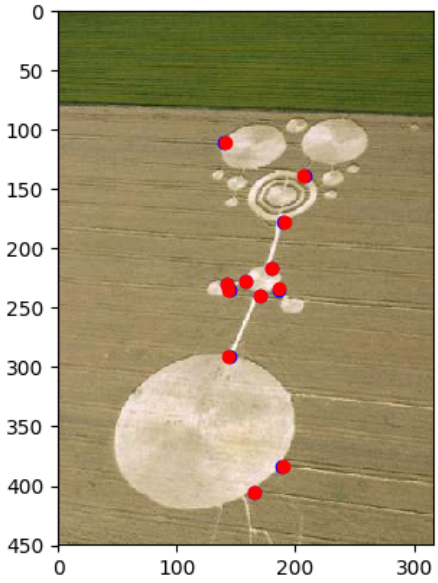
CS 4476 - Intro to Computer Vision
PS1 worksheet
Jingbo Wang

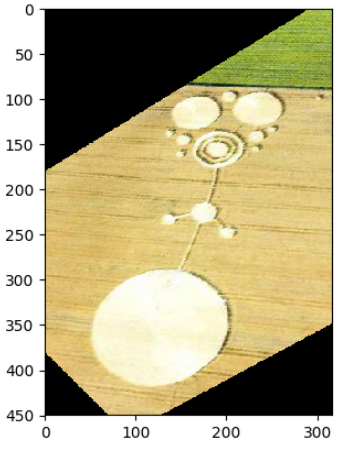
(All print statements commented out. Uncomment print statements and progress bar in 'warpImage.py' for detailed current status of the running program.)

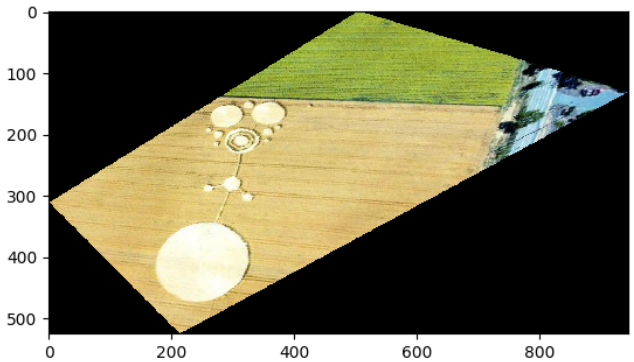
Verify Homography matrix computed from computeH function:
- the blue dots are originally selected points on image, and the red dots, which are covering the blue ones, are the transformed points from another crop image using the homography matrix



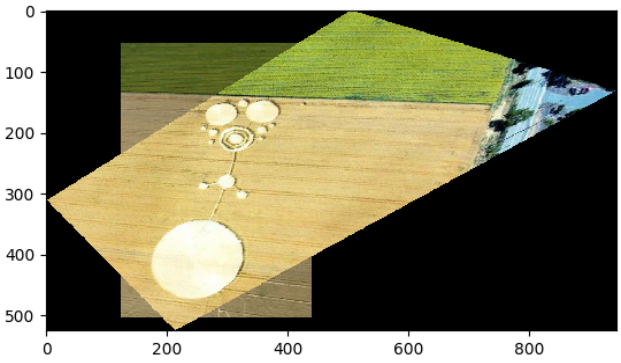
1. Crop Images

Warped Image (refIm size & full size):
(Uncomment the lines following 'display the full warped image' in 'warpImage.py' for display of the full-size warped image)



