Computer Vision

Ch.3 Color Space

Prof. Po-Yueh Chen (陳伯岳)
E-mail: pychen@cc.ncue.edu.tw
Ext: 8440

NCUE CSIE

The computer vision (1/7)

Any ideas?

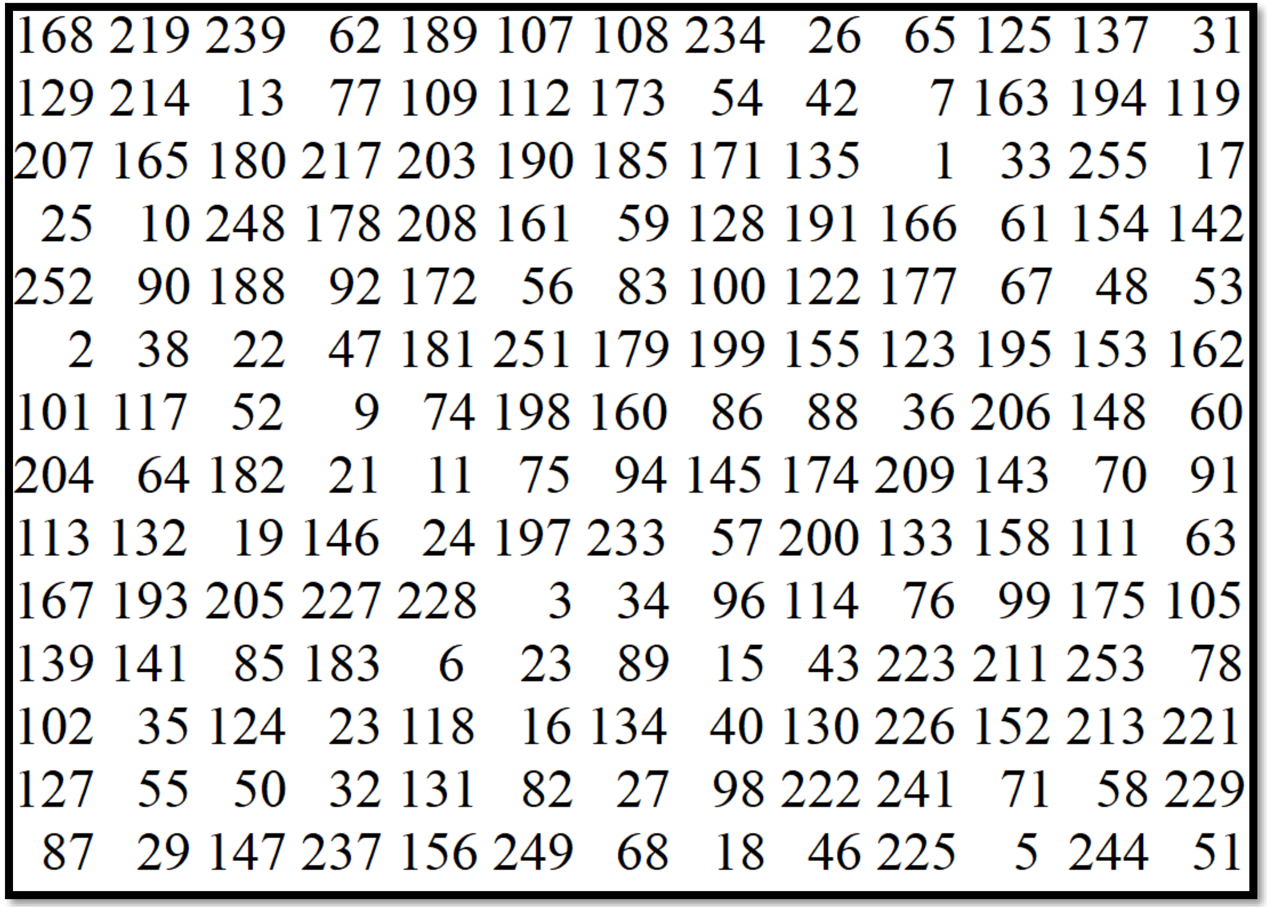
How does the computer see this world?

The computer vision (2/7)

In your eyes. . .

How beautiful she is.

The computer vision (3/7)

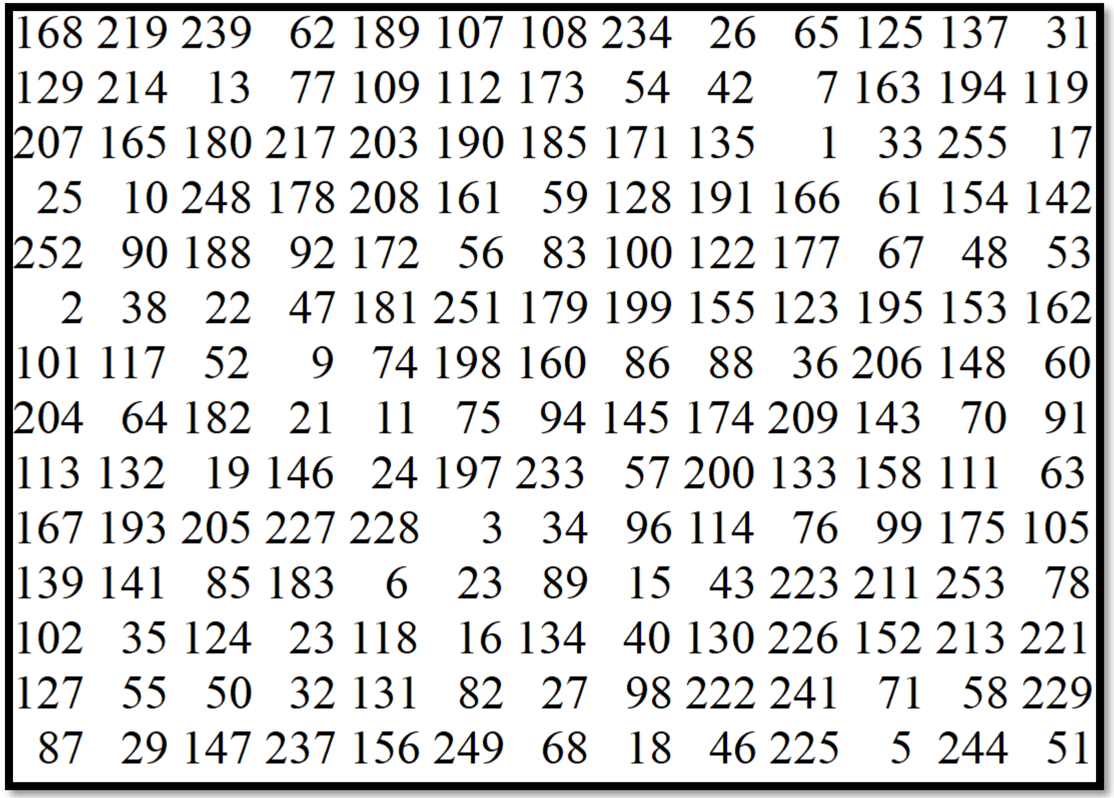


168	219	239	62	189	107	108	234	26	65	125	137	31
129	214	13	77	109	112	173	54	42	7	163	194	119
207	165	180	217	203	190	185	171	135	1	33	255	17
25	10	248	178	208	161	59	128	191	166	61	154	142
252	90	188	92	172	56	83	100	122	177	67	48	53
2	38	22	47	181	251	179	199	155	123	195	153	162
101	117	52	9	74	198	160	86	88	36	206	148	60
204	64	182	21	11	75	94	145	174	209	143	70	91
113	132	19	146	24	197	233	57	200	133	158	111	63
167	193	205	227	228	3	34	96	114	76	99	175	105
139	141	85	183	6	23	89	15	43	223	211	253	78
102	35	124	23	118	16	134	40	130	226	152	213	221
127	55	50	32	131	82	27	98	222	241	71	58	229
87	29	147	237	156	249	68	18	46	225	5	244	51

In computer. . .

She is like this…

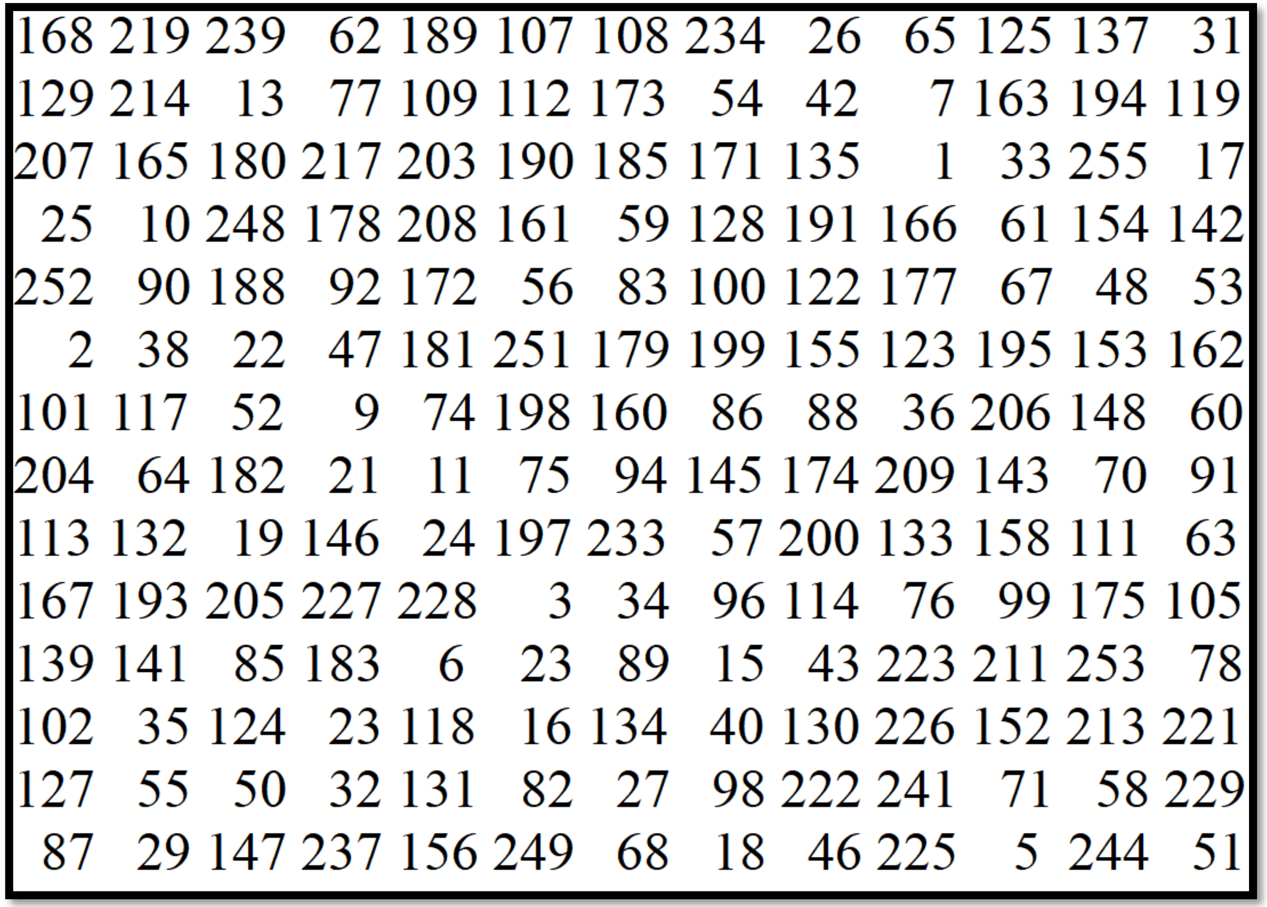
The computer vision (4/7)



