An-Najah National University

Faculty of Engineering and Information Technology

Computer Engineering Department

Graduation Project

Image Visualizing Pen Plotter

Students:

Ibramin Masri
Rami Theeb

Advisers:

Rae'd Alqadi, Ph.D.

Presented in partial fulfillment of the requirements for Bachelor degree in Computer Engineering.

Dec 20, 2021

Abstract

Pen Plotters (also called XY plotters) are a 2D CNC graphic plotting machines. This paper looks at a pen plotter machines that can take a picture in any extension and convert it into a drawable shape using image processing techniques, it then plots it on a canvas. Processing is done through a full software pipeline using modern technologies that can work on mobile and web, the pipeline is served using a Raspberry pi.

The drawing part consists of two stepper motors and one servo motor connected to an Arduino Uno. The motors control the movement of a gondola placed on top of stainless steel rods to achieve a movement along the X and Y axes, where at on end of the Y axis, a drawing device is mounted.

Acknowledgment

We would like to thank our families for their support and encouragement to always aim high and go beyond expectations. We then would like to show gratitude to Dr.Rea'd Alqadi for his supervision throughout the project.

And finally a thank you to all the professors and lecturers in the Computer Engineering department who we owe for the knowledge we posses today.

Disclaimer

This report was written by students at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the students. An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Table of Contents

Lists of Figures and Listings

Chapter 1 Introduction

Computer Numerical Controls (CNC) are motorized machines that are computer-controlled, they accept instructions that dictate their movement along the different axes. This movement can then be associated with some defined functionality by mounting different devices (such as a Laser Engraver, a 3D printing nozzle or a drawing device) on the moving parts.

A Pen Plotter (also called an XY Plotter) is a "specimen" of CNC machines. It plots against the X and Y axes carrying a pen that moves up and down in discrete steps, drawing different shapes on a canvas.

G-Code (which stands for Geometric Code) is one of the most common CNC instruction formats out there. It consists of different operations that represent movements along the axes, each instruction starts with the word G followed by the a number that distinguishes each instruction [1], parameters are then passed to the instruction to create the desired movement.

Generating shapes to use against a CNC machine is a process that can be simplified by a Computer-Aided Design (CAD) program, which are used to "create, manipulate, and optimize objects, pictures, or other designs" [2]. These program output usually need to go through another step to convert them to proper G-Code that the CNC machine would be able to understand, a popular example of such processors are Slicer programs, which are used in 3D printing to convert 3D models (usually in STL formats) into printing instructions that create and represent the shape with its printing specification (amount of filling, speed of the machine and precision).

This paper discusses the Image Visualizing Pen Plotter system, which plots an image captured through a camera on a canvassing platform. The system has a UI that the takes in an image, which then goes into the Processing Pipeline that exposes the image to different layers to produce the proper G-Code representation. The representation then gets passed to the Plotter, which is a GRBL-controlled pen plotter that takes in G-Code instructions and moves its parts to draw the passed image.

Chapter 2 discusses the constraints of the IVPP design and earlier courses that are relevant to this project. Chapter 3 looks at similar literature work done in the same field. The implementation and architecture of IVVP are covered in chapter 4. Chapter 5 discusses

the mechanical aspects of the system. Chapter 6 looks at the results of tests run on the system and analyzes them. Chapter 7 concludes the paper and talks about possible future work.

Chapter 2 Constraints and Earlier Course Work

2.1 Constraints

- **Motors Noise:** most of the commercial motors/drivers that exists in the market produce a noisy sound. To solve this problem, a high-grade stepper drivers are needed, but unfortunately those type of drivers were hard to find, so an alternative with acceptable price was chosen instead.

2.2 Earlier Course Work

- **Microcontroller Course:** This course provided the basics of dealing with microcontrollers, and how to interface them with different hardware components like motors.
- **Digital Electronic Circuits Course:** This course helped with understanding electronic circuits and dealing with different modules.
- **Digital Image Processing Course:** This course provided an understanding of basic image processing techniques which are used in the machine pipeline to turn the image into drawable shape.

