EE/CprE/SE 4910 WEEKLY REPORT #3

3/5/25 - 3/11/25

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 have continued to research other softwares that can do things similar to what we are looking for to see if they may be used for our project. We have also all created environments so that we will be able to start running test scripts for our project.

Past Weekly Accomplishments

Eric Wittrock

This week was dedicated to experimentation with pre-trained depth estimation models, taking into account model size, granularity, and persistence of output across multiple frames. The Input and output of the model are shown in the image below.
The depth estimation model will not replace depth estimation via parallax, but it will aid it, especially in cases where the camera doesn't move much.

To accompany this new approach, I worked on the math for plotting the points in 3d space in a consistent manner, but haven't yet implemented this theory. This alone can be used to get a set of crude camera positions.

Isaac Kenyon

- Got the main GUI window for blender working and the ability for different UI elements like a slider, value input, etc.
- Working on an error detection function for optical flow.
    - Implements the ability to set a threshold for removing errors, could work well if camera speed is able to be calculated.
    - Another parameter that can be set to attempt to automatically detect when there is an error and remove it without having to set a threshold.

Will Ernatt

- For this week I explored OpenCV and its Structure from Motion algorithms to see whether or not it is feasible to implement into our final project.
    - Set up OpenCV environment (in linux).
    - Implemented a basic OpenCV camera path reconstruction algorithm(in C++) that takes in a set of 3d tracked points to recreate a camera path. I had to use existing data for these tracked points as I have not yet written an algorithm to do this (and the alicevision algorithm did not produce something usable by OpenCV)
- Started setting up OpenCV environment and windows and writing an algorithm to accomplish the same task using python rather than C++ (due to its portability and ability to be used in blender plugins).

Andrew Gooding

- This week I continued working on blender scripting in blender so that I have scripts that can start automating processes for me.
- I will continue to do research on camera vision and other softwares that will be able to do what we are looking for. This will allow us to gain a better understanding of the competition out there and help us make our project that much better.
- I also started looking into things that Iowa State can provide us to help us with our project.

Pending Issues

- **Eric Wittrock:** The depth estimation model alone is not enough to solve for the camera motion, so I plan to use it as a crutch for the traditional algorithm. I have yet to figure out how to get these techniques to work well together.
- **Isaac Kenyon:** None
- **Will Ernatt:** I did most of my work in Linux this week because of struggles with OpenCV in Windows, I'll resolve those issues this week.
- **Andrew Gooding:** Some issues I am facing are making the environment I created work cohesively. This will help me reach the goal this week of creating a script that can automate some of the building camera vision process blender has included in it.

Individual contributions

Name	Individual Contributions	Hours This Week	Hours Cumulative
Eric Wittrock	Experimentation with pre-trained depth estimation models. Worked on mathematics for camera estimation using this depth parameter instead of parallax.	4	19
Will Ernatt	Wrote basic camera path reconstruction algorithm in OpenCV, taking tracked points as input.	6	19
Andrew Gooding	This past week I was away at a convention in Chicago so I was not able to work on the project as much as I would have liked, but I was able to start scripting in blender to start automating some processes.	4	13
Isaac Kenyon	Built a rough	6	18

	template for the Blender plugin. Messed around with different UI components, sliders, value inputs, etc. Started work on the import and export of data for users and debugging purposes.		

Plans for the upcoming week

Andrew Gooding

- Continue using blender to create a script that will automate processes in blender.
- I will also continue to find resources at Iowa State that will help our project succeed.

Eric Wittrock

- Plot estimated sparse point positions in 3d space. It may be possible to get a rough camera motion estimation from this

Isaac Kenyon

- Keep working on the optical flow error detection. Attempt to get the automatic detector working with multiple levels of error removal.

Will Ernatt

- Port my camera path reconstruction algorithm to Python if possible, and figure out how to transfer the camera path to Blender(currently only able to visualize using Viz).
- Set up OpenCV in Windows.
- Begin writing an algorithm to track 3d points using OpenCV.