**A close up of a logo

Description automatically generated**

**A Learning & Teaching Research Collaboration**

***Building and improving a robot drawing arm / pen plotter based on BrachioGraph[[1]](#footnote-1)***

Student Partners:

Ali Rashidi

Tibebe Demissie

Brownwell Ehimare

Amirreza Nozari Zarmehri

Staff Partner:

Muharrem Maloku

**School of Computer Science and Engineering**

**Academic Year 2023-2024**

Contents

[1. Introduction 3](#_Toc176254932)

[2. Background and Aims 3](#_Toc176254933)

[3. Methods 4](#_Toc176254934)

[4. Results 4](#_Toc176254935)

[5. Discussion 5](#_Toc176254936)

[6. Conclusion and Recommendations 5](#_Toc176254937)

[7. Dissemination 6](#_Toc176254938)

[8. Team Reflection 6](#_Toc176254939)

[9. References 6](#_Toc176254940)

# Introduction

This project was built upon the original design of the BrachioGraph, a low-cost, simple drawing robot created by Daniele Procida. Our objectives were twofold: first, to improve the hardware/software design wherever possible while giving full credit to the original creator; and second, to create and share detailed design files, including a Bill of Materials (BOM), in order to make it easier for others to replicate and further build upon our work.

The motivation behind this project was to provide a more robust and accessible version of the BrachioGraph, specifically tailored for students, educators, and hobbyists. We aimed to preserve the simplicity and affordability of the original design while addressing any potential limitations that could hinder its effectiveness or usability. Additionally, we explored ways to simplify the accompanying software, seeking to strike a balance between the speed of drawing and the quality of the output, particularly when working with vector images.

To achieve these goals, we developed a custom 3D holder designed to fully support the weight of the rotators and the pencil. We integrated essential electrical components, including a Raspberry Pi, a breadboard, and rotators, all connected with colour-coded wires to ensure ease of assembly and understanding. The resulting design is both lightweight and quick to construct, making it particularly well-suited for educational and demonstrative purposes.

The outcome of our project includes a refined hardware design that offers improved stability and ease of assembly, coupled with optimised software that facilitates quicker yet still recognisable line drawings. The device is capable of accurately replicating any drawing it is provided with. We have meticulously documented all improvements and made the design files and BOM publicly available on GitHub. This project serves as an invaluable educational tool, offering deep insights into fundamental concepts in computer science, such as machine learning, artificial intelligence, and computer vision, while simultaneously demonstrating the immense potential of simple, cost-effective robotics.

# Background and Aims

BrachioGraph is such a robotic drawing arm that is low cost and "is easy-to-build" and "about €15 in an hour or so, using a Raspberry Pi computer, hobby servo motors and household items" and this is accompanied by operating software that drives the motors and converts images to plottable vector images (Procida, 2019).

We are proposing to build and improve upon this work and design parts that are more robust for a sturdier robot and look to improve software components if required. We will build an interface that is easier to use. This will allow for others other than project team to use a this without much assistance. Once completed we hope that the resulting artefact will produce a 'kit' that can be used in educational settings for students to build their own with it.

Our project seeks to enhance and build upon the original BrachioGraph design by Daniele Procida. Our primary goals are: 1) to improve the design where possible, while giving full credit to the original creator, and 2) to develop comprehensive files that make it easy for others to access a kit for parts and a Bill of Materials (BOM), which we will share with the community once finalised. It's important to note that we are not claiming this as our own design, and we plan to share the final version through an appropriate online repository, such as GitHub.

While the main focus will be on hardware improvements, we will also explore opportunities to simplify the software, particularly in light of the arm's basic design. We aim to strike a balance between producing recognisable drawings from vector images and optimising speed. For example, can the software be adjusted to create simpler line drawings that are quicker to produce while still maintaining the integrity of the original image?

This project is designed for students, academics, professors, and learners as a demonstration of how a simple, low-cost idea and its mechanisms can be dissected to understand machines with similar principles. Additionally, this project provides valuable insights into core computer science topics such as machine learning, artificial intelligence, and computer vision. As mentioned, this project is intended to be accessible to everyone.

# Methods

In this project, we built upon the existing BrachioGraph, an easy-to-build pen plotter driven by simple Python applications. The original BrachioGraph can be constructed with minimal expense and time, using a Raspberry Pi computer, hobby servo motors, and household items. Following the guidelines provided in the BrachioGraph documentation, we designed new holders for the motors, pen, and more efficient arms using 3D printers. This allowed for more stability and precision in the drawings.

We did not involve any participants in our project as it was focused on hardware and software development rather than user interaction. Therefore, ethical considerations related to participant involvement were not applicable. Our primary activities included designing, 3D printing, assembling the mechanical components, and developing software. The software development involved enhancing the original Python code to incorporate a user-friendly interface that allows users to upload images. These images are then processed and vectorised, enabling the robotic arm to produce corresponding sketches.

