

[My Workspace](#)[CS-6210-001 FALL14](#)[CS-6210-001 SPR15](#)[CS-6250-001 SUM15](#)[CS-6475-001](#)

ASSIGNMENTS

Engineering

[Home](#)[Syllabus](#)[Announcements](#)[Resources](#)[Assignments](#)[Gradebook](#)[Email Archive](#)[Roster](#)[Site Info](#)[Section Info](#)[Peer Feedback](#)[Schedule](#)[Tests & Quizzes](#)[Piazza](#)[Help](#)

Assignment #8: Panoramas - Returned

Title	Assignment #8: Panoramas
Student	Tran, Ngoc T
Submitted Date	Oct 19, 2015 12:46 am
Grade	93.0 (max 100.0)

Instructions

Assignment 8: Panoramas

Introduction

In this homework assignment, we will focus on the core algorithms behind building a panorama.

Before we get started, download the assignment files under Resources > Assignments > Assignment 8 that contains the files you wish. It also contains the assignment description if you wish to view it offline.

Once you have extracted the zip file above, you are ready to get started.

As with previous assignments, running `assignment8_test.py` directly will apply a unit test to your code and print out helpful feedback. Use this to debug your functions.

Part 1: Programming the core of panoramas.

This is a quick highlight of the functions. `assignment8.py` has extensive function descriptions with step by step guidelines.

1. `getImageCorners`

For an input image, return its four corners.

2. `findMatchesBetweenImages`

Use your implementation from A7 for this function. We will not deduct points if you got this function wrong in A7 (we will use our implementation if yours does not work).

3. `findHomography`

Returns the homography between the keypoints of image 1, image 2, and its matches. The unit test for this function is in `t_assignment8.py`.

testing script may have value differences and thus may not pass. If your output warped image looks fine, don't worry too much about this test.

4. `warpImagePair`

Warps image 1 so it can be blended with image 2 (stitched).

5. `blendImagePair`

For Part 1, you can leave this implementation as is.

Part 2: Get Creative with Programming and Photography

`blendImagePair`

This is what we want you to modify. We want you to write a function that blends the warped image with the second image at a simplest implementation is to average pixels where the images overlap. If you do this, try to incorporate a weight function from overlap point so that you get a 'fade' effect and properly transition from image 1 to image 2. You cannot modify the arguments c

We want you to be creative here. Good blending may be difficult and time-consuming, and we do not expect you to implement perfect and seamless blend. We want to see good effort and creative solutions. Feel free to compare and discuss different high approaches with other students.

Taking Panorama Images

We want you to take 3 images (feel free to take more and test your output) for this panorama. Keep in mind that planar panorama is better to visualize than rotational ones, you'll see this in the output, either is fine but you may want to look into the different panorama types for a better output.

The Writeup

This is what we want you to do for the PDF.

1. Demonstrate your 3+ Input Panorama images
2. Demonstrate the Panorama result.
3. Explain how you implemented your `blendImagePair`. We don't need you to explain the other functions but feel free to do so.

Answer the following questions:

- What effect did increasing the number of matched features have on your output? You can modify it either in the test file at the bottom of `assignment8.py` where we provided some commented out code. Just make sure it is commented out before submitting.
- What type of panorama did you take?
- Were you happy with your result? If so / If not, why?

What to turn in:

Please turn in the following files. **Keep the size limit of each file to 6MB.** If you need to compress your PDF before submitting, use <http://smallpdf.com/compress-pdf>:

- **assignment8.py** - Your code.
 - **assignment8.pdf** - See above for the writeup, don't forget to include your input and output images in the PDF (only in th
-

Submitted Attachments



[assignment8.py](#) (15 KB; Oct 18, 2015 11:15 pm)



[assignment8.pdf](#) (2 MB; Oct 19, 2015 12:38 am)

Additional instructor's comments about your submission

Programming section (68/70):

getImageCorners: 10/10

findHomography: 20/20

warpImagePair: 20/20

blendImagePair: 18/20

Writeup (25/30)

Why do we multiply x_min and y_min by -1? (-5)



Copyright 2003-2011 The Sakai Foundation. All rights reserved. Portions of Sakai are copyrighted by other parties as described in the Acknowledgments screen.
T-Square - gatech-sakai-2-8-x-10 - Sakai 2.8.x (Kernel 1.2.5)- Server pinch3.lms.gatech.edu