Project Proposal: Laser Cutter

Larisa Thorne
2015-11-22

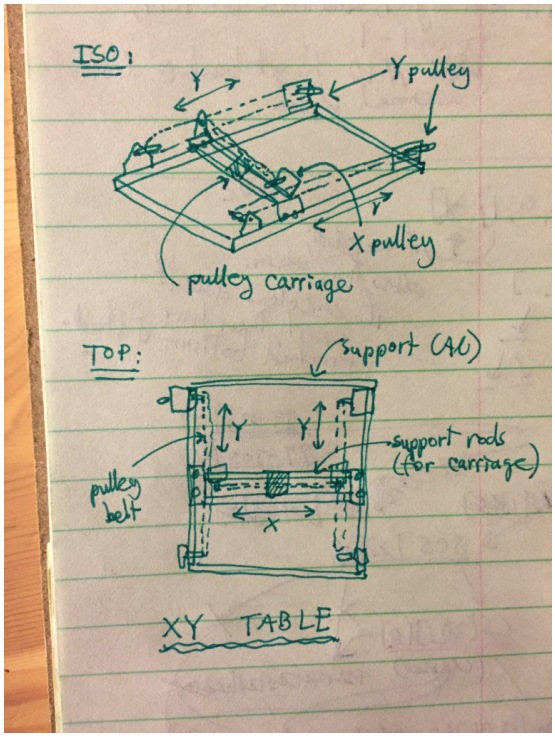
As my Term Project for the course 15-112 at CMU, I chose to make a laser cutter. This project will be divided into two parts: hardware and software.

Hardware:

On a most basic level, I will need to construct what is known as an 'XY table', and affix a laser to the carriage.

The XY table consists of an aluminum support frame, with (2) stationary Y stepper motors, which will support a moving platform for the (1) X stepper motor. The Y stepper motors are attached to the aluminum frame, and the X stepper motor to the Y timing belt pulley, with specially 3D-printed parts. They will move/rotate a timing belt pulley.

The platform which the two Y stepper motors support will be a rigid structure supported by two rods. These rods will support the carriage, which will be attached to a point on the X timing belt pulley, which will contain the laser.



The stepper motors will be controlled via Arduino, which will be connected to the host computer via USB to receive serial movement instructions.

A list of components will be included in the storyboard.

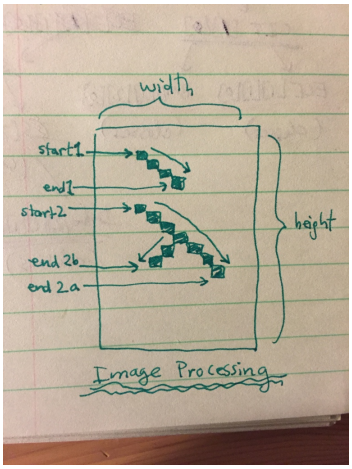
Software:

The role of the software is to take an input bi-color image (black and white), and to do some operation on it that will translate to motion of the carriage containing the laser head.

Modules to be used:

- pySerial: to communicate with the Arduino (via serial, over USB).
- PIL/Pillow: to get pixel information from the inputted image.
- Tkinter: to display and test code.
- (from Hardware: Arduino)

The method for translating bi-color image to Arduino movement instructions involves the use of backtracking. First, create a 2D list containing pixel location and color from original input image. This then becomes a 'backtracking' problem, different in flavor than previously encountered: we want to go through each pixel of the image and ask if it's the beginning of a line. If so, look at the adjacent eight pixels for more black pixels, with indicate a continuation of the line in that direction. This involves recursion, where the base case would be arriving at a black pixel that is either (1) surrounded by non-black pixels or (2) any black pixels around it have already been visited. The recursive case allows us to 'climb' back up the line to investigate branches, where our original line split into multiple lines.



The total path traced will be stored in a list, which will be handed to the Arduino when it's finished, so that it's not done in real-time.

I will write the software code in steps (version denoted by the # in file of form "LaserCutter#.py). A description of the goal of the version will be provided at the top. Simple test images will be used, which will be located in the same directory. I will create separate files for Arduino control (version denoted by the # in "ArduinoPySerialTest#.ino")