168	219	239	62	189	107	108	234	26	65	125	137	31
129	214	13	77	109	112	173	54	42	7	163	194	119
207	165	180	217	203	190	185	171	135	1	33	255	17
25	10	248	178	208	161	59	128	191	166	61	154	142
252	90	188	92	172	56	83	100	122	177	67	48	53
2	38	22	47	181	251	179	199	155	123	195	153	162
101	117	52	9	74	198	160	86	88	36	206	148	60
204	64	182	21	11	75	94	145	174	209	143	70	91
113	132	19	146	24	197	233	57	200	133	158	111	63
167	193	205	227	228	3	34	96	114	76	99	175	105
139	141	85	183	6	23	89	15	43	223	211	253	78
102	35	124	23	118	16	134	40	130	226	152	213	221
127	55	50	32	131	82	27	98	222	241	71	58	229
87	29	147	237	156	249	68	18	46	225	5	244	51

✓ **By analyzing an image, how does the computer know what the object is.**

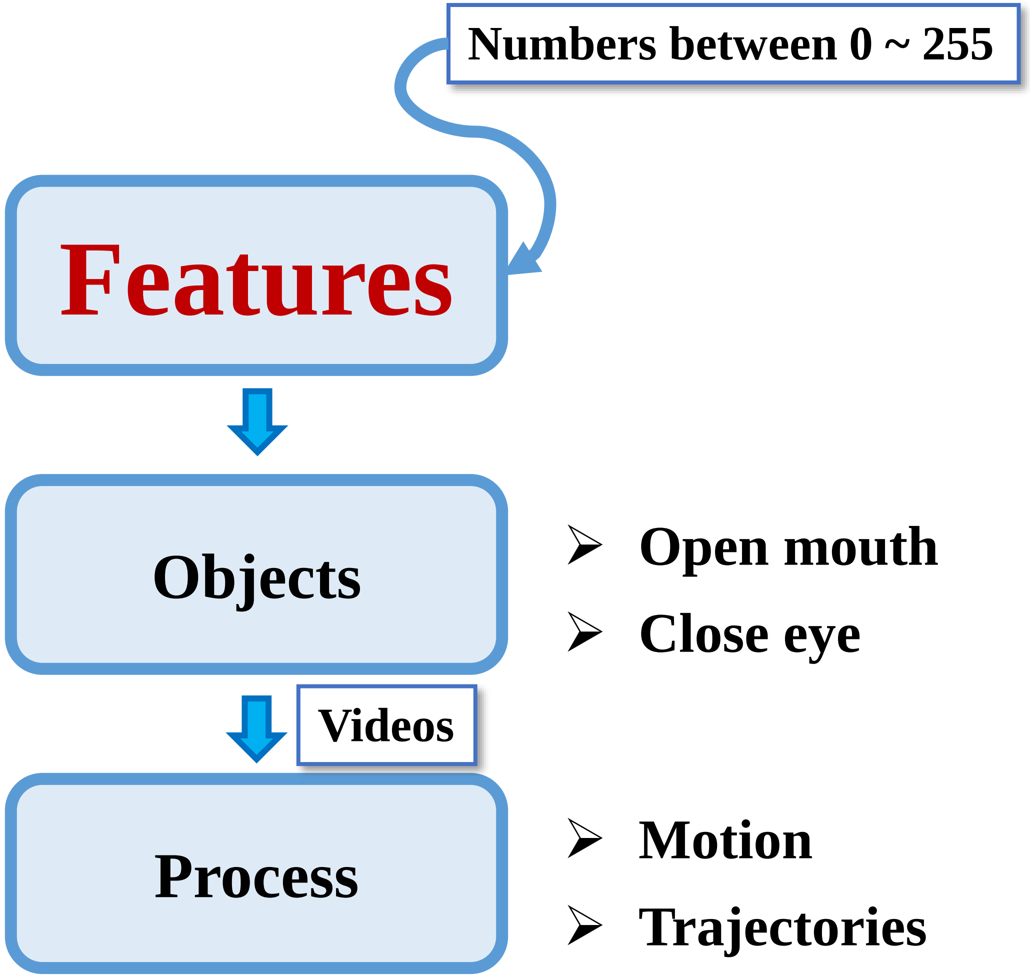
The computer vision (5/7)



168	219	239	62	189	107	108	234	26	65	125	137	31
129	214	13	77	109	112	173	54	42	7	163	194	119
207	165	180	217	203	190	185	171	135	1	33	255	17
25	10	248	178	208	161	59	128	191	166	61	154	142
252	90	188	92	172	56	83	100	122	177	67	48	53
2	38	22	47	181	251	179	199	155	123	195	153	162
101	117	52	9	74	198	160	86	88	36	206	148	60
204	64	182	21	11	75	94	145	174	209	143	70	91
113	132	19	146	24	197	233	57	200	133	158	111	63
167	193	205	227	228	3	34	96	114	76	99	175	105
139	141	85	183	6	23	89	15	43	223	211	253	78
102	35	124	23	118	16	134	40	130	226	152	213	221
127	55	50	32	131	82	27	98	222	241	71	58	229
87	29	147	237	156	249	68	18	46	225	5	244	51

✓ **An image is a big grid of numbers between 0 ~ 255**

The computer vision (6/7)



Numbers between 0 ~ 255

Features

Objects

- ➢ Open mouth
- ➢ Close eye

Videos

Process

- ➢ Motion
- ➢ Trajectories

Many algorithms are proposed to transform these numbers into features for further analysis.

The computer vision (7/7)

- **Features**

- **Colors** ⇨ ✓ ***Intensity*** of information.
- **Shape**
- **Size**
- **Motion** ⇨ *In video frames.*
- **… etc.**

Color Space (1/20)

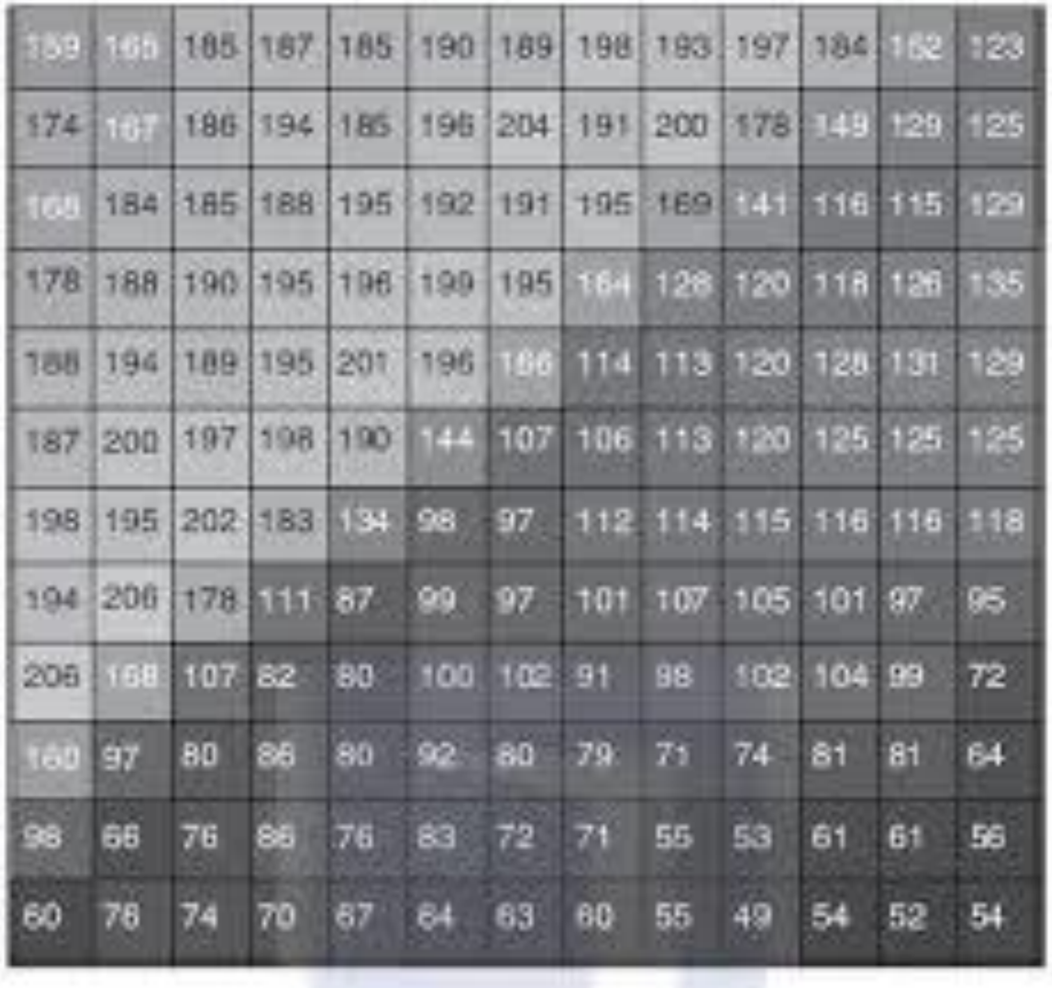
➢ Image and Matrix

An image is a long series of numbers saved in a matrix format, where each number representing the intensity of light at a particular wavelength.

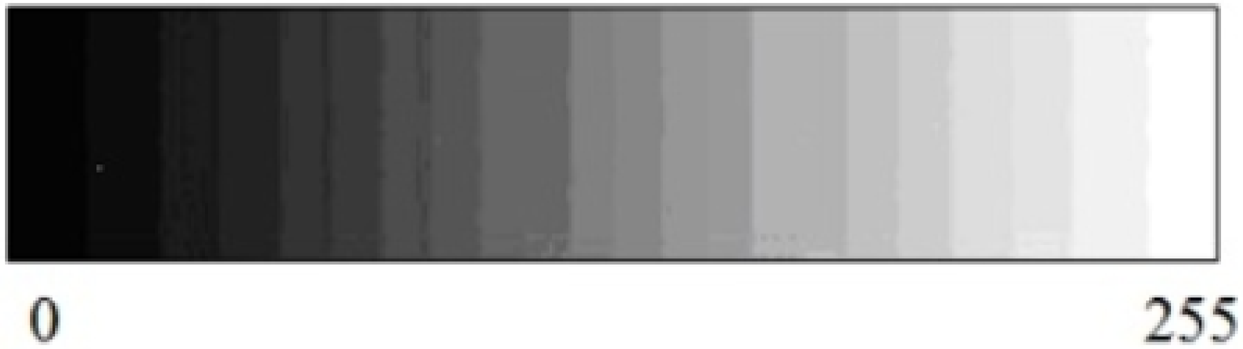
In an image, every pixel has its own position, which is identified by column and row indexes. Therefore, OpenCV defines the Mat category for representing an image and its pixels.

