Initial Design of an Automatic Crochet Machine: How to Breakdown Crocheting into Simple Mechanized Movements

A Thesis Presented

By

Sofia Irene Handzy

to

The Department of Mechanical and Industrial Engineering

in partial fulfillment of the requirements

for the degree of

Master of Science

in the field of

Robotics

Northeastern University

Boston, Massachusetts

April 2025

To my dog, Kashi, and my grandmother, Alexandra Handzy.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF TERMS

1. Arduino Uno: microcontroller used to send signals to hardware components to actuate the motors
2. Automated Crochet Machine: a machine designed to repeat the motions of crocheting
3. Blocking: the process of wetting or steaming crochet projects to shape the textile
4. Capstan Tensioner: small rotating 3d printed part that keeps yarn tension steady
5. Chain Stitch: simplest stitch in crochet that forms the base of most projects
6. Crochet Ellipsoid: a 3D shape made from stitches that form a sphere with certain dimensions for radii based on user input
7. Crocheter Subsystem: part of my machine that moves the crochet hook linearly to make stitches
8. Crocheting in the Round: technique in crocheting where you spiral outward from a center point
9. DC Motor: steady motor that spins when connected to a motor driver and microcontroller
10. Double Crochet (dc): a type of stitch that has the height of three chain stitches
11. Fast Fashion: business model where clothes are produced quickly and unethically contributing to global pollution and exploiting low income workers
12. FDM Printing: 3D printing method where thermoplastic filament is heated and extruded layer by layer
13. Forced Labor: when people are made to work against their will
14. Foundation Chain: the first row of chain stitches that forms the base of a project
15. Granny Square: a beginner-friendly crochet pattern that is crocheted in the round
16. Half Double Crochet (hdc): a type of stitch that has the height of two chain stitches
17. Holder Subsystem: part of my machine that grips and rotates the fabric being crocheted
18. Hook Size: the measurement of the width of a crochet hook's shaft (mm)
19. Hook Track: the pathway my crochet hook slides along which is driven by a rack and pinion system
20. IMU (Inertial Measurement Unit): a sensor that measures the motion, orientation, and speed of an object
21. Inline Hook: a crochet hook where the shaft and head are perfectly aligned
22. Kinematic Motion Capture: the method of recording motion using markers
23. Motion Capture System: the full setup of motion capture systems including cameras, markers, and software that records the kinematic motion
24. Motor Driver: electric circuit that helps the microcontroller communicate with the motors

25. Open-loop Control: type of system where commands are sent without checking if they worked
26. Pattern Generator (MATLAB Code): a program I wrote that calculates how big your crocheted piece will be based on the yarn, hook, and tension
27. Rack and Pinion: a mechanical setup that turns rotational motion into linear motion
28. Rigid Body: a group of motion capture markers treated as a single object
29. Servo Motor: a motor that can move to a specific position
30. Single Crochet (sc): a type of stitch that has the height of one chain stitch
31. SLA Printing: 3D printing that uses lasers to cure resin into shapes
32. Slip Stitch: a type of stitch that has the height of one chain stitch
33. Sustainable Manufacturing: a way of making products that reduces environmental damage and treats workers fairly
34. Tapered Hook: a crochet hook where the head is narrower than the shaft
35. Tension: the amount of stretch or slack in the yarn while crocheting
36. Tensioner Subsystem: part of my machine that controls how tight the yarn is while crocheting
37. Textile Waste: unwanted fabric that ends up in landfills
38. Triple Crochet (tr): a type of stitch that has the height of four chain stitches
39. Yarn Holder: part of my machine that feeds the yarn smoothly into the tensioner
40. Yarn Weight: a classification of yarn thickness ranging from super thin (lace - 0) to super thick (jumbo weight - 7)

ACKNOWLEDGEMENTS

Thank you to my thesis advisor, Professor Andrew Gouldstone, for all of his guidance and encouragement throughout this entire project. His passion and commitment for making the world a better place has inspired me for the rest of my life and I am so grateful I was able to work with him this past year. Thank you to Professor CJ Hasson for letting me use his motion capture laboratory and teaching me how to use the motion tracking software. Lastly, thank you to Professor Marguerite Matherne for serving on the thesis committee and providing thoughtful comments that strengthened my work.

Abstract

Recently, crochet designs have become popular on social media and there has been a huge increase in demand for crochet textiles to be manufactured. Currently, there are no crochet machines that exist to mass produce these textiles, and workers make each crocheted piece by hand. This thesis presents the initial design of an automatic crochet machine that is made from off-the-shelf low-cost parts to produce crocheted products in America before outsourcing to other third-world countries. The goal of the machine is to replicate the basic movements of crocheting using simple linear and rotational motions, with the broader purpose of promoting sustainable and ethical textile production. This paper summarizes the kinematic motion capture tests that informed the design of the crochet machine, the complete design process and component selection, and a pattern generator that was developed in MATLAB to estimate textile dimensions based on yarn size, hook size, and user input tension. Although the system is still in early stages, the results demonstrate that crochet automation is achievable and could help reduce the fashion industry's reliance on forced labor while minimizing textile waste.

Need

Currently there are no crochet machines that exist to mass produce crocheted textiles and people make each crocheted piece by hand. Knitting and weaving has been automated for centuries but crocheting has not because of the specific way the stitches are made. Crocheting takes long hours of work and this time commitment has an incredibly negative impact on people's mental and physical long-term health.

The need for a crochet machine stems from both humanitarian and environmental concerns. On the humanitarian side, the manual labor involved in crocheting is performed under exploitative conditions in law-wage regions where workers, including children, are paid far below living wages. Automating this process could reduce the demand for forced labor and help protect vulnerable people from exploitation.

On the environmental side, fashion companies' massive garment output has severely harmed the planet. High volume garment production contributes significantly to global carbon emissions, generates large amounts of textile waste, and pollutes freshwater ecosystems with toxic dyes and chemicals. Although crocheted textiles currently represent a small portion of overall fashion production, as demand grows through social media the environmental footprint of crocheted items is going to increase.

A crochet machine that is low-cost and made in the US could be a potential solution. By reducing reliance on outsourced, labor intensive production, this project aims to provide a sustainable and ethical alternative to traditional crochet manufacturing methods. The long term vision is not just to automate for convenience, but to reimagine textile production in a way that protects human dignity and reduces environmental harm at its source.

Background and Significant Prior Work

Textile Labor

Labor laws in many manufacturing countries are weaker and less enforced than in the United States. The lack of protection allows companies, especially fast fashion companies, to exploit vulnerable populations. Workers in low-income countries are specifically targeted because they have limited alternatives for employment. One of the most heartbreaking outcomes of this system is the widespread use of child labor, which remains a major hidden cost of the fashion industry.

Currently there are 160 million children that are forced into menial and tedious jobs due to the economic circumstances which they were born into. In order to survive, many children stop going to school to work long hours and even work in hazardous conditions. [1] For example, a study into India's garment sector in 2019 found that garment workers have to work 14-16 hours a day, seven days a week, while facing verbal and physical abuse from overseers. [2] Kara also found that 36% of the children documented by the researchers indicated that they were paid nothing for their work. Children become full-time workers and drop out of school to try and support their families. Kara found that 21% of children aged five to ten years and 71% of children aged 11 to 14 were involved in home based garment work.[2]

This exploitation is not confined to India alone. In many manufacturing countries such as China, Cambodia, and Bangladesh, the minimum wage only ranges from a half to a fifth of the living wage required for a family to meet its basic needs. Furthermore, the average worker in an Indian sweatshop makes just 58 cents an hour, and in Bangladesh this drops to 33, linking fast fashion to the cycle of poverty. [2]

Once children enter the workforce, it becomes much harder for them to escape it. Dropping out of school permanently limits their future earning potential, ensuring that poverty, hardship, and exploitation continue into the next generation. Studies have shown that the loss of education at an early age due to forced labor has long-term negative effects not only on individual families but also on broader national economies, stifling innovation and perpetuating dependence on low-wage industries. [2] Unfortunately, many of these children who work in the textile industry have no other option to support themselves and their families and must continue to work in awful conditions.

The global fashion industry depends on this cheap labor force to maintain its low production costs and quick turnaround times. Without meaningful reform to labor standards enforcement, education access, and living wage policies, the cycle of forced labor will continue to fuel the production of garments, including crochet textiles.

Fast Fashion

High fashion brands typically produce custom designs made to order, creating limited quantities of carefully tailored garments that sell at very high prices. In contrast, fast fashion brands mass produce huge quantities of average designs as quickly and cheaply as possible. Fast fashion companies are able to flood the market with low-cost clothing by prioritizing speed and volume over ethical and sustainably sourced materials, which encourages constant consumerism that leads to waste.

This business model has devastating environmental consequences. National Geographic reports that an estimated 85% of textiles end up in landfills every year, the equivalent of a garbage truck's worth of clothing being dumped every second. [3] Many of these cheap products are made from synthetic fibers like polyester which does not compose which contributes to long-term soil and water contamination.

The production side also contributes greatly to environmental consequences. The fashion industry generates 1.2 billion tonnes of carbon dioxide per year, which is 10% of global carbon emissions. [3] Textile production also requires massive amounts of water. For example, it takes 2,700 liters of water to produce a single cotton t-shirt, which is enough drinking water for one person for two and a half years. Also, dyeing textiles accounts for 20% of the global industrial water pollution and it releases harmful chemicals into waterways which contaminate drinking water and destroy aquatic ecosystems. [3] The mass production of textiles caused by fast fashion brands is severely polluting the environment as well as exploiting low-income workers and children.

To begin to understand how to automate crocheting to minimize waste pollution I researched all about the art of crocheting. Crochet is a unique fiber art with distinct materials, motions, and stitch structures that differ from knitting or weaving. Breaking down these fundamentals was essential to designing a machine that could replicate the movement, tension, and flexibility that human crocheters intuitively control by hand.

Crochet Basics

Crocheting is a fiber craft that involves one hook and yarn, as shown in Figure 1.

Fig 1: Crochet [4]

Crocheting is extremely versatile and users are able to make any shape they want using the variety of stitches that is available.

Weight sizes and hook sizes

When you begin crocheting you choose a yarn and a hook size. Yarn weights vary from 0-7, with 0 being the thinnest and 7 being thickest, as shown in Figure 2.

Fig 2: Yarn weights and what they are used for [5]

Hook sizes vary from 2.25mm to 30mm. Figure 3 shows all the crochet hook sizes that are available.

Crochet Hook Size Chart (US & UK)

Millimeter Size	US Size
2.25 mm	B-1
2.5 mm	--
2.75 mm	C-2
3.125 mm	D
3.25 mm	D-3
3.5 mm	E-4
3.75 mm	F-5
4 mm	G-6
4.25 mm	G
4.5 mm	7
5 mm	H-8
5.25 mm	I
5.5 mm	I-9
5.75 mm	J

Millimeter Size	US Size
6 mm	J-10
6.5 mm	K-10½
7 mm	--
8 mm	L-11
9 mm	M/N-13
10 mm	N/P-15
11.5 mm	P-16
12 mm	--
15 mm	P/Q
15.75 mm	Q
16 mm	Q
19 mm	S
25 mm	T/U/X
30 mm	T/X

Fig 3: Hook sizes and their labels [6]

Hooks are either tapered or in-line, as shown in Figure 4. Tapered hooks are easier to use because they give a little more length to the diameter of the hook. In-line hooks hold better constant tension and stitches look more uniform.

