



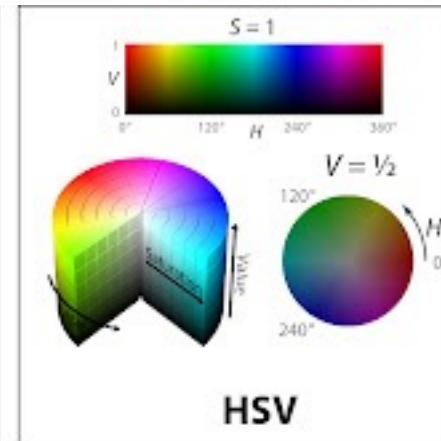
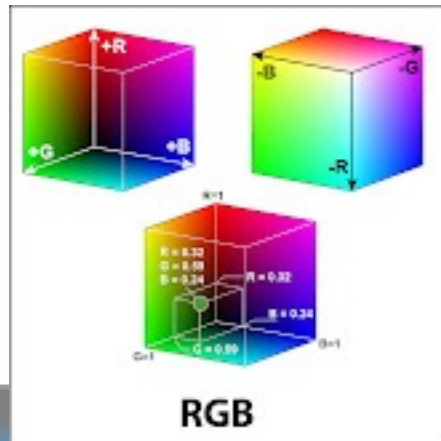
Computación Gráfica

MG. R. Jesús Cárdenas Talavera

Espacios de color

Espacios de color

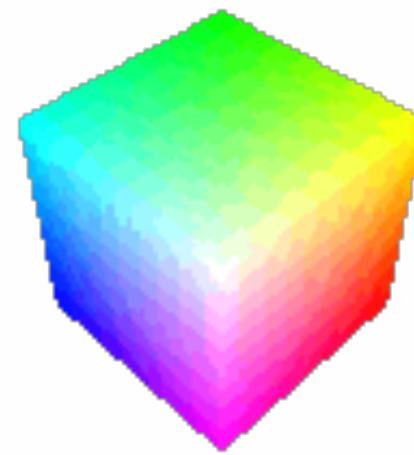
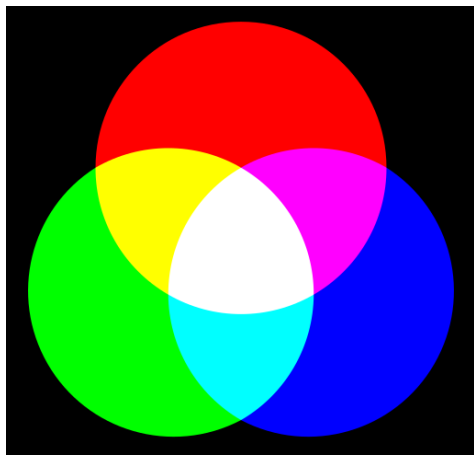
- Un modelo o espacio de color define la manera en que un color se define
- Cada espacio define su característica a separar como tono, saturación, iluminación, intensidad, etc.
- El espacio de color RGB trabaja con la suma de los valores de los colores **rojo**, **verde** y **azul**



RGB

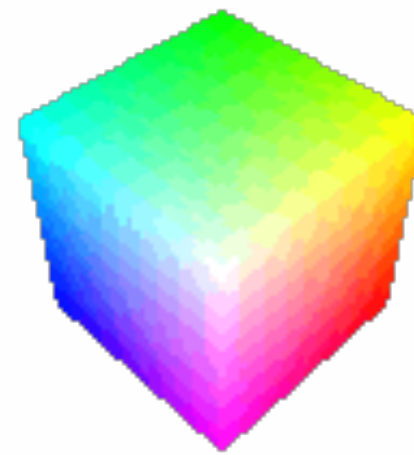
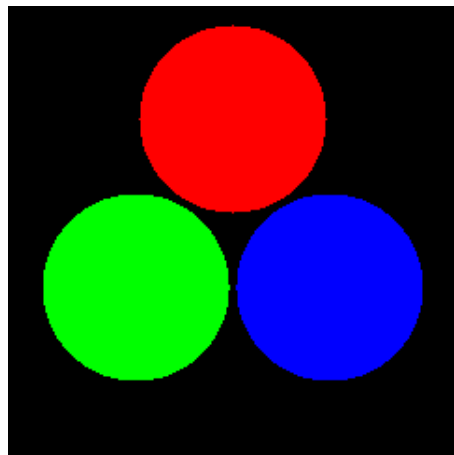
RGB

