^\circ$$

$$g = \frac{I}{3} \left[ 1 + \frac{S \cos(H)}{\cos(60^\circ - H)} \right]$$

$$r = \frac{I}{3} (1 - S)$$

$$b = I - (r + g)$$

# Mengkonversi HSI ke RGB

- Untuk BR sector ( $240^\circ \leq H \leq 360^\circ$ ):

$$H = H - 240^\circ$$

$$g = \frac{I}{3} \left[ 1 + \frac{S \cos(H)}{\cos(60^\circ - H)} \right]$$

$$r = \frac{I}{3} (1 - S)$$

$$b = I - (r + g)$$

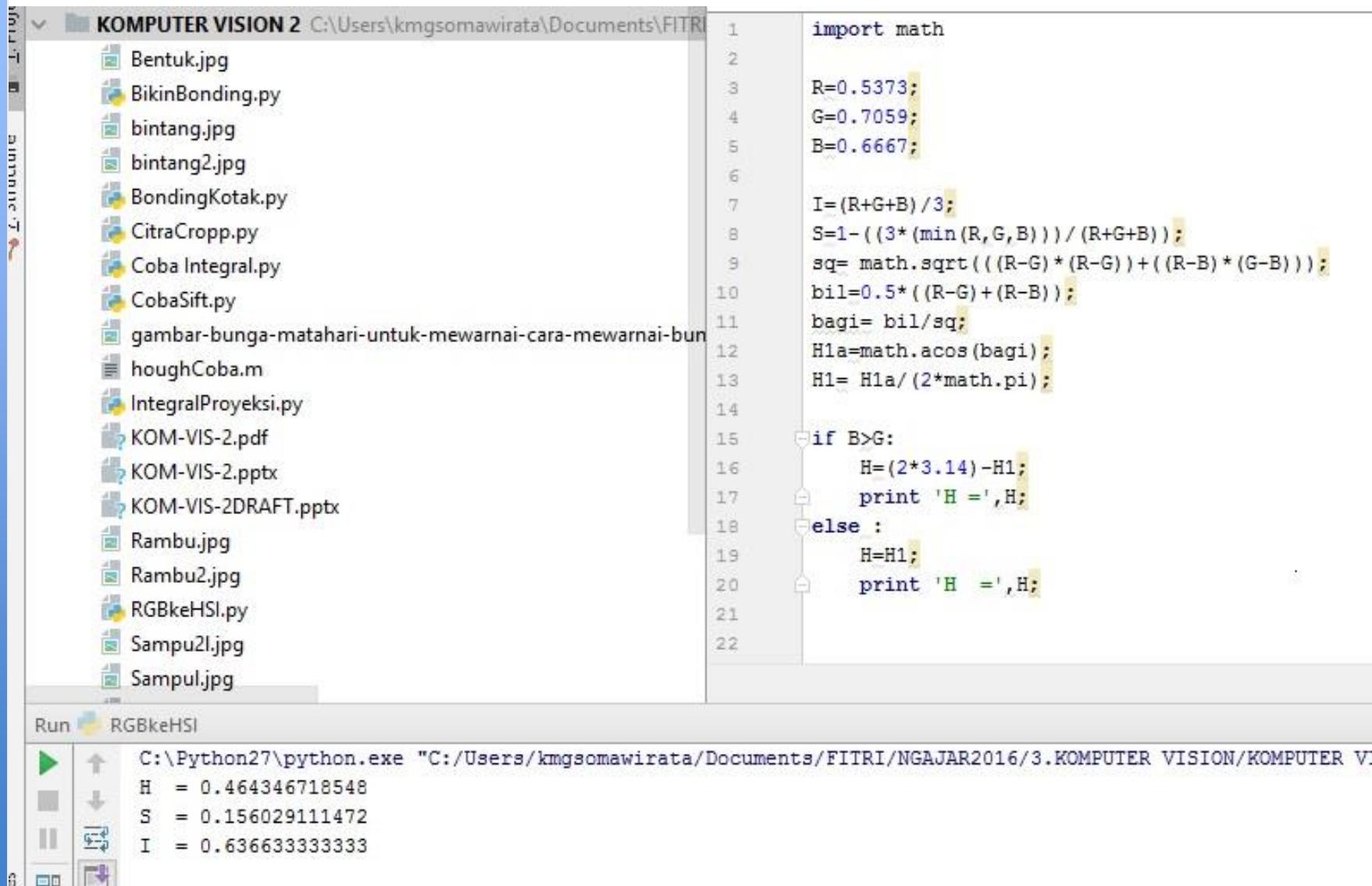
# Mengkonversi HSI ke RGB

- Untuk mengkonversikan dari r, g dan b ke R, G dan B maka:
  - $R=3Ir$
  - $G=3Ig$
  - $B=3Ib$

# LATIHAN

1. Konversikan data digital RGB ke HSI
  - R = 0.5373;
  - G = 0.7059
  - B = 0.6667
2. Konversikan kembali hasilnya HSI ke RGB

