Lab Assignment 5

Problem Statement

Write a program to implement Image stiching panorama using Python

Description

In this assignment, your goal is to stitch two images together to form a panoramic view. You will use Python libraries, mainly OpenCV and NumPy, to accomplish this task. Follow the instructions and hints below to get started.

Instructions

- Downloading Images: Run the first cell provided to download the images (image1.png and image2.png) for this assignment. Ensure the images are downloaded to your working directory before attempting the stitching process.

- Required Libraries: You can solve this assignment using the OpenCV and NumPy libraries. OpenCV provides various utilities for image processing, and NumPy helps manage arrays effectively. Approach:

- Here's a brief outline to help you approach the problem:

  - Load Images: Use OpenCV's cv2.imread to load both images.
  - Stitch Images: Use the OpenCV Stitcher module to stitch images together.
  - Display and Save: Display each image individually, then display and save the stitched image.

Hints and Functions

- To stitch images, use the cv2.Stitcher_create() method to initialize a stitcher and call stitcher.stitch(images) on your list of images.

- Use cv2_imshow from google.colab.patches library to display images in Google Colaboratory

- Refer to OpenCV's error status codes if you encounter any issues with stitching.

By following these steps and hints, you should be able to complete the panorama stitching and achieve a merged output image representing both input images side-by-side.

In [ ]:

```
# Download assignment files
!wget https://github.com/buntyke/vnr_dlcv2024_labs/releases/download/DLCVLab5/image
!wget https://github.com/buntyke/vnr_dlcv2024_labs/releases/download/DLCVLab5/image
```

```
--2024-12-08 07:04:42--  https://github.com/buntyke/vnr_dlcv2024_labs/releases/dow
nload/DLCVLab5/image1.png
Resolving github.com (github.com)... 20.27.177.113
Connecting to github.com (github.com)|20.27.177.113|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e
65be/878811324/a85440f7-c2a6-46f9-b3c8-a534af49e2e1?X-Amz-Algorithm=AWS4-HMAC-SHA2
56&X-Amz-Credential=releaseassetproduction%2F20241208%2Fus-east-1%2Fs3%2Faws4_requ
est&X-Amz-Date=20241208T070442Z&X-Amz-Expires=300&X-Amz-Signature=9f662c5f0771e64c
bbb49ab7db4d633626a7c28ad1c12cbfd9f4f7d797306f79&X-Amz-SignedHeaders=host&response
-content-disposition=attachment%3B%20filename%3Dimage1.png&response-content-type=a
pplication%2Foctet-stream [following]
--2024-12-08 07:04:42--  https://objects.githubusercontent.com/github-production-r
elease-asset-2e65be/878811324/a85440f7-c2a6-46f9-b3c8-a534af49e2e1?X-Amz-Algorithm
=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20241208%2Fus-east-1%2
Fs3%2Faws4_request&X-Amz-Date=20241208T070442Z&X-Amz-Expires=300&X-Amz-Signature=9
f662c5f0771e64cbbb49ab7db4d633626a7c28ad1c12cbfd9f4f7d797306f79&X-Amz-SignedHeader
s=host&response-content-disposition=attachment%3B%20filename%3Dimage1.png&response
-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.19
9.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.19
9.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 476625 (465K) [application/octet-stream]
Saving to: ‘image1.png’

image1.png          100%[===================>] 465.45K  2.31MB/s    in 0.2s

2024-12-08 07:04:43 (2.31 MB/s) - ‘image1.png’ saved [476625/476625]

--2024-12-08 07:04:43--  https://github.com/buntyke/vnr_dlcv2024_labs/releases/dow
nload/DLCVLab5/image2.png
Resolving github.com (github.com)... 20.27.177.113
Connecting to github.com (github.com)|20.27.177.113|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e
65be/878811324/b948f860-03b0-49ea-af88-12561064cd1b?X-Amz-Algorithm=AWS4-HMAC-SHA2
56&X-Amz-Credential=releaseassetproduction%2F20241208%2Fus-east-1%2Fs3%2Faws4_requ
est&X-Amz-Date=20241208T070443Z&X-Amz-Expires=300&X-Amz-Signature=efdbec7d87cae565
6c3546bf57de94b7971ed13c4669b9adb25431335712df61&X-Amz-SignedHeaders=host&response
-content-disposition=attachment%3B%20filename%3Dimage2.png&response-content-type=a
pplication%2Foctet-stream [following]
--2024-12-08 07:04:44--  https://objects.githubusercontent.com/github-production-r
elease-asset-2e65be/878811324/b948f860-03b0-49ea-af88-12561064cd1b?X-Amz-Algorithm
=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20241208%2Fus-east-1%2
Fs3%2Faws4_request&X-Amz-Date=20241208T070443Z&X-Amz-Expires=300&X-Amz-Signature=e
fdbec7d87cae5656c3546bf57de94b7971ed13c4669b9adb25431335712df61&X-Amz-SignedHeader
s=host&response-content-disposition=attachment%3B%20filename%3Dimage2.png&response
-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.19
9.110.133, 185.199.108.133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.19
9.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 720417 (704K) [application/octet-stream]
Saving to: ‘image2.png’

image2.png          100%[===================>] 703.53K  2.84MB/s    in 0.2s

2024-12-08 07:04:46 (2.84 MB/s) - ‘image2.png’ saved [720417/720417]
```

In [ ]:

```
### WRITE CODE HERE ###
import cv2
import numpy as np
from google.colab.patches import cv2_imshow

def stitch_images(images):
    stitcher = cv2.Stitcher_create()

    # Convert the list of images to a NumPy array
    images_np = np.array(images)

    # Stitch the images
    status, stitched_image = stitcher.stitch(images_np)

    # Check if the stitching was successful
    if status != cv2.Stitcher_OK:
        raise Exception("Image stitching failed with status code:
                                            {}".format(status))
    return stitched_image

image1 = cv2.imread("./image1.png")
image2 = cv2.imread("./image2.png")

# Stitch the images
stitched_image = stitch_images([image1, image2])

# Save the stitched image
cv2.imwrite("panorama.jpg", stitched_image)

cv2_imshow(image1)
cv2_imshow(image2)
cv2_imshow(stitched_image)
```

In [ ]: