

# **Day 5: Drawing on Images**

(lines, rectangles, circle, text)

## **Outcomes:**

- Draw lines, rectangles, circles, and text on images
- Control color, thickness, and position of drawings
- Modify an image array directly

# Drawing a Line

Syntax of line function:

```
cv2.line(img, start_point, end_point, color, thickness)
```

Parameters:

<code>img</code>	Image array ( <code>img</code> )
<code>start_point</code>	( $x_1, y_1$ )
<code>end_point</code>	( $x_2, y_2$ )
<code>color</code>	(B, G, R)
<code>thickness</code>	Integer (1, 2, 3)

### 3-line.py

```
import cv2

img = cv2.imread("OpenCV-codes/images/image.png")

if img is None:
    print("Error: Image not found")
    exit()

#Draws a blue line from (50,50) to (200,200) with thickness 3
cv2.line(img, (50, 50), (200, 200), (255, 0, 0), 3)

cv2.imshow("Line", img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

# Drawing a Rectangle

Syntax of rectangle function:

```
cv2.rectangle(img, pt1, pt2, color, thickness)
```

Parameters:

img	Image array (img)
pt1	Top-Left Corner (x <sub>1</sub> , y <sub>1</sub> )
pt2	Bottom-Right Corner (x <sub>2</sub> , y <sub>2</sub> )
color	(B, G, R)
thickness	Integer (1, 2, 3) -1 for filled rectangle

## 4-rectangle.py

```
import cv2

img = cv2.imread("OpenCV-codes/images/image.png")

if img is None:
    print("Error: Image not found")
    exit()

# Draw a green rectangle from top-left (100,100) to bottom-right (300,300)
cv2.rectangle(img, (100,100), (300,300), (0,255,0), 2)

cv2.imshow("Line", img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

# Drawing a Circle

Syntax of circle function:

```
cv2.circle(img, center, radius, color, thickness)
```

Parameters:

img	Image array (img)
center	(x, y)
radius	Integer (1, 2, 3...)
color	(B, G, R)
thickness	Integer (1, 2, 3...) -1 for filled circle

## 5-circle.py

```
import cv2

img = cv2.imread("OpenCV-codes/images/image.png")

if img is None:
    print("Error: Image not found")
    exit()

# Draw a red circle at center (250,250) with radius 50
cv2.circle(img, (250,250), 50, (0,0,255), 2)

cv2.imshow("Line", img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

# Adding Text

Syntax of text function:

```
cv2.putText(img, text, org, font, fontScale, color,  
thickness)
```

Parameters:

<code>img</code>	Image array ( <code>img</code> )
<code>text</code>	String to display
<code>org</code>	Bottom-Left Corner ( $x, y$ )
<code>font</code>	Font-Type
<code>fontScale</code>	Size of Text
<code>color</code>	(B, G, R)
<code>thickness</code>	Integer (1, 2, 3...)

6-text.py

```
import cv2

img = cv2.imread("OpenCV-codes/images/image.png")

if img is None:
    print("Error: Image not found")
    exit()

# Add text at position (50,400)
cv2.putText(img, "Hello OpenCV", (50,400),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,0), 2)

cv2.imshow("Line", img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```