# RGB ke HIS Python



The image shows a screenshot of a Python code editor. On the left, there is a file browser window titled "KOMPUTER VISION 2" showing various files in the directory C:\Users\kmgsomawirata\Documents\FITRI. On the right, the code editor displays a Python script for converting RGB values to HSI values.

```
1 import math
2
3 R=0.5373;
4 G=0.7059;
5 B=0.6667;
6
7 I=(R+G+B)/3;
8 S=1-((3*(min(R,G,B)))/(R+G+B));
9 sq= math.sqrt(((R-G)*(R-G))+((R-B)*(G-B)));
10 bil=0.5*((R-G)+(R-B));
11 bagi= bil/sq;
12 H1a=math.acos(bagi);
13 H1= H1a/(2*math.pi);
14
15 if B>G:
16     H=(2*3.14)-H1;
17     print 'H =',H;
18 else :
19     H=H1;
20     print 'H =',H;
21
22
```

The Run tab at the bottom shows the output of the script:

```
C:\Python27\python.exe "C:/Users/kmgsomawirata/Documents/FITRI/NGAJAR2016/3.KOMPUTER VISION/KOMPUTER VI
H = 0.464346718548
S = 0.156029111472
I = 0.6366333333333333
```

# RGB ke HIS seluruh data Citra (Matlab)

```
>> RGBkeHSI2
H =
    0.4643
S =
    0.1560
I =
    0.6366
fx >>
```

	PosisiWajah.m	detmata.m	DeteksiKulit.m	HISkeRGB.m	RGBkeHSikonvesi.
1 -	clear all;				
2 -	r=0.5373;				
3 -	g=0.7059;				
4 -	b=0.6667;				
5 -	th=acos((0.5*((r-g)+(r-b)))./((sqrt((r-g).^2+(r-b).*(g-b)))));				
6 -	H=th;				
7 -	H(b>g)=2*pi-H(b>g);				
8 -	H=H/(2*pi);				
9 -	S=1-3.*min(min(r,g),b)./(r+g+b);				
10 -	I=(r+g+b)/3;				
11 -	H, S, I				
12					

# HIS ke RGB

Command Window

```
>> HSIkeRGB
```

R =

0.5373

G =

0.7059

B =

0.6666

fxt >>

+2 Editor - C:\Users\kmgsomawirata\Documents\FITRI\NC MainTraffic.m MixNeighbourhood.m PosisiWajah

```
1 - clear all
2 - H1=0.4643;
3 - S= 0.1560;
4 - I= 0.6366;
5 -
6 - H=H1*2*pi;
7 - if H>0 & H < 2*pi/3
8 - b=(1-S)/3;
9 - r=(1+S*cos(H)/cos(pi/3-H))/3;
10 - g=1-(r+b);
11 - elseif H>2*pi/3 & H < 4*pi/3
12 - H=H-2*pi/3;
13 - r=(1-S)/3;
14 - g=(1+S*cos(H)/cos(pi/3-H))/3;
15 - b=1-(r+g);
16 - else
17 - H=H-4*pi/3
18 - g=(1-S)/3
19 - b=(1+S*cos(H)/cos(pi/3-H))/3
20 - r=1-b-g
21 - end
22 - R=3*I*r
23 - G=3*I*g
24 - B=3*I*b
25 -
26
```

# Hue Saturation Value

- Untuk mengkonversi dari RGB ke HSV (assuming normalised RGB values) dengan mencari nilai maximum dan minimum dari RGB. Sehingga S:

$$S = \frac{(\max - \min)}{\max}$$

- Jika  $S = 0$  maka Hue tidak terdefinisikan
- Nilai V:

$$V = \max$$

- H dihitung dari:

$$R' = \frac{\max-R}{\max-\min}$$

$$G' = \frac{\max-G}{\max-\min}$$

$$B' = \frac{\max-B}{\max-\min}$$

# Menghitung nilai H

if R = max and G = min

$$H = 5 + B'$$

else if R = max and G  $\neq$  min

$$H = 1 - G'$$



else if G = max and B = min

$$H = R' + 1$$

else if G = max and B  $\neq$  min

$$H = 3 - B'$$

else if R = max

$$H = 3 + G'$$

otherwise

- Kemudian H dikonversikan menjadi derajat dengan mengalikannya dengan 60. Sehingga H memiliki ring 0 s/d 360, sedangkan S dan V memiliki ring 0 s/d 1

# RGB ke HSV

## Python

```
1 import cv2
2 import numpy as np
3
4 img=cv2.imread('parot.jpg')
5 cv2.imshow('Citra Input',img)
6 hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
7 cv2.imshow('Citra RGB ke HSVI',hsv)
8
9 cv2.waitKey(0)
10 cv2.destroyAllWindows()
11
```