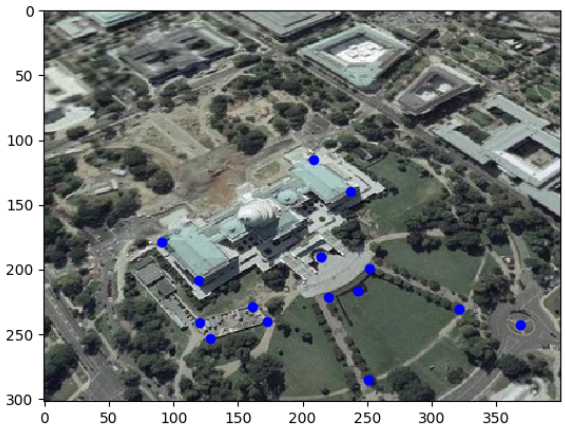


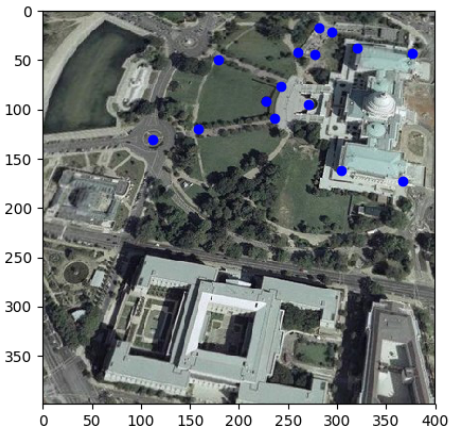
Merged Image:



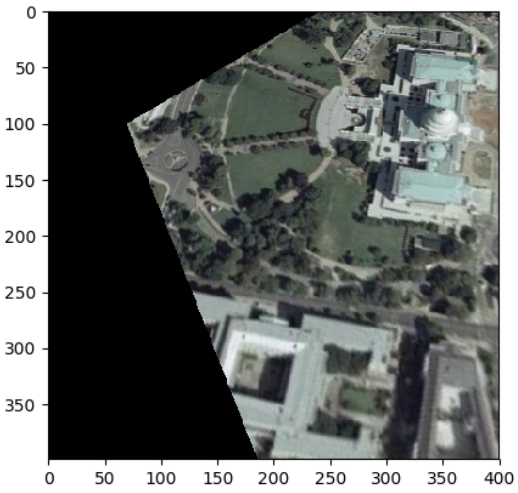
2. WDC Images

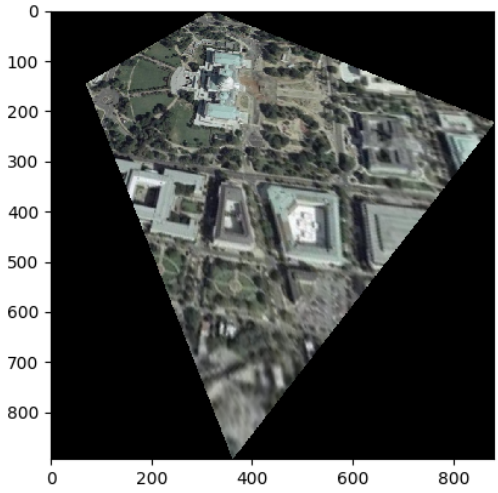
15 Selected Corresponding Points:



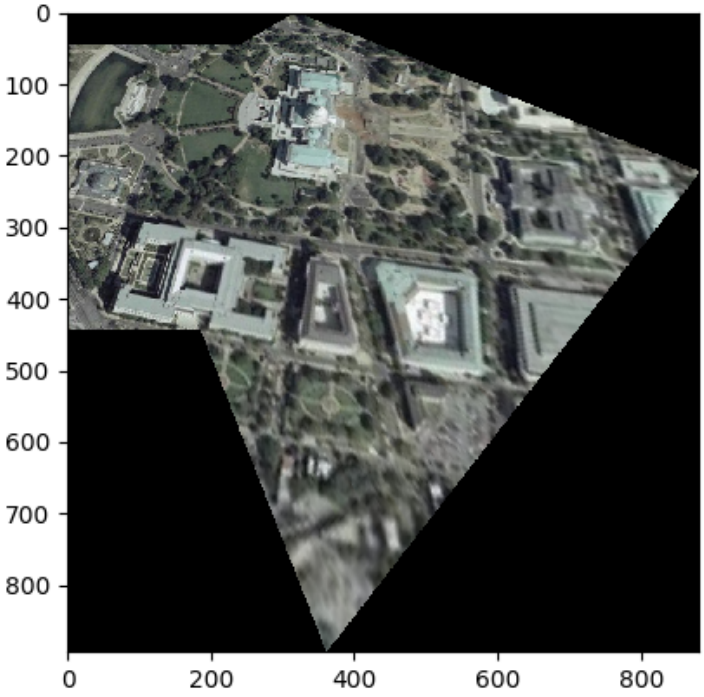


Warped Image (refIm size & full size):
(Uncomment the lines following 'display the full warped image' in 'warpImage.py' for display of the full-size warped image)