Color Space (2/20)

➢ Image and Matrix

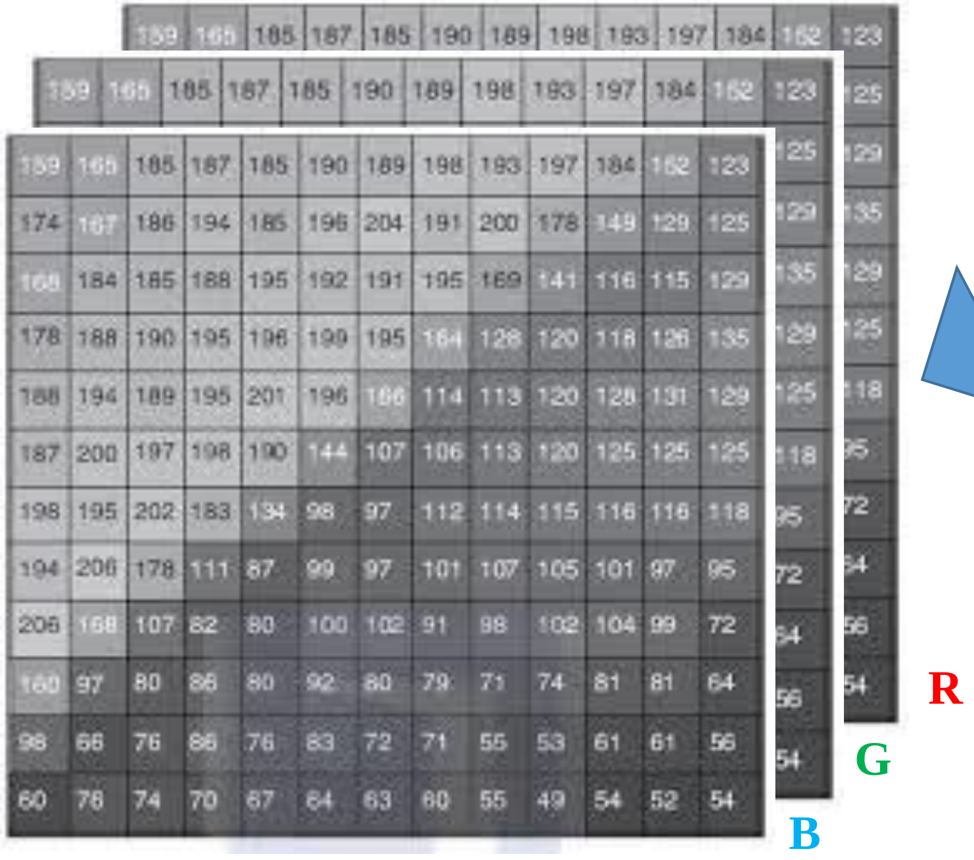


159	165	185	187	185	190	189	198	193	197	184	162	123
174	167	186	194	185	196	204	191	200	178	149	129	125
168	184	185	188	195	192	191	195	169	141	116	115	129
178	188	190	195	196	199	195	164	128	120	118	126	135
188	194	189	195	201	196	166	114	113	120	128	131	129
187	200	197	198	190	144	107	106	113	120	125	125	125
198	195	202	183	134	98	97	112	114	115	116	116	118
194	206	178	111	87	99	97	101	107	105	101	97	95
206	168	107	82	80	100	102	91	98	102	104	99	72
160	97	80	86	80	92	80	79	71	74	81	81	64
98	66	76	86	76	83	72	71	55	53	61	61	56
60	76	74	70	67	64	63	60	55	49	54	52	54



Color Space (3/20)

➢ Image and Matrix

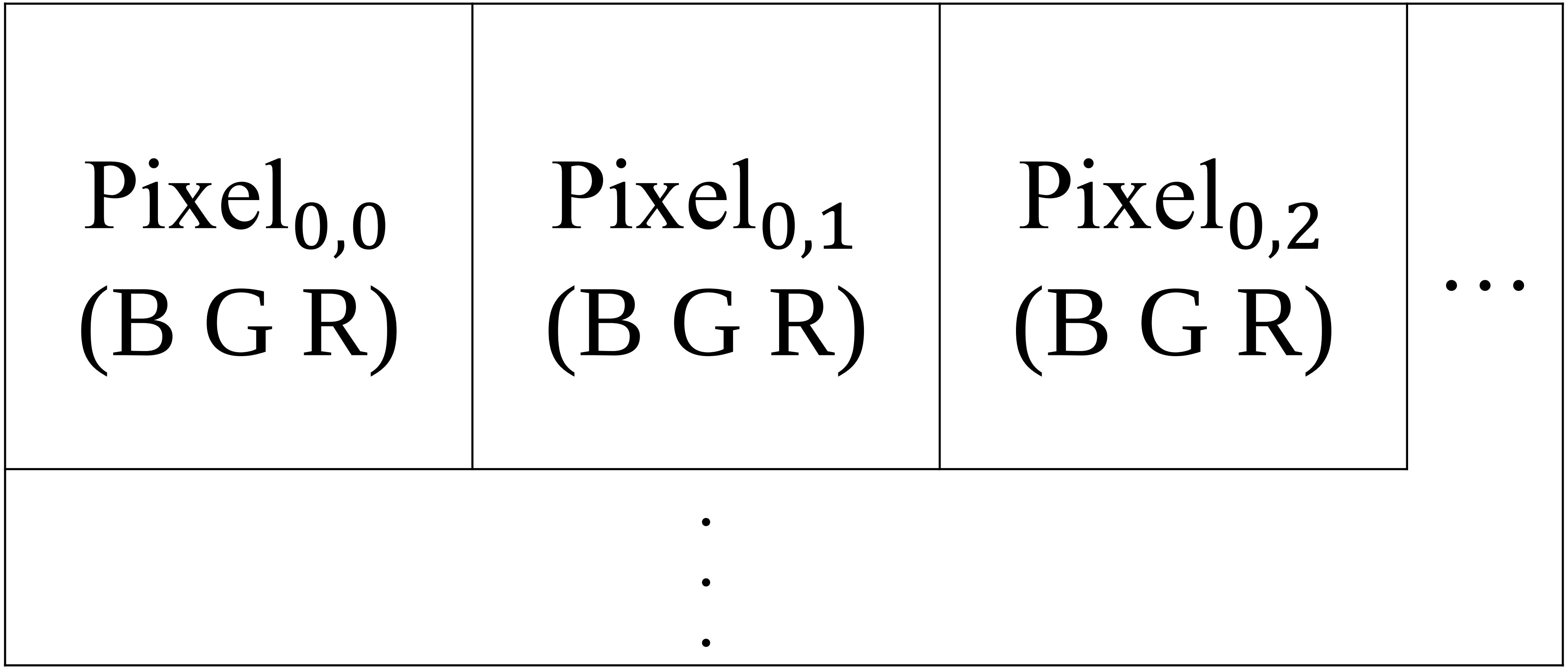


159	165	185	187	185	190	189	198	193	197	184	162	123
174	167	186	194	185	196	204	191	200	178	149	129	125
168	184	185	188	195	192	191	195	169	141	116	115	129
178	188	190	195	196	199	195	164	128	120	118	126	135
188	194	189	195	201	196	166	114	113	120	128	131	129
187	200	197	198	190	144	107	106	113	120	125	125	125
198	195	202	183	134	98	97	112	114	115	116	116	118
194	206	178	111	87	99	97	101	107	105	101	97	95
206	168	107	82	80	100	102	91	98	102	104	99	72
160	97	80	86	80	92	80	79	71	74	81	81	64
98	66	76	86	76	83	72	71	55	53	61	61	56
60	78	74	70	67	64	63	60	55	49	54	52	54

- ➢ **OpenCV sets colors order by BGR. (Not always presented as RGB)**

Color Space (4/20)

- **Image and Matrix**



$\text{Pixel}_{0,0}$ (B G R)

$\text{Pixel}_{0,1}$ (B G R)

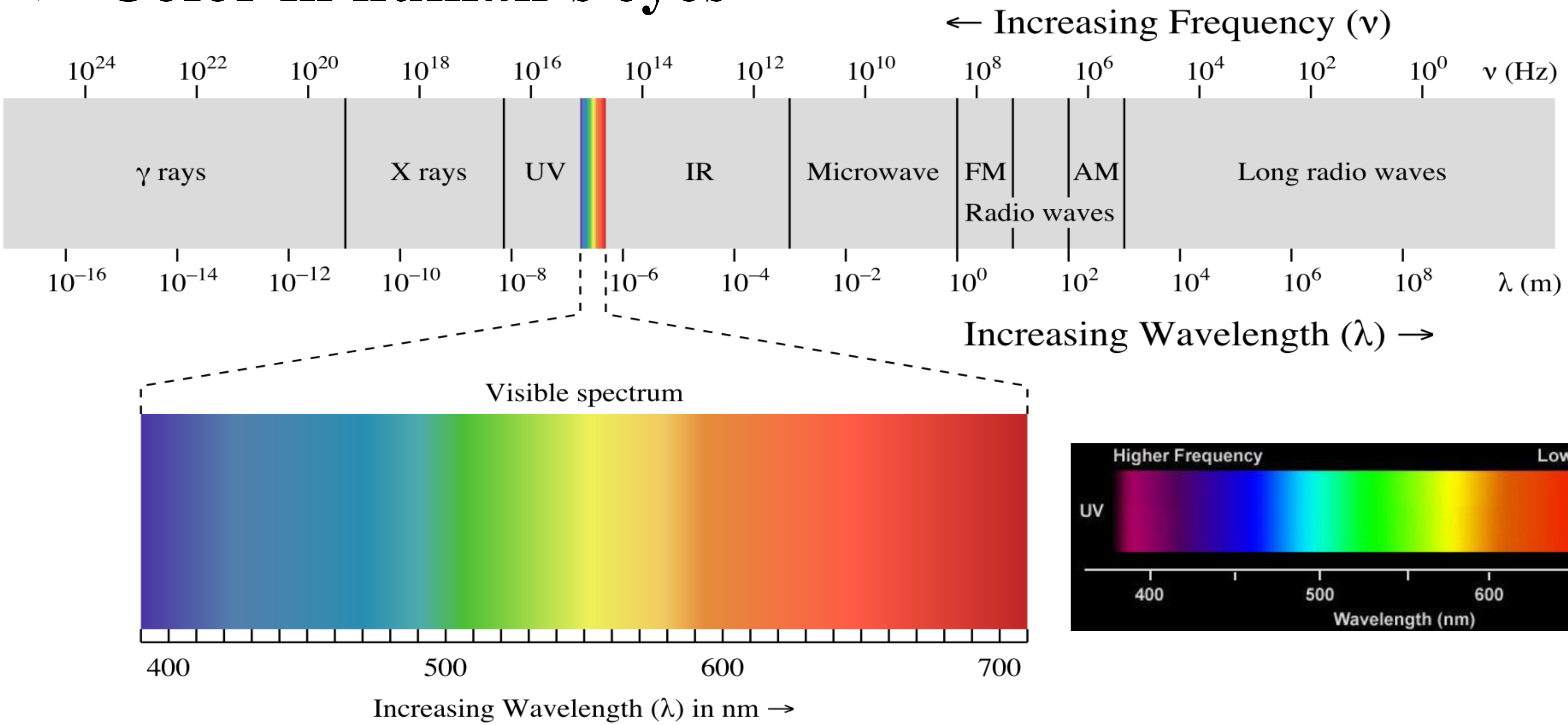
$\text{Pixel}_{0,2}$ (B G R)