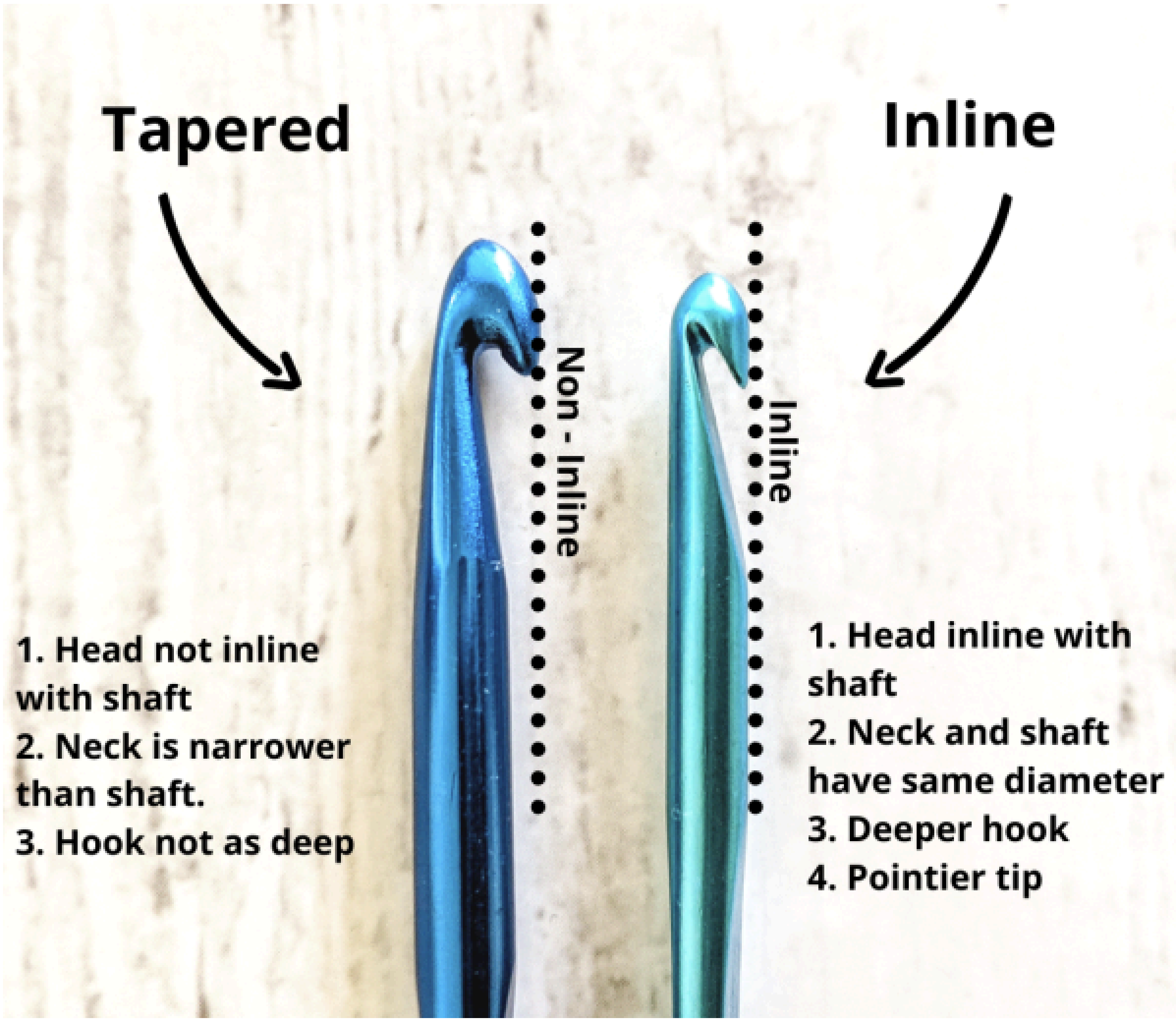

Fig 4: Tapered vs. Inline Crochet Hooks [7]

Crochet Stitches

All crochet stitches are made from loops - the dimensions of each stitch depend on the dimensions of the loops, or chains, that the material is made from. Figure 5 displays a figure of the dimensions of one chain stitch or "loop". The length, width, and height are all determined by the diameter of the yarn, the diameter of the hook, and the tension the user applies during their crocheting. I defined the equations to determine the dimensions of a chain stitch because they are extremely important in estimating the overall size of a crochet project. These equations helped me further develop my code and define tension factors later on during the thesis.

Here are the equations to define the dimensions of a chain stitch:

Equation #	Name	Equation
(1)	Chain Length	$\text{Chain}_{\text{length}} = \text{tension} \times (\text{diameter}_{\text{yarn}} + \text{diameter}_{\text{hook}})$
(2)	Chain Height	$\text{Chain}_{\text{height}} = \text{tension} \times (2 \times \text{diameter}_{\text{yarn}} + \text{diameter}_{\text{hook}})$
(3)	Chain Width	$\text{Chain}_{\text{width}} = \text{diameter}_{\text{yarn}}$

Table 1: Chain stitch dimensions

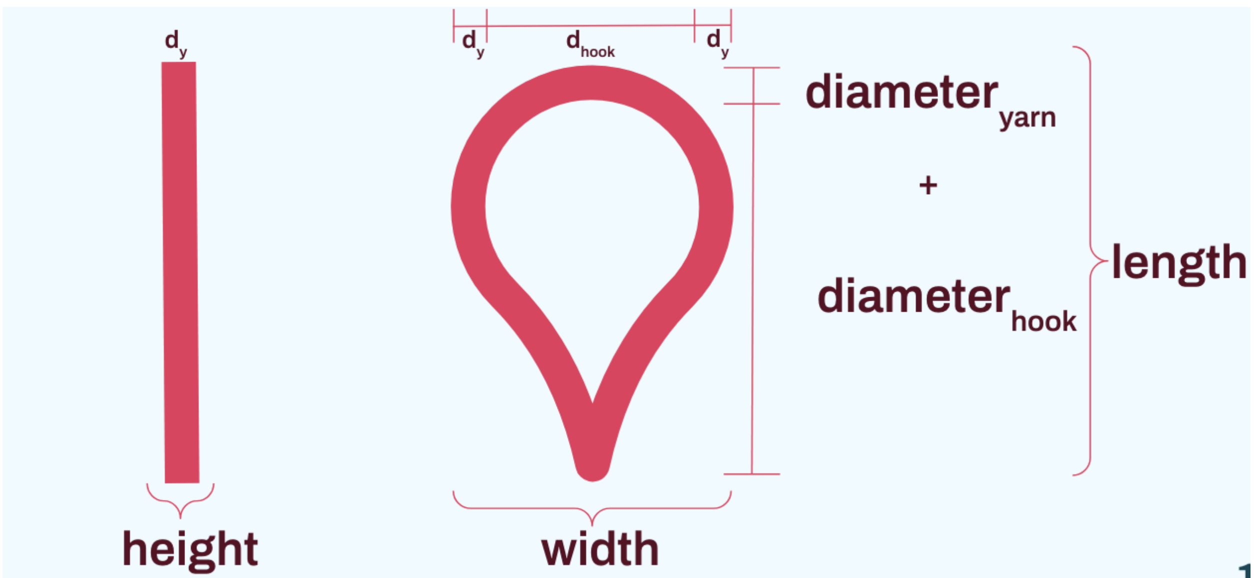

Fig 5: Dimensions of a chain stitch with “perfect” tension

All crochet stitches are made up of these loops and the length of these loops then becomes the height of single crochet, half-double crochet, double crochet, and triple crochet. Figure 6 shows an example of how different heights vary between stitches.

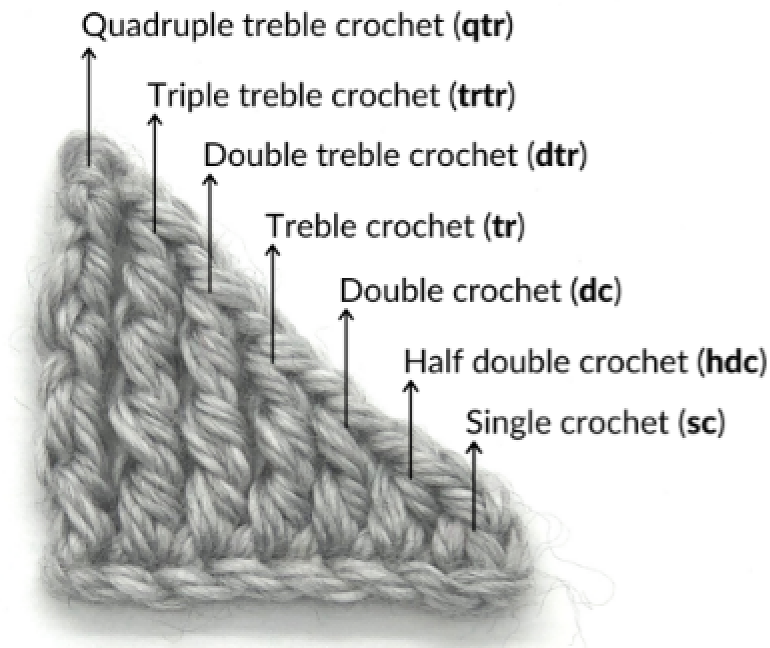

Fig 6: Heights of qtr, trtr, dtr, tr, dc, hdc, and sc [8]

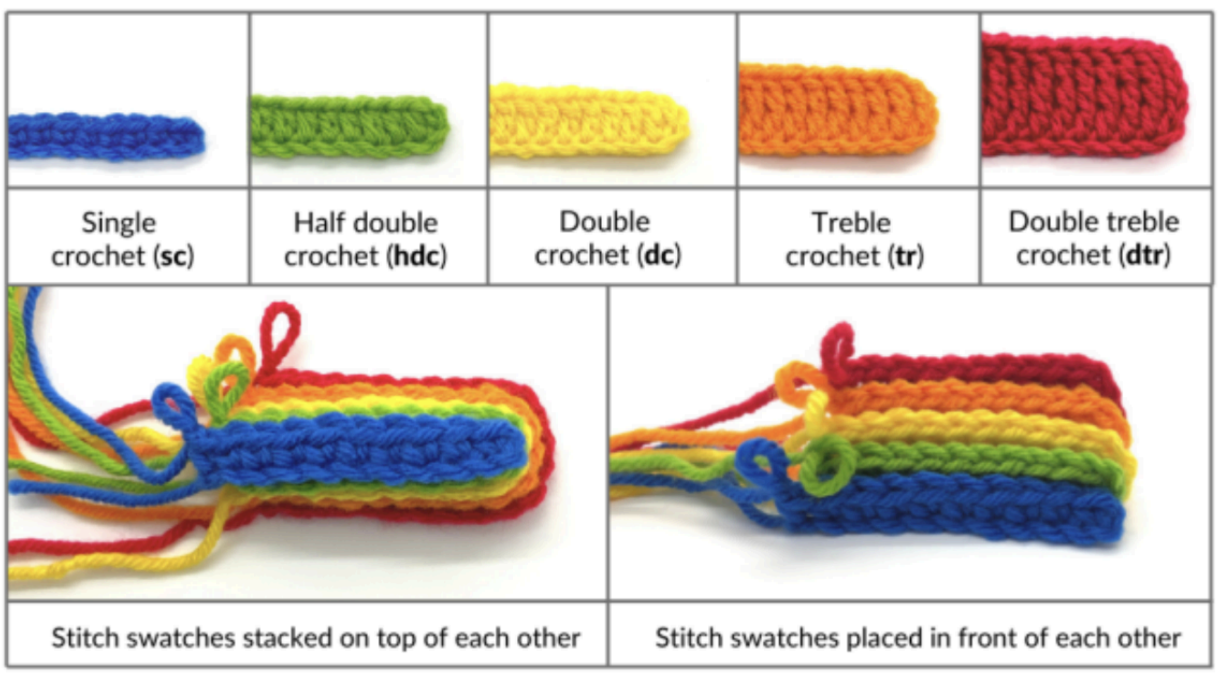

Fig 7: Heights of sc, hdc, dc, tr, dtr compared to one another [8]

The most common stitches in crochet are chain, slip, single, half double, double, and triple.

In general, each stitch height is equivalent to a multiple of a chain length. Figure 8 displays a diagram showing stitches and their height in chains.

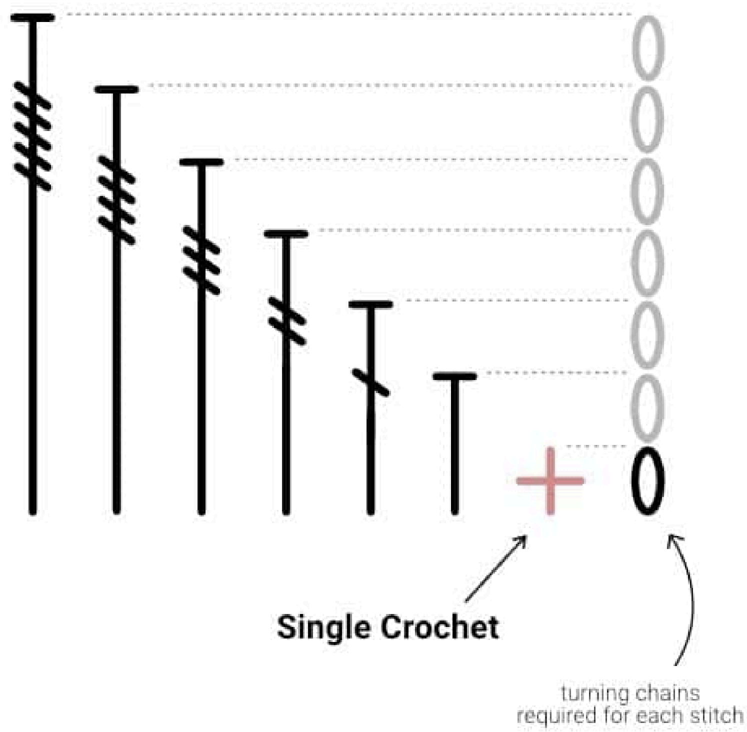

Fig 8 : Crochet stitches and their heights in chain stitches [9]

Crocheting in the round vs rectangle

Crocheting can either be done in a rectangular shape or it can be done in the round. If something is crocheted in a rectangular shape the crocheter crochets a chain of initial loops and that length is the set length for the rectangle. The height of the rectangle is determined by the amount of rows the crocheter crochets and the stitch type. Each row that is added is the height of the tallest stitch in that row and when one row is over the crocheter needs to flip their work 180 degrees so they can continue onto the next row. This is better shown in Figure 9:

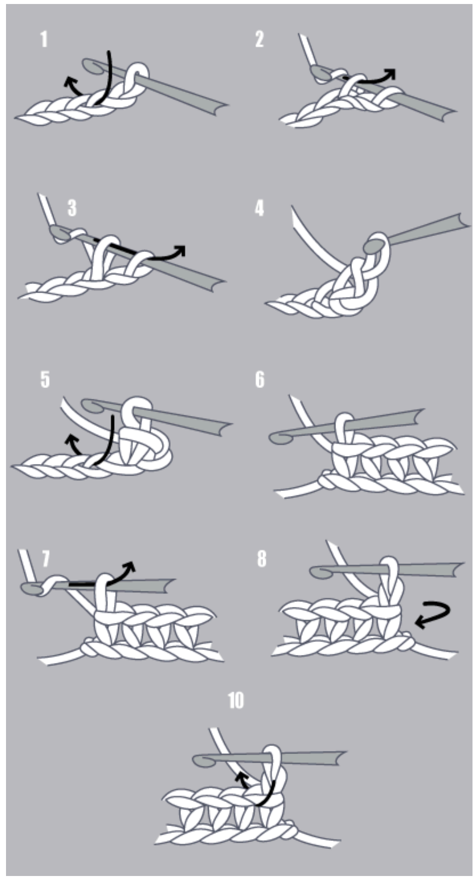

Fig 9: Single crochet and turning work [10]

Steps 1-6 show a crocheter crocheting one row of single crochets. Step 7 requires the crocheter to make a chain and then step 8 shows flipping the work 180 degrees. This allows the crocheter to start the next row.

