[COLOR MODEL OF LIGHT (RGB)]

LECTURE 16

We know,

I = intensity of light

There, is three things to light : i) brightness
ii) color
iii) Saturation (colorness)

In Real Life :

$$I \rightarrow \left.\begin{matrix} I_R \\ I_G \\ I_B \end{matrix}\right]$$

whoever holds the highest value will dominate.

(There's two important things to consider)

1) Monitor, color (emits light as a source) → RGB

2) Paper print, pigment (absorbs lights) → CMY

(white)

$W = R + G + B$

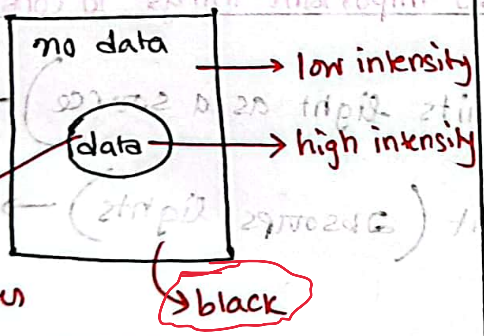
$C = W - R \longrightarrow R+G+B-R \longrightarrow (G+B)$

$M = W - G \longrightarrow R+G+B-G \longrightarrow (R+B)$

$Y = W - B \longrightarrow R+G+B-B \longrightarrow (R+G)$

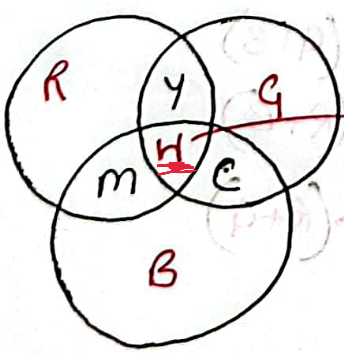
⊛ objects that emit light

Additive Color Model (RGB)

1. Refers to RGB
2. Works on the principle of emission
3. Additive : increase in color causes effective increase in intensity
4. For making monitor color
5. 

has to be brighter to be displayed

eg: monitor

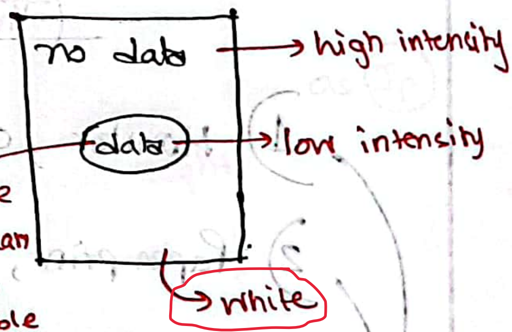
6. 

due to overlap ↑ (bright) white

Ø additive so highest intensity or brightness at white

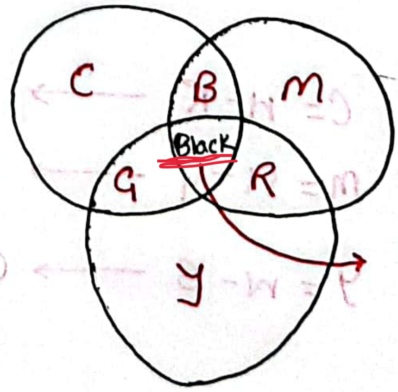
⊛ objects that can't be seen without light

Subtractive Color Model (CMY)

1. Refers to CMY
2. Reflection (seen through reflection)
3. Subtractive : increase in color causes effective decrease in intensity.
4. For making paper color
5. 

has to be darker than white to be visible

eg: paper

6. 

(dark) black ↓ due to overlap.

Ø subtractive so lowest intensity at black.

$$\overleftarrow{(1-C,\ 1-M,\ 1-Y)}$$

R , G , B $\longleftrightarrow$ C, M, Y

$$\underrightarrow{(1-R,\ 1-G,\ 1-B)}$$

(0.7, 0.6, 0.8) (0.3, 0.4, 0.2)

R = 1 - 0.3
G = 1 - 0.4
B = 1 - 0.8

← by subtracting RGB or CMY from 1 we can move to either side →

C = 1 - 0.7
M = 1 - 0.6
Y = 1 - 0.8

Example for pure (RED)

