EE/CprE/SE 4910 WEEKLY REPORT #6

4/2/25 - 4/8/25

Group number 15

Vision Based Camera Motion Tracking

Advisor: Ashraf Gaffar

Team Members:
Andrew Gooding
Eric Wittrock
Isaac Kenyon
Will Ernatt

Weekly Summary

This week we furthered our research on camera solving algorithms and continued development on our camera solving solution. We also began development on our error detection and correction programs to detect and remedy accuracies in the generated camera path. Some of this week's work was dedicated to setting up our code to test on real data next week.

Past Weekly Accomplishments

Eric Wittrock

- Worked on gradient-descent-based camera solving algorithm
    - Ported solution to python
    - The algorithm now handles geometry and camera path separately so they can be solved using different hyperparameters if need be
- Optimization
    - Worked on mathematics to try to express the projection of all points simultaneously. Large matrix multiplications are faster in python than the iterative small operations of the current solution

Isaac Kenyon

- Wrote a dense optical flow estimation algorithm and used openCV components to get it implemented.
    - Faster than what I currently had implemented
    - It is better for slower movement, I tested with a larger jump and it did not seem to like that.

    - From reading online it says that this algorithm is prone to drift. Such that the object ends up shifted across the scene from where it was originally placed.
- Looked into direct methods of camera motion calculation, and started working on an implementation
    - Supposed to work well with low features in rooms. Like a white room with only a few objects that could be used for tracking.
    - It does not handle light exposure very well though, if a light were to be turned on or off, rather than a consistent source it would mess up the algorithm.

Will Ernatt

- Wrote a feature matching program using OpenCV using the ORB algorithm
    - SIFT is used by AliceVision, which I was modeling these programs after previously, however ORB is an order of magnitude faster.
- Researched ORB-SIFT algorithm as a replacement for StructureFromMotion
    - ORB-SIFT is typically used for robotics applications(Visual Odometry) but has camera localization capabilities, which could be utilized for our applications. If it produces accurate results it could completely replace StructureFromMotion. This would be ideal as ORB-SIFT is far more promising in its ability to meet our performance goals. SfM's sluggishness is good enough for its typical photogrammetry applications, but we are searching for something that can provide quick and relatively accurate results.

Andrew Gooding

- Finished creating a script that will automate the process of uploading a video to blender and complete the camera tracking process. This includes tracking process and solving process.

<u>Pending Issues</u>

- **Eric Wittrock:** Overcoming performance issues is a work in progress.
- **Isaac Kenyon:** Get the new implementation fully working for the dense optical flow and direct methods.
- **Will Ernatt:** Need to push code to GitHub repository
- **Andrew Gooding:** Just cleaning up the script so that it can find and eliminate errors.

Individual contributions

Name	Individual Contributions	Hours This Week	Hours Cumulative
Eric Wittrock	Refactored, made minor modifications to, and optimized the camera solving algorithm	4	38
Will Ernatt	Wrote OpenCV feature matching script(ORB). Researched ORB-SIFT as replacement to SfM algorithm.	6	37
Andrew Gooding	Finished blender script to automate the tracking and solve process of blender camera tracking feature.	6	30
Isaac Kenyon	Wrote a dense optical flow estimation algorithm and used openCV components to get it implemented. Looked into direct methods of camera motion calculation, and started working on an implementation	7	35

Plans for the upcoming week

Andrew Gooding

- Refine the script made this week so that it is better at finding and eliminating errors.

Eric Wittrock

- Test the camera solving algorithm on a small subset of the tracking points from real footage

Isaac Kenyon

- Continue working on the algorithms implemented and testing for the pros and cons of each, along with the speed of the algorithms.

Will Ernatt

- Implement ORB-SIFT with OpenCV
- Push code to GitHub
- Test output with Blender