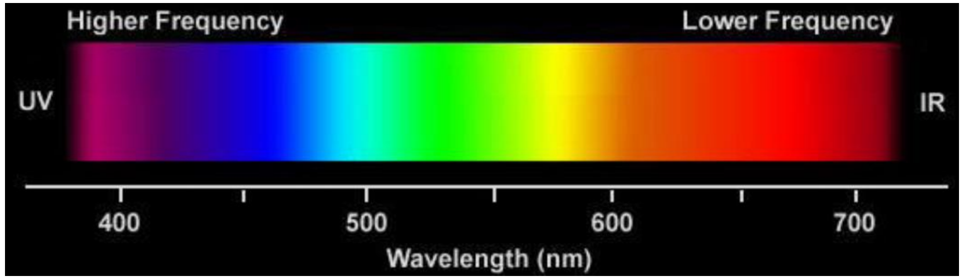
…

⋮

Color Space (5/20)

- ➢ **Color in human's eyes**

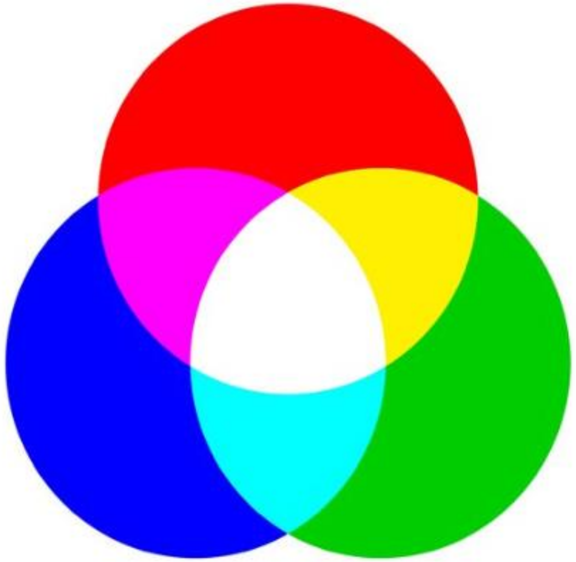




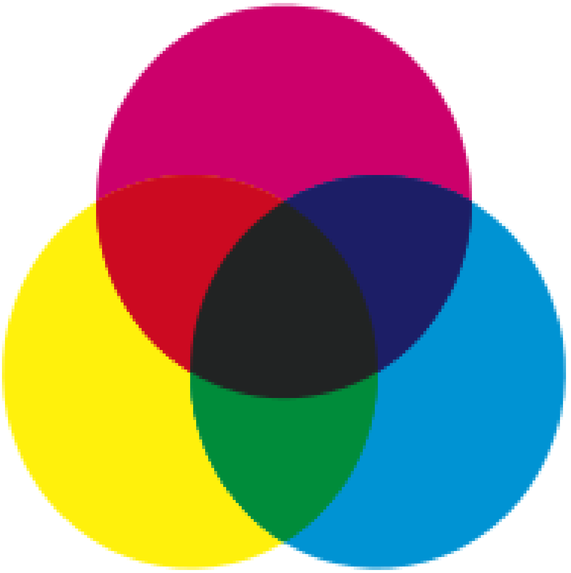
Source: https://zh.wikipedia.org/wiki/%E5%85%89#/media/File:EM_spectrum.svg

Color Space (6/20)

➢ **Commonly used color spaces**

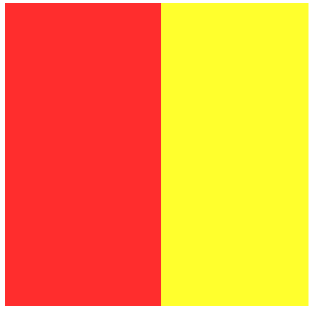


- **RGB**



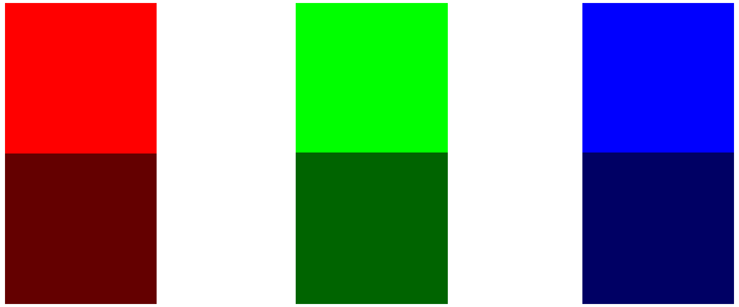
- **CMYK**

Note: ✓*Brightness* **comes mainly from red and yellow.**



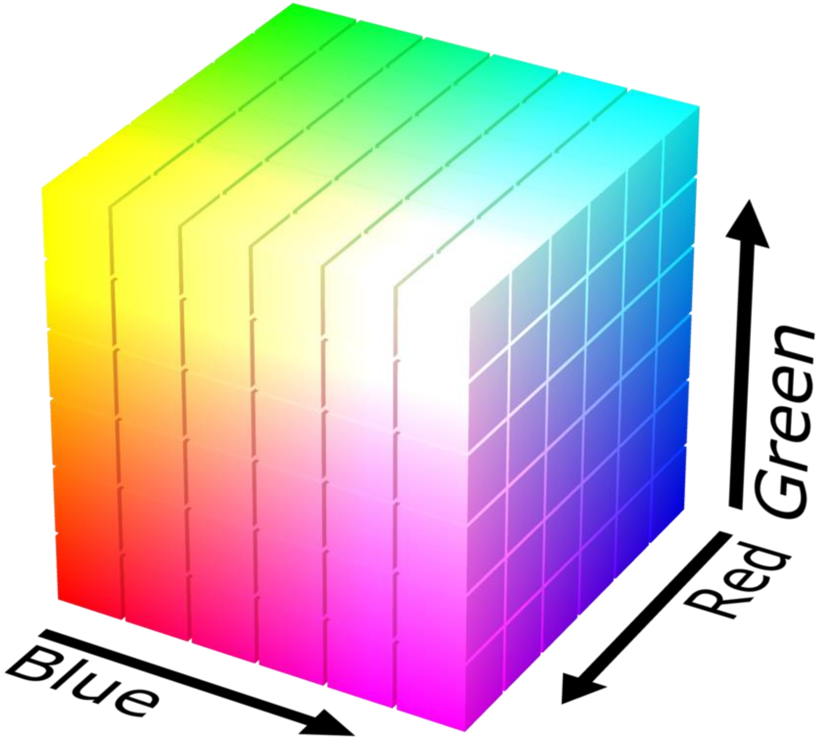
✓**Shape detection is determined by the boundary**

Using different colors and different brightness to highlight the boundary



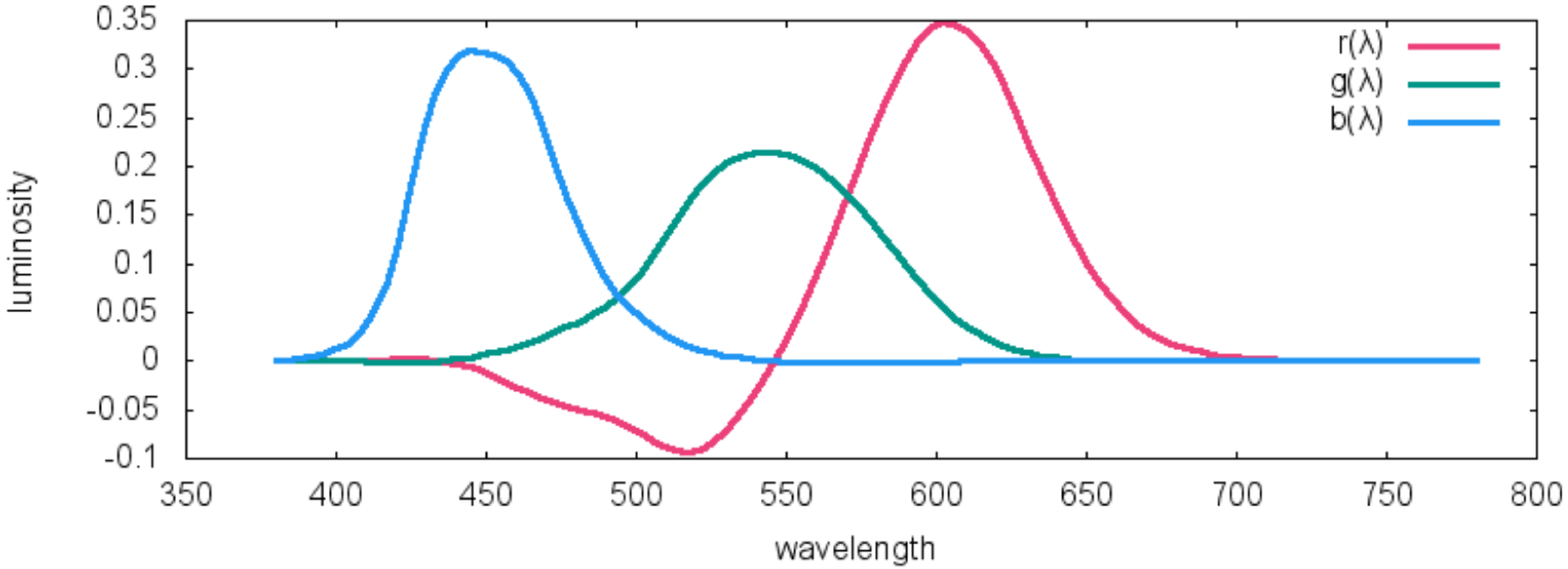
Color Space (7/20)

✓ **In a streaming digital video, color represents in 3 different channels, i.e. , Red, Green, and Blue.**



- **RGB represent by cube**

Color Space (7/20)



Source: https://medium.com/hipster-color-science/a-beginners-guide-to-colorimetry-401f1830b65a

