Jul 27th, 3:15 PM - 4:15 PM

A Semi-Automated System for 3D Scene Analysis: From Detection and Tracking to Reconstruction

Anthony Trofe
Susquehanna University

A Semi-Automated System for 3D Scene Analysis: From Detection and Tracking to Reconstruction

Anthony Trofe

Department of Mathematics, Susquehanna University, Selinsgrove PA

Abstract

Detection and tracking of flags from a handheld video allows for the computation of distance between these flags in a three dimensional space. With this tool, scientific experiments requiring spatial arrangements indicated by markers would become easier to both map and determine the area.

The initial video, composing of different sets of flags laid out, is run through a detection program which provides locations of flags inside of the frame. From these coordinates, a second program is run, which determines tracking. The tracking works by looking to see if there are points in sequential frames that are closer than a threshold. If they are sequential and close, they are joined together to make a track. This track is built upon until there are no more points within a tolerable distance. Following tracking, the user is able to select which specific tracks they want. The reconstruction gives a three dimensional environment based on the flags that were detected.

Objective

The goal is to be able to develop a program that can track objects within a scene, such as flags, and then reconstruct this same scene within a three dimensional space to scale. This would then be able to work from a drones aerial footage. In total, the ability to reconstruct a scene within a three dimensional environment would allow scientists and researchers the ability to gather data from large plots of land more efficiently.

Methods

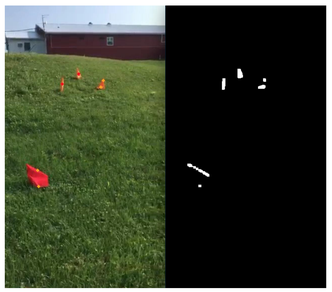

Figure 1.

Figure 2.

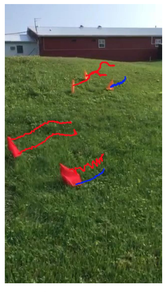

Figure 3.

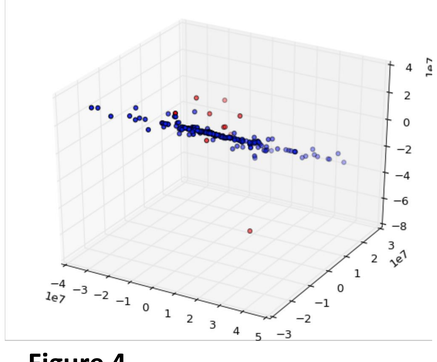

Figure 4.

In order to effectively create a three dimensional environment from a recording, multiple steps must be taken. The first is to turn the video into a series of JPG files. From these JPGs, the detection program is able to scan the images for corresponding HSV values. When the HSV values for a pixel are within the tolerable range, it is marked and added to a mask (Figure 1). After completion of the detection, tracking occurs. Tracking is done by referencing the masks for each frame, and connecting corresponding pixels that are within the range of each other. This connection is then shown by a line to indicate the travel of the object (Figure 2). Following tracking, the user is then able to select which tracks are to be used for reconstruction (Figure 3). With the tracks selected, the reconstruction can begin with the use of all the selected tracks. After completion of reconstruction, a virtual environment is shown, as seen in figure 4. The blue dots represent camera position while the red dots are flag locations. Some tweaking is still needed in order to correct bundle positioning.

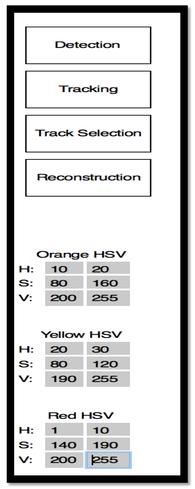

Figure 5.

The addition of a GUI allows for more efficient workflow without compounding all programs into one direct running one. The thresholds for color detection can be manipulated at the bottom then carried over into the detection. All that is needed for execution is a button press on the corresponding program.

Results and Conclusion

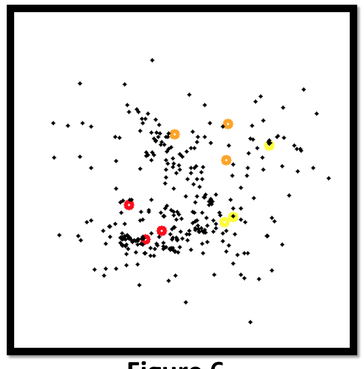

Figure 6.

Shown in Figure 6 are the results after running through the program. Three glaringly obvious problems arise: outliers, camera position, and flag location. Outliers can be explained through the second problem, camera positioning. The video shot was equivalent to a semi-circle of movement. That is not the case according to the reconstruction program. This inequivalence in camera location causes not only outliers but also sets off the resulting camera locations. In order for this to return acceptable results, more work needs to be done to both track and localize the flags. In the end, camera position should have no determinacy in flag location.

Acknowledgements

This work was supported by NSF grant CCF-1421734.