ROBOTICS AND MECHATRONICS

CONTROLLING SUNRAM 7 WITH A SPACE MOUSE

T.H.J. (Thijs) Vissers

BSC ASSIGNMENT

Committee:
dr. ir. D. Dresscher
dr. V. Groenhuis, MSc
dr. ir. H. Wormeester

July, 2024

013RaM2024
Robotics and Mechatronics
EEMCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | TECHMED CENTRE
UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Summary

MRI-guided biopsies are essential for diagnosing lesions in the breast. These biopsies are time-consuming and require well-trained staff.
The Sunram 7 is an MR-safe biopsy robot that aims to improve the quality of biopsies, or make them more accessible by needing less training to achieve the same quality level.

This report will focus on controlling the Sunram 7 in a way that is quick to set up, accurate within 2 mm, and easy to use for inexperienced people. Furthermore, it was tested whether there is an optimum control velocity in terms of speed versus accuracy.
The final design consists of the 3D SpaceMouse connected to a Teensy 4.1 microcontroller. The microcontroller handles communication between the SpaceMouse and the controller board of the Sunram 7. It also allows for changing the control velocity and imposing physical constraints in the software.

Tests were carried out to measure how long it takes for the system to be assembled completely and how much time it takes to start up when the power is turned off and on. Secondly, the accuracy of the control method was tested. Moreover, a survey was carried out to test the usability of the system.

It was found that the system could be assembled completely in an average of 65 seconds, while the mean start-up time was 2.1 s. The mean error at the lowest velocity setting was 0.43 mm while the average overall error was 0.75 mm. Participants who tried the system gave an average SUS score of 78.8 which indicates good to excellent usability.
If one wants the absolute most accuracy, velocity setting 3 (6.3 mm/s) was better. But if a little bit less accuracy can be tolerated, setting 9 (19 mm/s) yields a good trade-off between accuracy and movement time.

Overall, the Sunram 7 can be controlled in a relatively intuitive way using the 3D SpaceMouse.

Contents

1 Introduction

1.1 General

In 2023, more than 15,000 women were diagnosed with breast cancer in The Netherlands [1]. The standard way of arriving at these diagnoses is to first make a mammogram. This is an x-ray image of the breasts. Mammography can be used as a screening tool, as well as for diagnostic purposes [2]. After this procedure, the doctor can either diagnose a benign tumour or further research is needed. Depending on the classification, this further research consists of waiting and doing an additional screening after 6 months, or performing a biopsy [3]. In challenging cases or when it is known there is a higher risk of cancer due to genetic reasons, the choice can be made to do an MRI scan. With the information from the MRI scan, an MRI-assisted vacuum biopsy can be performed if necessary [1]. The Sunram 7 is an MR-safe robot that can do MRI-assisted biopsies when controlled by a doctor (Figure 1.1) The robot is able to do directional core needle biopsy (CNB) on the breast of a patient. This means that the needle will take a portion of tissue and put it inside the needle. In this report, a new method of controlling this robot will be explored.

Figure 1.1: The Sunram 7 robot.

1.2 State of the art

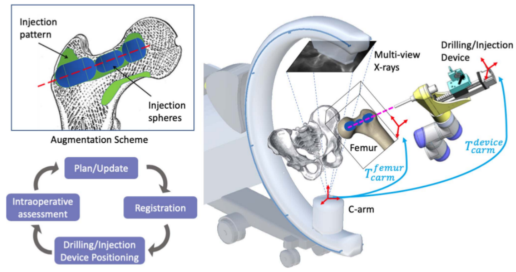
Surgical robots have begun to be more present in operating rooms. Major advancements have been in the development of haptic feedback mechanisms for the surgeon, as well as 3-dimensional visualization techniques and head-mounted displays [4].

In using haptic feedback control, it is necessary to have some kind of physical force sensing. This can be in the form of fibre optic force sensing [5] (Figure 1.3a). This allows the doctor to have a feel for how the robot behaves in the tissue.

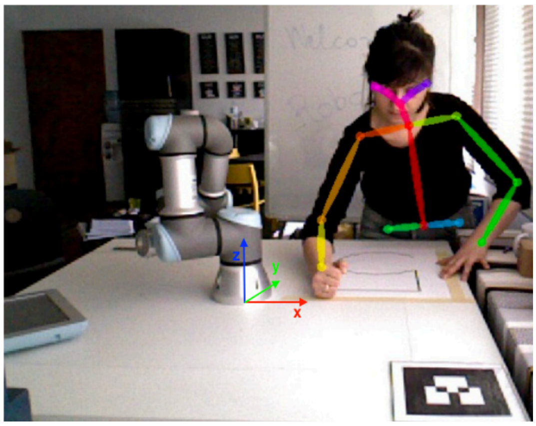
An alternative method of controlling robots is using an optical sensor. The advantage of this is that the doctor is free to make movements in 3D space without being restricted much by the input device. The set-up shown in Figure 1.2b uses wearable devices as well as an optical camera to detect the movement of the person.

Other possibilities that become available with the use of robots are to perform the biopsies within the MRI bore[6], which is the tunnel-like structure of the MRI machine in which the patient is positioned. This method saves time and allows for a quick feedback loop in which the position of the needle can be verified. The robot is shown in Figure 1.3b. A drawback of this method however, is that the angle of the needle cannot be controlled.

It is a possibility to make a model of the part that has to undergo surgery. A possible method for this is doing a CT scan. The CT scan is done by letting an X-ray machine take images at multiple different angles to construct a 3D model. Image recognition can be used on this model to calculate where the end effector has to be placed. In this case for a femoroplasty, a type of surgery to prevent hip fractures in the elderly [7].

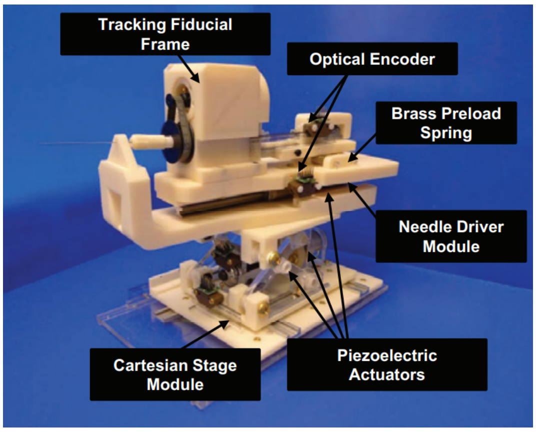


(a) A 3D model of how the hip can be characterized to control the end effector. [7]

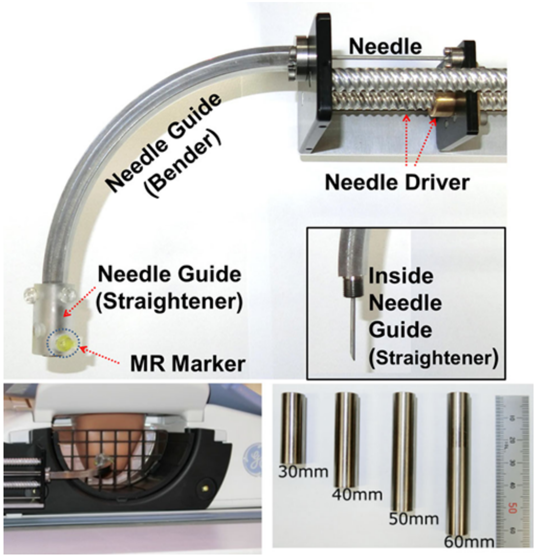


(b) RGB camera that is able to detect movement of pencil. [8]

Figure 1.2: State of the art of input for surgical robots.



(a) Physical prototype of 6-DOF piezoelectric actuated needle placement robot. [5]



(b) Bendable needle that is able to operate inside the bore. [6]

Figure 1.3: State of the art of MR-safe robots.

While it is not suitable for MRI-assisted biopsies, the Da Vinci® robot is a leading advancement in biomedical robots. It aims to be minimally invasive, which can lead to less pain and faster recovery times [9]. Multiple arms with tools allow the surgeon to perform the surgeries more efficiently than what would otherwise be possible. The system is able to let the surgeon see better using 3D high-definition views and allows its instruments to be used like a human hand but with a greater range of motion [10].

