Division of Biomedical Engineering
Department of Human Biology
University of Cape Town

Development of a Solution Method to Promote Proper Bandaging Techniques for Transfemoral Amputees

Minor Dissertation
In partial fulfilment of the requirements for the degree:
MSc. in Biomedical Engineering by Coursework and Dissertation

Author: Bhaveen Lalla (LLLBHA004)

Supervisor: A/Prof. Sudesh Sivarasu

Co-supervisors: Dr Roopam Dey

October 29, 2021

Declaration

I, Bhaveen Lalla, hereby declare that the work on which this thesis is based is my original work (except where acknowledgements indicate otherwise) and that neither the whole work nor any part of it has been, is being, or is to be submitted for another degree in this or any other university.

I authorise the University to reproduce for the purpose of research either the whole or any portion of the contents in any manner whatsoever.

Signature: Signed by candidate

Date: October 29, 2021

Abstract

Post-operative care for lower limb amputees focuses on reducing swelling and promoting healing of the residual limb. Healing and desired residual limb maturation is necessary to achieve a conical shape suitable for prosthesis integration. Elastic bandaging is applied to the residual limb to achieve this conical shape. Bandaging requires skill and frequent reapplication. In many cases, the amputee is unable to bandage the residual limb effectively following surgery and professional assistance may not be readily available, increasing the risk of permanent damage due to improper bandaging. Transfemoral amputees are often confined to a wheelchair and spend the most time without a prosthesis fitted compared to any other lower limb amputee. An effective method in achieving the optimum residual limb shape, while promoting proper bandaging is therefore required.

The aim of this study was to develop a solution method to assist transfemoral amputees with proper bandaging to achieve a shorter healing period and promote faster prosthesis integration through residual limb re-shaping. An overall solution method comprising of three sub-systems was designed to instruct bandaging. This included the development of a bandaging template, a mobile application and an elastic bandage dispenser.

A printable bandaging template generated by a Python script based on the measurements of the patient's residual limb circumferences was developed to instruct bandaging. Development of a mobile application allowed for the design of an interface to control the dispensing device and instruct bandaging steps. The developed Bandage Utility Dispenser (BUD) transmits unrolled bandage length measurements to the mobile application using Bluetooth, to meet the target bandage lengths. Target bandage lengths are calculated by a second Python script to recommend sufficient bandage lengths. Target bandage lengths are expected to apply adequate pressure, while covering the exposed area of the residual limb for the current bandaging step.

Testing occurred on an anatomically correct residual limb model (ACM) and five constructed models (CM) assembled by altering proximal and distal ACM circumferential measurements in increments of 5 cm. Measurement and pressure testing were performed after successfully validating the use of the CMs using a Bland-Altman analysis on the ACM and a CM of equal dimensions. Measurement testing utilised a chi-square goodness of fit test to compare observed and expected bandage length measurements for each model. Results for all models indicated that observed and expected measurements did not differ significantly. Pressure testing was performed by measuring the pressure application along the perimeter of the models with assembled pressure pads. Readings were analysed using a one sample t-test to compare differences between sample means and the recommended pressure application from literature. Testing results indicated that pressure application for five of the six models were in an acceptable range. Future recommendations have been established to potentially improve design, functionality and testing of the current iteration of the BUD solution method.

Acknowledgments

I would like to express my utmost gratitude to the following people:

My parents, Mr Deenesh Lalla and Mrs Dharmista Lalla for their love, support and motivation. I am truly grateful for this opportunity that you have given me and words cannot express how thankful I am.

My supervisor, Associate Professor Sudesh Sivarasu and co-supervisor, Dr Roopam Dey, for their assistance, mentorship and guidance in the completion of this study and throughout this degree.

My sister, Mishka Lalla and brother-in-law, Bhavesh Bhikha for their continued support and motivation to pursue this degree.

My friends, Depesh Cara and Harshil Govan for their friendship, motivation and support throughout my academic journey.

My friends and colleagues within the department of Biomedical Engineering for their their friendship, guidance and assistance.

I would like to sincerely thank Mr Michael Awood, the clinical partner associated with this study, for his guidance and assistance in the completion of this study.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Background and Problem Identification

Bandaging an amputee's residual limb requires skill and frequent reapplication. In many cases, the amputee is unable to bandage the residual limb effectively post-surgery, and professional assistance may not be readily available to perform bandaging as often as required. This leads to improper bandaging that may result in permanent damage (Lorraine, Cake, & Badger, 1957). Proper bandaging is therefore required to shape and shrink the residual limb for successful prosthetic fitting. An effective method to apply consistent pressure to the desired areas of the residual limb to control swelling is therefore needed. Transfemoral (above knee) amputees find it more difficult to adjust to life post-amputation than through knee or transtibial amputees (below knee) amputees, due to the higher level of amputation (Hagberg, Berlin, & Renstrom, 1992). An efficient way to bandage the residual limb will influence residual limb volume and achieve the desired conical shape. This will speed up the healing process and ensure that a prosthesis can be fitted sooner. As a result, this will enable the amputee to start their rehabilitation process earlier, as well as adjust to their disability sooner.

This study involves the research and development of a solution method to assist transfemoral amputees with the bandaging process during the rehabilitation period. The first component is a patient specific bandaging template generated by a custom script that is applied to the residual limb to coordinate bandaging. The second component of the solution method in the form of a mobile application, coordinates and instruct amputees on proper bandaging techniques to apply sufficient pressure to desired locations on the residual limb for the purpose of residual limb re-shaping. The third component is an elastic crepe bandage dispenser that measures the length of bandage unrolled, while transmitting the unrolled length to the mobile application for display and instructive purposes. Required bandage lengths for each step of the bandaging process is predicted by a second custom script to coordinate bandaging with the bandaging template, mobile application and dispenser. The proposed solution method aims to enable a transfemoral amputee to become more independent during the rehabilitation period through assisted bandaging by providing consistent bandaging guidance.

1.2 Hypothesis

Through the use of a guided bandaging technique and the application of calculated predicted bandage lengths required to complete the bandaging technique, an ideal pressure of 20 mmHg can be uniformly applied to a residual limb. Continuous use by a transfemoral amputee of a guided bandaging technique that applies consistent pressure may promote independency, result in a shorter healing period, prevent resultant injuries and abnormalities from improper bandaging and enable prosthesis integration to occur sooner as a result.

1.3 Aim

The aim of this study was to develop a solution method to assist transfemoral amputees with performing proper bandaging to achieve a shorter healing period and promote faster prosthesis integration through residual limb re-shaping.

1.4 Research Objectives

Objectives that were met for this study involved the formulation and evaluation of a solution method to enable efficient residual limb bandaging for transfemoral amputees and promote independency as a result. Testing and verification for each sub-system of the solution method was done to evaluate performance and measurement accuracy. These sub-systems included a patient specific template to instruct bandaging, a mobile application that instructs bandaging and a bandaging dispenser that measures the length of unrolled elastic bandage. Objectives met included:

1. Design of a bandaging template that is patient specific and can be adjusted according to changes in residual limb volume.
2. Development of a mobile application to convey bandaging guidelines and display dispenser measurements.
3. Design and development of a bandaging dispenser to measure unrolled bandage lengths and transmit measurements to the mobile application.
4. Determination of ideal pressure application of a common crepe elastic bandage used to conically shape the residual limb.

5. Perform testing to assess solution method sub-systems by evaluating:
    - Template generation and application.
    - Mobile application installation and functionality.
    - Dispenser measurement accuracy and communication with the mobile application.
    - Predicted bandage length accuracy and pressure application testing.

1.5 Study Parameters

Difficulty with proper residual limb bandaging leads to injury and abnormal maturation. Long periods between clinical visits and paucity of contact with clinicians may influence improper bandaging techniques. Alternate forms of residual limb shape control that require no supervision are often expensive and difficult to obtain. Highlighted problem areas have influenced design parameters of each solution method sub-system to develop a solution that instructs proper bandaging, is easily accessible and is cost effective. Design parameters include the design of a bandaging template that can clearly instruct bandaging through easily interpretable guidelines. The template medium should be easily accessible and cost effective. The second design parameter involves the development of a mobile application that facilitates installation on lower-end smartphone devices. The final design parameter includes the development of a bandaging dispenser with dimensions large enough to incorporate elastic crepe bandage rolls, but small enough to prevent difficulty when bandaging the residual limb. Parameters to be validated before final evaluation of the overall solution method include: the evaluation of pressure applied by calculated predicted bandage lengths, measurement accuracy of unrolled bandage lengths from the dispenser and successful installation of the mobile application.

1.6 Scope of the Study

The study focuses on and is limited to the design and development of a solution method to instruct transfemoral amputees on proper bandaging. Transfemoral amputees are the primary focus of research, due to problems specific to the level of lower limb amputation. The scope of the study is to design and experimentally validate the device's ability to conduct proper bandaging necessary to achieve the desired residual limb shape suitable for prosthesis integration. Functionality of the device is tested by applying the solution

method to six residual limb models, and determining pressure application, predicted bandage length calculations and mobile application efficacy.

1.7 Dissertation Overview

Dissertation chapters detail design and testing methodologies and outcomes as illustrated in Figure 1. Chapter 2 provides information on relevant literature on the anatomy of the transfemoral residual limb, post-operative care, ideal pressure application and existing residual limb re-shaping techniques. Chapter 3 outlines the design methodology for the overall solution method. Each sub-system component is discussed to describe assembled functionality. Chapter 4 details the outcome for each of the three solution method sub-systems. Design, assembly and coded scripts are discussed. Chapter 5 outlines the testing methodology that was used to determine bandage length measurements and pressure application accuracy using a statistical analysis for each test. Chapter 6 highlights the results of each test and discusses observations made when analysing results. Final observations on similar trends in results are discussed, with a discussion on comparative studies from literature concluding the chapter. Chapter 7 concludes the document by discussing final observations and recommending future considerations to potentially improve design, functionality and testing of the current iteration of the solution method.

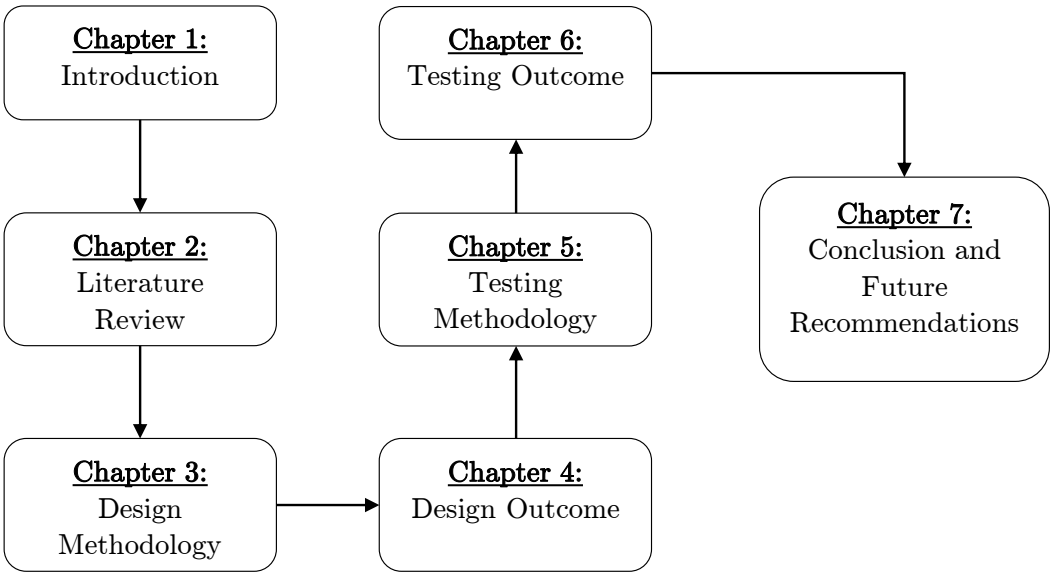


Figure 1: Dissertation overview summarised

2 Literature Review

Research of relevant literature regarding primary causes of lower limb amputations, surgical preparation done to ensure successful prosthesis integration post-amputation, as well as transfemoral residual limb anatomy post-amputation was conducted. Research for this study concentrated on the rehabilitation of the transfemoral residual limb, therefore challenges specific to transfemoral amputees are discussed to highlight the purpose and scope of this study. Diseased states and abnormalities of the transfemoral residual limb are discussed to emphasise causes due to improper bandaging. An overview on the purpose of bandaging to achieve a healthy state for the amputee's residual limb suitable for prosthesis integration follows. Research into bandage pressure application was conducted to determine the ideal pressure to be exerted onto the residual limb while bandaging. The chapter concludes with research on existing solutions that influence residual limb shape, researched to compare with the proposed solution method.

2.1 Causes of Amputation

Available statistics limited to the U.S. shows that currently, there are 185 000 amputations performed each year in the U.S., with the number of total amputees rising to 3.6 million by 2050. Of these amputations, approximately 20% of them are transfemoral amputations with 3 500 transfemoral amputations performed each week (Ziegler-Graham, MacKenzie, Ephraim, Travison, & Brookmeyer, 2008). Lower limb amputations are estimated to cost between $30 000 and $60 000 for initial hospital costs, followed by between $43 000 and $60 000 in post-operative care (Margolis et al., 2013). In total, hospital costs associated with amputations cost more than $8.3 billion annually (Ziegler-Graham et al., 2008).

A total of 82% of transfemoral amputations are caused by peripheral artery disease and diabetes, with the remaining 18% caused by trauma, cancer, infection, and congenital defects (Dillingham & Pezzin, 2005). Diabetes is recognised as one of the most common non-traumatic causes of lower limb amputations (Lazzarini, Clark, & Derhy, 2011). The total number of global diabetes cases is expected to rise from 424.9 million in 2017, to 628.6 million by 2045 (Standl, 2019). Data from the International Diabetes Federation has shown that 3.85 million South Africans between the ages of 21 and 79 have diabetes, which is approximately 7% of the country's population (Atlas, 2015). In South Africa, it is estimated that 6 lower limb amputations occur daily with 2 500 lower

amputations occurring yearly in KwaZulu-Natal alone, with the primary cause being diabetes (Potgieter, 2018). The International Diabetes Federation provided an estimation claiming that there were 22 million diabetes patients living within Africa in 2014, but this excludes 62% of patients that are undiagnosed (Potgieter, 2018). With an increase in the number of lower limb amputations being performed each year and an increase in the leading cause of lower limb amputations, effective post-operative care is needed to assist amputees with adjusting to their disability and improving the chance for prosthesis integration.

2.2 Challenges Specific to Transfemoral Amputees

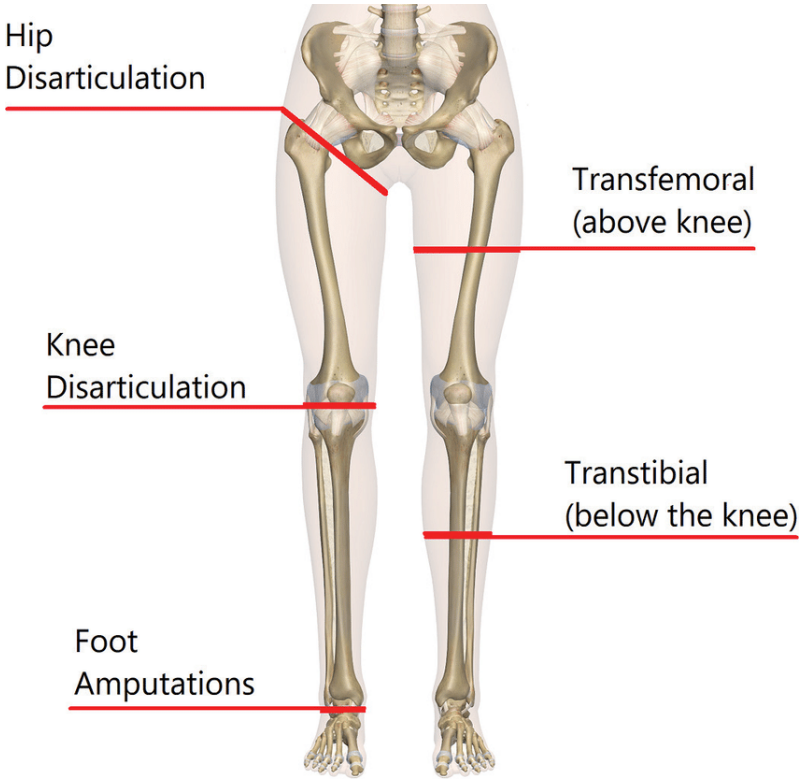


Figure 2: Levels of lower limb amputations (Hussain et al., 2019, p. 3)

Transfemoral amputees encounter specific challenges due to the high level of amputation of their residual limb. Figure 2 shows the classification of the levels of lower limb amputations ranging from the hip to the foot. Higher levels of amputation require higher levels of energy expenditure while using a prosthesis. Transfemoral amputees utilise twice as much energy for mobility than any other lower limb amputee (Albino, Seidel, Brown, Crone, & Attinger, 2014). This not only effects their rate of mobility, but also the frequency of prosthesis use.

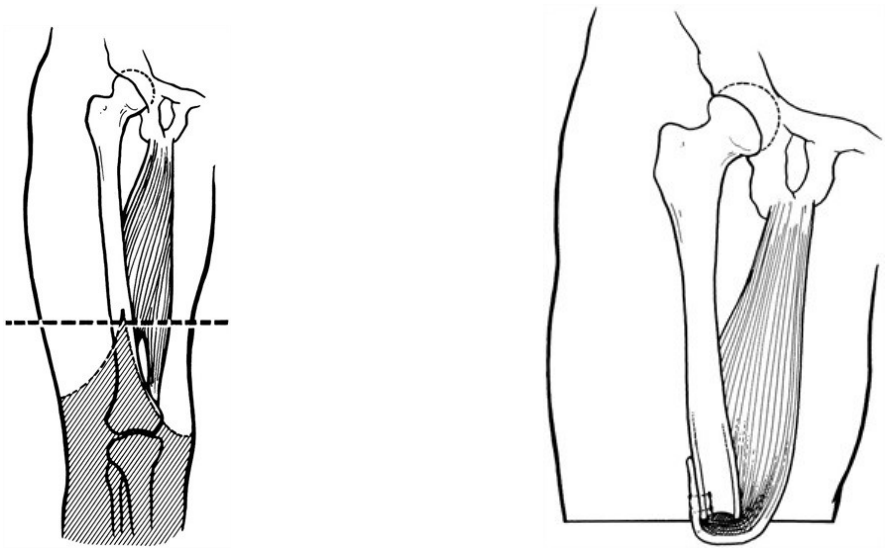
Transfemoral amputees spend less time per day wearing a prosthesis than any other lower limb amputee (Albino et al., 2014). As a result, constant pressure is not applied to the residual limb to maintain its conical shape. A prosthetic liner and socket would conventionally apply this constant pressure, however transfemoral amputees spend more time without the prosthesis fitted than any other lower limb amputee, with many patients not having the resources to acquire a prosthesis (Albino et al., 2014). Bandages are therefore required to wrap the residual limb to apply constant pressure. Application of an elastic bandage shapes the distal end of the amputated leg, where correct shaping is needed to support soft tissue of the residual limb. This prevents swelling due to fluid build-up (oedema) and shapes the residual limb to fit within a prosthetic socket (Amputation Rehabilitation Program, 2018). Elastic bandaging is effective in the re-shaping process, and is advantageous in its application when compared to other re-shaping methods, since distribution and exertion of pressure can be controlled (Moroz, 2017).

2.3 Anatomy and Diseased State of the Transfemoral Residual Limb

Transfemoral amputations are performed with the intention that a prosthesis will be fitted post-surgery, due to the potential for mobility with the aid of a prosthesis (Browner, 2009, p. 2425). Planning of the amputation accounts for the length of the amputation, as well as adequate positioning for sufficient blood supply. A transfemoral amputation is performed when blood supply is insufficient to heal an existing amputation at a lower level, or when salvaging the knee joint is not possible (Devinuwara, Dworak-Kula, & O'Connor, 2018). Pre-determining the length of the residual limb will aid in ensuring that a broad selection of prosthetic knee components can be used post-amputation. This length is planned in such a way that 12 cm of clearance above the fulcrum of the contralateral lower limb should be left (Devinuwara et al., 2018). Essentially, the femur is divided 12 cm proximal to the femoral condyles (Sidawy & Perler, 2018, p. 1507).

Alternative amputation lengths may be considered, but hinder future mobility and gait (Devinuwara et al., 2018). The correct length and positioning of residual limb length not only affects mobility, but also the energy expenditure the amputee will experience post-amputation (Devinuwara et al., 2018). Figure 3 (a) shows the location of the amputation technique that begins with a fish-mouth incision to create anterior and posterior skin flaps around the femur (Sidawy & Perler, 2018). After skin and soft tissue have been cut, the divided superficial femoral artery and femoral vein must be controlled with suture ligature (Sidawy & Perler, 2018). Muscles in the area will be divided in the same plane,

unless a myodesis (connection of muscle to bone) is performed (Sidawy & Perler, 2018). The myodesis between the adductor magnus and lateral femur is shown in Figure 3 (b).



(a) (b)

Figure 3: (a) Positioning of skin flaps and skin resection, (b) Adductor magnus and lateral femur myodesis (Gottschalk, 1999)

The femur is then transected proximally of the fish-mouth incision corners. Following transection, the sciatic nerve is lengthened, divided and retracted (Sidawy & Perler, 2018). After amputation, the residual limb length should be adequate enough to fit a prosthesis with enough support and leverage. Preservation of additional residual limb length can assist with generating more torque, which will improve prosthetic functionality and will decrease pressure exerted on remaining soft tissue (Devinuwara et al., 2018). Ideal residual limb length is determined prior to amputation, and is aimed at ensuring optimal mobility and functionality between the residual limb and the prosthesis. Maintenance and rehabilitation of the residual limb will ensure that functionality is kept (Gottschalk, 1999). General maintenance will assist with preventing pain during mobility and will aid in preventing further abnormalities, such as pressure sores or abnormal skin growth (muscle or skin bulges). Certain characteristics are used to determine whether residual limb condition is ideal for prosthesis integration. These characteristics range from the shape, appearance, temperature and complexion of the residual limb (Gottschalk, 1999). Ideal residual limb state for prosthesis integration is described as having a rounded and smooth distal end after a sufficient healing period has occurred (Gottschalk, 1999). Scarring should be low enough as to not interfere with prosthesis integration, while excessive pressure should not be applied to the scar present at the amputation site. Skin flaps and remaining muscle should not bulge from prosthesis use or bandaging, and skin present at the amputation site should be clean and clear of pressure sores and blisters. The length of the residual limb also influences

stability and mobility of prosthesis use post-amputation (O'Keeffe & Rout, 2019). Ideal residual limb state is obtained through post-operative care, regular physical checks and monitoring. Development of abnormal transfemoral residual limb anatomy occurs when these criteria are not met, or when further complications occur. Abnormal residual limb anatomy may be caused by improper care after amputation, incorrect bandaging or poor integration between the residual limb and the prosthesis.

Physical appearance, pain experienced by the patient and hindrance in mobility may be used to identify abnormalities in residual limb anatomy. One of these abnormalities includes phantom pain. Phantom pain is described as the sensation of perceiving the portion of the amputated limb (Browner, 2009). The symptom may affect as many as 80% of amputee patients, where they experience pain that can be described as sharp and electric, or a burning sensation within the amputated limb (Browner, 2009). While phantom pain effects the majority of amputees, the primary focus of abnormal residual limb anatomy will be on physical pain and visible abnormalities. Problems in residual limb healing may arise from the following abnormalities:

- Aggressive bone edges refer to distal bone ends of a residual limb that have not been blunted. This can cause ulceration and put stress on soft-tissue of the residual limb (Henrot et al., 2000).
- Heterotopic ossification refers to pathologic ectopic bone growth (forming bone material where it does not belong) in soft tissue (Amendola & Harris, 2018).
- Osteomyelitis (bone infection) (Henrot et al., 2000).
- Neuroma, which may occur in residual limbs 1-12 months after amputation (Henrot et al., 2000).
- Bursitis and soft-tissue inflammation may be caused by unequal loading during mobility, causing inflammation of soft tissue and bursitis where weight-bearing is affected (Henrot et al., 2000).
- Stress fractures that may be caused by improper alignment between the residual limb and prosthesis (Pascale & Potter, 2014).
- Excess inflammation that may lead to cutaneous lesions (Henrot et al., 2000).

Above mentioned symptoms are identified by a clinician. Characteristics identifiable by the amputee will consist of residual limb shape, complexion and feeling that may require attention from a clinician. Medical attention is required if the residual limb displays

abnormal colouration. This may be indicated by a red complexion accompanied by swelling, or a residual limb with skin that is grey or black in complexion. Temperature is another indicator when assessing the health of the residual limb. Identifiable conditions may be a residual limb that is hot to the touch, and pain that is experienced by the amputee when the residual limb is touched. A strong odour and visible indicators such as drainage from the residual limb and persistent sores are signs of abnormalities that require attention by a physician (Clinicalkey., 2020). Discussed abnormalities are visible in the residual limb's unbandaged state. It is advised that the amputee have their bandages unbandaged every 4-6 hours to inspect the residual limb before re-bandaging (Hayes, 2003).

2.4 Purpose of Residual Limb Bandaging

The primary purpose of bandaging post-amputation is to prepare the residual limb for prosthesis integration (Lorraine et al., 1957). Swelling due to oedema or poor maturation of the residual limb through coning, make it difficult to suitably fit a prosthesis. Re-evaluation and correction of residual limb shape is costly and time consuming for both the patient and amputation team if repeated visits to the clinic are required (Visser, 1998). Correct bandaging techniques ensure that the residual limb is shaped and sized correctly to fit a prosthesis. Shape correction results in the distal end of the residual limb being conical in shape, with a reduced volume post-amputation. Additional adjustments would need to be made to the prosthetic socket if residual limb maturation and volume is incorrect, as the residual limb continues to decrease in volume. This is time consuming for the amputee, the physiotherapist and the prosthetists. The desire to re-shape and shrink the residual limb is due to excess fluid build-up that occurs within the residual limb post-amputation (Lorraine et al., 1957). Bandaging is not solely used to re-shape the residual limb, but also to avoid the formation of adduction rolls that may occur due to excess adipose tissue (Lorraine et al., 1957). Furthermore, the bandaging process assists with supporting soft tissue during the healing phase, where the vascular system is the least efficient and results in the accumulation of fluid (Lorraine et al., 1957). External support plays a crucial role in reducing the build-up of this fluid.

Excess fluid and incorrect residual limb bandaging may lead to adduction rolls, resulting in many adjustments to the prosthesis fitting process. This delays the amputee's ability to adjust to their disability and could discourage them from using a prosthesis (May, 1964). The process to decrease residual limb volume for the transfemoral residual limb is more challenging than the transtibial residual limb, since a specific conical shape is needed for

prosthesis integration, and higher levels of oedema build-up is experienced (May, 1964). Elastic bandage usage holds certain advantages when compared to alternative coning methods. Elastic bandages may be applied immediately after amputation, without excess pressure or discomfort to the amputee (Lorraine et al., 1957).

Bandaging material is elastic to apply pressure to the residual limb (for the coning process) and is re-applied whenever tension is lost throughout the day. Pressure needs to be applied across the entire residual limb to decrease overall volume. Common guidelines of the bandaging technique emphasise a figure-8 pattern that builds pressure from the proximal to distal end of the residual limb to force the excess fluid proximally (Hayes, 2003). Bandaging is performed in diagonal increments during the stump-coning process, with the end result ensuring that all skin of the residual limb is covered and that a pressure gradient is created from proximal to distal of the residual limb (Hayes, 2003). The technique uses a standard elastic crepe bandage and follows the technique needed to achieve a conical shape for the residual limb. Primary objectives of bandaging can therefore be summarised by three points (May, 1964):

1. Bandaging prepares the residual limb for prosthesis integration.
2. The process maintains the overall physical condition of the residual limb.
3. The bandaging process assists amputees to adjust to their disability psychologically.

2.5 Ideal Pressure Application

Exertion of an optimum pressure gradient to the residual limb will aid in controlling swelling caused by oedema, support circulation, and minimise inflammatory reactions (Manella, 1981). oedema control and circulation restoration are influenced by external pressure application, and are significant factors in the promotion of healing and maturation of the residual limb. Shaping of the residual limb is achieved through external pressure application to conically shape the residual limb. Optimal pressure application is needed to maximise healing of the residual limb without inducing further harm caused by incorrect external pressure application. Determination of the ideal range of pressure application has been established in previous studies with the following parameters:

1. External pressures applied to the residual limb of 15 millimeters of mercury (mmHg) or less are not sufficient enough to increase venous return (Varghese, Hindle, Zilber, Perry, & Redford, 1981).

2. Constant application of external pressures above 25 to 30 mmHg, may be potentially harmful (Varghese et al., 1981).

3. An external pressure application of 30 mmHg or greater decreases the rate of venous flow of the leg (Varghese et al., 1981).

4. Application of external pressure in excess of 0.04 kg/cm^2 (approximately 25 mmHg) is potentially harmful to the healing of the residual limb. An external pressure application of 20 mmHg was determined to be optimal (Isherwood, Robertson, & Rossi, 1975).

5. Pressure should be graded so that more pressure is applied to the distal portion of the residual limb than the proximal portion (Manella, 1981).

2.6 Existing Solutions

Current solutions include: elastic bandages, shrinker socks, rigid bandages and semi-rigid bandages. The most commonly used bandage is the soft elastic crepe bandage that is used in conjunction with the standard figure-8 bandaging technique. Figure 4 shows an amputee's transfemoral limb being bandaged during their rehabilitation period.

Figure 4: Transfemoral residual limb being wrapped with elastic bandage (Ottobock, 2020)

Shrinker socks, as seen in Figure 5 (a), are elastic compression socks worn post-amputation to control swelling of the residual limb. The shrinker sock applies a constant, uniform pressure to the residual limb and is a tapered to conically shape the residual limb (Manella, 1981). Easy application of the shrinker sock makes it an ideal option for controlling swelling, but cost and availability may make them inaccessible.

Semi-rigid and rigid dressings are formed from a mould of the amputee's residual limb and is expected to be constantly worn to influence volume of the residual limb. The process of producing moulds is costly and time consuming, requiring multiple iterations as changes in residual limb volume occur during the rehabilitation period. Other limitations include loosening of the dressings over time and restriction of wound inspection. Dressings need to be applied by a physician possessing the necessary skills, as well as close supervision following application. Figure 5 (b) shows the semi-rigid bandage applied to an amputee's transibital residual limb.

(a) (b)

Figure 5: (a) Shrinker sock applied to a transibital residual limb (Datta & Atkinson, 2009), (b) Removable rigid dressing on residual limb (Janchai et al., 2008)

The primary function of the prosthesis is mobility, but certain components assist with maintaining correct shape and volume of the residual limb. The liner of the prosthesis, illustrated in Figure 6, is the layer applied over the residual limb before the socket is applied. The liner is padded and made of silicone or various gels to prevent friction and excess pressure while the prosthesis is in use (Boldt & Maguire, 2013). The socket is the most proximal part of the prosthesis and acts as the interface between the residual limb and the remainder of the prosthesis. Sockets are commonly made from laminated material or thermoplastic to be moulded from the amputee's mature residual limb (Kapp, 2000). Amputees experiencing vascular problems would continue to bandage

their residual limb to maintain a constant volume and shape to ensure secure fitting of the prosthesis throughout the day (Boldt & Maguire, 2013).

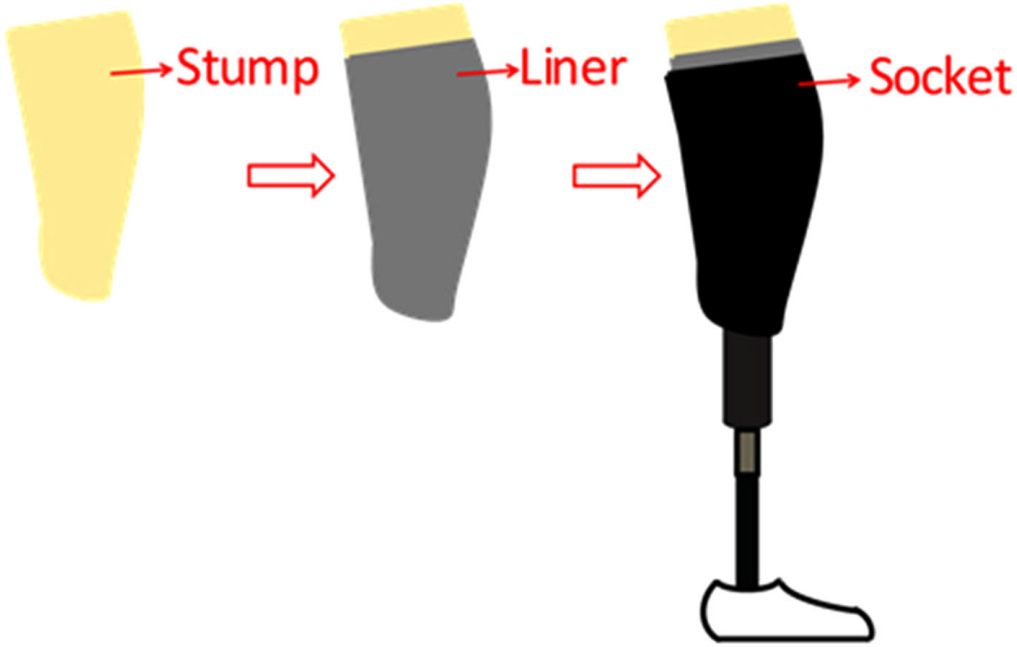


Figure 6: Proximal components of transfemoral prosthesis (Paternò et al., 2018)

Proper bandaging is required to prepare the residual limb for optimal prosthesis integration and comfort during mobility, dependent on the fit of a prosthetic liner and socket. Current discussed solutions are costly, require skill to apply and are not easily accessible. The researched solution method developed for this study would promote proper bandaging and address limitations of current solutions regarding independence, cost and pressure application.