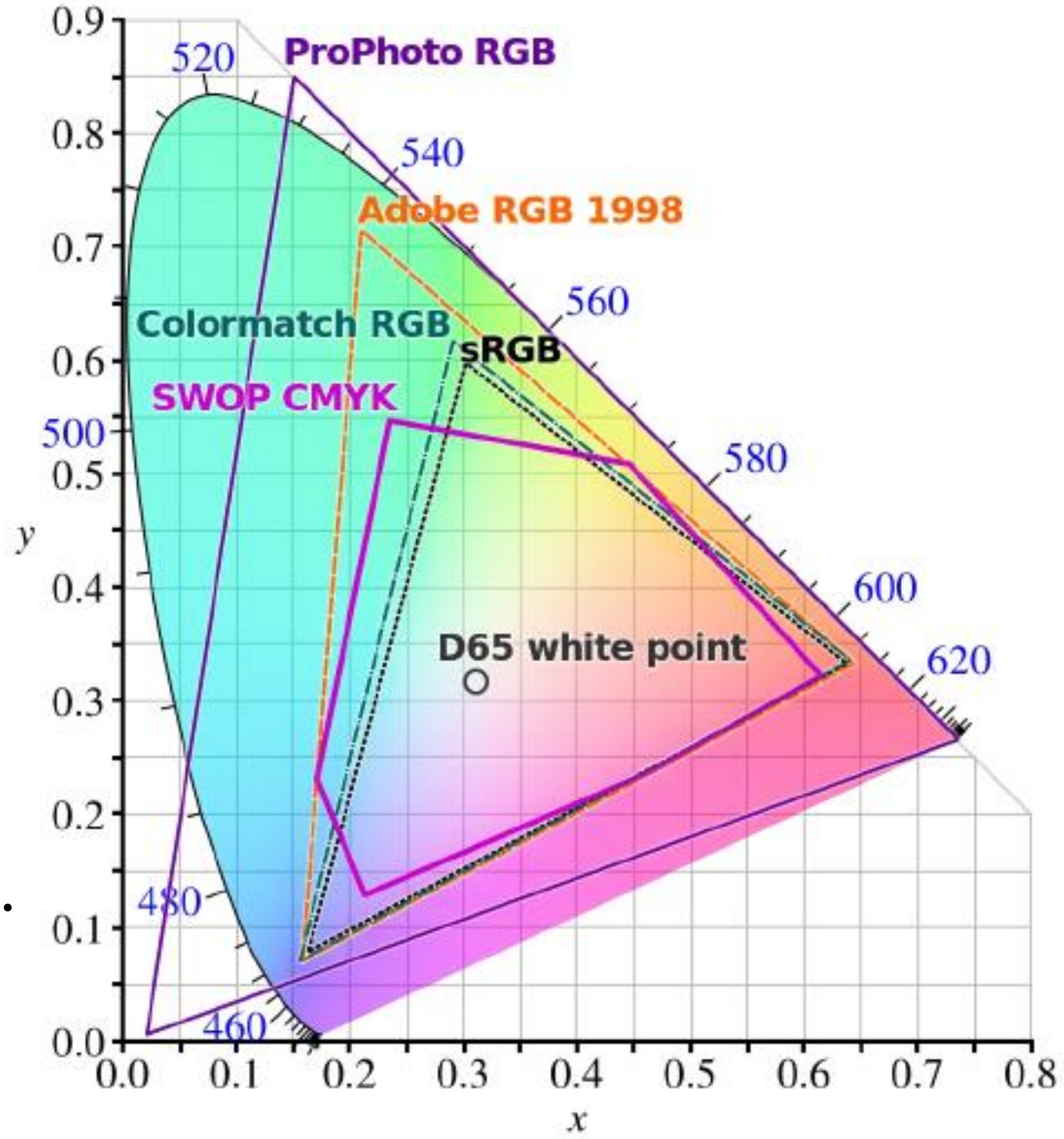
★ Color Space (8/20)

- A color model with no associated mapping function to an absolute color space is a more or less arbitrary color system with no connection to any globally understood system of color interpretation.

- The usual reference standard is the CIELAB or CIEXYZ color spaces, which were specifically designed to encompass all colors the average human can see.

✓ There are various color models used in digital video. *Ex: YCbCr, HSV, HLS, XYZ, Lab, etc.*

✓ Digital video mostly uses a matrix transform called *YCbCr*, that is closely related to *YUV*. *(RGB → YCbCr)*



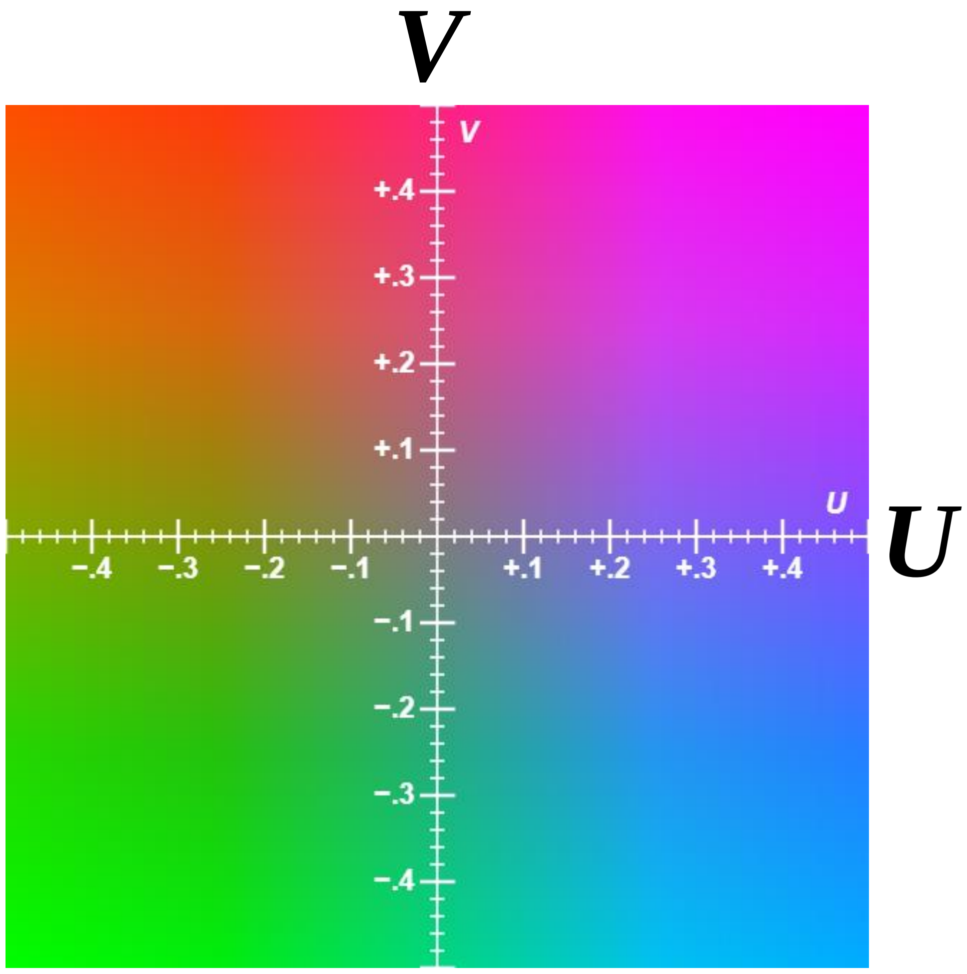
Comparison of some RGB and CMYK color gamuts on a CIE 1931xy chromaticity diagram.

Color Space (9/20)

➢ YUV Model

- ✓ **Since the eye is more sensitive to black-and-white variations, luminance is separated from color information.**

Y: Brightness Component (Luminance, Luma)
U and V: Color Components (Chrominance, Chroma)



- **YUV**

Color Space (10/20)

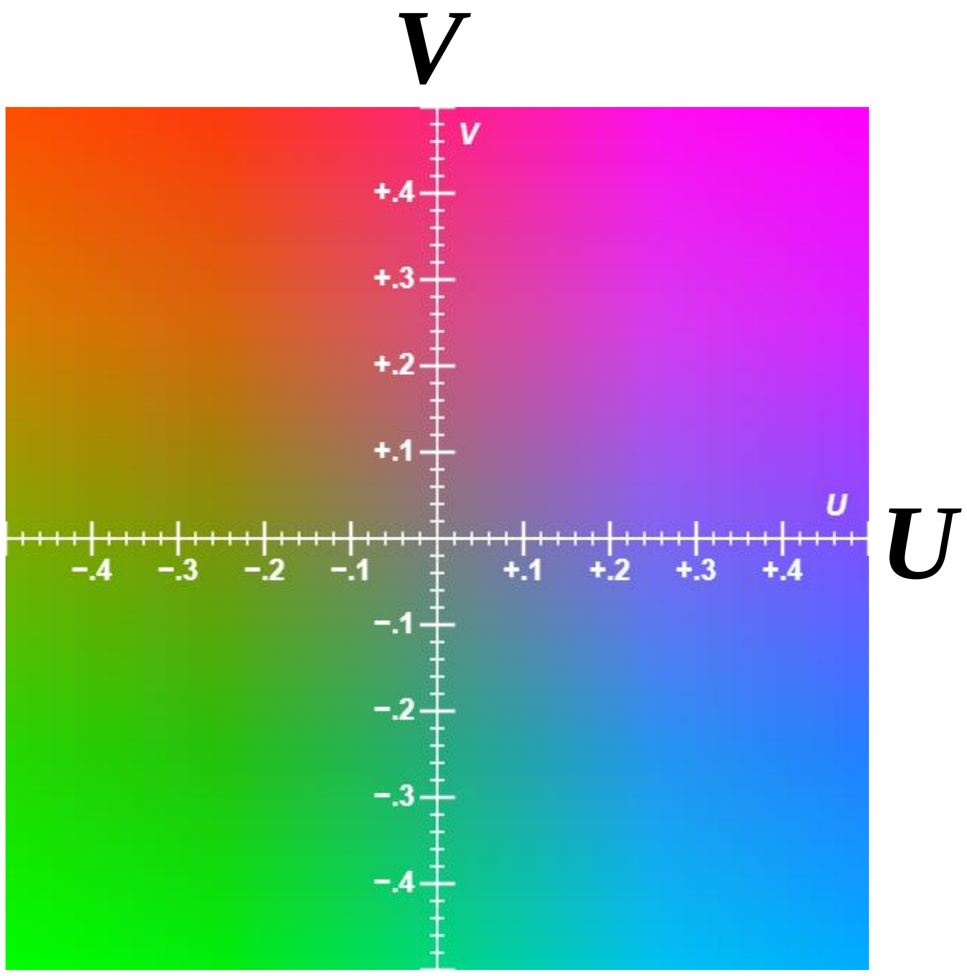
➢ YUV Model

For *U* and *V*, zero is not the minimum value.