- RGB – **Red**, **Green**, **Blue**
- Modelo Genérico
- Sus colores primarios son el rojo, verde y azul
- Representa los colores mediante la *suma* de los valores de los colores primarios



RGB

- Su rango de valores va de 0 a 255
- Permite generar más de 16 millones de colores (16 777 216)
- Se puede representar todas las combinaciones como un cubo
- Cada eje coordenado representa un color, siendo el origen el color negro



RGB

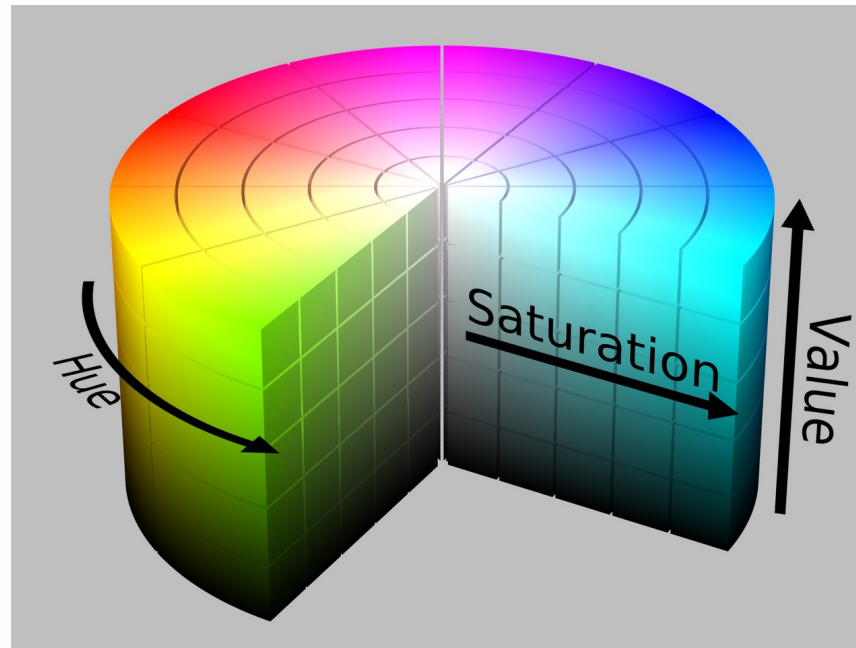
- *OpenCV* trabaja bien con RGB
- Cada imagen se representa como una matriz de 3 dimensiones
- Los canales se presentan **BGR** no como RGB
- Algunos sistemas trabajan con BGR



HSV

HSV

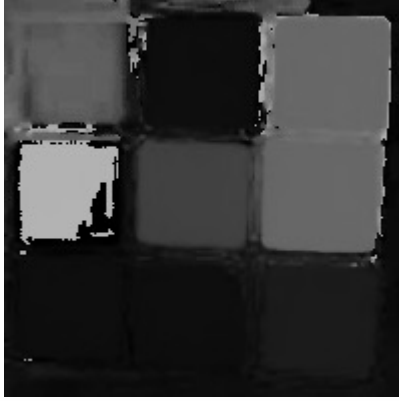
- HSV – Hue, Saturation, Value
- Matiz, Saturación y Valor
- Es un espacio de color representado de forma cilíndrica