3 Design Methodology

3.1 Introduction

This chapter outlines developed sub-systems forming the overall solution method assisting with residual limb bandaging. The aim of this study was to develop a solution method to assist and guide an amputee with the early stages of residual limb bandaging post-surgery. Overall design was influenced by inexperience in residual limb bandaging, with the possibility of amputees performing bandaging for the first time. Design strategies required the solution method to have concise and easily interpretable instructions on residual limb bandaging to achieve the desired residual limb shape suitable for prosthesis integration. This assistance guides the amputee through the residual limb bandaging process, while promoting the ideal conical shape achieved through pressure application by an elastic crepe bandage. The overall solution method was partitioned into three sub-systems: A visual aid to guide bandaging, an interactive guidance system that incorporated instructions given by the visual aid and a bandaging dispenser that measured the length of unrolled bandage. Each sub-system is discussed in more detail in Chapter 4.

3.2 Visual Bandaging Aid

A visual aid concept is required to assist amputees with residual limb bandaging and promote independence. Current guidelines and techniques for residual limb bandaging require assistance, and do not utilise residual limb measurements. Incorporating current measurements of the residual limb into bandaging guidelines promotes the development of the desired conical shape necessary for prosthesis integration through custom guidelines based on current residual limb measurements. The bandaging aid would allow for patient specificity through the incorporation of residual limb circumferential measurements to guide bandaging. Constant measuring and inspection of the residual limb enable the ability to track and estimate progress of residual limb maturation. Consideration of the residual limb's current state and measurements influences the design of the visual aid, allowing it to adjust to changes in residual limb volume. To convey bandaging guidelines, the visual aid was printed onto an A4 page that acts as a template for bandaging. The template is intended to be directly applied to the residual limb to conduct bandaging over it. Bandaging is completed following guideline markers present on the template, with further instructions to be conveyed by the guidance system. An A4 page medium

was chosen to facilitate accessibility, cost-effectiveness and a medium that would exert minimal pressure to the residual limb. A thin medium exerting minimal pressure allows for the removal of the template from the residual limb after completing bandaging. The process is summarised in Figure 7.

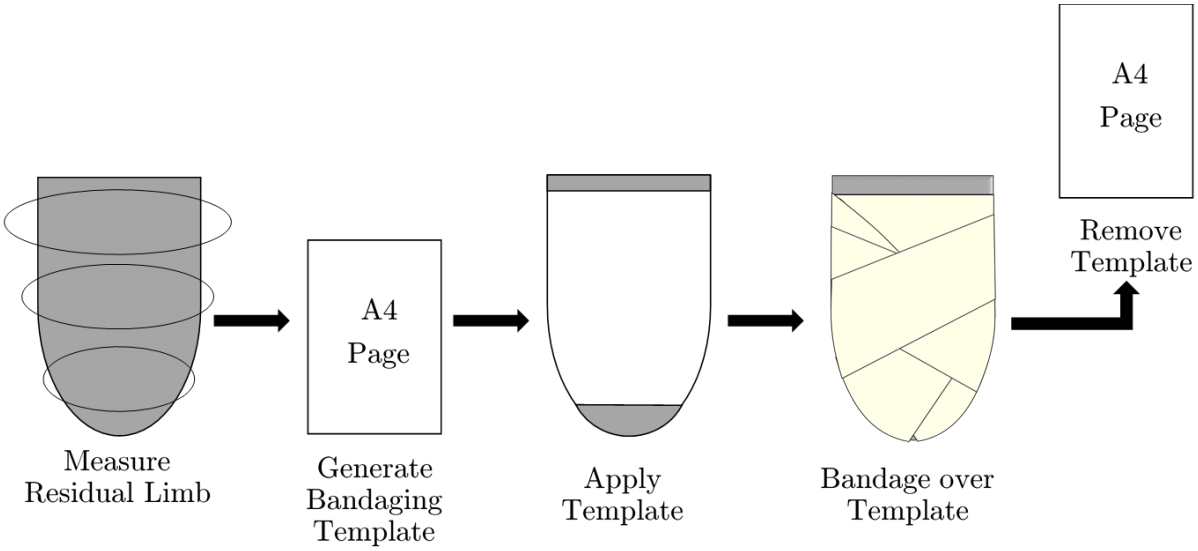


Figure 7: Incorporation of the bandaging template in the bandaging process

3.2.1 Bandaging Template Design

To achieve the desired conical shape for the residual limb, design of the template shape was modelled after a truncated cone. Template specifications were correlated to the Ace bandaging technique, described in Figure 8, used to control swelling caused by fluid accumulation at the distal portion of the residual limb. The bandaging method concentrated on diagonal bandaging, similar to a figure-8 pattern, to control the flow of excess fluid and achieve the desired conical shape. Bandaging in this manner applies adequate pressure to the corners of the residual limb without inflicting any injury to the healing wound site from the amputation procedure.

Ideal bandaging techniques were conveyed by the template through the placement of calculated landmarks, positioned according to current circumferential measurements of the residual limb. Calculated landmark positioning allowed the template to adapt to changing circumferential measurements. Maintaining a constant bandaging technique, illustrated by the bandaging template, allows for a controlled and predictive form of residual limb volume and shape control.

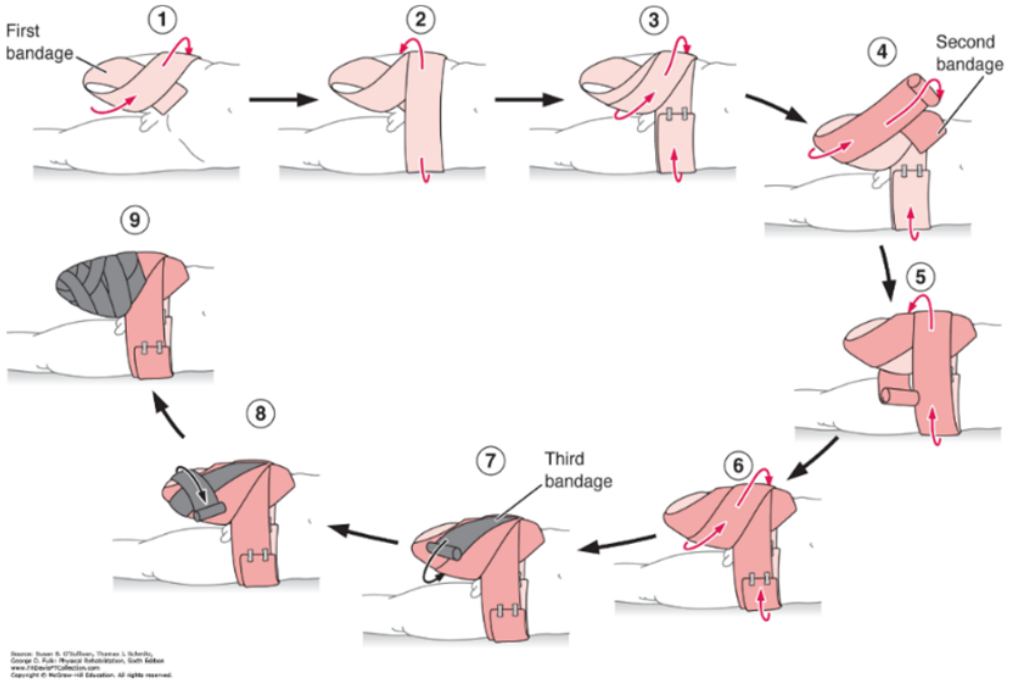


Figure 8: Transfemoral residual limb bandaging technique (O'Sullivan et al., 2019)

3.2.2 Bandaging Template Landmark Generation

Landmarks were generated on each template to fix bandaging guidelines in place. Numbered regions represent each guideline and function in tandem with instructions displayed by the guidance system. Landmarks were positioned according to circumferential residual limb measurements and were distributed in a manner that accommodated the width of a common 4-inch elastic crepe bandage. Positioning of landmarks was accomplished using plotting and mathematical functions and libraries of the Python programming language. The NumPy Python library supports multi-dimensional arrays, as well as imports a large collection of high-level mathematical functions (Oliphant, 2006, p. 13). The library enabled the calculation of coordinates for each landmark point to be plotted on a graph that represents the bandaging template. Calculations determining landmark coordinates utilised formulae such as: the distance between two points equation $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, the equation of a line $y = mx + c$ and methods to determine intersecting points between two lines. After determining landmark coordinates, coordinates were plotted using the Matplotlib Python library on a 2-dimensional graph scaled to the dimensions of an A4 page. Simulation of A4 page dimensions was accomplished by scaling the x and y-axis of the graph in inches, constraining the x-axis and y-axis to the dimensions of an A4 page (210 mm $\times$ 297 mm).

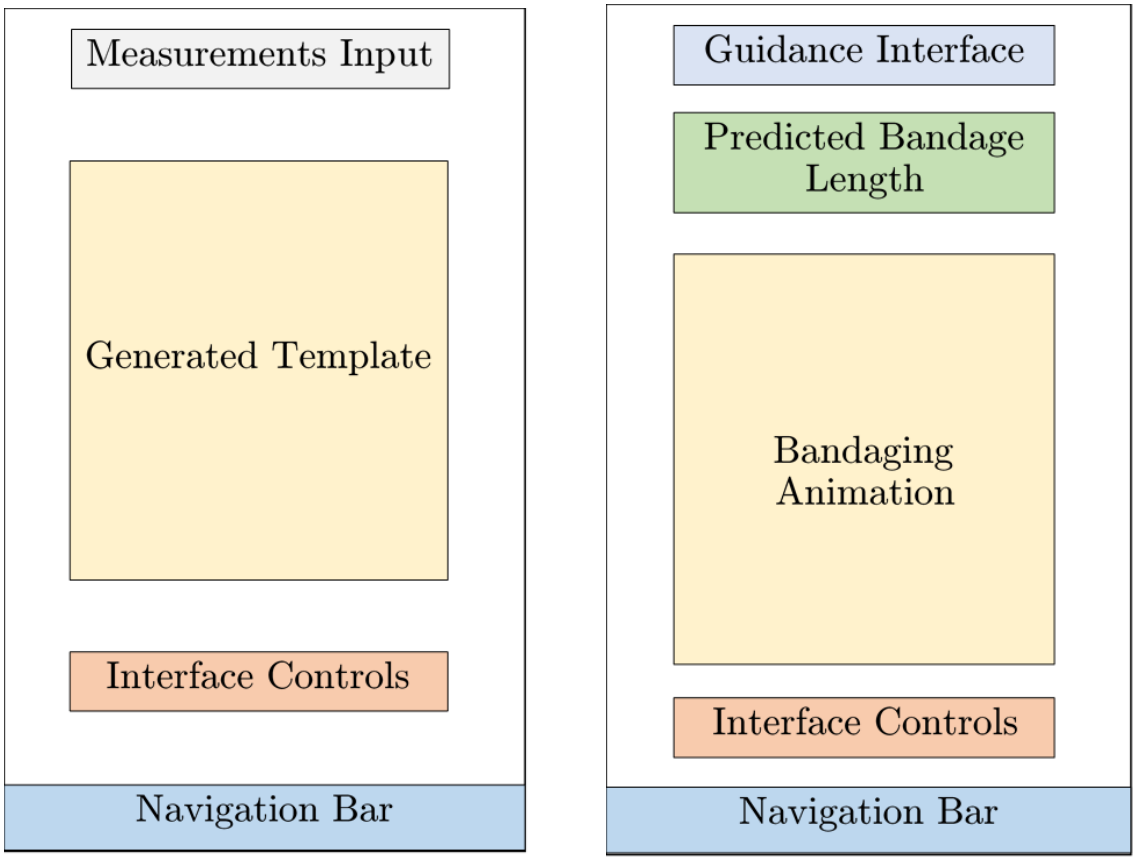
3.3 Guidance System

After template generation was completed, the bandaging guidance system assisted with bandage application. Instructions were given to complete bandaging based on numbered guidelines present on the bandaging template. The guidance system indicated the method of bandage application required for each step of the total bandaging process. Information displayed by the system included a demonstration on how bandaging should be performed, as well as target bandage lengths to be used for the current bandaging step of the process.

Target bandage lengths for each step were generated by a custom Python script discussed further in Sub-section 4.2 and presented in Appendix C.2. The script utilised the position of each landmark to predict required bandage lengths. Predicted lengths were calculated by adding the circumferential measurements to linear distances between two landmarks and applying a compression factor. The compression factor adjusted the relaxed bandage length to a stretched length holding sufficient tension to apply desired pressure to the edges of the residual limb. Predicted target lengths and bandage instructions were displayed by a user interface present on a mobile application.

3.4 Mobile Application Features and Design

For portability and easy accessibility, the bandaging template and guidance system were adapted to operate on a standard smartphone. Both systems were represented through a single mobile application user interface that operated according to feedback in the form of user input and control. The mobile application incorporated the previously mentioned custom Python scripts. Bandaging templates were generated by the first script, with the second script calculating required bandage lengths visually represented by the guidance system through the user interface, as seen in Figure 9 (a). The page directing the generation of the bandaging template contained editable text fields to allow the input of residual limb circumferential measurements. Interface controls allowed for the generation of the template, utilising the first Python script responsible for plotting bandaging templates through the use of mathematical and plotting libraries. The guidance interface in Figure 9 (b) displayed the target length of bandage to be unrolled and applied for the current bandaging step. Positioned below the predicted length, a bandaging animation continuously animated to demonstrate the technique needed to complete the current bandaging step. Interface controls were placed below the animation field to navigate between bandaging steps and to manually establish a Bluetooth connection with the bandaging dispenser.



(a) (b)

Figure 9: (a) Template generation interface, (b) Guidance interface

3.5 Bandaging Dispenser

The final sub-system involved a bandaging dispenser device that unrolled an attached elastic crepe bandage roll, while measuring the output length. The dispenser encasing was modelled using a modeling computer-aided design program, allowing the dispenser to be 3D printed for rapid production and prototyping, Design specifications were established before modelling and 3D printing. This included design specifications of the encasing that would prevent injury upon contact by rounding dispenser edges. Further specifications enabled the incorporation of a measuring mechanism that outputted bandage length and measured the unrolled length. An elastic bandage was rolled onto a rotating mechanism that allowed for the re-attachment and unrolling of the bandage. A rotary encoder and an ESP32 microcontroller measured and recorded unrolled bandage lengths and transmitted these measurements to the mobile application using Bluetooth.

4 Design Outcome

Development of each sub-system of the overall solution method assisting transfemoral amputees with residual limb bandaging is outlined in this chapter. The objective through assisted bandaging is to achieve a shorter healing period and promote faster prosthesis integration through residual limb re-shaping. The first sub-system comprised of a bandaging template applied directly onto the residual limb for bandaging to occur over. The second sub-system was a mobile application that received input in the form of circumferential measurements of the amputee's current residual limb volume, and generated a patient specific bandaging template. The final sub-system was a bandaging dispenser that measured unrolled bandage lengths. This was used to control bandage application guided by instructions conveyed by mobile application and bandaging template specifications.

4.1 Bandaging Template Design

Appendix A displays a generated bandaging template with landmark points, template dimensions and bandaging step numbers to coordinate the bandaging process. Final design of the bandaging template was achieved through the plotting of landmarks by a custom Python script to construct a bandaging template pattern. Landmarks were used to establish the position of boundaries used to govern the placement of elastic bandage after applying the bandaging template to the residual limb. Positioning of landmarks and template dimensions were influenced by circumferential measurements of the residual limb. Patient specific bandaging templates can therefore be generated by incorporating circumferential measurements of the residual limb as a design input, allowing template guidelines to adapt to changes in residual limb volume as maturation occurs. Guidelines were generated by a custom script written in the Python programming language for the purpose of utilising plotting libraries, such as Matplotlib to plot coordinated landmarks and areas of the template using residual limb circumferential measurements as inputs.

4.1.1 Required Residual Limb Measurements

In total, five circumferential measurements were required to complete the bandaging process. Two initial parameters were used by the Python script as an input to generate each template. This included the most distal circumferential measurement, followed by a

circumferential measurement 30 cm proximal to the initial measurement, as seen Figure 10. A length of 30 cm was chosen to facilitate the dimensions of an A4 page, measured to have a length of 29.7 cm. A4 page dimensions were used to generate landmark point coordinates, and position bandaging guidelines. Generated landmark points were joined using line segments to display areas controlling bandage placement on the residual limb.

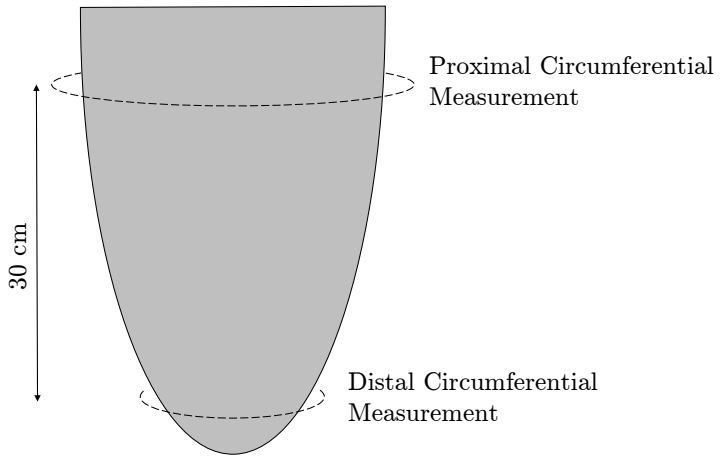


Figure 10: Initial circumferential measurements

Following template generation and placement onto the residual limb, three additional circumferential measurements displayed in Figure 11, were taken to be used as input for a second Python script. This script calculated predicted bandage lengths needed to complete residual limb bandaging in a four step process.

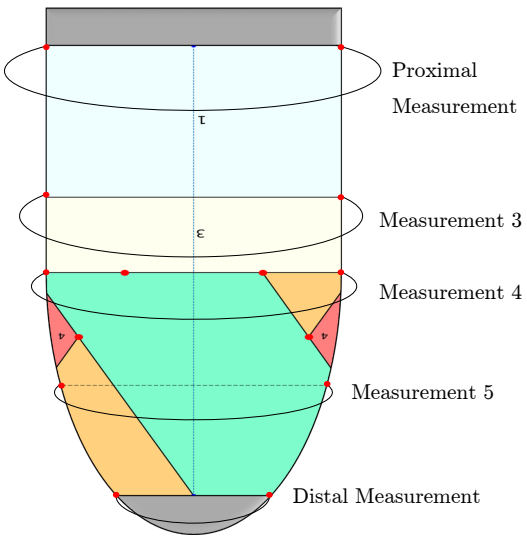


Figure 11: Additional circumferential measurements

4.1.2 Four-Step Bandaging Process

Predicted bandage lengths are calculated by multiplying unstretched bandage lengths required to cover each step of the bandaging process by a compression factor to achieve a stretched bandage length tense enough to exert an ideal pressure of 20 mmHg. The overall bandaging process is comprised of four steps, seen in Figure 12.

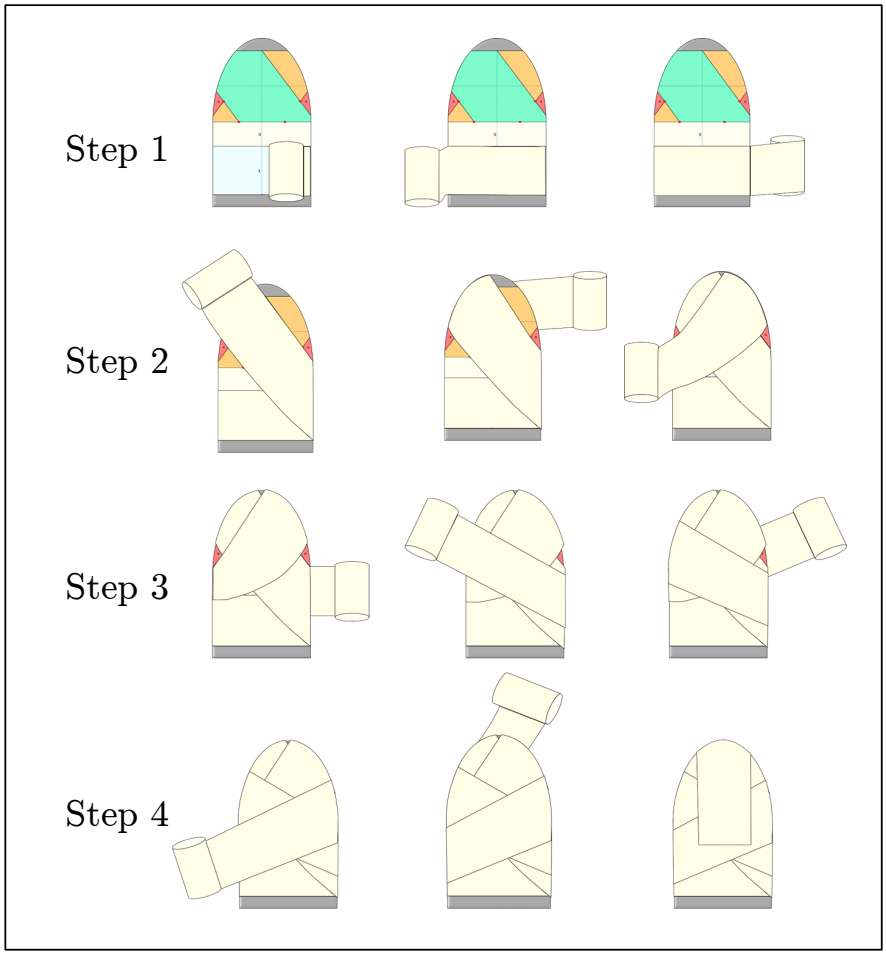


Figure 12: The four steps of the bandaging process

Predicted bandage length requirements and completion of bandaging is accomplished by the four bandaging steps described:

1. Apply the bandage anchor, by laterally wrapping the residual limb at the proximal most region indicated by the blue area of the bandaging template. The length required for this step is the initial circumferential measurement, C_1, multiplied by the compression factor of 0.895 calculated in Sub-section 4.2.1.

2. Wrap the distal ends of the residual limb by diagonally applying the elastic bandage within the green and orange regions of the bandaging template. The required

bandage length is calculated by adding the distance between points P_1 and P_{13}, the distance between points P_{12} and P_5, twice the arc length of the distal end of the residual limb and subtracting the posterior region between points P_{12} and P_{13}, estimated to be a sixth of the fifth circumferential measurement. The total bandage length requirement is multiplied by the compression factor to achieve the stretched bandage length.

3. The following step covers the remaining unbandaged areas of the body of the residual limb, by diagonally wrapping the areas marked in red. The total length is obtained by adding the second circumferential measurement, the distance between points P_4 and P_{13}, the fourth circumferential measurement and multiplying the summed length by the compression factor.

4. The final step covers the open area at the distal end of the residual limb, where the amputation scar is found. Calculation of the required length involves the multiplication of the distal residual limb arc length by the compression factor, ensuring that bandage application is tense enough to maintain its position without harming the amputation scar. Bandaging is fixed to the residual limb body by a bandage clip or tape.

Following sub-sections within this chapter detail the design process of the bandaging template, the calculation of the compression factor and the calculations of predicted bandage lengths required to complete the four-step bandaging process

4.1.3 Template Pattern and Script

Template shape was modelled after a truncated cone, with the template covering approximately a third of the total residual limb area. A truncated cone shape was chosen to reflect the conical shape of a residual limb and to facilitate "residual limb coning" to influence maturation of the residual limb into a shape suitable for prosthesis integration. Template design was therefore based on the mathematically generated pattern of a truncated cone illustrated in Figure 13.

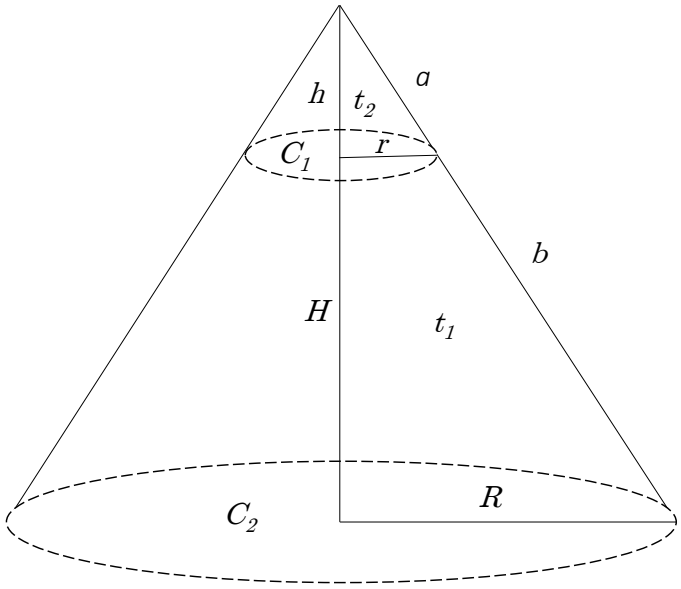


Figure 13: Truncated Cone Pattern

Requirements for each pattern included the top (C_1) and bottom (C_2) circumferential measurements of a truncated cone, as well as the height (H) of the cone. A fully generated pattern of a cone is adjusted into the desired truncated cone shape that could be applied to the residual limb. Generation of a truncated cone pattern is achieved by first producing a standard cone shape and removing the peak forming a point (Wilson, 2020). Required dimensions to produce a 3D truncated cone are calculated using the following formulae:

$$\frac{h}{r} = \frac{h+H}{R}, \tag{1}$$

$$a = \sqrt{h^2 + r^2} \tag{2}$$

$$\begin{aligned} (a+b)^2 &= (h+H)^2 + R^2 \\ b &= \sqrt{(h+H)^2 + R^2} - a \end{aligned} \tag{3}$$

where:

t_1 = Triangle 1

t_2 = Triangle 2

h = Height of t_2

H = Height of $t_1 - h$

r = Radius of C_1

R = Radius of C_2

Height h corresponds to the height of $t1$, and is calculated using Formula 1 for similar triangles. The height of the truncated cone is represented by H, with a constant dimension of 29.7 cm to accommodate the height of an A4 page onto which the template is printed. The lengths of a and b are calculated using Formula 2 and Formula 3 respectively. A side length of b is calculated to determine the length of each side of the truncated cone, while a is calculated to determine the side length of the cone's peak to be removed. The two radii displayed in Figure 13 correspond with the two circumferential measurements of the residual limb needed to generate a bandaging template. After calculating the height of t_1 and the outer edge of the truncated cone, b, the 2D representation of the truncated cone is produced on a circular disc, shown in Figure 14.

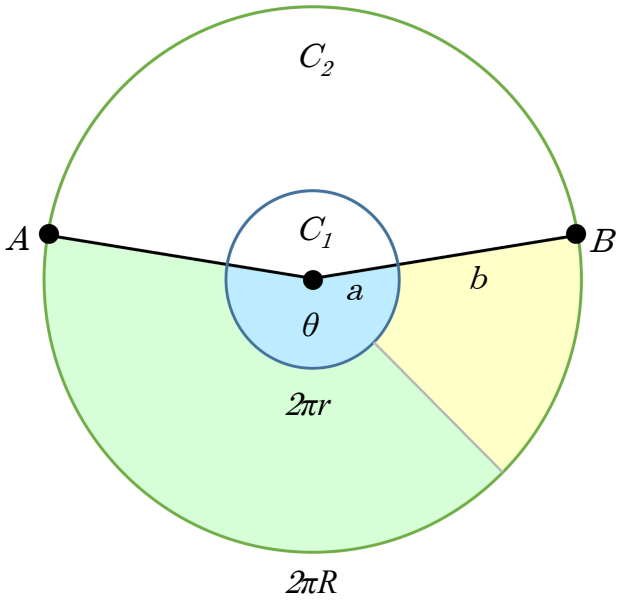


Figure 14: Flattened, 2-Dimensional view of the Truncated Cone Pattern

Required variables needed to construct the disc pattern are calculated using the following formulae:

$$\begin{aligned} S &= \theta r \\ 2\pi r &= \theta a \\ \frac{2\pi r}{a} &= \theta \\ \frac{2\pi r}{a}\frac{360}{2\pi} &= \theta \end{aligned} \tag{4}$$

where:

S = Arc length

θ = Angle in degrees at which the truncated cone is constructed

Figure 14 shows the 2D representation of the truncated cone pattern (green and yellow regions) positioned on a disc with circumference $2\pi R$. The inner circle with a circumference of $2\pi r$ marks the area to be removed from the pattern to convert the cone to a truncated cone. Angle θ, in degrees, is calculated to determine the position of points A and B. The two points, together with the centre of the circle, O, are used to position radii $\overleftarrow{AO}$ and $\overleftarrow{BO}$. The two radii mark the boundary of the truncated cone pattern to be extracted from the full circle. The yellow region is calculated as a third of the total truncated cone pattern to facilitate A4 page dimensions for printing, and marks the area onto which the bandaging guidelines are generated. Generation of the truncated cone template is achieved by taking in three parameters as input for Python Script 1 located in Appendix B.1: the proximal circumference of the residual limb (C_2), the distal circumference of the residual limb (C_1), and the height of the truncated cone pattern given by the length of a standard A4 page ($H = 29.7$ cm).

Circumferences $C1$, $C2$ and height H are passed as parameters into the *SideLength* function found at line 1, and outlined in Figure 15. The function utilises Formulae 1, 2 and 3 to generate the side length, b, and establish a side boundary for the bandaging template. Outer boundaries, as well as bandaging guideline boundaries are plotted by calling the *PlotLine* function at line 10. Two points are passed as parameters, with a line segment plotted to join these two points. A line segment is plotted on a graph scaled in inches (measurement units supported by Python), with x and y coordinates

plotted in units of 2.54 cm (1 inch = 2.54 cm) per point. Bandage guideline boundaries require landmark points to plot each line segment using the *PlotLine* function. These points are determined by calling the *EquationOfLine* function at line 16 and the *SolveForx* function at line 23.

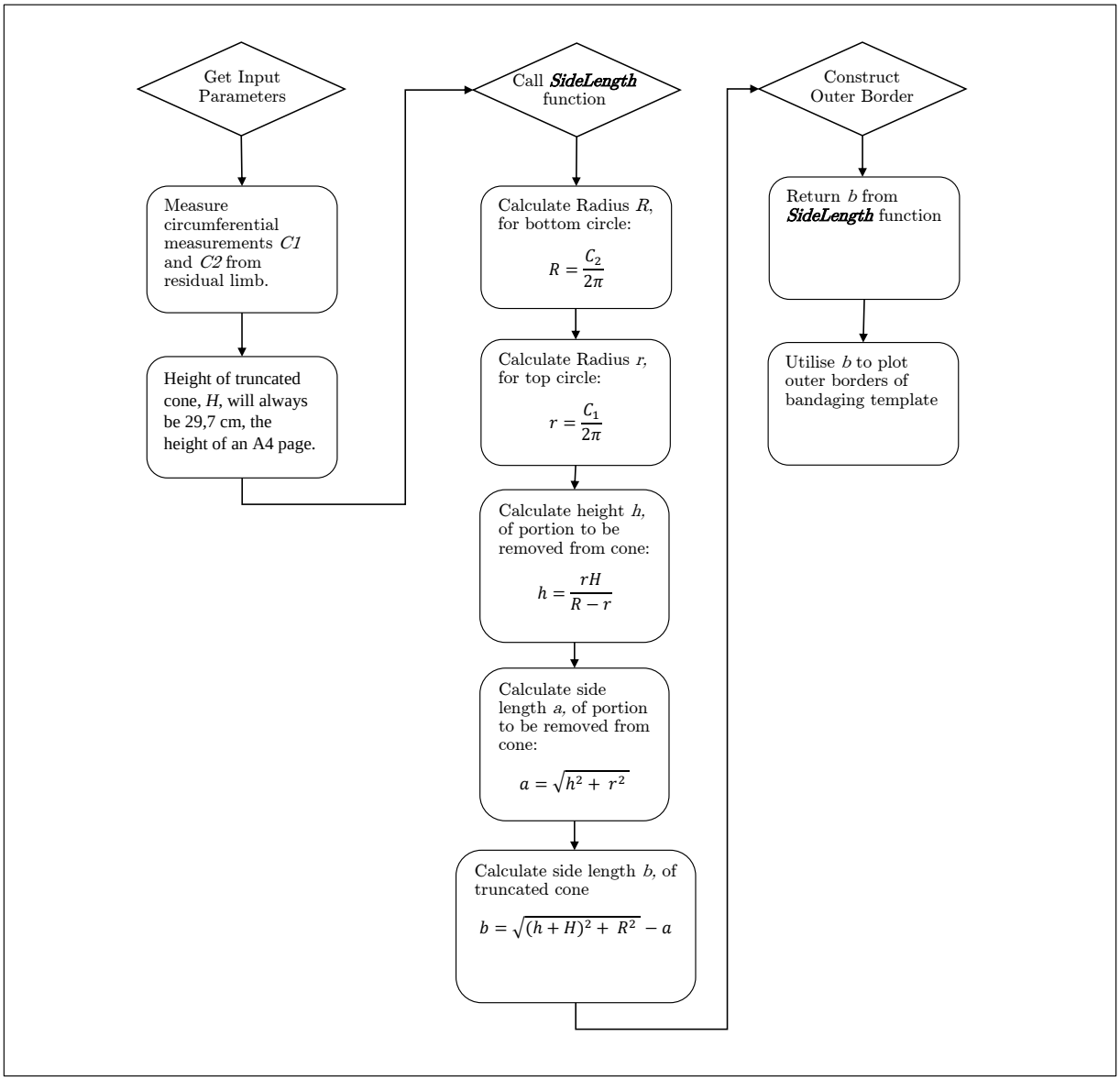


Figure 15: Overview of the *SideLength* function, used to calculate template border dimensions

The *EquationOfLine* function passes in two points as parameters and returns the equation of a line in the form of $y = mx+c$. Equations of lines of length b forming outer boundaries are used to solve for x and y coordinates of points coincident with line. The *SolveForx* function is called to determine coordinates of points coincident with side boundary lines. This function solves for the unknown x_1-coordinate within the distance formula for two points, $d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, where d, x_2 and y_2 are substituted into the formula.

