Project 2 – Visual Odometry – Report

1) Important ideas.

a) **Demosaic**. Restore the color images from Bayer format input images using GBRG alignment.

b) **UndistortImage**. Reduce the distortion.

c) Using **imgaussfilt** to denoise images.

d) Using **detectSURFFeatures**, **extractFeatures**, **matchFeatures** to get the matched positions between the current image and the next image.

e) Using **RANSAC** algorithm to eliminate outliers, only preserve the perfectly matched points.

f) Using **the normalized 8-point algorithm** to calculate the fundamental matrix F and the essential matrix E.

(Book: *3D Reconstruction with two Calibrated Cameras*, chapter 9.6)

Since $x'^T Fx = 0$, we could transfer to another denotation:

$$Af = \begin{pmatrix} x_1'x_1 & x_1'y_1 & x_1' & y_1'x_1 & y_1'y_1 & y_1' & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n'x_n & x_n'y_n & x_n' & y_n'x_n & y_n'y_n & y_n' & x_n & y_n & 1 \end{pmatrix}$$

The least-squares solution for f is the singular vector corresponding to the smallest singular value of A, that is, the last column of V in the SVD (A) = UDV^T. The solution vector f found in this way minimizes $\|Af\|$ subject to the condition $\|f\| = 1$.

Then The essential matrix is found by $E = K'FK$. While K is the camera intrinsic matrix.

g) Extract the rotation and translation matrices from the essential matrix E.

(Book: *3D Reconstruction with two Calibrated Cameras*, chapter 11.1)

Essential matrix property*: A 3x3 matrix is an essential matrix if and only if two of its singular values are equal, and the third is zero.*

Note [U, S, V] = svd(E). We should reconstruct the E using $\mathrm{U}\ \mathrm{diag}(1,1,0)\ V^T$ and use the svd function again to update U and V.

Given $W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $Z = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

Property: For a given essential matrix $E = \mathrm{U}\ \mathrm{diag}(1,1,0)\ V^T$, and first camera matrix P = [I | 0], there are four possible choices for the second camera matrix P’, namely $P' = [UWV^T|{+}\mathrm{u}_3]\ or\ [UWV^T|{-}\mathrm{u}_3]\ or\ [UW^TV^T|{+}\mathrm{u}_3]\ or\ [UW^TV^T|{-}\mathrm{u}_3]$.

h) 3D reconstruction of the matched points to select correct solution from the 4 possible camera matrices P’.

(Paper: *Triangulation*, author: *Richard I. Hartley and Peter Sturm*.)

Using the mid-point method discussed in this paper. The camera matrix P can be dissected to $(M| - Mc)$, while c is the camera center position. The infinity maps to $M^{-1}u$. Therefore, random point maps to $c + \alpha M^{-1}u$. Then we can get a equation $\alpha M^{-1}u - \alpha' M'^{-1}u = c' - c$ because the 2 rays intersect in space.

Finally, the mid point between the two rays is then given by $(c + \alpha M^{-1}u + c' + \alpha' M'^{-1}u)/2$.

i) Plot the trajectory using only the x and z value. The z value to show the forward movement and the x value shows the turns of the car.

2) Abandoned Ideas.

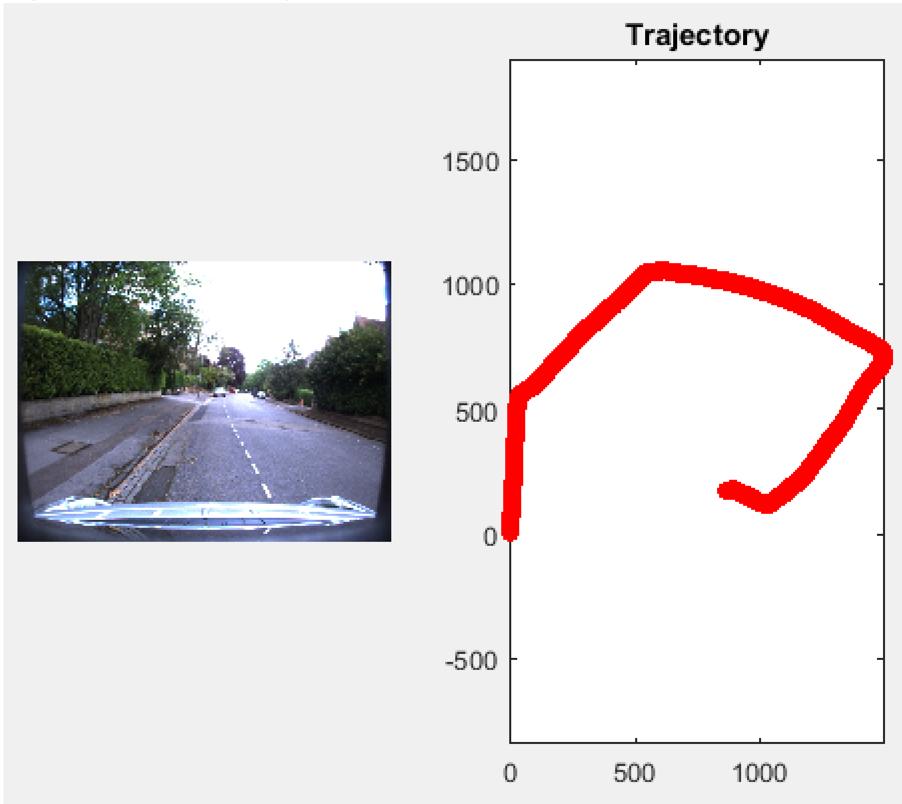
a) 3D reconstruction using *Carlo Tomasi*'s algorithm in the paper *3D Reconstruction with two Calibrated Cameras.*
The results match the video perfectly until the second turn. All the plotting is reversed from that turn – left and right. Do you have any idea on why?
Paper link:
https://pdfs.semanticscholar.org/398a/05ba68dfc145164fb932dd4c251896d71174.pdf

3) Results.

a) The final plot is from the 200th to the last image. The software I coded cannot solve some of the first 200 images due the saturation. Since the **detectSURFFeatures** cannot detect more than 8 points in an image, the rest of the algorithm is not working any more.

b) The computer vision toolbox result is shown below:

The plotting result based on my code is shown below:



My plot is worse in the first and second turn than the toolbox result, while it performs the same for the 3rd turn, and much better in the 4th turn. As you can see, my result actually shows the right turn after the waiting, but the toolbox result shows something wrong.