

Table 2: MCP Measurements

the centroid function, the coordinates for the center of the blob were recorded. Blob detection enabled the program to

It was observed that most participants used multiple fingers to swipe. All participants used a two finger pinching motion to zoom. The average measurements and gesture observations were used to write the program.

Setup and Design

In order to viably test the Kinect sensor, a table was designed which could mount the camera above the projection. The table was fashioned from 40/40 aluminum bars of dimensions 2ft width, 4ft length, and 3ft height. The depth sensor was mounted 3 ft above the tabletop. The actual tabletop was composed of two sheets of Plexiglas encasing a projection screen. From below the table, a projector was positioned to reflect off a mirror, which angled the projection onto the screen.

Figure 2: Tabletop Setup

Touch Detection

The first phase in testing the accuracy of the Kinect depth sensor was to assess its ability to track the motion of a person's finger across the table. To do this, multiple thresholds were established using C++ to detect objects within specified distance ranges.

To determine the threshold for if a person was touching the table, the initial distance of the depth sensor in relation to the table was recorded using a key press command. The code then checked if the depth sensor found an object within a range between the initial distance and the average measurement for the DIP joint. This range was used as the touch threshold.

The detected pixels within the range were recorded as blobs. The program can specify how many separate blobs to detect and the minimum and maximum pixel size. Using

Applications

The applications written to test the Kinect depth sensor were linked in the program using a menu screen with navigation options. This gave the program the illusion of a typical touch screen.

Coloring

The coloring application analyses the ability of the Kinect to continuously track the motion of a touch on the tabletop. Initial attempts to display the motion provided evidence that the tracing was not fluid. To adjust for the discontinuity between points, a line function was written to link the centroids of each respective finger using the slope between the current and previous points. A time elapsed function determined whether a segment should break. This was controlled by the total frames that passed between the recognition of