The equation of the boundary line is substituted as y_1. Distance is known to be 10 cm to accommodate the placement of a 4-inch (10.16 cm) elastic crepe bandage, indicated by green and orange coloured regions of the bandaging template shown in Appendix A. Coordinates of points coinciding with the intersection of two lines, as shown in Figure 16, are calculated by the *IntersectionOfTwoLines* function at line 29.

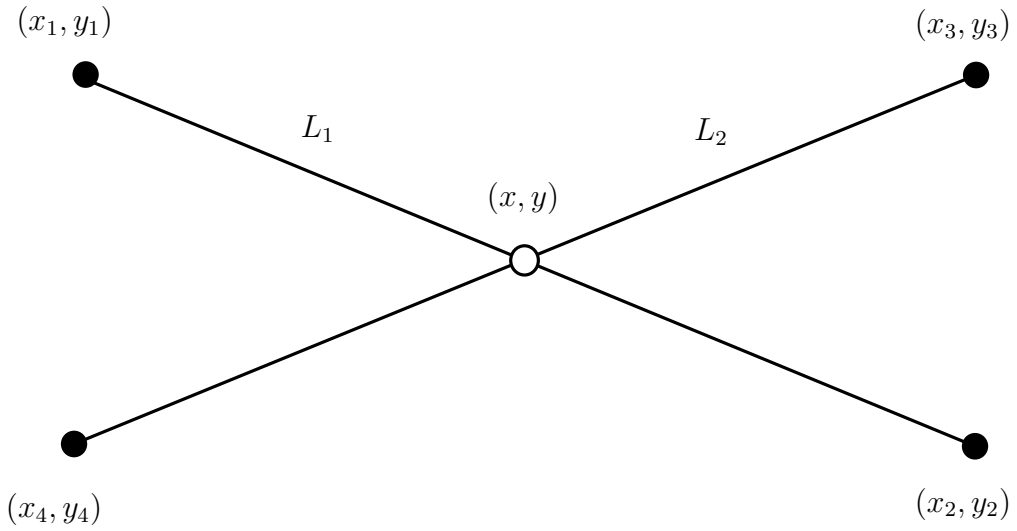


Figure 16: Two intersecting lines with point of intersection (x, y)

Four points are passed into the function as parameters, where respective x and y coordinates are used to solve for the intersecting point between two lines, using the line–line intersection method (Weisstein, 2020). Given two lines, L_1, defined by the points (x_1, y_1) and (x_2, y_2) and L_2, defined by the points (x_3, y_3) and (x_4, y_4), the intersecting point P can be defined using determinants:

$$P_x = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}} \qquad P_y = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}}$$

Which can then be written as:

$$(P_x, P_y) = \left(\frac{(x_1y_2 - y_1x_2)(x_3 - x_4) - (x_1 - x_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)}, \frac{(x_1y_2 - y_1x_2)(y_3 - y_4) - (y_1 - y_2)(x_3y_4 - y_3x_4)}{(x_1 - x_2)(y_3 - y_4) - (y_1 - y_2)(x_3 - x_4)} \right) \tag{5}$$

Remaining functions are called to plot additional points and colour bordered regions used to indicate bandage placement while wrapping the residual limb. Running the Python script generates the bandaging template shown in Appendix A for a residual limb with a proximal circumferential measurement of 60 cm and a distal circumferential measurement of 40 cm. Generated templates cover approximately one-third of the anterior region of residual limb area, based on two circumferential measurements (each circumference is divided by 3). Restricting template dimensions allows for the placement and printing of the template onto an A4 page. Four steps conduct the entire bandaging process, indicated by numbered and coloured regions present on the bandaging template. Each region of the four bandaging steps, as seen in Appendix A, is generated by Python Script 1.

4.1.4 Bandaging Process Steps Generation

The four steps of the bandaging process are generated in the following manner, with the generation process summarised in Figure 17:

Step 1 creates an anchor for the bandaging technique, by applying a single lateral wrap around the proximal circumference of the residual limb. The region is colour-coded in blue, and is 10 cm in height to accommodate a 4-inch (10.16 cm) bandage width. Points P_1, P_2 and P_3 are plotted using coordinates corresponding to the dimensions of an A4 page (21 cm $\times$ 29.7 cm). A y coordinate of 29.7 is used for each point, with an x coordinate of 0, 10.5 and 21 respectively. Coordinates for point P_{14} are determined by substituting a y value of 0, a distance of b calculated by the *SideLength* function and the coordinates of P_1 into the distance formula, and solving for the unknown x coordinate. The x coordinate is subtracted from 21 (total width of an A4 page) to achieve the x coordinate of point P_{16}, reflected about line $\overleftrightarrow{P_2P_{15}}$, the centre of the template. Calculation of the equation of the line between points P_1 and P_{14}, with length b, is achieved by calling the *SideLength* and *EquationOfLine* functions. The same method is used to calculate the equation of the line between points p_3 and P_{16} to establish the right border of the template. Determining each

equation makes it possible to calculate the x-coordinate for p_4 and p_5 by substituting their respective y-coordinates (y-coordinate calculated as: 29.7 - 10 = 19.7) into the equations of the lines coincident with the points. Segments are plotted to join points p_1, p_3, p_4 and p_5 to construct and colour the region governing step 1 of the bandaging process.

Step 2 is marked by green and orange coloured regions on the template, and controls the diagonal placement of elastic bandage at the distal corners of the residual limb. Each region promotes the diagonal, figure-8 bandaging technique to control swelling at the distal end of the residual limb. Calculation of the x-coordinate for points P_{12} and P_{13} are determined when calling the *SolveForx* function and substituting the equation of line $\overleftrightarrow{P_1P_{14}}$ or line $\overleftrightarrow{P_3P_{16}}$ as parameters for the y_1 coordinate into the distance formula and solving for the x_1 coordinate. A fixed distance of 10 is substituted into the distance formula to accommodate the placement of a 4-inch elastic crepe bandage. Coordinates of P_{15} are substituted into the distance formula as x_2 and y_2 to solve for the unknown x-coordinate. The same method is applied to determine the coordinates of point P_{13}. Points P_7 and P_8 are calculated as the midpoint between the centre of the template and points P_6 and P_9 respectively. Segments between points P_6, P_8, P_{13}, P_{15} and P_{16} are plotted to form the green shaded region of the template, whereas points P_7, P_9, P_{12}, P_{14} and P_{15} are used to form the orange shaded region. Both regions are used to conduct diagonal bandaging of the distal end of the residual limb.

Step 3, marked by the yellow region, acts as an additional anchor to secure the lateral wrap performed at step 1, by overlapping the previously placed bandage by 5 cm. Performing the step applies an additional lateral wrap and re-orientates bandaging to prepare for step 4. Positioning of the region is accomplished by determining the coordinates of P_6 and P_9, by substituting their y-coordinates (y-coordinate calculated as: 29.7 - 10 - 5 = 14.7) into the equation of coincident lines and solving for the unknown x-coordinate of each point. Segments are plotted to join points P_4, P_5, P_6 and P_9 and establish the region.

Step 4, highlighted by red coloured regions, is the final step of the bandaging process appearing on the template. Coordinates for points required to border the region are found on the lines $\overleftrightarrow{P_1P_{14}}$ and $\overleftrightarrow{P_3P_{16}}$. Coordinates for points P_{10} and P_{11} are calculated by calling the *IntersectionOfTwoLines* function. The x and y coordinates for point P_{10} are returned after passing in the coordinates of points P_6, P_7, P_{12} and P_{15} for the purpose of implementing the line-line intersection method to calculate the coordinates of a point at the intersection of two lines. The same method is used to calculate the coordinates of point P_{11}, where points P_8, P_9, P_{13} and P_{15} are passed into the *IntersectionOfTwoLines* function. Bordering points for each region are joined by segments and are coloured red to indicate the final step coordinating diagonal bandaging.

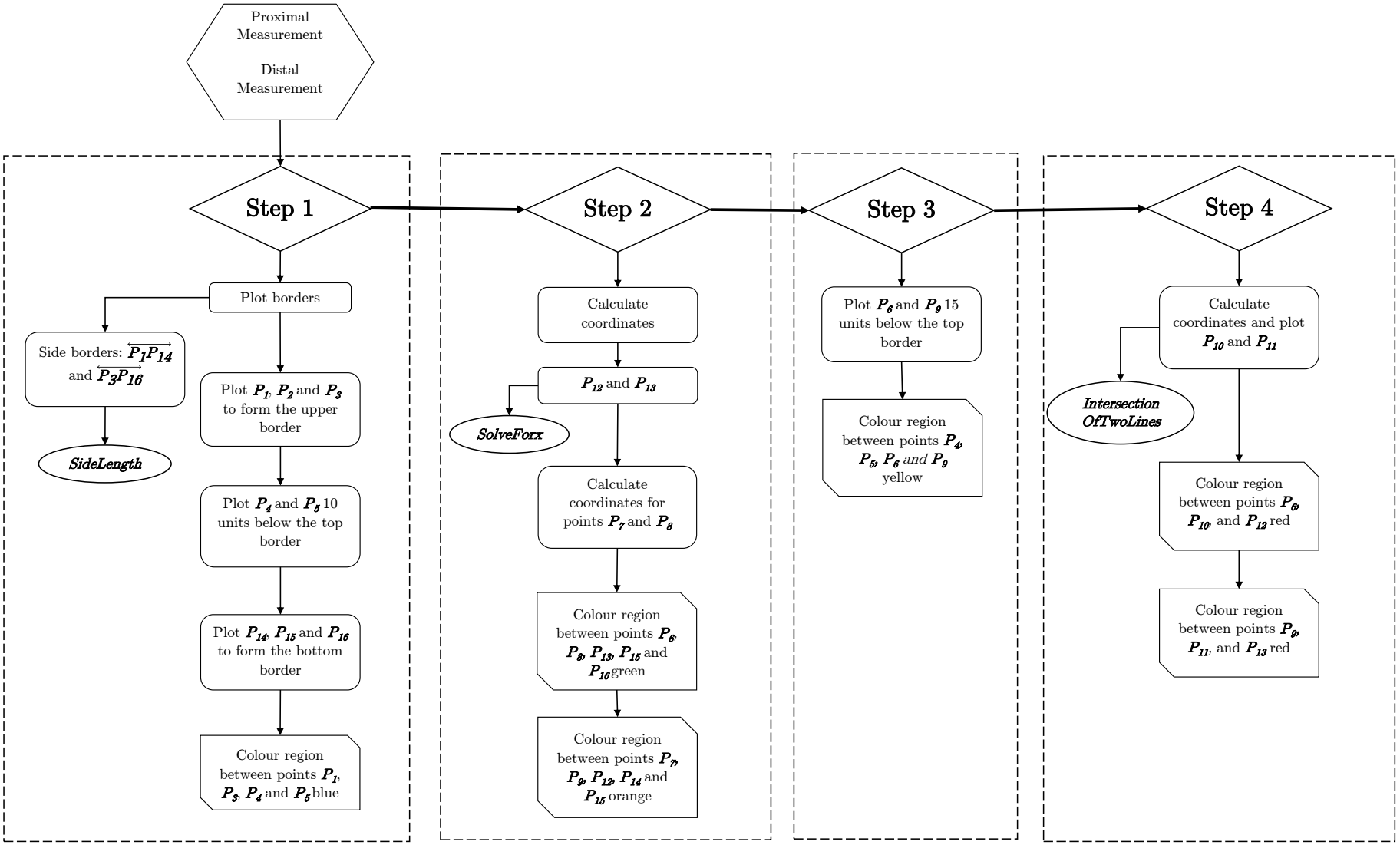


Figure 17: Bandaging template generation flow chart

4.2 Predicted Bandage Length Requirements

Predicted bandage lengths are generated by Python Script 2, located in Appendix B.2, using: the distance formula between two points to calculate the linear distance between two template landmarks, five residual limb circumferential measurements, taken as input parameters and the arc length of a segment to determine the arc length of the distal portion of the residual limb. A 3-dimensional representation of the residual limb is stretched to achieve a flat 2-dimensional view of the residual limb, similar to the 2D truncated cone pattern as seen by Figure 14. This allows for the calculation of the distance between two points linearly, despite curved segments present, as seen in Figure 18.

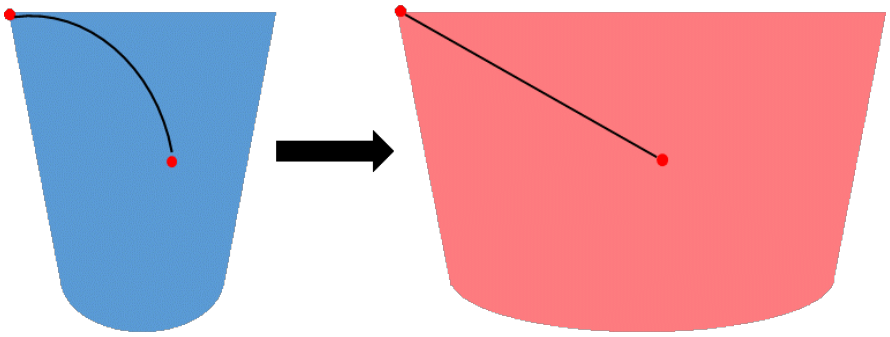


Figure 18: Curved to linear measurements

Required bandage lengths to be unrolled and applied to the residual limb are calculated by Python Script 2, by adding subsequent bandage lengths calculated by the distance formula, circumferential measurements and the arc length of the distal end of the residual limb. The arc length estimates the length of bandage needed to cover the distal corners of the residual limb, while diagonally wrapping the limb. The length can be derived using the formulae for a circular segment (Weisstein, 2015).

$$R = \frac{W^2}{8H} + \frac{H}{2}, \tag{6}$$

$$\theta = 2 \arcsin \frac{W}{2R}, \tag{7}$$

$$S = \theta R \tag{8}$$

where:

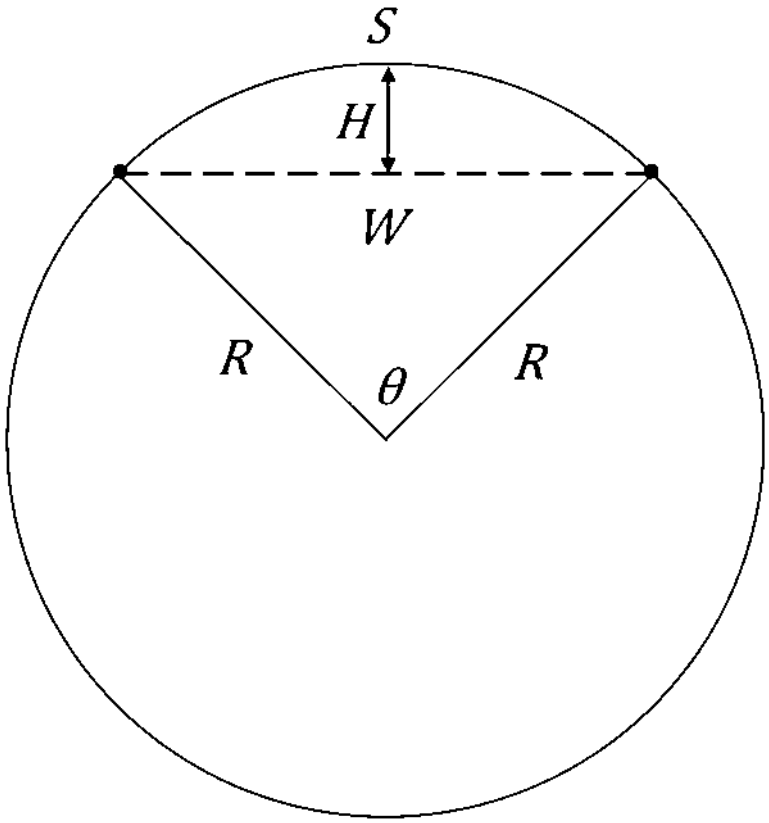


Figure 19: Arc length of a segment

R = Radius of the circle

W = Width of the chord length

H = Height of the segment

S = Arc length

θ = Angle in radians between the lines forming the arc

Formulae required to determine the arc length of a segment are applied using template landmarks and dimensions of the patient's residual limb to determine the length of bandage needed to wrap distal corners of the residual limb. Arc length is calculated using the *ArcLength* function at line 1, called in the Python 2, where the fifth circumferential measurement, shown in Figure 11, is passed in as a parameter. Height (H) is fixed at 6.65 cm, measured from the base of the A4 page to the segmented line positioned between points P_{12} and P_{13}. Chord length (W), of the segment is established as the diameter of the fifth circumferential measurement. The arc length of the segment is then returned and used to predict required bandage lengths in conjunction with the *Distance* function present at line 10. The *Reflectedx* function at line 15 is passed a list of coordinates as a parameter, and reflects these points about the y-axis to mirror

existing coordinates and establish additional landmarks needed to calculate required bandage lengths per step of the bandaging process.

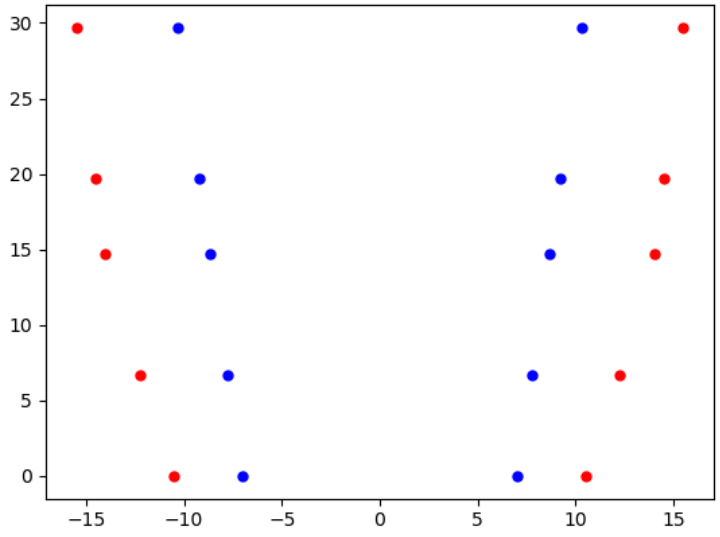


Figure 20: Bandaging template and residual limb landmarks

Coordinates are plotted on a graph scaled in inches, as seen in Figure 20, where blue points outline the placement of the bandaging template, and red points outline the anterior half of the residual limb. Circumferential measurements, linear distance between two points and calculation of the arc length simulating the distal portion of a residual limb are all used to predict bandage length requirements for each step of the bandaging process. Initial predicted bandage length measurements represent unstretched lengths of elastic bandage experiencing no tension, resulting in bandage application exerting minimal pressure to the residual limb, therefore limiting the ability to correctly shape the residual limb. Inclusion of a compression factor allows predicted lengths to be adjusted to apply adequate pressure for residual limb re-shaping. Unstretched bandage lengths are multiplied by the compression factor to obtain extended lengths of bandage to be applied, exerting sufficient pressure for residual limb re-shaping and swelling control. Calculation of the compression factor is influenced by the ideal pressure application of 20 mmHg from literature. Influence of the ideal pressure requirement ensures that compression factor application does not result in predicted bandage lengths applying excessive or insufficient pressure. Predicted lengths should be sufficiently extended to influence residual limb shape without causing harm to the body of the residual limb or healing amputation scar. Application of the compression factor will also assist in applying a pressure gradient between distal and proximal portions of the residual limb to enforce residual limb coning.

4.2.1 Compression Factor

Calculation of the compression factor was achieved using a force sensitive resistor (FSR) to detect the ideal pressure of 20 mmHg, when stretching an elastic crepe bandage over it. The FSR changes resistive value when pressure or a load is applied to it. When no load is applied, the sensor will have a high resistance and will appear as an open circuit with infinite resistance. As more pressure is applied, resistance decreases. The FSR was connected to an Arduino Uno shield to convert force applied to the sensor into data represented as an analogue reading. To achieve an analogue reading corresponding to the applied pressure, the target pressure of 20 mmHg was first converted to 27.19 g/cm^2 and multiplied by 16 cm^2 (the active area of the FSR) to obtain the mass of the load to be applied to the FSR in grams. This resulted in a load of 435.04 g needed to be placed above the FSR to achieve a pressure reading equivalent to 20 mmHg. Positioning of the load above the FSR was done to determine an analogue reading that would reflect the ideal pressure application. The process was repeated 15 times to obtain an average analogue reading of 890. A total of 15 readings were recorded after obtaining similar readings in each test. A feedback circuit, as shown in Figure 21 and circuit diagram found in Appendix D.1, was designed to illuminate an LED when pressure applied to the FSR was greater than or equal to 890. This method was used to determine the maximum length of elastic bandage that could be stretched over the FSR until the circuit indicated an application of excessive pressure. A pressure pad was created by interpolating the FSR between a square foam equal in area to the FSR and rigid cardboard equal in width to the elastic bandage used (10.16 cm). Elastic bandage was stretched over the pad in equal length on both sides of the pad, and pulled downwards to determine the maximum stretched length that could be applied before the circuit indicated excessive pressure exertion. Readings from the feedback system were recorded after repeating the test four times for bandage lengths between 5 cm and 20 cm in increments of 5 cm. A maximum length of 20 cm was chosen for measurement accuracy and to accommodate testing rig dimensions.

Calculation of the compression factor involved dividing the stretched length of bandage by the measured unstretched length for all four increments. An average compression factor of 0.895 was achieved after recording the four measurements. Results for each increment is summarised in Table 1:

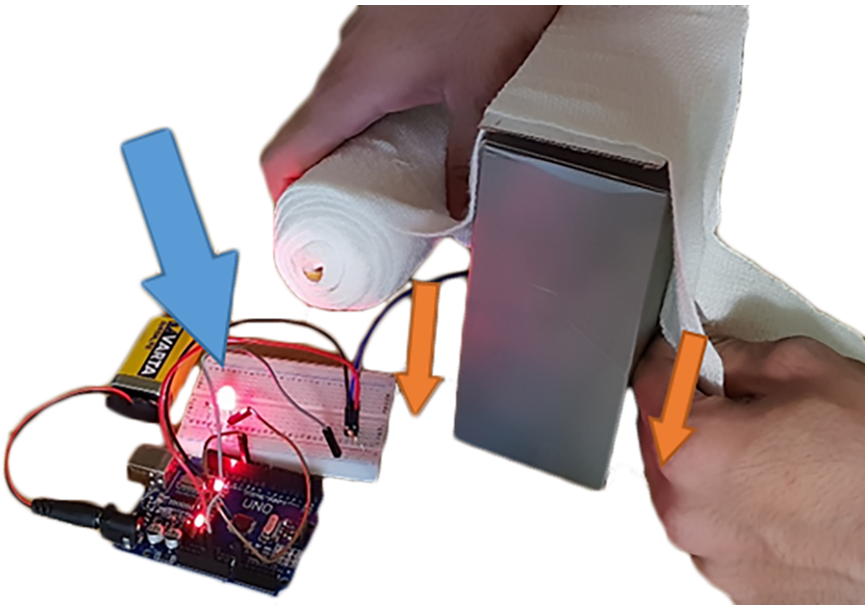


Figure 21: FSR feedback system

Table 1
Summarised results of compression factor calculation

	Unstretched Length (cm)	Stretched Length (cm)	Compression Factor	FSR Reading
	5.00	4.60	0.92	888
	10.00	8.80	0.88	892
	15.00	13.40	0.89	894
	20.00	17.80	0.89	887
Average			0.89	

A comparison was made between the calculated compression factor of 0.895, and a compression factor calculated from the recommended bandage tension advised by rehabilitation centres. Advised bandage tension is achieved by stretching required bandage lengths to approximately two-thirds of the maximum stretched length (Muilenburg & Wilson, 1996). Calculation of the recommended factor was achieved using Equation 9. Measurements were taken for unstretched lengths of bandage from 10 cm to 50 cm in increments of 10 cm. Bandage lengths were stretched to maximum length and then recorded to be used in Equation 9. Original, unstretched lengths were divided by two-thirds of the maximum stretched length (as recommended by rehabilitation centres) to determine the recommended compression factor. Summarised results for six increments are shown in Table 2.

$$Compression\ Factor = \frac{Unstretched\ Length}{\frac{2}{3}Maximum\ Stretched\ Length} \tag{9}$$

Table 2
Summarised results of recommended compression factor calculation

	Unstretched Length (cm)	Max Stretched Length (cm)	(2/3)Max Stretched Length (cm)	Recommended Compression Factor
	1	1.7	1.13	0.88
	10	16.4	10.93	0.91
	20	33.5	22.33	0.89
	30	49.1	32.70	0.92
	40	65.8	43.86	0.91
	50	84.6	56.37	0.89
Average				0.90

An average compression factor of 0.901 was achieved for six incremental measurements. A comparison between the compression factor of 0.895 calculated using the FSR feedback system yielded a difference of 0.006 between the two factors. A minimal difference between compression factors would not greatly influence extended bandage application length or pressure exertion applied to the residual limb. A comparison between the calculated compression factor and recommended factor was performed to validate usage of the calculated compression factor achieved through testing with the FSR feedback system. Incorporation of the compression factor into a second custom Python script allowed the prediction of extended bandage lengths.

4.3 Mobile Application

The developed Python scripts were incorporated into a mobile application, where user input in the form of circumferential measurements are entered through an interface. The decision to use a mobile application to convey bandaging guidelines and general residual limb care, stemmed from the rising number of smartphone users in South Africa. The estimated smartphone penetration in South Africa increased from 81.7% in 2018 to 91.2% in 2019, according to the Independent Communications Authority of South Africa's 2020 State of the ICT Sector report. In addition, the use of smartphone

features minimised the overall cost of the bandaging dispenser by reducing the number of required components. These features included alert systems that were programmed into the application using Android Studio, an integrated development environment (IDE) for the Android operating system. Development of the mobile application was done in the Android Studio environment due to experience with the Java programming language supported by the IDE.

4.3.1 User Interface and Design

Primary functionality of the mobile application termed, the Bandage Utility Dispenser (BUD) Integration app, is to provide an interactive way to assist amputees with residual limb bandaging. Assistance is provided by communicating and interpreting data from the dispenser and displaying relevant information on the application interface. The application is partitioned into six sub-menus that are navigated between using the application's dashboard or bottom navigation bar, as seen in Figure 22.

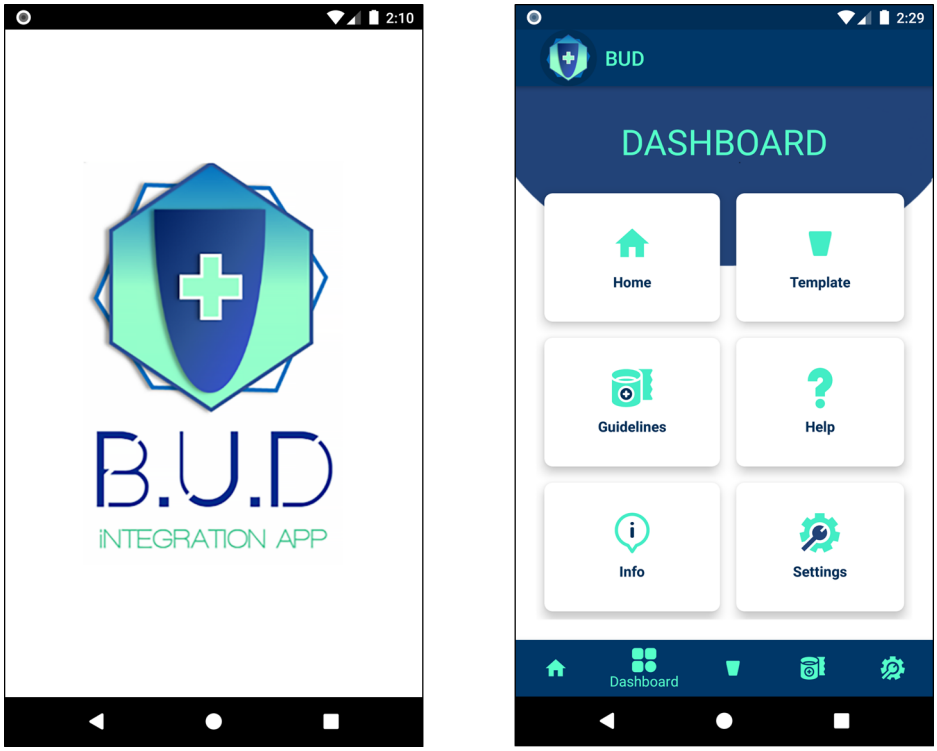


Figure 22: Dashboard and bottom navigation bar

Current instances of a sub-menu are stored while navigating between different sub-menus of the application for the purpose of retaining information inserted by the user. Mobile application sub-menus include the:

1. Home menu - An instructive menu containing a brief description about application functionality through text, as well as an introductory video explaining the application features and usage.

2. Template Generation menu - Generation of bandage templates and calculation of predicted bandage lengths occur within this sub-menu after user input in the form of circumferential measurements has been entered.

3. Bandaging guideline menu - Target bandage lengths to be used during each step of the bandaging process is displayed on this sub-menu's interface. An animation instructing the bandaging technique for each step of the bandaging process is displayed below target bandage lengths.

4. Help menu - Additional assistance on application operation and navigation, as well information on dispenser usage is found in the Help menu.

5. Information menu - Information regarding application version, features and user permissions needed for functionality is described in this sub-menu.

6. Settings menu - Adjustments to alert systems and Bluetooth integration with the dispenser are conducted in the Settings menu.

Each sub-menu is declared within the Main Activity class in Android Studio, with each sub-menu referred to as a fragment in the Android Studio IDE. A fragment is constructed by a combination of a Java class and an Extensible Markup Language (XML) layout. Each fragment acts as a single independent activity (single screen with a user interface) used to display subsequent interfaces within a single application. Each fragment is managed by the application's Main Activity class by initialising and declaring new instances of each fragment. Identifiers (IDs) are established for each fragment for the purpose of navigation. Tracking of each fragment using an ID, allows for saved instances of each fragment to be revisited and makes navigation to previously visited fragments possible. An array is used to track the order of IDs during navigation for the purpose of determining which corresponding fragment to display.

4.3.2 Template Generation Fragment

Generation of bandaging templates and calculation of predicted bandage lengths occur at the template generation fragment by incorporating user input into each Python script. The fragment contains an introductory text field, explaining input requirements

needed to generate a bandaging template. These requirements include distal and proximal circumferential residual limb measurements, indicated by an image displayed on the fragment's interface. Required measurements are inputted into the Measurement 1 and Measurement 2 editable text fields, as seen in Figure 23, by the mobile application user. Template generation occurs after completing required measurement fields and clicking the Generate button to generate a bandaging template using Python Script 1. Utilisation of Python scripts in a Java application programming interface (API) is possible by using a Python Software Development Kit (SDK) known as Chaquopy, that has been integrated into the mobile application. The SDK allows for a Python interpreter to be used in a Java API (Smith, 2017). The interpreter is able to run and execute Python code on command, and returns output to the Java API. Incorporation of the SDK allows for previously written Python scripts and exclusive Python libraries to be used within Java written programs (Smith, 2017). Libraries used for mathematical plotting such as Matplotlib are accessible to generate a bandaging template on the Android mobile application. A Java method running the Python interpreter is called when interacting with the Generate button. User input in the form of circumferential measurements from the Measurement 1 and Measurement 2 text fields are passed into Python Script 1 by a method. The interpreter runs and executes the script's code and returns a generated bandaging template displayed on the fragment interface, demonstrated in Figure 23 .