R G B → C M Y

(1, 0, 0) → (0, 1, 1)

only red with high intensity

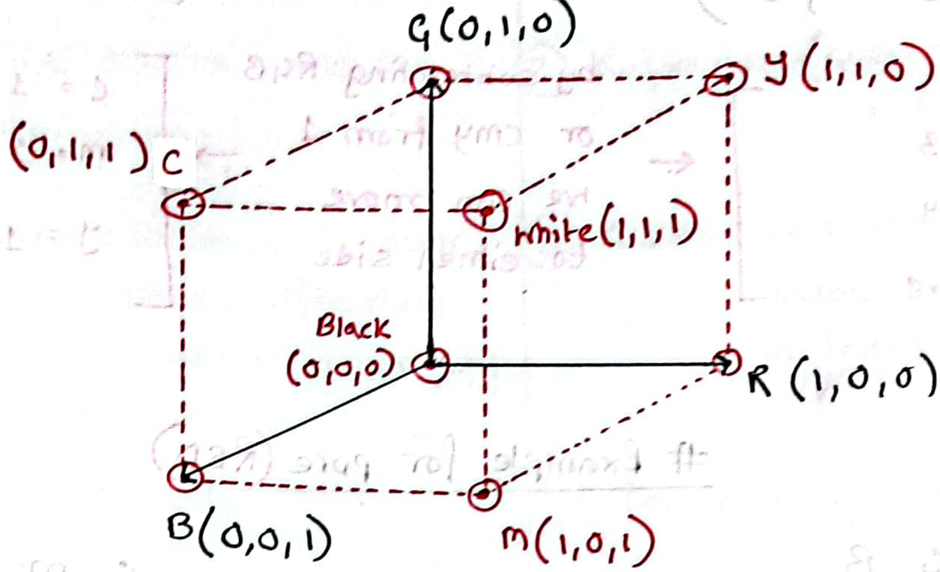
most common reflector of red, So high red color.

Ø

R = Red
G = Green
B = Blue

C = Cyan
M = Magenta
Y = Yellow

∅ (RGB color Cube Model)



∅ each axis is 8 bit : (0 ~ 255)

So this model can represent : $\boxed{2^8 \times 2^8 \times 2^8 = (16 \text{ millon colors})}$

∅ (32 bit color)

RGBA

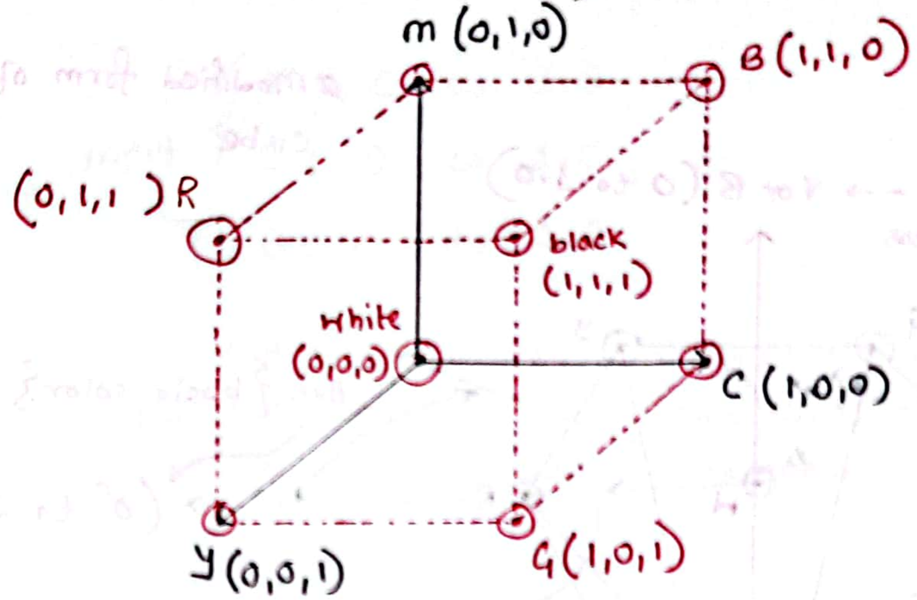
↳ Alpha channel

(helps to create more convincing gradients, shadows & transparenencies & can support 4 billion color combinations)

↳ can be used to correctly produce real life object.

∅ Now, for cmy cube, the diagonal opposites will be switched.

(CMY Color Cube)



(R, G, B) ⟶ is theoretically a floating point value

- Good for hardware
- But no one uses RGB model in software because it doesn't present us with the following infos
  - Brightness ?
  - Hue ?
  - Saturation ?