Data analysis in this project was primarily qualitative, focusing on the functionality and performance of the robotic arm. We tested the arm's ability to draw various images and evaluated the accuracy and quality of the produced sketches. The evaluation was conducted through iterative testing and refining the hardware and software components until satisfactory results were achieved.

# Results

Our project resulted in improvements to the original BrachioGraph design. We developed a user-friendly interface that simplifies the process of uploading images, making it accessible to users without extensive technical knowledge. The enhanced robotic arm demonstrated the ability to paint a wide range of pictures with a good degree of accuracy and quality. The sketches produced by the robotic arm were evaluated based on their fidelity to the original images and their overall artistic quality. Our results indicate that the developed interface is easy to work with, and the robotic arm can reliably produce acceptable drawings. The improvements we made have successfully enhanced the functionality and usability of the original BrachioGraph, providing a fun tool for creating robotic sketches.

# Discussion

Our improvements in the implementation of the Brachiograph's addresses several limitations of previous model, offering a less technical setup and usability. The original model of the Brachiograph, although well developed, still requires some level of technical understanding to getting it setup and working. While trying to make a prototype, we face a lot of trouble at each step, having to flash the raspberry pi with an operating system, downloading their code but experiencing very cryptic errors when trying to run the application due to some of its dependencies being deprecated, and having to configure the GPIO pins on the Raspberry Pi while also figuring out which commands to use to control the device, The setup is a very involved process.

When students are trying to learn, letting them focus on one thing is the most ideal for the consumption of knowledge. Having them juggle figuring out the Linux system, the raspberry pi, the electronics and their software was not very conducive to learning. In order to address this, we planned to make a containerised tool that can handle all the code and system configuration. It would be a layer above where all the technicality is abstracted away by a web interface which allows users to focus solely on building the robotic arm and then with a few clicks have their raspberry setup. As compared to the previous process, the user experience with this new method would be to install the Raspberry Pi operating system, then download our code and run. This would host a website where you can interact with the robotic arm to do all the necessary functions like uploading an image, starting a drawing cycle and stopping it. We believe that this redesign of the process lowers the barrier to entry in the electronics field and allows students to get their hands dirty with a real project where they can learn lots. It doesn’t hold the hands of the students too much and if they wanted to explore more of the technical side, they could try to add more features to the website we created.

# Conclusion and Recommendations

The redesign of the Brachiograph project marks a step forward in making electronics more accessible and user-friendly. By abstracting away the technical challenges of the original model, our changes simplify the setup process, allowing students to focus on the creative and educational aspects of the project rather than being bogged down by technical details. This approach not only broadens the potential applications of the Brachiograph in educational settings but also paves the way for students and hobbyists to engage with robotics in a more intuitive and enjoyable manner.

Future Recommendations:

* Prototype Development - Proceed with the construction of a working prototype
* Feature Improvement - Implement other features of the Brachiograph code, like the calibration on to the interface.
* New Vectorisation Methods - There is room for research on a better vectorisation method to be implemented rather than using the one provided by the Brachiograph project.
* User Feedback - Engage with students to gather feedback and identify further areas for enhancement.

# Dissemination

As our project is not as much research as it is the creation, testing and utilisation of a mechanical drawing arm, we do not have specific dissemination methods. As mentioned in our original proposal, we plan to upload online the new steps to create this version of the Brachiograph, with both the code and designs used in the project becoming publicly available. We could even go with a more intuitive approach, such as a tutorial or showcase video.

For the physical result of our project, we would like to donate one of the Brachiograph’s we made to the University so it can used as a display, either for the Students as Co-Creators or for the University in general. We would like to use our model for demonstration purposes in workshops, either to demonstrate our usage of laser printing or to demonstrate the coding and electronics work put into it. We would also like to create a kit containing the necessary materials and techniques to start a project similar to ours.

# Team Reflection

Working on the Brachiograph project has been a transformative experience for our team. We gained quite a lot of knowledge about what goes into a project such as this from the collaborative effort) as well as the technical effort (to the physical design and setup to the coding and electronics that went into it). Although, on reflection, it did suffer from the differing workflows between the team and academics. Various scheduling issues were encountered, from academic duties interfering with the assignment, members' personal lives limiting availability, and misplaced holidays, which led to strained progress and us exceeding our estimated timeline.

Through collaborative problem-solving, creative thinking and managing to rally the team to get together to push through the project, we managed to resolve these issues and finish our project. Ultimately, we should have planned and coordinated ourselves in a more timely manner or set a schedule to follow. While there could still be quite a lot to research, from researching different design possibilities to changing certain aspects of the code, we feel as if we have managed to not only recreate the Brachiograph but also improve through laser printing.

# References

Procida, D. (2019) Brachiograph#, BrachioGraph 0.1 documentation. Available at: https://www.brachiograph.art/en/latest/ (Accessed: 07 March 2024)

1. <https://www.brachiograph.art/> [↑](#footnote-ref-1)