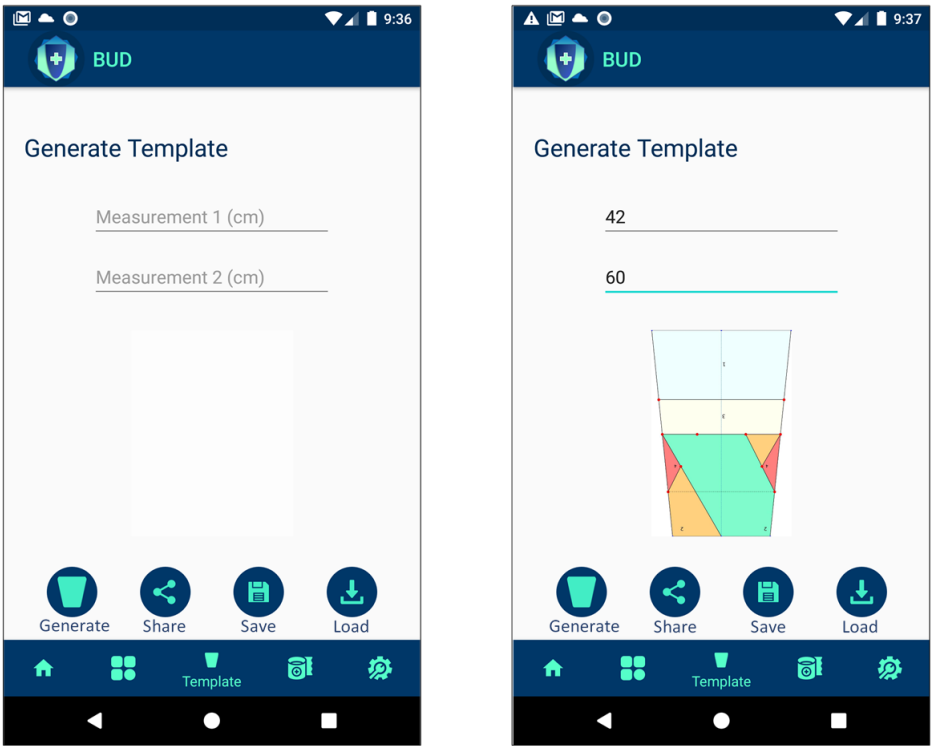


Figure 23: Template generation interface

Sharing of the generated template is possible by using the Share button, allowing the

distribution of the template to social media applications, email or online storage drives. The Share button converts the template from a Joint Photographic Experts Group (JPEG) image format to a Portable Document Format (PDF) format for scaling and printing purposes. Storage of the template as an image within the smartphone's gallery is possible by clicking the Save button after storage permission has been granted by the user. Previously saved circumferential measurements can be loaded using the Load button after storage permission has been granted. Stored measurements are saved within two text files termed "Current Measurements" and "Stored Measurements", both generated at the first instance of template generation. The Current Measurements text file stores most recent measurements entered into the application for template generation and bandage length calculations, and is accessed when the Load button is clicked. The Stored Measurements text file stores new measurements entered by the user to build a data set of residual limb measurements specific to the user/amputee. Every iteration of circumferential measurements are stored in individual arrays within the text file. Stored data can be used to monitor residual limb maturation through changes in circumferential measurements.

A scrollable layout is included in the bandaging template fragment interface, allowing bandage length requirements to be calculated using the same interface, preventing the need to input measurements on multiple interfaces. A text field describing the three additional circumferential measurements can be seen when scrolling further down the bandaging template interface, demonstrated by Figure 24, followed by three editable text fields: Measurement 3, Measurement 4 and Measurement 5.

After measuring and inputting each measurement, the Calculate button is clicked to determine required bandage lengths required for each step of the bandaging process. Python script 2 is accessed within a Java method that opens a Python interpreter of the Chaquopy SDK, and is passed the five required circumferential measurements. Predicted bandage lengths are returned and appended to established arrays within the previously defined text files: Current Measurements and Stored Measurements. Storage of circumferential measurements in text files prevents the need to re-insert measurements when re-generating bandaging templates or re-calculating predicted bandage lengths for current residual limb dimensions. Previously stored measurements can be loaded from text files by clicking on the Load button.

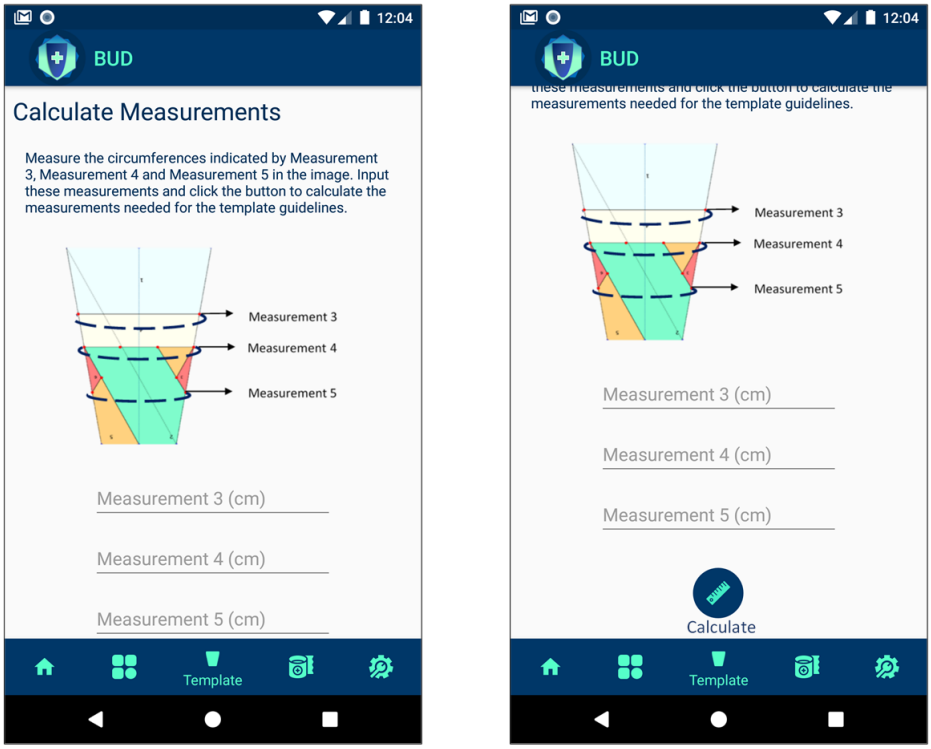


Figure 24: Bandage length calculation interface

4.3.3 Bandaging Guidelines Fragment

Navigation to the Bandaging Guidelines fragment can be accomplished by directly interacting with the application's bottom navigation bar or the application's dashboard. The interface is partitioned into three sections to guide the amputee when bandaging their residual limb. Interaction between the mobile application and BUD bandaging dispenser is controlled by the first partition. Data is sent and received from the dispenser to the application using an ESP32 microcontroller. Bluetooth is used to transfer data across devices in the form of bytes. The dispenser is assigned a device address that can be recognised when pairing the smartphone with the dispenser using Bluetooth. Pairing is established after initiating the connection within the smartphone's settings interface. A unique pairing code will be sent to the smartphone from the dispenser to validate the correct device to be paired with. The pairing and validation process occurs once, allowing the dispenser to communicate with the smartphone during future uses of the mobile application. Communication between devices is established after pairing has been completed.

Clicking the Bluetooth button on the application's interface opens a communication channel between the two devices for the transferal of data. Data is received by the mobile application by using an *InputStream* constructor that receives data in the form

of raw bytes. Data is converted into a string format to be displayed on the mobile application's interface. Unrolled bandage length measurements are transmitted to the mobile application as raw data to be converted into a string format and displayed on the interface as an integer, as seen in Figure 25. Current unrolled bandage length is displayed by an updating figure on the left hand side of the arrow in the first partition of the interface. The predicted bandage length required for the current step of bandaging, is displayed on the right hand side of the arrow. Predicted bandage lengths are displayed on the fragment interface after the Bluetooth button is clicked. Display of the current predicted bandaging length is achieved by sorting through the array present in the Current Measurements text file and retrieving the measurement corresponding to the current bandaging step represented by a counter.

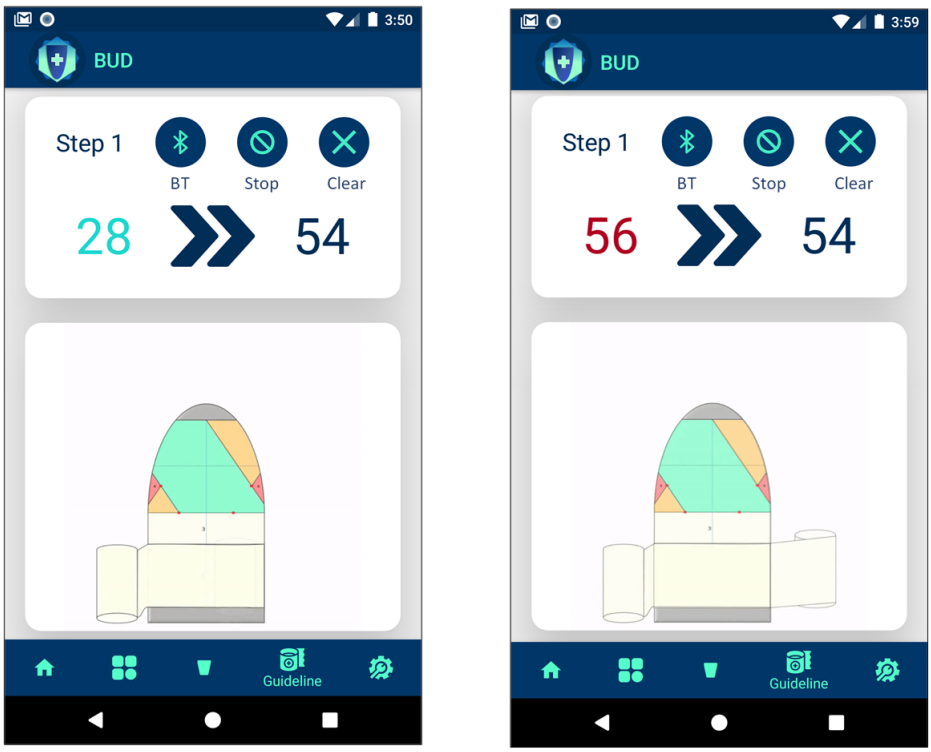


Figure 25: Bandaging guideline interface

The figure indicating the current length of unrolled bandage is presented in a green font and is updated as an elastic bandage is unrolled from the dispenser. Conditional statements have been implemented to detect when unrolled bandage length exceeds the target length for the current step of the bandaging process. Exceeded bandage lengths will be converted to a red font, as shown in Figure 25, while the smartphone emits an audible alert to indicate excess bandage length unrolled for the current bandaging step.

Remaining fields of the first partition include a step count to indicate the current bandaging step, and two buttons to remove the bandage length figures from the interface. Bluetooth connection between the dispenser and mobile application can be terminated by clicking the Stop button. Current figures present in the target and current bandage length fields are cleared by clicking the Clear button, while still maintaining Bluetooth communication with the dispenser.

The second partition of the interface contains an animated Graphics Interchange Format (GIF), derived from images seen in Figure 12. Each GIF corresponds to the respective bandaging step, and updates as each step progresses. GIF animations are updated using conditional statements within the fragment's Java class. Updating the current bandaging step results in a change to a step counter used to determine the current bandaging step, the target bandage length and the animated GIF. Each indicator used to conduct bandaging is updated according to the current bandaging step ranging from 1 to 4. The final partition of the fragment's interface, as seen in Figure 26, contains two buttons dedicated to navigating to previous steps of the bandaging process. The Back button navigates to the previous step by updating the animated GIF, step counter and target bandage length using conditional statements. The Reset button clears and resets the current length of bandage displayed on the fragment interface received by the dispenser. Both buttons rely on feedback from the dispenser to perform assigned tasks. Feedback is received from the dispenser's microcontroller through a Bluetooth input stream.

4.3.4 Alert Dialogues

Alert dialogues illustrated in Figure 27 (a), have been implemented to guide the application user when incorrect usage of the mobile application occurs. Alerts appear for a period of 2 seconds at the bottom of the interface. Alert dialogues indicate when an insufficient number of measurements have been entered when attempting to generate a template or calculate predicted bandage lengths, as well as used to display Bluetooth connection status. Additional dialogues indicate successful Bluetooth pairing with the bandaging dispenser, as well as Bluetooth disconnection. Additional functions of the

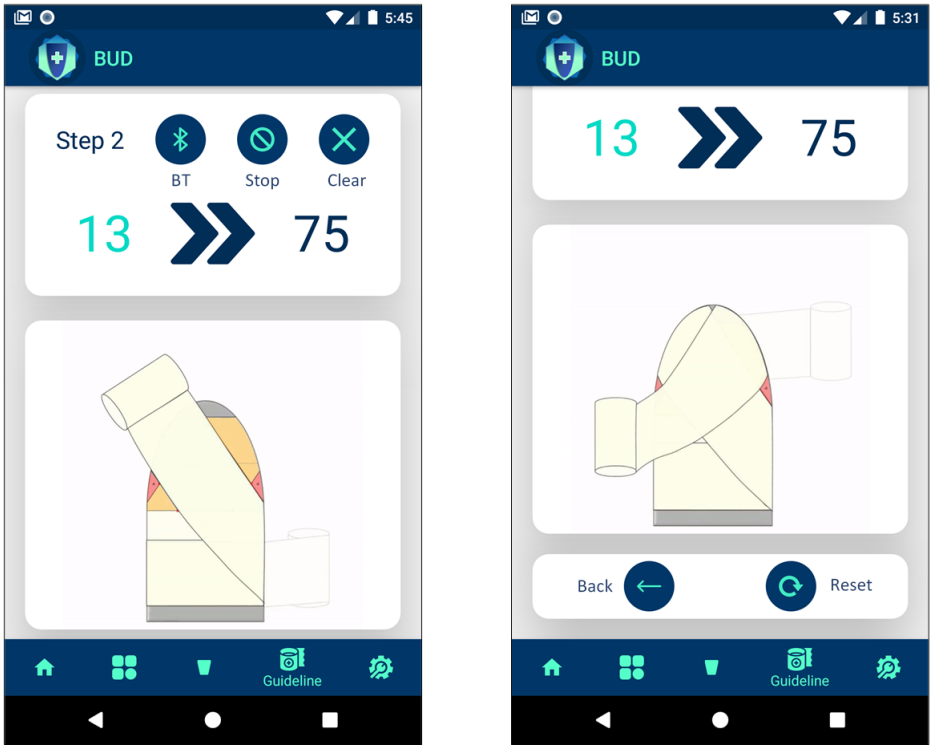
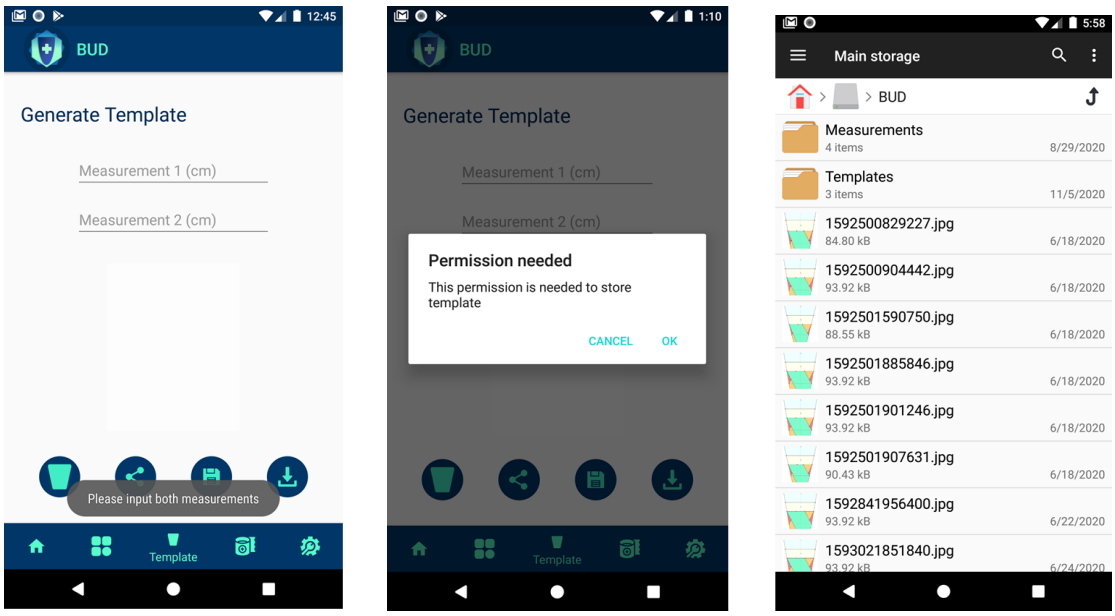


Figure 26: Guideline animation and additional interface buttons

alert dialogues include assistance when performing bandaging. This includes specifying when the next bandaging step should be performed, indicating when an excessive bandage length has been unrolled by the dispenser for the current bandaging step and when bandaging of the residual limb has been completed. Alert dialogues are continuously implemented to enforce proper application use and provide user assistance.

4.3.5 Storage Permission Alerts

Upon attempting to save the initial template generated after installing the application, the user will be prompted with an interactive text dialogue to accept or deny application storage permission. Rejection of storage permission prompts an additional text field as shown in Figure 27 (b), to highlight functionality requiring storage permission. Storage permission is granted during a single instance, allowing the application to store generated templates and residual limb circumferential measurements. Files produced by the mobile application are stored in a generated folder termed "BUD", on the smartphone's internal storage. Generated templates will be stored as images in a folder named "Templates", with circumferential measurements and predicted bandage length measurements stored as text files within a folder named "Measurements". Folders containing these files can be accessed using the smartphone's file manager, as shown in Figure 27 (c), at any instance for extraction or review purposes.



(a) (b) (c)

Figure 27: Alert dialogues and storage of application files

4.4 Bandaging Utility Dispenser (BUD)

Unrolled lengths of elastic crepe bandage are measured by the BUD dispenser, while wrapping the residual limb. Electrical components needed to record and convey bandage length measurements to the mobile application include an ESP32 microcontroller and a rotary encoder. The encasing holding these components has been 3D modelled for the purpose of 3D printing. Component functionality, code and encasing design are discussed in further detail within this sub-section.

4.4.1 Microcontroller

An ESP32 microcontroller, illustrated in Figure 28, was used to measure and transmit bandage length measurements to the mobile application. An ESP32 was chosen due to Bluetooth capability with mobile phones, low power consumption and compatibility with a rotary encoder. A Bluetooth link controller is present on the microcontroller, operating in three primary sates: standby, connection and sniff (a power-saving mode that transmits data in set intervals) (Espressif, 2020). This link controller enables the microcontroller to connect to multiple devices, be discoverable by devices and establish Bluetooth connections. The Bluetooth link controller interfaces with the BUD Integration Application to establish a Bluetooth connection for the purpose of

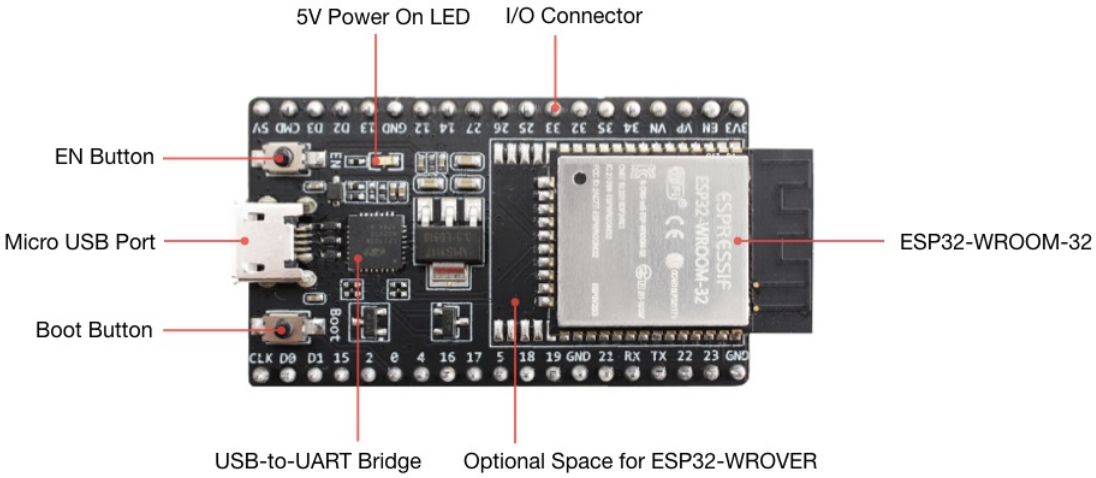


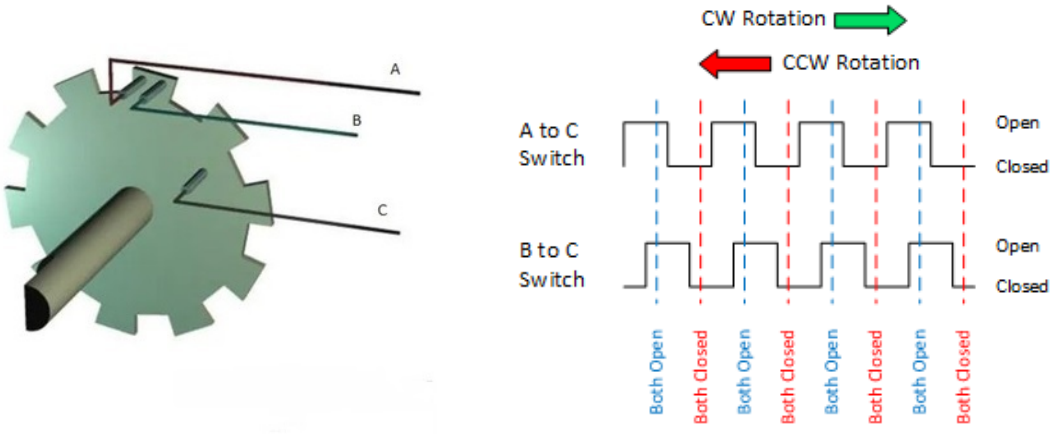
Figure 28: ESP32 microcontroller (Espressif, 2020)

transmitting data. Input pins required to integrate with the rotary encoder are present to receive readings from the rotary encoder. Readings are transmitted using Bluetooth to the mobile application, as instructed by code uploaded to the microcontroller using the Arduino IDE.

4.4.2 Rotary Encoder

A rotary encoder is a rotary input device that indicates angular movement and direction of a dial attached to the rotary encoder. A total of twenty pulses or increments occur in one full 360° revolution of the dial. Increments are contained in fixed positions and can be felt when rotating the dial, allowing for the recording of incremental changes to dial position in the form of digital pulses by the ESP32 microcontroller. Attachment of the rotary encoder to a rotating mechanism onto which an elastic crepe bandage roll is applied, allows for the measurement of unrolled bandage lengths by tracking the number of incremental revolutions per length of unrolled bandage.

Output for the incremental rotary encoder is interpreted as a series of square wave form pulses, as illustrated in Figure 29 (b). These pulses or signals are generated by three contact pins A, B and C. Contact pins A and B are fixed in place and come into contact with the rotating pin C. Inside each encoder, two switches are present to indicate rotation direction. Switch one connects pin A to pin C, while switch two connects pin B to pin C. Each switch is opened or closed during rotation, depending on each encoder increment. If both switches are open, rotation of the dial results in both switches closing, while the inverse is true if both switches are closed. Rotation direction is determined by the



(a) (b)

Figure 29: Rotary encoder direction and signal output (Handson Technology, 2017)

angular position of pins A and B, as illustrated in Figure 29 (a). Rotation of the switch in a clockwise direction, first results in the change of state of the switch connecting A and C, while rotation in a counterclockwise direction will result in the switch connecting B and C to change states first. A visual representation of the opening and closing of switches A and B as square wave forms can be seen in Figure 29 (b).

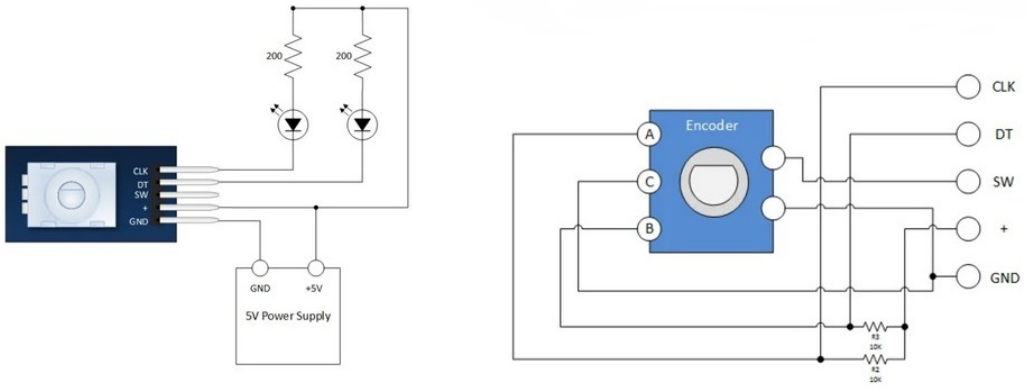


Figure 30: KY-040 module and output pins (Handson Technology, 2017)

A KY-040 module, illustrated in Figure 30, was used to simplify connection to the ESP32 microcontroller, by using four input pins instead of sixteen, and incorporating built-in pull-up resistors. The module operates by outputting a low voltage when switches are closed, and a high voltage when switches are open (Handson Technology, 2017). Low voltage output is generated by placing a ground connection to pin C and passing it to the Clock (CLK) and Digital (DT) pins when switches are in a closed state. High voltage output is generated by a 5 Volt supply input and pull-up resistors, resulting in a high voltage output for the CLK and DT pins when switches are open. Changes to switch states are indicated by the Switch (SW) pin, used to indicate when the rotary encoder dial button is pushed down. Interaction with the button results in the closing of the open switch, with a signal being sent to indicate this change. Incorporation of this switch mechanism was used to send a string message to the mobile application to progress the current bandaging step. Reception of the message in string format by the mobile application was used to update the current step counter, target bandage length and bandaging animation to the following step of the bandaging process.

4.4.3 Microcontroller Code

The open-source Arduino Software integrated development environment (IDE) was used to program the ESP32 microcontroller to receive and transmit data to the mobile application using Bluetooth. Data to be transferred included bandage length measurements recorded by the rotary encoder, as well as unique key strings used to execute specific Java methods present in mobile application classes. The script installed on the controller can be seen in Appendix C.3. Three libraries were imported for the following functionality:

1. The standard *Arduino* library enables code written within the Arduino IDE to be installed on the ESP32 microcontroller.

2. Bluetooth functionality and communication is established by the *BluetoothSerial* library. The library opens a Bluetooth serial to send and receive messages in the form of bytes for interpretation.

3. Rotary encoder support is achieved by importing the *AiEsp32RotaryEncoder* library. Imported methods are used to measure unrolled bandage lengths through rotation of the rotary encoder and to indicate when the rotary encoder dial button has been pressed.

The *setup* method initialises variables, pin modes and enables the start of library usage. Initialisation of the Bluetooth serial and rotary encoder methods occurs within the script's setup method. Bluetooth serial communication opens a channel between the mobile application and ESP32 microcontroller to receive feedback or transmit information obtained from rotary encoder readings. Channel identification occurs within the smartphone's Bluetooth settings, and is termed "BUD" for ease of location.

After initialisation, the *loop* method continuously runs the script and updates required readings from the rotary encoder until connection between the microcontroller and mobile application is terminated. Readings and messages are sent and received through the IDE serial in addition to the Bluetooth serial monitor. An example of this interaction is the function of the string message, "reset", received from the mobile application, as seen in the final conditional statement of the *loop* method. Current bandage length value and step counter value are reset to zero when this conditional statement calls the *reset* method of the rotary encoder library.

Following initialisation and connection establishment between the rotary encoder and mobile application, the *rotary_Button* and *rotary_loop* methods are used to indicate changes in rotary encoder readings. The *rotary_Button* method sends the string message ">>", when pressing the rotary dial button, signaling a change to update the current bandaging step to the mobile application. Execution of the *rotary_Button* method only occurs when the encoder dial is pressed, while the *rotary_loop* method is called whenever the encoder dial is rotated. Encoder readings are continuously transmitted to the mobile application, where readings are converted from raw bytes to strings. Encoder values are divided by 2, since each incremental encoder rotation sends two digital readings per increment.

4.5 BUD Dispenser

Design of the BUD dispenser enabled the ability to unroll an elastic bandage and store components needed to communicate with the BUD Integration mobile application. Design specifications were influenced by the dimensions of a 4-inch elastic crepe bandage and the electrical components housed within the dispenser. Placement of these components influenced design and overall size, with the objective to minimise total size. Achieving a minimally sized dispenser allows for easy maneuverability while bandaging the residual limb. This would not only ease the bandage application process, but allow for single-handed use, enabling simultaneous use of the dispenser and mobile application to complete residual limb bandaging. Modelling and dispenser design is detailed within this sub-section.

4.5.1 Encasing Body

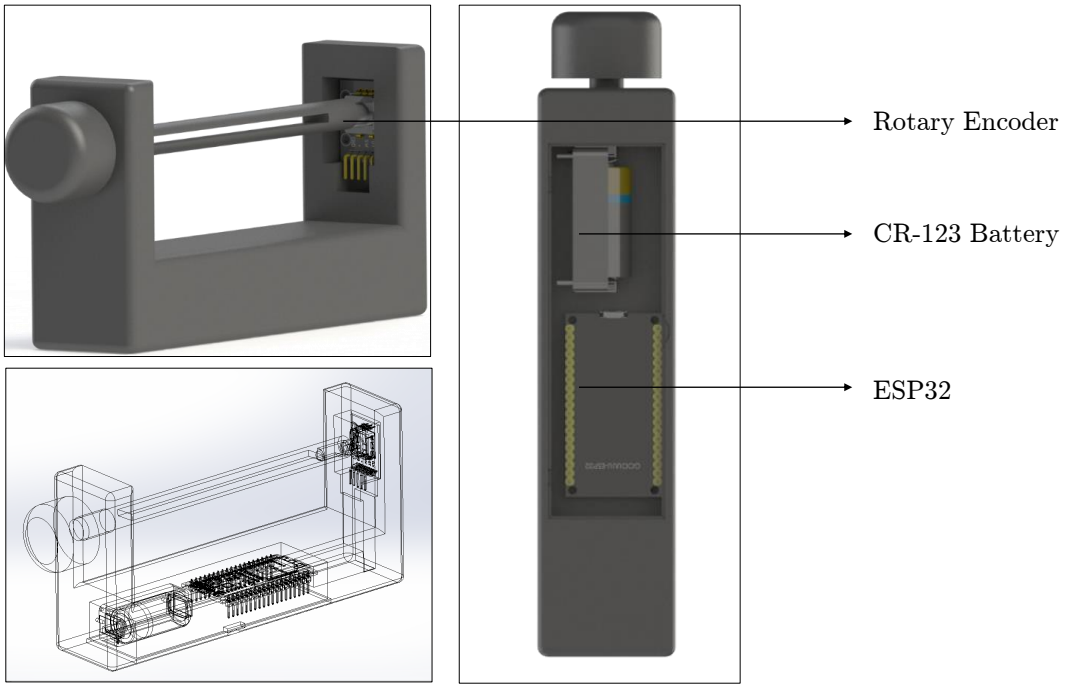


Figure 31: BUD encasing

The encasing, illustrated in Figure 31, contains compartments to hold the rotary encoder, ESP32 microcontroller and a CR-123 battery cell. Placement of the rotary encoder occurred within a compartment situated in one of the encasing pillars. The encoder is fixed in this position to allow the attachment of a rotary rod to the encoder, allowing for the detachment of the rotary rod for the purpose of fixing an elastic crepe bandage

to the rod. This mechanism permits the rotation of the rod, while rotating the dial of the rotary encoder for the purpose of measuring unrolled bandage lengths. A hollowed compartment at the base of the encasing was modelled to hold the CR-123 battery and ESP32 microcontroller. A hollow channel is present between this compartment and the compartment holding the rotary encoder situated on the encasing pillar. Wires connecting the rotary encoder and microcontroller are placed in this channel to be secured within the encasing.

An open area between the two pillars exists to place an elastic crepe bandage and to minimise physical contact between the dispenser and the residual limb. The design decision allows the bandage to be applied directly onto the residual limb, preventing contact between the dispenser and residual limb for sanitary and safety purposes. Overall dispenser design features rounded corners and angles for safety and to prevent injuries from occurring while bandaging the residual limb. Dispenser design maximises elastic bandage exposure, allowing for the ability to directly apply elastic bandage to the residual limb and control bandage tension by hand.