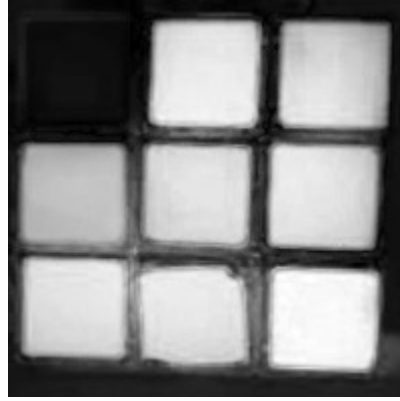
HSV



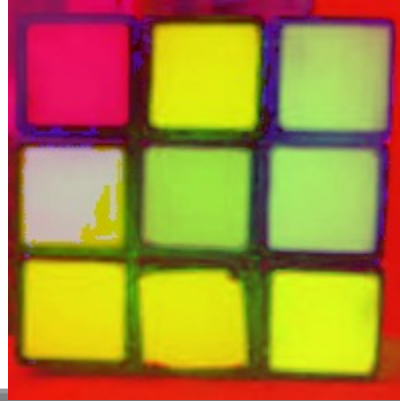
Hue



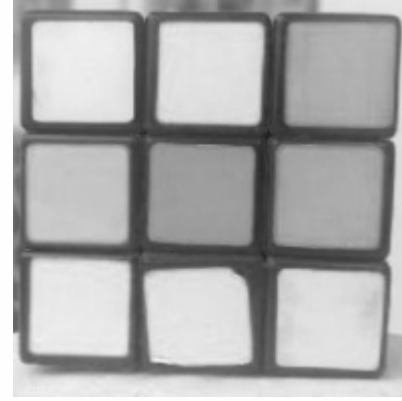
Saturation



Falso Color



Value



RGB a HSV

- Primero, se tiene que normalizar los valores RGB

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255}$$

- Se calcula el *máximo* y *mínimo* entre los valores R', G' y B'

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

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

RGB a HSV

- Calcular el valor delta Δ :

$$\Delta = C_{max} - C_{min}$$

- Cálculo de Hue:

Si $\Delta = 0, S = 0$, caso contrario:

$$\text{Si } c_{max} = R' \text{ entonces } H = 60^\circ \left(\frac{G' - B'}{\Delta \% 6} \right)$$

$$\text{Si } c_{max} = G' \text{ entonces } H = 60^\circ \left(\frac{B' - R'}{\Delta + 2} \right)$$

$$\text{Si } c_{max} = B' \text{ entonces } H = 60^\circ \left(\frac{R' - G'}{\Delta + 4} \right)$$

RGB a HSV

- Cálculo de la saturación:

$$\text{Si } \Delta = 0, S = 0$$

- Caso contrario:

$$S = \frac{\Delta}{C_{max}}$$

- Cálculo de Value:

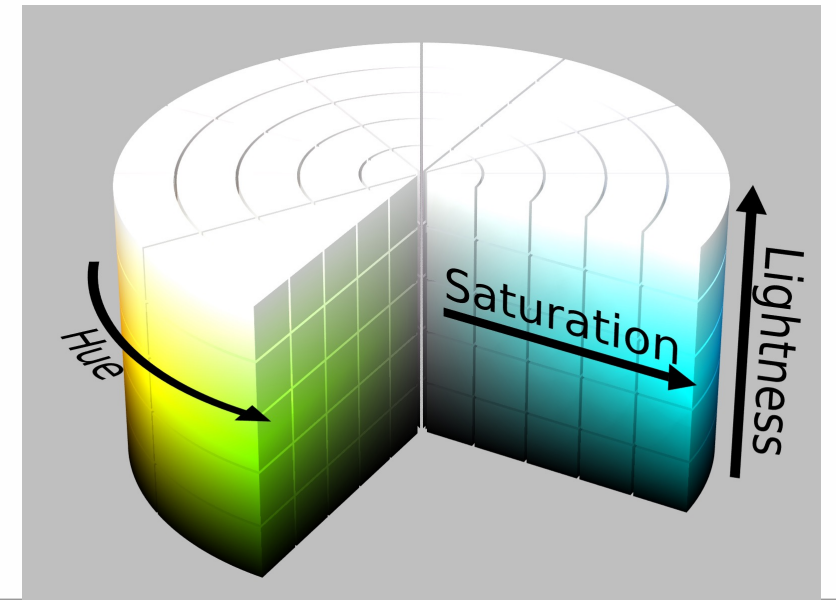
$$V = C_{max}$$

- **H : 0° a 360°, S y V: 0% a 100%**

HSL

HSL

- HSL – Hue, saturation, lightness
- Matiz, saturación, luminosidad
- Cuando la luminosidad esta en un 100%, no importa los demás valores, el color será blanco
- *OpenCV* lo maneja como HLS



