Guide Mouse: Tactile Feedback for Virtual Environments

Angela Zhang

Background/Introduction

Those who are visually impaired commonly interface with computers through screen-readers such as VoiceOver for MacOS and JAWS for Microsoft Windows. Often, tactile interaction is lacking and graphical information cannot be conveyed. In computer graphics, one way to define a polyhedron is through a shell model that represents the surfaces of the polyhedron like an infinitesimally thin eggshell. In polygon meshes, the surfaces of the polyhedron are represented by polygons (facets), which may be triangles, quadrilaterals, or simple convex polygons. Users can use a tactile feedback device to explore the form of a polyhedron in a virtual environment through touch.

Purpose

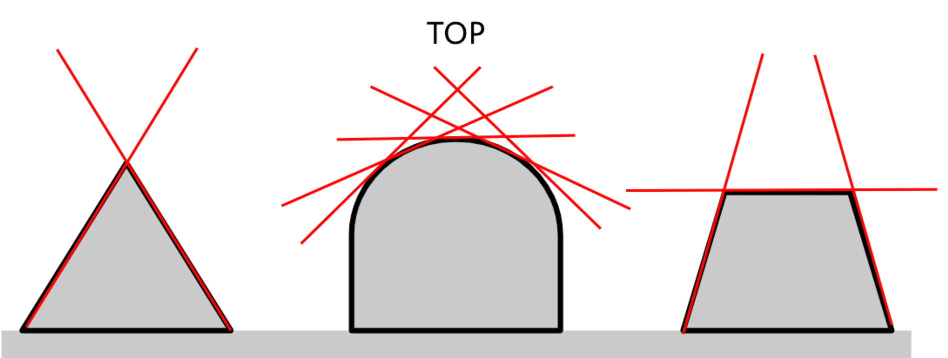
Guide Mouse was developed with the aim of allowing the exploration of 3D objects in a virtual environment. The program displays a 3D object on the screen. A user moves the tactile mouse to explore the shape. The program tracks the cursor position and identifies the visible facet the cursor is hovering over. The program then instructs the tactile mouse to position a feedback plane on the mouse to mimic the visible facet.

Project Summary

The development of Guide Mouse is broken down into two parts: the desktop software and the tactile mouse. The software inputs a 3D model and displays its 2D projection on the screen. When the cursor moves, the software identifies the topmost facet the cursor is hovering on and sends a command to the Arduino board via a serial port. The Arduino program executes the command to drive three stepper motors and sends back the motor positions to the program. The 3D-printed rack and pinion acts as a linear actuator and translates the rotary motion of the motors to linear motion, positioning the feedback plane to mimic the visible facet.

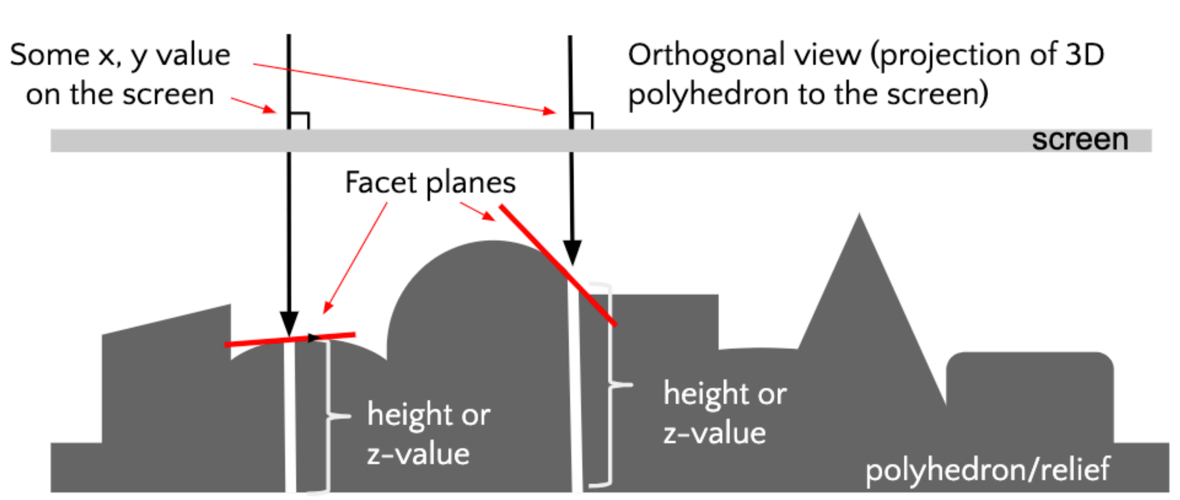
The Applicate Interpolator for a Facet of a 3D Polygon Mesh Given the Abscissa and Ordinate Program (Applicolator)

A program was developed in Processing (graphical library and IDE) that would allow for the user to import a 3D shape and input their cursor position relative to the 3D shape.