4.5.2 Rotary Rod

Figure 32: Rotary Rod

Suspension of a rod, illustrated in Figure 32, between the two encasing pillars allows for rotation of the rod to unroll incorporated elastic bandages from the dispenser encasing. A thin space is present in the centre of the rod, where elastic bandage is fed through to secure the bandage onto the rod for dispensing purposes and allow the dispenser to incorporate new bandage rolls or a previously used one. New bandage rolls are re-rolled onto the dispenser rod due to their initial compressed state. New bandage rolls are packaged in a compressed form, making bandage length measurements inaccurate as the bandage is unrolled. Unrolling the bandage from a compressed state will cause the unrolled length of bandage to relax, decreasing the measurement of unrolled bandage lengths as a result.

5 Testing Methodology

5.1 Introduction

This chapter outlines the testing methodology used to validate the design and performance of the overall solution method comprising of the bandaging template, mobile application and BUD dispenser. Testing of individual components for each sub-system was first performed before evaluating overall performance.

5.2 Sub-System Validation

Individual sub-system component testing was first performed to validate functionality of each sub-system before evaluation of the overall solution method could occur. Sub-system validation involved:

1. Generating bandaging templates for multiple residual limb models to test template adaptability to changing circumferential measurements.
2. Verifying the compatibility of the mobile application with varying Android versions and screen sizes.
3. Validating the accuracy of the displayed length of bandage unrolled, as measured by the BUD dispenser.
4. Comparing expected bandage length requirements calculated by Python Script 2 with the observed bandage length measurements achieved through physical measuring using a standard measuring tape.
5. Comparing observed pressure readings with the ideal pressure application from literature needed to control swelling of the residual limb.

Upon completion of sub-system testing, two statistical analysis tests were done to compare observed and expected results of bandage length measurements and pressure application for six residual limb models.

5.3 Residual Limb Models

To perform template and bandage application testing, five residual limb models were constructed to simulate the dimensions of an anatomically correct transfemoral residual limb model (ACM), acquired from Groote Schuur Hospital. The ACM served as the baseline model, with five constructed models (CM) simulating ACM dimensions. Each CM differed in circumferential measurements, achieved by increasing or decreasing model proximal and distal circumferential measurements by increments of 5 cm. Incremental adjustments of 5 cm to the ACM measurements resulted in an 8.3% change in proximal circumferential measurements and a 12.5% change in distal circumferential measurements. An incremental change of 5 cm was chosen to observe noticeable differences in volume when generating bandaging templates and predicting bandage length requirements. A total of five models were constructed to simulate adjusted measurements of a residual limb with the proportions of the ACM.

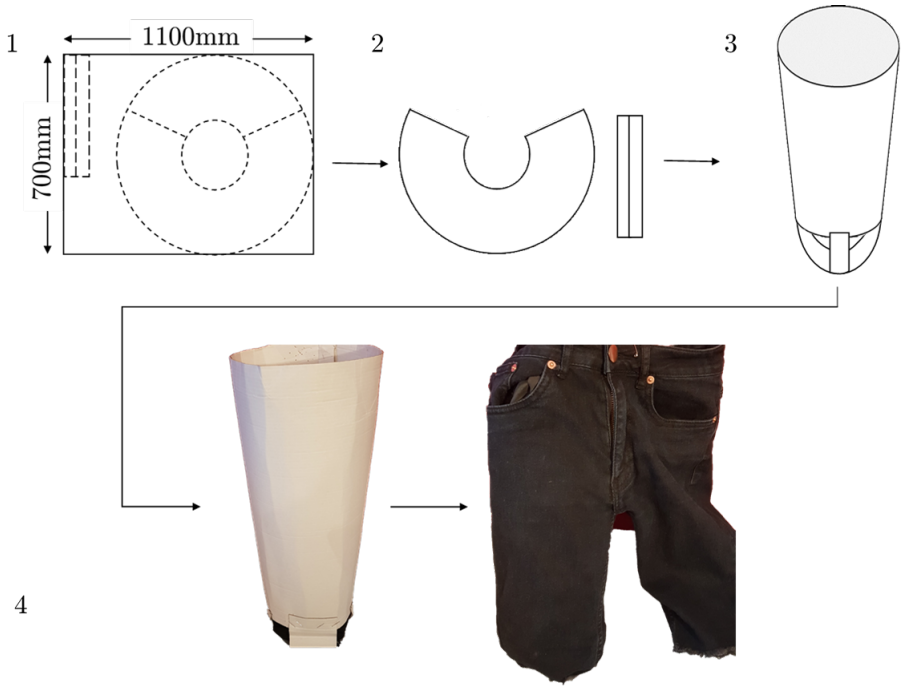


Figure 33: Construction of residual limb model

Constructed models used the truncated cone pattern formulae outlined in Sub-section 4.1.3 to simulate the shape of a residual limb. The distal portion of a residual limb was simulated using the arc length of a segment formulae in Sub-section 4.1.3. The summarised construction of a residual limb model is described in Figure 33. The 2D pattern was drawn on a 700 mm × 1100 mm cardboard and cut out to reveal the truncated cone pattern, as illustrated in steps 1 and 2 of Figure 33. A 3D truncated cone was achieved by bending the extracted cone pattern to form a conical shape, and

was maintained using tape. Simulated arcs were produced from the remaining piece of board and were attached to the truncated cone's body, as seen in step 3 and 4 of Figure 33. Rigidity of the constructed model was achieved by placing a plastic container in the centre of the truncated cone to fix the model's shape. Maintenance of the desired residual limb shape was established by placing the constructed model in a closed off pair of trousers to simulate the tapered outline of the residual limb. The final constructed model can be seen in the fourth and final step of Figure 33.

5.4 Pressure Testing

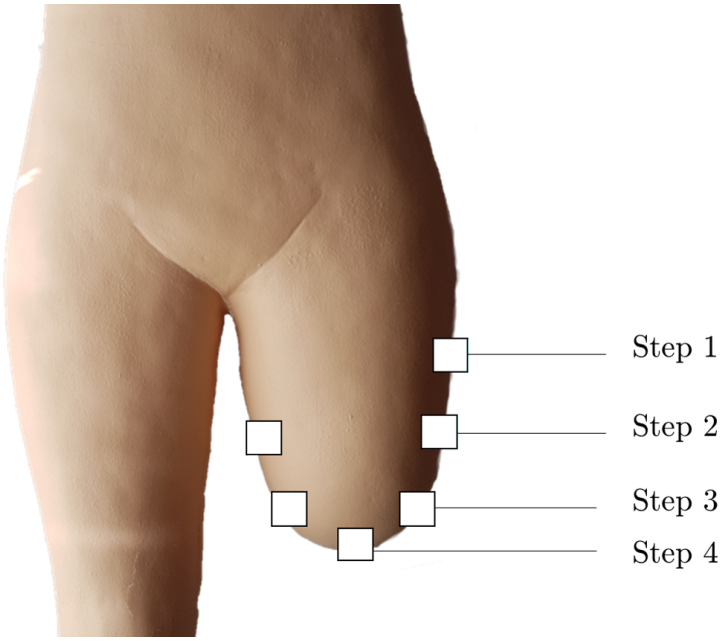


Figure 34: Pressure pad placement on ACM

Pressure testing was accomplished by attaching assembled pressure pads using FSRs at the edges of the distal portion of the residual limb, the perimeter of the residual limb and by measuring pressure application as bandaging occurs. Diagonal bandaging is prominent in the areas marked in Figure 34, where constructed pressure pads were placed. Pressure readings near these areas required recording to determine if predicted bandage lengths were sufficiently tense enough to influence residual limb volume. Assembled pressure pads utilised a FSR interpolated between a piece of cardboard equal to the width of the elastic bandage (10 cm) and a foam square equal in area (16 cm^2) to the FSR. Cardboard material was used to maintain rigidity of the pressure pad when bandaging each residual limb model, and foam was used due to compression capabilities, allowing for the recording of pressure application. An assembled pressure pad can be seen in Figure 35.

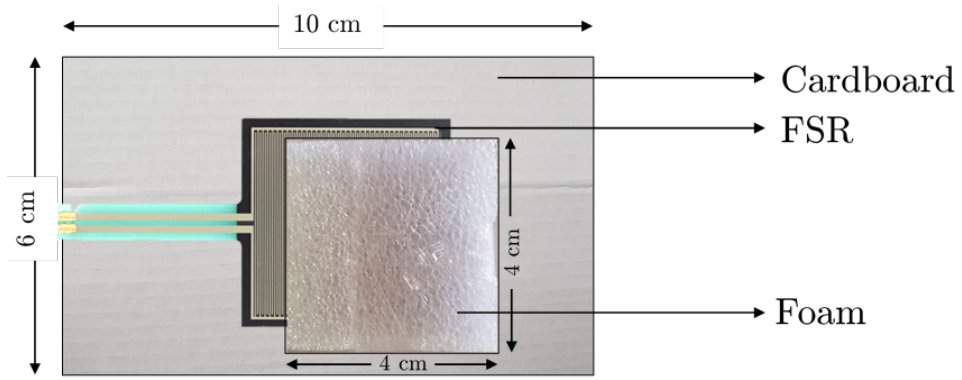


Figure 35: Assembled Pressure pad

FSR readings were monitored using an Arduino Uno microcontroller and displayed on a 16×2 character LCD screen. To reduce required input pin usage, an I2C backpack was used. The attachment reduces the required pins needed for connectivity from 16 pins to 4 pins. The pressure pad circuit diagram can be seen in Appendix D.2. Pressure testing was performed on the ACM and CMs to record elastic crepe bandage pressure application readings for comparison with the ideal pressure application of 20 mmHg from literature.

5.4.1 Pressure Testing Data Analysis

A statistical analysis in the form of a one sample t-test was used to test whether the mean pressure reading (sample mean) for each residual limb model differed significantly from the hypothesised pressure reading value (test value) of 20 mmHg. The proposed null hypothesis and alternative hypothesis for the one-sample t-test is defined as:

> $\boldsymbol{H_0}$: The sample mean is equal to the proposed mean pressure application of 20 mmHg.
>
> $\boldsymbol{H_a}$: The sample mean is not equal to the proposed mean pressure application of 20 mmHg.

The calculated critical value was compared with the significance level of 0.05 for a 95% confidence interval to determine whether to reject or accept the null hypothesis. The test statistic was manually calculated in Microsoft Excel using the following formula:

$$t = \frac{\overline{x} - \mu}{s_{\overline{x}}} \tag{10}$$

where:

μ = Hypothesised constant for the population mean

$\overline{x}$ = Sample mean

$s_{\overline{x}}$ = Estimated standard error of mean

Acceptance of the null hypothesis would validate pressure application achieved by applying predicted bandage lengths to each residual limb model. Readings provide an indication on whether adequate pressure was applied to influence residual limb shape by comparing the observed pressure reading with the desired pressure application from literature.

5.5 Constructed Model Validation

A Bland-Altman analysis and a correlation analysis was performed to validate usage of constructed models for pressure and measurement testing respectively. This was performed on the ACM and a CM with equal circumferential measurements. The analysis was used to compare pressure readings and observed measurements for the ACM and CM, visually represented using a Bland-Altman plot. Representing data on a Bland-Altman plot indicated whether differences in pressure readings between the two models lied within acceptable chosen limits of agreement, permitting the use of additional CMs for further testing as a result. The use of the Bland-Altman plot provided a graphical method to observe anomalies existing outside limits of agreement (Kalra et al., 2017). The x-axis represents the mean of two measurements, whereas the y-axis represents the difference between two measurements. In Cartesian coordinates, where a sample S is chosen, each point is established as:

$$S(x, y) = \left(\frac{S_1 + S_2}{2}, S_1 - S_2 \right) \tag{11}$$

Figure 36 illustrates an example of a Bland-Altman plot for the distribution of points for two measurement methods, A and B, as well as lower and upper limits of agreement (LOA). A standardised confidence interval for the LOAs for each comparison is 95%, indicating that 95% of differences will be between mean difference $\pm$ 1.96 $\times$ standard deviation (SD) and indicates how far apart measurements for methods A and B are likely

to occur. Differences occurring outside the LOAs indicate that no agreement between the measurement methods could be reached for outlying points, resulting in a true change between measurements. Differences within mean difference $\pm$ 1.96 $\times$ SD indicate a low significance, allowing the two methods of measurement to be interchangeable (Kalra et al., 2017). These criteria were used to validate further constructed model usage for measurement and pressure testing, after establishing interchangeability between the ACM and a CM equivalent in measurements.

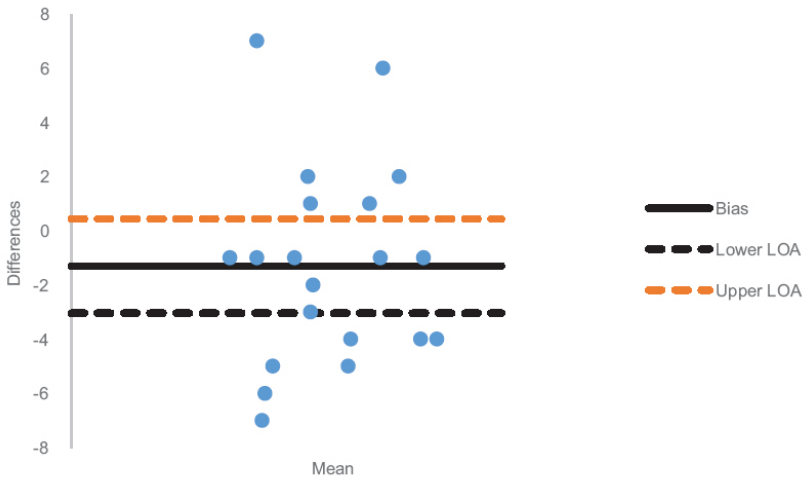


Figure 36: Bland-Altman plot example (Kalra et al., 2017)

5.6 Predicted Bandage Length Measurement Testing

A statistical analysis in the form of a chi-square goodness of fit test is performed to measure the accuracy of the correlation between observed bandage length measurements and expected (predicted) bandage length measurements when wrapping each residual limb model. The analysis is used to determine the magnitude of the difference between an observed phenomena and the expected value. The term "goodness of fit", refers to the comparison of the observed sample distribution with the expected probability distribution (Habib, Thomas, et al., 1986). Test results therefore indicate how well the theoretical distribution fits the empirical distribution, by manually computing and analysing the chi-square test statistic in Microsoft Excel:

$$X^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i} \tag{12}$$

where:

X^2 = Chi-square test statistic

O = Observed value

E = Expected Value

k = The number of mutually exclusive classes (for $i = 1, 2, ..., k$)

The value of the chi-square test statistic along with a significance value of 0.05 for a 95% confidence interval, is used to determine whether expected frequencies differ greatly from observed frequencies. The analysis indicates if the null hypothesis is rejected or accepted. Null hypothesis and alternative hypothesis definitions are defined as:

$\boldsymbol{H_0}$: The observed residual limb measurements do not differ significantly from the expected predicted measurements.

$\boldsymbol{H_a}$: The observed residual limb measurements differ significantly from the expected predicted measurements.

The statistical test was used to validate predicted bandage length measurements generated by Python script 2 for each residual limb model. Testing methodology for each sub-system and the overall solution method is summarised in Figure 37.

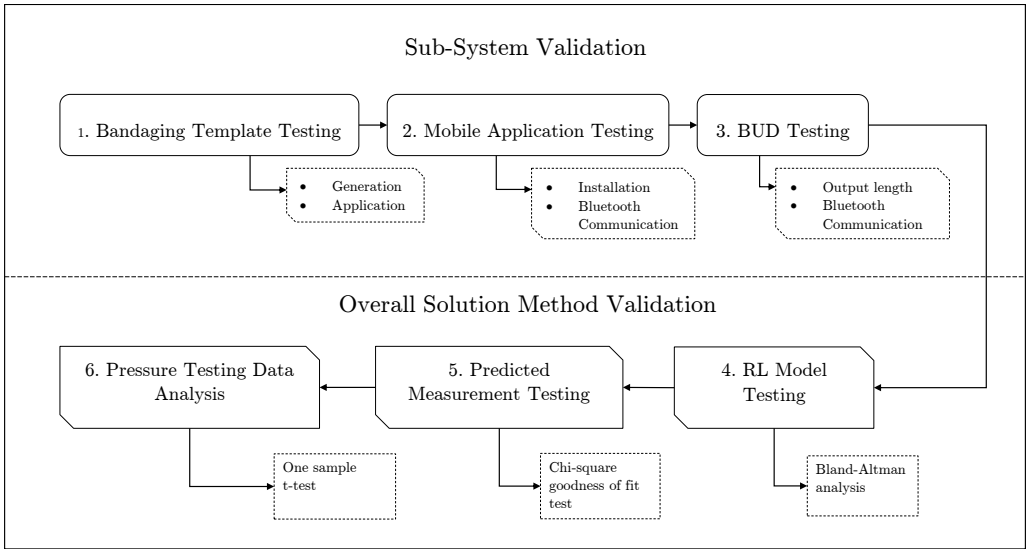


Figure 37: Summary of testing methodology

6 Testing Outcome

6.1 Introduction

This chapter discusses the results obtained after performing design validation and statistical analysis on individual system components and the overall solution method. Testing focuses on successful utilisation of the solution method to bandage a residual limb model, while evaluating the accuracy of predicted bandage length measurements and observed pressure readings. Each sub-system was therefore evaluated independently, with validation criteria unique to the specific sub-system. The bandaging template, Python scripts and BUD dispenser were evaluated by testing the correlation between observed and expected bandage length outcomes. Pressure readings were statistically compared to a predicted test-value to determine if a significant difference between pressure values occurred. The mobile application was evaluated based on performance and successful communication with the BUD dispenser, and will therefore not contain any quantitative testing. The entire testing methodology is summarised in Figure 37, and outlines the sections to be covered in this chapter.

6.2 Constructed Model Validation

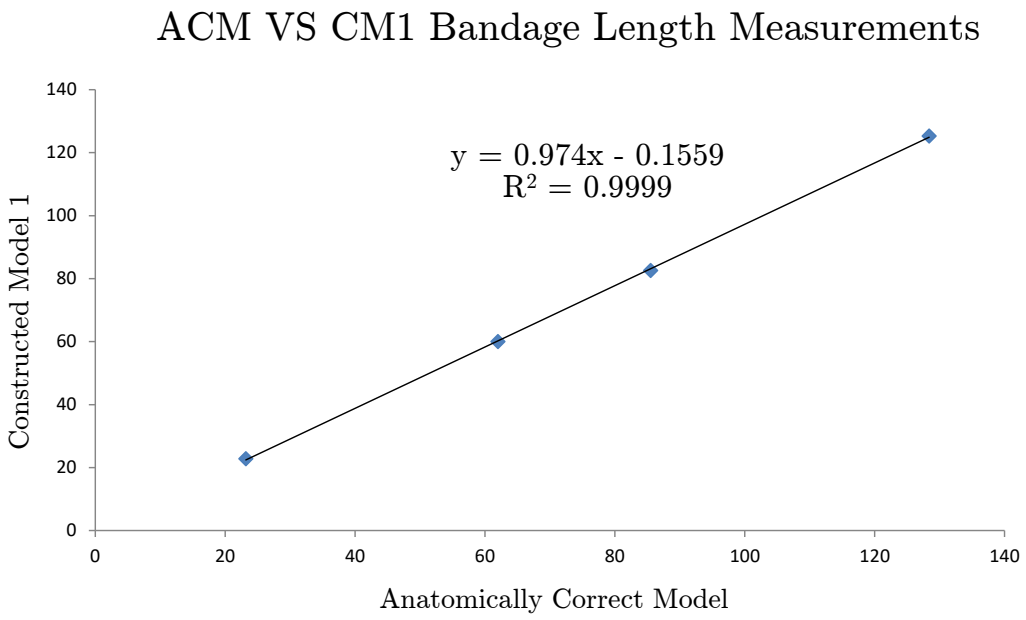


Figure 38: Linear correlation between ACM and CM observed measurements

Validation of the usage of the constructed models (CM) involved the use of a Bland-Altman analysis and plot. Analysis results and plots were used to indicate the significance of the differences between pressure readings and bandage length measurements for each model. A clinically significant correlation enabled the usage of additional constructed models that differed in size from the ACM. A linear correlation graph is seen in Figure 38 to compare the observed measurements of bandage application for the four steps of the bandaging process between the ACM and CM. Plotting of the linear correlation graph for the observed measurements allowed for the deduction of a positively linear graph. This was further validated by the R^2 value of 0.99 and a corresponding p-value of < 0.001, indicating that measurements were close to the trend line and that correlation was significant. The R^2 value measures the proportion of variation experienced by the dependent variable that can be explained by the independent variable. A high R^2 value therefore indicated that the two sets of measurements had a high correlation, despite minimal differences in observed measurements. Pressure validation utilised a Bland-Altman and a linear correlation plot to determine the relationship between pressure readings for the ACM and CM. The Bland-Altman plot is presented in Figure 39.

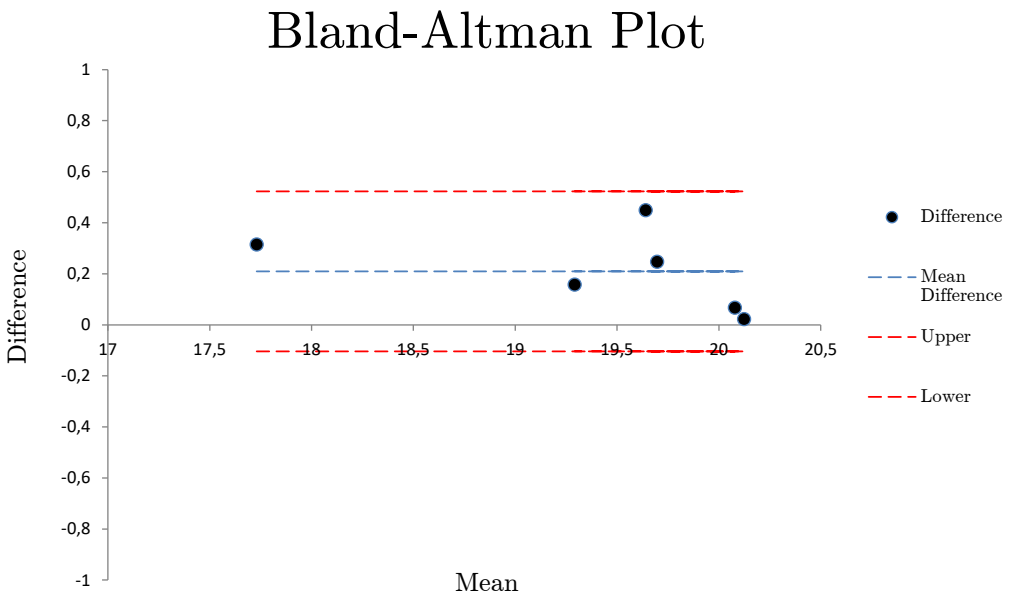


Figure 39: Bland-Altman plot for pressure readings of the ACM and CM

Plotted points represented by the difference in pressure readings for the ACM and CM were contained within the LOAs, indicating that an agreement between the two measurements was achieved. Microsoft Excel was used to generate the plot, calculate a mean difference of 0.21 mmHg, the upper limit for the LOA of 0.52 mmHg ($0.21 + 1.96 \times SD$), lower limit for the LOA of -0.1 mmHg ($0.21 - 1.96 \times SD$) for pressure reading differences between the ACM and CM. Analysis of the graph indicated that all calculated differences occurred between the limits of agreement and no significant

difference was observed between pressure readings for each model for the chosen confidence interval of 95%. The difference between the LOAs resulted in a range of 0.62 mmHg for the confidence interval, indicating that differences contained within the LOAs differed minimally. Plotted differences contained within this interval indicated that no significant change occurred between pressure readings for each model, resulting in an agreement being reached. An established agreement specified that residual limb models could be interchangeable if differences occurred within the limits of agreements. This would enable the usage of the CM to conduct pressure and bandage length measurement testing, as well as additional constructed models differing in circumferential measurements.

Further validation was achieved with the use of a linear correlation plot and the calculation of the R^2 value for observed pressure readings. Figure 40 illustrates the correlation between pressure readings for the ACM and CM, displaying positioning of readings close to the trend line. A R^2 value of 0.97 and a p-value of < 0.001 was achieved, indicating a large positive linear association between readings. Validation from the Bland-Altman and linear correlation plots allow for the use of additional constructed models to be used for measurement and pressure testing. These tests were performed to validate constructed model usage, with additional constructed models to be used to assess the relationship between observed and expected bandage length measurements, as well as pressure application readings for multiple residual limb models.

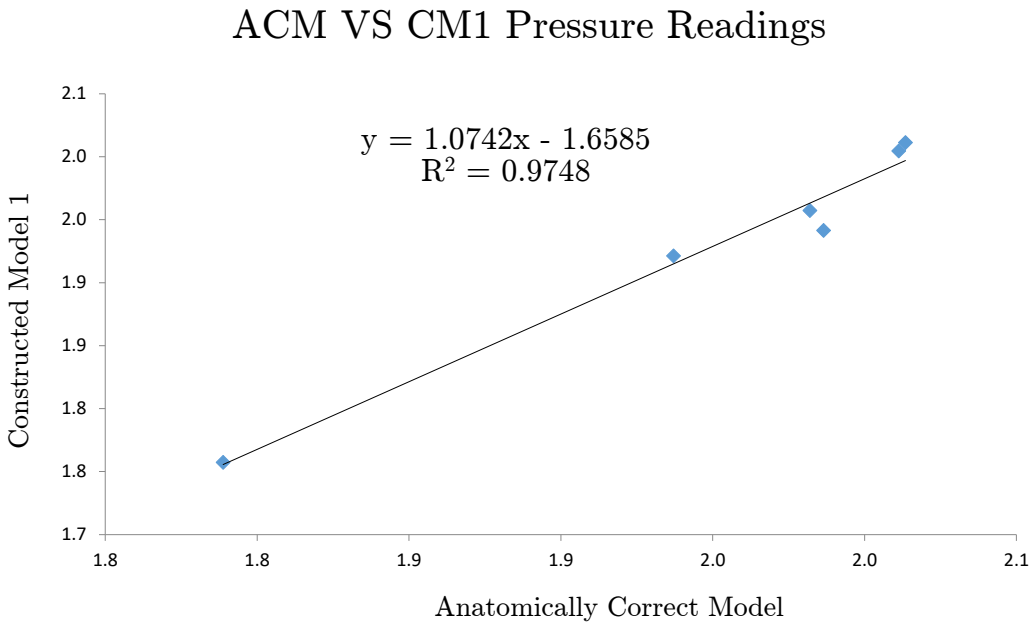


Figure 40: Linear correlation between ACM and CM pressure readings

6.3 Bandaging Template Generation and Application

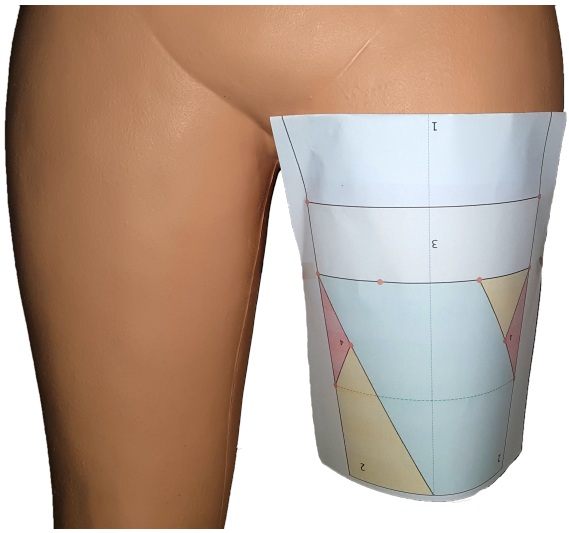


Figure 41: Bandaging template applied to the anterior region of the ACM

Bandaging templates were successfully generated for each of the six residual limb models. Five circumferential measurements, as seen in Figure 11 of Sub-section 4.2, were recorded for each residual limb model and served as input for Python Script 1 and 2, used to successfully generate six bandaging templates and predict required bandage lengths for each step of the bandaging process. Each template was correctly scaled to the dimensions of an A4 page and was successfully printed. Bandaging templates are generated in a PDF format for correct scaling and for ease of printing. Templates were then placed on anterior portions of residual limb models, as demonstrated in Figure 41, to instruct bandaging in conjunction with the mobile application and BUD dispenser. This sub-section details the outcome for the generated template of each model and illustrates each generated template.

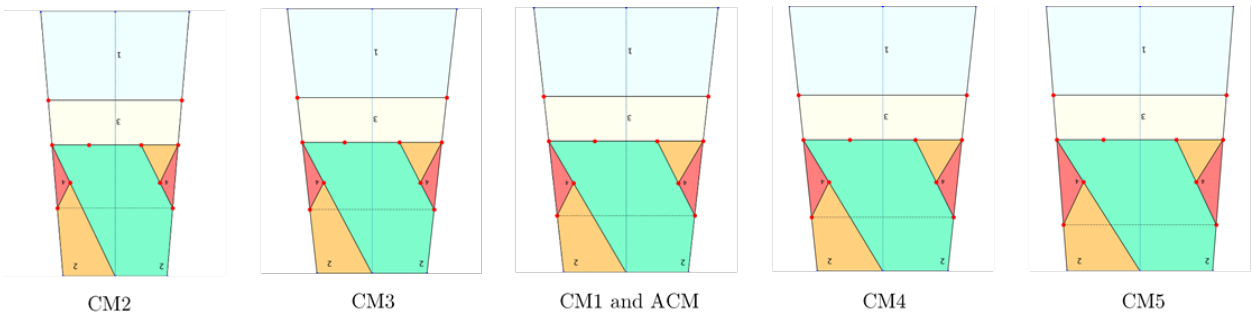


Figure 42: Generated templates for each residual limb model

A bandaging template can be seen for each of the residual limb models in Figure 42. The figure also illustrates the difference in landmark positions and bandage placement guides generated by Python Script 1, influenced by varying circumferential

measurements. Differences in template width is presented in Figure 42. Each constructed model differs in proximal and distal circumferential measurements in increments of 5 cm from the base model (CM1) and ACM. Templates were successfully generated from varying circumferential measurements, while adjusting the width of the template and maintaining landmark positioning on an A4 page to facilitate bandage placement. Proximal to distal circumferential measurements and template widths calculated to facilitate the dimensions of an A4 page (21 cm × 29.7 cm) in centimetres, are summarised in Table 3.

Table 3
Model circumferential measurements and template widths

Model	Circumferential measurements	Proximal template width	Distal template width
ACM	60, 58, 56, 49, 40	20.0	13.3
CM1	60, 58, 56, 49, 40	20.0	13.3
CM2	55, 53, 50, 46, 35	18.3	11.7
CM3	50, 45, 43, 39, 30	16.7	10.0
CM4	65, 60, 55, 50, 45	21.0	15.0
CM5	70, 63, 60, 56, 50	21.0	16.7

6.4 Mobile Application Testing

Mobile application testing involved the installation of the BUD Integration mobile application on smartphones that differed in price, screen size, processing capabilities and Android version. The described specifications for each smartphone is summarised in Table 4.

Table 4
Smartphone specifications

Smartphone	Price	Android Version	Screen Size (inches)	Successful Installation
Vodacom Kicka 5	R499	GO	4.95	Yes
Sony Xperia Z1	R3 298,32	5.1	5	Yes
Samsung S7	R10 178,27	8	5.1	Yes
Samsung S8	R10 954,19	9	5.8	Yes
Samsung Tab 2	R2 300	4.0.3	10.1	Yes

Installation was performed on devices varying in price, with the smartphone with the lowest cost being the Vodacom Kicka 5. The cost of the smartphone was R500, and served as the entry level device on which the application was tested on. The smartphone uses the Android GO operating system, designed for smartphones that are low-end and ultra-budget (Google, 2020). Installation on the device established the minimum requirements needed for successful installation of smartphone devices.

Further assessment was done to ensure that application features were functional on the base level smartphone. This included template generation, template saving and Bluetooth communication with the BUD dispenser. Successful template generation was achieved on the smartphone, but the speed of template generation was reduced due to the low random access memory (RAM) of 512 megabytes. Communication between the smartphone and dispenser was successfully established, where the application interface received all information transmitted from the dispenser and messages from the smartphone were successfully transmitted to the dispenser. Data storage in the form of residual limb measurements was successful after granting storage permission in the application interface, resulting in the successful storage of text files on the device's internal memory.

Devices with smaller screens such as the Kicka 5 and S7, utilised scrollable views in the application interface. Scrollable views are layout features present in the Android Studio environment that was added to mobile application pages for smaller sized devices. This allowed all the required information for each fragment to be displayed on the same page, while accommodating the smaller screen size. The Xperia Z1, S7 and S8 were used to feature mid-level to premium level smartphones when installing the application. Installation and dispenser communication were successful with these smartphones, with a single differing trait when compared with the Kicka 5 being the processing speed of the application and template generation speed. An Android tablet (Tab 2) was used to test installation of the application on tablet devices. Installation was successful and allowed the application to run on a device with a larger screen size, making visibility and navigation easier. Improvement on visibility was accomplished by increasing font and image sizes, as well as distributing editable text fields with the additional space present.

