# Test Plan

To test the success of the product we split the testing 2 parts. The first part consists of testing the overall accuracy of the movement tracking module. The success of this part is inversely proportional to the numerical value obtained for error, and should be time independent(a given requirement as haircuts can last more than 30 minutes, and so the tracking precision should not decrease with time). The second part of the testing is to implement the full end-to-end product using a Robix robotic arm. In this test we aim to perform a ‘data collection’ hair cut using a real model, and replicate the hair cut using the robotic arm. The success will then be a qualitative measurement proportional to the similarities of the too haircuts. We then define the success of the product as a sum of both test results.

## Part 1: Tracking Module

To test the tracking module we seek a numerical value that is representative of the error between the measured movements, and the ‘true’ movements made over some time period. To find the true movements we set up a 2-camera system to calculate the position of the tracking module in 3D space, indicated by the 2 cameras at each frame. The pixel positions of the tracker are then obtained from the cameras via an open-source object tracking script. The pixel positions are then equated to physical coordinates via an experimentally determined function to account for the focal length of the individual cameras. Once the known coordinates are obtained we then calculate an error value by taking the absolute difference of the known coordinates, and measured coordinates. This final function then represents the success of the test. The function should be constant(a horizontal line with respect to time), where the mean represents the overall precision. To obtain a more realistic representation of the error it is expected to perform several independent series of movements, and combine the error functions.

As an example, in developing the proof of concept we execute the test procedure as described above for 1 dimension, and hence using 1 camera. The goal here is then to find an error function by comparing the measured x displacements vs the known x displacements over some time period. To adequately test the measurements we test a number of different movement test cases. To track the motion tracker we use a generic object tracker written in python, obtained from the MOSSE tracker accessible from the OpenCV library(See [here](https://youtu.be/DO40fLr32f8) for a full video demonstration of the MOSSE tracker, where the tracking is measured in cm). The pixel to displacement conversion can be obtained analytically knowing the focal length of the camera used, and should be the preferred method when working in 3 dimensions. Here, we exploit the simplicity of working with 1 dimension and approximate the pixel to displacement conversion function via several measured lines on the grid, and manually finding the pixel coordinates for those measured lines. These known pairs of pixel and displacement then allows us to linearly approximate to find the conversion function. The result is shown in the linked video above.

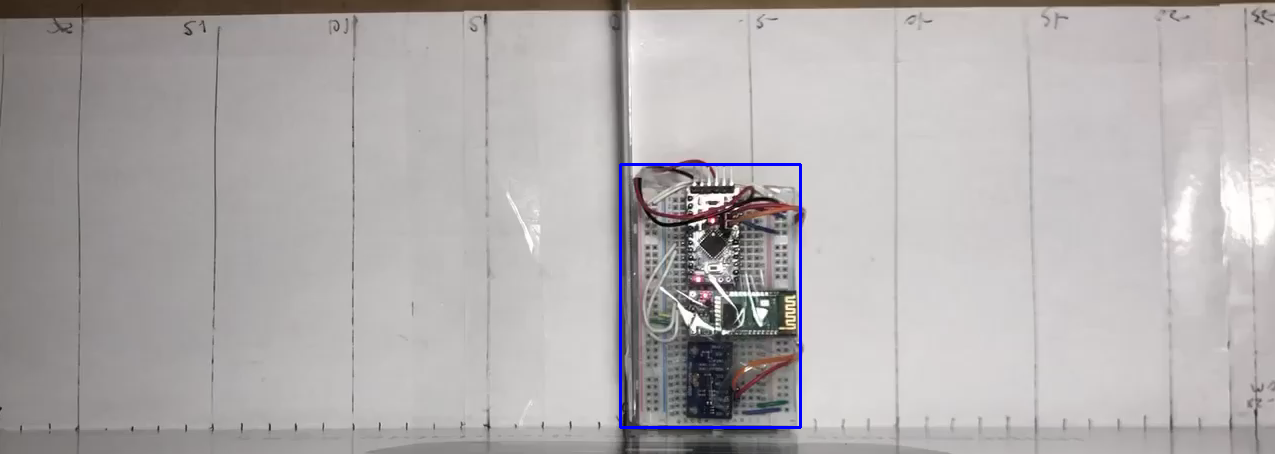


Figure 1: Object tracking screenshot

The movements to be made in each test case to be performed are up to the tester, however, there are some limitations to consider. First, the movements should operate within a few cubic meters at most to replicate a hair cutting booth. Second, the time without power on/off should be long enough to demonstrate time independence, i.e. approximately 10-30 minutes.

## Part 2: Robotic Arm Implementation

To test the full implementation of the product we use 2 replicated model heads with the same hair. An arbitrary haircut is given to one of the models, the data is collected, and the same haircut should be able to be replicated via the robotic arm and the remaining model head. The evaluation of the success is qualitative, but should resemble the haircut given if not exactly.