Crocheting in the round is different from crocheting in a rectangle because you begin with one loop and you crochet a certain number of stitches into the first loop. This forms a circumference and is the crocheters first round. A round is completed once the hook has traveled 360 degrees back to its original starting point.

Fig 10: Crocheting in the round [11]

Granny Squares

Granny squares are designs beginners often learn first because the pattern is very repetitive and only requires knowledge of chain, double crochet, and slip stitches. Granny squares are crocheted in the round. Figure X displays a diagram of a granny square

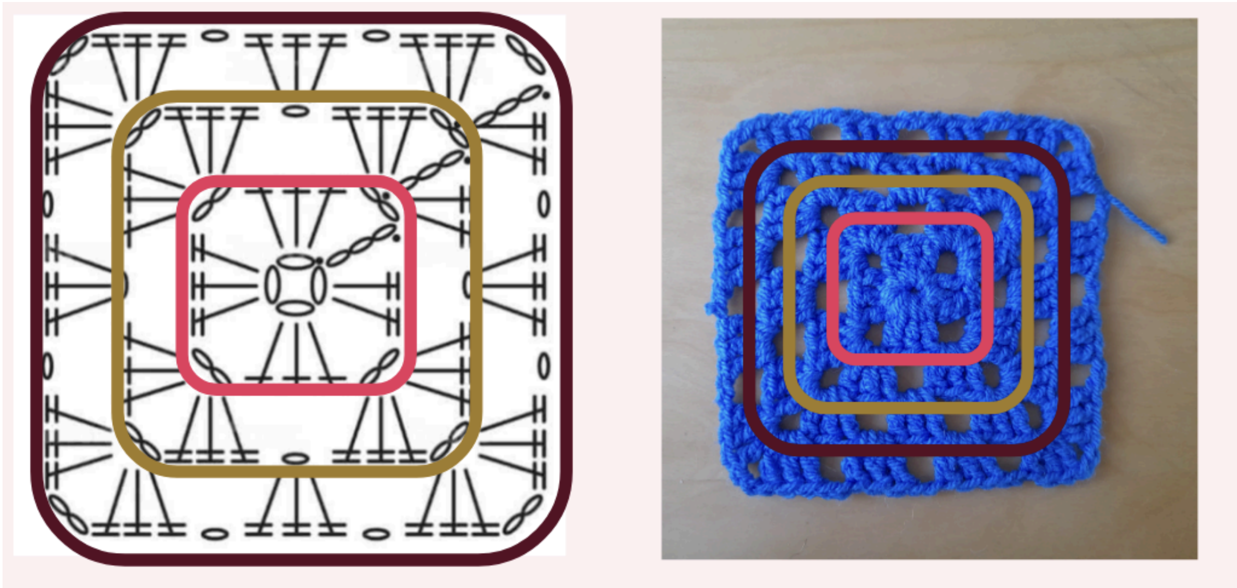

Fig [11]: Crochet granny square diagram and real granny square

Blocking

Blocking is a technique used in crocheting to stretch out the yarn to fit the desired shape the user wants. Users either dry, wet, or steam block their work. Blocking involves getting the project wet or damp so that the fibers of the yarn can loosen and crocheters can manipulate the piece to fit their desired shape. Block also helps with long term use of the crochet project and will prevent it from curling up.

Existing Crochet Machines

Currently, there are very few crochet machines that exist, and none have yet achieved full automation of basic crochet stitches at a reliable and scalable level. Most crocheted textiles are still made entirely by hand and attempts at mechanization are still in the prototype stage.

One of the first notable attempts was the CroMat, developed by Jan Lucas Storck. [12] The CroMat showed early promise by demonstrating that slip stitches could be successfully automated. However, the machine faced significant limitations. It was unable to produce more complex stitches like single crochet or half double crochet, largely due to mechanical inaccuracies and component flexibility. Storck specifically cited the use of 3D-printed parts as a major source of error, as these parts could not consistently achieve the precision and strength required for reliable stitch formation.

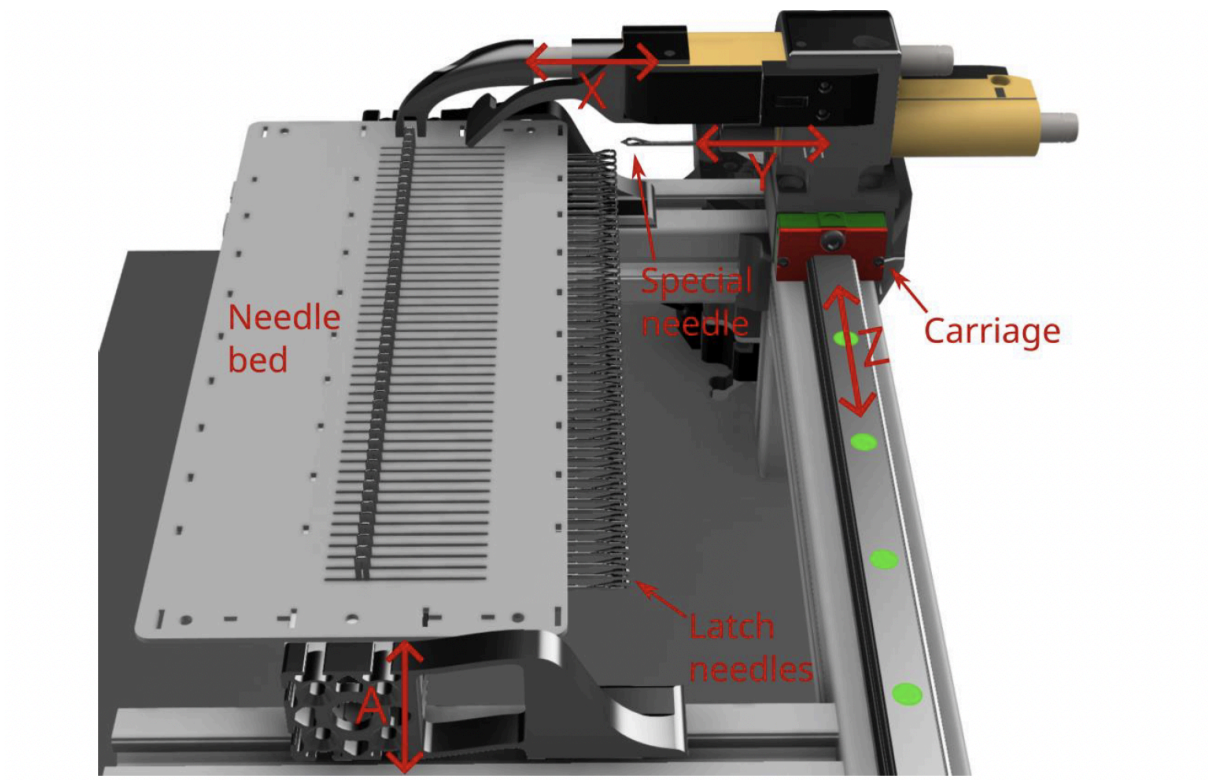

Fig 12: CroMat by Storck [12]

Another machine that exists is the Croche-matic made by Gabrielle Perry at Harvard University. Her machine has a success rate of 50.7% for single crochet and can crochet at most 4 single crochets in a row. Her machine was the first crochet machine to successfully make increase and decrease stitches when crocheting in the round. Gabrielle also noted in her paper that crocheting in the round was the same as crocheting a rectangle. She cited the inability of securing the fabric as the main source of error in her machine. [13]

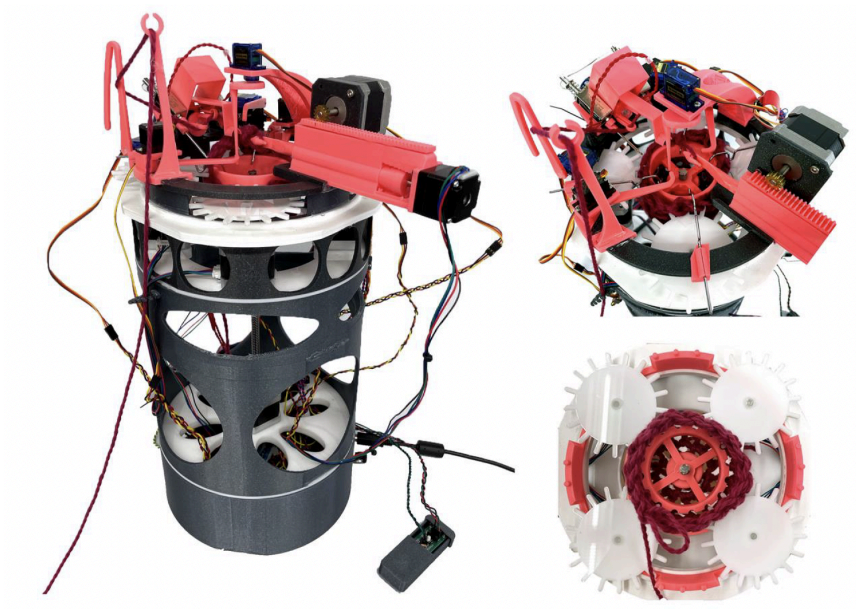

Fig 13: Croche-matic by Gabby Perry [13]

Nix and Sprecher came up with an idea of using a KUKA robot arm to crochet, but this idea has not been implemented in the real world yet. Their idea was to move the textile rather than the hook. It uses 3d scanning of existing textiles to locate the openings for insertion of the hook. The only thing Nix and Sprecher were uncertain about was how to add a new row by flipping the textile. [14]

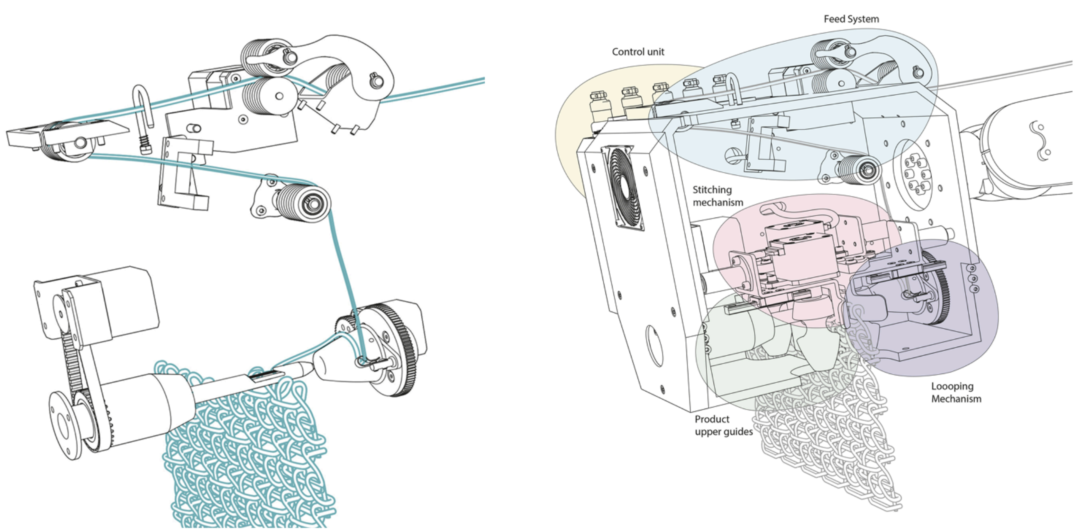

Figure 4: Left: A looping mechanism designed to adopt the angles of the manual yarn gripping process. Right: Crochet robotic end effector mechanisms (MTRL 2023, Technion IIT)

Fig 14: KUKA robot arm proposed by Nix and Sprecher [14]

Beyond academic prototypes, some adaptive solutions exist in the crochet community for people with disabilities. For instance, individuals who cannot use both hands to crochet often secure their crochet hook to a stand or tabletop with a set screw, creating a semi-automated setup that stabilizes the hook while they manipulate the yarn.

