4 Testing

Our Machine to Human Vision project contains many variables outside our control of feasible testing. Each user will be of different proportions, have different sensation receptors, and variaying levels of ability to understand the incoming information from the haptic motors. That said, the following testing methods will attempt to mitigate any variability that might be encounterd to procude the most linear results amongst users.

4.1 UNIT TESTING

1. **D435i Camera** - The camera will be tested to ensure it can generate accurate depth information with low latency and at a framerate of 30 fps. We will test it to ensure it registers important objects, such as a pole, even if they do not take up a significant portion of the camera's vision. This testing will be done using python script and having a team member validate the depth data retrieved by the camera.

2. **Raspberry Pi** - We will test the raspberry pi using a multitool to ensure it can power itself, the motors, and the camera at the same time. We will also test it to ensure it can operate at 30 fps, reliably vibrate the motors as desired, and accurately interpret the camera's depth information. This will be done via human validation.

3. **Haptic Motor Array** - The haptic motor array is the unit that will require the most testing. We will test different arrangements of the haptic motors, such as a pure 4 x 4 grid or a 4 x 4 grid that widens at the bottom in order to determine which is the easiest to navigate with. Effectiveness testing will be done by having a person wear the array and attempt to navigate using it. We will also test different fabrics and fabric thicknesses for the sleeve to find a design that is easy to navigate with but does not damage the wearers skin. We will test each motor to ensure they vibrate at the same strength using an accelerometer.

4. **Battery Pack** - We will test our battery pack using a multimeter to ensure it can generate at least 4.5 amps to power the Raspberry Pi, camera, and haptic motors.

4.2 INTERFACE TESTING

In our project, there is an interface where the stereo optic camera is connected to the hardware that controls the haptic feedback motors. In our current prototype, we are interfacing an Xbox Kinect and a Raspberry Pi 3. In our later prototypes, we will interface a real stereo optic camera and a Raspberry Pi 4. In order to test the interface between the camera and raspberry pi, there will have to be a physical connections test, where it is checked to make sure the camera is properly mounted onto the raspberry pi. There are also raspberry pi terminal commands that return whether or not there is a camera module connected to the pi which can be used for interface testing.

There is also a physical wire connection between the pi and the haptic feedback motors which will have to be tested. This will have to be tested through the software and the hardware. The software test will entail making sure the proper amount of voltage is sent to each haptic motor based on the data from the camera. The hardware components like the GPIO pins and haptic motors will have to be tested for proper wire connections by running current through the wires via a raspberry pi python script to make sure the haptic motors and GPIO ports are properly functioning.

Another interface that will need to be tested will be the connection between the transmission of data between the camera and the raspberry pi. A software test will have to be written where we know the depth of a set of objects in front of the stereo optic camera, and we must see if that data is properly received and processed within our raspberry pi program that handles outputs to the haptic motors. The GPIO pin outputs will also be tested by turning them on or off from a raspberry pi script.

4.3 INTEGRATION TESTING

Four main units have been identified as crucial during the integration process. Each unit plays a critical role in the overarching project goal, and undergo extensive requirement testing. They are D435i camera for vision, Raspberry Pi for compute, haptic motor array for feedback, and battery pack for power.

The integration process can be broken down into three critical integration paths. No specialized tools are needed to ensure proper integration outside the assumed.

1. **Compute Vision** - The compute vision path seeks to integrate the D435i Camera module with the Raspberry Pi. The camera provides real time depth maps and captures that allows us to map an environment, while the Raspberry Pi provides the compute power to process the data. The two units will be connected with a USB C cable, providing 5 Gbit/s signaling rate over 1 lane using 8b/10b encoding (nominal data rate: 500 MB/s). The physical connection ensures a quick and reliable connection despite external factors. The D435i Camera scans will all be processed through software on the Raspberry Pi. Therefore, the main form of testing will be performed through software unit tests. Main unit tests that will be used involve:

   (a) Receiving a correctly formatted data sequences from the D435i Camera.
   (b) Splitting depth maps into a cell grid to represent the Haptic Motor Array.
   (c) Converting a cell into a single representative depth point.

   Other unittests that may be required are:

   (a) Compression of depth maps for simpler processing.
   (b) Depth map formatting of values.

2. **Feedback from Compute** - The feedback from compute path seeks to integrate the Haptic Motor Array with the Raspberry Pi. The Raspberry Pi provides a formatted frequency response array, while the Haptic Motor Array relays those frequencies in a physical manner to the user. The two units will be connected physically to provide both power and data through a single wire. The software aspect will be assumed correct, as the Compute Vision path will test the Rasberry Pi to always provide a correct frequency response array. Therefore, the main form of testing will be performed through hardware unittests. Main unittests that will be used involve:

   (a) Stable connection between motors and board.
   (b) Instant motor frequency changes based on incoming data from Raspberry Pi.
   (c) Secure mounting of components to a wearable vest that allows for the previous.