Figure 1.4: The da Vinci 5 surgical system. [10]

1.3 Possible problems with automation

This paper will focus just on the control and feasibility of a new control method (Section 1.4). However, it is good to note that taking away the human factor does not need to be the best solution. While visiting the hospital to look at how MRI assisted vacuum biopsies were performed, it was noticed that the human element can be very helpful. To give an example, a patient in distress wanted to abort the procedure because she was in too much pain. The doctor was able to calm her down and convince her to continue with the biopsy albeit a slightly sub-optimal one. This gave some insight in how much one should want to automate when dealing with real people in changing circumstances. Trade-offs can be made for this, for example, the robot only places the needle guide, while a doctor performs the biopsy. These questions also have to be taken into consideration.

1.4 Scope

The Sunram 7 robot is already built completely and there are multiple other ways ways in which it can be controlled already. Examples of these are haptic control or optical tracking. One drawback of these methods is they require significant time to set up. This means that there is a need for a new control device that is easier and more intuitive to use than sliders or joysticks, but also quick to set-up and start.
This research paper will focus on controlling the Sunram 7 robot in an accurate, user-friendly and fast way using a 6 DOF joystick [11]. This 6 DOF joystick, the 3D SpaceMouse, will be connected to a microcontroller [12]. This microcontroller will communicate with a separate controller board that controls the valves that regulate the pressurized air and thus the stepper motors (see Section 3.1). When this is accurate enough, the usability for people with minimal training was looked at. A survey was be carried out to test and improve the performance of the control method.

1.5 Hypothesis

Accuracy and set-up times can be measured in an objective manner. Ease of use is slightly different. A hypothesis regarding the ease of use was tested. The hypothesis is as follows "There is an optimum speed-accuracy trade-off to control the Sunram 7". It is expected that when the control speed is high, the accuracy will be low. When the speed is decreased, it is likely that accuracy will increase. The expectation is that this is true up to a certain point, when the accuracy will no longer increase when lowering control speed. That point could be considered an optimum. This was tested by combining objective and subjective methods.

Accuracy and set-up times can be measured in an objective manner. Ease of use is slightly different. A hypothesis regarding the ease of use was tested. The hypothesis is as follows "There is an optimum speed-accuracy trade-off to control the Sunram 7". It is expected that when the control speed is high, the accuracy will be low. When the speed is decreased, it is likely that accuracy will increase. The expectation is that this is true up to a certain point, when the accuracy will no longer increase when lowering control speed. That point could be considered a Pareto optimum, where one parameter cannot be improved without another one worsening. This was tested by combining objective and subjective methods.

2 Background & Literature review

2.1 Sunram 7

The manipulator part of the Sunram 7, which is the robot on its base plate without the titanium needle, is MR-safe. This is because the parts are 3D-printed and made of materials such as polylactic acid (PLA) or other non-metallic, non-magnetic and non-conductive materials such as plastic for the tubing. There is only one metallic part, which is the titanium needle. This is an MR-conditional part, meaning that it is allowed to go into an MRI scanner given that it is tested for RF-induced, heating, image artefacts, force and torque [13]. For the robot to be used in actual real-life scenarios, it has to be clinically tested on MR safety regardless of ferromagnetism of the materials.

It is driven by stepper motors that are actuated by pulses of pressurized air. This means that it can do a biopsy within the MRI scanner making it able to verify the position of the needle while the biopsy is being performed. Using the Sunram 7 could improve the quality of biopsies, or make them more accessible by needing less training to achieve the same quality of biopsy, since current MRI-guided biopsies are time-consuming and complex procedures that require experienced and well-trained staff [14].

The Sunram 7 can move in five different directions that will be more extensively covered in Section 3.3. It consists of a base plate on which the manipulator part can move sideways, turn left or right, move up and down and tilt forwards or backwards (Figure 3.3). These are the main movement directions, but there is also the option to insert or withdraw the needle. The needle is a biopsy gun that can be inserted into the tissue and then 'shoot' forward to take a sample piece [15].

2.2 Speed-accuracy trade-off

To make the control method as effective as possible, a trade-off between the speed of the end-effector and the desired accuracy has to be made. Some research has already been done on this matter.

In one study participants were asked to make a horizontal hand movement on a kymograph [16]. This was done at multiple speeds, with both their left and right hand. The experiment was also conducted blind to evaluate the factor of using eyesight. The findings were that the error increases with the speed at which the exercise is carried out. This turned out to not be simple linear behaviour. When using the right hand, the error increased below proportional, while when using the left hand the error increased more than proportional.

Another experiment was to let subjects draw a line between points A and B, which were 50 cm apart. They were given different times to complete the action. Woodworth found that most of the accuracy of the movement is obtained due to current visual control of the movement, consisting of finer movements towards the end. This could mean that the maximum control velocity is not necessarily the determining factor, but mostly the minimum velocity, which allows for the finer adjustments during the trajectory.

A similar phenomenon was found when evaluating the performance of a joystick for pointing at a target in 2-dimensional space [17]. The research consisted of having subjects move the pointer to a certain target. Three parameters were varied in this experiment, namely the target distance, target size, and target direction. The target size is especially interesting since accuracy is an important factor when controlling the Sunram 7.
It was found that the movement time was affected by target distance and target size. Also, the movement towards the target tended to consist of two peaks. The primary peak had the largest acceleration and maximum speed. This peak was followed by a smaller secondary peak of movement which was for correction of the first movement and mainly to improve the accuracy. The study found that when the accuracy increased, the amount of times the secondary peak showed up in the movement pattern also increased.
This is in line with the findings of Woodworth [16]. The accuracy of the movement seems to rely on finer adjustments based on visual feedback. This could mean that, to find an optimum control speed, it is important to choose a maximum velocity which allows for a reasonable velocity during the initial stage of the movement while allowing the user the make finer, more accurate adjustments later in the movement.

The effect of joystick sensitivity has also been evaluated for Telemanipulative Microassembly [18], in the context of picking up microchips. The task for participants was to track a certain path towards the microchip with a joystick. The researchers varied the sensitivity of the joystick in a way that the velocity ranged between 0.056-0.195 cm/sec with a total of five different levels. The participants relied on visual feedback given on a monitor. It was found that when trying to follow a path, the deviation from the path increased with higher sensitivity levels. When only the final position of the end effector was measured with different sensitivity levels, the results showed that not the lowest, but the second to lowest velocity performed the best.
The results of the deviation from the ideal path showed a trend that was also found in the papers previously discussed. The result of the end position could be slightly surprising. A possible explanation that is given in the paper is that the boredom effect seems to affect the motivation of the participants. Therefore they seem to react too late and have to make corrections more often. It thus concludes that velocity level 2 was the best trade-off between speed and accuracy.

2.3 Controllers

Controllers, in the sense talked about in this report, are devices that act as a medium to convert a type of human input into something a computer, or microcontroller in this case, can understand. There are different types, such as controllers that detect velocity or position changes. An example of velocity detection is a normal computer mouse, which detects motion relative to a surface [19]. Position detection can be found in analog joysticks [20] commonly seen on gamepads, or on the controller board of the Sunram 7 (Figure 3.5).
Another difference can also be made between haptic and non-haptic controllers. Haptic controllers can simulate a sense of touch by applying vibrations to the user [21]. Haptic feedback can help during telesurgery to help reduce inadvertent tissue contact, making the surgery as minim-

ally invasive as possible [4]. In this report, the 3D SpaceMouse will be used. It is a non-haptic, position-based controller that can be held in one hand.

2.4 3D mouse control

The use of a SpaceMouse comes with three main obstacles. The SpaceMouse provides limited feedback to the user compared to a normal mouse [22]. The mouse is held in place by several springs, therefore the force of the feedback is proportional to the amount it is deviated from the centre. The SpaceMouse can be moved a small amount, namely 1 mm in the lateral direction and 5 degrees rotationally. This makes it suitable for extended use since the hand of the user does not have to move much but it does come with a caveat.