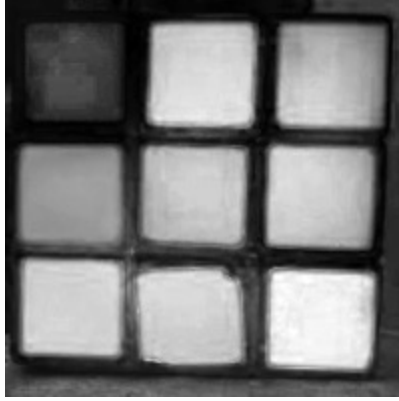
HSL



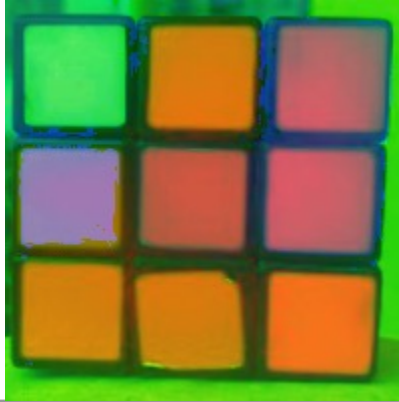
Hue



Saturation



Falso Color



Lightness



RGB a HSL

- Primero, se tiene que normalizar los valores RGB

$$R' = \frac{R}{255}, G' = \frac{G}{255}, B' = \frac{B}{255}$$

- Se calcula el *máximo* y *mínimo* entre los valores R', G' y B'

$$c_{max} = \max(R', G', B'), c_{min} = \min(R', G', B')$$

RGB a HSL

- Calcular el valor delta Δ :

$$\Delta = C_{max} - C_{min}$$

- Cálculo de Hue:

Si $\Delta = 0, S = 0$, caso contrario:

$$\text{Si } c_{max} = R' \text{ entonces } H = 60^\circ \left(\frac{G' - B'}{\Delta \% 6} \right)$$

$$\text{Si } c_{max} = G' \text{ entonces } H = 60^\circ \left(\frac{B' - R'}{\Delta + 2} \right)$$

$$\text{Si } c_{max} = B' \text{ entonces } H = 60^\circ \left(\frac{R' - G'}{\Delta + 4} \right)$$

RGB a HSL

- Cálculo de la saturación:

$$\text{Si } \Delta = 0, S = 0$$

- Caso contrario:

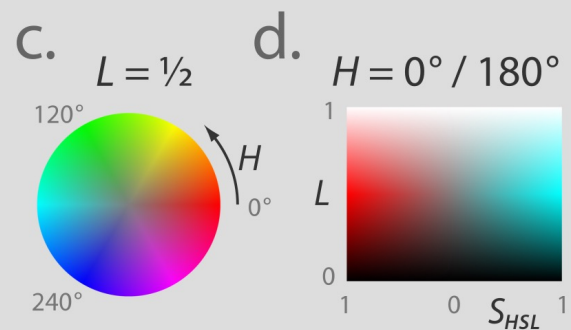
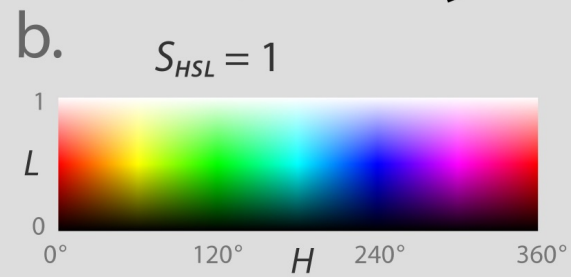
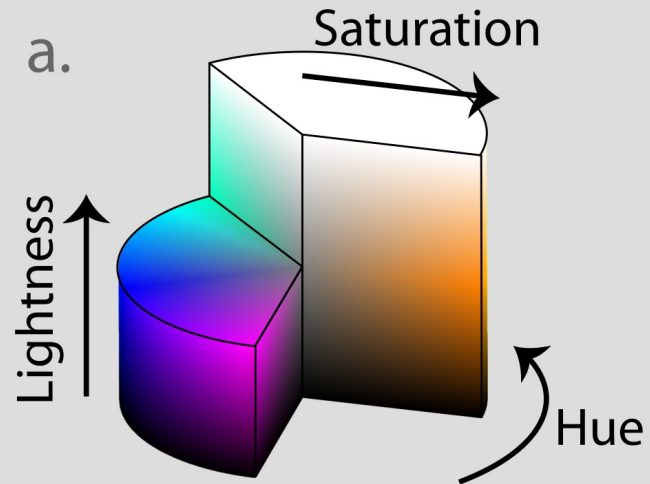
$$S = \frac{\Delta}{C_{max}}$$