SIDE-VIEW of polyhedrons

Applicolator calculates three values corresponding to the heights of three shafts that would produce a plane with the same roll, pitch and yaw as the facet of the virtual object. Then, Applicolator sends the calculated values through the Arduino serial port to actuate the three shafts to move to their specified heights.



Side-view of screen and polyhedron/relief

Applicolator Procedure

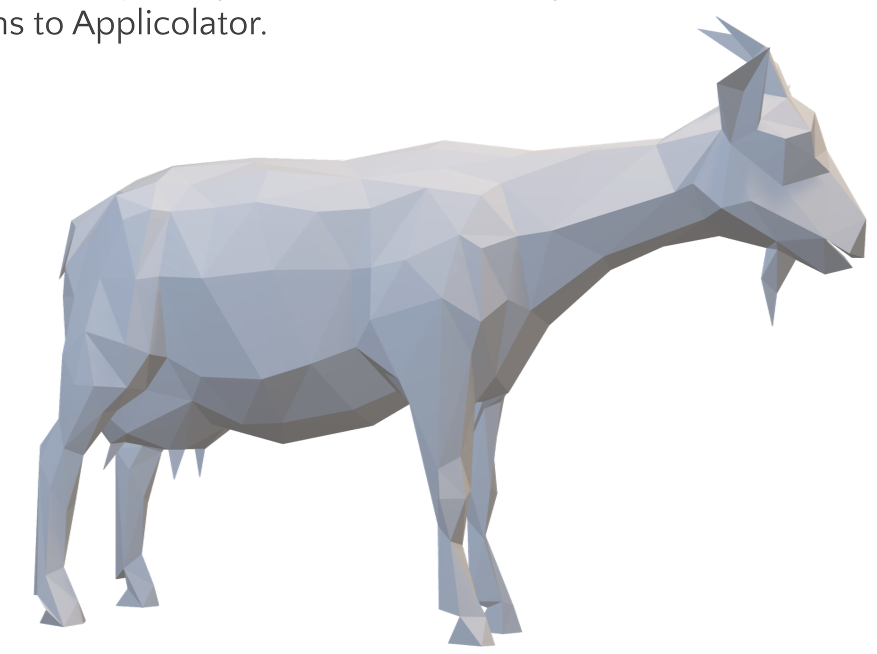
1. In order to efficiently locate the visible facet, Applicolator first checks if the x,y-value from the cursor is in the convex hull of the polyhedron's projection on the x,y-plane. Applicolator builds the convex hull of all vertices of the 3D shape using the Graham scan algorithm.
2. Applicolator builds an interval tree of the smallest enclosing rectangles of the projections on the x,y-plane of all facets of the 3D shape. This guarantees that the k facets whose enclosing rectangle the x,y-value is in, can be located in O(k log n) time.
3. For each enclosing rectangle, Applicolator uses the ray-casting algorithm to test whether the x,y-value is in the facet's projection on the x,y-plane.
4. Applicolator finds the largest z-value of each of the facets that the x,y-value belongs to. The z-value for the x,y-value is calculated by using the normal vector and a point on the facet. The normal vector is calculated by taking the cross product of the two vectors obtained from the three points on the facet. The facet that is visible from "a top view" is identified by finding the greatest z-value.
5. The feedback plane is positioned using three shafts. Using the positions of the shafts relative to the mouse and the height and tilt of the plane, Applicolator calculates the heights of the three shafts to position the plane. The max slope of the plane is limited to 45 degrees.
6. Applicolator calculates the three motors' target positions using the heights of the three shafts, the rack and pinion parameters, and the motor steps per revolution. Finally, applicolator sends a command with the target motor positions via a serial port to the Arduino board on the tactile mouse to position the feedback plane.