The second is the high sensitivity of the SpaceMouse. Users tend to be surprised with how sensitive the SpaceMouse is to even rather unnoticeable movements. They find it hard to rest their hand without actuating the SpaceMouse. An upside to this is that the sensitivity can rather easily be modified using software. Lastly, dimensional coupling tends to make the usage harder. The SpaceMouse does not 'force' the user to stay in a particular reference frame, which makes it easy to actuate along the wrong axes. Also, the input of the lateral movement and rotational movement tend to overlap a lot if not done carefully. Since the SpaceMouse is quite rotationally symmetric it can be easily rotated on the table without the user noticing. This has a large impact on the output and should be kept in mind.

The sensitivity and dimensional coupling can be mitigated by applying an input sensitivity curve. These curves apply a deadband to the input curve when the input curve is close to zero. Two options are to make the output linear to the input after the deadband or to apply a cubic function. In the cubic function, the output increases little when relatively close to zero and increases a lot when the input is rather high. The input sensitivity curve that will be used can be found in Figure 3.4.

Another solution to the dimensional coupling is dimensional weighting. If the translational x-axis is actuated a lot and the y-axis a little, a dimensional weighting function can prefer the x-axis to make handling easier for beginner users. A drawback of this could be that, if one does want to control multiple axes at once this behaviour can be more unpredictable.

2.5 Usability testing

The testing of usability can be done in multiple ways, both objective and subjective, therefore it is important to evaluate all the options and choose the appropriate method for the problem at hand.

First, the objective measurements were looked at. In a review paper [23] that evaluated 15 Human Robot Interaction (HRI) related studies, it was found that 47% of the studies measured task execution time. This was the factor that was evaluated the most. This could be because the execution time can show performance and improvement at the task and is relatively easy to measure in most cases. The person's idle time was measured in one of the studies [24]. Idle time could mean that the mental load for the task is too demanding for the person. Therefore the task is either too complicated for the person, or the control is not intuitive enough. The last factor that could be evaluated is the error from the centre of the target after being hit. It is expected that taking the average of multiple people will result in a clear picture of what the error distance is when the control speed is varied.

The SpaceMouse output and the joint positions of the robot will also be recorded. A possible application for this could be to verify if the same phenomenon as in studies [16; 17] of there being first a primary peak mainly for covering the most distance in a short amount of time. And after that a secondary peak which is meant to accurately bring the end-effector to the desired position.

After evaluating the possible questionnaires that could be used, three questionnaires were considered. These were the IBM Computer Usability Satisfaction questionnaire [25], the NASA Task Load Index (NASA-TLX) [26], and the System Usability Scale (SUS) [27].
The IBM questionnaire consists of 19 questions and asks participants how much they agree with a certain statement on a scale from 1 to 7. This questionnaire is more focused on the user's experience when using software on a computer, therefore it was deemed not ideal for the case of rating the experience when using the SpaceMouse.
As the name suggests, the NASA-TLX is more focused towards rating the demand and effort of the task rather than rating how effective the system is. It consists of 6 questions that ask to rate the experience of the user. This rating is done on a scale with 21 increments. Of the six questions, the questions "How physically demanding was the task?" and "How hard did you have to work to accomplish your level of performance?" were deemed to not be relevant for this task. Therefore it could be better to use a more suitable questionnaire.
The System Usability Scale consists of 10 statements (Table 8.1) and the user is asked how much they agree with the statements, on a scale from 1 to 5. The statements alternate between positive and negative statements, to avoid response bias. Response bias can occur when questions are phrased similarly and participants don't have to think much about whether they agree. Alternating the questions makes a participant have to think thoroughly about whether they agree with the statement or not. The SUS is a quick way to assess the usability of a broad range of systems. The questionnaire, among other things, assesses how easy to use, accessible, and overly complex the system is. These are factors that can be interesting for this report. Also, the SUS has previously been used in the assessment of control methods for the Sunram 7. This makes it possible to do a comparison between them. For these reasons, the System Usability Scale was chosen as the questionnaire to be carried out.

The SUS score [28] that results from the answers is a number between 1 and 100. It has to be noted that this scale is not linear. A score above 71 is considered 'good' and is in the 60^{th} percentile. This score can be calculated by taking all answers from odd-numbered questions and subtracting 1 from them, to get a number between 0-4. And by subtracting the answers from even questions from 5. Summing these numbers up, and then multiplying by 2.5 results in the SUS score (Equation (2.1)).

$$O = \text{answer} - 1 \qquad E = 5 - \text{answer}$$
$$\text{SUS score} = 2.5 \cdot \sum (O1), (E2), (O3), (E4), (O5), (E6), (O7), (E8), (O9), (E10) \tag{2.1}$$

2.6 Conclusion from literature

To summarize, the SUS was deemed the best option due to its simplicity, broad applicability, and prior use in evaluating the usability of control methods for the Sunram 7. Its ability to measure usability factors such as ease of use, accessibility, and complexity makes it a good fit for evaluating control with the SpaceMouse.

The studies of Woodworth [16] and Smyrnis [17] showed that error rates increase with speed, emphasizing the need for visual feedback and fine adjustments for movement accuracy. Pongrac [18] further supported these findings, showing that moderate joystick sensitivity levels provide the best balance between speed and accuracy.

Challenges in using the SpaceMouse include limited feedback, high sensitivity, and dimensional coupling. These can be mitigated by using a custom function for input sensitivity. This can reduce the impact of small, unintentional movements, while simultaneously reducing dimensional coupling.

3 Design

3.1 Equipment

The robot is being controlled with the SpaceMouse® Compact by 3Dconnexion[11]. This SpaceMouse can move in 6 degrees of freedom (6DoF), namely three direction dimensions and the knob is also able to rotate in three dimensions. Since the Sunram 7 has the ability to move in 5DoF, the SpaceMouse should be able to control the robot. The SpaceMouse outputs its data via a USB connection. This data cannot be fed directly to the interface of the Sunram 7 and therefore has to be modulated by a microcontroller board first.
For this, the Teensy 4.1 Development Board [12] will be used. It is the board of choice since it has the essential USB Host feature, which allows for communication with a USB input device completely separate from the USB serial port. This allows it to communicate with the SpaceMouse and with the Arduino Due on the controller board. Another advantage is that it can programmed using the Arduino IDE. After the appropriate code is uploaded, the Teensy 4.1 is able to take the raw output of the SpaceMouse and modify it in such a way that it can be read by the interfacing Arduino on the controller board of the Sunram 7. The Sunram 7 can be controlled in two ways, the first is to use the joysticks that are attached to the controller board. These analog joysticks can move in two directions and each direction controls one of the motors (Figure 3.5).
The second method is to send commands over USB serial to the Arduino Due on the controller board. The absolute positions of all the joints can be sent and the Arduino Due will handle the communication with the valves to move the joints to this position. This is the method that will be used.

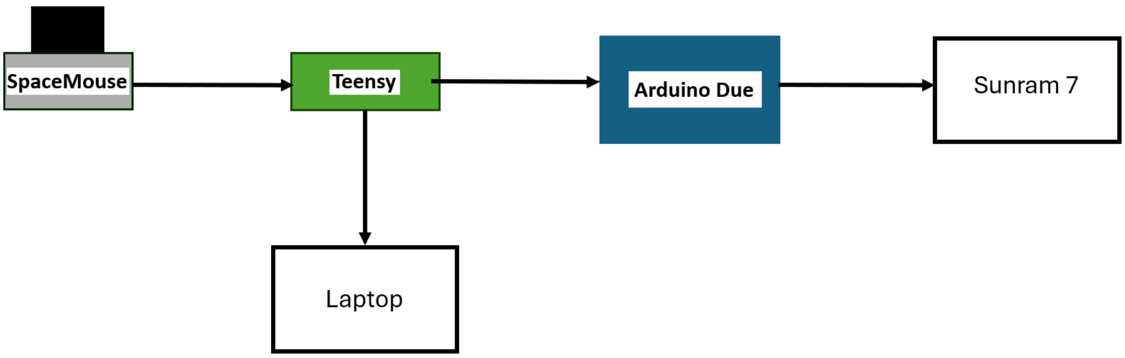


Figure 3.1: Schematic of how individual components are connected with arrows showing primary data flow.

