Team Name: sdmay24-27
Team Members:

1) Alexander Black
2) Jacob Burns
3) Jacob Lyons
4) Sergio Perez-Valentin
5) Sami Bensellam
6) Yogi Chander

Report Period: Oct 23-Nov 19

Summary of Progress in this Period

Hardware Progress:
We have completed a first loop, connecting all of our components in a rudimentary fashion. Our first loop was able to vibrate two motors depending on the vision of the camera, connected through the Raspberry Pi. We ordered several new hardware components, including a D435i depth camera which is suitable for our final product, two AdaFruit Pi hats, and improved the wires of the haptic motors for better testing and usability. We are continuing to do research on the compatibility of these hardware parts including voltage/amp requirements, testing parameters, and wire compatibility.

Software Progress:
More research has been put into figuring out an algorithm to detect the closest prominent object. This is important because, for each cell, we want a single depth point to represent the most important depth data for the user. Our first algorithm uses an OpenCV template matching, while the second algorithm we've developed uses a histogram-style algorithm. The template matching at the moment seems to produce better results, while the histogram produces more consistent ones.

Pending Issues

Hardware Issues:

- Our Raspberry Pi does not get enough voltage from a typical computer USB port. We must make sure the power from our battery is enough to power the Raspberry Pi and the rest of our machine.
- The Kinect, which was used for testing, is not sufficient for our final product due to its low range, limited software support, and large size.
- When the haptic feedback motors are vibrating at maximum intensity, the skin sensation of the vibration becomes less intense after a while of continuous maximum vibration.

Software Issues:

- Currently, our Python script that outputs a depth array taken from the Kinect data shows only two values to distinguish if an object is near or far. Currently, the only values shown in the depth array are -1 and 255, with -1 meaning an object is close by and 255 meaning an object is far. In our final design, we want the data to be more distinguishable than -1 and 255. This problem might be due to the Kinect's limited range of detection.

Plans for Upcoming Reporting Period

Hardware Upcoming:

- Finish first prototype of the sleeve thATby the end of Thanksgiving Break

- Test the sleeve to ensure the user is able to distinguish between the different motors.

Software Upcoming:

- Reconfigure the scripts to work with the Intel 435i instead of the Kinect
- Install all software and libraries needed for interfacing with the 435i onto our Raspberry PI.