**COMS30068: Image Processing and Computer Vision Coursework 2022 Part 2**

Submission by Ben Browne – nm20529

**Component 1 – Sphere Correspondence**

First we needed to detect circles for each View Camera image. Cv2.HoughCircles() was used with the following parameters:

Chart, bubble chart

Description automatically generatedThese parameters detected all circles with relative accuracy on subsequent runs. Param1 – the threshold for edge detection resulted in false edges when <~10, param2 – the circle vote threshold resulted in false circles <~25. A balance was required between these 2 parameters to ensure all circles were detected and radi were accurate.

Figure 1.1 – Hough Circles

In order to take points in one image and calculate the corresponding epipolar lines in the other, the *Essential* and *Fundamental* matrices had to be calculated, which both encode the epipolar geometry between both cameras.

The Essential Matrix E, is a 3x3 matrix that relates corresponding points in stereo images: x’T · E · x = 0 where x’ is the homogeneous coordinates of point in image 2 and x is the homogenous coordinates of point in image 1.

The Fundamental Matrix F, is a 3x3 matrix that also relates corresponding points in stereo images, but can be applied to non-canonical cameras where as E only applies to canonical/calibrated cameras, where the camera intrinsics K = K’ = I.

Shape

Description automatically generatedOur setup requires the computation of the Fundamental Matrix, which in turn requires the computation of the Essential matrix.

Figure 1.2 describes the epipolar geometry setup between the two cameras. To calculate E, the relationship between the 2 cameras [R (Rotation) , T (Translation)] must be defined. We can do so by finding HRL = , which allows us to calculate S = and finally **E = RS**

Figure 1.2 – Epipolar Geometry

Now we calculate **F = MTR · E · ML**, where MR = ML = K.intrinsic-1

Chart

Description automatically generatedBy analysing the image from our left camera (H1), we can create a list of detected circle centres which using the fundamental matrix, calculate the corresponding epipolar line in the right image l’ = . Shown in Figure 1.3, this gives us a 1D ‘search window’ for the corresponding point in the right image.

Finding the corresponding circle centre is then trivial, traversing the epipolar line until a circle centre is found. Tolerance was applied when searching for centres to account for slightly off lines.

Figure 1.3 – Corresponding Image Points

Diagram

Description automatically generated with medium confidenceIt was later discovered that due to the randomness of the sphere generation, on rare occasions, an epipolar line can lie on the centres of 2 spheres, which can result in the ‘wrong’ circle centre being added. This can be mitigated by not adding a centre already in the set and continuing down the line or doing sweeps from either side of the line to ensure no centre was missed.

Figure 1.5 – ‘Right Camera H0

A picture containing stationary, businesscard, envelope

Description automatically generated

Figure 1.4 – ‘Left’ Camera H1

**Component 2 – Centre Reconstruction**

We now should have 2 sets of circle centers – the reference set and coresponding set. Checks are carried out to ensure both are of size 6 before continuing (On the rare ocassion of failure SystemExit(0) is called).

Points must now be matched up into pairs using the equation: **prT · F · pl = 0** (actual values lay ~0.01)

In order to reconstruct each centre in the 3D space the following equations must be satisfied:

Finding a, b, c s.t: **a[pL]– b[RTpR] – c[pL × RTpR] = [T]**

Diagram

Description automatically generatedEach [] is a 3x1 vector which can be used to form matrix H where: **H · [abc]T = T**

Leading to our solution **[abc]T = H-1 · T**

Figure 2.1 shows the geometric motivations behind these equations: where we project our corresponding image points into the 3D space (in the left coordinate system) to find the orthogonal line: [pL × RTpR]. Having 2 representations of the same line means we can set the difference to 0 and rearrange to form the equations above.

Figure 2.1 – 3D Reconstruction

Solving for a and b means we now have the scalars for both line projections from each image point, the average of these 2 vectors gives us our centre estimate in the 3D space: **p3 = (apL + bRTpR + T) / 2**

The c value is the distance between our 2 projections at p3 (our uncertainity), a low c therefore represents a good correspondance.

Figure 1.1 – Hough Circles

Figure 1.1 – Hough Circles

https://web.stanford.edu/class/cs231a/course\_notes/03-epipolar-geometry.pdf