3.2 Design goal

The goal of this setup is that it can be used for demonstration purposes. This is the reason why the mapping methods were kept relatively simple by trying to map the movement of the SpaceMouse as similar as possible to the movement of the Sunram 7.

The first standard that the method of control needs to meet is that it can be up and running within several seconds after the power is switched on.
Also, no additional software should be required.
Furthermore, the method has to be sufficiently accurate. The smallest lesions that are usually biopted are at least 5 mm [29]. Therefore, the error has to be smaller than 2 mm. The last requirement is that the control should be easy to use with minimal training.

3.3 Final design

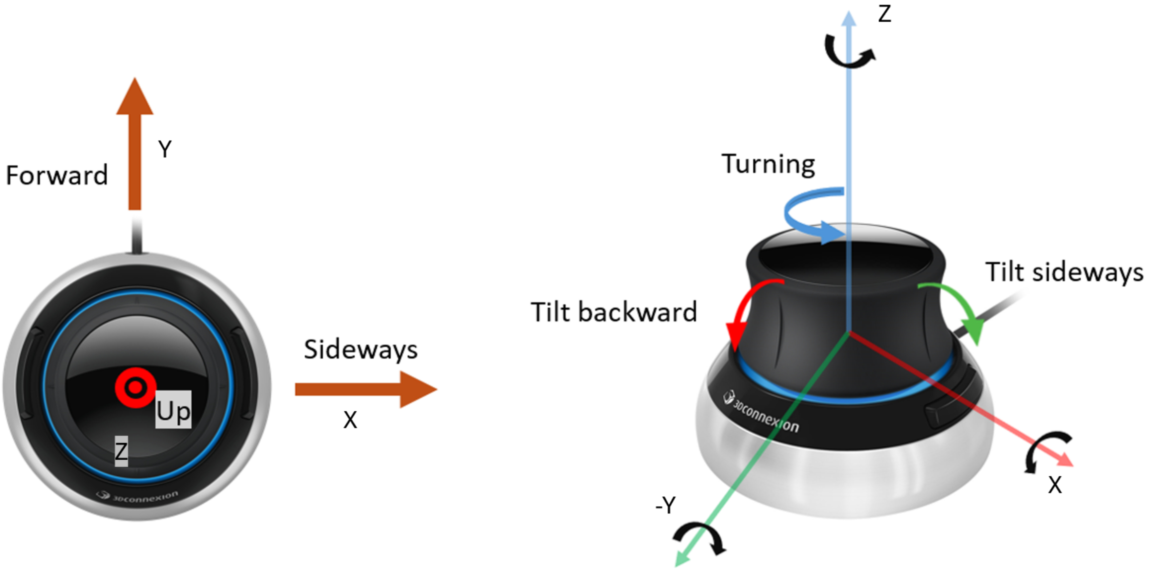


Figure 3.2: The reference frame of the SpaceMouse.

For the sideways direction (X) this meant that the robot will also move sideways by actuating motor (1). While doing this, the robot will also pivot left or right. The reason for this is that the rack that the gear is rotating on, is curved. This can be counteracted by manually pivoting the robot back.

Pivoting the Sunram 7 is mapped to rotating the SpaceMouse around its Z-axis (turning). These motions are relatively similar. Moving the SpaceMouse in this direction will actuate motor (2).

The forward direction (Y) was used to insert the needle, since this is the only forward motion that the robot can make. The needle is controlled by motor (5). To prevent accidental insertions of the needle, it can only be moved forward or backward when the left button is pressed. An additional option was added to lock the movement of the robot when the needle is inserted. This function was turned off for the accuracy tests but can be enabled by changing one variable in the code.

The Z axis of the mouse is used for up and down movement of the robot. Using it in this way makes the robot perform a very similar movement as the SpaceMouse. To move directly up or down, the robot actuates motors (3) and (4) simultaneously in the same direction.

Tilting the SpaceMouse backward or forward will result in the Sunram 7 also tilting backward or forward. This is done by actuating motors (3) and (4) in opposite directions. The actuation is also alternating between the two motors, to minimize sudden 'jumping' of the needle.

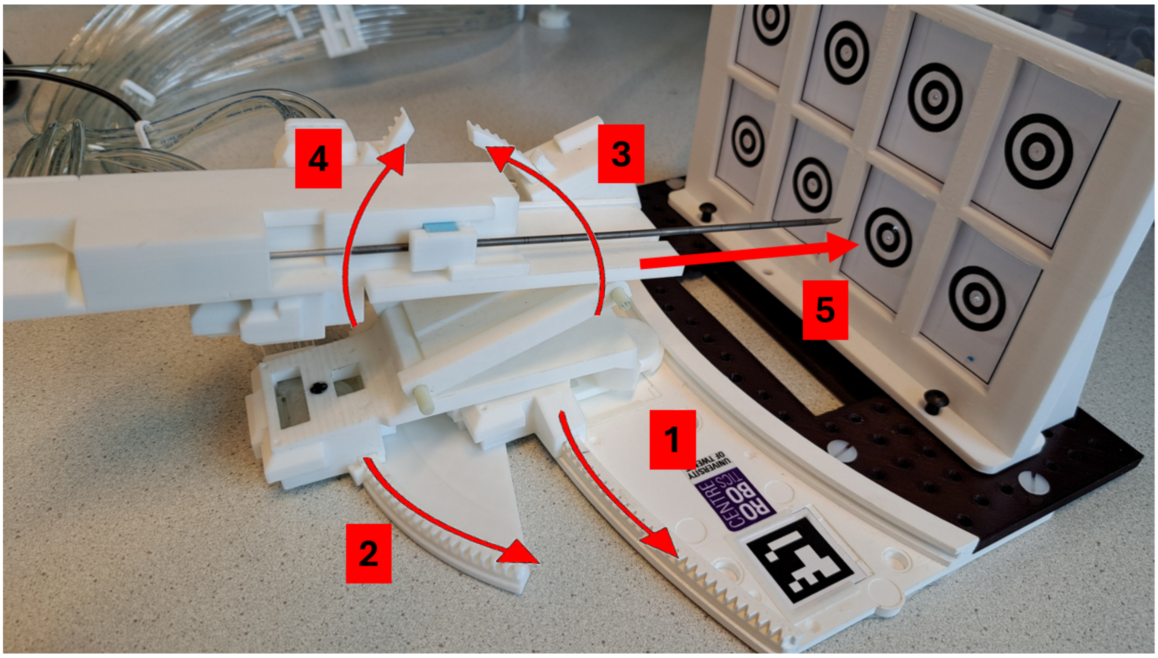


Figure 3.3: The five motors and the directions in which they are actuated over the rack.

Since the high sensitivity and dimensional coupling seemed to be hindering the control of the Sunram 7, a deadzone was added to reduce this problem (Figure 3.4). The SpaceMouse data, initially integers from -350 to 350, was scaled to floating-point numbers ranging from -5 to 5 before they were multiplied by the velocity scale factor. The reason for this scaling is that the velocity scale factor is in a range that is easier to deal with, namely between 3 and 15.
While preparing the tests and doing trial runs, it was noticed that the unwanted dimensional coupling resulted in inputs of about 1 in other directions than intended. To make the system easy to use for beginning users it was decided to choose a deadzone slightly higher than 1. It was set to 1.5 during the tests, but this could be lowered for more experienced users. The output after the deadzone was scaled down so that it started from zero at the boundary. The output was chosen to be linear for simplicity reasons, however this can rather quickly be turned into a cubic function.