6.5 BUD Testing

6.5.1 Dispenser Assembly and Functionality

Testing of the BUD dispenser involved 3D printing of the dispenser encasing, evaluating the unrolled bandage length measurement measured by the rotary encoder and ESP32 microcontroller, and successful communication with the mobile application. Modelling of the dispenser was done using the computer-aided design software SolidWorks. Assembly of the dispenser, as seen in Figure 43 (a), involved inserting the rotary rod through a perforated section of the dispenser encasing and connecting it to the rotary encoder. An elastic crepe bandage was rolled onto the rotary rod, and remaining dispenser components were inserted into the encasing. The rotary mechanism allowed for easy rotation of the rod, allowing for the elastic bandage to be unrolled. A 3D printed BUD dispenser with elastic bandage and added components can be seen in Figure 43 (b).

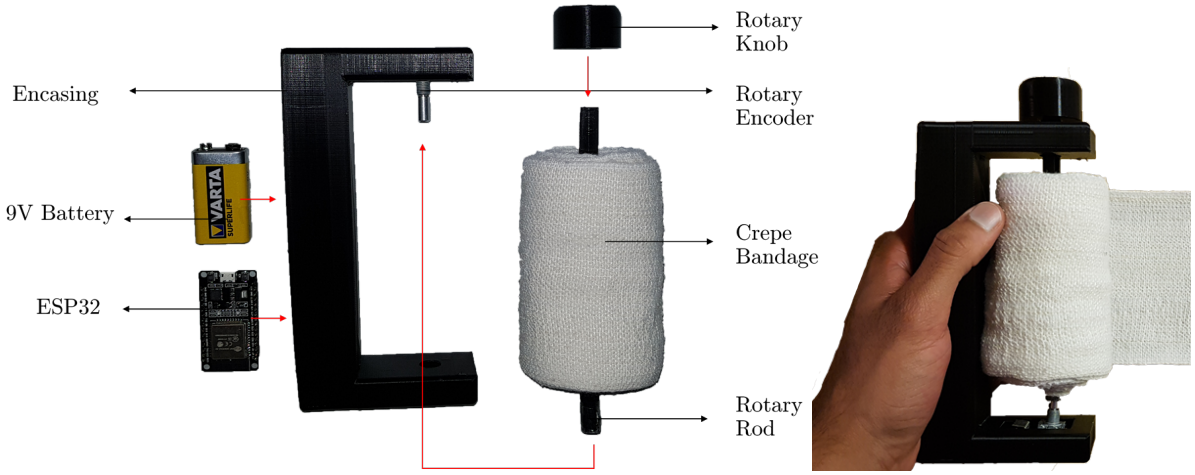


Figure 43: (a) BUD Assembly, (b) Assembled dispenser

6.5.2 Output Bandage Length Measurement Testing

Measurement testing for unrolled bandage lengths was performed to validate the unrolled length of elastic crepe bandage to be displayed by the mobile application recorded by rotary encoder readings. Measurements were recorded by utilising a measuring tape to measure the length of bandage unrolled for each incremental rotation (20 incremental rotations in one full revolution) of the rotary encoder. Multiple readings were taken by measuring the length of bandage unrolled in intervals of 10 incremental rotations.

A limit of 90 incremental rotations was chosen to facilitate the total length of the roll of bandage (100 cm). Testing resulted in the calculation of an average of 1.1 cm per incremental rotation of the rotary encoder. Measurements and results are summarised in Table 5. Validation of the output bandage length measurement for the dispenser allows for accurate bandage length measurement readings to be displayed on the mobile application interface, required to convey bandaging steps and trigger alert systems by the mobile application when required bandage length output is exceeded. Accurate readings allow for adequate pressure to be applied by the elastic bandage for residual limb re-shaping through predicted bandage length requirements calculated by Python Script 2.

Table 5
Bandage length measurements for each set of incremental rotations

Incremental Rotation	Physical Measurement (cm)	cm/Incremental Rotation
1	1.1	1.1
10	11.16	1.116
20	22.11	1.106
30	33.14	1.105
40	44	1.1
50	55.1	1.102
60	66.2	1.103
70	77.14	1.102
80	88.1	1.101
90	99.2	1.102
Average		1.104

6.6 Bandage Length Measurement and Pressure Testing

Testing and analysis was done on the observed and expected bandage length measurements to validate predicted bandage lengths calculated by Python Script 2. Pressure testing was done to assess bandage pressure exertion when applying predicted bandage lengths to the residual limb models. Evaluation of the script was done using a chi-square goodness of fit test, with pressure evaluation utilising a one sample t-test to compare observed pressure readings with the ideal pressure application from literature (20 mmHg). Partitioning of this sub-section allows for bandage length measurement and pressure testing results for each residual limb model to be summarised and tabulated independently. A discussion about noticeable observations follows to highlight similarities that may improve future iterations of the solution method.

Discussions about observations relating to accuracy, statistical analysis results and comments describing the bandaging process for each residual limb model follows each result for bandage length measurement and pressure testing. Bandage length measurement and pressure reading results and analysis have been partitioned into two sub-sections to discuss observations separately. Observations for bandage length measurements and pressure readings are independently discussed for each residual limb model, with observations indicating potential relationships between bandage lengths and pressure readings discussed in Sub-section 6.9.

6.7 Bandage Length Measurement Results and Analysis

Summarised bandage length measurement recordings and results for each residual limb model are tabulated in Table 6 containing the following sub-headings

- Residual Limb Model: Residual limb model corresponding to tabulated results.
- Bandaging Step: Current step of the bandaging process.
- Physical Measurements: Measurements observed after wrapping the residual limb model with elastic bandage. These measurements represent the unstretched bandage lengths and are measured using a measuring tape. Measurements were conducted by using template landmarks generated by Python Script 1, and the predicted bandage length calculation techniques of Python script 2. Applied bandage lengths for each step of the bandaging process were measured and recorded after wrapping a residual limb model.
- Expected Measurements: Bandage length measurements required to apply sufficient pressure to promote residual limb re-shaping. These measurements are predicted by Python script 2.
- Observed Measurements: Physical measurements multiplied by the compression factor to calculate applied bandage lengths to be compared with predicted bandage lengths calculated by Python script 2.
- Difference: Difference between observed and expected bandage length measurements. The difference for step 1 of the bandaging process will always be 0, as expected measurements equal observed measurements for the first bandaging step. Minimum, maximum and average calculations for the difference column therefore excluded step 1 to prevent inaccurate results.

Table 6
Residual Limb Model Bandage Length Measurements

Residual Limb Model	Bandaging Step	Physical Measurements (cm)	Expected Measurements (cm)	Observed Measurements (cm)	Difference (cm)
ACM	1	62	55.5	55.5	0.0
	2	88.5	75.8	78.8	3.0
	3	133.4	114.9	118.7	3.8
	4	27.2	20.0	24.2	4.2
CM1	1	60	55.2	55.2	0.0
	2	86.6	75.8	77.5	1.7
	3	127.3	114.9	113.9	1.0
	4	26.8	20.0	24.0	4.0
CM2	1	57	51.0	51.0	0.0
	2	86.4	74.1	77.3	3.3
	3	123.8	106.6	110.8	4.2
	4	23.8	19.4	21.3	1.9
CM3	1	50.0	44.8	44.8	0.0
	2	82.9	70.7	74.2	3.5
	3	114.8	99.4	102.7	3.4
	4	20.9	16.7	18.7	2.0
CM4	1	65.0	58.2	58.2	0.0
	2	87.9	76.8	78.7	1.9
	3	137.8	118.2	123.3	5.2
	4	26.9	20.2	24.1	3.9
CM5	1	70.0	62.7	62.7	0.0
	2	93.9	79.3	84.0	4.7
	3	148.5	127.5	132.9	5.4
	4	28.9	21.3	25.9	4.5

6.7.1 ACM Bandage Length Results Analysis

Bandage length measurement testing was performed on the ACM to obtain results that would best simulate bandage application on an amputee's residual limb. Predicted bandage length requirements were calculated by Python Script 2, using the recorded circumferential measurements of the ACM from proximal to distal: 60 cm, 58 cm, 56cm, 49 cm, 40 cm.

Figure 44: Bandaged ACM

Recorded bandage length measurements for each bandaging step was taken three times and recorded as the average measurement. From measurements recorded, the smallest difference between observed and expected measurements was 3 cm for step 2 of the bandaging process, while the largest difference between measurements resulted in 4.2 cm for step 4 of the bandaging process. The groin area of the ACM is fused to the thigh region of the ACM, limiting positioning of the elastic bandage during the first bandaging step. This caused the bandage to lose tension and resulted in a longer length of bandage needed to complete step 4 of the bandaging process, and secure the bandage to the model. This resulted in a greater length of bandage required to cover the final exposed area of the residual limb as instructed by step 4. Wrinkling of bandage due to model design is shown in Figure 44.

Performing a chi-square goodness of fit test on the observed and expected bandage length measurements in Table 6 resulted in a p-value of 0.77, using a 95% confidence interval. The null hypothesis for the ACM case can therefore not be rejected, indicating that observed and expected measurements did not differ significantly. Further evidence is seen in Figure 45, used to chart the comparison between observed and expected bandage length measurements for the ACM.

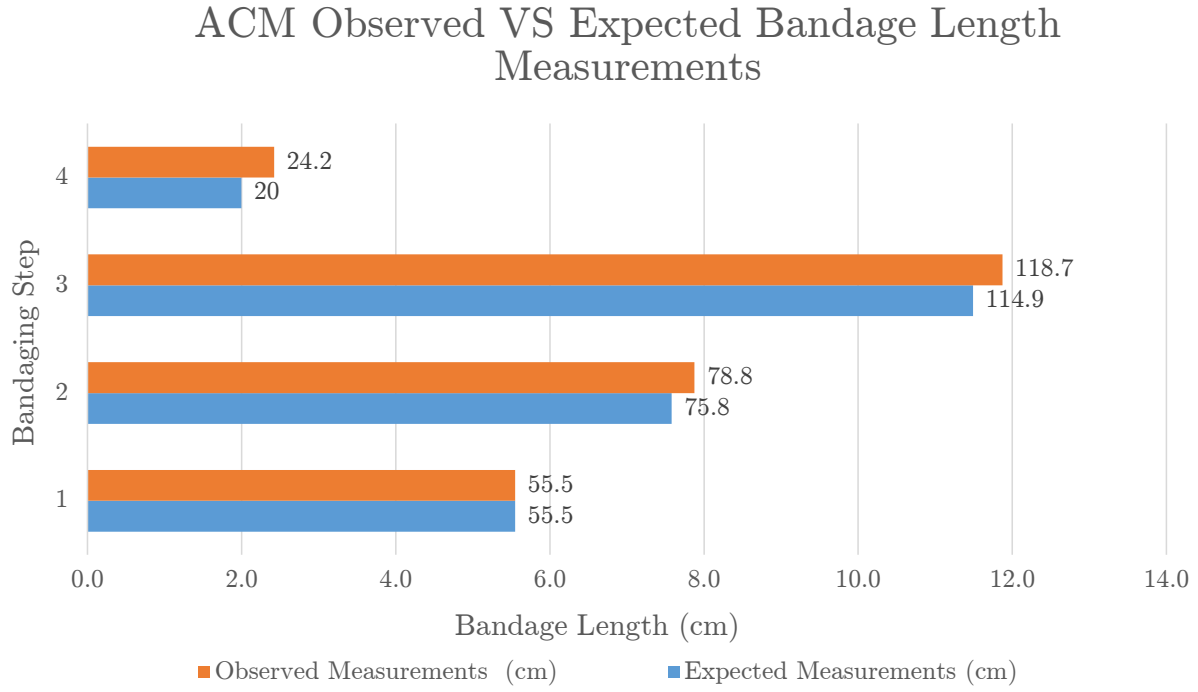


Figure 45: ACM bandage length measurement comparisons

6.7.2 CM1 Bandage Length Results Analysis

Construction of the first residual limb model was done to validate the use of additional constructed models for further bandage length measurement and pressure testing. The Bland-Altman plot in Sub-section 6.2 indicated that pressure readings between the ACM and CM1 were not significantly different, permitting the use of CMs for further testing. Equivalent statistical analysis tests applied to the ACM were performed on the CM1 model, with proximal to distal circumferential measurements equal to the ACM (60 cm, 58 cm, 56cm, 49 cm, 40 cm). Differences between observed and expected measurements were less significant when compared to ACM results. The smallest measurement difference for the CM1 was 1 cm, with the greatest difference being 4 cm. Minimal differing measurements resulted in a lower mean difference of 2.2 cm, when compared to the mean difference of 3.7 cm for the ACM.

Differences in mean values between CM1 and the ACM were not exclusive to bandaging accuracy, as problems in ACM design may have influenced bandaging and measuring of the ACM as a result. Comparing observed and expected bandage length measurements using a chi-square goodness of fit test resulted in a p-value of 0.84. A p-value higher than the level of significance (0.05) specifying that the null hypothesis could not be rejected, and measurements did not significantly differ. A comparison between bandage length measurements is shown in Figure 46.

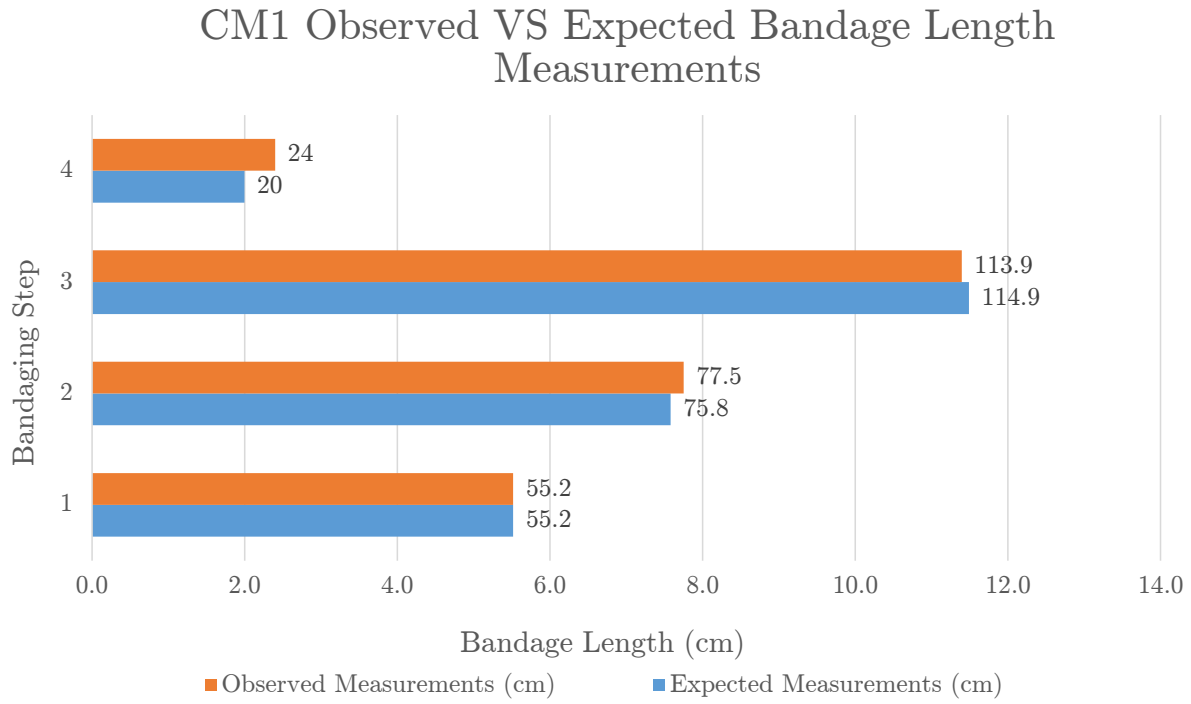


Figure 46: CM1 bandage length measurement comparisons

6.7.3 CM2 Bandage Length Results Analysis

Construction of the second model resulted in adjusting circumferential measurements to differ from the ACM and CM1 models. Proximal and distal circumferential measurements of the ACM were decreased by 5 cm to achieve a model with smaller dimensions. This resulted in a model with the following proximal to distal measurements: 55 cm, 53 cm, 50 cm, 46 cm, 35 cm.

Calculating the differences between expected and observed differences yielded a minimum difference of 1.9 cm for step 4 of the bandaging process and a maximum difference of 4.2 cm for step 2 of the bandaging process, with an average difference of 3.1 cm. The comparison between observed and expected bandage length measurements can be seen in Figure 47. The p-value obtained when performing a chi-square goodness of fit test

resulted in a value of 0.92, indicating that there was no significant difference between observed and expected measurements, as stated by the null hypothesis.

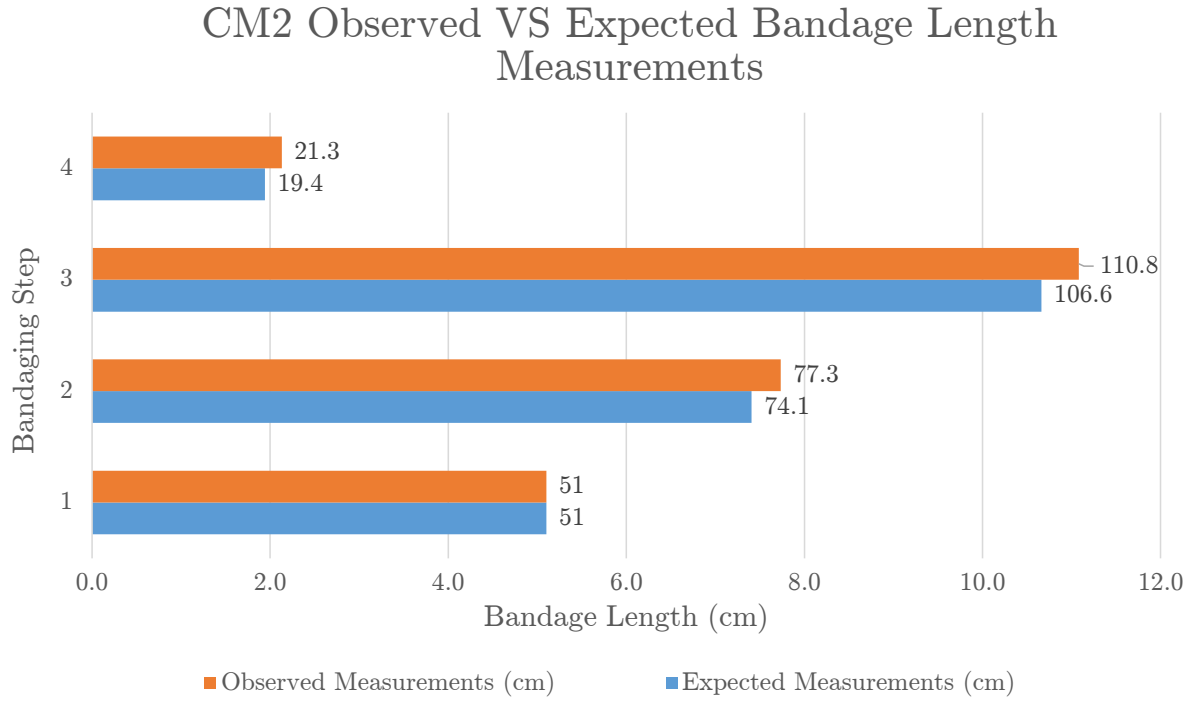


Figure 47: CM2 bandage length measurement comparisons

6.7.4 CM3 Bandage Length Results Analysis

The third CM was constructed with proximal and distal circumferential measurements 10 cm less than the measurements of the ACM. The model aimed to validate expected measurements and pressure application for a residual limb reasonably narrow given the length of the residual limb model. Dimensions of CM3 were used as the lower limit for circumferential measurements for each residual limb model. Circumferential measurements for CM3 were measured from proximal to distal as: 50 cm, 45 cm, 43 cm, 39 cm, 30 cm.

A comparison between the observed and expected measurements for CM3 can be seen in Figure 48. From the differences between observed and expected measurements, a minimum difference of 1.9 cm was observed for step 2 of the bandaging process, while a maximum difference of 5.2 cm was observed for step 3 of the bandaging process. An average difference of 2.9 cm was calculated for the difference between measurements.

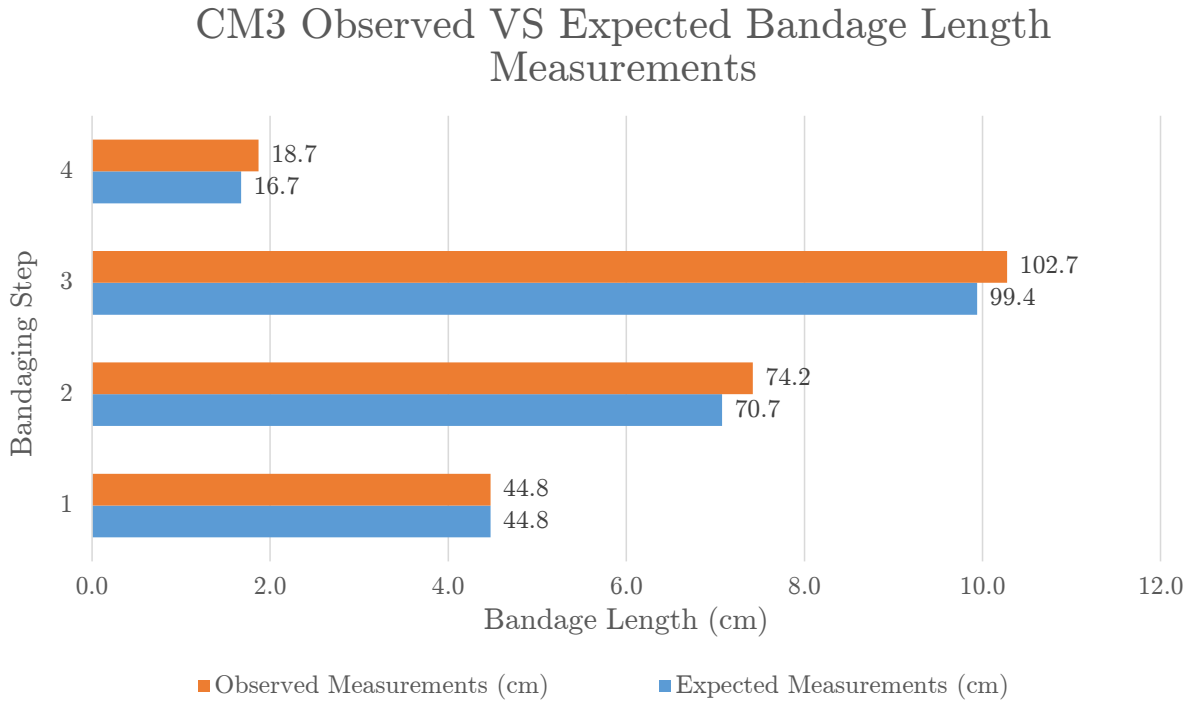


Figure 48: CM3 bandage length measurement comparisons

6.7.5 CM4 Bandage Length Results Analysis

CM4 was constructed with circumferential measurements 5 cm more than the distal and proximal circumferential measurements of the ACM. Circumferential measurements for CM4 were measured from proximal to distal as: 65 cm, 60 cm, 55 cm, 50 cm, 45 cm. Construction of CM4 allowed for the testing of predicted bandage length measurements and pressure readings on a residual limb model larger than the ACM. Results from models greater in size than the ACM were used to further evaluate testing criteria and highlight similar observations between models that differ in dimension.

Differences in observed and expected measurements resulted in a minimum difference of 1.9 cm for step 2 of the bandaging process, a maximum difference of 5.2 cm for step 3 of the bandaging process and a mean difference of 3.7 cm. A p-value of 0.79 was calculated for the chi-square goodness of fit test, indicating that no significant differences between observed and expected measurements for CM4 were observed. A comparison between the bandage length measurements can be seen in Figure 49.

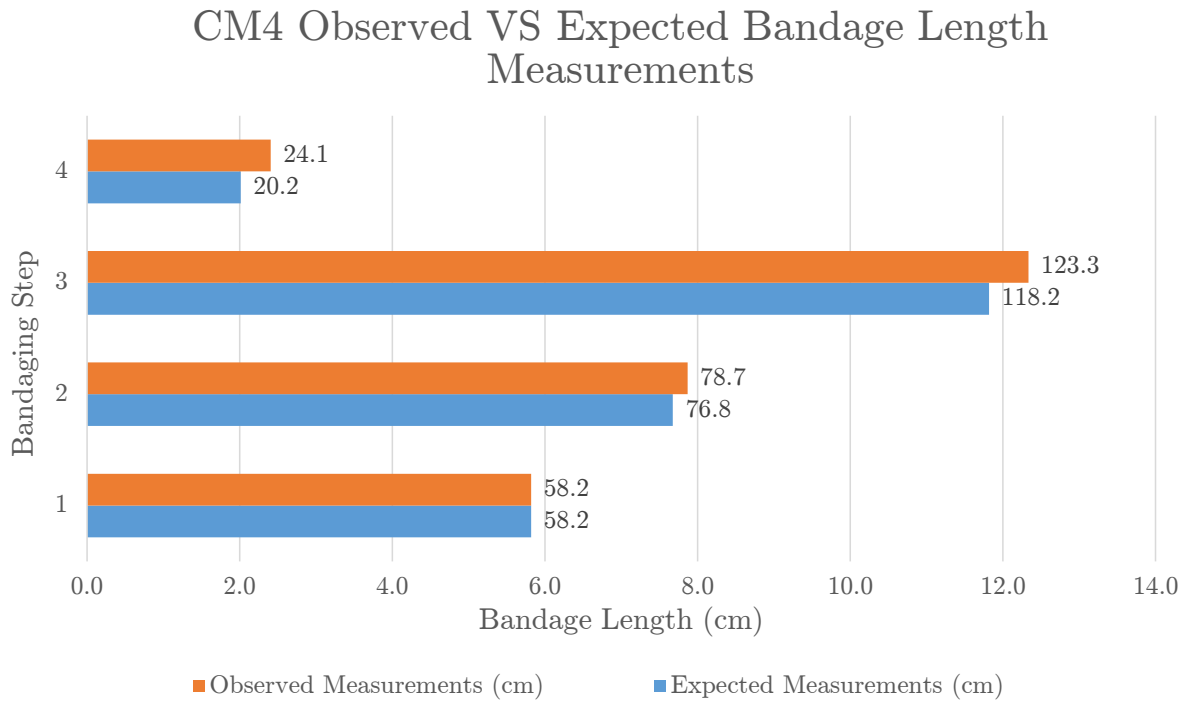


Figure 49: CM4 bandage length measurement comparisons

6.7.6 CM5 Bandage Length Results Analysis

Construction of the final simulated model involved increasing the distal and proximal circumferential measurements of the ACM by 10 cm to produce CM5. CM5 was used as the upper limit for circumferential measurements to construct a residual limb model realistic in volume given the length of the model. Circumferential measurements for CM5 were measured from proximal to distal as: 70 cm, 63 cm, 60 cm, 56 cm, 50 cm.

Bandaging of the CM5 resulted in a minimum difference of 4.5 cm between observed and expected bandage length measurements for step 4, a maximum difference of 5.4 cm for step 3 and an average difference of 4.9 cm. Larger differences in observed and expected measurements were recorded for the model, but a chi-square goodness of fit test indicated that observed the measurements do not differ significantly from expected measurements, A p-value of 0.69 was obtained when performing the test, resulting in the acceptance of the test's null hypothesis. Differences between observed and expected measurements can be seen in Figure 50.

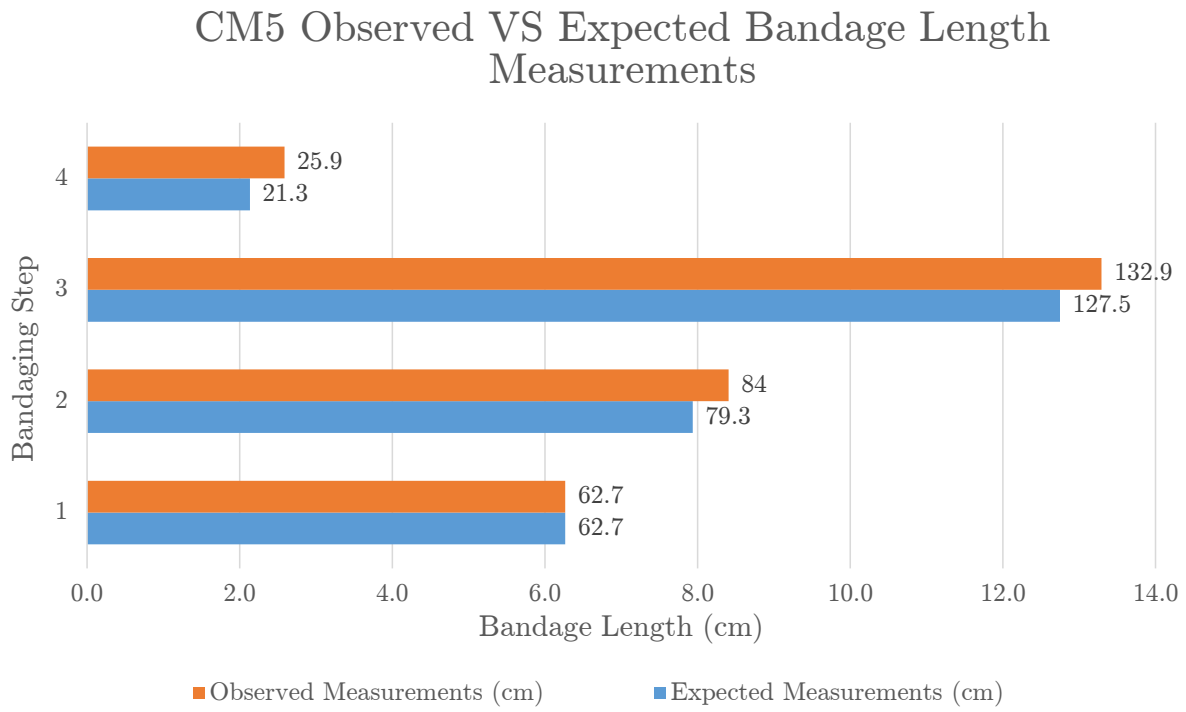


Figure 50: CM5 bandage length measurement comparisons

6.8 Pressure Readings Results and Analysis

Pressure readings for each step of the bandaging process for each residual limb model have been tabulated in Table 7 with the following sub-headings:

- Residual Limb Model: Residual limb model corresponding to tabulated results.
- Bandaging Step: Current step of the bandaging process.
- Pressure Readings: Pressure readings obtained by the FSR circuit, converted to mmHg using the same calculations outlined in Sub-section 4.2.1. Each pressure reading was taken three times, with the average reading presented in the table. Bandaging steps 2 and 3 contain two readings to measure pressure applied when diagonally wrapping left and right areas of the residual limb.
- Target Pressure: Target pressure of 20 mmHg, researched from literature to be the ideal pressure application to influence residual limb volume.
- Difference: Difference in pressure applied by the elastic crepe bandage compared to the ideal pressure application of 20 mmHg. Comparisons for all four steps of the bandaging process for each residual limb model were calculated.

Table 7
Residual Limb Model Pressure Readings

Residual Limb Model	Bandaging Step	Pressure Readings (mmHg)	Target Pressure (mmHg)	Difference (mmHg)
ACM	1	17.9	20	2.1
	2	19.8, 20.1	20	0.2, -0.1
	3	19.9, 20.1	20	0.1, -0.1
	4	19.4	20	0.6
CM1	1	17.6	20	2.4
	2	19.6, 20.1	20	0.4, -0.1
	3	19.4, 20.0	20	0.6, 0.0
	4	19.2	20	0.8
CM2	1	17.5	20	2.5
	2	19.0, 20.1	20	1.0, -0.1
	3	19.4, 20.1	20	0.6, -0.1
	4	19.2	20	0.8
CM3	1	18.7	20	1.3
	2	20.2, 19.4	20	-0.2, 0.6
	3	20.0, 19.8	20	0.0, 0.2
	4	19.2	20	0.8
CM4	1	18.5	20	1.5
	2	20.0, 19.0	20	0.0, 1.0
	3	20.2, 19.1	20	-0.2, 0.9
	4	19.2	20	0.8
CM5	1	16.7	20	3.3
	2	17.6, 17.8	20	2.4, 2.2
	3	19.4, 19.9	20	0.6, 0.1
	4	17.7	20	2.3

A comparison between the target pressure application of 20 mmHg and pressure readings for each step of the bandaging process for each residual limb model can be seen in Figure 51. Trends in pressure change between bandaging steps for each model is illustrated in the figure, where similarities in pressure application during each step of the bandaging process for each model can be seen.