- Calculo de Lightness:

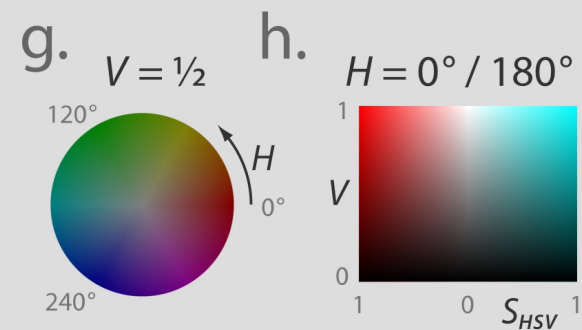
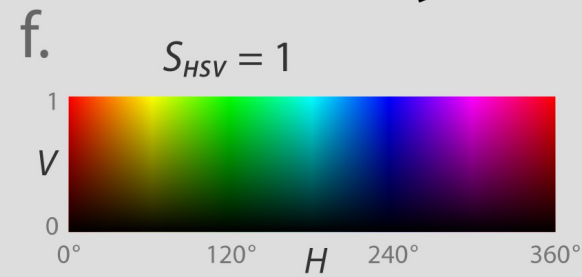
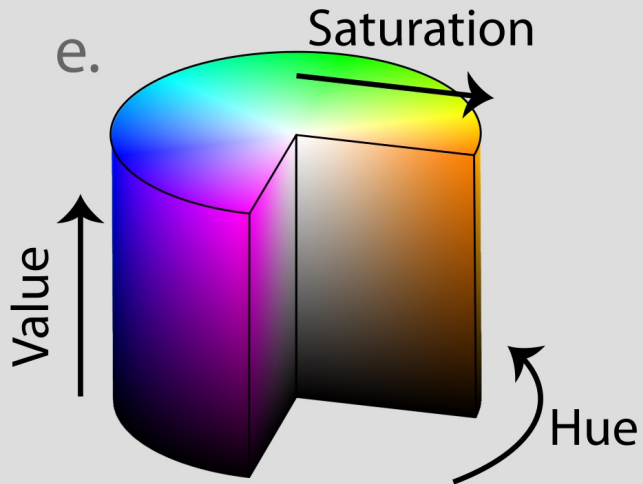
$$L = \frac{1}{2}(C_{max} + C_{min})$$