# Konversi balik HSV ke RGB

- Langkah pertama hue H dibagi dengan 60

$$\text{Hex} = \frac{H}{60}$$

- Selanjutnya menghitung nilai, primary color, secondary color, nilai a, b dan c.
- Primary color diperoleh dengan membulatkan ke bawah dari Hex diatas.
- Secondary color, a, b, dan c:

$$\text{secondary colour} = \text{Hex} - \text{primary colour}$$

$$a = (1 - S)V$$

$$b = (1 - (S * \text{secondary colour}))V$$

$$c = (1 - (S * (1 - \text{secondary colour})))V$$

# Sehingga R, G dan B didapat:

if primary colour = 0 then

$$R = V, G = c, B = a$$

if primary colour = 1 then

$$R = b, G = V, B = a$$

if primary colour = 2 then

$$R = a, G = V, B = c$$

if primary colour = 3 then

$$R = a, G = b, B = V$$

if primary colour = 4 then

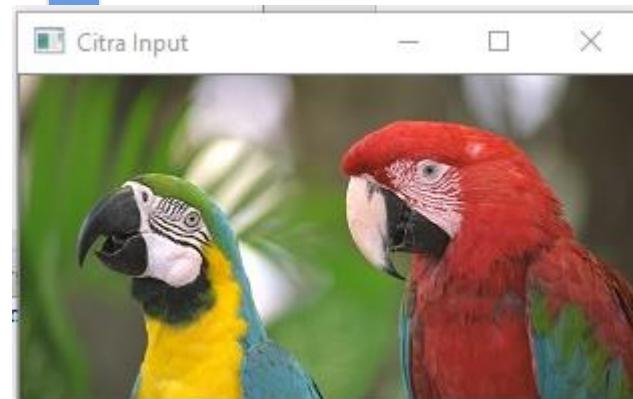
$$R = c, G = a, B = V$$

if primary colour = 5 then

$$R = V, G = a, B = b$$

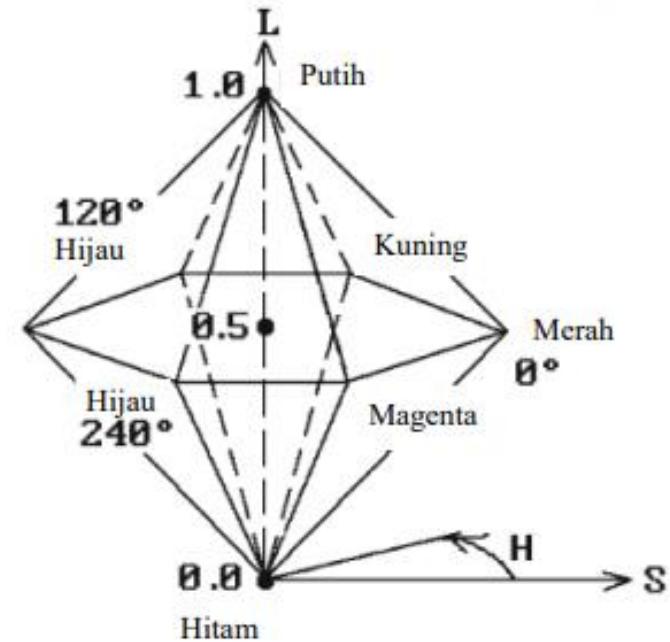
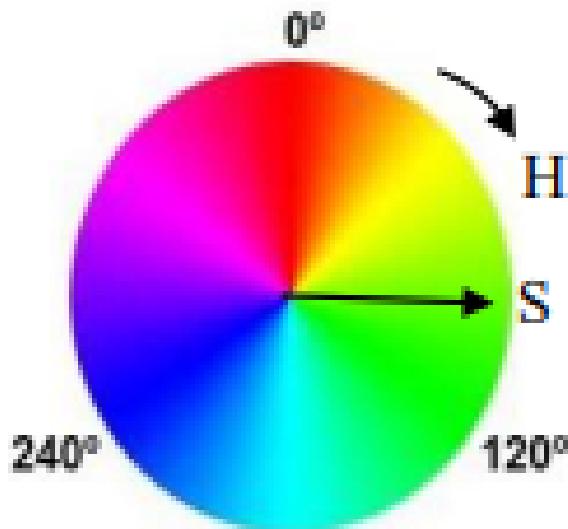
# HSV ke RGB

```
1 import cv2
2 import numpy as np
3
4 img=cv2.imread('parot.jpg')
5 cv2.imshow('Citra Input',img)
6 hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
7 cv2.imshow('Citra RGB ke HSV',hsv)
8
9 rgb = cv2.cvtColor(hsv, cv2.COLOR_HSV2BGR)
10
11 cv2.waitKey(0)
12 cv2.destroyAllWindows()
```



# HSL

Model ruang HSL, Besaran kecerahan (dinamakan lightness) disimbolkan dengan L. Perhitungan komponen H, S, dan L berdasarkan komponen R,G, dan B adalah seperti berikut (Agoston, 2005).