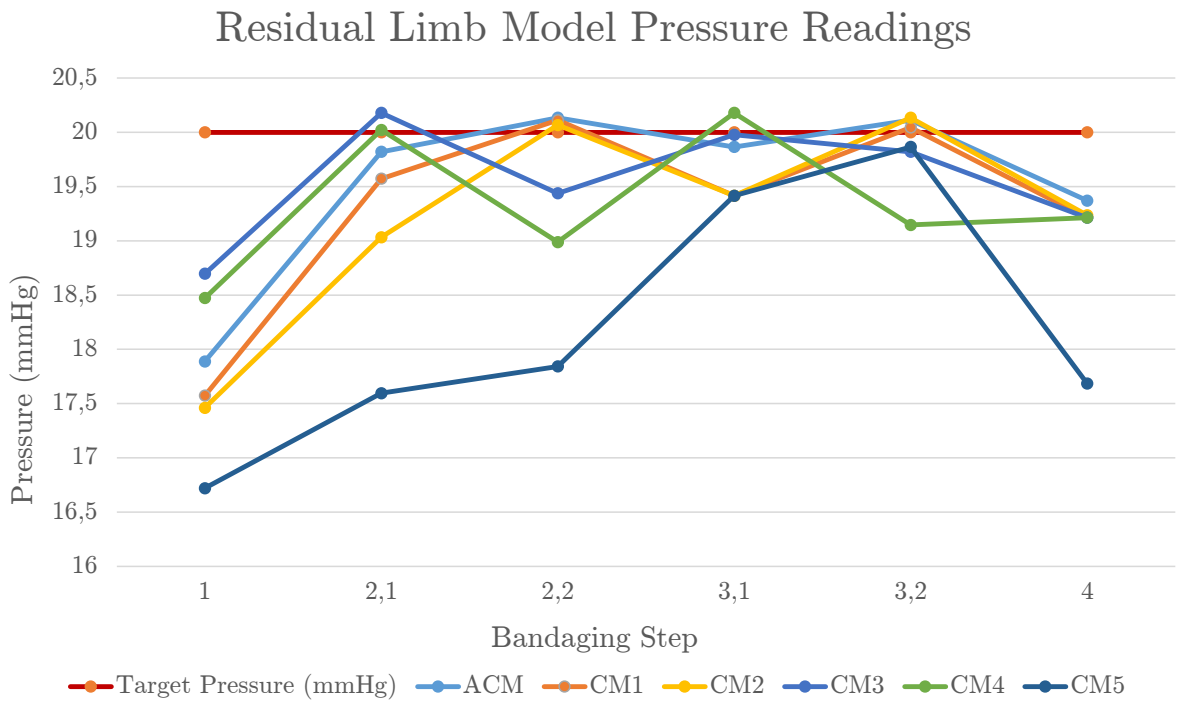


Figure 51: Residual limb model pressure readings comparison

6.8.1 ACM Pressure Reading Results and Analysis

A one sample t-test was performed to evaluate the mean observed pressure reading (sample mean) with the hypothesised mean of 20 mmHg from literature. A p-value of 0.24 was calculated in Excel using a confidence interval of 95%, indicating that the null hypothesis cannot be rejected. Calculation of the sample mean (19.5 mmHg) indicated a difference of 0.5 mmHg from the hypothesised mean proposed to be the ideal pressure value of 20 mmHg, proving that mean values did not differ significantly. The lowest pressure reading (17.9 mmHg) was recorded during the first step of the bandaging process. Wrinkling of the elastic bandage during the bandaging process may have resulted in less pressure applied to the region of step 1, resulting in a lower pressure reading. The highest pressure reading (20.1 mmHg) was recorded at steps 2 and 3 of the bandaging process. Both steps 2 and 3 experienced an increase in pressure when completing the second half of their respective bandaging technique. Possible reasoning may be due to pressure application being increased when the elastic bandage was stretched further to complete bandaging of the ACM. The general trend of pressure application is seen in Figure 51 for each reading, with target pressure application indicated by the red line. The general trend of pressure application recorded for the ACM indicated an increase in pressure exertion when diagonally wrapping the distal end of the model, a decrease in pressure when laterally wrapping the model at step 1 and a decrease in pressure when covering the final exposed area at step 4.

6.8.2 CM1 Pressure Reading Results and Analysis

The p-value (0.13) of the one sample T-test for pressure readings was higher than the significance level (0.05). This indicated that the null hypothesis could not be rejected, and that no significant difference between the sample mean and hypothesised mean could be determined. The lowest pressure reading of 17.6 mmHg was recorded for the first bandaging step, with the largest pressure readings of 20.1 mmHg and 20 mmHg occurring during the second half of the bandaging technique for steps 2 and 3. An overall mean pressure application of 19.3 mmHg was achieved, indicating a minimal difference in pressure exertion when compared to the target application pressure of 20 mmHg. Pressure application for each bandaging step can be seen in Figure 51. A similar trend in pressure exertion was recorded for CM1 when compared to the ACM. An increase in pressure was recorded for steps 2 and 3, conducting distal diagonal bandaging, with lower pressure readings at steps 1 and 4.

6.8.3 CM2 Pressure Reading Results and Analysis

Pressure readings indicated that the sample mean (19.2 mmHg) differed from the target pressure application by 0.8 mmHg. This was reinforced by the calculated p-value of the one sample t-test, resulting in a value of 0.13. The lowest pressure reading recorded was 17.5 mmHg, recorded at step 1 of the bandaging process, with higher pressure readings being recorded during steps performing diagonal bandaging around the model. The trend and difference in pressure for each reading can be seen in Figure 51. A similar trend in pressure application was observed for CM2, with an increase in pressure application recorded for steps 2 and 3.

6.8.4 CM3 Pressure Reading Results and Analysis

Calculating the p-value of the chi-square goodness of fit test resulted in a value of 0.92, resulting in the acceptance of the null hypothesis, stating that observed and expected measurements do not differ significantly. A similar outcome for the pressure readings was achieved. The p-value calculated from the one sample t-test equated to 0.1. Results of the test specified the relativity between the sample mean (19.6 mmHg) and the hypothesised mean, validating the null hypothesis that the two means did not differ significantly for a 95% confidence interval. Pressure readings relative to the target reading is graphed in Figure 51, with the trend in pressure application displayed.

6.8.5 CM4 Pressure Reading Results and Analysis

Recorded Pressure readings specified a minimum pressure reading of 18.5 mmHg for step 1, a maximum reading of 20.2 mmHg for step 3 and a mean applied pressure of 19.3 mmHg. The mean pressure reading (19.3 mmHg) did not differ significantly from the target pressure of 20 mmHg. Validation was achieved by calculating a p-value of 0.05 for the one sample t-test used to evaluate the sample mean pressure reading with the hypothesised mean. Changes in pressure readings for each bandaging step is graphed in Figure 51.

6.8.6 CM5 Pressure Reading Results and Analysis

Pressure testing performed on the CM5 model resulted in a difference between the calculated sample mean (18.2 mmHg) and the hypothesised value. A p-value of 0.01 was calculated for the one sample t-test, indicating strong evidence against the null hypothesis and that mean pressure application for CM5 differed from the ideal mean pressure application of 20 mmHg. A minimum pressure reading of 16.7 mmHg was recorded for step 1, a maximum reading of 19.9 mmHg was recorded for step 3 and an average pressure application of 18.2 mmHg was recorded. Differences in pressure application for each bandaging step is graphed in Figure 51.

6.9 Results Summary

Results for each model were within acceptable ranges, with each statistical analysis test indicating that the difference between observed and expected bandage length measurement results were not significant. Differences between expected and observed bandage length measurements varied for each step of the bandaging process when assessing each model. Mean differences calculated for each model increased as model dimensions increased, due to increases in bandage length usage and predicted lengths for each bandaging step. The lowest pressure readings were recorded at step 1 of the bandaging process for each residual limb model. This may have been due to the orientation of bandaging, since step 1 used a lateral wrap instead of diagonal wraps, resulting in lower pressure application. Minimum differences were experienced during step 4 of the bandaging process for 50% of residual limb models, but models were not consecutive in size. These models included the CM2, CM3 and CM5. A potential observation may be that an increase in required bandage length to wrap a larger model

results in less bandage available to perform the final bandaging step, therefore producing a minimum difference. A recording error while bandaging CM4 may have influenced this observation, as the minimum difference recorded for CM4 occurred at step 2 of the bandaging process.

Maximum differences for 50% of the models occurred at step 3 of the bandaging process, where the largest predicted bandage length is calculated. Models with largest differences at step 3 include CM2, CM4 and CM5. Maximum differences between observed and expected bandage lengths occurring at step 3 may be a result of the required length of bandage required to perform the bandaging step. Step 3 required the greatest length of bandage to be applied during bandaging, resulting in the greatest potential for larger differences between observed and expected bandage length measurements, Greater bandage length requirements may therefore be correlated with larger measurement errors. Recorded measurements were successfully assessed for residual limb models individually. Testing performed on additional models in the future may highlight similarities and trends within the bandaging process for similar sized models.

Pressure testing was evaluated using the one sample t-test statistical analysis to compare observed sample means with the hypothesised desired mean pressure application of 20 mmHg. CM5 was the only model where evidence against the null hypothesis was established, indicating that the sample mean (18.2 mmHg) differed from the hypothesised mean (20 mmHg), when a 95% confidence interval was chosen. Dimensions for CM5 were substantially larger than the base dimensions of the ACM and CM1. Testing results of the CM5 may therefore indicate an upper limit for template and compression factor applications, resulting in additional adjustments to be made for residual limbs with dimensions equal to or larger than CM5. Minimum pressure applications at step 4 of the bandaging process occurred for 50% of the models. Step 4 is the final step of bandaging and covers the final exposed area of the residual limb where the healing amputation scar would be situated. A lower pressure application is therefore needed to secure bandaging in place without causing irritation or harm to the wound. Maximum pressure readings occurred at steps 2 and 3 while bandaging the models. Both steps required diagonal bandaging that would re-shape and control swelling in an amputee's transfemoral residual limb. Higher pressure readings are therefore expected in these areas to best shape the residual limb. A maximum pressure reading of 20.2 mmHg was recorded for CM4 during step 3 of bandaging. No visible relationship between pressure application and residual limb size could be established for the six models. Further testing on multiple models in the future could potentially provide more information on common pressure application readings.

6.10 Discussion

External pressure is required to re-shape the transfemoral residual limb to achieve a shape suitable for prosthesis integration. The most commonly used form of pressure application for lower limb amputees is the elastic wrap bandage. Bandaging of the residual limb however, requires skill and frequent reapplication. Improper bandaging of the residual limb may lead to abnormal growth, requiring additional time to re-shape the residual limb, prolonging the prosthesis fitting process as a result. C. Visser conducted a study to determine lower limb residual limb bandaging efficiency amoung amputees prior to prosthesis integration (Visser, 1998). Out of 33 candidates, only 49% (16) received education on proper residual limb bandaging, while the remaining 51% received no form of education or demonstration regarding residual limb bandaging. Of the 16 test subjects who were educated with residual limb bandaging, 63% applied sufficient pressure distally, with only 9% of subjects covering all criteria for effective residual limb bandaging. He concluded that patients face difficulty in understanding and retaining information and bandaging instructions during the initial stages of rehabilitation, but that written instructions may help ensure proper post-operative care of the residual limb through proper bandaging.

Table 8
Residual limb bandaging assessment scale (Manella, 1981)

1	Pressure decreased from distal to proximal
2	Uses figure of 8 and oblique wraps only
3	Even distribution of layers of wrap
4	All areas are covered
5	Wrap is secure and is properly anchored above the knee
6	No wrinkling of bandage
7	Re-wraps every 4 hours (patient report)
8	No areas of highly concentrated pressure (redness)
9	Pressure is not directed at the cut end of tibia
10	Patient wraps independently
	Total Score

Clinical staff therapists developed a 10-point rating scale to assess the effectiveness of residual limb bandaging performed by an amputee, as well as to evaluate independence in bandaging procedures (Manella, 1981). Table 8 lists the criteria assessed when determining a final score to evaluate patient bandaging. The researched solution method has addressed each of the rating scale's evaluation criteria in the following manner:

1. Pressure application for each residual limb model indicated that greater pressure readings were recorded at steps 2 and 3 of the bandaging process. These steps incorporated diagonal wrapping at the distal end of the residual limb to control oedema. Lower pressure readings were recorded during step 1, where bandaging occurred to anchor the bandage. An average pressure application for steps 2 and 3 for all six residual limb models was calculated as 19.6 mmHg, with an average pressure application of 17.8 mmHg for step 1. Pressure readings at the distal ends of the residual limb models were therefore higher than the proximal pressure readings recorded at step 1 of the bandaging process.

2. Design of the bandaging template promotes the use of oblique turns while bandaging the residual limb by coordinating bandaging with diagonal segments. This ensures the use of the figure-8 pattern while bandaging the residual limb.

3. An even distribution of elastic bandage is achieved through calculated positioning of each bandaging guideline on the bandaging template. Guides on the bandaging template ensure that bandage placement is consistent, while accommodating for a slight overlap to ensure full coverage of the residual limb.

4. Bandaging template guidelines have been designed to cover the entire residual limb with elastic crepe bandage. Bandage placement is concentrated on uncovered areas of the residual limb for each successive step of the bandaging process. Template guidelines accommodate for a slight overlap of elastic bandage while covering the exposed area of the residual limb.

5. Step 1 of the bandaging process acts as an anchor to secure the elastic bandage before performing diagonal bandaging during steps 2 and 3. Step 4 covers the final exposed area of the residual limb and secures the remaining crepe bandage to the residual limb using a bandaging clip.

6. With exception to the ACM, all CMs were successfully bandaged with no wrinkling occurring. Bandage placement covered the exposed areas of the residual limb, while allowing for a slight overlap of bandage. Wrinkling of bandage was prevented by positioning each bandaging template segment far enough to ensure that bandage overlap was minimal. Segment dimensions were wide enough to accommodate bandage width (10 cm) and were distant enough to establish an overlap of less than half the bandage width.

7. Inspection of the residual limb and re-application of elastic bandage is required multiple times during the day. The researched solution method provides a way to wrap the residual limb in a consistent manner to maintain consistent healing of the residual limb with the use of the bandaging template and predicted bandage

lengths. Re-application of a previously used bandage is facilitated by the BUD dispenser to allow for re-wrapping of the residual limb with existing or new elastic crepe bandage rolls.

8. The calculated compression factor of 0.895 adjusts unstretched bandage lengths to exert an optimal pressure application of 20 mmHg. Pressure application is sufficient to control oedema without causing damage to the residual limb or prevent venous flow. Bandage application with the assistance of the bandaging template, mobile application and BUD dispenser aims to apply a uniform pressure to the residual limb, preventing the application of highly concentrated pressure exertion.

9. Step 4 of the bandaging process covers the final exposed area of the residual limb where the amputation scar is present. Bandaging of this area exerts enough pressure (20 mmHg) to influence residual limb volume without causing damage to the healing wound.

10. A primary objective of the solution method is to promote independence in amputees to bandage their residual limb. The solution method acts as a portable guidance system that instructs bandaging to ensure consistent healing and maintenance of the residual limb. A consistent method to instruct bandaging will enable the amputee to bandage their residual limb independently, while adjusting to the rehabilitation process.

A comparison between the rating scale and solution method testing results indicated that the most common challenges experienced during residual limb bandaging could be addressed by incorporating the solution method into an amputee's rehabilitation period. Utilisation of the bandaging template, mobile application and BUD dispenser will assist in ensuring: optimal pressure is applied to the residual limb, a consistent technique of bandaging is maintained, patient independency is promoted and injury due to improper bandaging is prevented.

Improper bandaging results in incorrect pressure application that may negatively affect residual limb volume control and prevent proper circulation. Varghese, et. al. conducted a study to measure the pressure applied by elastic bandages for the purpose of determining the optimal pressure required to influence residual limb volume without causing harm. Four pressure sensors were used to measure exerted pressure readings applied by elastic bandaging. Two sensors were placed on the distal anterior region of the residual limb, with the remaining two sensors placed on the posterior region. From the results obtained, Varghese, et. al observed that of the thirteen test subjects, only five subjects applied elastic bandage within an acceptable pressure range of 20 - 25

mmHg. Pressure application ranged from 15 to 65 mmHg, with a vast difference in pressure application due to skill and experience with residual limb bandaging (Varghese et al., 1981).

With the use of the bandaging template, mobile application guidelines and BUD dispenser, pressure application on each residual limb model for each step of the bandaging process was close to the optimal ideal pressure application of 20 mmHg. Bandaging techniques were kept constant by following template guidelines, ensuring that optimal pressure was constantly applied when re-bandaging the residual limb. Incorporation of the solution method into the rehabilitation would assist with applying optimal pressure, ensure constant healing through consistent bandaging techniques, instruct amputees on proper bandaging methods and promote independence.

7 Conclusion and Future Recommendations

Research into a novel method of transfemoral residual limb re-shaping was done to establish a method that could instruct amputee's on how to perform proper bandaging, while promoting a residual limb shape suitable for prosthesis integration. A solution method was designed and comprised of three sub-systems: A bandaging template, a mobile application and a bandaging dispenser. Successful development of each sub-system occurred, and individual component testing for each sub-system was performed before evaluating the overall solution method. Successful validation of individual components was achieved for each sub-system. A summary of the sub-system and component testing results follows:

- Successful template generation, printing and application for multiple residual limb models.
- Successful bandage length predictions required for each step of the bandaging process, coinciding with the bandaging template.
- Validation of a compression factor that adjusted unstretched bandage lengths to extended length for the purpose of applying sufficient pressure.
- Successful installation of the BUD Integration mobile application on five devices. Each device successfully interfaced (using Bluetooth) with the BUD dispenser and correctly displayed the unrolled bandage length measurements of the dispenser.
- Successful 3D printing of the BUD dispenser encasing.
- Validation of the unrolled bandage length measured by the dispenser, calculated by the ESP32 microcontroller and rotary encoder.

Testing occurred on an anatomically correct residual limb model and five additional constructed models. The constructed model circumferential measurements were adjusted to form residual limb models of varying sizes. Overall evaluation in the form of a statistical analysis for each testing criteria was performed. A chi-square goodness of fit test assessed the differences in bandage length measurements between observed measurements and expected bandage length measurements calculated by a custom Python script to predict bandage length requirements. Similarly, a statistical analysis in the form of a one sample t-test was performed to test pressure application for each model. Average mean pressure application for each model was used to compare the ideal pressure application from

literature, with pressure application for five of the six models recorded at an acceptable range.

While using the developed solution method, the five residual limb models were successfully bandaged with pressure in acceptable range of the ideal pressure application of 20 mmHg. Bandaging was completed for each model following template and mobile application guidelines. Testing on a broader range or residual limb models is needed to further validate the solution method's efficacy, while patient testing is required to assess usage and functionality. Future recommendations relating to design, testing and future applications are discussed to outline areas where improvements can be made.

7.1 Future Recommendations

7.1.1 Design Improvements

Initial iterations of each sub-system was successfully designed and developed for this study, with the aim of instructing proper bandaging to transfemoral amputees. Design improvements to the bandaging template, mobile application and dispenser serve to advance current iterations of each sub-system to make the overall solution method more efficient and easier to use. Design improvements include the following:

- An A4 page medium was used to print the bandaging template to instruct proper bandaging. Template designs incorporated the use of an A4 page, due to low pressure application and easy removal. Alternative mediums such as a projected light source, temporary markings or augmented reality mediums by which the template may be applied can be explored to streamline the process of generating and applying the template. Calculated landmarks generated by Python Script 1, may be marked onto the residual limb, where bandaging steps are conveyed by the mobile application.
- Current versions of the mobile application have been exclusively installed on Android devices, to meet the scope of the study and due to experience with the application software. Future versions of the application should have the ability to run on smartphones running the iPhone IOS or Windows operating systems to broaden the range of devices capable of installing the application.
- Application features to aid amputees on basic residual limb care can be implemented as an additional page in the mobile application interface. Details involving stump

care, clinician contact details and rehabilitation centre locations provide additional support and can be added to the interface.

- Dispenser dimensions can be reduced to produce a more compact device, resulting in shorter 3D printing periods, less material used and easier maneuverability while bandaging the residual limb.
- Incorporation of existing bandaging techniques can be adapted to be used with mobile application and dispenser features. Existing bandaging techniques may be preferred over the current template guidelines or commonly used re-shaping techniques. Future development of the predicted bandage length script should include features to predict bandage lengths and compression for alternative bandaging techniques.

7.1.2 Database Generation

Early design concepts in aiding with residual limb shape control included the use of statistical shape modelling to predict residual limb maturation and healing from a data set. Lack of data resulted in an alternative solution to be researched that could both instruct bandaging and collect residual limb data. Generating each bandaging template and predicting required bandage lengths requires the input of several residual limb circumferential measurements. Measurements are stored in text files for reuse and progress observation purposes. Future uses of the stored data for each amputee can be used to generate a database comprising of residual limb measurements for amputees categorised by age, gender, weight and height. Data can then be used to forecast residual limb maturation and predict shape change before an amputee's healing period is complete. Predicted measurements can be used to develop prosthetic sockets in advance, decreasing the time taken to fit a prosthesis and allow amputees to resume daily acts of living sooner.

7.1.3 Mobile Application Feedback System

Data storage of residual limb measurements allows for the tracking of volume change and healing progress through the changes of residual limb measurements. Integrating a feedback system into the mobile application to transmit updated residual limb measurements of an amputee to associated clinicians allows for remote inspection of the residual limb. Images of the current state of the residual limb and measurements can be

sent to clinicians before scheduled appointments to review progress before clinical visits. This will decrease the required time for each clinical visit, allowing for more patients to be seen by clinicians within a day, and shorten clinical visits for amputees. Clinicians can then inspect progress of healing and advise amputees on further recommendations to maintain healing progress or advise changes to be made remotely.

7.1.4 Testing Future Recommendations

Future recommendations for testing procedures aim to improve accuracy of current testing criteria, add additional testing variables and assess system usability. Accuracy of testing may be improved with the inclusion of additional residual limb models to apply the solution method to. Observations may highlight trends in data received from multiple models that have been overlooked by the current array of models. Current testing was limited to six models to facilitate ACM dimensions and shape. Future models may be developed from patient residual limb measurements to increase the range of residual limb sizes to be tested on. Data collected from new models will allow for improved bandage length measurement and pressure testing. Testing for pressure may be improved by measuring pressure application of the entire surface area where bandaging is applied. Measurement of covered surfaces will indicate the uniform pressure applied to the residual limb, instead of solely measuring pressure application at the perimeter of each residual limb model, as performed in this study.

Further testing of the solution method involves the incorporation of clinicians to assess the usability of the method. A system usability scale may be used to assess functionality, ease of use and the overall value that the solution method may have in promoting proper bandaging. Unfortunately, patient testing could not be performed. Patient feedback regarding usability is a crucial continuation of the testing process regarding the evaluation of the solution method's overall efficacy and value. Usability testing with amputees will provide necessary feedback and information on the usability of the solution method from the perspective of the patient. Researching factors such as the promotion of independence, ease of use for experienced and inexperienced individuals in residual limb bandaging and potential improvement areas for design and testing methods that would improve future iterations of the developed solution method.

References

Albino, F. P., Seidel, R., Brown, B. J., Crone, C. G., & Attinger, C. E. (2014, sep). Through Knee Amputation: Technique Modifications and Surgical Outcomes. *Archives of Plastic Surgery*, *41*(5), 562–570. Retrieved from https://www.ncbi.nlm.nih.gov/pubmed/25276650https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4179362/ doi: 10.5999/aps.2014.41.5.562

Amendola, M. F., & Harris, S. (2018). IP129. Heterotopic Ossification After Above-the-Knee Amputation: Case Report and Review of the Literature. *Journal of Vascular Surgery*, *67*(6), e120–e121.

Amputation Rehabilitation Program. (2018). *A step in the right direction.* Hamilton Health Sciences. Retrieved 2020-11-17, from https://www.hamiltonhealthsciences.ca/a-step-in-the-direction-amputation-rehabilitation-program/

Atlas, D. (2015). International Diabetes Federation. *IDF Diabetes Atlas, 7th edn. Brussels, Belgium: International Diabetes Federation*, 33.

Boldt, J., & Maguire, M. T. (2013). *Transfemoral (Above Knee) Amputation.* Advanced Prosthetics Centre. Retrieved 2020-04-12, from http://www.betterlimbs.com/{_}uls/resources/downloads/Transfemoral{_}Manual-2.pdf

Browner, B. D. (2009). *Skeletal Trauma: Basic Science, Management, and Reconstruction* (Vol. 1). Elsevier Health Sciences.

Clinicalkey. (2020). *Stump and Prosthesis Care.* Retrieved 2020-04-12, from https://www.clinicalkey.com/{#}!/content/patienthandout/5-s2.0-peExitCareDIStumpandProsthesisCareen

Datta, D., & Atkinson, G. (2009). *Amputation, Rehabilitation and Prosthetic Developments.* Elsevier, Edinburgh, UK.

Devinuwara, K., Dworak-Kula, A., & O'Connor, R. J. (2018). Rehabilitation and Prosthetics Post-Amputation. *Orthopaedics and Trauma*. doi: 10.1016/j.mporth.2018.05.007

Dillingham, T. R., & Pezzin, L. E. (2005). Postacute Care Services Use for Dysvascular Amputees: A Population-Based Study of Massachusetts. *American Journal of Physical Medicine & Rehabilitation*, *84*(3), 147–152.

Espressif. (2020). *Esp32 series datasheet.* Retrieved 2020-08-23, from https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

Google. (2020, Sep). *Android (go edition).* Android. Retrieved 2020-06-01, from https://www.android.com/versions/go-edition/

Gottschalk, F. (1999). Transfemoral Amputation: Biomechanics and Surgery. *Clinical*

Orthopaedics and Related Research, *361*, 15–22.

Habib, M., Thomas, D., et al. (1986). Chi-square goodness-if-fit tests for randomly censored data. *The Annals of Statistics*, *14*(2), 759–765.

Hagberg, E., Berlin, O. K., & Renstrom, P. (1992, dec). Function after through-knee compared with below-knee and above-knee amputation. *Prosthetics and Orthotics International*, *16*(3), 168–173. doi: 10.3109/03093649209164336

Handson Technology. (2017, May). Handson Technology. Retrieved 2020-06-05, from https://www.handsontec.com/dataspecs/module/Rotary%20Encoder.pdf

Hayes, D. D. (2003). How to wrap an above-the-knee amputation stump. *Nursing2018*, *33*(1), 70.

Henrot, P., Stines, J., Walter, F., Martinet, N., Paysant, J., & Blum, A. (2000). Imaging of the Painful Lower Limb Stump. *Radiographics*, *20*(suppl_1), S219–S235.

Hussain, S., Shams, S., & Khan, S. (2019). *Impact of Medical Advancement: Prostheses*. doi: 10.5772/intechopen.86602

Isherwood, P., Robertson, J., & Rossi, D. (1975). Pressure measurements beneath below-knee amputation stump bandages: Elastic bandaging, the puddifoot dressing and a pneumatic bandaging technique compared. *British Journal of Surgery*, *62*(12), 982–986.

Janchai, S., Boonhong, J., & Tiamprasit, J. (2008, oct). Comparison of Removable Rigid Dressing and Elastic Bandage in Reducing the Residual Limb Volume of below Knee Amputees. *Journal of the Medical Association of Thailand*, *91*, 1441–1446.

Kalra, A., et al. (2017). Decoding the Bland–Altman plot: basic review. *Journal of the Practice of Cardiovascular Sciences*, *3*(1), 36.

Kapp, S. L. (2000). Decoding the Bland–Altman plot: Basic review. *Physical Medicine and Rehabilitation Clinics*, *11*(3), 569–584.

Lazzarini, P. A., Clark, D., & Derhy, P. H. (2011). What are the major causes of lower limb amputations in a major Australian teaching hospital? The Queensland Diabetic Foot Innovation Project, 2006–2007. *Journal of Foot and Ankle Research*, *4*(S1), O24.

Lorraine, O., Cake, N., & Badger, V. (1957). How To Bandage An A.K. Amputation Stumpe. *Orthopedic and Prosthetic Appliances Journal*, *8*.

Manella, K. J. (1981). Comparing the Effectiveness of Elastic Bandages and Shrinker Socks for Lower Extremity Amputees. *Physical Therapy*, *61*(3), 334–337.

Margolis, D. J., Gupta, J., Thom, S. R., Townsend, R. R., Kanetsky, P. A., Hoffstad, O., ... Mitra, N. (2013). Diabetes, lower extremity amputation, loss of protective sensation, and neuronal nitric oxide synthase associated protein in the Chronic Renal Insufficiency Cohort study. *Wound Repair and Regeneration*, *21*(1), 17–24.

May, B. J. (1964). Stump Bandaging of the Lower-Extremity Amputee. *Orthopedic Prosthetic Appliance Journal*, *44*, 808–814.

Moroz, A. (2017, Jul). *Leg Amputation Rehabilitation.* MSD Manuals. Retrieved 2020-03-15, from https://www.msdmanuals.com/professional/special-subjects/rehabilitation/leg-amputation-rehabilitation

Muilenburg, A. L., & Wilson, A. B. (1996). Dankmeyer, Inc. Retrieved from https://www.dankmeyer.com/assets/manuals/akindex.htm

O'Keeffe, B., & Rout, S. (2019, jan). Prosthetic Rehabilitation in the Lower Limb. *Indian Journal of Plastic Surgery : Official Publication of the Association of Plastic Surgeons of India*, *52*(1), 134–143. doi: 10.1055/s-0039-1687919

Oliphant, T. E. (2006). *A guide to NumPy* (Vol. 1). Trelgol Publishing USA.

O'Sullivan, S. B., Schmitz, T. J., & Fulk, G. (2019). *Physical Rehabilitation.* FA Davis.

Ottobock. (2020, Dec). *Info for lower limb amputees and their families.* Author. Retrieved 2020-03-17, from https://www.ottobockus.com/prosthetics/info-for-new-amputees/information-for-lower-limb-amputees-and-their-families/

Pascale, B. A., & Potter, B. K. (2014). Residual Limb Complications and Management Strategies. *Current Physical Medicine and Rehabilitation Reports*, *2*(4), 241–249.

Paternò, L., Ibrahimi, M., Gruppioni, E., Menciassi, A., & Ricotti, L. (2018). Sockets for Limb Prostheses: A Review of Existing Technologies and Open Challenges. *IEEE Transactions on Biomedical Engineering*, *65*(9), 1996–2010.

Potgieter, K. (2018). *2500 leg amputations each year in KZN.* Retrieved from http://www.kznhealth.gov.za/mediarelease/2018/2500-leg-amputations-each-year-in-kzn-12112018.htm

Sidawy, A. N., & Perler, B. A. (2018). *Rutherford's Vascular Surgery and Endovascular Therapy, E-Book.* Elsevier Health Sciences.

Smith, M. (2017). *The easiest way to use Python in your Android app.* Retrieved 2020-06-25, from https://chaquo.com/chaquopy/

Standl, E. (2019). *Global Statistics on Diabetes.* European Society of Cardiology. Retrieved 2020-02-27, from https://www.escardio.org/Education/Diabetes-and-CVD/Recommended-Reading/global-statistics-on-diabetes

Varghese, G., Hindle, P., Zilber, S., Perry, J. E., & Redford, J. (1981). Pressure Applied By Elastic Prosthetic Bandages: A Comparative Study. *Orthotics and Prosthetics*, *35*(4), 30–36.

Visser, C. (1998). Knowledge and skill of patients with regard to amputation stump bandaging, prior to a prosthesis. *South African Journal of Physiotherapy*, *54*(3), 8–10.