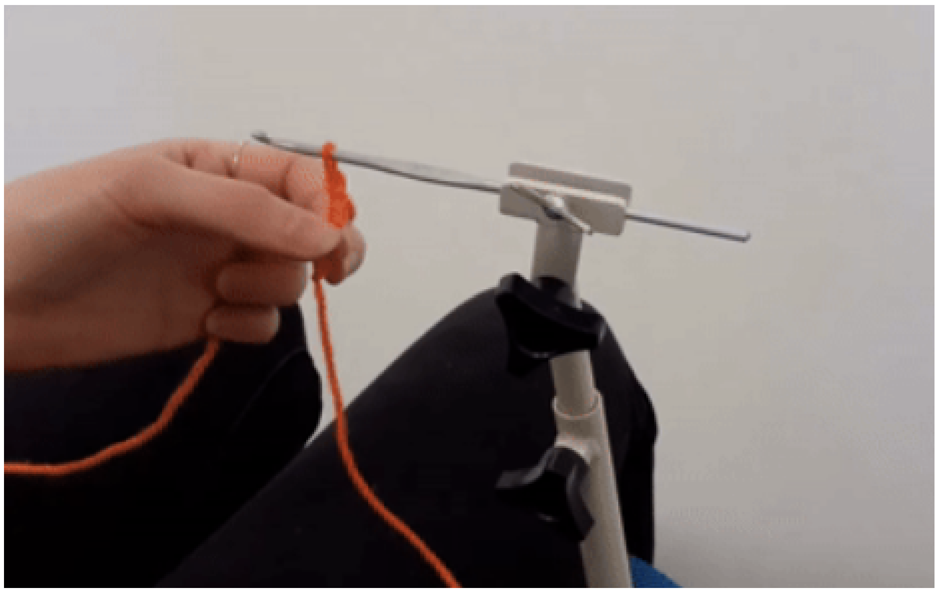

Fig 15: Crochet hook holder [15]

This insight -that the hook itself does not always need to move - inspired an important design choice for my project. Rather than trying to replicate every motion a human crocheter makes, it is possible to design a system where some motions (such as yarn tensioning and fabric movement) are automated independently, allowing the hook motion to remain simple and linear.

These early prototypes highlighted both the possibilities and the persistent technical challenges of crochet automation. Building on their insights, my design focused on simplifying the hook motion, securing the fabric, and maintaining consistent yarn tension.

Design Solution

Design of My Machine

My machine is made with an 8020 Aluminium T-slotted frame [20], two FS90R servo motors [21], two 5V DC motors [22][23], a motor driver [24], and the rest of the components were 3D printed from PLA on SLA printers. I used Ultimaker and Prusa printers to print every component out. Every electronic component is connected to an Arduino Uno [25].

The machine first actuates the hook up, then the tensioner wraps the yarn around the hook, the hook actuates down, and then the holder spins the circular plate and the second servo motor re-tensions the new stitch. This process is repeated so that a chain length can be formed.

Here is the CAD Model for my crochet machine:

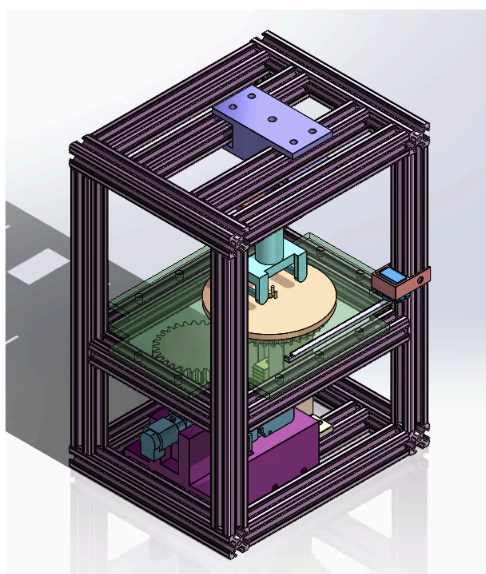

Fig 16: Isometric view of my crochet machine

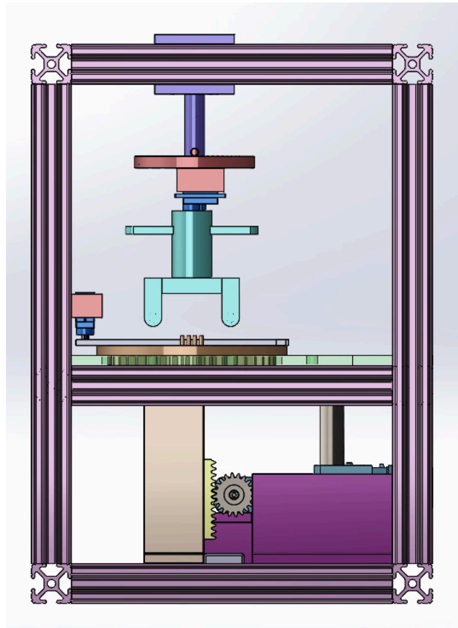

Fig 17: Side view of my crochet machine

The design of my machine was broken up into 3 subsystems: the crocheter, the tensioner, and the holder.

The Crocheter:

The crocheter subsystem is composed of a rack and pinion, a hook track, motor holder, and a 5V DC motor that spins the pinion. The crochet hook is press-fit into the yellow rack and actuates between ½ inch to a full inch.

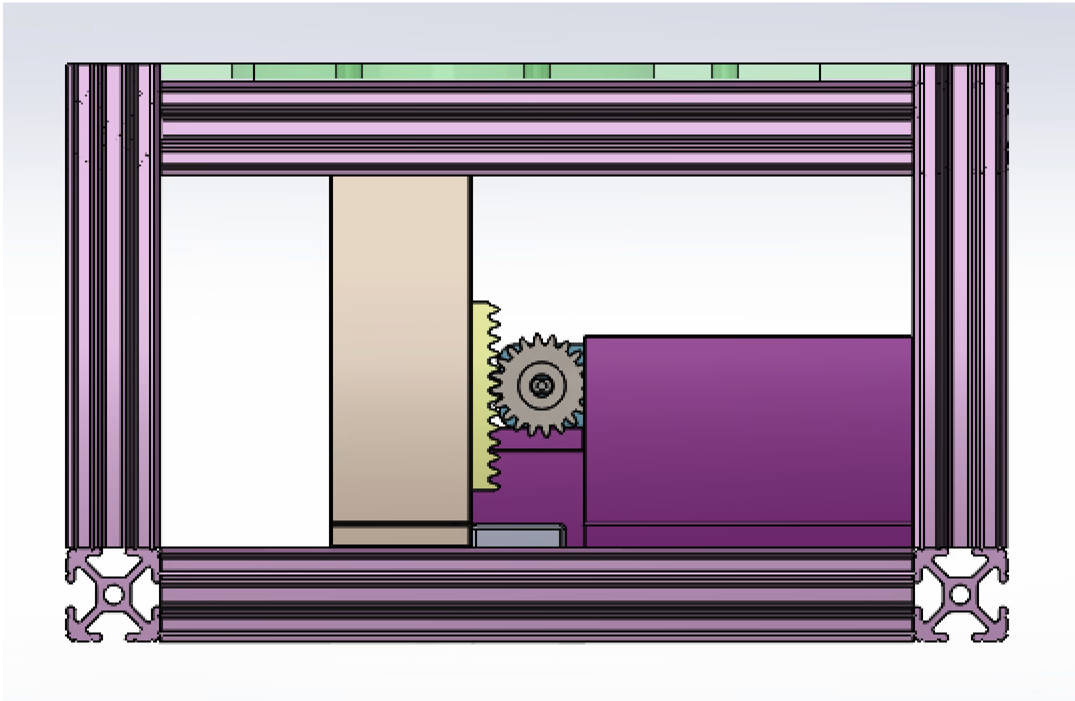

Fig 18: Close up of rack and pinion, hook track, and motor holder.

The Tensioner:

The tensioner subsystem is composed of a yarn holder, a servo motor, and a capstan tensioner. The servo motor is attached to the yarn holder and the capstan tensioner is attached to the servo motor. The yarn holder and servo motor remain static while the teal capstan tensioner rotates the yarn around the crochet hook.

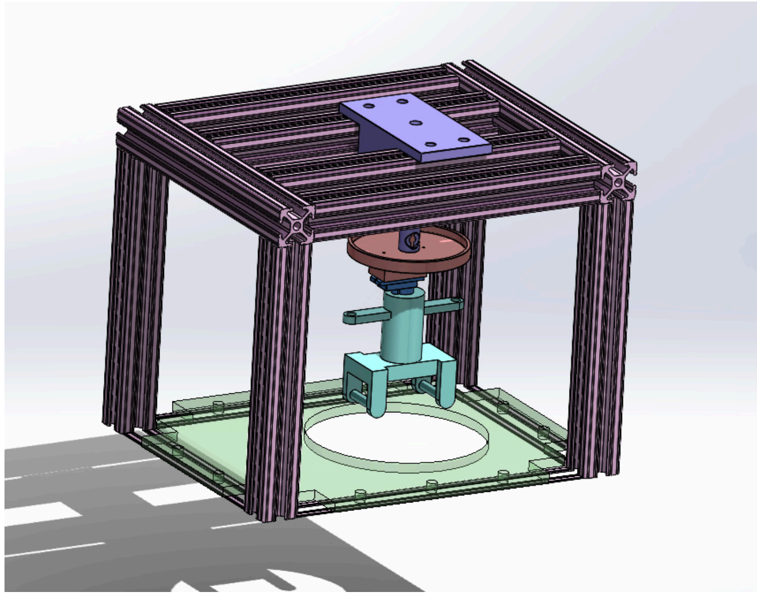

Fig 19: Tensioner system of crochet machine

The Holder:

The holder system is composed of an acrylic plate, a servo motor, a 5V DC motor, a gear with a shaft, and a gear with a circular plate on top with pegs. The DC motor spins the gear with the shaft which then spins the circular plate. The circular plate holds the crochet work and rotates it so that the machine can crochet in the round. The servo motor is connected to a 3d printed stick that re-tensions the stitch after the crocheter makes a stitch.

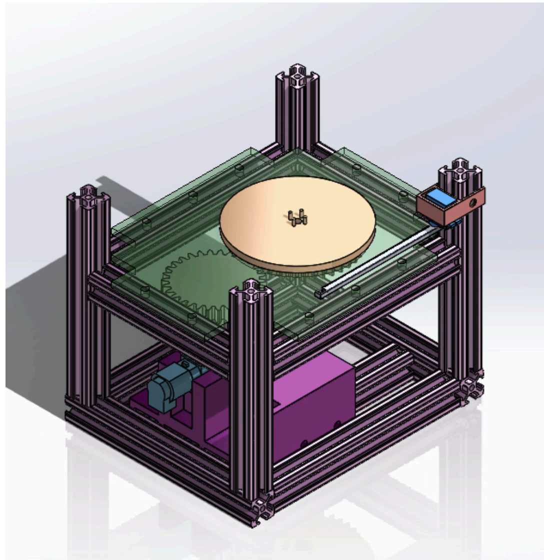

Fig 20: Holder system of crochet machine

Code

My pattern generator was coded on MATLAB and helps users estimate the size of their rectangular projects or spherical projects. All clothes are made out of rectangular textiles and lots of stuffed animals are made from spherical shapes.

I first set a dictionary of weight sizes of yarn and sizes of hooks into the database. The diameter of the yarn depends upon the material it's made from and also what weight it is - this data can be found on the internet or is displayed on the packaging of the yarn. Once I set the dictionary, I defined the height, length, and width of a chain stitch using the diameter of the yarn and diameter of the hook. In my code, I assume that "perfect" tension means there is no extra or loose yarn that makes the stitch smaller than these equations or larger than these equations.

I used the equations from Table 1 and diagram from Figure 5 to help me develop my code. The input to my rectangular code is the hook size, the weight size, the tension scale, and either the number of loops you want in your initial chain, or the length you want your chain to be in inches or millimeters.

Figure 21 displays a screenshot of the gui:

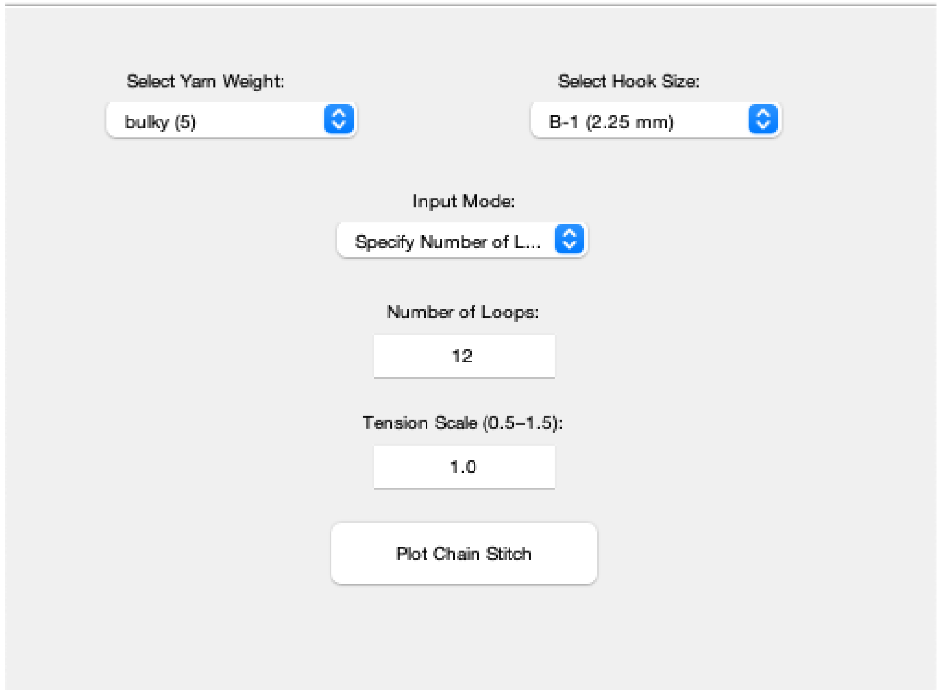

Fig 21: Input of crochet code with number of loops specified

The code outputs the height and length of the chain foundation that is necessary for the user to envision their pattern. This length is the length of their desired rectangle.