- **H : 0° a 360°, S y L: 0% a 100%**

HSL



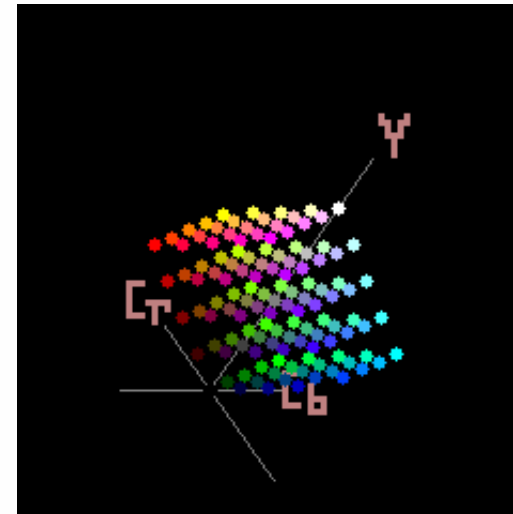
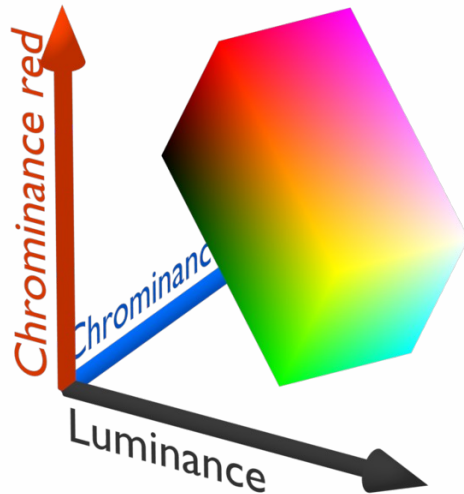
HSV



YCrCB

YCrCb

- YCrCb, Y'CbCr o Y Pb/Cb Pr/Cr
- YCrCb – Y : Luma, crominancia rojo y crominancia azul
- Es una forma de codificar información RGB
- Utilizado en sistemas de fotografía digital



YCrCB



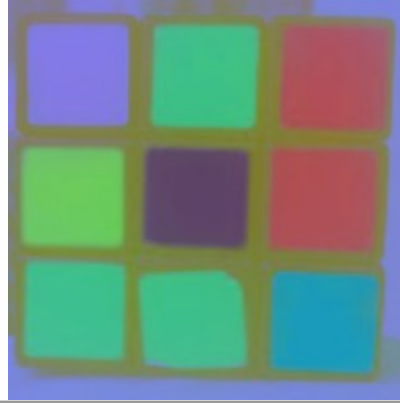
Luma



Cr



Falso Color



Cb



RGB a YCrCb

- *OpenCV* utiliza la siguiente fórmula:

- Para el cálculo de Y:

$$Y = 0.299 * R + 0.587 * G + 0.114 * B$$

- Cálculo de Cr:

$$C_r = (R - Y) * 0.713 + \Delta$$

- Cálculo de Cb:

$$C_b = (B - Y) * 0.564 + \Delta$$

RGB a YCrCb

- Donde:

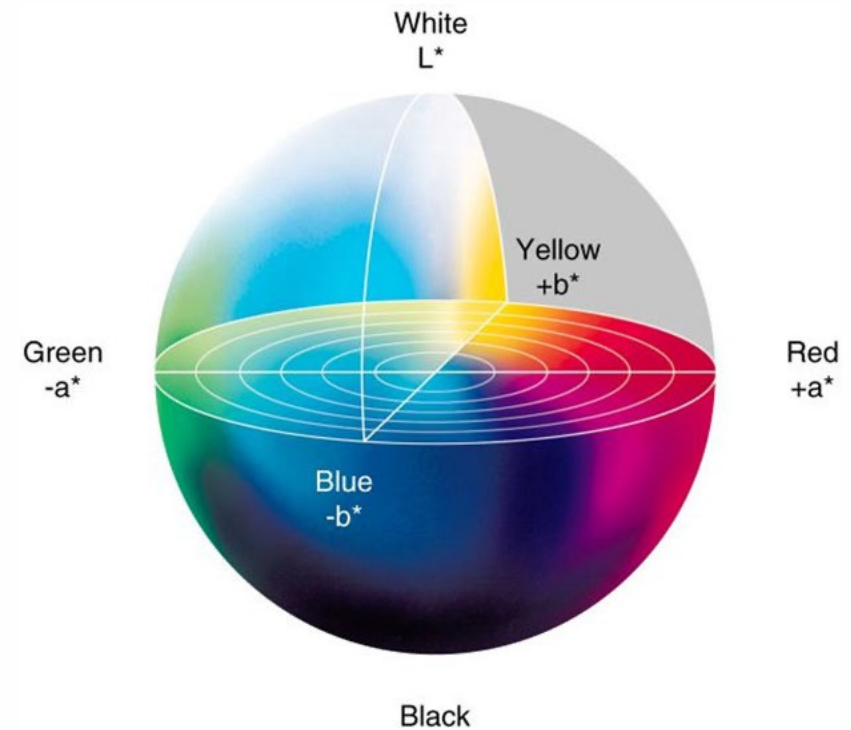
$$\Delta = \begin{cases} 128 & \text{imagenes de 8 – bit} \\ 32768 & \text{imagenes de 16 – bit} \\ 0.5 & \text{imagenes punto – flotante} \end{cases}$$