Chapter 3 Literature Review

CNC machines and Pen Plotters in particular, have been used in many categories. Moreover, Pen Plotters use can be traced back to a century ago, where it was used in analog measuring devices like early Seismographs[3], which recorded the movement of the earth [4]

3.1 Development and Implementation of a Control System for a Retrofitted CNC Machine by Using Arduino

[5] Looked at the prospects of creating a subtraction CNC machine by repurposing an existing 3D printer and controlling it through a copy of the Marlin Firmware installed on an Arduino Mega.

The context of the research was to study the ability of creating a CNC machine with simple components, eleminating parts from the old CNC machine that were not necassary for the operations of the new one.

3.2 Continuous-Ink, Multiplexed Pen-Plotter Approach for Low-Cost, High-Throughput Fabrication of Paper-Based Microfluidics

[6] Studies the use of Pen Plotters in the rapid and low-cost prototyping of paper-based microfluidics. They used a Pen Plotter that has a multi-tray pen holder to draw hydrophobic outlines of a hydrophilic zones, that had the sample liquid flow through it.

Another aspect of the [6] design was the use of continuous ink supply to the pens to increase the throughput.

[6] System was able to draw 500 circular region in less than 3 minutes, with an estimated cost of 1 USD, which would allow for creating solutions for different fields, including the medical, environmental and food safety fields by using a Pen Plotter.

3.3 AxiDraw V3

AirDraw V3 is a commercial Pen Plotter, it offers the basic functionality of a Pen Plotter, with the ability to mount different drawing devices on

it. It has its own software to operate it, which can take SVG formatted files and draw them.

The Image Visualizing Pen Plotter (IVPP) offers a very similar functionality to the AirDraw V3, however, instead of focusing on printing SVG files, the IVPP offers the ability to draw actual images that can be captured by a camera, with non-vector extentions like PNG and JPG, with an acceptable accuracy.

Chapter 4 System Architecture

This chapter discusses the development details of this application including architecture, tech stack, and implementation.

4.1 General Design

In Image Visualising Pen Plotter is a full system that can take a captured image and draw it on a paper without the need for the any extra modifications by the user to make some sort of manual conversions or changes to the input.

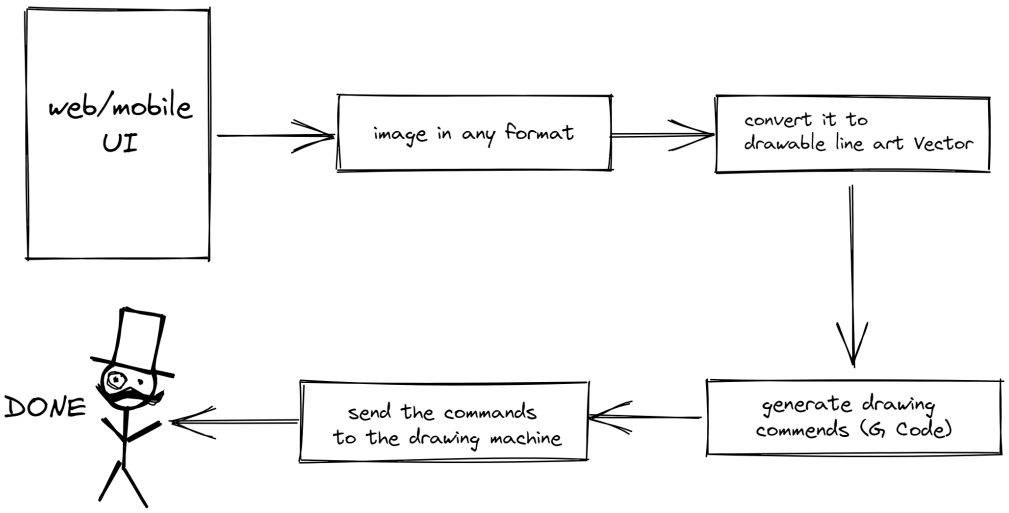

Figure 4-1: system flow

The flow showed in the previous figure simplifies the process of printing printing/drawing any image, even for people who do not have any technical knowledge in dealing with CNC machines. The user simply uploads their image with whatever format they have, this image then gets processed in a pipeline that includes different steps to produce a proper G-Code Representation of the imported image. This first part of the system is called the Processing Pipeline, and is done separately on Raspberry Pi-based component.

The G-Code is then sent from the Pipeline to the actual plotting component, which is referred to as just the Plotter.