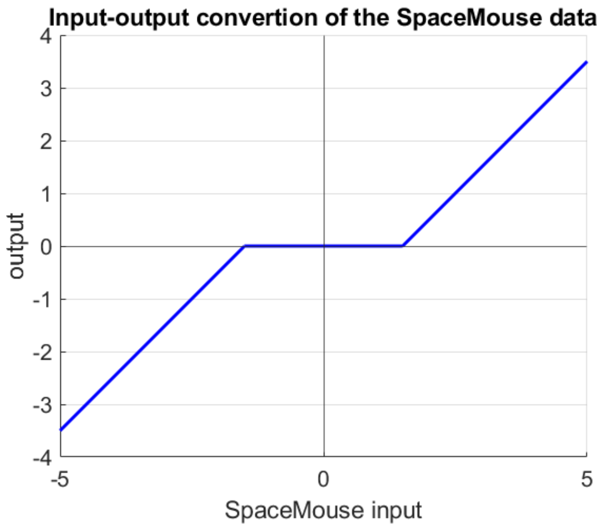


Figure 3.4: The joystick input versus motor velocity.

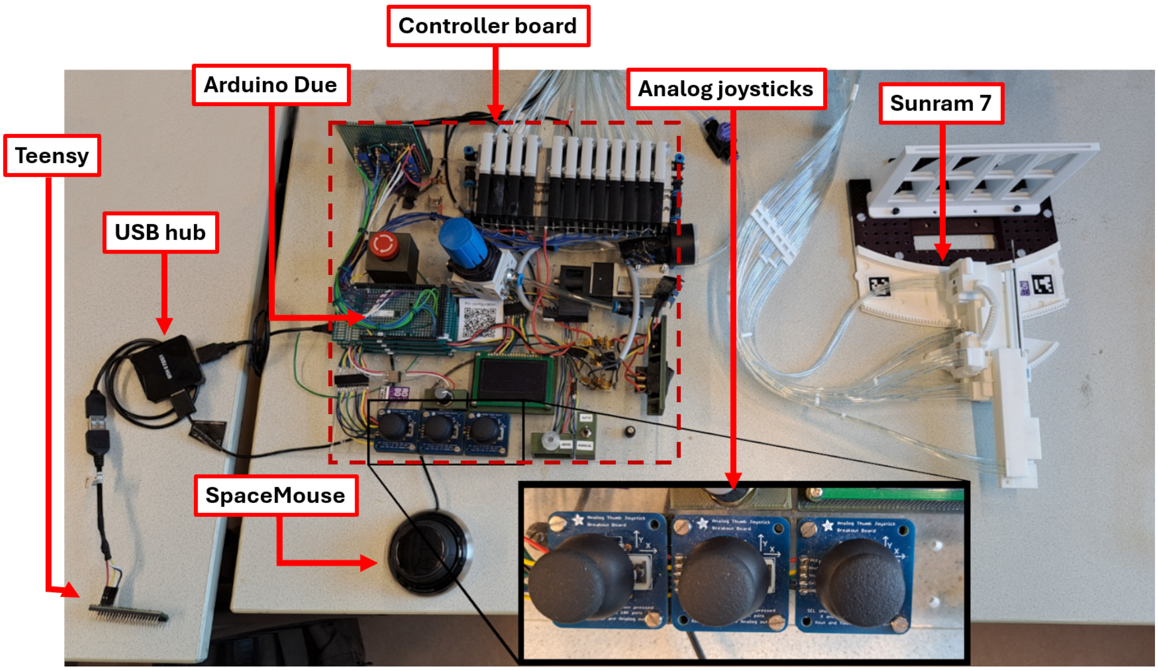


Figure 3.5: Picture of the complete setup. The controller board is outlined in dark red

4 Methodology

4.1 Programming

The code for sending the desired joint positions to Arduino Due on the controller board of the Sunram 7 is run from the Teensy 4.1 and is divided into several functions with their own task. The main loop in which these functions are called, is executed repetitively. It can run as a stand-alone system, only needing to be plugged into the mains for power.

The code consists of a global section where libraries and global variables are initialized, then the setup function followed by the loop, in which multiple functions are called sequentially.

In the global section, the parameters such as the controller deadzone, motor velocity and needle safety can be controlled. This is also where the boundaries of the motor movement are initialized. Furthermore, the libraries and drivers for USB devices are initialized here.
In the setup, only the serial connection between the devices is initialized.
In the loop, first it is checked whether there are any changes to the connected devices. After that, it is checked whether there is connection with the Arduino Due. If not, it tries to establish connection and get the device to respond back. Then, the SpaceMouse data is extracted from its object. When this is done, the motor work function (Algorithm 1)is called. Finally, the position data gets put into a function which handles USB communication with the Arduino Due for output and the laptop for data gathering.

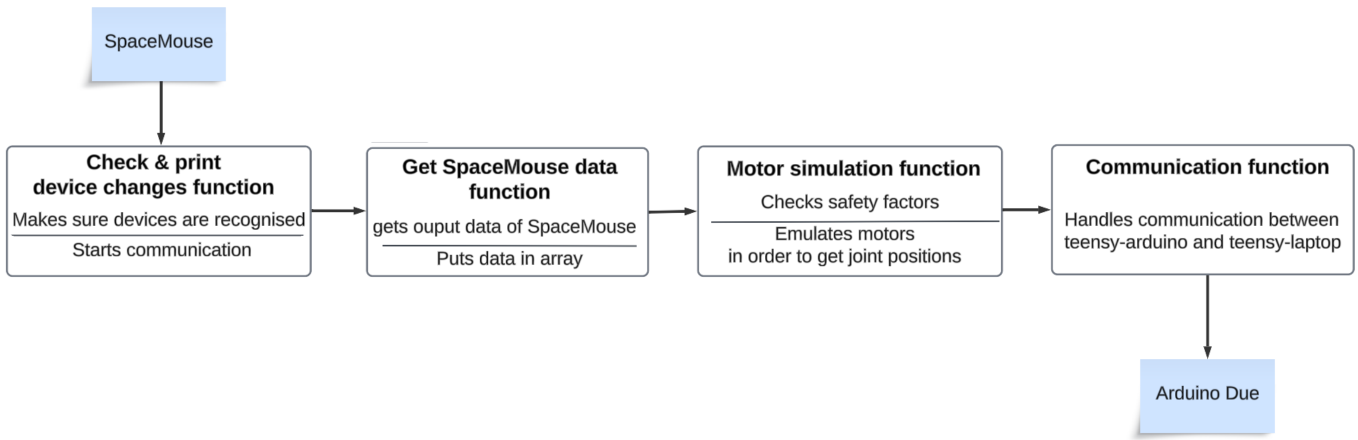


Figure 4.1: Flowchart of functions in the Arduino code.

```
Data: SpaceMouse output
Result: Motor position
initialization;
if spacemousedata >offset then
    motor_setVelocity = velocityScaleFactor · ( abs( spacemousedata) - offset )
    while physical constraint == false do
        if spacemousedata >0 then
            motor_setpoint = boundary_2
            motor_approachDirection = positive
            motor_work
        end
        if spacemousedata <0 then
            motor_setpoint = boundary_1
            motor_approachDirection = negative
            motor_work
        end
        position = motor_getPosition
        return position
    end
end
```

Algorithm 1: The motor control function.

The pseudo-code shows a simplified version of the motor simulation function. The code is only for one movement direction of the SpaceMouse, since different directions have to do different things. The code works as follows: the input is first checked on whether it is larger than the offset. If this is the case, the motor velocity gets scaled according to the input. A visual representation of this is shown in Figure 3.4. Then, the code checks whether the robot is not running into its physical limitations. After this, the actual work functions are called. They are split into two parts to allow for movement both ways. If the SpaceMouse moves right and thus the input is positive, then the setpoint will be set to the right boundary. The appropriate direction is chosen and the work function is called. After this, the current position of the motor is put into an array which gets printed to the Arduino Due to execute.

4.2 Physical constraints

The Sunram 7 is not able to make certain combinations of joint positions. To make sure components do not break, or that the motors experience too much resistance, these areas are blocked by the software running on the Teensy 4.1. The way this was done was by implementing if-statements before the motor work functions. The statements check if the robot does not cross a boundary that it is not allowed to. An example of this is that the Sunram 7 can not pivot right when the front is down. The software checks if the robot is pivoted enough to the left or that the front is moved up, if one of these conditions is true then the robot is allowed to pivot right. The largest domain which is prohibited for the Sunram 7 is that it can not pivot right on the left side of its working space. Also, the maximum angle for tilting frontward or backward is limited, otherwise it can press the backside of the needle on the ground plane.

