Development of a Low-Cost Automated Braille Coordinate Measuring Machine

Jamie Glen
School of Computing & Communications
Lancaster University
Lancaster, UK
jamie.a.glen@outlook.com

Abstract—This project presents the design and development of a low-cost semi-automated Coordinate Measuring Machine (CMM) for the precision measurement of Braille embossing. The system incorporates a cartesian gantry, Arduino-based motor control, and a strain gauge for sub-15-micron resolution. It significantly reduces manual labor and improves measurement repeatability, offering a practical in-house calibration tool for industry use.

I. Introduction

This project was developed over a 10 week period during an industry placement.

Hardy & Ellis Inventions (HE) are an SME who specialise in solutions for the packaging industry. They required a way to calibrate their Braille inspection systems in-house as they currently had to either rely on costly third-parties, or use their custom manual method.

The current manual method was a time consuming process as each dot required up to 21 repeat measurements for statistical significance. Despite the micrometres ±2.5 µm accuracy, practical use yielded ±10 µm due to misalignment, deformation, inconsistent force application, and human error. All of these could be reduced or eliminated with an automated solution.

The brief given was to develop a low-cost, semi-automated Coordinate Measuring Machine (CMM) capable of delivering repeatable, accurate measurements of Braille embossing, using only commodity hardware. This led to the development of a custom Cartesian gantry system, built around widely available components and controlled via an event driven Python program.

II. Prototype Construction

A. Hardware Construction

For the electronics an Arduino Mega 2560 micro-controller with a RAMPS 1.4 board and A4988 stepper drivers were chosen due to their lower price and mature platform with a strong user base. The Arduino will function as the controller for the system, with the RAMPS board interfacing between the rest of the electronics and the stepper drivers providing power to the motors.

The physical system, shown in Figure 1, is constructed with a moving XY bed and a separate Z probe on a fixed gantry. Each movement is controlled through stepper motors with mechanical limit switches on the x-min, y-min and z-max axis, and a Z probe set up for measurements. Despite the original brief calling for a coreXY belt setup, I chose to go with a simpler design where the X and Y axis are independent of each other, reducing the potential for backlash and allowing for more precise small movements.

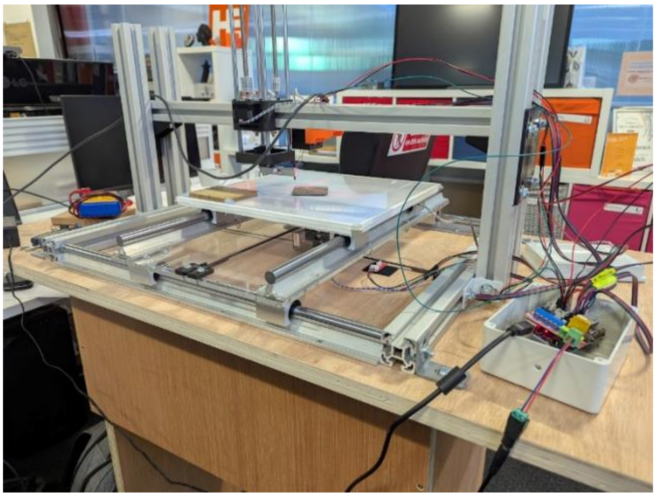

Figure 1 - The full machine

The XY bed is mounted on linear rails, with the Y movement mechanics mounted on top of the X mechanics, each with a stepper motor mounted underneath as described in Figure 2. Both stepper motors are attached via a toothed drive pulley to a toothed belt. One stepper moves the Y axis on its rails, which are then mounted on top of the X axis bed, and the other stepper then moves the whole XY bed along the X axis. Having individual motors operating each axis of movement offers a very simple but effective solution to XY movement with a theoretical step size of 0.0125mm.

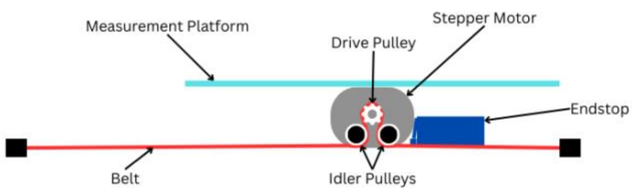

Figure 2 - Diagram of the XY Movement Mechanics

The Z axis can be mechanically thought of as its own system, with a separate structure not dependent on the X or Y movement. It moves through the use of two steppers where each stepper runs up a linear screw, with 4 guide rails running through linear bearings to ensure there is X/Y stability. Considering the pitch of the lead screw, the motors step size and the motor drivers allowing for 1/16th micro-stepping, the Z axis has a theoretical step size of 2.5 microns.