Merged Image:

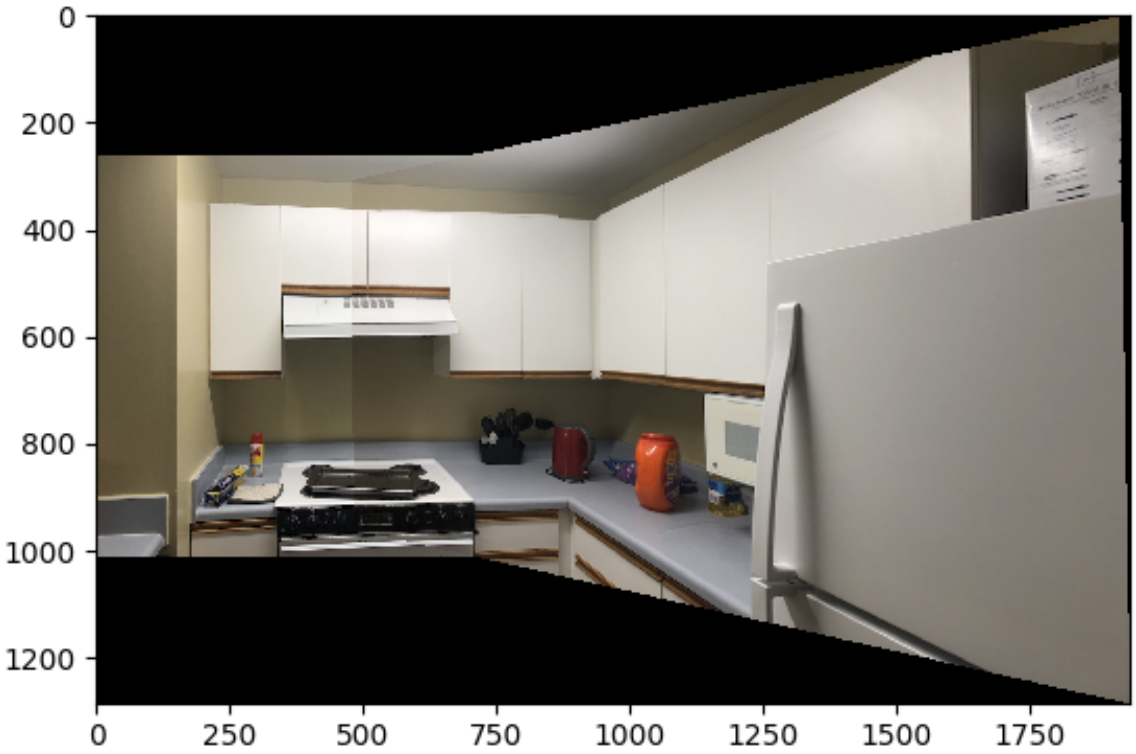


Self-taken Images: (Dorm Room: dorm1.jpg, dorm2.jpg)

dorm1.jpg

dorm2.jpg

Merged Image:

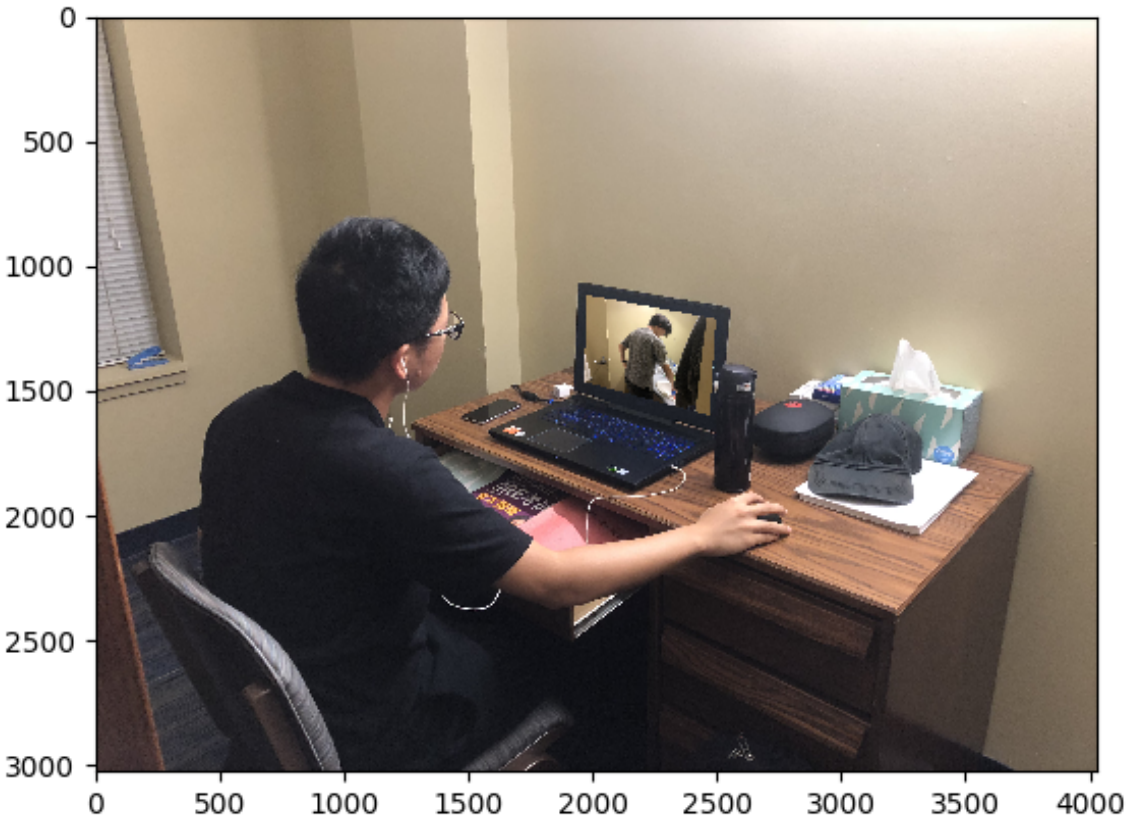


Frame Region Images: (frame.jpg, content.jpg)

frame.jpg

content.jpg

Merged Image:



Other Files Submitted:

'selectPoints.py'
Run 'python2 selectPoints.py'
- User can select corresponding points on two images
- (*commented out*) the result of selection for each image is printed in command line in 2xN format with x-axis on first row and y-axis on second row
- (*commented out*) the result of selection for both images is printed in command line in 2x2xN format with points from first image on the first column and points from second image on the second column
- edit the image1 and image2 URL to change the selecting images
- (*commented out*) save the overall result in a file named 'selectPoints.npy'

'run.py'
Run 'python2 run.py'
- reproduce all the answers in worksheet
  - verifying homography matrix
  - warped image for crop
  - merged image for crop
  - 15 selected points on wdc1.jpg
  - 15 selected points on wdc2.jpg
  - warped image for wdc
  - merged image for wdc
  - merged image for self-taken dorm images
  - merged image for frame region replacement
  - (the full size warped image if the lines following 'display the full warped image' in 'warpImage.py' are commented out)

'points.npy'
- Contains a 2x2xN matrix. N = 15 in this case.
- The first column is a 2xN matrix, which is points1. Points1 are the 15 corresponding points selected from the image wdc1.jpg.
- The second column is a 2xN matrix, which is points2. Points2 are the 15 corresponding points selected from the image wdc2.jpg.

'selectPoints_Dorm.npy', 'selectPoints_Frame.npy'
- 'selectPoints_Dorm.npy' is a 2x2xN matrix containing the corresponding points selected from 'dorm1.jpg' and 'dorm2.jpg'
- 'selectPoints_Frame.npy; is a 2x2xN matrix containing the corresponding points selected from 'content.jpg' and 'frame.jpg'
- points selected and saved using 'selectPoints.py'

'content.jpg', 'frame.jpg'
- 'content.jpg' is the image to be warped and put into a frame region in the 'frame.jpg'
- The frame in the 'frame.jpg' is the laptop screen

'dorm1.jpg', 'dorm2.jpg'
- 'dorm1.jpg' and 'dorm2.jpg' are the two self-taken images to be stitched together

'progressBar.py' (*all usage commented out in 'warpImage.py'*)
- helps to print a progress bar in command line as the warping process is pretty slow, so the user knows how the process is going

- very buggy when running on Mac small-size terminal, but seems fine when running in PyCharm terminal or maximized screen Mac terminal