Weisstein, E. W. (2015, Oct). *Circular Segment.* Retrieved 2020-04-12, from https://mathworld.wolfram.com/CircularSegment.html

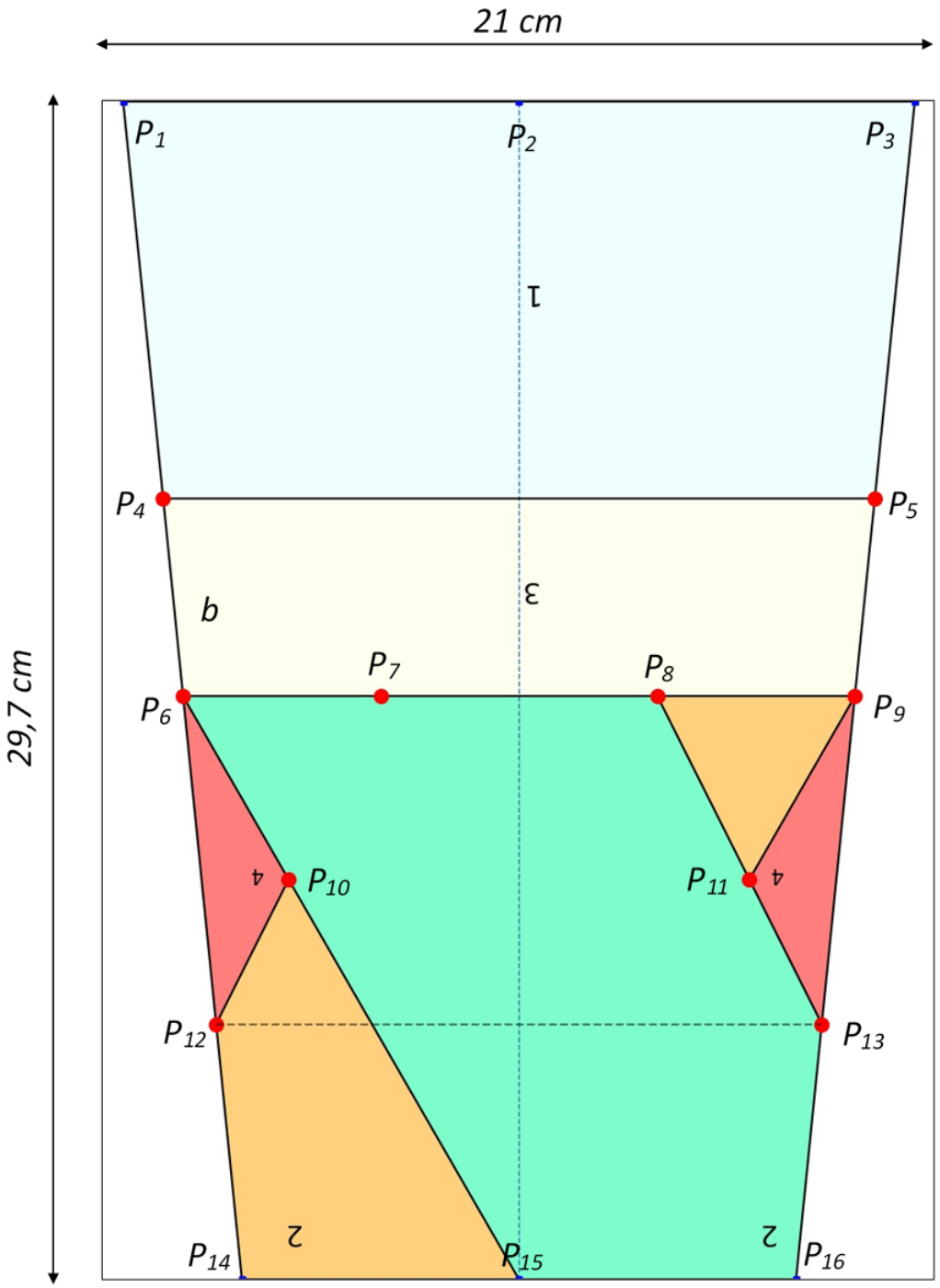
Weisstein, E. W. (2020). *Line-Line Intersection.* Retrieved 2020-04-12, from https://mathworld.wolfram.com/Line-LineIntersection.html

Wilson, j. (2020). *Pattern for a Truncated Cone: One Solution .* The University

of Georgia. Retrieved 2020-03-02, from http://jwilson.coe.uga.edu/emt725/CarlCone/Solution.html

Ziegler-Graham, K., MacKenzie, E. J., Ephraim, P. L., Travison, T. G., & Brookmeyer, R. (2008). Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050. *Archives of Physical Medicine and Rehabilitation*, *89*(3), 422–429.

A Bandaging Template



B Algorithms

B.1 Template Generation Algorithm

Python Script 1 Bandaging Template Script

function SIDELENGTH(C_1, C_2, H)
    $R = \frac{C_2}{2\pi}$
    $r = \frac{C_1}{2\pi}$
    $h = \frac{rH}{R-r}$
    $a = \sqrt{h^2 + r^2}$
    $b = \sqrt{(h+H)^2 + R^2} - a$
    return b
end function

function PLOTLINE(P_1, P_2)
    Plot line between two points: P_1 and P_2
    Scale x coordinate units to 2.54 cm
    Scale y coordinate units to 2.54 cm
end function

function EQUATIONOFLINE(P_1, P_2)
    Construct an array to store the x and y coordinates for each point
    Assign a symbol variable for x
    Calculate the least-squares solution to determine gradient m and constant c
    return $mx + c$
end function

function SOLVEFOR$x(y)$:
    Solve for x, by substituting y into the distance formula, where the distance is
    equal to 10
    return x
end function

function INTERSECTIONOFTWOLINES($x1, y1, x2, y2, x3, y3, x4, y4$):
    Find the intersection between two lines using the Line-Line Intersection method
    $P_x = \frac{(x1*y2-y1*x2)*(x3-x4)-(x1-x2)*(x3*y4-y3*x4))}{(x1-x2)*(y3-y4)-(y1-y2)*(x3-x4)}$
    $P_y = \frac{(x1*y2-y1*x2)*(y3-y4)-(y1-y2)*(x3*y4-y3*x4)}{(x1-x2)*(y3-y4)-(y1-y2)*(x3-x4)}$
    return P_x and P_y
end function

B.2 Bandage Length Prediction Algorithm

Python Script 2 Bandage Length Prediction Script

function ArcLength(C_4)
    $H = 6.65$
    $W = \frac{C_4}{\pi}$
    $R = \frac{W^2}{8H} + \frac{H}{2}$
    $\theta = 2 \arcsin \frac{W}{2R}$
    $S = \theta R$
    return S
end function

function Distance(P_1, P_2):
    Linear distance between two points plotted on the bandaging template
    return distance
end function

function Reflected$x$(list):
    Return the reflected x values of existing points
    return updated list
end function

$Step_1 = C_1 \times CompressionFactor$
$Step_2$ = (Distance(P_1, P_{13}) + Distance(P_{12}, P_5) + 2*ArcLength(C_4) - $C_4/6$) × $CompressionFactor$
$Step_3 = (C_2$ + (Distance(P_4, P_{13}) + $C_4) \times CompressionFactor$
$Step_4$ = ArcLength(C_5) × $CompressionFactor$

C_1 = Proximal circumferential measurement
C_2 = Circumferential measurement three
C_3 = Circumferential measurement four
C_4 = Circumferential measurement five
C_5 = Distal circumferential measurement

$CompressionFactor = 0.895$

C Code

C.1 Bandaging Template Script

```
import sys
import matplotlib.pyplot as plt
from numpy import ones,vstack
from numpy.linalg import lstsq
from sympy.solvers import solve
from sympy import Symbol
from scipy.spatial import ConvexHull
import numpy as np
import math
from PIL import Image, ImageChops
import img2pdf


def SideLength(C1, C2, H):
    R = C2/(2*math.pi)
    r = C1/(2*math.pi)
    h = (r*H)/(R-r)
    a = math.sqrt(h**2 + r**2)
    b = math.sqrt((h+H)**2+R**2) - a
    return b


def trim(im):
    bg = Image.new(im.mode, im.size, im.getpixel((0,0)))
    diff = ImageChops.difference(im, bg)
    diff = ImageChops.add(diff, diff, 2.0, -100)
    bbox = diff.getbbox()
    if bbox:
        return im.crop(bbox)


def connect(p1, P3, line):
    if line == 1:
        ax.plot([p1[0], P3[0]], [p1[1], P3[1]], 'k-', zorder=12 ,
   xunits=2.54, yunits=2.54)
    elif line == 2:
        ax.plot([p1[0], P3[0]], [p1[1], P3[1]], 'k--', alpha = 0.5,
   zorder=14 , xunits=2.54, yunits=2.54)


def eqn_line(point1, point2):
```

```
40      points = [point1,point2]
41      x_coords, y_coords = zip(*points)
42      A = vstack([x_coords,ones(len(x_coords))]).T
43      xx = Symbol('xx')
44      m, c = lstsq(A, y_coords)[0]
45      eqn_ln = m*xx + c
46      return eqn_ln
47
48
49  def solve_x(yy):
50      solved = solve(((10.5- xx)**2+(0 -yy )**2)**(1/2) -10 , xx)
51      print('solved: ' + str(solved))
52      if yy.subs({xx:solved[0]}) > 0:
53          return solved[0]
54      else:
55          return solved[1]
56
57
58  def findIntersection(x1,y1,x2,y2,x3,y3,x4,y4):
59      px= ( (x1*y2-y1*x2)*(x3-x4)-(x1-x2)*(x3*y4-y3*x4) ) / ( (x1-x2)*(y3
        -y4)-(y1-y2)*(x3-x4) )
60      py= ( (x1*y2-y1*x2)*(y3-y4)-(y1-y2)*(x3*y4-y3*x4) ) / ( (x1-x2)*(y3
        -y4)-(y1-y2)*(x3-x4) )
61      return [px, py]
62
63
64  def colour(pts_c, seg):
65      if seg == 1:
66          pts = np.array(pts_c)
67          hull = ConvexHull(pts)
68          plt.fill(pts[hull.vertices,0], pts[hull.vertices,1],'lightcyan'
        , alpha=0.5)
69
70      if seg == 2:
71          pts = np.array(pts_c)
72          hull = ConvexHull(pts)
73          plt.fill(pts[hull.vertices,0], pts[hull.vertices,1],'
        mediumspringgreen', alpha=0.5)
74
75      if seg == 3:
76          pts = np.array(pts_c)
77          hull = ConvexHull(pts)
78          plt.fill(pts[hull.vertices,0], pts[hull.vertices,1],'
        lightyellow', alpha=0.5)
79
80      if seg == 4:
81          pts = np.array(pts_c)
```

```
82          hull = ConvexHull(pts)
83          plt.fill(pts[hull.vertices,0], pts[hull.vertices,1],'red',
        alpha=0.5)
84
85      if seg == 5:
86          pts = np.array(pts_c)
87          hull = ConvexHull(pts)
88          plt.fill(pts[hull.vertices,0], pts[hull.vertices,1],'orange',
        alpha=0.5)
89
90
91  xx = Symbol('xx')
92
93  """Starting parameters, upper and bottom circumferences"""
94  ucircum = int(sys.argv[1])
95  bcircum = int(sys.argv[2])
96  h = 29.7
97
98
99  """Axis where template will be plotted"""
100 fig, ax = plt.subplots()
101 plt.gca().set_axis_off()
102 plt.subplots_adjust(top = 1, bottom = 0, right = 1, left = 0, hspace =
        0, wspace = 0)
103 plt.margins(0,0)
104
105
106 """Reduce measurements to 1/3 to fit on A4 page"""
107 uwidth = ucircum/3
108 bwidth = bcircum/3
109
110
111 """Plot points of template perimeter"""
112 P1 = [0, h]
113 P2 = [10.5, h]
114 P3 = [21, h]
115 boundary = SideLength(bcircum, ucircum, h)
116 P14x = solve(((xx-0)**2+(0-29.7)**2)**(1/2) - boundary , xx)
117 P14 = [float(abs(P14x[0])), 0]
118 P16x = solve(((xx-21)**2+(0-29.7)**2)**(1/2) - boundary , xx)
119 P16 = [21- float(abs(P14x[0])), 0]
120 P15 = [10.5, 0]
121
122
123 """Perimeter points as blue squares"""
124 ax.plot([P1[0], P3[0], P14[0], P16[0], P15[0], P2[0]], [P1[1], P3[1],
        P14[1], P16[1], P15[1] , P2[1]], 'bs', zorder = 14, markersize = 5 )
```

```
125
126
127 """Draw Boundary"""
128 connect(P1,P3,1)
129 connect(P1,P14,1)
130 connect(P3,P16,1)
131 connect(P14,P16,1)
132
133
134 """Solve eqn line 1 and get x1 and y1 coordinates to plot"""
135 line_1 = eqn_line(P1, P14)
136 print("Eqn of line 1: "+ str(line_1))
137 solved_x1 = solve_x(line_1)
138 print("x1: " + str(solved_x1))
139 solved_y1 = line_1.subs({xx:solved_x1})
140 print("y1: " + str(solved_y1))
141 print(" ")
142
143
144 """Solve eqn line 2 and get x2 and y2 coordinates to plot"""
145 line_2 = eqn_line(P3, P16)
146 print("Eqn of line 2: "+ str(line_2))
147 solved_x2 = solve_x(line_2)
148 print("x2: " + str(solved_x2))
149 solved_y2 = line_2.subs({xx:solved_x2})
150 print("y2: " + str(solved_y2))
151 print(" ")
152
153
154 """Plot points P4 - P13"""
155 P12 = [solved_x1, solved_y1]
156 ax.plot(solved_x1, solved_y1, 'ro', zorder=13, markersize = 10 )
157
158 P13 = [solved_x2, solved_y2]
159 print(P13[1])
160 ax.plot(solved_x2, solved_y2, 'ro', zorder=13, markersize = 10 )
161
162 p4 = solve(line_1-(h-10), xx)
163 P4 = [p4[0], h-10]
164 print("P4 ", P4)
165 ax.plot(P4[0], P4[1], 'ro', zorder=13, markersize = 10 )
166
167 p5 = solve(line_2-(h-10), xx)
168 P5 = [p5[0], h-10]
169 print("P5 ", P5)
170 ax.plot(P5[0], P5[1], 'ro', zorder=13, markersize = 10 )
171
```

```
172 p6 = solve(line_1-(h-10-5), xx)
173 P6 = [p6[0], h-10-5]
174 ax.plot(P6[0], P6[1], 'ro', zorder=13, markersize = 10 )
175
176 p9 = solve(line_2-(h-10-5), xx)
177 P9 = [p9[0], h-10-5]
178 ax.plot(P9[0], P9[1], 'ro', zorder=13, markersize = 10 )
179
180 P7 = [(10.5 + P14[0] )/2, h-10-5]
181 ax.plot(P7[0], P7[1], 'ro', zorder=13, markersize = 10 )
182 print("P7 ", P7)
183
184 P8 = [(10.5 + P16[0] )/2, h-10-5]
185 ax.plot(P8[0], P8[1], 'ro', zorder=13, markersize = 10 )
186 print("P8 ", P8)
187
188
189 connect(P4,P5,1)
190 connect(P6,P9,1)
191 connect(P15,P6,1)
192
193
194 """Find points of intersection for triangle pieces"""
195 P10 = findIntersection(P15[0],P15[1],P6[0],P6[1],P12[0],P12[1],P7[0],P7
    [1])
196 ax.plot(P10[0], P10[1], 'ro', zorder=13, markersize = 10 )
197 P11 = findIntersection(P15[0],P15[1],P9[0],P9[1],P13[0],P13[1],P8[0],P8
    [1])
198 ax.plot(P11[0], P11[1], 'ro', zorder=13, markersize = 10 )
199
200
201 connect(P13,P8,1)
202 connect(P12,P10,1)
203 connect(P11, P9,1)
204 connect(P12,P13,2)
205
206
207 """Colour each segment"""
208 colour([P1,P3,P4,P5], 1)
209 colour([P6,P8,P13,P15,P16], 2)
210 colour([P4,P5,P6,P9], 3)
211 colour([P12,P6,P10],4)
212 colour([P13,P9,P11],4)
213 colour([P9,P8,P11],5)
214 colour([P14,P12,P15,P10],5)
215
216
```

```
"""Plot the numbered steps on the template, steps 1-6"""
plt.text(P2[0], P2[1]-5, "1", size=20, rotation=180)
plt.text(P2[0], P2[1]-12.5, "3", size=20, rotation=180)
plt.text(P16[0]-1, P16[1]+1, "2", size=20, rotation=180)
plt.text(P14[0]+1, P14[1]+1, "2", color = 'k', size=20, rotation=180)
plt.text(P10[0]-1, P10[1], "4", color = 'k', size=15, rotation=180)
plt.text(P11[0]+0.5, P11[1], "4", color = 'k', size=15, rotation=180)
plt.axis('scaled')
plt.ylim(0, 29.7)
plt.xlim(0,21)


"""Plot centre of template"""
plt.axvline(x=21/2, ymin=0, ymax = 29.7, linewidth=1, linestyle='--')
#plt.axis('off')
fig.set_size_inches(8.27,11.69)
fig.savefig('out4.jpg', bbox_inches='tight', pad_inches=0)


"""Crop image, remove unused space, then convert to pdf for printing"""
im = Image.open("out4.jpg")
im = trim(im)
im = im.convert("RGB")
im.save('out5.jpg')


with open("EDITTemplate.pdf","wb") as f:
    f.write(img2pdf.convert('out5.jpg'))
```

C.2 Predicted Bandage Length Script

```
from matplotlib import pyplot as plt
import numpy as np
import math
from numpy.linalg import lstsq
from sympy.solvers import solve
from sympy import Symbol
from numpy import ones,vstack


def main(m1, m2, m3, m4, m5):


    def Neg(lst):
        lst = np.array(lst)
        return list(-lst)


    def dist(x1,y1,x2,y2):
        distance = (((x1-x2)**2)+((y1-y2)**2) )**(1/2)
        return distance


    def eqn_line(point1, point2):
        points = [point1,point2]
        x_coords, y_coords = zip(*points)
        A = vstack([x_coords,ones(len(x_coords))]).T
        xx = Symbol('xx')
        m, c = lstsq(A, y_coords)[0]
        eqn_ln = m*xx + c

        return eqn_ln


    def plot_template(points_y, eqn_line):
        xx = Symbol('xx')
        x_vals = []
        for j in range(1,4):
            solve_x = solve(eqn_line - points_y[j] , xx)
            x_vals.append(solve_x[0])
            ax.plot(solve_x[0], y[j], 'bo', zorder=13, markersize = 5 )



        return x_vals

```

```
46
47      def arc_length(circum_4):
48          H = 6.65
49          r = circum_4/(math.pi)
50          R = (r**2/(8*H))+(H/2)
51          print("R: ", R)
52          print("r: ", r)
53          theta = 2*math.asin(r/(2*R)) #in radians
54          print("theta=" ,theta)
55          perimeter = theta*R
56          print("perimeter = ", perimeter)
57          arc = perimeter/2
58
59          return arc
60
61
62      fig, ax = plt.subplots()
63
64      C1_circum = m1
65      C2_circum = m2
66      C3_circum = m3
67      C4_circum = m4
68      C5_circum = m5
69
70      compression = 0.895
71      arc_cm = arc_length(C4_circum)
72
73
74      x1 = [C1_circum/4, C2_circum/4, C3_circum/4, C4_circum/4, C5_circum
        /4]
75      x2 = [C1_circum/6, C5_circum/6]
76      y = [29.7, 19.7, 14.7, 6.65, 0]
77      y2 = [29.7, 0]
78
79      ax.plot(x1, y, 'ro', zorder=13, markersize = 5 )
80      ax.plot(Neg(x1), y, 'ro', zorder=13, markersize = 5 )
81
82      ax.plot(x2, y2, 'bo', zorder=13, markersize = 5 )
83      ax.plot(Neg(x2), y2, 'bo', zorder=13, markersize = 5 )
84
85      xx = Symbol('xx')
86      line_1 = eqn_line([x2[0],y[0]], [x2[1],y[4]])
87      line_2 = eqn_line([-x2[0],y[0]], [-x2[1],y[4]])
88
89      diag_1 = eqn_line([-x2[0],y[0]], [x2[1],y[4]])
90      diag_x1 = solve(diag_1 - 6.65 , xx)
91
```

```
92
93      print( solve(diag_1 - 6.65 , xx))
94
95      newx = plot_template(y, line_1)
96      newx.extend(plot_template(y, line_2))
97
98
99      L1 = C1_circum* compression
100
101     L2 = (dist(-x1[0], y[0], diag_x1[0], y[3]) + 2*arc_cm) *
        compression
102
103     L3 =  (2*arc_cm  + (dist(newx[0], y[1], -diag_x1[0], y[3])) -
        C5_circum/6) * compression
104
105     L4 = (C2_circum + C4_circum + (dist(newx[2], y[3], newx[3], y[1])))
106     *compression
107
108     L5 = (2*arc_cm) * compression
109
110     print("Step 1: " + str(L1) + " cm")
111     print("Step 2: " + str(L2 + L3) + " cm")
112     print("Step 3: follow the yellow band")
113     print("Step 4: " + str(L4) + " cm")
114     print("Step 5: " + str(L5) + " cm")
115
116     llist = [L1, L2, L3, L4, L5]
117
118     print(str(L1))
119     print(L2)
120     print(llist)
121
122
123 if __name__ == "__main__":
124     main(int(sys.argv[1]), int(sys.argv[2]), int(sys.argv[3]),
125     int(sys.argv[4]),int(sys.argv[5]))
```

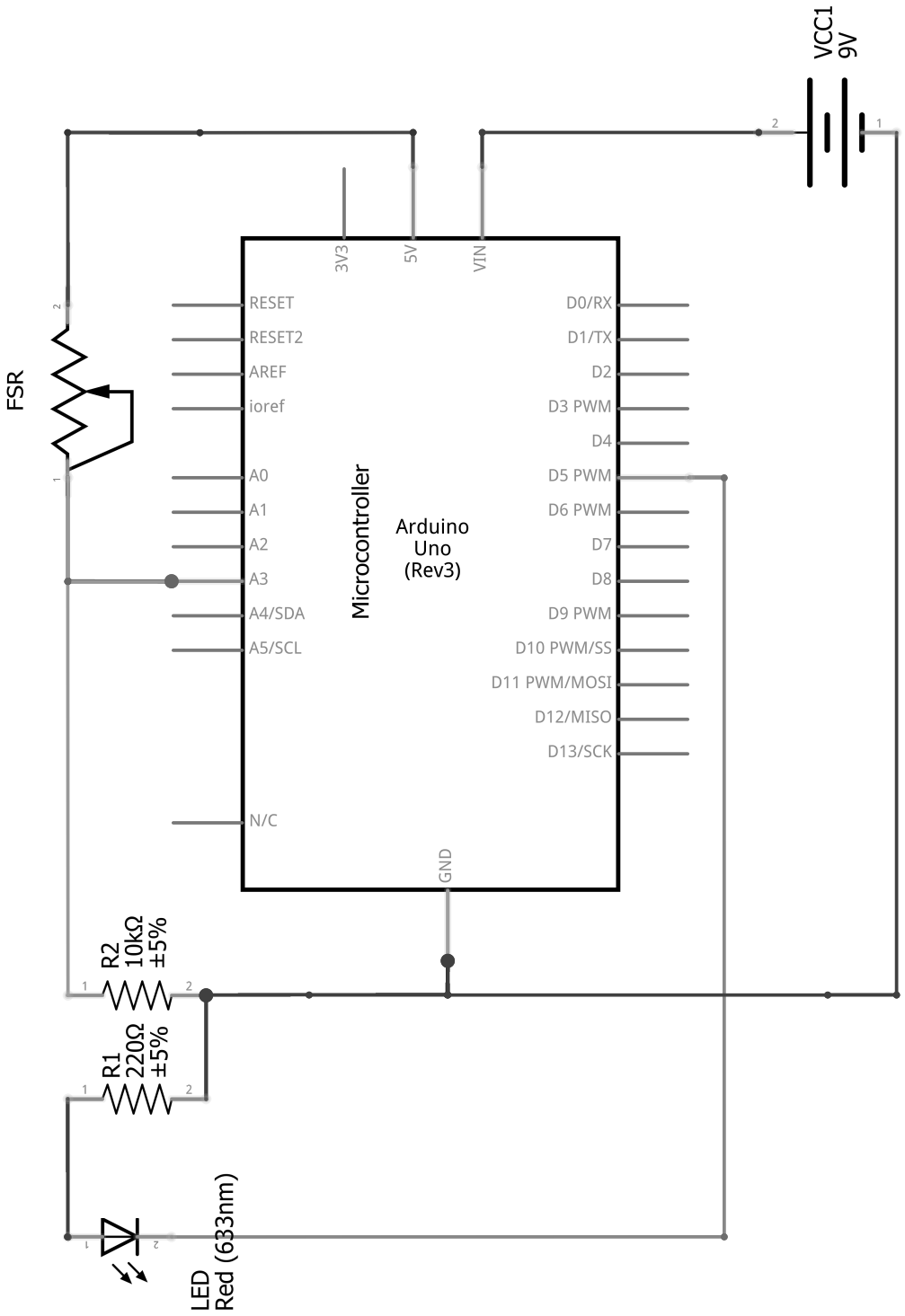
C.3 BUD Code

```
#include "BluetoothSerial.h"
#include "AiEsp32RotaryEncoder.h"
#include "Arduino.h"
#define ROTARY_ENCODER_A_PIN 32
#define ROTARY_ENCODER_B_PIN 21
#define ROTARY_ENCODER_BUTTON_PIN 25
#define ROTARY_ENCODER_VCC_PIN -1

AiEsp32RotaryEncoder rotaryEncoder = AiEsp32RotaryEncoder(
    ROTARY_ENCODER_A_PIN, ROTARY_ENCODER_B_PIN,
    ROTARY_ENCODER_BUTTON_PIN, ROTARY_ENCODER_VCC_PIN);

#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and
    enable it
#endif

BluetoothSerial SerialBT;

int test_limits = 2;
int counter = 1;

void rotary_Button() {
  SerialBT.print(">>");
  rotaryEncoder.reset();
  delay(10);
}

void rotary_loop() {
  if (rotaryEncoder.currentButtonState() == BUTTON_RELEASED) {
    rotary_Button();
  }

  int16_t encoderDelta = rotaryEncoder.encoderChanged();

  if (encoderDelta == 0)return;

  if (encoderDelta!=0) {
    int16_t encoderValue = rotaryEncoder.readEncoder();
    if (encoderValue < 0){
      encoderValue = 0;
    }
    Serial.print(String(encoderValue/2));
    SerialBT.print(String(encoderValue/2));
    delay(10);
```

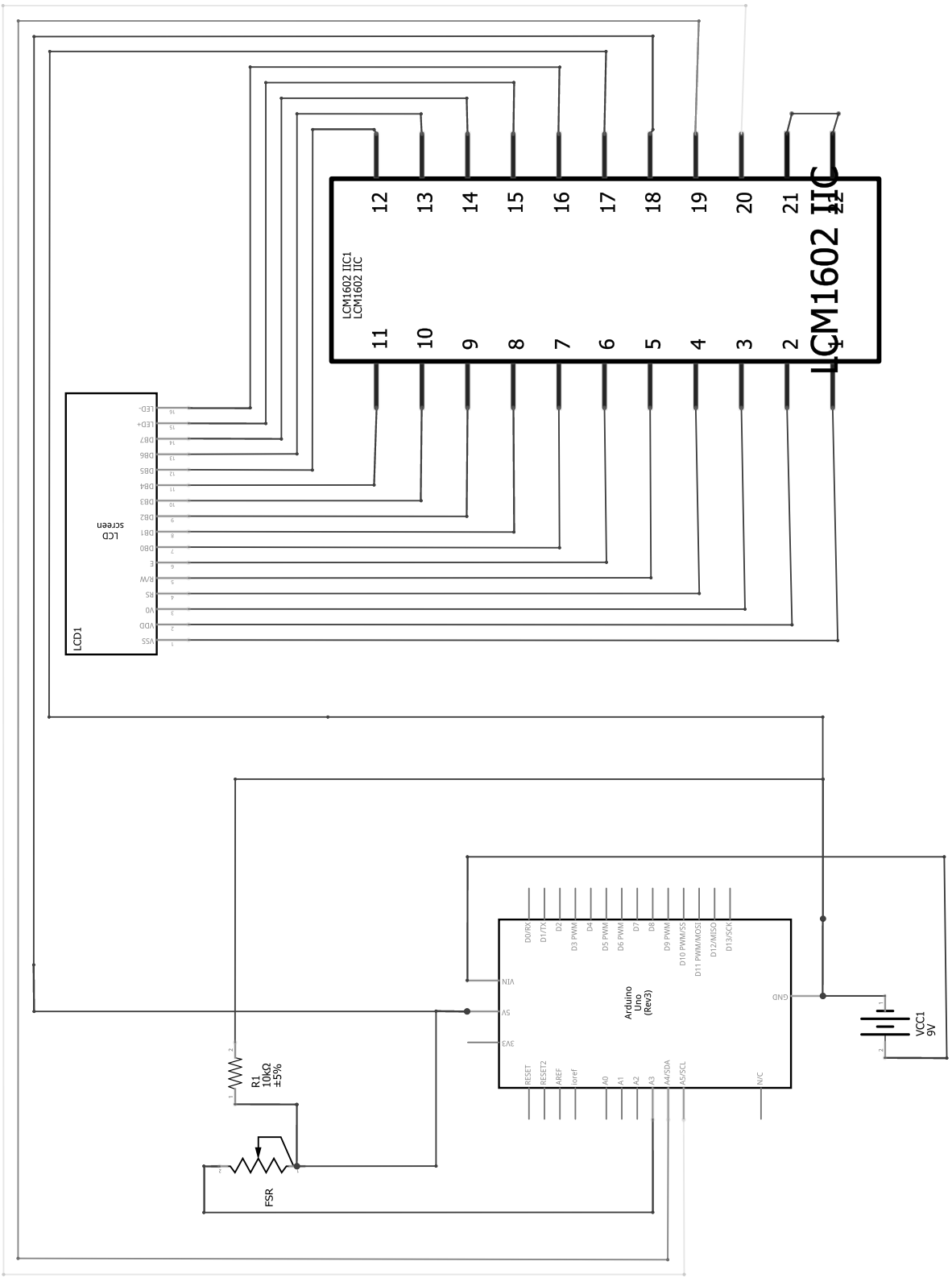
```
43   }
44 }
45
46
47 void setup() {
48   Serial.begin(115200);
49   rotaryEncoder.begin();
50   rotaryEncoder.setup([]{rotaryEncoder.readEncoder_ISR();});
51   rotaryEncoder.setBoundaries(-20, 1000, true);
52   SerialBT.begin("BUD");
53 }
54
55 void loop() {
56     rotary_loop();
57     if (Serial.available()) {
58     SerialBT.write(Serial.read());
59     }
60
61   if (SerialBT.available()) {
62     String string = SerialBT.readString();
63     if(string.equals("reset")){
64       rotaryEncoder.reset();
65     }
66   }
67
68   delay(20);
69
70   if (millis()>20000) rotaryEncoder.enable ();
71
72 }
```

D Circuit Diagrams

D.1 Testing Rig

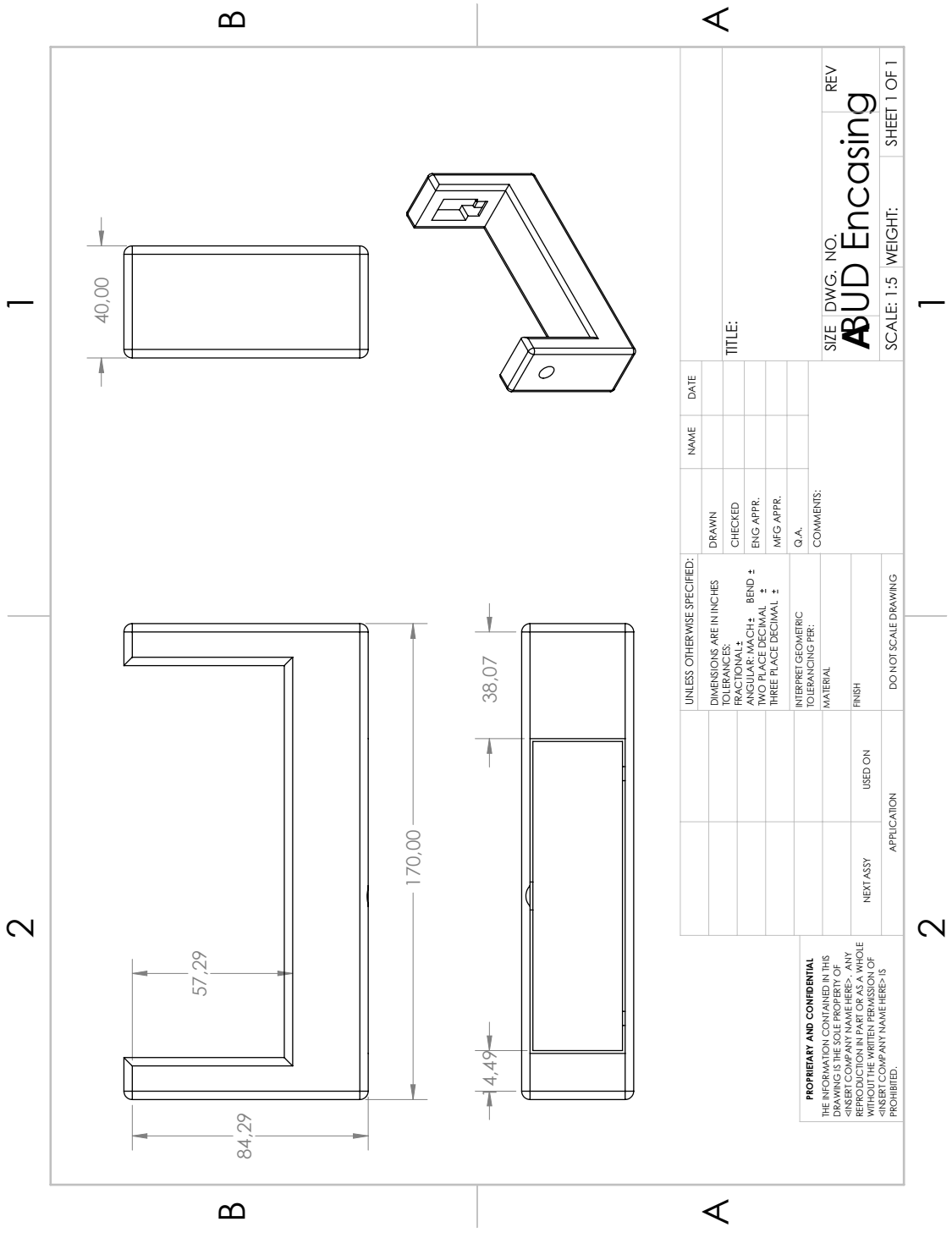


D.2 Pressure Pad



E BUD Drawings

E.1 Encasing Drawing



E.2 Rotary Rod Drawing