**EX.NO: 7 DATE:**

**Load an image file and do crop and flip operation using NumPy Indexing**

**AIM:**

To load an image file and perform cropping and flipping operations using NumPy indexing.

**Procedure:**

1. **Load an Image:**

Use the PIL (Pillow) library to load an image and convert it to a NumPy array.

1. **Crop the Image:**

Define the coordinates for cropping and use NumPy indexing to extract the desired region.

1. **Flip the Image:**

Use NumPy indexing to flip the image both horizontally and vertically**.**

**PROGRAM:**

import numpy as np from PIL import Image

import matplotlib.pyplot as plt

**# 1. Load an image file**

**# Replace 'your\_image.jpg' with the path to your image file**

image\_path = "G:\\AYYAPPAN\\SKCET\\3rdYear\\Python\\LabManual\\PythonLab\\1234.jpeg"

image = Image.open(image\_path) image\_np = np.array(image)

**# Display the original image** plt.imshow(image\_np) plt.title("Original Image")

plt.axis('off') plt.show()

**# 2. Crop the image**

**# Define the coordinates for cropping: (start\_row:end\_row, start\_col:end\_col)** crop\_x\_start, crop\_x\_end = 50, 200 # Example crop coordinates for x-axis crop\_y\_start, crop\_y\_end = 50, 200 # Example crop coordinates for y-axis cropped\_image = image\_np[crop\_y\_start:crop\_y\_end, crop\_x\_start:crop\_x\_end]

**# Display the cropped image** plt.imshow(cropped\_image) plt.title("Cropped Image") plt.axis('off')

plt.show()

**# 3. Flip the image # Flip horizontally**

flipped\_image\_h = np.flip(cropped\_image, axis=1) **# Display the horizontally flipped image** plt.imshow(flipped\_image\_h) plt.title("Horizontally Flipped Image") plt.axis('off')

plt.show()

**# Flip vertically**

flipped\_image\_v = np.flip(cropped\_image, axis=0) **# Display the vertically flipped image** plt.imshow(flipped\_image\_v)

plt.title("Vertically Flipped Image") plt.axis('off')

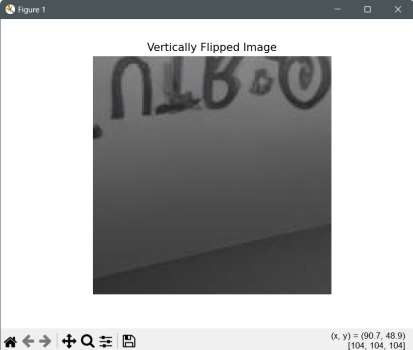
plt.show()

**OUTPUT:**









**Result:**

The program successfully loaded an image file, performed cropping, and flipped the image both horizontally and vertically using NumPy indexing.