- *U* is approximately from blue (U > 0) to yellow (U < 0).
- *V* is approximately from red (V > 0) to cyan (V < 0).

$$\begin{bmatrix} Y' \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.14713 & -0.28886 & 0.436 \\ 0.615 & -0.51499 & -0.10001 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix},$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.13983 \\ 1 & -0.39465 & -0.58060 \\ 1 & 2.03211 & 0 \end{bmatrix} \begin{bmatrix} Y' \\ U \\ V \end{bmatrix}.$$

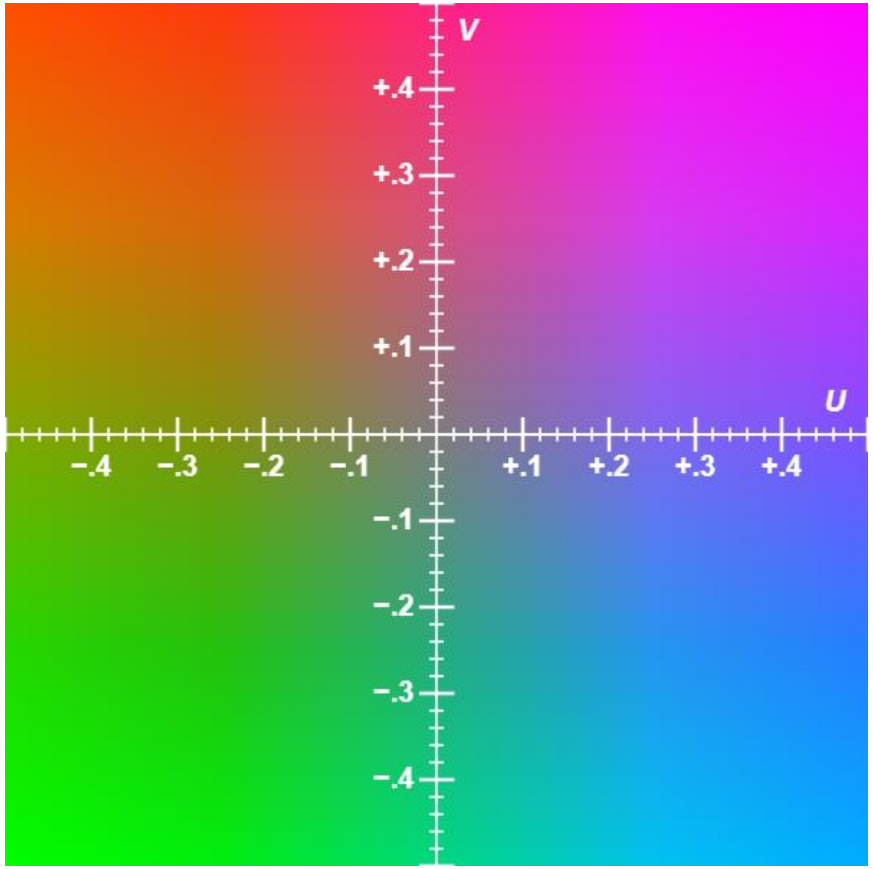


- YUV

Color Space (11/20)

➢ Example of YUV Model

Baboon

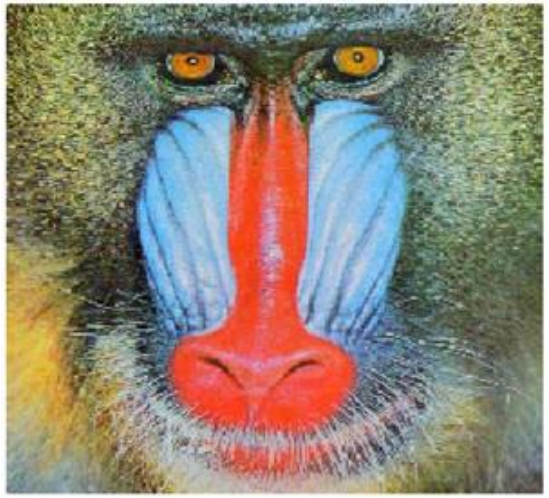


- YUV

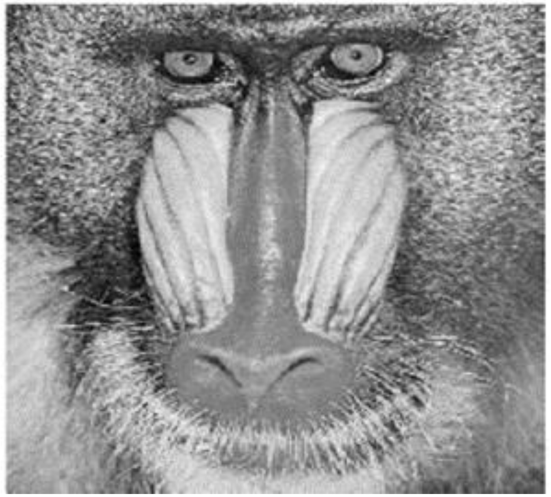
Color Space (12/20)

➢ Example of YUV Model

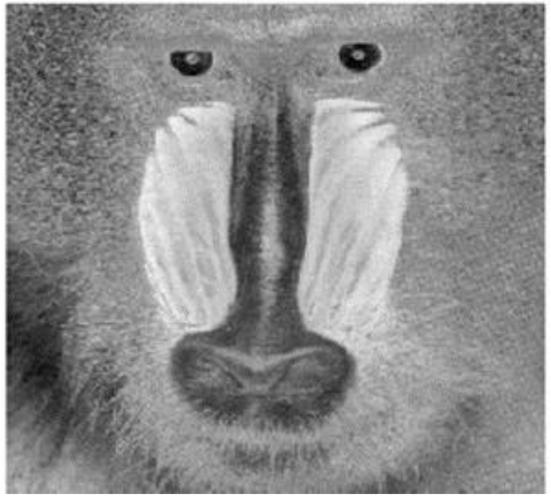
Y: Luma
U: Blue > Yellow
V: Red > Cyan



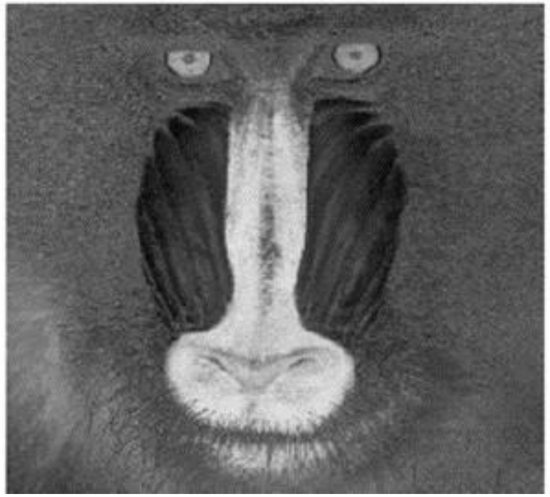
Origin



Y



U

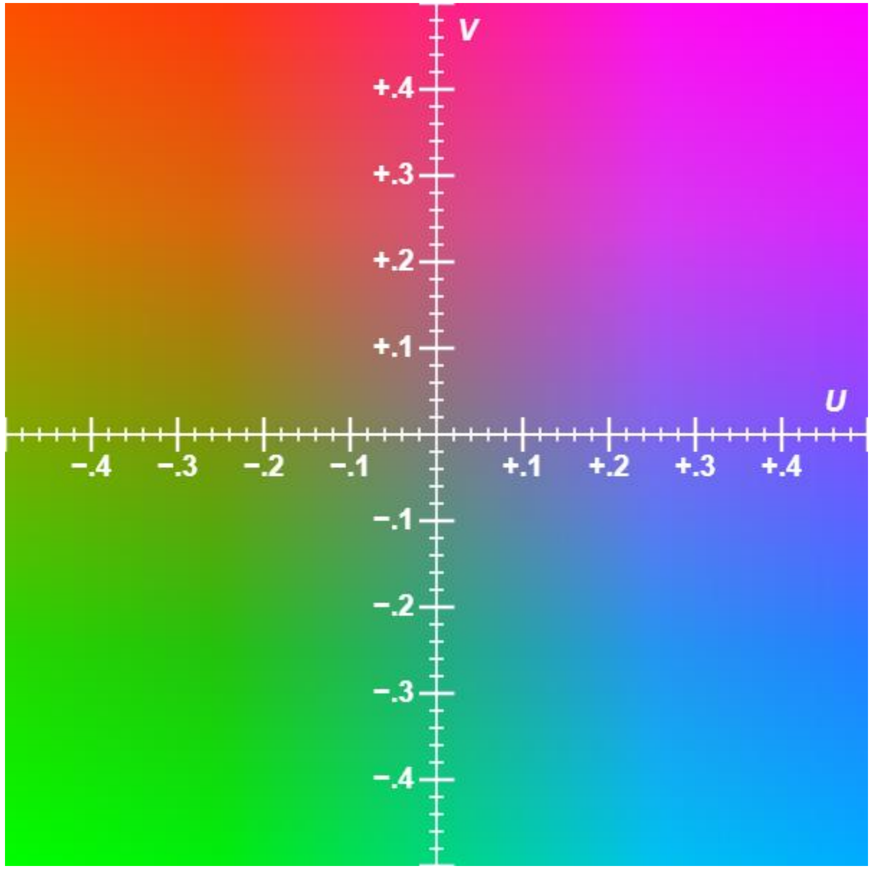


V

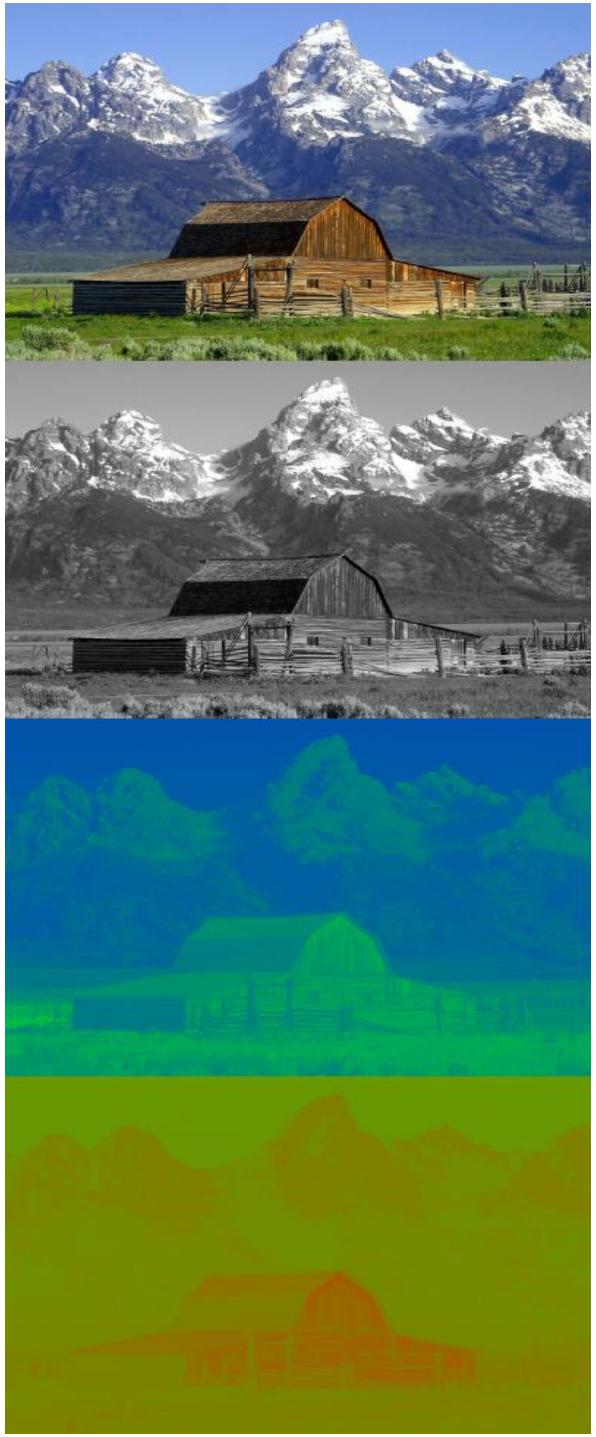
Generally, brighter pixels represent higher Y, U, V values.

Color Space (13/20)

➢ Example of YUV Model



- YUV



Y: Luma
U: Blue ~ yellow
V: Red ~ cyan

Color Space (14/20)

➢ YCbCr Model

✓ Y CbCr is the color space used by the "Rec. 601" standard for digital video.

- Used in JPEG image compression and MPEG video compression.

