COS 429 PS3: Stitching a Panorama

Due November 4th
Goal

• Find key features in images and correspondences between images
• Use RANSAC to find the best correspondences
• Map one image plane to another to create a panoramic image
Problem 1: Preprocessing

• Most feature descriptors only work with grayscale images

• Task:
  – Convert color images to grayscale
    • You can use Matlab function `rgb2gray`
  – Copy the lines of code you wrote in the report
Problem 2: Detecting Key Points

• Want to detect the key points in both images and find corresponding key points between both images

• Task:
  – Find SURF features in both images
    • You can use Matlab function `detectSURFFeatures`
  – Copy the lines of code you wrote in the report
Problem 3: Extracting Descriptors

• Extract feature descriptors at each key point detected in Problem 2

• Task:
  – Extract feature for each key point
    • You can use Matlab function `extractFeatures`
  – Visualize the descriptors and include in the report
  – Copy the lines of code you wrote in the report
Problem 4: Matching Features

• Task:
  – Find matching features between both images
    • You can use the Matlab function `matchFeatures`
  – Visualize the matching results and include a figure in your report
  – Copy the lines of code you wrote in the report
Problem 5: RANSAC to Estimate Homography

• We want to exclude outlier matches and compute a homography to map one image plane to the other

• Task:
  – Use RANSAC to estimate a homography
  – You can use Matlab function \texttt{estimateGeometricTransform}
  – Visualize the matching results and include \texttt{in your report}
  – Copy the lines of code you wrote \texttt{in the report}
Problem 6: Stitching Panorama

• Need to warp images to make a panorama
  – Map pixels in the warped image to pixels in the input image to avoid holes in the final image
  – Code provided to warp the first image

• Task:
  – Write similar code to warp and paste the second image to produce a final panoramic image
    • You can use MATLAB functions `imwarp`, `vision.AlphaBlender`, and `step` to overlay the second image on the first
  – Add the resulting panorama to your report
  – Copy the lines of code you wrote in the report
• Notes:
  – Don’t worry about blending (visible seams)
  – Results will vary since RANSAC is a randomized algorithm
Problem 7: Take Your Own Pictures for Princeton Campus

- Task:
  - Take two pictures of Princeton’s campus, run the code to stitch them together
  - Include the original two photos and the final panorama in your report
Extra Credit

• Many possible ways to get extra credit:
  – Try alpha blending to merge the overlapping image regions to get rid of boundary
  – Use Graph Cut to find an optimal seam between the two images
    • Use Poisson blending to blend the two images
  – Handle more than 2 images
  – Combine photographs into a 360° x 180° panorama (equirectangular projection)
  – Convert the panorama into a stereographic projection
  – Reconstruct the 3D geometry of the panorama
What to Submit:

• One PDF file report
• One ZIP file containing all the source code, and a “ps3.m” file that takes no parameters as input and runs directly in Matlab to generate the results in the pdf report