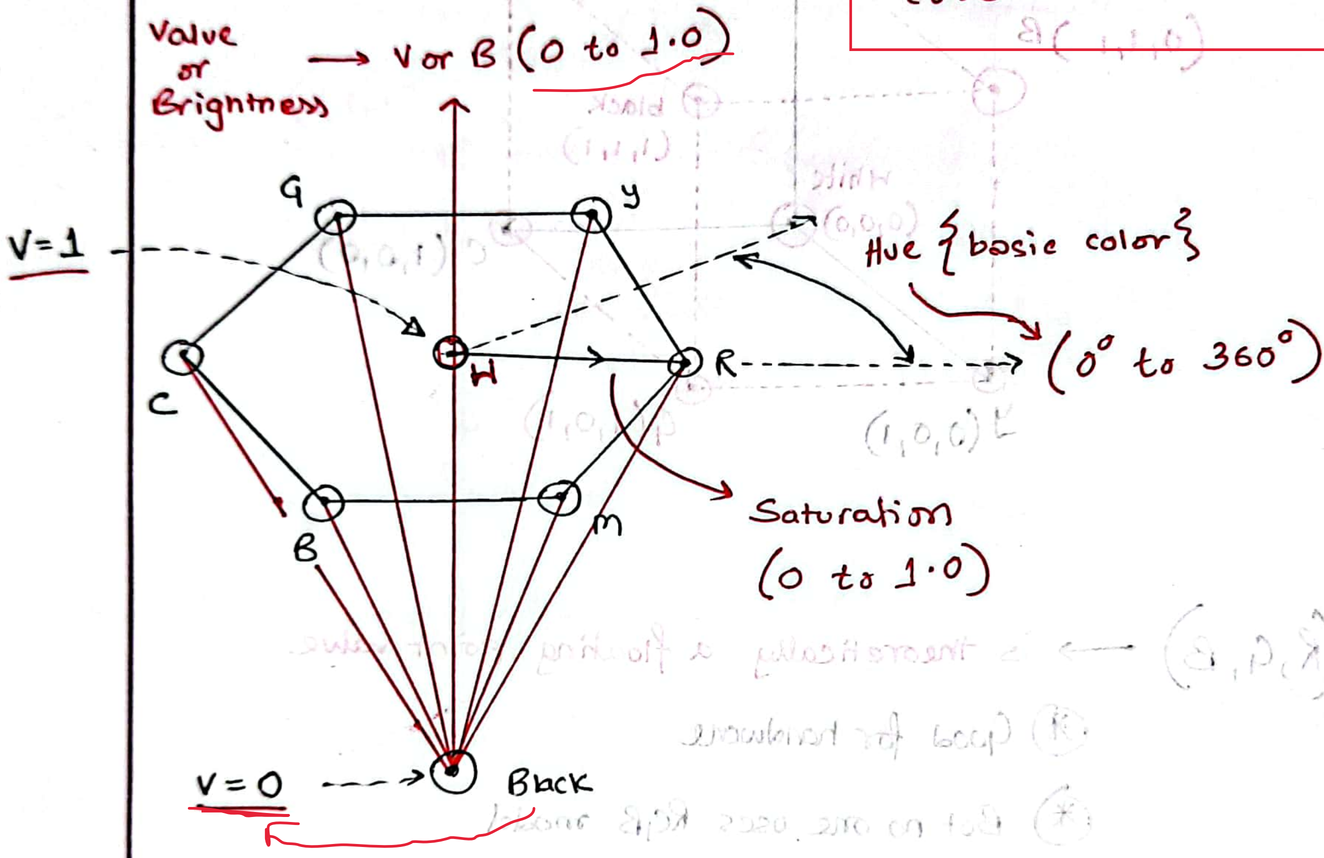
Hence for representation we use :

(i) HSV / HSB
(ii) HSL / HLS

models.

(i) ∅ HSV / HSB Color Model : (Representation model)

∅ Modified form of RGB cube



eg:

RGB

$(0.8, 0.6, 0.2)$ → This two RGB values represent the same color, which is based on the mutual proportion of the 3 values

$R_1 \quad G_1 \quad B_1$

↓

$(0.4, 0.3, 0.1)$ But, the upper one has a higher brightness.

$R_2 \quad G_2 \quad B_2$

R1/R2 = G1/G2 = B1/B2

So the same color but more brighter

(RGB to HSV: conversion)

input
- R : 0 to 1.0
- G : 0 to 1.0
- B : 0 to 1.0

output
- H : 0° to 360° → Hue (basic color)
- S : 0 to 1.0 → saturation
- V : 0 to 1.0 → Brightness.

where,

$\boxed{V = \max(R, G, B)}$ → the value of V will depend on the maximum in between R, G & B

$$\boxed{S = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)}}$$

$\boxed{H = \text{will depend on dominance,}}$

for code (P.T.O) →

Next Lecture (17)