# Install required libraries if not already installed

!pip install opencv-python pillow matplotlib

# Import necessary libraries

import cv2

import numpy as np

from google.colab import files

from PIL import Image

import matplotlib.pyplot as plt

# Step 1: Upload an image

uploaded = files.upload()

# Open the image using PIL and convert it to a format OpenCV understands

image\_path = list(uploaded.keys())[0]

image = Image.open(image\_path)

image\_cv = np.array(image)

# Convert from RGB to BGR (since OpenCV uses BGR by default)

image\_cv = cv2.cvtColor(image\_cv, cv2.COLOR\_RGB2BGR)

# Get the image dimensions (height, width)

height, width = image\_cv.shape[:2]

# Step 2: Define rotation angle

angle = 45 # Rotate 45 degrees (positive for counterclockwise, negative for clockwise)

# Step 3: Calculate the rotation matrix for clockwise rotation

center = (width // 2, height // 2) # Center of the image

rotation\_matrix = cv2.getRotationMatrix2D(center, angle, 1) # 1 means no scaling

# Step 4: Apply the rotation for counterclockwise (positive angle)

rotated\_image\_ccw = cv2.warpAffine(image\_cv, rotation\_matrix, (width, height))

# Apply rotation for clockwise (negative angle)

rotation\_matrix\_cw = cv2.getRotationMatrix2D(center, -angle, 1) # Negative angle for clockwise

rotated\_image\_cw = cv2.warpAffine(image\_cv, rotation\_matrix\_cw, (width, height))

# Step 5: Convert images back to RGB for displaying with matplotlib

rotated\_image\_ccw\_rgb = cv2.cvtColor(rotated\_image\_ccw, cv2.COLOR\_BGR2RGB)

rotated\_image\_cw\_rgb = cv2.cvtColor(rotated\_image\_cw, cv2.COLOR\_BGR2RGB)

# Step 6: Display the original, counterclockwise rotated, and clockwise rotated images

plt.figure(figsize=(15, 5))

# Display original image

plt.subplot(1, 3, 1)

plt.imshow(image)

plt.title("Original Image")

plt.axis('off')

# Display counterclockwise rotated image

plt.subplot(1, 3, 2)

plt.imshow(rotated\_image\_ccw\_rgb)

plt.title("Counterclockwise Rotated Image (45°)")

plt.axis('off')

# Display clockwise rotated image

plt.subplot(1, 3, 3)

plt.imshow(rotated\_image\_cw\_rgb)

plt.title("Clockwise Rotated Image (-45°)")

plt.axis('off')

plt.show()

# Optional: Save and download the rotated images

cv2.imwrite("rotated\_image\_ccw.jpg", rotated\_image\_ccw)

cv2.imwrite("rotated\_image\_cw.jpg", rotated\_image\_cw)

files.download("rotated\_image\_ccw.jpg")

files.download("rotated\_image\_cw.jpg")