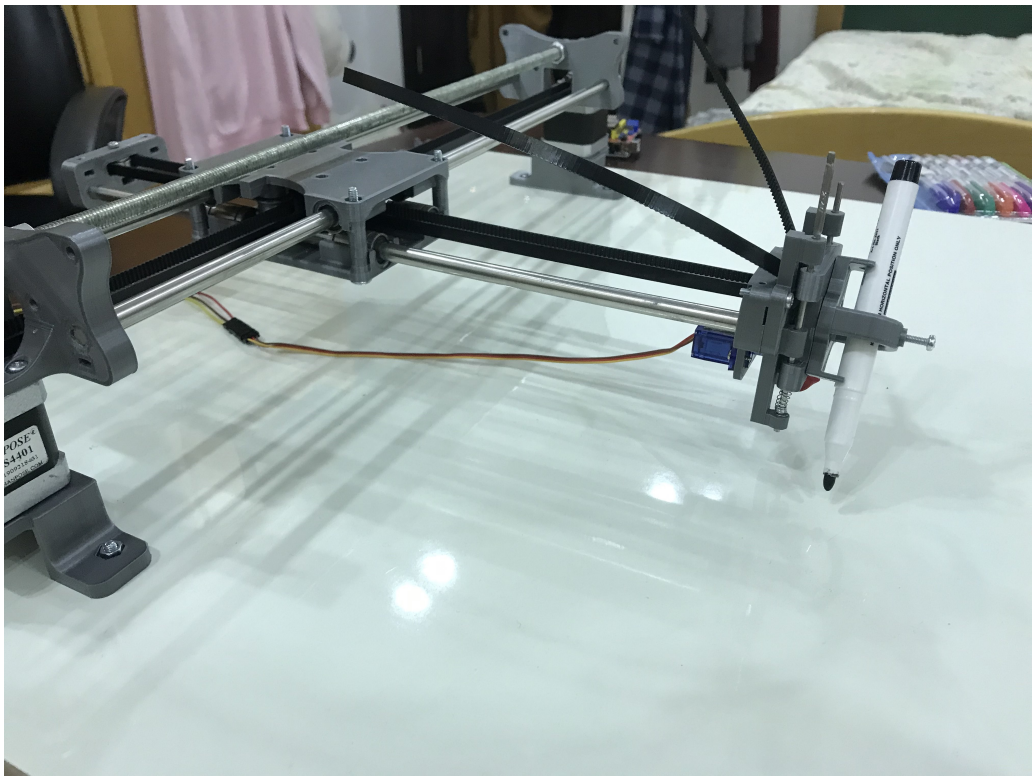

Figure 4-2: Drawing Machine

4.2 Processing Pipeline

in this section we will explain the Processing pipeline components in details

4.2.1 Web / Mobile UI

A react/Nodejs website was created to make it easy for the users to access our machine smoothly. the react side is responsible for the user interface and giving the user the power of coping their pics. and the node JS side is responsible of executing the pipeline scripts to do image transformations and g code generations and manage the communication with the arduino. the website is PWA (Progressive web app) that works on web and also can be installed as a mobile application. the user can choose the image from his file system. then the user has to crop the picture to match the aspect ratio of out machine.

then we will show the result to be drawn before the user chooses to proceed.

Figure 4-3: cropping the image

4.2.2 Convert The Image Into Line Art

in this step we managed to convert any image input into vectorized line drawing to be plotted