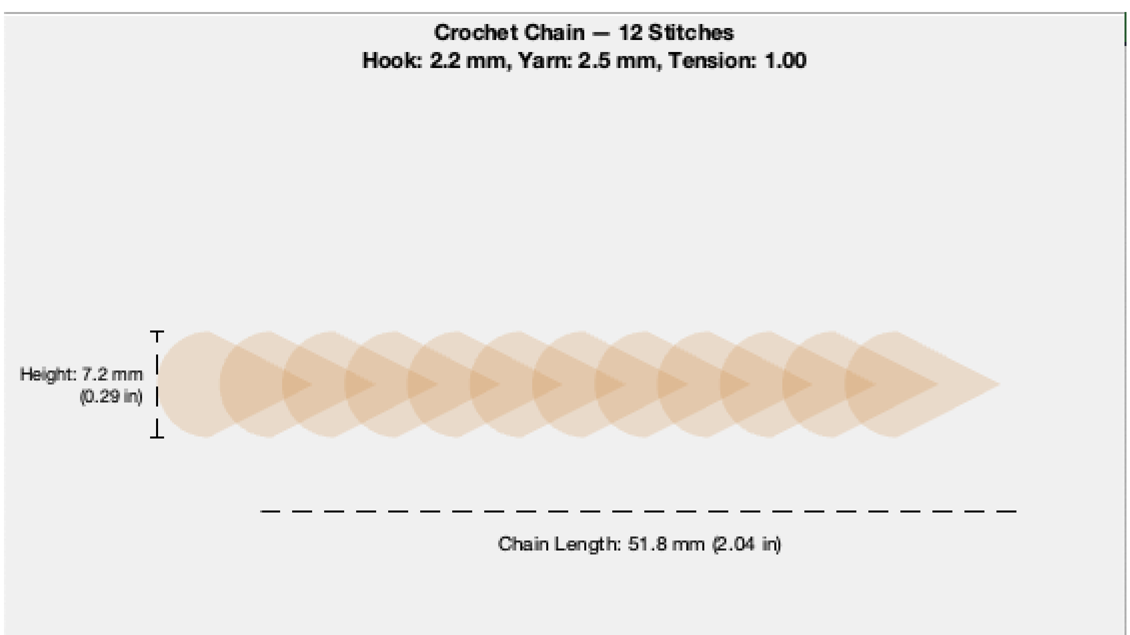

Fig 22: Output of crochet code with number of loops specified

If the user wants to specify a certain length in inches or millimeters, the code outputs the recommended amount of chain stitches to achieve their desired length.

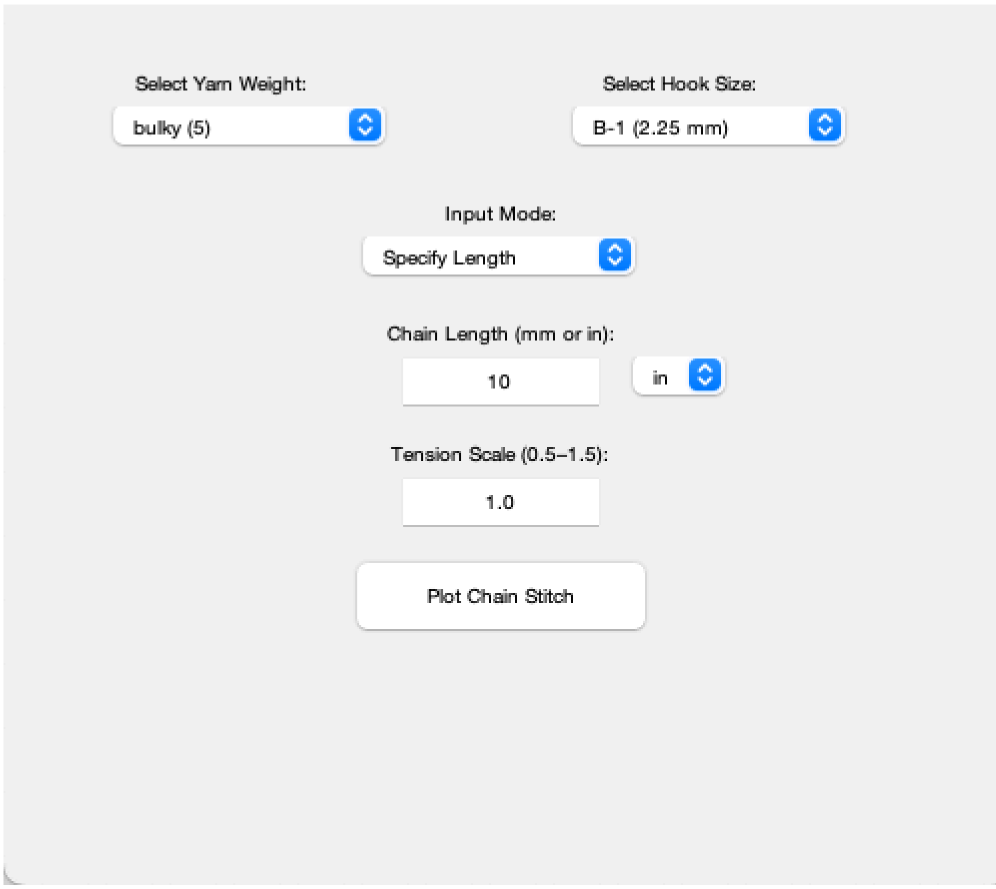

Fig 23: Input of crochet code with length in inches specified

Output:

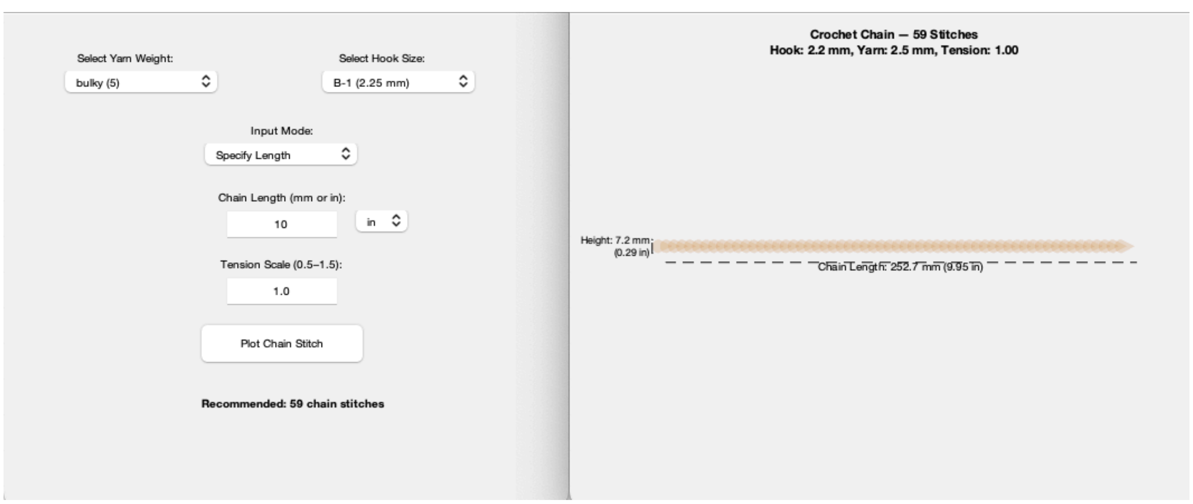

Fig 24: Output of crochet code with length specified and recommended number of chains (59)

My code also works for crocheting in the round, specifically making ellipsoids. The input is the maximum radius (in stitches) and the amount of constant rows the user desires. The code then outputs a graph of how many stitches are in each row and which rows are increasing, constant, and decreasing.

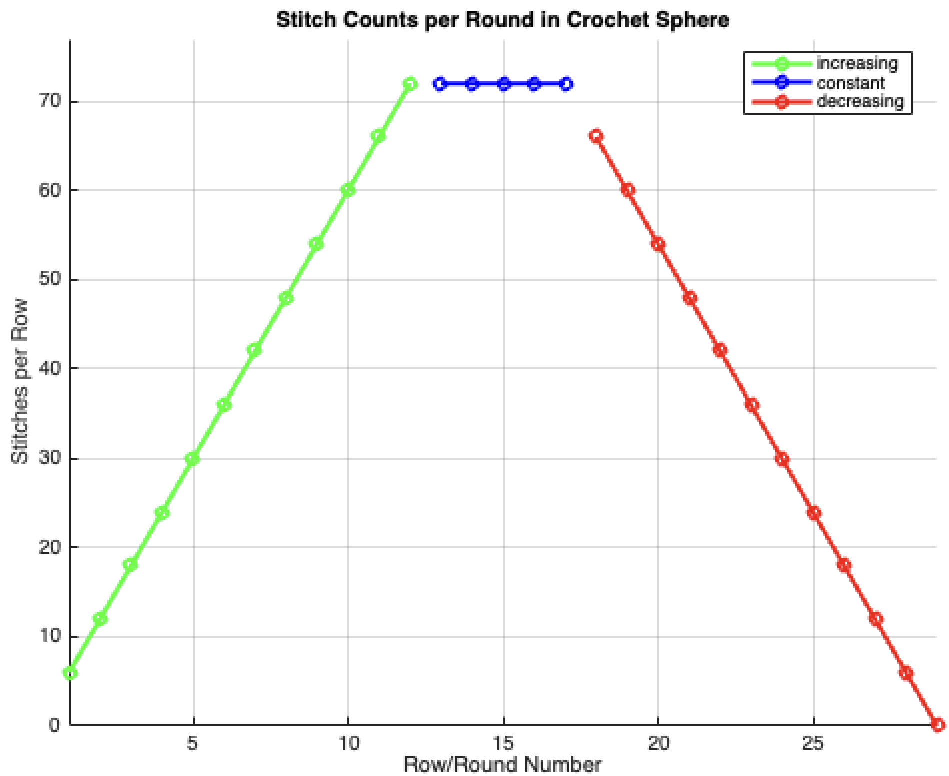

Fig 25: Graph showing Row Number and stitches per row

The code also outputs a 3d graph of the ellipsoid that the user is trying to make, shown in Fig 26:

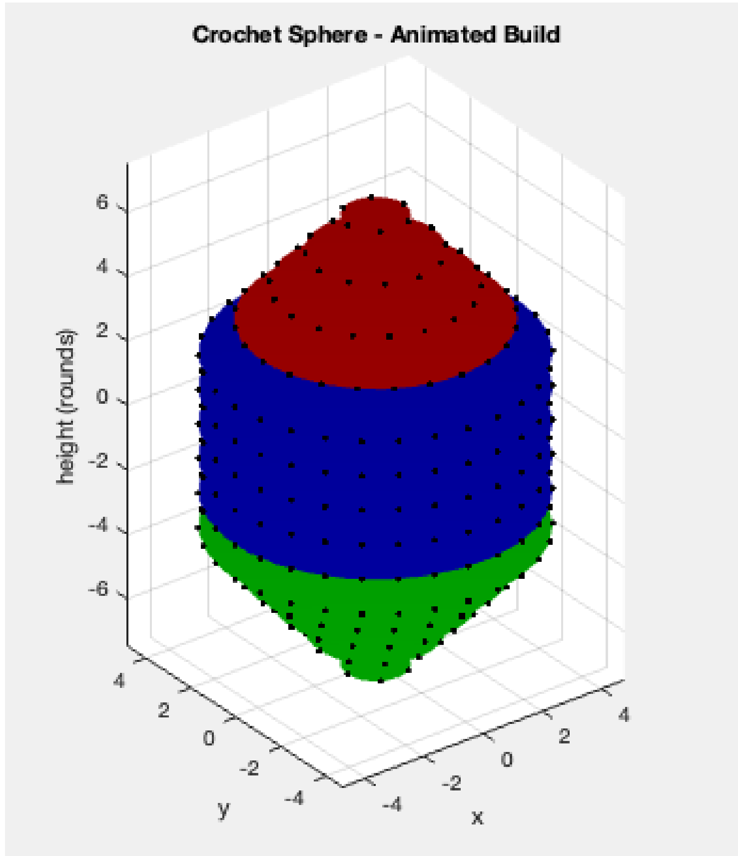

Fig 26: 3d plot of ellipsoid with max radius = 30 stitches and 5 constant rows.

Both scripts are extremely useful in helping determine the size of crocheted projects.

Design Process

Kinematic Experiments

I conducted three different kinematic motion tests - one with an IMU, one with an at home motion capture set up, and one with an actual motion capture setup. The purpose of tracking the motion of the hook was to determine if the crochet motion could be automated using gears. By studying the motion of the hook and what exact location the base of the hook is throughout a crochet stitch we can better mimic the motion using actuators. I wanted to see if the motion of the hook could be solely linear and have other subsystems make up for the rotational motion of the hook.

The first kinematic motion test I completed was tested using an accelerometer. The materials I used were a Qwiic Readboard [16], 6 DoF IMU [17], my computer, a 4.5mm crochet hook, and weight 5 acrylic yarn.

The test procedure is as follows:

1. Start by calibrating the accelerometer
2. Verify the accelerometer data is being received by the computer
3. Start crocheting and complete the task
4. Once done with the task, stop recording the data and export
5. Cut the first and last couple of data points
6. Repeat steps 1-4 3 times per task (to get average results)
7. Save all the data and graph the points to see the motion of the back of the crochet hook

Tasks to test:

- 6 chain stitches
- 3 double crochets
- Granny square with one round
- Granny square with 2 rounds

The results from this experiment were not what I expected and I unfortunately could not use the data I collected since each accelerometer I tested failed to calibrate correctly.

