**Compiling the Code**

Though this step should not be necessary, to compile the code, navigate to the project daredevil folder:

/home/pi/projectDaredevil/

And run the compile.sh scripts in the following folders (in this order):

./sound\_library

./outlier\_detection

./virtual\_cane

**Calibrate the Device**

First, set up markers around the room of interest. For best results, set up markers in groupings of at least four (they don’t have to be in a 2x2 grid, but should be in very close proximity). While hooked up to a monitor (ideally), stand at the location of the chair. On the Pi, navigate to the virtual\_cane folder inside the repository.

1. Run the capture\_chair algorithm and point the device at a group of aruco markers.
2. Wait for the pose angle to be visibly correct (based on observations from the monitor)
3. Press spacebar to capture all aruco markers currently in sight\*
4. Repeat this step until all aruco markers have been captured.

\* Two things to note here: if you capture the same marker’s information twice, it will overwrite the old information with the latest information. The second thing of note is that the system will only record aruco markers it has seen for long enough to be “confident” in the data output.

**Run the project**

Set up the Pi to talk to your favorite Bluetooth audio device

Run virtual\_cane

Unplug any monitor interface and walk around