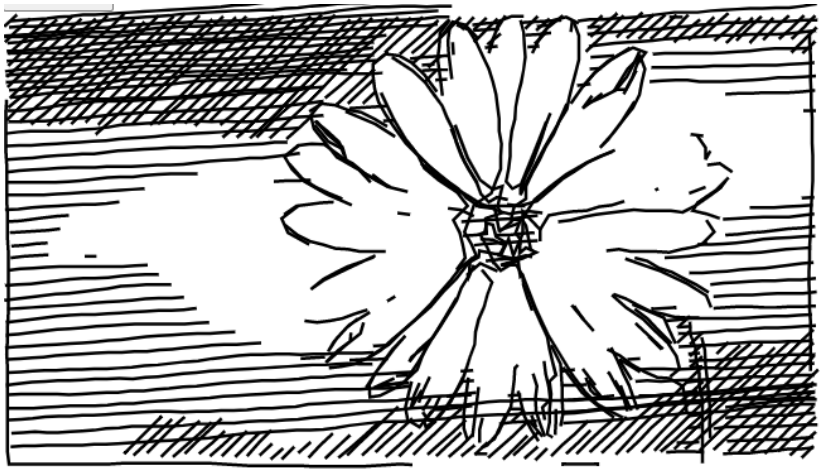

Figure 4-4: conversion result

4.2.3 Convert SVG Into Gcode

for this step the canvas feature of the web platform was utilized to help converting the SVG image into a web canvas that contains a certain paths which could be scaled easily. And to convert it into g code a well known library from Alibaba called GCanvas was used. it provided a preferment 2D-WebGL rendering for JavaScript run-times such as

nodejs. using this technique also gave us the ability to specify some important parameters easily. such as speed, feed rate, and etc... .

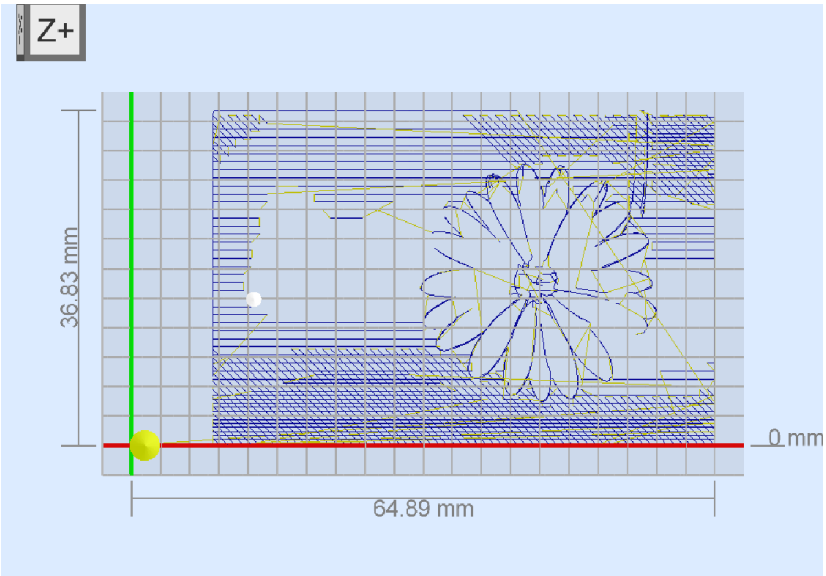

Figure 4-5: g code visualization

4.2.4 Sending The Code To The Machine

The code is transmitted from the pipeline to to the plotter through TTL Asynchronous serial communication between the Raspberry Pi and the Arduino.

Before initiating the G-Code instruction transmission, the Processing Pipeline checks which of its COMs connect to the Plotter. It then starts sending the instructions as characters.

4.2.5 Raspberry PI

The pre-processing pipeline takes place in a Raspberry pi model 4. to allow the user to connect to Rassbery PI , it's been converted into an access point with a DHCP and DNS services.

Figure 4-6: plotter wifi

when the user connect to this network then he could access our website/mobile app to do the printing process, this layer also helped

with preventing the collisions between more one printing process by allowing single connection to the access point.

4.3 Plotter

The plotter represents the physical drawing component of the IVPP, it communicates with the Processing Pipeline to get the drawing instruction, which are then executed through a GRBL-based Arduino-controlled system.

4.3.1 Electronic Components

4.3.1.1 Stepper Motors

The plotter consists of two stepper motors, which have a timing belt attached to them, allowing them to move the device along the X and Y axes.

Core XY configuration was chosen to operate the stepper motors instead of the usual Cartesian configuration. Core XY combines the rotations of the two motors to produce the linear movement, for instance, to move along the X axis, both motors would have to be rotating in different directions, Opposed to the Cartesian operation mode, which has each stepper motor control a single axis, Core XY makes it so that the movement of the X and Y axes are comprised of the rotation of both motors.

This allows for both motors to be stationary, which reduces the inertia on the motors, and increases the speed of the plotter[7].

Figure 4-7 shows how the Core XY configuration moves.

4.3.1.2 Servo Motor

While the two stepper motors are responsible for the X and Y axes movement, the servo moves the Z axis of the plotter. More specifically, the servo moves the drawing device up and down, distinguishing actual drawing movement from positioning movements.