Communication between Applicolator and Arduino

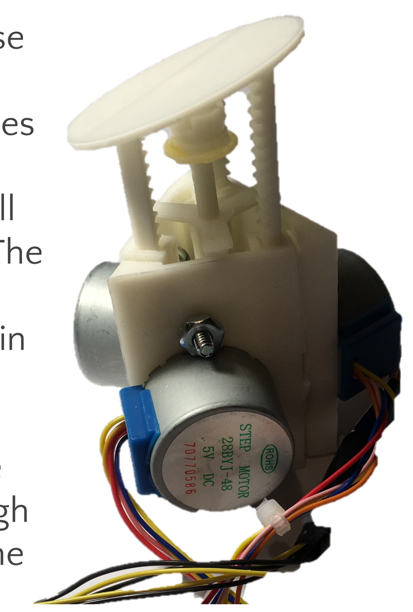
Applicolator communicates with Arduino via an asynchronous serial port. Applicolator sends a command to Arduino and waits for the response before sending the next command. The Arduino program listens for a command in an infinite loop and returns a response after executing each command. Commands and responses consist of printable characters and end with the LF character. The command is 13 bytes and includes the target positions of the three motors. The success response is 13 bytes and includes the current positions of the three motors. The error response returns the cause of the error with different lengths. At a baud rate of 9600, it takes more than 10 milliseconds to send a command or response.

Arduino Program

The Arduino board connects to three ULN2003 driver boards. Each driver board controls a 28BYJ-48 step motor, which has 4096 steps per rotation. The Arduino program drives the motors using the AccelStepper library. On receiving a command, it calculates the steps to move using the three motors' current and target positions, determines the max speed and acceleration according to the largest steps, and configures the three drivers such that the three motors can reach their target position at approximately the same time. Then in the infinite loop, the program calls the run function to move each motor step by step until all three motors reach their target positions. Finally, the program returns a response with the three motors' positions to Applicolator.

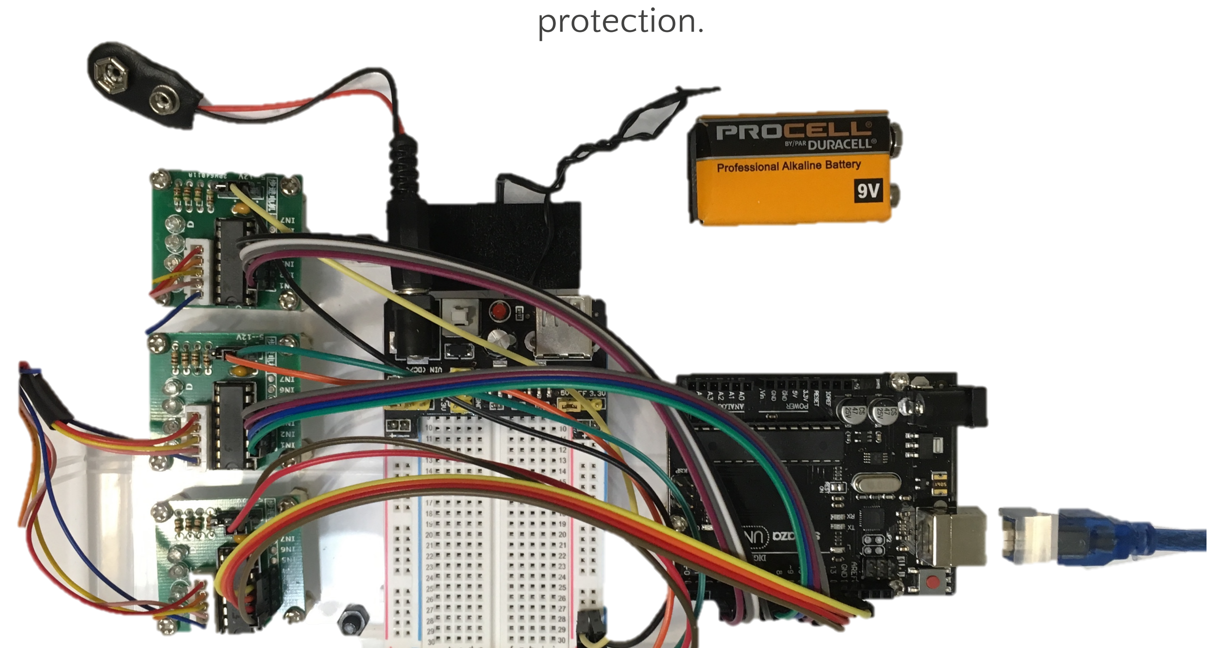


Guide Mouse



An actual mouse will be held in place by sponges so that cursor position can still be registered. The wires will feed through a hole in the side of the box. The feedback plane will stick through an incision in the top of the box.