# KONVERSI RGB TO HSL

$$C_{min} = \min(R, G, B)$$

$$C_{max} = \max(R, G, B)$$

$$L = \frac{C_{max} + C_{min}}{2}$$

$$S = \begin{cases} 0, & C_{min} = C_{max} \\ \frac{C_{max} - C_{min}}{C_{max} + C_{min}}, & L \leq 0,5 \\ \frac{C_{max} - C_{min}}{2 - (C_{max} + C_{min})}, & L > 0,5 \end{cases}$$

$$H = \begin{cases} \text{UNDEFINED}, & C_{min} = C_{max} \\ \frac{G - B}{C_{max} - C_{min}}, & R = C_{max} \\ 2 + \frac{B - R}{C_{max} - C_{min}}, & G = C_{max} \\ 4 + \frac{R - G}{C_{max} - C_{min}}, & \text{untuk lainnya} \end{cases}$$

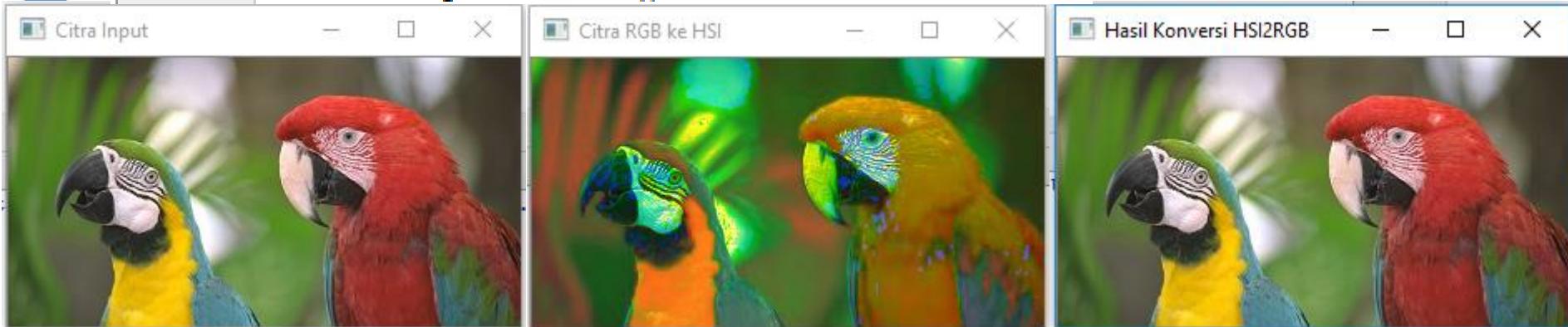
$$H = H \times 60$$

Jika  $H < 0$  maka  $H = H + 360$

# KONVERSI RGB TO HSL

## HSL to RGB

```
1 import cv2
2 import numpy as np
3
4 img=cv2.imread('parot.jpg')
5 cv2.imshow('Citra Input',img)
6 hsl = cv2.cvtColor(img, cv2.COLOR_BGR2HLS)
7 cv2.imshow('Citra RGB ke HSI',hsl)
8 hslkergb = cv2.cvtColor(hsl, cv2.COLOR_HLS2BGR)
9 cv2.imshow('Hasil Konversi HSI2RGB',hslkergb)
10
11 cv2.waitKey(0)
12 cv2.destroyAllWindows()
```



## YC<sub>b</sub>C<sub>r</sub>

- YC<sub>b</sub>C<sub>r</sub> adalah suatu keluarga ruangan warna yang dipakai sebagai bagian dari *Color image pipeline* dalam sistem video dan fotografi digital.
- Pada ruang warna ini, komponen Y menyatakan intensitas/luminan, sedangkan C<sub>b</sub> dan C<sub>r</sub> menyatakan informasi warna. Blue didekodekan ke C<sub>b</sub> dan Red menjadi C<sub>r</sub>