Color Space (15/20)

➢ YCbCr Model

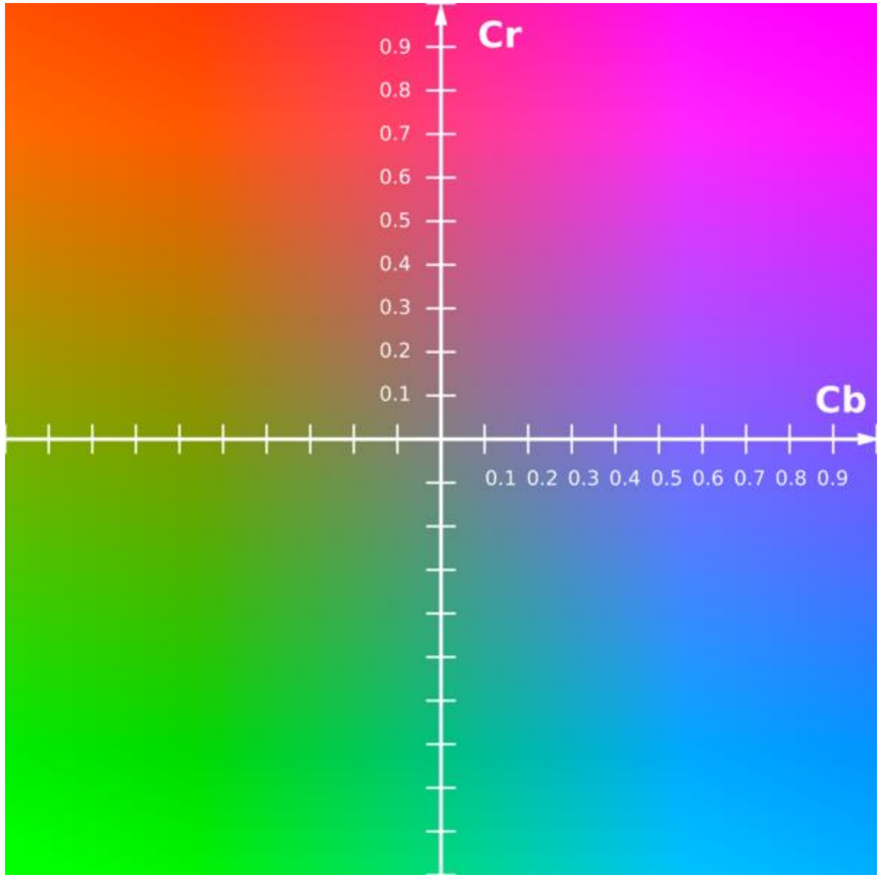
- **RGB to YCbCr**

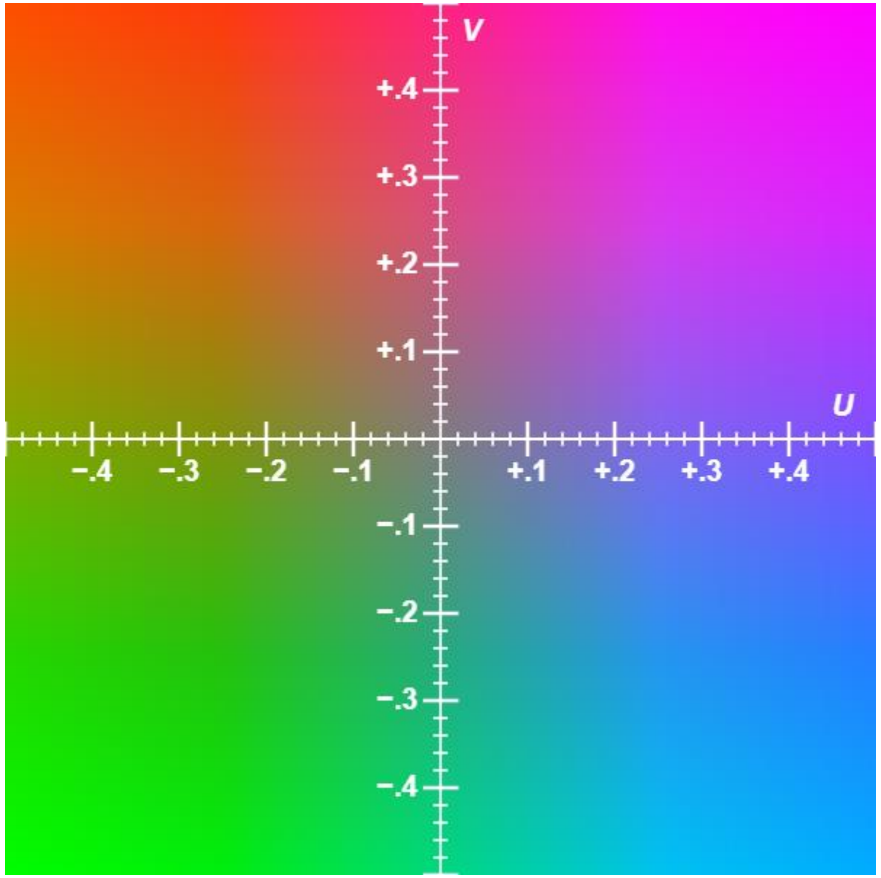
$$
\begin{aligned}
Y &= 0.299 * R + 0.587 * G + 0.114 * B \\
Cr &= (R - Y) * 0.713 + 128 \\
Cb &= (B - Y) * 0.564 + 128
\end{aligned}
$$

$$
\begin{aligned}
R &= Y + 1.403 * (Cr - 128) \\
G &= Y - 0.344 * (Cr - 128) - 0.714 * (Cb - 128) \\
B &= Y + 1.773 * (Cb - 128)
\end{aligned}
$$

Color Space (16/20)

➢ YCbCr Color Model





- YCbCr

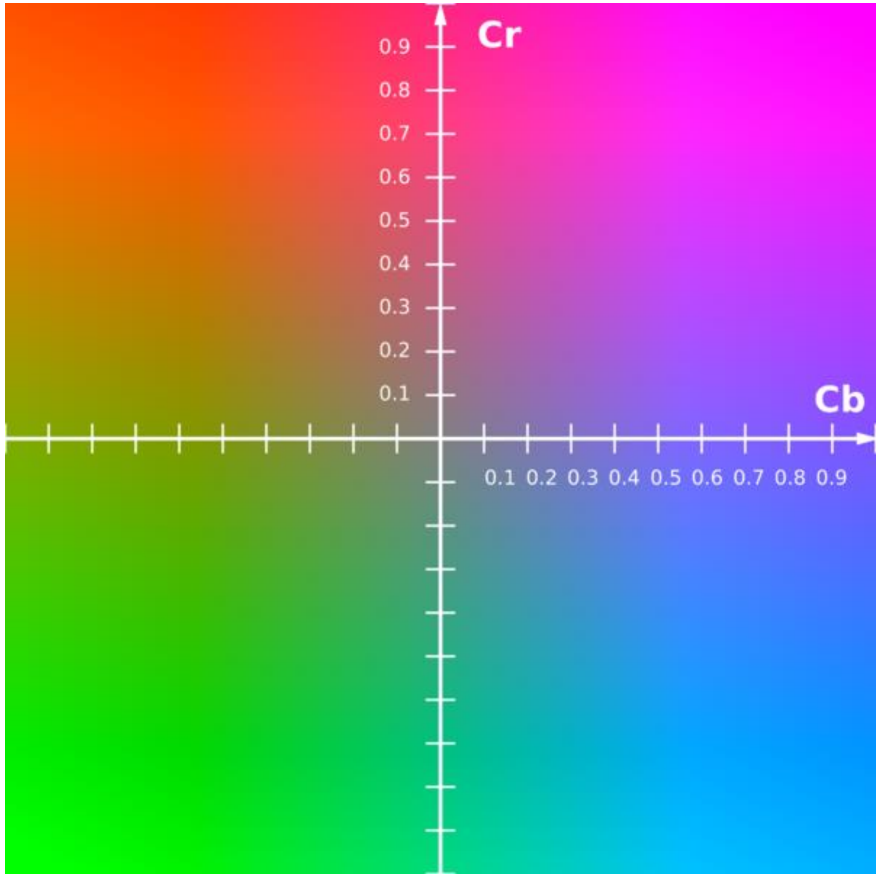
- YUV

Source: https://en.wikipedia.org/wiki/File:YCbCr-CbCr_Scaled_Y50.png

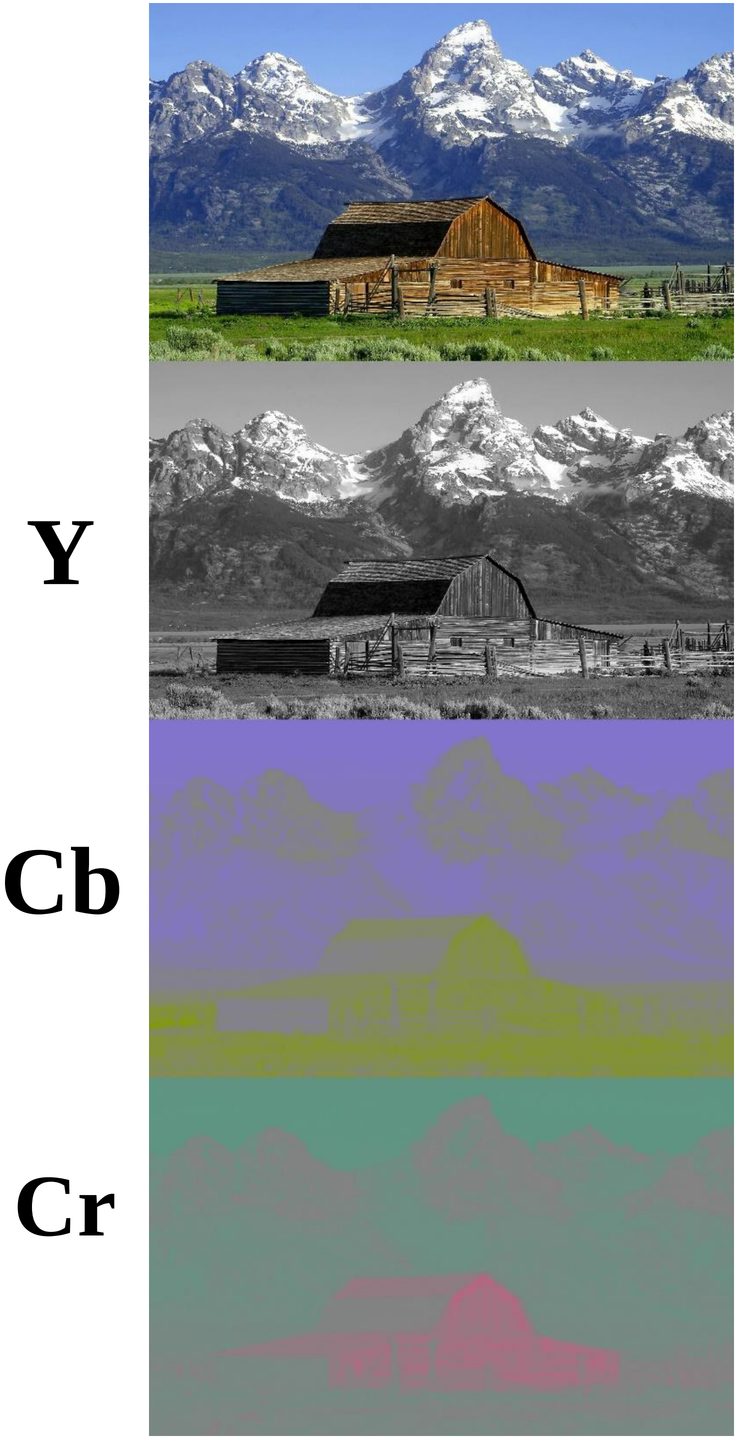
Source: https://en.wikipedia.org/wiki/File:YUV_UV_plane.svg