The motors will be attached to motor drivers that are attached to pins on the Arduino Uno. The motors will receive power through a USB cable. The breadboard serves to distribute 5V to each motor driver. The board is secured to an acrylic sheet inside a plastic case for protection.



Construction of Guide Mouse

Some prototypes were created before the design took place for the rotary to linear motion conversion mechanism. Prototyping was done with foam, construction paper, and toothpicks. Once a general idea was conceived, measurements were flushed out using Google Sheets, and product design transitioned to CAD.

Design Diagrams (AutoCAD)

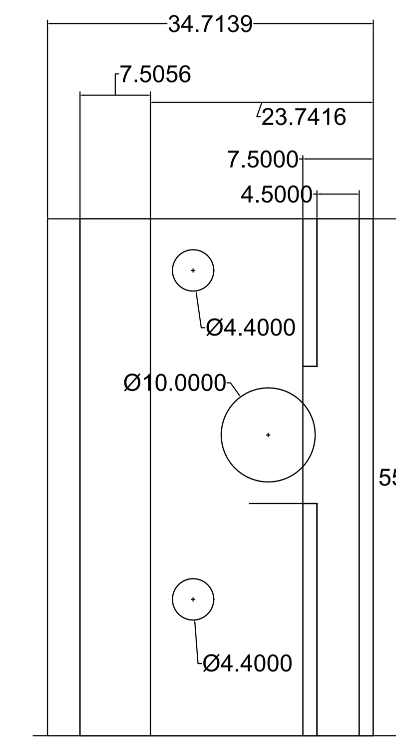


Fig 1. Front view of the case that the motor is attached to

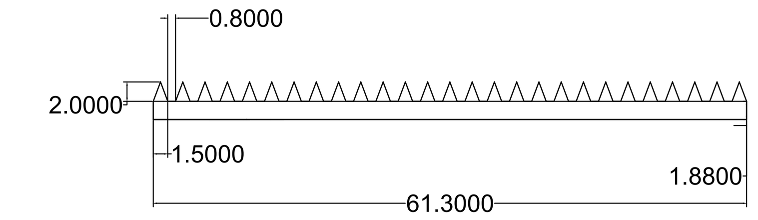


Fig 2. Side-view of the rack (the rack is one of three shafts that support the top plane)

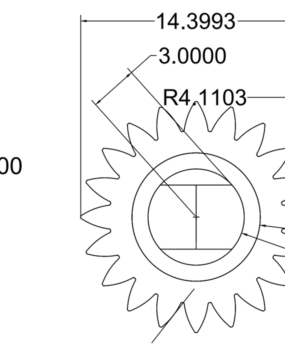


Fig 3. Top-view of the pinion which transforms the motor's rotary motion into the rack's linear motion

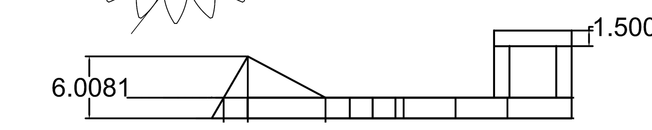


Fig 4. Side-view of the case

3D Models (AutoCAD and Blender)

The construction of the mouse required several custom-designed parts consisting of a rack and pinion, a case to that the motor is secured to, a ball-joint mechanism, a cylindrical peg, and a case for the cylindrical peg. All parts except for the ball-joint mechanism were designed in Autocad. The ball-joint mechanism was remixed in Blender from a Thing (Customizable Ball-and-Socket Mount) on Thingiverse designed by Mason Stone.