Figure 4.2: Picture of target that was hit at (error: 0.6 mm, angle 90°).

4.3 Testing & gathering data

4.3.1 Set-up

Since one of the requirements of the design is that it should be easy and quick to set up, different set-up times were measured.
The first time that was measured was how much time is needed to set up the whole system when nothing is connected yet. This was done by placing all parts separately on the table, connecting them and waiting for the Sunram 7 to be ready for use (Figure 8.1). The second measurement was to measure when everything is plugged in correctly, but the power is shut off. For this, the normal setup was used and the amount of time between the controller board being switched on and the Sunram 7 being ready to use was measured.

4.3.2 Accuracy

To test if the Sunram 7 can be controlled sufficiently accurate with the SpaceMouse, it was attempted to hit a 2D target (Figure 4.2) with the needle tip as accurately as possible. The test setup is shown in Figure 3.3. The target was made of paper because the titanium needle can be damaged easily. For the test, the Sunram 7 was placed in its home position and moved to the top right target. There was no time constraint, or limit on the amount of movements that could be made. For this test, as well as the ease of use tests, different velocity settings were used. The levels were scaled from 3-15, with a step size of 2 between them. The actual velocities are shown in Table 4.1.

Velocity setting	3	5	7	9	11	13	15
Actual speed (mm/s)	6.3	10.6	14.8	19.0	23.3	27.5	31.7

Table 4.1: Table showing the actual speed at which the motor is driven over the rack.

4.3.3 Ease of use

To test the effectiveness of the control method at different speeds, multiple test subjects have tried to hit a target. This target was switched with a new one after it was hit once and the robot was placed back to the home position. The task was to hit a paper target that was placed in front of the robot. The robot will be set to different speeds to test its usability at these speeds. To prevent bias towards later tries after subjects have gotten used to controlling the robot, half of the participants started at the lowest setting with increasing velocity levels while the other half started at the highest setting and descended through the levels.

The usability testing will partly depend on data that can be gathered from the interactions. This is because it can be gathered rather easily by measuring hits manually and collecting SpaceMouse output and robot input data on a computer. The data that was collected manually is the error of the hit and spread of the hits. This was done with precise paper targets (Figure 4.2), that are similar to MATLAB's polar plots so they can be converted to digital rather easily. Furthermore, the time it takes to hit the target, the path data, and the SpaceMouse outputs were collected. This data is gathered using a serial connection to a laptop to convert the output of the Teensy to a text file.

To get a more complete picture of how effective the control method is, the System Usability Scale questionnaire was carried out. Participants were asked to fill out this questionnaire on paper. This resulted in a number that can be used to evaluate the system and compare it to other systems.

5 Results

5.1 Set-up

	startup times (s)	setup times (s)
run 1	2.09	69
run 2	2.12	70
run 3	2.18	62
run 4	1.99	62
run 5	2.06	62
mean	**2.09**	**65**
std	0.07	4.1

Table 5.1: Table showing the start-up and set-up times over 5 runs. The mean and standard deviation are also given.

Table 5.1 shows all measured times of starting and setting up the system. There is little variation in start-up times, and the average measured time is 2.09 seconds. The largest factor seemed to be in the Teensy 4.1 trying to establish connection with the SpaceMouse.

The setup times start with 69 and 70 seconds and then reduce to 62 seconds for the rest of the tries. This brings the average setup times to 65 seconds.

5.2 Accuracy

The accuracy was evaluated by looking at the average deviation from the middle of the target, the patterns of how the errors spread around the target, and the movement patterns during navigating the robot.

The average error was taken after four runs on each velocity setting (Figure 5.1). The lowest velocity setting also resulted in the lowest deviation from the middle (0.43 mm). After this, setting 9 performed the best with a mean error of 0.53 mm. Velocity setting 11 showed the largest deviation from the desired target. The average of all runs was 0.75 mm with a standard deviation of 0.24 mm.

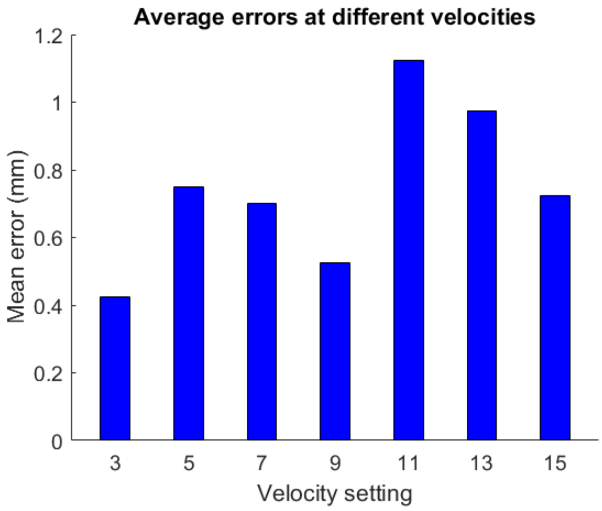


Figure 5.1: Graph showing the average error at each velocity setting. The mean of all runs is 0.75 mm.

The patterns of where the errors are most present can be seen from the polar scatter plot in Figure 5.2. The horizontal error does not exceed 1 mm, while the vertical error does so seven times.

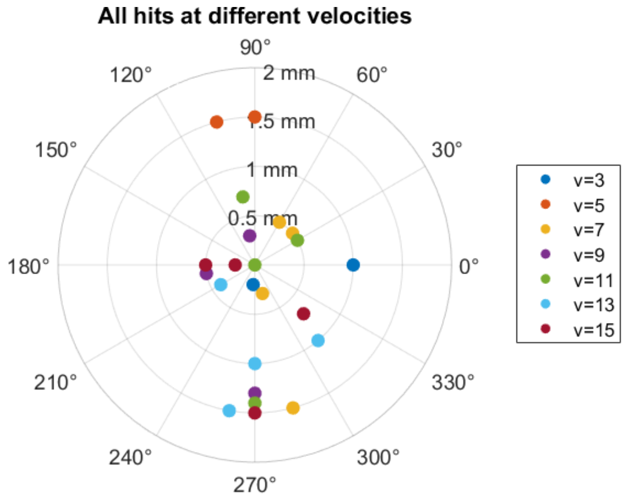


Figure 5.2: Polar plot showing all hits at different velocities. These hits were measured after the accuracy testing.

5.3 Ease of use

5.3.1 Measurement data

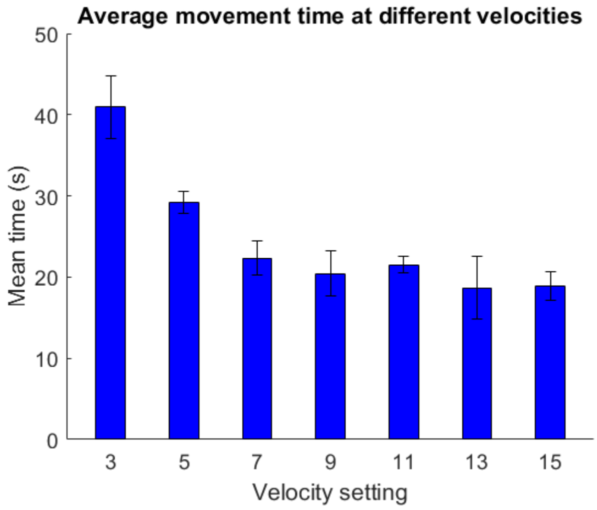


Figure 5.3: Bar graph of the average movement time per velocity setting. The movement time tends to decrease with higher velocity settings.

The average time it took to hit the target at different velocity settings is shown in Figure 5.3. The times drop quite hard between the lowest three settings, and after this, they tend to stay at around 20 s. The standard deviation of the times is plotted with error bars. The average movement time overall was 24.6 s.