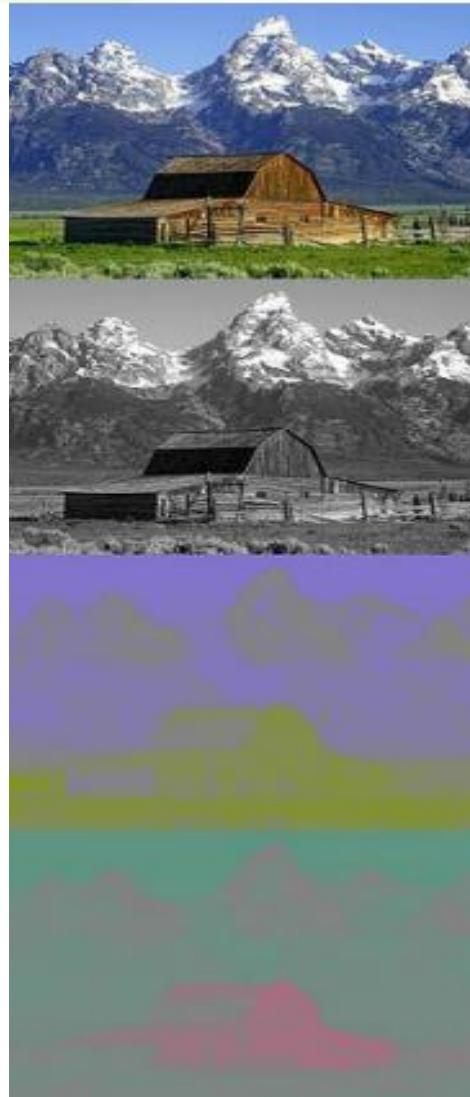
## YC<sub>b</sub>C<sub>r</sub>

- YC<sub>b</sub>C<sub>r</sub> adalah ruang warna lain yang memisahkan antara luminan dan informasi warna:
- Untuk mengkonversikan RGB ke YC<sub>b</sub>C<sub>r</sub> (Crane, 1997):
  - $Y = 0.29900R + 0.58700G + 0.11400B$
  - $C_b = -0.16874R - 0.33126G + 0.50000B$
  - $C_r = 0.50000R - 0.41869G - 0.08131B$

# Konversi YC<sub>b</sub>C<sub>r</sub> ke RGB

- $R = 1.00000Y + 1.40200C_r$
- $G = 1.00000Y - 0.34414C_b - 0.71414C_r$
- $B = 1.00000Y + 1.77200C_b$

# Konversi RGB ke YC<sub>b</sub>C<sub>r</sub>



RGB

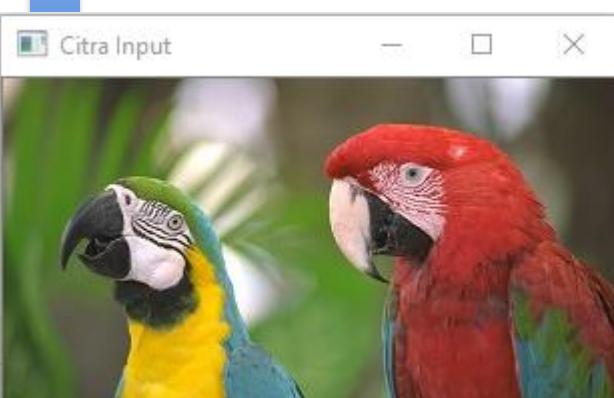
Y

C<sub>b</sub>

C<sub>r</sub>

# Konversi RGB ke YC<sub>b</sub>C<sub>r</sub>

```
1 import cv2
2 import numpy as np
3
4 img=cv2.imread('parrot.jpg')
5 cv2.imshow('Citra Input',img)
6 ycbcr = cv2.cvtColor(img, cv2.COLOR_BGR2YCrCb)
7 cv2.imshow('Citra RGB ke YCrCb',ycbcr)
8 ycbcrkergb = cv2.cvtColor(ycbcr, cv2.COLOR_YCrCb2BGR)
9 cv2.imshow('Hasil Konversi YCrCb ke RGB',ycbcrkergb)
10
11 cv2.waitKey(0)
12 cv2.destroyAllWindows()
```



# YIQ dan YUV

- Memiliki property kompresi yang lebih baik
- Luminance Y dikodekan dengan menggunakan bit yang lebih banyak dibanding chrominance I dan Q; mata manusia lebih sensitif terhadap Y dibanding I,Q
- Luminance digunakan pada TV hitam putih.
- Ketiga nilai Y, I dan Q, digunakan pada TV berwarna.
- Pengkodean YUV digunakan pada video digital dan kompresi JPEG dan MPEG

## KONVERSI RGB TO YUV

Model YUV terdiri dari komponen luminance/brightness (Y) dan dua komponen konten warna / chrominance (U dan V).

Konversi dari RGB ke model YUV :

- Gonzales (2002)

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- V. Santhi dan Dr. Arunkumar (2009)

$$y = (0.257 \times R) + (0.504 \times G) + (0.098 \times B) + 16$$

$$v = (0.439 \times R) - (0.368 \times G) - (0.071 \times B) + 128$$

$$u = (0.148 \times R) - (0.291 \times G) + (0.439 \times B) + 128$$

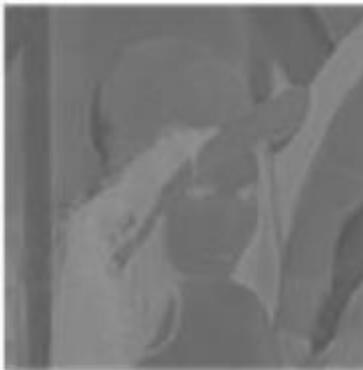
# KONVERSI RGB TO YUV



RGB



Y



U



V

# KONVERSI RGB TO YUV

```
1 import cv2
2 import numpy as np
3
4 img1=cv2.imread('parot.jpg')
5 cv2.imshow('Citra Input',img1)
6 img=cv2.imread('parot.jpg')
7
8 img[:, :, 0]= 0.299*img[:, :, 0]+0.587*img[:, :, 1]+0.114*img[:, :, 2]
9 img[:, :, 1]= -0.169*img[:, :, 0]-0.311*img[:, :, 1]+0.5*img[:, :, 2]
10 img[:, :, 2]= 0.5*img[:, :, 0]-0.419*img[:, :, 1]-0.081*img[:, :, 2]
11
12 cv2.imshow('Hasil YUV',img)
13 cv2.waitKey(0)
14 cv2.destroyAllWindows()
```