Color Space (17/20)

➢ Example of YCbCr Model



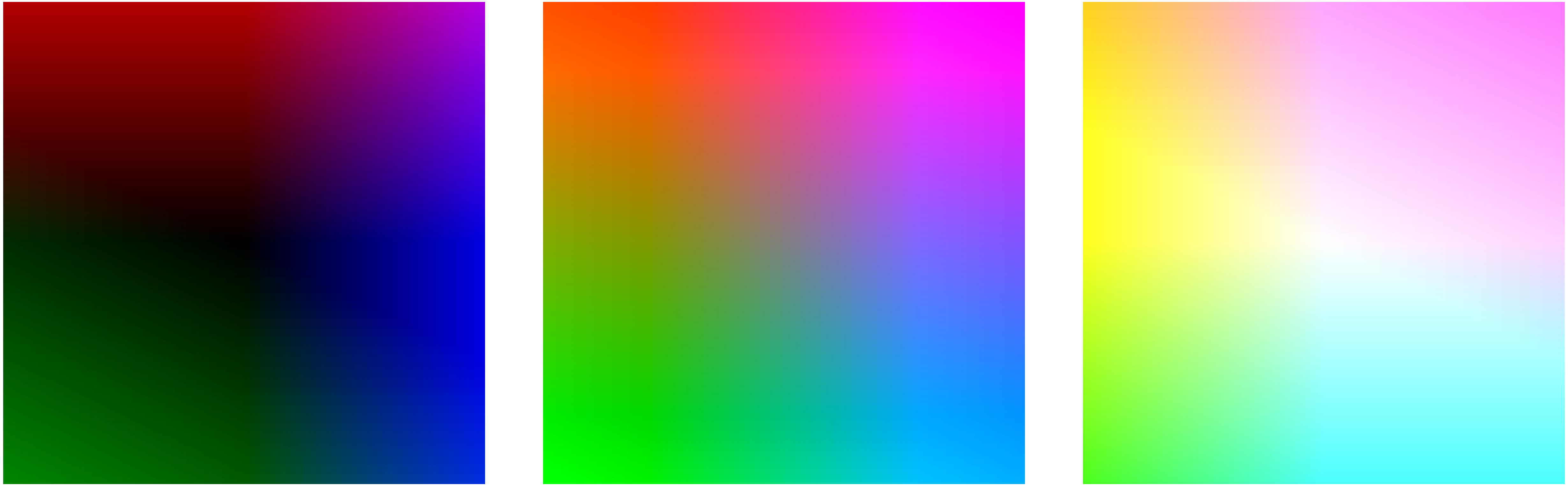
- YCbCr



Color Space (18/20)

➢ YCbCr Color Model

CbCr Color Planes at different Y values



Y=0

Y=128

Y=255

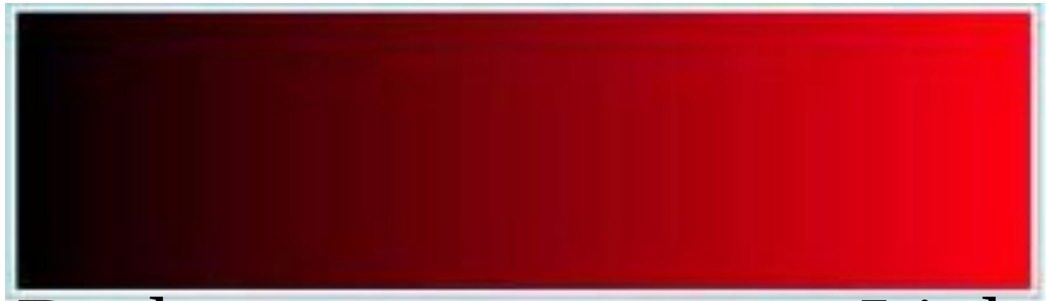
Color Space (19/20)

- **HSV color model**

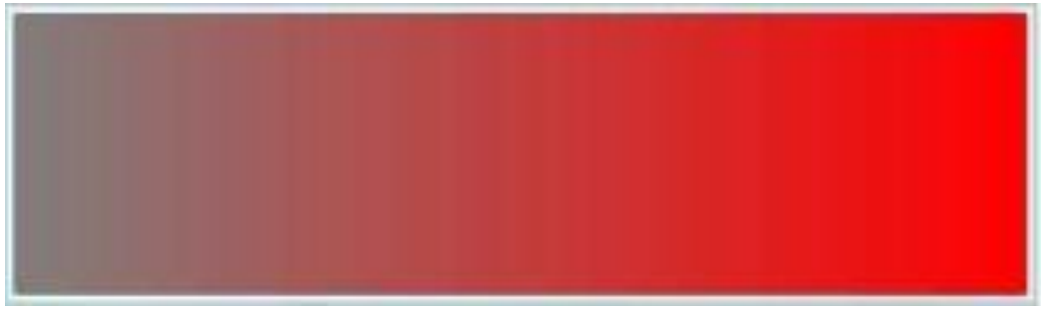
- ***HSV (Hue Saturation Value)***
- **HSL (Hue Saturation Lightness)**
- **HSI (Hue Saturation Intensity)**

Color Space (20/20)

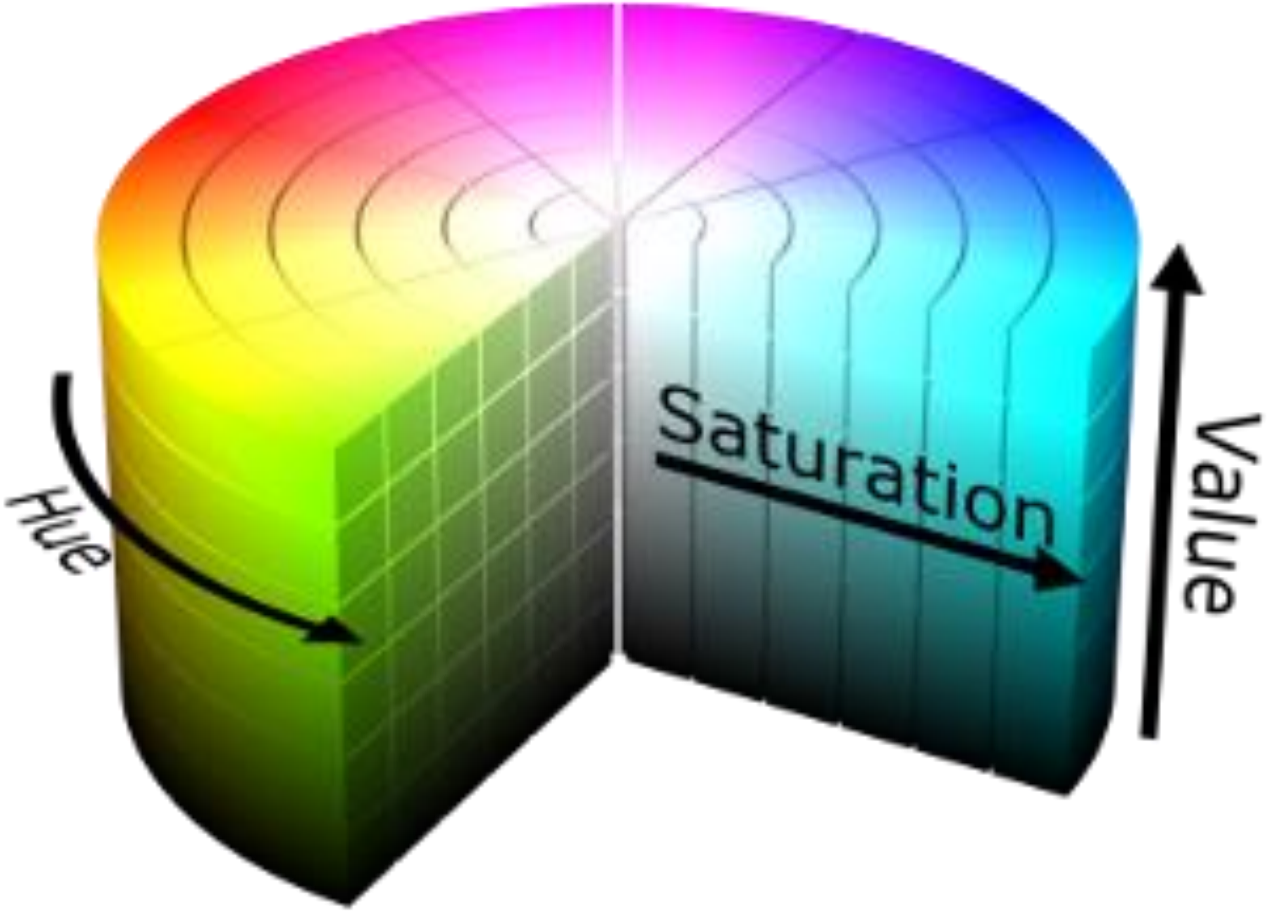
- **HSV color model**



Dark **Value** Light



Saturation



Color Space in OpenCV (1/4)

- Header

```
#include "opencv2/imgproc.hpp"

#include "opencv2/imgcodecs.hpp"

#include "opencv2/opencv.hpp"
#include "opencv2/core.hpp"
```

Colors in OpenCV (2/4)

➢ Code

Syntax:

cvtColor(src, dst, Convert Type);

src − Input image: 8-bit unsigned, 16-bit unsigned, or single-precision floating-point.
dst − Output image of the same size and depth as src.
Convert Type − Color space conversion code. **Ex: COLOR_BGR2GRAY**

Colors in OpenCV (3/4)

Note:

***Ex*: COLOR_BGR2GRAY**

↓

- ✓ Note that, in OpenCV, the order of colors is usually BGR, not RGB.

Colors in OpenCV (4/4)

➢ Code

Example:

```
int main()
{
Mat Lady = imread("D:/lena.jpg");
imshow("Original Image", Lady);

Mat Lady_Gray;
cvtColor(Lady,Lady_Gray, COLOR_BGR2GRAY);
imshow("Gray Image",Lady_Gray);

Mat Lady_HSV;
cvtColor(Lady, Lady_HSV, COLOR_BGR2HSV);
imshow("HSV Image", Lady_HSV);

Mat Lady_YUV;
cvtColor(Lady, Lady_YUV,COLOR_BGR2YUV);
imshow("YUV Image", Lady_YUV);

Mat Lady_YCrCb;
cvtColor(Lady, Lady_YCrCb,COLOR_BGR2YCrCb);
imshow("YCrCb Image", Lady_YCrCb);

waitKey(0);
destroyAllWindows();

return(0);
}
```

Demo

Practice

Thanks!

Any questions?