The servo has two positions, either push the pen to meet the drawing canvas (90° angle), or retract and pull the pen upwards (0° angle).

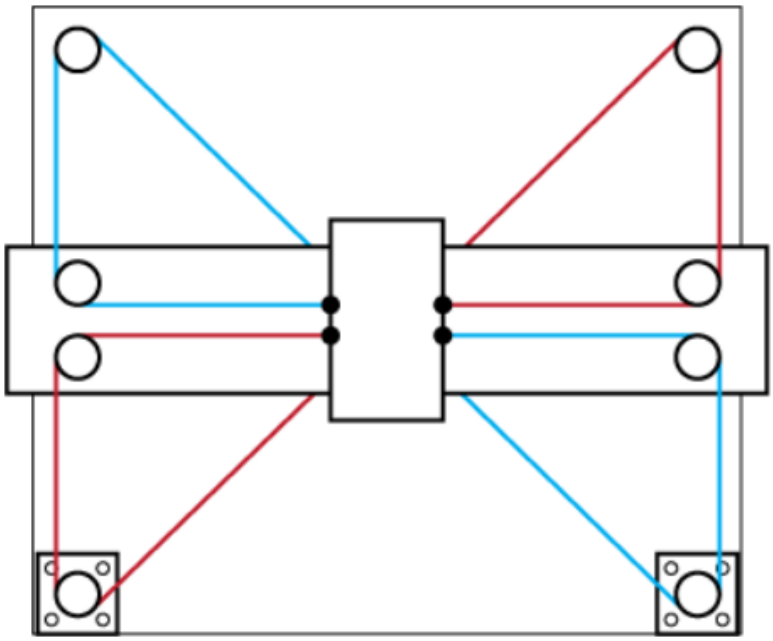

Figure 4-7: Core XY Movement

4.4 GRBL

"Grbl is a free, open source, high performance software for controlling the motion of machines that move, that make things, or that make things move"[8]

GRBL firmware offers great extensibility and configuration, it supports movement on the three axes (X,Y and Z).

The plotter used GRBL firmware to convert G-Code instructions into stepper motor steps, however, for a pen plotter, only movement on the X and Y are required, for the Z axis, values should be converted to either a push or pull servo movements, which requires an extra output pin that translates different G-Code Z axis instruction into the proper PWM values required by the servo to move the drawing device up and down.

Chapter 5 Mechanical Design

This chapter looks at the mechanical design of the Plotter and its mechanical components that create the overall movement of the Plotter.

5.1 Structure

The Plotter consists of different mechanical movement parts that are mounted on 3D printed components. These components are open-source parts for a generic plotter. They, however, are made to be used with a specific type of mechanical parts, some of which were not available, creating the need to modify the 3D models of the parts to better suit the what's available.

The 3D printed parts included holders for the X and Y rods and steppers and sliders for the drawing device.

5.2 Linear Movement

The movements of the X and Y axes were achieved by having a gondola surround two pairs of smooth stainless steel rods, one for the X axis and one for the Y axis. A timing belt went through this gondola and wrapped around a 16-tooth pulley on top of each stepper motor, the timing belt then wrapped around different pulleys inside the gondola, creating the Core XY configuration of movement.

Rotating the motors in the same directions, creates a movement in the X axis, while having them rotate in different directions creates the Y axis movement.