# RUANG WARNA YIQ

Ruang warna YIQ, yang juga dikenal dengan nama ruang warna NTSC, dirumuskan oleh NTSC ketika mengembangkan sistem televisi berwarna di Amerika Serikat. Pada model ini, Y disebut luma (yang menyatakan luminans) dan I serta Q disebut chroma.

# KONVERSI RGB TO YIQ

Transformasi linear dari RGB ke YIQ:

$$\text{luminance } Y = 0.30R + 0.59G + 0.11B$$

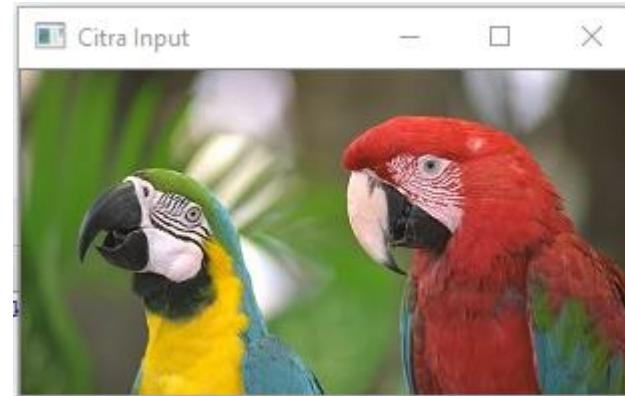
$$R - \text{cyan } I = 0.60R - 0.28G - 0.32B$$

$$\text{magenta} - \text{green } Q = 0.21R - 0.52G + 0.31B$$

Persamaan di atas sering digunakan untuk mengkonversi citra berwarna ke citra keabuan

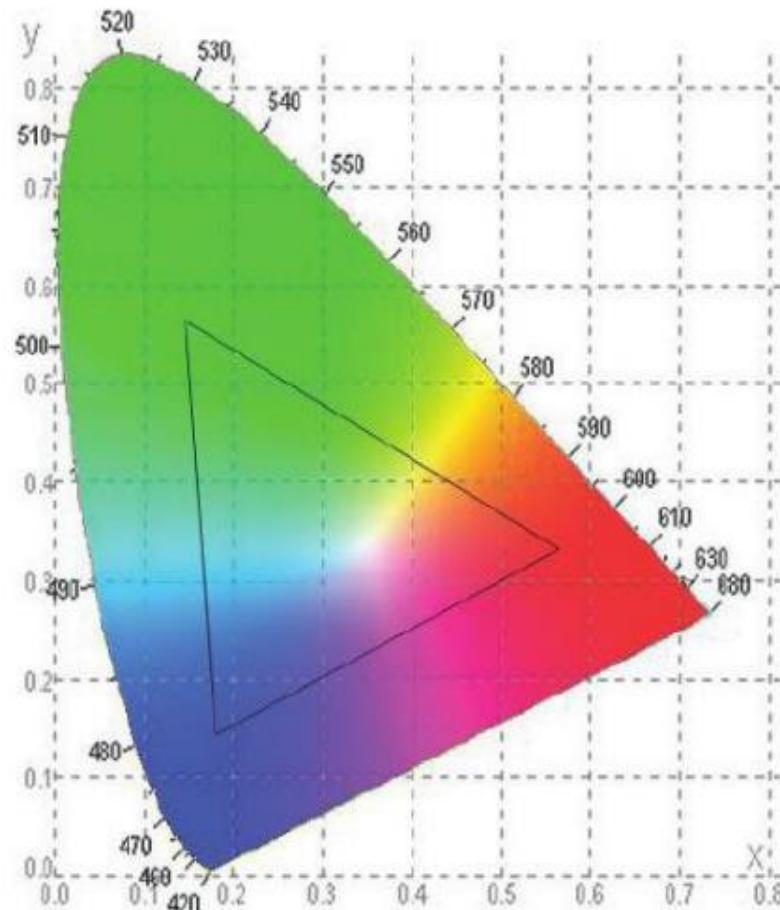
# KONVERSI RGB TO YIQ

```
1 import cv2
2 import numpy as np
3
4 img1=cv2.imread('parrot.jpg')
5 cv2.imshow('Citra Input',img1)
6 img=cv2.imread('parrot.jpg')
7
8 img[:, :, 0]= 0.3*img[:, :, 0]+0.59*img[:, :, 1]+0.11*img[:, :, 2]
9 img[:, :, 1]= -0.6*img[:, :, 0]-0.28*img[:, :, 1]-0.32*img[:, :, 2]
10 img[:, :, 2]= 0.21*img[:, :, 0]-0.52*img[:, :, 1]+0.31*img[:, :, 2]
11
12 cv2.imshow('Hasil YIQ',img)
13 cv2.waitKey(0)
14 cv2.destroyAllWindows()
```