3. **Power System** - The power system path seeks to integrate each of the other units to the battery pack to provide mobile power. Each unit requires a unique power output and connection. Therefore, the main form of testing will be performed through hardware unit tests. Main unit tests that will be used involve:
   (a) Battery endurance tests to ensure prolonged use during high compute periods.
   (b) Correct amps and voltage output for each unit.
   (c) Overheating and safety testing.

4.4 SYSTEM TESTING

1. **Unit Tests** - As mentioned in 5.1, we will test the camera, motor array, battery, and the Raspberry Pi to ensure they all work. Unit tests can also be used to ensure the end-to-end operation of the system. One way to do this is by comparing the final vibration levels of the motor (both by feeling them and looking at the quantitative vibration data in the program controlling the motors) to different environments that the camera is looking at to ensure that they all correspond.
2. **Interface Tests** - We will test the outputs of our conversion of the depth data and the and verify that it is the same as the inputs to our haptic motor script to ensure that the data is being transmitted correctly. In addition, we will test that each component is receiving the voltage that it needs by using a multimeter.
3. **Integration Tests** - One way we will test the integration of the system as a whole is by trying to use the system as our target audience would, by attempting to navigate in a room (and other places, such as outdoors, through a hallway, etc) with the device as opposed to our eyes. This ensures the device meets requirements by way of us knowing that the device will be useful to its target users.

4.5 REGRESSION TESTING

1. **Regression Test Cases** - We will develop a comprehensive set of regression test cases that cover all critical features. These test cases will ensure that the depth data is accurately converted and sent to the haptic motors at the correct intensity.
2. **Continuous Integration** - Introducing a continuous integration pipeline will ensure that no faulty code will be committed to our GitLab repository. CI allows us to run automated regression tests whenever there is a change in the repository. This will help to mitigate the risk of newly added code breaking any currently operational functions.
3. **Data Input/Output testing** - Confirm that the Raspberry Pi is able to trigger the haptic motor vibrations with low latency. This can be done by ensuring that the response time of the device is below 200 ms. It is important to confirm that response time is not affected by changing code or adding additional features.

4. **Battery Testing** - We will ensure that battery consumption does not increase greatly as the device changes and improves. This can be done by monitoring the energy usage of the battery pack.

4.6 ACCEPTANCE TESTING

To ensure a more comprehensive approach to our project deliverables, we will establish concrete, measurable objectives that encompass a variety of specific testing scenarios. This will include tests where the user is tasked with identifying distinct objects, reflecting the product's capability in object recognition. Furthermore, we will evaluate the user's ability to successfully navigate through a meticulously designed obstacle course, which will serve as a practical demonstration of the product's interactive navigational features.

The paramount indicators of our success prior to releasing the final product will be its reliability and usability. To gauge reliability, we will conduct rigorous and repeated testing under varying conditions to ensure consistent performance and stability. Usability metrics will be derived from user testing sessions, where feedback on the product's interface and interaction design will be collected and analyzed. This feedback will inform iterations that aim to create an intuitive and user-friendly experience.

The structured objectives and precise metrics established for this project are intended to ensure that the quality standards are not merely met but surpassed. The emphasis on these elements is to produce a product that demonstrates consistent reliability across various operational environments and is marked by a significant focus on user-centric design principles.

4.7 SECURITY TESTING

Every component and software aspect of this project is contained within a close loop of connections. Therefore, unless physical access to the device is achieved (which is outside our scope of control), the product is secure from any malicious intent.

4.8 RESULTS

The first tests conducted were tests to determine the haptic feedback vibration motors' ability to reliably transmit information. For the first testing we used an oscilloscope to transmit different Voltages and tested the ability for the user to identify the different vibrational intensities. Our results were pretty unreliable. But we had issues with the testing as the project was in it's infancy and we did not have a proper prototype leading to inaccuracies in the results.

We were also able to create a sleeve where we connected the haptic feedback motors and conducted a series of tests where we measured the user's ability to differentiate the different vibration intensities along with the differences in the ability for a user's ability to identify the motors. The testing with this bench came out with promising results as the user was able to reliably identify the vibration intensity with a margin of error of +/-10%.

We have conducted a series of tests with the new adafruit hat where we connected the haptic feedback motors to the adafruit hat and tried to replicate the same tests conducted using the native GPIO ports of the raspberry pi. The tests conducted came out to be unreliable leading us to walk back and refine our connections, specifically the soldering of the board, as it is a main concern in terms of the reliability of the Adafruit hat.

We were able to create our first prototype using a kinect along with 2 haptic feedback motors. thus we were able to have a proof of concept where we got a functional testing bench. The kinect was able to identify objects and relay that information to the motors but we did experience some difficulties when it comes to reliably transmitting that information. Reliably identifying the motors that were running and the response time.