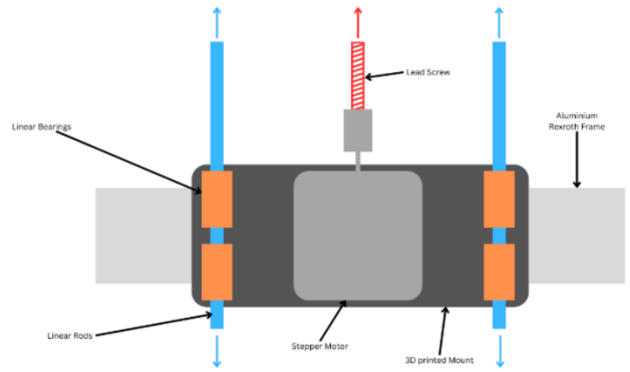

Figure 3 - Diagram of Z Movement Mechanics

For a Z probe the system makes use of a strain-gauge based load cell with an amplifier module, as pictured in Figure 4. This allows the load cell to function as a digital trigger once a force threshold is surpassed. This threshold can be configured and would allow for a consistent deformation of the measurement subject as the same pressure is applied for each measurement - this is not ideal but is easier to calibrate out then an inconsistent deformation.

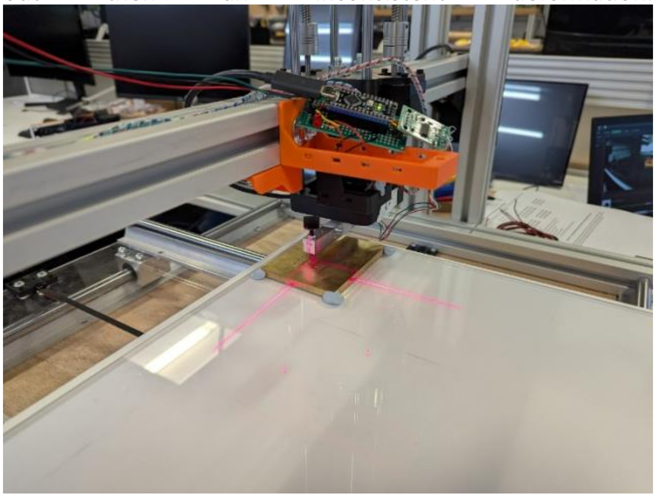

Figure 4 - The Z probe implementation

B. Software Development

The solution uses Marlin firmware with a custom configuration file setup designed for accuracy. The Marlin G code interpreter was used to enable the use of industry standard G code to directly control the machine.

The final solution utilizes an object-oriented, event-driven architecture based around a PySide6 GUI, allowing the user to manage every aspect of the machine for Braille measurements.

Upon start the machine homes itself and the user opens to a GUI with a blank 300 x 300 mm bed plot. The user can move the machine bed around using WASD movement and Shift to enter a 'precision mode' that allows for fine adjustments. Keyboard inputs are executed live by the machine and also displayed with a slight delay to avoid overloading the machine with polling position update (M119) requests.

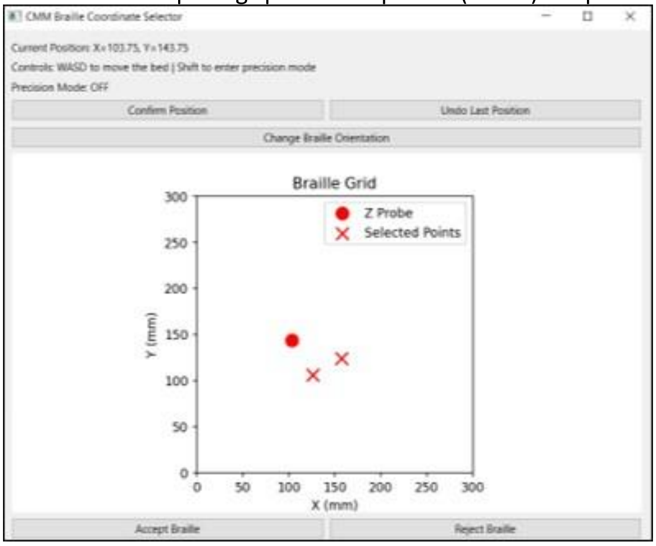

Figure 5 - The GUI after the user has selected 2/3 points

To begin measurement, the user selects three points, with each point being a corner cell in the braille they wish to measure, as shown in Figure 5. The system then automatically generates the braille grid according to ISO 17351 Marburg Medium [1] within these coordinates, and the UI zooms to fit the space (Figure 6). At any point, the user can undo the last point, reject the generated grid, or adjust the braille orientation if needed.

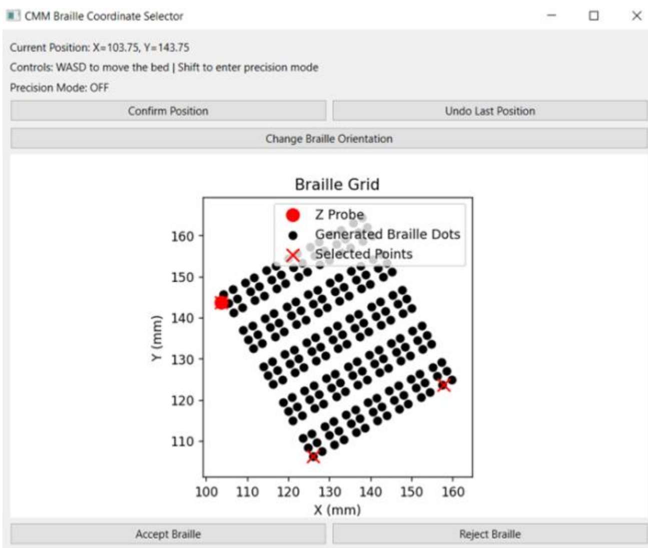

Figure 6 - The Braille grid generated from user points

From here the system converts the generated points into G code commands and then begins to execute measurements. All measurements are stored in a .csv file for later use by the users.