# Ruang Warna CIELAB

- CIELAB adalah nama lain dari CIE L\*a\*b\*. Diagram kromasitas CIE (Commission Internationale de l'Eclairage)
- Setiap perpaduan x dan y menyatakan suatu warna. Namun, hanya warna yang berada dalam area ladam (tapal kuda) yang bisa terlihat.
- Angka yang berada di tepi menyatakan panjang gelombang cahaya.
- Warna yang terletak di dalam segitiga menyatakan warna-warna umum di monitor CRT, yang dapat dihasilkan oleh komponen warna merah, hijau, dan biru



Gambar 9.7 Diagram kromasitas CIE  
(Sumber: Russ, 2011)

# Konversi RGB to CIELAB

$$X = 0,412453R + 0,357580G + 0,180423B$$

$$Y = 0,212671R + 0,715160G + 0,072169B$$

$$Z = 0,019334R + 0,119193G + 0,950227B$$

Selanjutnya,  $L^*a^*b^*$  didefinisikan sebagai berikut:

$$L^* = 116f\left(\frac{Y}{Y_n}\right) - 16$$

$$a^* = 500 \left[ f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right]$$

$$b^* = 200 \left[ f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right]$$

Dalam hal ini,  $f(q)$  dihitung seperti berikut:

$$f(q) = \begin{cases} q^{\frac{1}{3}}, & \text{jika } q > 0,008856 \\ 7,787q + 16/116, & \text{untuk yang lain} \end{cases}$$

$X_n, Y_n, Z_n$  diperoleh melalui  $R=G=B=1$  dengan jangkauan  $R, G, B$  berupa  $[0, 1]$ .

## Manualisasi RGB to L\*a\*b\*

- Ruang warna L\*a\*b\* mampu menggambarkan semua warna yang dapat dilihat oleh mata manusia dan seringkali digunakan sebagai referensi ruang warna.
- Perhitungan konversi ruang warna L\*a\*B\* dapat dilakukan melalui ruang warna XYZ

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.412452 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

## Manualisasi RGB to L\*a\*b\*

- Dari matrik RGB sebelumnya dapat digitung RGB to XYZ sbb:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.412452 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

$$\begin{aligned} X &= (\text{red} * 0.412453) + (\text{green} * 0.357580) + (\text{blue} * 0.180423) \\ &= (255 * 0.412453) + (124 * 0.357580) + (168 * 0.180423) \\ &= 179.83 \end{aligned}$$

$$\begin{aligned} Y &= (\text{red} * 0.212671) + (\text{green} * 0.715160) + (\text{blue} * 0.072169) \\ &= (255 * 0.212671) + (124 * 0.715160) + (168 * 0.072169) \\ &= 155.04 \end{aligned}$$

$$\begin{aligned} Z &= (\text{red} * 0.019334) + (\text{green} * 0.119193) + (\text{blue} * 0.950227) \\ &= (255 * 0.019334) + (124 * 0.119193) + (168 * 0.950227) \\ &= 179.35 \end{aligned}$$

Hasil Perbaikan Nilai Rescaling

R			G			B		
255	255	255	124	128	131	168	172	175
255	55	255	121	73	128	165	30	172
58	62	56	83	54	76	40	36	33
255	57	255	118	79	123	162	36	167

## Manualisasi RGB to L\*a\*b\*

- Dari matrik RGB sebelumnya dapat digitung RGB to XYZ sbb:

Hasil Perbaikan Nilai Rescaling

R			G			B		
255	255	255	124	128	131	168	172	175
255	55	255	121	73	128	165	30	172
58	62	56	83	54	76	40	36	33
255	57	255	118	79	123	162	36	167



Hasil Ruang Warna XYZ

X		
179,83	181,98	183,59
178,21	54,20	181,98
60,82	51,38	56,23
176,60	58,25	179,29

Y		
155,04	158,18	160,55
152,67	66,07	158,18
74,58	54,40	68,64
150,31	71,22	154,25

Z		
179,35	183,63	186,83
176,14	38,27	183,63
49,02	41,84	41,50
172,93	44,73	178,28

## Manualisasi RGB to L\*a\*b\*

- Ruang warna L\*a\*b\*, diperlukan membagi masing-masing nilai ruang warna XYZ dengan masing-masing white reference-nya.
- White reference adalah nilai maximal daripada masing-masing penyusun ruang warna XYZ disimbolkan dengan X<sub>0</sub>, Y<sub>0</sub> dan Z<sub>0</sub>.
- X<sub>0</sub> bernilai 242.36628, Y<sub>0</sub> bernilai 255, dan Z<sub>0</sub> bernilai 277,63227 sehingga rentan nilai dengan pembagian tersebut pada 0 sampai dengan 1.

