Department of Computer Science, University of Bristol

COMS30068: Image Processing and Computer Vision, AY: 2022/23

Assignment Part 2: 3-D from Stereo

Andrew Calway, andrew@cs.bris.ac.uk
Yuhang Ming, yuhang.ming@bristol.ac.uk

This part of the assignment assumes that you are familiar with the 3-D simulator used in Lab Sheets I and II and that yoiu have completed all of the tasks in those lab sheets. The assignment involves determining corresponding points in two views of a scene and using the correspondences to reconstruct 3-D points and spheres in the scene. You will be required to submit a report describing the tasks completed and analysing the results. You are also required to submit your code. Both are required to obtain marks. You are strongly encouraged to keep copies of all code and results.

1. Download and run the assignment version of the 3-D simulator used in lab sessions. Refer to Lab Sheet I for details.

2. When you run the simulator you should see 6 spheres of different sizes located on a plane. You should also find two images in your current directory corresponding to the images captured by two virtual cameras (VCs). Each time you run the code you will see a different arrangement of the spheres and images captured from different viewpoints. You should also be able to switch to a second visualisation which shows the plane and centres of the spheres.

3. Use the openCV function [INSERT] to detect circles in each VC image. Check and document the results, noting any dependencies on any parameter settings.

4. Select one of the VCs as the reference view and for each detected circle centre in its image, compute the corresponding epipolar line for the other image. Draw the line and check that it is correct.

5. Hence use the epipolar line for each detected circle centre in the reference view to find the corresponding circle in the other view.

6. For each circle correspondence, compute the 3-D location of the associated sphere centre using the 3-D reconstruction algorithm described in the lectures.

7. Display the estimated sphere centres alongside the ground-truth centres in the visualisation. Compute the errors in the sphere centre estimates. Note: think carefully about how you should compute the errors.

8. Using the circle detections in the reference and viewing images, estimate the radius of each sphere.

9. Hence display the estimated spheres alongside the ground truth spheres and compute the error in the radius estimates. You will need to think carefully about how you display the estimated and ground truth spheres so as to allow comparison, since comparison

may be difficult if you display the estimates as solid spheres. For example, you may want to draw a single circle to represent each sphere to allow comparison.

10. Test and evaluate the performance of your implementation on multiple instances of running the simulator. Ideally it should be successful for every case. If not, note when it does not work and try to find out why.

Submission

1. **Code** that is clearly commented, explaining the key components and the functionality. Note: the code will be tested and so please make sure that any required inputs and outputs are clearly explained at the top of the code.

2. **Report**, detailing, discussing and analysing the tasks completed and the results obtained, including example visualisations and showing quantitive and qualitative results of the estimation and any failure cases. This should be a maximum of 4 pages. There is no set format for the report - you should present the material as you see fit.

Marking Scheme

This part is worth 50% of the CW mark for the unit. Marks will be allocated as follows. For components 1-3, marks will be based on the content of the report w.r.t description, analysis and evaluation and on the code.

1. Sphere correspondence - 15%

2. Centre reconstruction - 15%

3. Sphere reconstruction - 10%

4. Overall standard of presentation, analysis and evaluation - 10%