III. Evaluation of the System

The tolerance of the machine was calculated using a brass Braille block that was created to mimic the same NPL verified block HE already have. Whilst this is not ideal, it is as close to ground truth as could be realistically attained.

When calculating the standard error of the mean, the tolerance can be calculated from the variance and the line of average points can be calibrated by mapping it to the ideal line.

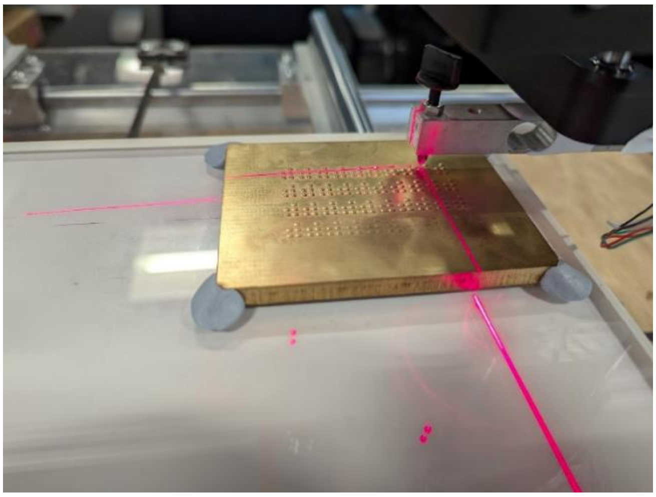

Figure 7 - The machine undergoing tolerance calculations

The braille block, shown in Figure 7, was 4 rows of 10 columns of braille cells, each cell containing 5 full dots and one empty dot. The dots in each cell gained height by 0.01mm per cell in a stepwise gradient along the block from 0 - 0.39mm. It is worth noting that in the machines measurements, specific dots are returned where the average deviates from the ideal line. This is due to faults in the Braille block, which is not perfectly calibrated, as opposed to the machine, and this has been confirmed by using the original micrometre setup to measure it.

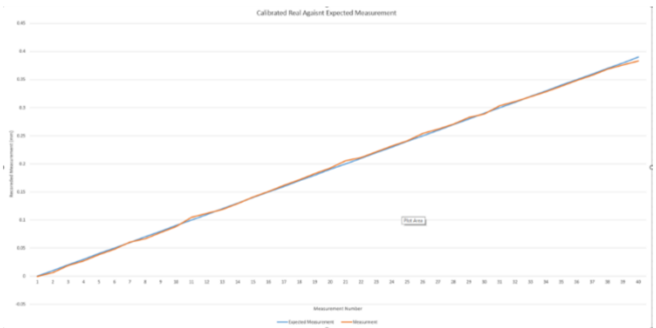

Figure 8 - Calibrated results against expected measurements

Figure 8 demonstrates the calibrated average measurement over 5 measurements. The similarity of the lines shows just how accurate the machine got after calibration. The tolerance of the machine was calculated at ±0.0134 mm at 95% confidence for a measurement of 0.2 mm; not quite the ±0.010 of the original manual system, however still an acceptable result considering the hardware used an its combined £450 price tag.

While the prototype succeeded in improving repeatability and reducing manual workload, it fell slightly short of the original precision target. This limitation is largely due to the performance ceiling of low cost components. Future improvements could include adding encoders to detect missed steps, increasing probe resolution, and refining the Z axis mechanics.

Despite these limitations the device represents a promising foundation for accessible, affordable quality control solutions. This system will also be advanced to include full point cloud generation and depth scanning in the near future as progress continues to be made on it.

IV. Responsible Innovation

This project reflects a commitment to responsible innovation by helping to address a critical accessibility challenge: verifying Braille quality on pharmaceutical packaging. Automating this process enhances safety and independence for visually impaired users while reducing human error. The design focus to encourage future design adaptations also extends the systems design beyond its initial use case.

Environmental sustainability was considered throughout development – the prototype utilised low cost, off the shelf parts that are easily repairable or replaceable, minimizing e-waste.

V. Author Bio(s) / Experiences

I am a fourth year integrated masters student at studying Computer Science at Lancaster University. Being completely honest, this was my first attempt, at any significant level, to bring a device to life. From being given an idea then turning that into a concept and finally a working prototype, this has been a huge learning curve for me.

Whilst it may have been my first attempt at a project like this, it was a huge success and will definitely not be my last – this has lit a fire inside me and I cannot wait to continue to improve this project into my dissertation.

VI. Acknowledgements

A massive thanks to everyone at Hardy & Ellis Inventions – without them this project would not have been possible.

VII. Refernces

[1] International Organisation for Standardization (ISO), "ISO 17351: Braille – marburg medium", https://www.iso.org/standard/66182.html, 2025, accessed 2025-05-12