



# **Process Automation Engineering**

## **Image Processing**

### **Laboratory Work 2**

Working with Images in Python with OpenCV

**Student's full name:** Avaz Asgarov

**Group number:** 22.2

**Report submitted on:** 28.09.2025

**Supervisor's name:** Dr. Leyla Muradkhanli

Baku Higher Oil School

## Task 1: Read and Display an Image

Code:

```
1 import cv2 as cv
2
3 # Task 1: Read and Display the Original Image
4 img = cv.imread("ronaldo.jpg")
5 cv.imshow("Original Image", img)
```

Result:



*Original Image*

## Task 2: Print Details about the Image

Code:

```
1 # Task 2: Print Details about the Image
2 print(f"Image Shape (H x W x C): {img.shape}")
3 print(f"Total Number of Pixels: {img.size}")
```

Result:

```
Image Shape (H x W x C): (920, 736, 3)
Total Number of Pixels: 2031360
```

### Task 3: Convert the Image to Grayscale

Code:

```
1 # Task 3: Convert the Image to Grayscale
2 img_gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
3 cv.imshow("Grayscale Image", img_gray)
4 cv.imwrite("ronaldo_grayscale.png", img_gray)
```

Result:



*Grayscale Image*

### Task 4: Find Color of Pixel at [70, 100]

Code:

```
1 # Task 4: Find Color of Pixel at [70, 100]
2 pixel_color = img[70, 100]
3 print(f"Color of Pixel at [70, 100]: {pixel_color}") # BGR
    format
```

Result:

Color of Pixel at [70, 100]: [249 247 247] (BGR values)

## Task 5: Modify the Pixel Value at [200, 150] to Red

Code:

```
1 # Task 5: Modify the Pixel Value at [200, 150] to Red
2 img_modified = img.copy()
3 img_modified[200, 150] = [0, 0, 255] # Red in BGR
4 cv.imshow("Red Pixel Modified", img_modified)
5 cv.imwrite("ronaldo_red_pixel.png", img_modified)
```

Result:



*Image with Red Pixel at [200, 150]*

## Task 6: Modify a Region to Green [100:300, 150:400]

Code:

```
1 # Task 6: Modify a Region to Green [100:300, 150:400]
2 img_modified[100:300, 150:400] = [0, 255, 0] # Green in BGR
3 cv.imshow("Green Region Modified", img_modified)
4 cv.imwrite("ronaldo_green_region.png", img_modified)
```

Result:



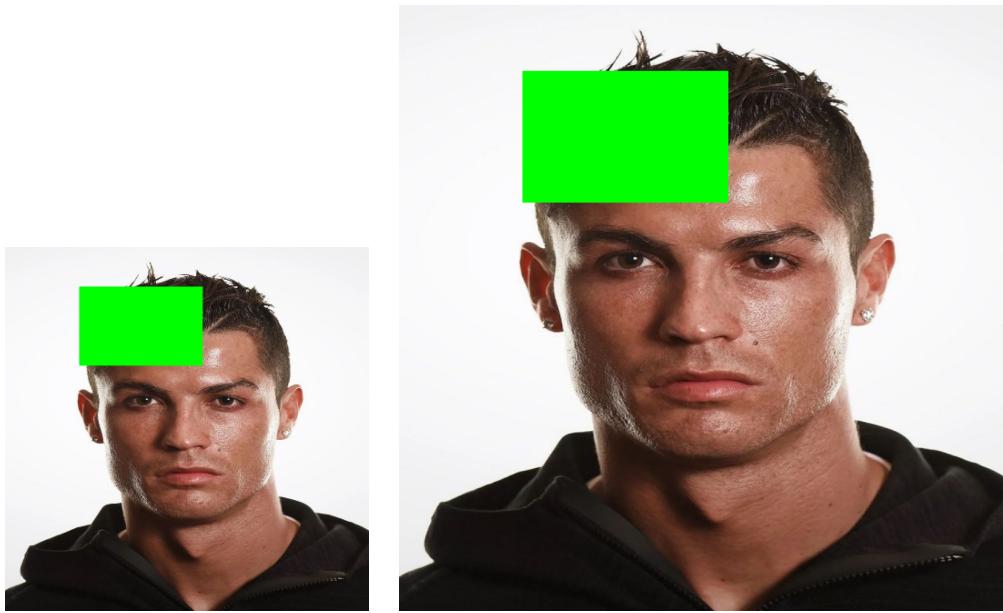
*Image with Green Region*

## Task 7: Resize the Image

Code:

```
1 # Task 7: Resize the Image
2 img_smaller = cv.resize(img_modified, (400, 400))
3 img_larger = cv.resize(img_modified, (1000, 1000))
4
5 cv.imshow("Resized Smaller Image", img_smaller)
6 cv.imshow("Resized Larger Image", img_larger)
7
8 cv.imwrite("ronaldo_smaller.png", img_smaller)
9 cv.imwrite("ronaldo_larger.png", img_larger)
```

Result:



Smaller (400x400)

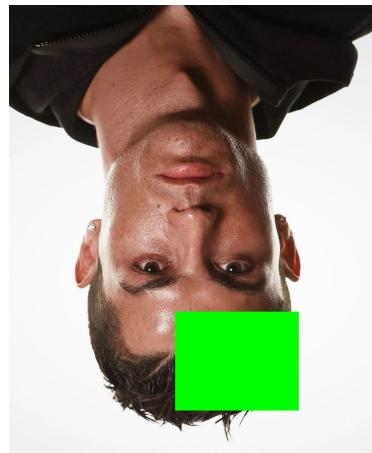
Larger (1000x1000)

## Task 8: Rotate the Image

Code:

```
1 # Task 8: Rotate the Image
2 img_rotate_180 = cv.rotate(img_modified, cv.ROTATE_180)
3 img_rotate_90_cw = cv.rotate(img_modified, cv.ROTATE_90_CLOCKWISE
     )
4 img_rotate_90_ccw = cv.rotate(img_modified, cv.
     ROTATE_90_COUNTERCLOCKWISE)
5
6 cv.imshow("Rotated 180 degrees", img_rotate_180)
7 cv.imshow("Rotated 90 degrees Clockwise", img_rotate_90_cw)
8 cv.imshow("Rotated 90 degrees Counterclockwise",
     img_rotate_90_ccw)
9
10 cv.imwrite("ronaldo_rotated_180.png", img_rotate_180)
11 cv.imwrite("ronaldo_rotated_90_cw.png", img_rotate_90_cw)
12 cv.imwrite("ronaldo_rotated_90_ccw.png", img_rotate_90_ccw)
```

Result:



*Rotated 180°*



*Rotated 90° Clockwise*



*Rotated 90° Counterclockwise*

## Task 9: Flip Image Horizontally and Vertically

Code:

```
1 # Task 9: Flip Image Horizontally and Vertically
2 img_flip_vertical = cv.flip(img_modified, 0)
3 img_flip_horizontal = cv.flip(img_modified, 1)
4
5 cv.imshow("Flipped Vertically", img_flip_vertical)
6 cv.imshow("Flipped Horizontally", img_flip_horizontal)
7
8 cv.imwrite("ronaldo_flipped_vertical.png", img_flip_vertical)
9 cv.imwrite("ronaldo_flipped_horizontal.png", img_flip_horizontal)
10
11 cv.waitKey(0)
12 cv.destroyAllWindows()
```

Result:



*Flipped Vertically*



*Flipped Horizontally*