- La fórmula puede variar!

Lab

Lab

- CIELAB o L^*a^*b
- Expresa 3 valores: **L:** **lightness**, **a^*** y **b^*** expresan los 4 colores únicos de la visión humana
- **Rojo**, **verde**, **azul** y **amarillo**



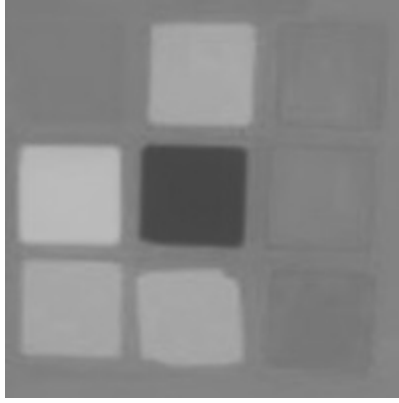
LAB



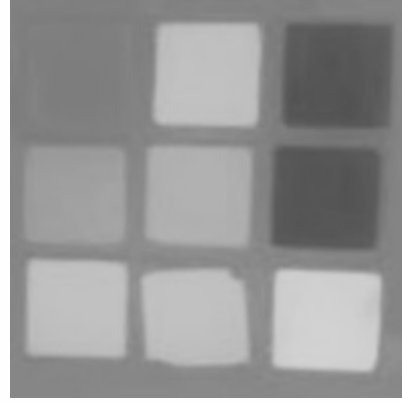
lightness



A^*



B^*



Falso Color



RGB to Lab

- Los canales RGB son convertidos en valores entre 0 y 1

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.3577580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072159 \\ 0.019334 & 0.119193 & 0.9550277 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- $X = \frac{X}{X_n}$, donde $X_n = 0.950456$
- $Z = \frac{Z}{Z_n}$, donde $Z_n = 1.088754$

RGB to Lab

$$L = \begin{cases} 116 * Y^{\frac{1}{3}} - 16 & Y > 0.008856 \\ 903 * Y & Y \leq 0.008856 \end{cases}$$

