**Overview of Calculations**

This article will cover the “behind-the-scenes” or “heavy-lifting” aspect of the program. The process by which information is accepted/encoded and animations are generated/stored will be included, but the GUI and display aspects of the program will be saved for another article. Links to example and proof of concept code are encouraged where applicable.

This article should by lengthened and revised as our understanding of the topic improves and steps are tested.

\*As of the initial commit of this article, none of its content has been proven to work. This is a simple summary of the cs.carleton web pages as I understand them.

To initialize the program:

**Receive Camera Specs:**

This includes resolution, frame rate, location of focal point or image plane, and focal length.

**Receive Room Information:**

This includes camera positions relative to the room’s defined origin point.

**Build 3D Scene:**

Generate a 3D world with the focal points of the cameras placed in the correct position.

**Receive Camera Data:**

This is the actual video of the camera. Sound data is not needed.

**Sync Camera Frames:**

Cameras should have identical frame rates, but if not this must be adjusted by selecting relevant frames from higher speed cameras to match lower speed cameras. The lights in the room should be turned off, once all cameras are transmitting data. They may be turned on again if desired.

To synchronize cameras, the program will scan through the frames looking for the first instance where the brightness of one frame is some specified fraction or less than that of the previous frame. The program will make these frames occur at the same time and continue streaming from there.

**Identify All Points:**

This is mostly a user-interface function, but the information is vital to the program.

Continuously loop the following steps for every point and frame.

**Decide Which Cameras to Use:**

Find the first two cameras in the list to show the same point.

\*\*From this point I do not understand the process enough to summarize it. If you do, please add to this article.

**Track Specific Points:**

After calibration and the assigning of names to each marker per actor (actor cannot leave the frame after calibration), the system will keep track of each marker, storing its current location, previous location, and distance traveled. These three values will help the system determine which marker is which.