The movement patterns were also evaluated (Figure 5.4). The graph shows the sum of all absolute values of the joint positions. This can also be seen as the total amount of steps away from the home position. Here, only the highest and lowest velocity settings are plotted and they show quite a different pattern. At the low velocity setting, the user is seen to move straight to the target and insert the needle almost directly. At the highest velocity setting, the stagnation in the curve shows an adjustment period. In some runs, there is also an overshoot visible, such as in one of the runs at v=15.

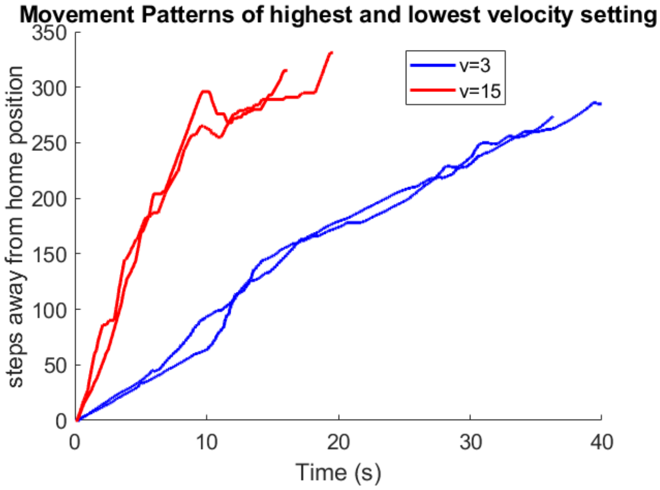


Figure 5.4: Graph comparing the movement times and patterns by plotting the sum of the joint positions over time. Setting 15 tends to be a lot faster, but also makes extra steps.

5.3.2 Participants

The tests were carried out with eight participants. However, one participant seemed to be such an outlier in terms of accuracy, movement time and technique that this data was taken out of the dataset. The average error of this user was 3.15 mm, while all other users were between 0.57 and 1.64 mm. A graph showing this can be found in the appendix (Figure 8.2).

The movement times of participants (Figure 5.5) show the largest standard deviations towards the highest and lowest velocity settings. The overall average movement time was 32.4 seconds.

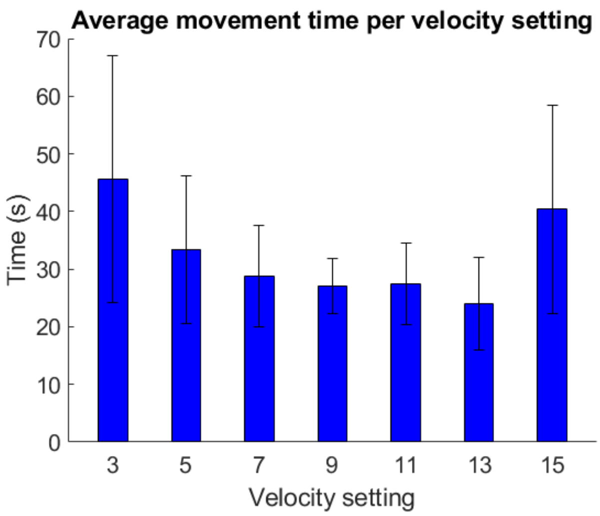


Figure 5.5: Bar graph of the average movement time per velocity setting of participants. The standard deviation is given with error bars.

Figure 5.6 shows the average deviation from the middle per velocity setting. Velocity setting 9 shows to have the lowest average error with an average of 0.84 mm. The highest velocity setting also results in the largest error of 1.86 mm.
The mean of all the runs was 1.30 mm with a standard deviation of 0.35 mm.

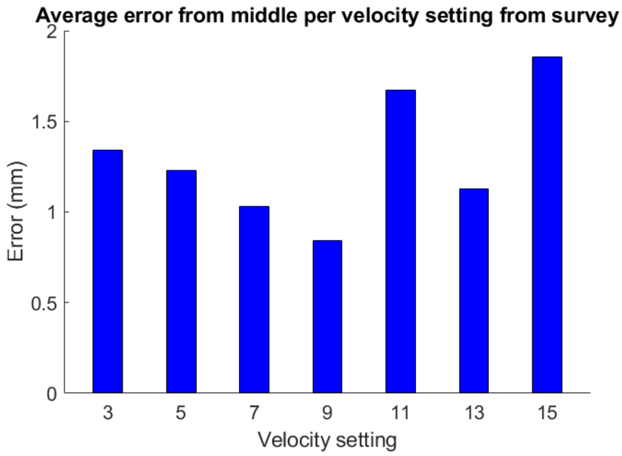


Figure 5.6: Average error per velocity setting of the seven participants. The average overall error is 1.30 mm.

5.3.3 System Usability Scale

The system was also evaluated by having the participants fill in the SUS questionnaire, this resulted in an average score of 78.75 with a standard deviation of 6.8. A score of 78.75 can be considered 'good' to 'excellent' [30]. All results can be found in Table 8.2. This table shows the unprocessed score from each user user per question. In Table 5.2, the average rating per question scaled from one to five can be found. It can be noted that statement four "I think that I would need the support of a technical person to be able to use this system" was filled in with the highest score by every participant.

Question one, "I think that I would like to use this system frequently" was filled in with the lowest score overall.

Question	1	2	3	4	5	6	7	8	9	10
Average (1-5)	3.5	4.5	4.13	5	4.13	3.75	4.5	4	3.63	4.38

Table 5.2: Table showing the average score per question on a scale from 1-5, negative questions inverted for readability.

6 Discussion

6.1 Set-up

The results showed a mean start-up time of 2.09 s with a standard deviation of 0.07 s. The main reason for the standard deviation is assumed to be the human factor in measuring the start-up time with a stopwatch. For this measurement, only one switch had to be flipped which triggers the Teensy 4.1 to establish communication with the SpaceMouse. This took a few seconds and it was assumed to be the largest contributor to the time.

The set-up times were achieved by connecting all individual components. This went more smoothly after multiple tries, which is not surprising. The first two runs are probably more representative of what can normally be expected in terms of the time it takes to prepare the system. The components were already placed on the table, which made the process relatively efficient.

6.2 Accuracy

The graph showing the average error per velocity setting (Figure 5.1) did not give a decisive answer to the hypothesis. It can be noted that the lowest three velocities are performing better in terms of accuracy compared to the highest three velocities. The relative differences are small and the fact that they are within tenths of millimetres is not what was expected beforehand. The output curve having an offset and starting at zero after the offset probably helps with finer adjustments towards the end. From the scatter plot (Figure 5.2) a clear pattern does emerge. The errors are spread quite a bit more in the vertical direction than in the horizontal direction. This is true for the spread of the participants, as well as of a more experienced user. However, there is the possibility that this effect is exaggerated due to the needle being extended relatively far when the adjustments are made. It could nonetheless be improved by mapping the output of the SpaceMouse differently.

For the participants, the error seems to get smaller when approaching the middle velocity setting. Setting 11 seemed to be a large outlier in this. One half of the group started at the lowest setting and moved up, while the other half moved down from the highest setting. It was expected that the error decreased when participants had practised a few runs, and Figure 5.6 was an average of two graphs where the error declines in different directions. In Figure 8.3, it can be seen that this is not the case. To be entirely sure that this is not a factor, the velocity setting order could have been randomized.

6.3 Movement Times

When doing four runs at each velocity level, it was observed that the time to hit the target did not decrease much after velocity setting 7. The difference in the pattern of slow and fast movements was shown in Figure 5.4. When the velocity is sufficiently high, the adjustment period becomes a large contributor to the movement times instead of the time it takes to move to the target.

Due to it being subjective data, it does not show up in the results, but participants, especially the users who descended through the velocity settings said they found it boring while performing the test with the lowest velocity setting. This may not be of much importance during medical procedures but it does show a similar boredom effect as observed in study [18]. Originally it was the idea to use velocity levels 1-9 with increments of 2, but it became apparent that this was too slow. To keep the data from the first test valid, higher levels were added instead of rescaling the input, a drawback of this was that the settings were not scaled from 1-10 anymore, as was intended.

