Assignment Three: Stereo Vision and Visual Odometry[1]

Haokun Geng
hgen001@aucklanduni.ac.nz

Submit a **report** about your work for this assignment by the due date (as announced in lectures). This assignment will contribute 10% towards your final marks. Solutions and results may be presented in seminars.

[1] Computer Science 775, Semester 2, 2014, Tamaki Campus

Part One: Stereo Vision

Figure : Stereo image pair with a ground truth (color-coded) disparity map.

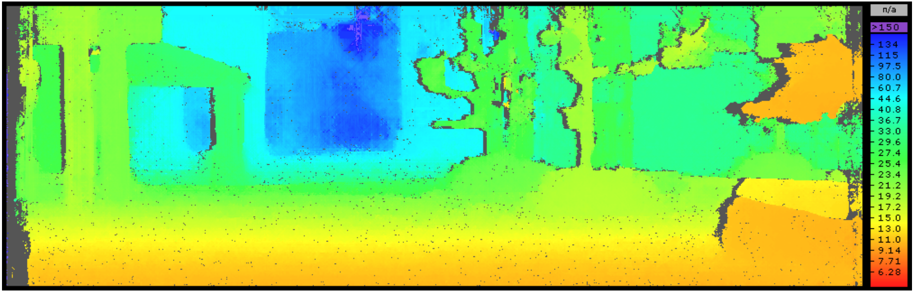
This part requires you to implement a stereo matcher for rectified image pairs. Rectified stereo-image data sets can be downloaded from EISATS, Set 2: `ccv.wordpress.fos.auckland.ac.nz/eisats/set-2/`

Part One: Stereo Vision

- Rectified image pairs are provided in two sequences in Set 2 of EISATS (see link above). Select one stereo pair (at a time) as the input of your program. You could use `OpenCV` functions to generate your own disparity map. Compare your result to the corresponding ground truth (suggestions: compare depth maps visually, calculate a map of errors, or calculate a histogram of disparity differences - design your way for comparing). Report any settings and parameters that are used in your program.
- Use your program to generate a disparity map for an outdoor environment (e.g. use real-world data from one of the sets on EISATS). Report settings or parameters changes in your program for getting the “best” disparity map for the given out-door stereo image pair.

Part One: Stereo Vision

- (**Optional**) In order to enhance the visual presentation of your disparity maps, you might like to add colours to your disparity images to indicate different distances. Read and understand the color-coding project on `ccv.wordpress.fos.auckland.ac.nz/data/sources/`. Your output should be something similar to the example colour-enhanced disparity map shown below.



Part Two: Visual Odometry

This part requires you to implement a program for stereo-vision based visual odometry (VO). The input is a sequence of rectified stereo pairs L_k and R_k, for $k = 0, \ldots, T$, and the corresponding sequence of disparity maps D_k, calculated with the stereo matcher of your choice in Part One.

The output of this program for pair k should be a 4×4 transformation matrix $\mathbf{H}_k$ (in homogeneous coordinates) which combines a 3×3 rotation matrix $\mathbf{R}_k$ and a 3×1 translation matrix $\mathbf{t}_k$ (in inhomogeneous coordinates), and the updated camera position $\mathbf{C}_k$ (in homogeneous coordinates).

At the beginning, the camera system is at $\mathbf{C}_0 = [0, 0, 0, 1]^\top$.

Part Two: Visual Odometry

A suggested VO method is given here:

1. Capture two stereo image pairs L_{k-1}, R_{k-1} and L_k, R_k.
2. Extract and match features between L_{k-1} and L_k (note: matching is not stereo-matching here).
3. For the matched features, derive 3D coordinates at $k-1$ and at k.
4. Compute $\mathbf{H}_k$ (being the combined $\mathbf{R}_k$ and $\mathbf{t}_k$) from the two sets of 3D features.
5. Update the camera position by $\mathbf{C}_k = \mathbf{C}_{k-1}\mathbf{H}_k$.
6. Repeat until end of the sequence (i.e $k = T$) is reached.

Part Two: Visual Odometry

- Camera recording comes with noise in the input images. Do some research and suggest a way to minimize the impact of noise for your VO program. Describe the selected method in your report.
- (**Optional**) 3D point-cloud reconstruction is a way to exam the accuracy of your estimated motion data. In order to demonstrate your VO program, you could visually present a 3D reconstruction for the used stereo sequence and the used stereo matcher.

To finalize your report,

- describe your research for both parts on about 2-3 pages,
- include samples of outputs of your program into this report on 1-2 additional pages, have 2-10 references included,
- do not copy from somewhere without proper citation and reference, but aim at writing in your own words;
- finally submit your report in PDF format (6 pages at most).