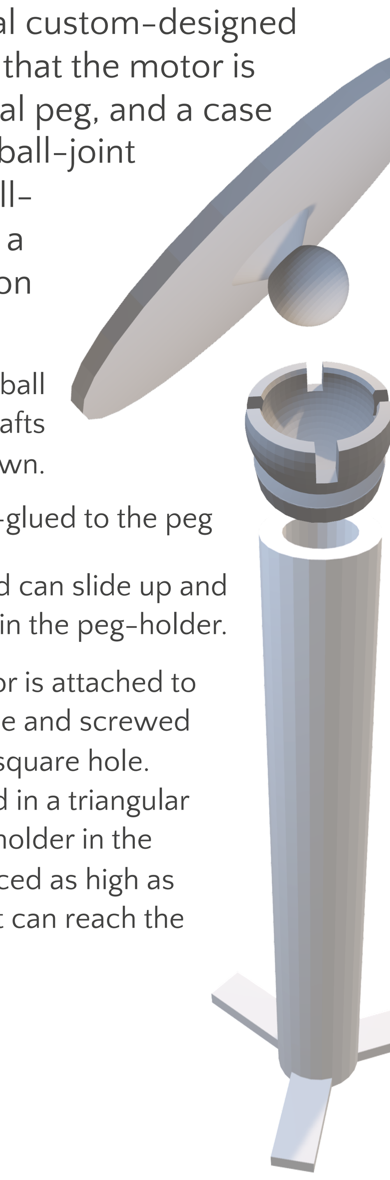


Fig 1. The top plane with the ball bearing attached. The ball bearing rotates around freely in the joint as the shafts push the sides up and down.

Fig 2. The ball joint, remixed in Blender. Is wood-glued to the peg

Fig 3. The peg-holder. A peg with the ball joint attached can slide up and down in the peg-holder.

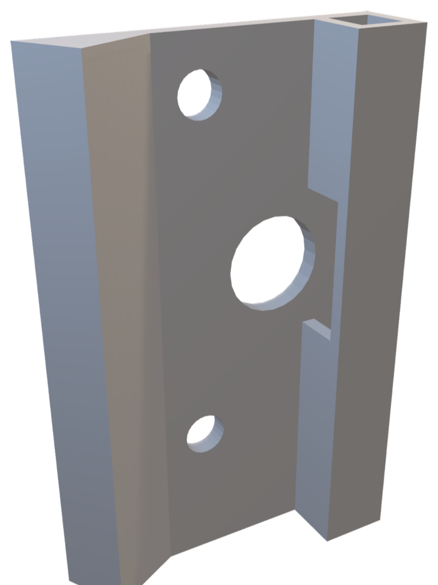


Fig 4. The case. The motor is attached to the case from the flat side and screwed in. The shaft goes in the square hole. Three cases are arranged in a triangular formation with the peg-holder in the middle. The motor is placed as high as possible so that the shaft can reach the maximum height.

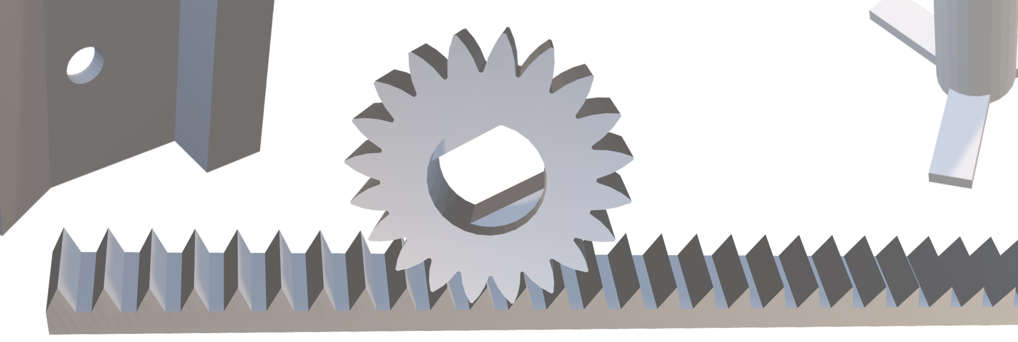


Fig 5. The rack and pinion. The pinion goes over the motor shaft and drives the rack through a cut in the rectangular box.

Applications and Future Work

Guide Mouse can be used to aid the visually impaired in sensing virtual environments. Guide Mouse will become especially pertinent as media is increasingly digitalized. The immediate next step for Guide Mouse is to test its effectiveness on human subjects. Surveys will be conducted where subjects use Guide Mouse to "sense" shapes and their accuracy with selecting the correct answer serves as a metric for Guide Mouse's performance. The final goal for the project is to make the code and design files open source so that anyone can download and modify the program and STL files, print a copy, and construct their own tactile mouse to use.

Acknowledgements

I would like to thank Mr. Robert Webb for allowing me to borrow a soldering station and other tools to complete my mouse. I would like to thank Ms. Effie Zimis for showing me how to use Cubicon Creator and Cubicon 3D Printer.