Figure 5-1: Linear Bearings

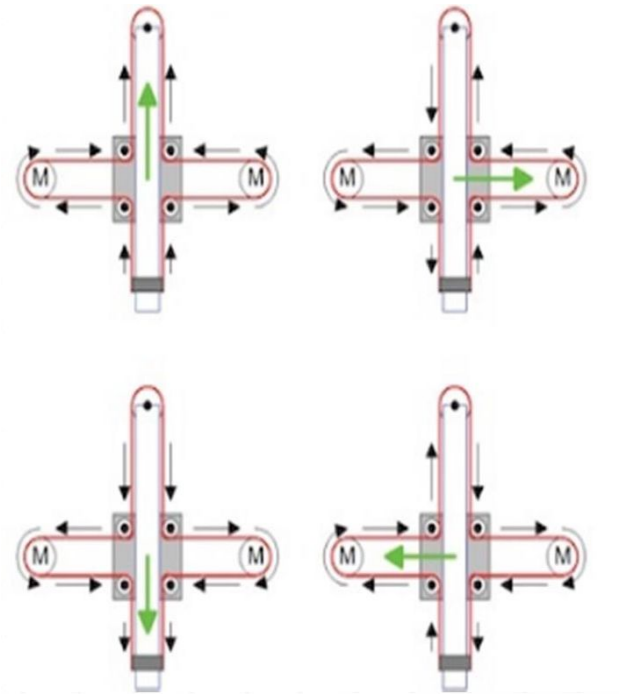

Figure 5-2: timing belt

5.3 Balancing

the stepper motor provides a precise movement control,by combining small step with a small angle angle of rotation. However, they exhibit small vibrations between these steps, which does translate into drawing noise in our case. in order to have the machine drawing precise graphics the device must be fixed and steady. and in order to do this the machine was installed on a wooden pallet using a 3d printed motor bracket as shown in the next figure .

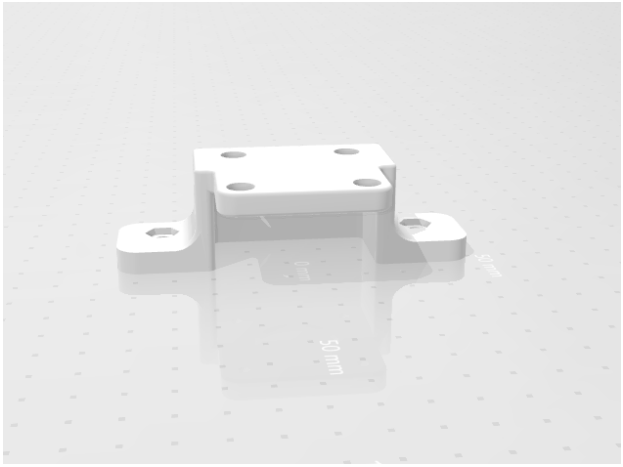

Figure 5-3: motor bracket 3d model

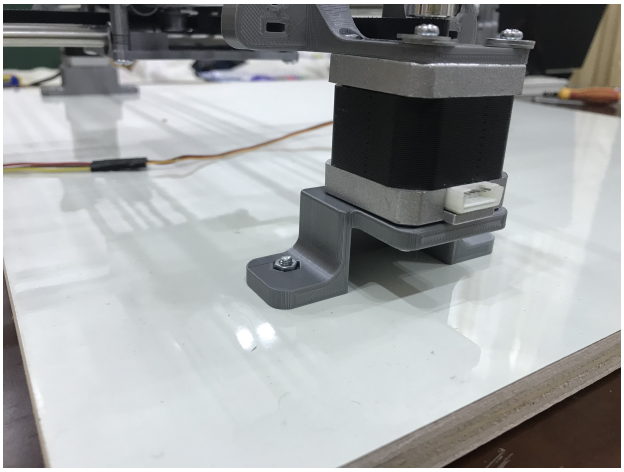

Figure 5-4: motor bracket

Chapter 6 Results and Analysis

6.1 Accuracy

The following tests were made to measure the accuracy of the system.

To test the Plotter's accuracy, commands were sent to the Plotter to draw some lines in different directions, and then, the opposite of those commands were issued, the expected result of such a test is to have a closed regular shape where the mounted pen would close that shape at the point it started from.

For the accuracy for the system as a whole, a PNG image with regular shapes was inserted into the system to draw using a markup pen and the drawn image was then compared to the original image. Figure 6-1 shows the original image, while 6-2 shows the line art representation of the image, while 6-3

