Day 7: Image Color Spaces

Outcomes:

- Understand what a **color space** is

- Know the difference between **BGR, Grayscale, and HSV**

- Convert images between color spaces using OpenCV

- Understand when and why different color spaces are used

What is a color space?

A color space is a way to represent colors using numbers.

BGR and Grayscale are two already known color spaces.

BGR color space

Shape:

```
(height, width, 3)
```

This is the default color space in OpenCV.

```
BGR → [Blue, Green, Red]
```

- Each channel ranges from 0-255

Grayscale

Grayscale removes color and keeps intensity.

Converting to grayscale:

```
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

Shape:

```
[height, width]
```

- One channel only, ranges from 0-255 (8-bit images, default)

- 0 → black, 255 → white

- Values in between → shades of gray

- Used in edge detection, face detection, thresholding

HSV color space (imp)

Converting into HSV:

```
hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
```

H → Hue

S → Saturation

V → Value

HSV separates **color** from **brightness**, which is extremely useful.

Why HSV

- In BGR, Color and brightness are mixed.

- In HSV, you can detect colors regardless of lighting.

- Hence, HSV is used in color detection, object tracking, segmentation.

Viewing HSV channels separately

```
h, s, v = cv2.split(hsv)


cv2.imshow("Hue", h)

cv2.imshow("Saturation", s)

cv2.imshow("Value", v)
```

12-hsv.py

```python
import cv2 as cv


img = cv.imread("OpenCV-codes/images/image.png")
hsv = cv.cvtColor(img, cv.COLOR_BGR2HSV)


h, s, v = cv.split(hsv)


cv.imshow("Hue",h)
cv.imshow("Saturation", s)
cv.imshow("Value", v)


cv.waitKey()
cv.destroyAllWindows()
```

Separated H, S, V View

What a separated h, s, v look like as an image (refer 12-hsv.py)

Hue

- Looks black and white

- Stores color type as numbers ranging from 0 – 179

- It does not represent brightness

Saturation

- Looks black and white and ranges from 0 – 255

  - 0 (black when separated) → no color (gray)

  - 255 (white when separated) → strong color

- Controls **color intensity**, not brightness

Value

- Controls light/dark

- No effect on color type

Note: If S = 0 (no color) → grayscale, even if hue has a value, it becomes meaningless.

Conclusion

BGR – good for display

Grayscale – good for processing

HSV – good for color-based processing.