I tried troubleshooting the IMU by changing the sampling rate, using more than just one IMU and comparing the data, and changing the speed at which I outlined the square. Unfortunately, none of these methods worked and I needed to pivot to another kinematic experiment.

The second kinematic experiment I conducted was with an at home motion capture setup. The materials I used were: 3 logitech desktop cameras [18], PFTrack (software) [19], a 5.5mm hook, and weight 5 yarn.

The test procedure is as follows:

1. Record a video of crochet hook being moved in a square formation
2. Verify the tracking software can track the hook with simple motion
3. Start crocheting and complete the task

4. Once done with the task, stop recording the data + export
5. Cut the first and last couple of data points
6. Repeat steps 1-4 3 times per task (to get average results)
7. Save all the data and graph the points to see the motion of the back of the crochet hook

Tasks to test:

- 6 chain stitches
- 3 double crochets
- Granny square with one round
- Granny square with 2 rounds

Unfortunately, the software I used wasn't tracking the back of the hook properly and I needed to pivot. I tried troubleshooting by switching the software and using Blender, but that only made the tracking worse.

I reached out to Professor CJ Hassen in the Bouve College of Health Sciences at Northeastern University because he has a motion capture system set up in his lab. He very graciously let me use his lab to record data.

I set up 10 motion capture markers in total: 3 on the crochet hook, 3 on my hand, and 4 on my wrist. The setup is shown in Figure 27:

Fig 27: Motion capture set up

The test began with calibrating the motion capture system to ensure that the markers were being tracked properly. I then attached the markers to the 3D printed 3 axis holder, 3 markers to my hand, and 4 markers to my forearm. The motion capture system recorded the x-y-z locations versus time for every marker when crocheting 3 granny squares, 10 chain stitches, 10 double crochets, and 10 slip stitches. I was able to export each marker's coordinates into a csv file and was able to make three rigid bodies - the hook, my hand, and my forearm. The x-y-z locations of the rigid bodies were also exported in a csv file.

This test was successful and I was able to grab data from the experiment. Figure 28 displays plots of the motion of the hook, my hand, and arm trajectory vs time.

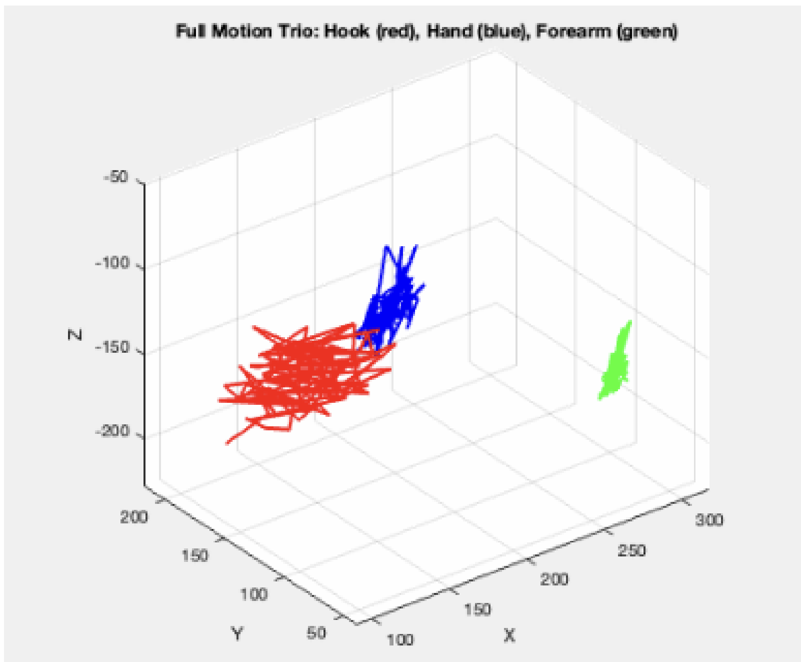

Fig 28: Motion of hook, hand, and forearm while crocheting a full granny square

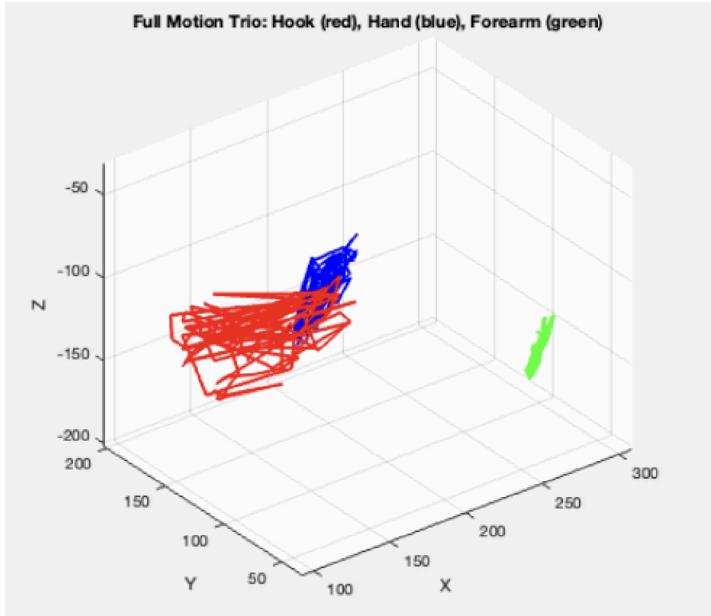

Fig 29: Motion of hook, arm, and forearm while crocheting 10 double crochets

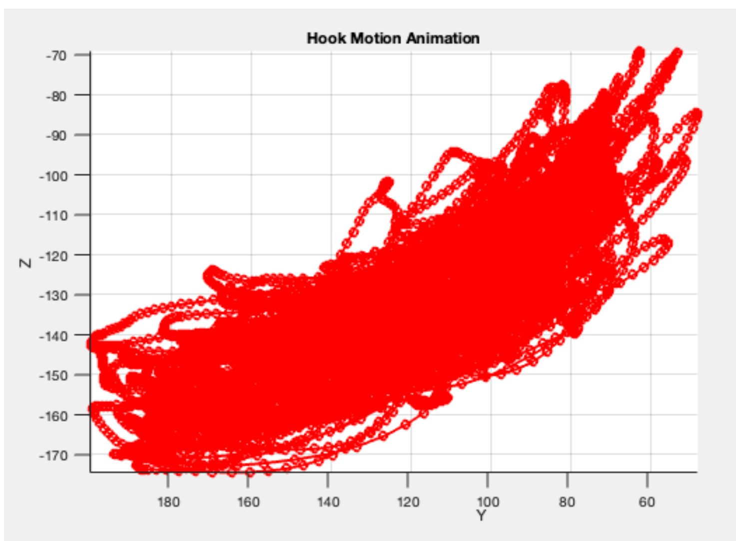

Fig 30: Motion of hook on the ZY plane when crocheting a granny square

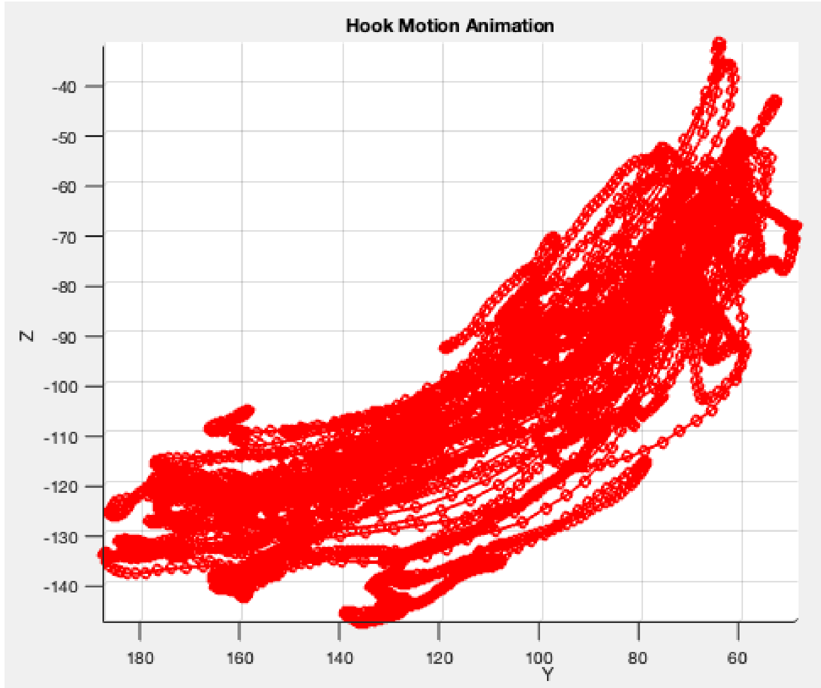

Fig 31: Motion of the hook on the ZY plane when crocheting 10 double crochet stitches

When overlapping the motion onto a video of me crocheting, the motion of the hook seemed linear which impacted the design of my machine. I was able to set the crochet hook to actuate linearly because of this discovery.

Results

My final prototype of the crochet machine was able to successfully crochet one chain stitch which validated the basic mechanical concept of automated crocheting. The machine was able to complete the entire cycle of making a chain stitch: lifting the hook, wrapping the yarn around the hook using the tensioner subsystem, pulling the yarn through the loop to make a new stitch, and rotating the fabric hodler to prepare for the next stitch. This sequence confirmed that simple, low-cost motors controlled through open-loop commands could replicate the fundamental mechanics of forming a crochet stitch.

Although the machine was only capable of reliably producing a single stitch at a time, this result is significant. It proves that crochet automation is possible using relatively simple actuation: a linear rack and pinion system for the hook, rotational tensioning of the yarn, and controlled rotation of the growing textile. This mechanical division of tasks mimics the way human crocheters intuitively coordinate their hands and arms, and validates the original hypothesis that crochet could be broken down into independent, mechanized subsystems.

The motion capture experiments conducted earlier in the project were critical in informing these design choices. By tracking the three dimensional movement of the crochet hook during manual crocheting, I discovered that the hook's motion is primarily linear with minor rotational deviations which justified the use of linear actuation for the prototype. Without this kinematic data, a more complex and unnecessary mechanical system might have been pursued.

The MATLAB pattern generator also worked as expected. It accurately estimated the foundation chain length and width for rectangular textiles based on user inputs like hook size, yarn weight, and tension scale. The pattern generator also produced stitch count graphs and three dimensional visualizations for ellipsoidal projects. This provides a foundation for future integration with machine controlled pattern execution.

Despite the positive results, several limitations were identified. The machine currently lacks closed-loop feedback, meaning any mechanical error or missed stitch cannot be automatically corrected. Material inconsistencies, particularly slight flexing in 3D-printed components, also contributed to occasional inaccuracies in hook positioning and yarn handling.

Overall, the successful formation of a chain stitch with a fully mechanized system serves as a strong proof of concept. The results confirm that the basic motions of crocheting can be replicated mechanically, laying the groundwork for future development of more robust, versatile, and automated crochet machines.

Future Work

While the initial prototype successfully formed a single chain stitch, there are several important areas for future development to move closer to a fully automated crochet machine.

First, the machine’s ability to create multiple stitches in a continuous sequence needs to be improved. Currently, the system is open-loop, meaning it assumes every motor action is executed perfectly without real-time feedback. Introducing closed-loop control would allow the machine to verify the completion of each stitch, correct small errors automatically, and significantly improve reliability.

Second, the mechanical precision of the system could be enhanced by using higher-quality materials and components. Since I printed all the components using SLA, it introduced part flexing and dimensional inaccuracies, which lead to rough hook motion. If I were to use FDM, then the parts would be less dimensionally inaccurate and could lead to smoother and more repeatable hook motion. Additionally, replacing the simple DC motors with stepper motors or servo motors that offer finer positional control would allow more complex crochet stitches such as single crochet, half double crochet, double crochet, and triple crochet to be attempted.

Another important improvement would be adjustability. The current prototype is designed for a fixed hook size. Future iterations should allow users to input their desired yarn and hook parameters, with the machine automatically adjusting tension, hook motion, and stitch spacing accordingly. Designing adjustable guides, modular hook mounts, and variable tension systems will be critical to making the machine more versatile and accessible to a broader range of crochet projects.

In addition to mechanical upgrades, the pattern generator could be expanded. Currently, it estimates the general dimensions of projects but does not produce specific stitch-by-stitch instructions. Future updates could include graphical interfaces for designing custom patterns, color-coding for multi-color projects, and the ability to output machine-readable pattern files. Eventually, integrating the pattern generator directly with the machine would allow for fully autonomous project execution from design to fabrication.

Computer vision is another promising avenue for future work. Incorporating cameras and simple vision algorithms could allow the machine to visually confirm successful stitch formation, identify yarn tangling or errors, and dynamically adjust mid-process without human intervention.

Finally, long-term future work could explore expanding beyond chain stitches to full 3D crochet structures. This could include creating spherical shapes, wearable items like hats and scarves, and even large-scale modular textile panels, pushing the boundaries of what automated crochet technology could achieve.

By continuing to improve mechanical precision, control strategies, material flexibility, and software integration, this research can evolve into a fully capable crochet automation platform — one that reduces reliance on manual labor, enables sustainable textile production, and opens up new creative possibilities for designers and makers alike.