$$a = 500 * (f(X) - F(Y)) + delta$$
$$b = 200 * (f(Y) - F(Z)) + delta$$

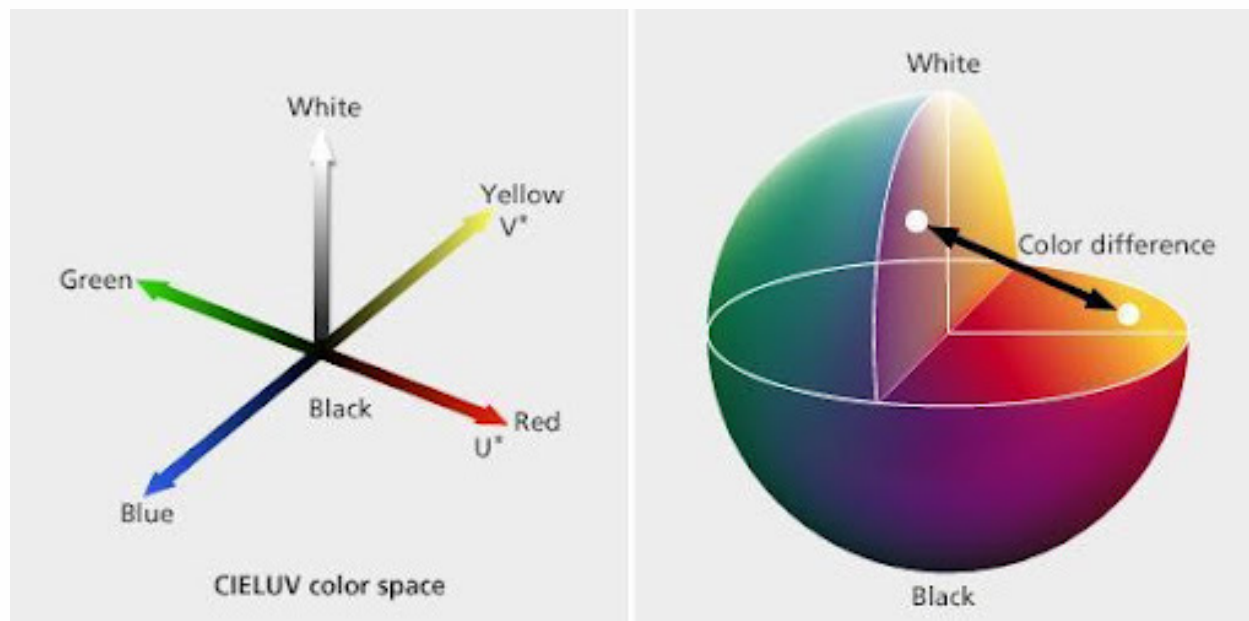
$$f(t) = \begin{cases} t^{\frac{1}{3}} & t > 0.008856 \\ 7.787t + \frac{16}{116} & t \leq 0.008856 \end{cases}$$

$$delta = \begin{cases} 128 & 8 - bit \ image \\ 0 & floating - point \end{cases}$$

LUV

LUV

- CIE $L^*u^*v^*$
- Similar a **Lab** pero más simple
- Utilizado en aplicaciones donde se tiene luces de colores



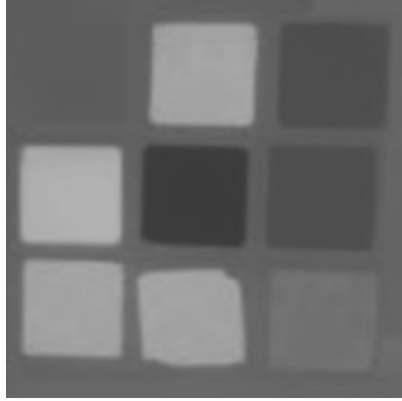
LUV



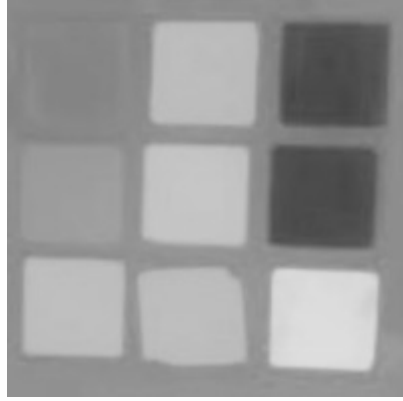
lightness



U^*



V^*



Falso Color



RGB a Luv

- Los canales RGB son convertidos en valores entre 0 y 1

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.3577580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072159 \\ 0.019334 & 0.119193 & 0.9550277 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$L = \begin{cases} 116 * Y^{\frac{1}{3}} - 16 & Y > 0.008856 \\ 903 * Y & Y \leq 0.008856 \end{cases}$$

RGB à Luv

$$u' = 4 * \frac{X}{X + 15 * Y + 3Z}$$
$$v' = 9 * \frac{Y}{X + 15 * Y + 3Z}$$

$$u = 13 * L * (u' - u_n) \text{ donde } u_n = 0.19793943$$

$$v = 13 * L * (v' - v_n) \text{ donde } v_n = 0.46831096$$

- $0 \leq L \leq 100, -134 \leq u \leq 220, -140 \leq v \leq 122$



Computación Gráfica

MG. R. Jesús Cárdenas Talavera