Auto-Stitching Multiple Images to Generate Panoramic Mosaic Images

Due Date: 23:59 on Monday, May 20th, 2013

Figure 1: The Rememberence Forest of Hacettepe University Computer Engineering Department: (a)-(c) taken with different view of the same scene by panning with smart phone; d) Stitched by following steps given in this assignment.

Assignment adapted from [1]

Background

Image stitching is one of the most attractive computer vision task. Two or more images taken with different view of same scene are combined to obtain panoramic image by using image processing techniques. New generation cameras present this utilization to the users to make panoramic photograph seamlessly in high resolution. There are various problem needs to be solved to achieve successful image stitching. The images taken for a scene with different view by panning the camera has common regions corresponds...
to same region in the real scene as seen in figure 1. To combine two images, these overlapping regions must be specified accurately and then these overlapping regions must be blended seamlessly.

**Overview**

The goal of this assignment is to practice basic image stitching steps. For this goal, you will complete code are given you and make observations on photos you will take.

**Details**

Your programming assignment consists of the following two parts:

To stitch two images, the following basic steps must be realized.

1. **Interest point detection**
   You will use Harris corner detection code given to specify interest points for each images.

2. **Feature extraction on interest points**
   You will implement a feature extraction function that calculates features of image patches for each interest point and returns features with their locations.
   
   \[ \text{Features, xPos, yPos} = \text{featureExtract}(\text{Image}, \text{InterestPoints}, \text{PatchSize}) \]

   You will extract feature with the following steps:
   
   (a) Blur the image with a Gaussian filter.
   (b) For each interest point, take the intensity values of the patch surrounding that point as features.

3. **Feature matching**
   
   (a) Specify interest points for each image
   (b) Extract features on interest points for each image will be stitched by using your function.
   (c) Calculate Euclidean distance between interest points of two images.
   (d) Sort the distances and use D. Lowe’s ratio test given in the starter code.
   (e) Take only points that have ratio under the threshold you will specify in range 0-1.

4. **Projection**
   After you specify matched points between two images, you will estimate the homography matrix by using Ransac code given.
   
   \[ \text{Homography, Inliers} = \text{ransacFithomography}([xposIm1; yposIm1], [xposIm2; yposIm2], \text{inlierThreshold}); \]

5. **Blending**
   Lastly you will warp and blend images by using two way blending steps.

Some important remarks:

- You must show each step clearly in your report.
- You will take own photos to generate mosaic images.
- At least 3 mosaic image for different scene must be in your report.

**Grading**

The assignment will be graded out of 100 points:

- 0 (no submission), 20 (an attempt at a solution), 40 (a partially correct solution), 60 (a mostly correct solution), 80 (a correct solution), 100 (a particularly creative or insightful solution)

Note: Preparing good report is important as well as your solutions!
What to Hand In

You are required to submit all your report along with a short webpage in HTML. For that purpose, prepare a folder containing

- HTML/README.txt (text file containing details about your project)
- HTML/code/ (directory containing all your code)
- HTML/ (directory containing all your documents, including your images)
- HTML/data/ (including your data images)
- HTML/result/ (including your result images)
- HTML/index.html (html report)

Archive this folder as studentid_pset3.zip and send to karacan@cs.hacettepe.edu.tr.

Each student must individually do the coding and prepare detailed report which contains a brief overview of the problems, details of implementation and the results with comments. All results you obtain must be put to your report and observations must be specified clearly. If your implementations failed to give a satisfactory results, provide a brief explanation of the reason(s).

Policies

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work (from internet), in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

References