Summary and Impact

This thesis successfully demonstrated the initial prototype of automating crochet using simple, low-cost mechanical systems and open-loop control. The final prototype was able to create a single chain stitch autonomously, proving that the core motions required for crocheting can be broken down into a combination of linear and rotational movements actuated by motors. Through motion capture experiments, the natural motion of a human crocheter was analyzed and used to guide key design decisions, particularly the use of a linear rack-and-pinion system for the hook motion.

In parallel with the mechanical design, a MATLAB based pattern generator was developed to help users estimate the dimensions of their crochet projects based on hook size, yarn weight, and tension factors. This tool also laid the groundwork for future integration between digital pattern creation and machine execution, a step toward fully automating not only the act of crocheting but the entire design-to-production workflow.

While the prototype remains in an early stage, the successful demonstration represents a critical proof of concept. It shows that crochet automation is not only theoretically possible, but practically achievable with accessible materials and basic control systems.

This proof of concept is a starting point for future crochet machines and hopefully, this research will be continued so that we can try to help reduce forced human labor and reduce global carbon emissions. In the future, I aim to fully integrate the pattern generation code with the mechanical system, allowing the machine to autonomously create more complex textiles with customized shapes, stitch patterns, and even multi-colored designs.

References

[1]
Organisation for Economic Co-operation and Development (OECD). (2015). Child labour. OECD iLibrary. https://www.oecd-ilibrary.org/social-issues-migration-health/child-labour_f6883e26-en

[2]
Kara, S. (2019). Tainted garments: The exploitation of women and girls in India's home-based garment sector. Blum Center for Developing Economies, University of California, Berkeley. https://media.business-humanrights.org/media/documents/files/documents/tainted-garments.pdf

[3]
Politecnico di Milano. (2023). Sustainable textile processing: An in-depth analysis of environmental and workers well-being in the fashion industry [Master's thesis]. POLITesi. https://www.politesi.polimi.it/handle/10589/222538

[4]
Daily Sabah. (2022, May 20). A beginner's guide to crocheting through the eyes of someone who used to hate it. https://www.dailysabah.com/arts/a-beginners-guide-to-crocheting-through-the-eyes-of-someone-who-used-to-hate-it/news

[5]
Knitted Clouds. (n.d.). Understanding yarn weights. https://www.knittedclouds.com/blogs/news/yarn-weights

[6]
Crochet.com. (n.d.). Understanding crochet hook sizes. https://www.crochet.com/learning-center/understanding-crochet-hook-sizes?srsltid=AfmBOoqzjQWJDgNrJWLl-Alc-HgypXjcOIC1xjsUROGNcysze1XFXBbN

[7]
Asmi Handmade. (n.d.). Inline hooks vs. tapered hooks. https://asmihandmade.com/inline-hooks-vs-tapered-hooks/

[8]
Aabhar Creations. (n.d.). Crochet stitch heights. https://aabharcreations.com/crochet-stitch-heights/

[9]
Easy Crochet. (n.d.). UK to US crochet terms. https://easycrochet.com/uk-to-us-crochet-terms/

[10]
Amigurumi.com. (n.d.). Single crochet. https://www.amigurumi.com/stitches/single-crochet/

[11]
Crochet and Stitches. (n.d.). How to crochet a spiral. https://www.crochetandstitches.com/crochet-a-spiral/

[12]
Storck, J. L. (2024). Automation of crochet technology and development of a prototype machine for the production of complex-shaped textiles [Doctoral dissertation, Technische Universität Dresden]. Hochschule Bielefeld Publikationsserver. https://www.hsbi.de/publikationsserver/record/4792

[13]
Wang, L., Janssen, P., & Stouffs, R. (2023). Teaching computational design optimization: An experimental course for performance-based building massing exploration. In W. Dokonal, U. Hirschberg, & G. Wurzer (Eds.), Digital design reconsidered: Proceedings of the 41st Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe 2023) (Vol. 1, pp. 179–188). eCAADe. https://doi.org/10.52842/conf.ecaade.2023.1.179

[14]
Grasser-Parger, A. (2023). Decentralized participation and agency in digital art and architecture: An exploration of pixel and voxel-based case studies. In W. Dokonal, U. Hirschberg, & G. Wurzer (Eds.), Digital design reconsidered: Proceedings of the 41st Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe 2023) (Vol. 1, pp. 691–700). eCAADe. https://doi.org/10.52842/conf.ecaade.2023.1.691

[15]
Fiber Spider. (2023, August 17). Crochet sphere [Video]. YouTube. https://www.youtube.com/watch?v=g600r_TW5fI

[16]
SparkFun Electronics. (n.d.). SparkFun RedBoard Qwiic. https://www.sparkfun.com/sparkfun-redboard-qwiic.html

[17]
SparkFun Electronics. (n.d.). SparkFun Micro 6DoF IMU Breakout - BMI270 (Qwiic). https://www.sparkfun.com/sparkfun-micro-6dof-imu-breakout-bmi270-qwiic.html

[18]
Logitech. (n.d.). Logitech webcam for meetings and streaming. https://www.amazon.com/Logitech-Webcam-Meetings-Streaming-Built/dp/B0BXGFFSL1

[19]
The Pixel Farm. (n.d.). PFTrack software. https://www.thepixelfarm.co.uk/pftrack

[20]
McMaster-Carr. (n.d.). T-slotted framing rails. https://www.mcmaster.com/products/~/t-slotted-framing-rail-profile~single/t-slotted-framing-and-fittings~/t-slotted-framing-rails-4/?s=8020

[21]
Pololu Robotics and Electronics. (n.d.). Pololu Micro Metal Gearmotor Bracket Pair. https://www.pololu.com/product/2820

[22]
Digi-Key Electronics. (n.d.). SparkFun Electronics ROB-16413. https://www.digikey.com/en/products/detail/sparkfun-electronics/ROB-16413/12178435

[23]

SparkFun Electronics. (n.d.). Hobby Gearmotor - 140 RPM (Pair).
https://www.sparkfun.com/hobby-gearmotor-140-rpm-pair.html

[24]
SparkFun Electronics. (n.d.). SparkFun Motor Driver - Dual TB6612FNG (with headers).
https://www.sparkfun.com/sparkfun-motor-driver-dual-tb6612fng-with-headers.html

[25]
Arduino. (n.d.). Arduino Uno Rev3.
https://store.arduino.cc/products/arduino-uno-rev3?srsltid=AfmBOooxayq0f1kT246PCEUMMWns3LPXOKHTOYZYoEcwqsbSuyHdEu5l

Appendix

Stitch Dimension Data Collection

Stitch Dimension data:

I wanted to find out what my tension factor was and so I crocheted with yarn weights 0-6 and hook sizes 2 mm, 3mm, 4mm, 5mm, 6mm, 7mm, 8mm, 9mm, and 10mm. For each combination of hook size and yarn weight I crocheted:

1. 10 chain stitches
2. 10 single crochets
3. 10 half double crochets
4. 10 double crochets
5. 10 triple crochets

I recorded the height, width, and length of each chain stitch, and I recorded the height of the single crochets, half double crochets, half double crochets, and triple crochets.

Table 2: Width and length of chain stitches in mm

hook size	weight	weight 0	weight 1	weight 2	weight 3	weight 4	weight 5	weight 6
2mm	length (mm)	45	56	45	48	47	55	64
	width (mm)	5	5	5	5	6	6	8
3mm	length (mm)	55	50	57	71	63	71	73
	width (mm)	4	3	5	6	5	7	9
4mm	length (mm)	58	55	53	76	60	67	75
	width (mm)	4	4	5	5	8	7	10

5mm	length (mm)	69	62	76	76	61	85	86
	width (mm)	6	4	7	8	6	9	11
6mm	length (mm)	80	62	71	80	77	83	84
	width (mm)	5	4	6	8	10	8	9
7mm	length (mm)	85	75	70	92	82	86	83
	width (mm)	7	5	6	8	8	9	11
8mm	length (mm)	97	94	83	84	82	94	87
	width (mm)	5	6	8	8	8	9	12
9mm	length (mm)	101	81	74	79	82	91	99
	width (mm)	6	5	5	9	7	9	11
10mm	length (mm)	91	90	72	85	92	87	97
	width (mm)	6	5	9	9	10	10	11

Table 3: Heights of sc. hdc, dc, tr, in mm

		height (mm)						
hook size	type of stitch	weight 0	weight 1	weight 2	weight 3	weight 4	weight 5	weight 6
2mm	sc	7	5	6	8	8	12	13

	hdc	9	7	7	11	10	13	14
	dc	11	8	10	14	13	17	19
	tr	14	13	12	18	17	21	25
3mm	sc	6	5	7	8	9	12	14
	hdc	9	8	9	10	10	14	18
	dc	12	10	10	12	17	17	23
	tr	18	16	15	20	20	23	28
4mm	sc	7	6	8	8	10	13	16
	hdc	9	9	10	13	11	15	19
	dc	12	12	13	13	15	19	22
	tr	17	17	19	20	23	27	29
5mm	sc	7	6	7	10	10	14	17
	hdc	10	10	9	12	13	15	19
	dc	16	10	15	15	15	20	23
	tr	23	17	21	21	23	26	32
6mm	sc	10	7	7	12	10	14	18
	hdc	10	14	9	15	12	17	20
	dc	19	11	12	17	15	22	25
	tr	26	18	22	23	26	26	33
7mm	sc	7	8	7	11	11	15	17
	hdc	12	14	15	13	15	17	21
	dc	22	14	18	18	18	21	26
	tr	29	21	19	22	28	28	36
8mm	sc	11	11	8	11	14	15	19
	hdc	16	11	11	17	15	17	22
	dc	17	18	14	18	20	22	24
	tr	28	20	20	24	28	30	37
9mm	sc	11	5	7	10	13	15	19

	hdc	11	6	8	13	17	19	21
	dc	23	12	12	14	19	22	27
	tr	31	12	17	25	29	31	38
10mm	sc	9	6	10	12	11	15	18
	hdc	11	13	15	15	15	18	22
	dc	22	17	17	18	18	25	26
	tr	37	26	20	27	29	30	40

2. Granny Square Measurements

I also crocheted a granny square with hook sizes 2-10mm and weights 0-6 yarn. The pattern I used to crochet each square is displayed in Figure 32.

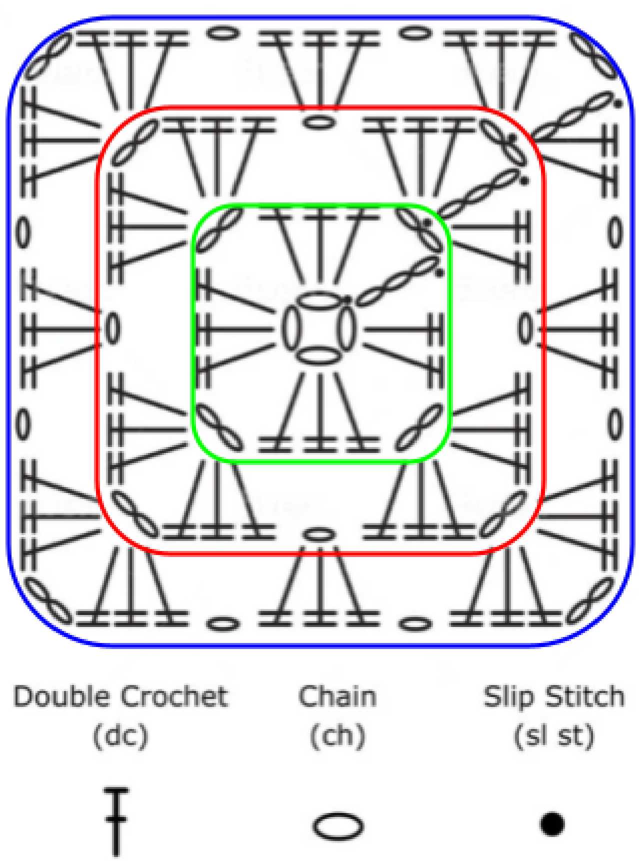

Fig 32: Granny square diagram pattern - row 1 is green, row 2 is red, row 3 is blue

I measured the length of each row while crocheting to see the average height increase.

Table 4: Measurements of side length of each granny square pre blocking

	weight	weight 0	weight 1	weight 2	weight 3	weight 4	weight 5	weight 6
2mm	length of one side (at the end of r2)	21	24	22	28	29	37	40
	length of one side (at the end of r3)	36	38	34	47	47	63	65
	length of one side (at the end of r4)	54	50	50	65	62	77	88
3mm	length of one side (at the end of r2)	25	27	24	32	32	39	42
	length of one side (at the end of r3)	45	42	39	51	52	60	68
	length of one side (at the end of r4)	62	58	54	77	70	83	94
4mm	length of one side (at the end of r2)	33	30	28	34	35	40	44
	length of one side (at the end of r3)	51	48	44	55	55	67	74
	length of one side (at the end of r4)	73	65	56	80	84	92	100
5mm	length of one side (at the end of r2)	35	33	28	38	37	42	46
	length of one side (at the end of r3)	56	53	46	58	60	70	77
	length of one side (at the end of r4)	85	72	63	86	90	97	105
6mm	length of one side (at the end of r2)	37	35	33	43	41	43	51
	length of one side (at the end of r3)	60	60	49	60	67	77	84

	length of one side (at the end of r4)	87	81		88	98	121	120
7mm	length of one side (at the end of r2)	40	40	36	48	44	46	60
	length of one side (at the end of r3)	66	65	58	71	73	79	88
	length of one side (at the end of r4)	95	86	82	92	100	115	125
8mm	length of one side (at the end of r2)	45	42	44	50	47	49	60
	length of one side (at the end of r3)	70	67	60	77	75	82	100
	length of one side (at the end of r4)	100	91	83	109	103	122	140
9mm	length of one side (at the end of r2)	48	48	45	48	44	55	63
	length of one side (at the end of r3)	77	71	68	85	84	84	100
	length of one side (at the end of r4)	106	100	88	110	116	112	140
10 mm	length of one side (at the end of r2)	55	50	40	50	50	56	65
	length of one side (at the end of r3)	80	81	61	90	78	93	110
	length of one side (at the end of r4)	110	104	91	120	110	121	150

After I finished crocheting all 72 squares, I blocked all of the squares and then retook the side length measurements.