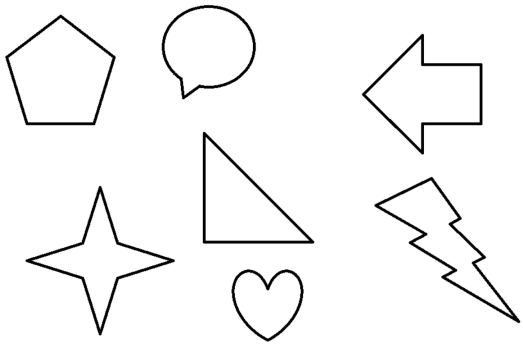

Figure 6-1: Original test PNG image

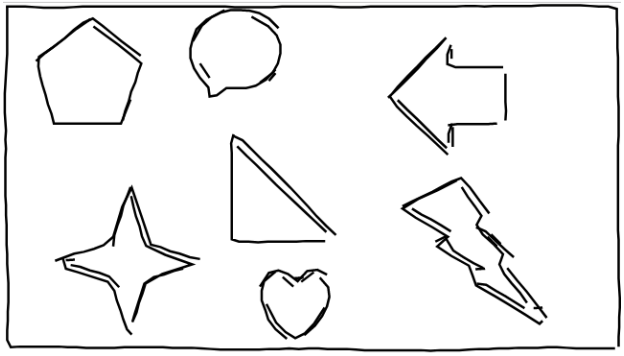

Figure 6-2: Line Art conversion of the test image

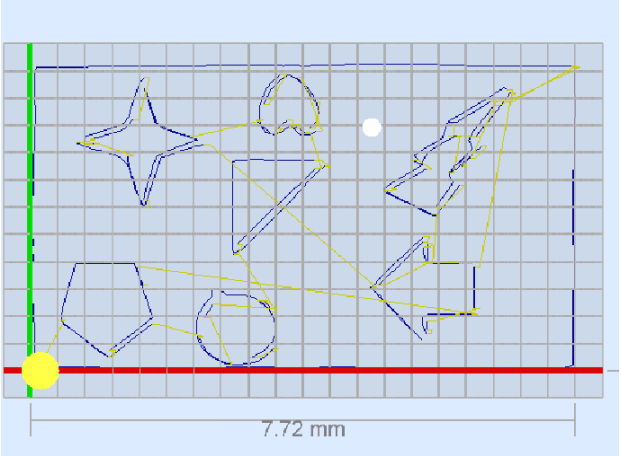

Figure 6-3: G-Code conversion (output) of the test image

6.2 Speed

The speed of the system can be set by changing the feed-rate value in the G-Code, to find a good balance point between fast and accurate, different feed-rate values were chosen to draw different shape and the differences between the shapes was then compared along with the time it took to draw them.

Chapter 7 Conclusion and Future Work

7.1 Conclusion

This project implements an easy-to-use, coast efficient and easy to control pen plotter that could automate the whole drawing process. starting form choosing the image using a web/mobile application, until drawing it on paper, with no need to understand any of CNC tech concepts or image transformations.

7.2 Future Work

- **Shape filling:** one of the near-future work will be supporting drawing filled shapes. currently the device supports only line drawing.
- **processed image customizability:** in the future the application will support the user with some image processing parameters that can change the output of the processing pipeline to meet the user's need.

References

[1] J. Green, D. Irkaev, B. Lee, Dejan, Alan, and Norman, *G-code explained: List of most important g-code commands*, Oct. 2021. [Online]. Available: https://howtomechatronics.com/tutorials/g-code-explained-list-of-most-important-g-code-commands/.

[2] K.-p. Yang, G. Alkadi, and T. Parker, "Converting SVG to g-code for 3d printers,"

[3] *Pen plotters: Best XY plotters of 2021*, Dec. 2021. [Online]. Available: https://all3dp.com/2/pen-plotters-best-xy-plotters/.

[4] U. none, *Seismometers, seismographs, seismograms - what's the difference? how do they work?* [Online]. Available: https://www.usgs.gov/faqs/seismometers-seismographs-seismograms-whats-difference-how-do-they-work.

[5] A. Quatrano, S. De, Z. Rivera, and D. Guida, "Development and implementation of a control system for a retrofitted CNC machine by using arduino," *FME Transactions*, vol. 45, no. 4, pp. 565–571, 2017.

[6] R. Amin, F. Ghaderinezhad, L. Li, E. Lepowsky, B. Yenilmez, S. Knowlton, and S. Tasoglu, "Continuous-ink, multiplexed pen-plotter approach for low-cost, high-throughput fabrication of paper-based microfluidics," *Analytical chemistry*, vol. 89, no. 12, pp. 6351–6357, 2017.

[7] *Cartesian motion platform*. [Online]. Available: http://corexy.com/index.html.

[8] Grbl, *Home · GRBL/grbl wiki*. [Online]. Available: https://github.com/grbl/grbl/wiki.