$$\frac{X}{X_0} = \frac{179.83}{242.36628} = 0.74 \quad \frac{Y}{Y_0} = \frac{155.04}{255} = 0.61 \quad \frac{Z}{Z_0} = \frac{179.35}{277,63227} = 0.65$$

- CIE standard illuminance D65 [ 0.9504, 1.0000, 1.088]
- Contoh (255\*0.9504 = 242,36)

## Manualisasi RGB to L\*a\*b\*

- Hasil perhitungan keseluruhan adalah sbb:

Hasil Pembagian XYZ Dengan *White Reference*

X/X <sub>0</sub>		
0,74	0,75	0,76
0,74	0,22	0,75
0,25	0,21	0,23
0,73	0,24	0,74

Y/Y <sub>0</sub>		
0,61	0,62	0,63
0,60	0,26	0,62
0,29	0,21	0,27
0,59	0,28	0,60

Z/Z <sub>0</sub>		
0,65	0,66	0,67
0,63	0,14	0,66
0,18	0,15	0,15
0,62	0,16	0,64

## Manualisasi RGB to L\*a\*b\*

- Menghitung fungsi yang disimbolkan dengan  $f(X/X_0)$ ,  $f(Y/Y_0)$ , dan  $f(Z/Z_0)$ .

$$f(w) = w^{1/3} \quad \text{for } w > 0.008856$$

$$f(w) = 7.787(w) + 0.1379 \quad \text{for } 0.0 \leq w \leq 0.008856$$

$X/X_0 = 0.74$  (masuk kondisi pertama)

$$f(X/X_0) = (X/X_0)^{1/3}$$

$$w = X/X_0 = 0.74^{1/3} = 0.91$$

$Y/Y_0 = 0.61$  (masuk kondisi pertama)

$$f(Y/Y_0) = (Y/Y_0)^{1/3}$$

$$w = Y/Y_0 = 0.61^{1/3} = 0.85$$

$Z/Z_0 = 0.65$  (masuk kondisi kedua)

$$f(Z/Z_0) = (Z/Z_0)^{1/3}$$

$$w = Z/Z_0 = 0.65^{1/3} = 0.86$$

## CONTOH APLIKASI

# Manualisasi RGB to

- Hasil perhitungan keseluruhan fungsi adalah sbb:

**L\*a\*b\***

Perhitungan Fungsi Terhadap XYZ

f(X/X0)		
0,91	0,91	0,91
0,90	0,61	0,91
0,63	0,60	0,61
0,90	0,62	0,90

f(Y/Y0)		
0,85	0,85	0,86
0,84	0,64	0,85
0,66	0,60	0,65
0,84	0,65	0,85

f(Z/Z0)		
0,86	0,87	0,88
0,86	0,52	0,87
0,56	0,53	0,53
0,85	0,54	0,86

## Manualisasi RGB to L\*a\*b\*

- Tahap akhir perhitungan XYZ menjadi L\*a\*b\*

$$L^* = 116\left(\frac{Y}{Y_0}\right)^{1/3} - 16 \quad \text{for } \frac{Y}{Y_0} > 0.008856$$

$$L^* = 903.3 \frac{Y}{Y_0} \quad \text{for } 0.0 \leq \frac{Y}{Y_0} \leq 0.008856$$

$Y/Y_0 = 0.91$  (masuk kondisi pertama)

$$\begin{aligned} L^* &= 116(Y/Y_0)^{1/3} - 16 \\ &= 116(0.91)^{1/3} - 16 \\ &= 82.27 \end{aligned}$$

## Manualisasi RGB to L\*a\*b\*

- Tahap akhir perhitungan XYZ menjadi L\*a\*b\*

$$\begin{aligned}a^* &= 500 (f(X/X_0) - f(Y/Y_0)) \\&= 500(0.91 - 0.85) \\&= 29.07\end{aligned}$$

$$\begin{aligned}b^* &= 200 (f(Y/Y_0) - f(Z/Z_0)) \\&= 200(0.85 - 0.86) \\&= -3.46\end{aligned}$$

- Hasil perhitungan keseluruhan adalah sbb:

Hasil Ruang Warna L\*a\*b\*

L*		
82,27	82,93	83,42
81,77	57,95	82,93
61,00	53,31	58,90
81,26	59,82	82,10

a*		
29,07	28,02	27,25
29,88	-15,26	28,02
-16,52	-0,64	-15,61
30,70	-15,95	29,34

b*		
-3,46	-3,68	-3,85
-3,29	24,19	-3,68
20,55	13,07	23,00
-3,11	21,91	-3,40

TERIMAKASIH