Table 5: Measurements of side length of each granny square post blocking

	weight	weight 0	weight 1	weight 2	weight 3	weight 4	weight 5	weight 6
2mm	length of one side (at the end of r4)	60	60	51	67	64	85	91
3mm	length of one side (at the end of r4)	70	60	58	71	71	91	95
4mm	length of one side (at the end of r4)	82	65	65	82	85	100	109
5mm	length of one side (at the end of r4)	97	83	70	96	86	105	113
6mm	length of one side (at the end of r4)	97	88	80	100	94	112	122
7mm	length of one side (at the end of r4)	98	96	85	105	105	120	132
8mm	length of one side (at the end of r4)	108	97	90	109	109	126	141
9mm	length of one side (at the end of r4)	110	103	94	113	120	130	145
10 mm	length of one side (at the end of r4)	113	116	99	118	116	132	145

Code

Matlab Crochet Sphere Code

```
%crochet single crochet sphere math
%parameters: a = one stitch
max_radius = 5; %(3*a) adjust this to control size of sphere
const_rows = 5; %(a+1) adjust this to make an ellipsoid
total_rows = 2* max_radius + const_rows; %(3*a)
rows = 1:total_rows;
```

```
stitches = zeros(1, total_rows);
for a = 1:total_rows
  dec_rows = a-1;
  inc_rows = a;
  constant_rows = a+1;
  if a<= max_radius
     stitches(a) = 6*a;
  elseif a <=max_radius + const_rows
     stitches(a) = 6*max_radius;
  else
     dec_row = a - max_radius - const_rows;
     stitches(a) = 6*(max_radius-dec_row);
  end
end
%2d plot
figure;
hold on;
inc = 1:max_radius;
const = (max_radius+1):(max_radius+const_rows);
dec = (max_radius+const_rows+1):total_rows;
plot(rows(inc), stitches(inc), 'g-o','LineWidth',2,'DisplayName','increasing');
if ~isempty(const)
  plot(rows(const), stitches(const), 'b-o','LineWidth',2,'DisplayName','constant');
end
plot(rows(dec), stitches(dec), 'r-o','LineWidth',2,'DisplayName','decreasing');
xlabel('Row/Round Number');
ylabel('Stitches per Row');
title('stitch counts per round in crochet sphere');
legend('Location','best');
grid on;
xlim([1 total_rows]);
ylim([0 max(stitches)+5])
hold off;
%3d plot
theta = linspace(0,2*pi, max(stitches)); %around the circle
r = stitches/(2*pi); %approximate radius per row
z = linspace(-max_radius-const_rows/2, max_radius+const_rows/2, total_rows); %height coordinate
```

```
row_colors = zeros(total_rows, 3); %RGB values
for i = 1:total_rows
  if i<= max_radius
      row_colors(i,:) = [0,1,0];
  elseif i<=max_radius + const_rows
      row_colors(i,:) = [0, 0, 1];
  else
      row_colors(i,:) = [1, 0, 0];
  end
end
figure;
xlabel('x');
ylabel('y');
zlabel('height (rounds)');
title('Crochet Sphere - Animated Build');
axis equal;
grid on;
view(3);
hold on;
xlim([-max(r) max(r)])
ylim([-max(r) max(r)])
zlim([-max(z) max(z)])
for i = 1:total_rows
  x_circ = r(i)*cos(theta);
  y_circ = r(i)*sin(theta);
  z_row = z(i)*ones(size(theta));
  fill3(x_circ, y_circ, z_row, row_colors(i,:) ...
      ,'EdgeColor','none','FaceAlpha', 1);
  num_stitches = stitches(i);
  for s = 1:num_stitches
      angle = 2*pi*(s-1)/num_stitches;
      x_dot = r(i)*cos(angle);
      y_dot = r(i)*sin(angle);
      z_dot = z(i);
      plot3(x_dot, y_dot, z_dot, 'k.', 'MarkerSize',6); %
  end
  pause(1); %increase = slower, decrease = faster
```

```
end
lighting gouraud;
camlight headlight;
```

Crochet Gui

```
function crochet_gui()
  % yarn and hook data
  yarn_weights = containers.Map( ...
      {'lace (0)', 'super fine (1)', 'fine (2)', 'light (3)', 'medium (4)', ...
       'bulky (5)', 'super bulky (6)', 'jumbo (7)'}, ...
      [0.5, 0.75, 1.0, 1.5, 2.0, 2.5, 3.0, 4.0]);
  hook_sizes = containers.Map( ...
      {'B-1 (2.25 mm)', 'C-2 (2.75 mm)', 'D-3 (3.25 mm)', ...
       'E-4 (3.5 mm)', 'F-5 (3.75 mm)', 'G-6 (4.0 mm)', ...
       'H-8 (5.0 mm)', 'I-9 (5.5 mm)', 'J-10 (6.0 mm)', ...
       'K-10.5 (6.5 mm)', 'L-11 (8.0 mm)', 'M/N-13 (9.0 mm)', ...
       'N/P-15 (10.0 mm)', 'P/Q (15.0 mm)'}, ...
      [2.25, 2.75, 3.25, 3.5, 3.75, 4.0, 5.0, 5.5, 6.0, ...
       6.5, 8.0, 9.0, 10.0, 15.0]);
  yarn_keys = keys(yarn_weights);
  hook_keys = keys(hook_sizes);
  % create figure
  f = figure('Name', 'Crochet Chain Stitch Visualizer', ...
           'Position', [100, 100, 500, 440]);
  % yarn weight dropdown
  uicontrol('Style', 'text', 'Position', [50 390 120 20], ...
          'String', 'Select Yarn Weight:');
  yarn_dropdown = uicontrol('Style', 'popupmenu', ...
                         'Position', [50 365 150 25], ...
                         'String', yarn_keys);
  % hook size dropdown
  uicontrol('Style', 'text', 'Position', [280 390 120 20], ...
          'String', 'Select Hook Size:');
  hook_dropdown = uicontrol('Style', 'popupmenu', ...
                         'Position', [280 365 150 25], ...
                         'String', hook_keys);
  % input mode dropdown
  uicontrol('Style', 'text', 'Position', [175 325 150 20], ...
          'String', 'Input Mode:');
  mode_dropdown = uicontrol('Style', 'popupmenu', ...
                         'Position', [175 300 150 25], ...
                         'String', {'Specify Number of Loops', 'Specify Length'});
  % number of loops input
  loop_label = uicontrol('Style', 'text', ...
                       'Position', [175 265 150 20], ...
                       'String', 'Number of Loops:');
```

```
loop_box = uicontrol('Style', 'edit', ...
                     'Position', [200 240 100 25], ...
                     'String', '12');
% chain length input (hidden by default)
length_label = uicontrol('Style', 'text', ...
                         'Position', [175 265 150 20], ...
                         'String', 'Chain Length (mm or in):', ...
                         'Visible', 'off');
length_box = uicontrol('Style', 'edit', ...
                       'Position', [200 240 100 25], ...
                       'String', '', ...
                       'Visible', 'off');
unit_dropdown = uicontrol('Style', 'popupmenu', ...
                          'Position', [310 240 60 25], ...
                          'String', {'mm', 'in'}, ...
                          'Visible', 'off');
% tension input
uicontrol('Style', 'text', 'Position', [175 205 150 20], ...
          'String', 'Tension Scale (0.5–1.5):');
tension_box = uicontrol('Style', 'edit', ...
                        'Position', [200 180 100 25], ...
                        'String', '1.0');
% plot button
uicontrol('Style', 'pushbutton', ...
          'String', 'Plot Chain Stitch', ...
          'Position', [175 125 150 40], ...
          'Callback', @plot_callback);
% output label for recommendation
recommendation_label = uicontrol('Style', 'text', ...
    'Position', [150 80 220 20], ...
    'String', '', ...
    'Visible', 'off', ...
    'FontWeight', 'bold');
% update input fields when mode changes
mode_dropdown.Callback = @(~, ~) update_mode();
function update_mode()
    mode = mode_dropdown.Value;
    if mode == 1  % number of loops
        loop_label.Visible = 'on';
        loop_box.Visible = 'on';
        length_label.Visible = 'off';
        length_box.Visible = 'off';
        unit_dropdown.Visible = 'off';
        recommendation_label.Visible = 'off';
    else  % length
        loop_label.Visible = 'off';
        loop_box.Visible = 'off';
        length_label.Visible = 'on';
        length_box.Visible = 'on';
```

```
        unit_dropdown.Visible = 'on';
    end
  end
  function plot_callback(~, ~)
    yarn_idx = yarn_dropdown.Value;
    hook_idx = hook_dropdown.Value;
    yarn_d = yarn_weights(yarn_keys{yarn_idx});
    hook_d = hook_sizes(hook_keys{hook_idx});
    tension = str2double(tension_box.String);
    if isnan(tension) || tension < 0.5 || tension > 1.5
        errordlg('Tension must be between 0.5 and 1.5');
        return;
    end
    len = hook_d + yarn_d;
    overlap_ratio = 0.9;
    spacing = len * overlap_ratio * tension;
    mode = mode_dropdown.Value;
    if mode == 1  % specify number of loops
        num_loops = str2double(loop_box.String);
        if isnan(num_loops) || num_loops <= 0
            errordlg('Please enter a valid number of loops.');
            return;
        end
        recommendation_label.Visible = 'off';
    else  % specify desired length
        chain_length = str2double(length_box.String);
        if isnan(chain_length) || chain_length <= 0
            errordlg('Please enter a valid chain length.');
            return;
        end
        if unit_dropdown.Value == 2  % in → mm
            chain_length = chain_length * 25.4;
        end
        num_loops = max(1, round((chain_length - len) / spacing + 1));
        recommendation_label.Visible = 'on';
        recommendation_label.String = sprintf('Recommended: %d chain stitches', num_loops);
    end
    plot_chain_stitch_matlab(yarn_d, hook_d, num_loops, tension);
  end
end
```

Plot Chain Stitch

```
function plot_chain_stitch_matlab(yarn_d, hook_d, num_stitches, tension)
  % ---- CONFIG ----
  curve_radius_scale = 1.5;   % roundness of each stitch
  overlap_ratio = 0.9;        % 1.0 = no overlap, <1 = overlap
```

```
    % ----------------
    % stitch dimensions
    len = tension*(hook_d + yarn_d);
    ht = tension*(2 * yarn_d + hook_d);
    r = (len / 2) * curve_radius_scale;
    spacing = len * overlap_ratio;
    % build one sideways teardrop shape
    theta = linspace(pi/2, 3*pi/2, 100);
    x_back = -r + r * cos(theta);
    y_back = (ht/2) * sin(theta);
    x_point = [r, -r];
    y_point = [0, y_back(1)];
    x_half = [x_back, x_point];
    y_half = [y_back, y_point];
    x_teardrop = [x_half, fliplr(x_half)];
    y_teardrop = [y_half, -fliplr(y_half)];
    % plot setup
    figure;
    hold on;
    axis equal;
    axis off;
    for i = 0:(num_stitches - 1)
        x_shift = i * spacing;
        % transparent fill with no interior line
        patch(x_teardrop + x_shift, y_teardrop, [0.8 0.5 0.2], ...
              'EdgeColor', 'none', 'FaceAlpha', 0.2);
    end
    % dimension guide
    chain_length = spacing * (num_stitches - 1) + len;
    y_offset = -1.2 * ht;
    plot([0, chain_length], [y_offset, y_offset], 'k--');
    text(chain_length / 2, y_offset - 0.3 * ht, ...
         sprintf('Chain Length: %.1f mm (%.2f in)', chain_length, chain_length / 25.4), ...
         'HorizontalAlignment', 'center');
    % height dimension on the far left
    x_offset = -1.5 * len;
    y_top = ht / 2;
    y_bot = -ht / 2;
    plot([x_offset, x_offset], [y_bot, y_top], 'k--');
    plot([x_offset - 0.1*len, x_offset + 0.1*len], [y_top, y_top], 'k');
    plot([x_offset - 0.1*len, x_offset + 0.1*len], [y_bot, y_bot], 'k');
    text(x_offset - 0.2*len, 0, ...
        sprintf('Height: %.1f mm\n(%.2f in)', ht, ht / 25.4), ...
        'HorizontalAlignment', 'right', ...
        'VerticalAlignment', 'middle', ...
        'FontSize', 9);
    title(sprintf('Crochet Chain — %d Stitches\nHook: %.1f mm, Yarn: %.1f mm, Tension: %.2f', ...
        num_stitches, hook_d, yarn_d, tension));
end
```