6.4 Patterns

It was hard to see notable differences between different groups such as the medical and non-medical students, or the participants that ascended or descended through the velocity levels. One notable difference was between male and female participants.

The mean of the sample female participants was 1.06 mm while the mean of male participants was 1.48 mm. To test if this is a significant difference, a confidence interval test for the difference in two population means is necessary. The calculation is shown in equation (8.1). From this, it can be seen that the null hypothesis of the means being the same can be rejected with 95% confidence and that there therefore is a significant difference between the two groups.

However, the deviation seemed to depend a lot on personal interpretation of the exercise. Some participants took a few extra seconds to adjust the needle to be positioned slightly better. Because the differences are small, this can make quite a large difference in overall results. A solution for this is to take a larger pool of participants to minimize this influence.

6.5 System Usability Scale

Question 1 "I think that I would like to use this system frequently" of the System Usability Scale was deemed the least relevant question of the survey. The other nine questions were more related to usability and complexity. It was not always interpreted the same. Overall, the answers were relatively consistent and a score of 78.8 can be seen as a good result. It is a higher result than other control methods on the Sunram 7 which were 70.4 and 67.5 for optical and haptic feedback control respectively.

7 Conclusions and Recommendations

7.1 Conclusions

The condition that one should be able to use the system within several seconds after turning the power on is met. It takes around 2 seconds for the system to start up, while it can be assembled in 65 seconds on average.

The average error when using the system was used by an experienced user was 0.75 mm. Participants in the testing managed an average of 1.3 mm. This fulfils the accuracy requirement that the error has to be smaller than 2 mm.

Lastly, the ease of use was evaluated. Question 3 of the SUS, "I thought the system was easy to use" was filled in with an average score of 4.13 out of 5. The overall SUS score of 78.8 indicates good to excellent usability of the system.

The hypothesis was "there is an optimum speed-accuracy trade-off to control the Sunram 7". If one would look only at the survey data, it could be concluded that there is an optimum, namely velocity setting 9. From the repeated runs by a more experienced user, setting 3 was the absolute best in terms of accuracy, while velocity setting 9 was the second best with the error going from 0.425 mm to 0.525 mm between the two settings. The movement time went from 41.0 s at setting 3 to 20.4 s at setting 9. This means that if one wants the absolute most accuracy, velocity setting 3 seemed better. But if a little bit less accuracy can be tolerated, setting 9 yields a good trade-off.

7.2 Recommendations

For use in demonstrations in the future, some improvements could be made. The first would be to look into the vertical sensitivity of the Sunram 7. There is a possibility that the effect is slightly exaggerated, but it is nonetheless present. A possible solution could be to let the Sunram 7 tilt around its tip, instead of around the motor axes. This could be a more intuitive reference frame. Another solution could be to only use motor 3 for tilting, this resulted in less of a jump than motor 4. Motor 3 is located closer to the front of the Sunram 7 which means that the step size is enlarged less.

Secondly, a small interface could help with usability. The Sunram 7 has a few position combinations that are not allowed. In this case, it will move no further. In some situations, it is hard to see that this is the case and it appears to be malfunctioning. An LED, or display could be implemented to let the user know that it hit such a boundary. With a display, there is also the option to show which boundary is hit.

Lastly, the solution can be packaged more nicely. A possibility is to add a shield on top of the Arduino Due, that handles the communication between the SpaceMouse and the Arduino Due. This would help people using the system on their own and make it more robust.

8 Appendix

8.1 Test methods

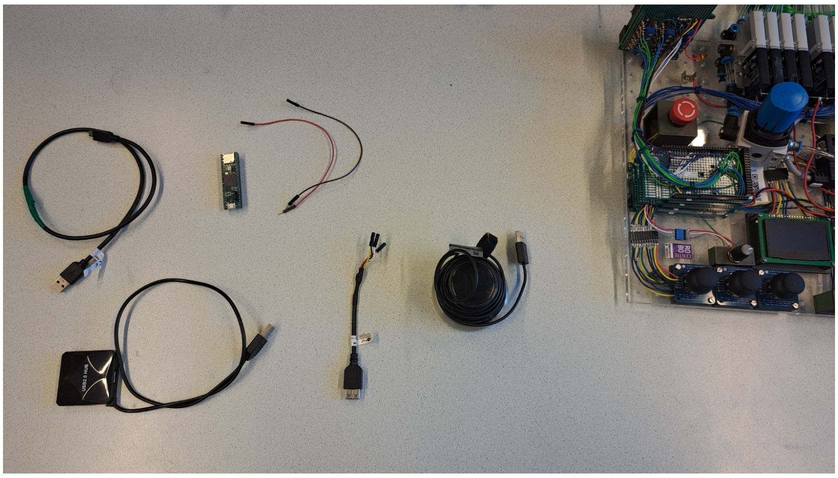


Figure 8.1: The separate components before assembling them.

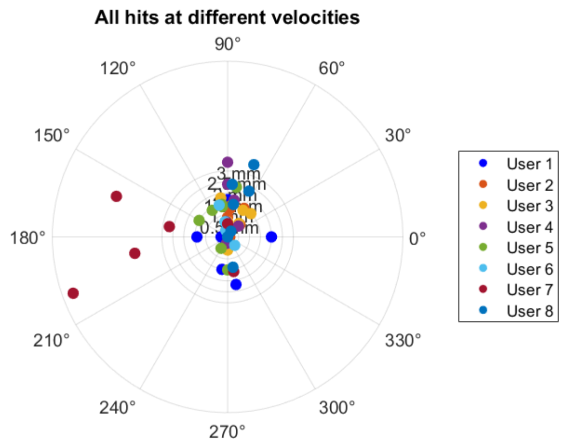


Figure 8.2: Figure showing hits of all users.

Q1	I think that I would like to use this system frequently.
Q2	I found the system unnecessarily complex.
Q3	I thought the system was easy to use.
Q4	I think that I would need the support of a technical person to be able to use this system.
Q5	I found the various functions in this system were well integrated.
Q6	I thought there was too much inconsistency in this system.
Q7	I would imagine that most people would learn to use this system very quickly.
Q8	I found the system very cumbersome to use.
Q9	I felt very confident using the system.
Q10	I needed to learn a lot of things before I could get going with this system.

Table 8.1: The questions of the System Usability Scale.

User	**Q 1**	**Q 2**	**Q 3**	**Q 4**	**Q 5**	**Q 6**	**Q 7**	**Q 8**	**Q 9**	**Q 10**
1	4	2	4	1	5	2	4	2	4	1
2	1	1	4	1	4	3	5	2	4	1
3	4	2	4	1	5	2	5	2	4	1
4	4	1	5	1	4	2	5	1	4	1
5	4	1	5	1	4	1	4	1	3	4
6	4	1	4	1	4	4	5	2	3	2
7	4	3	3	1	4	2	4	3	4	2
8	3	1	4	1	3	2	4	3	3	1

Table 8.2: Table with the complete results of the SUS questionnaire.

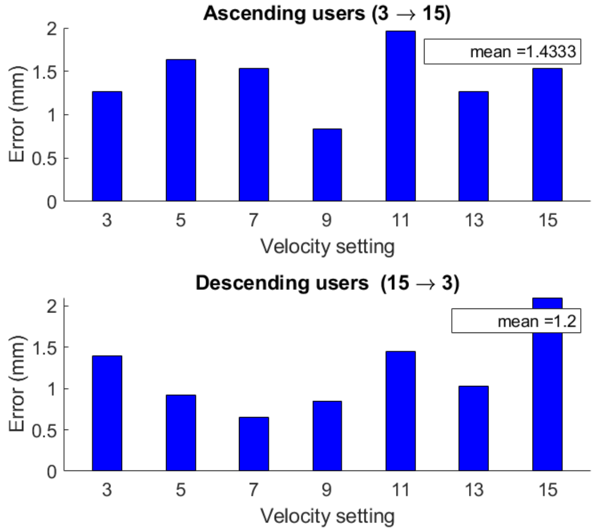


Figure 8.3: The error at different velocity settings between participants who started at the lowest or highest setting.

$$\bar{X} = 1.48 \quad S_1 = 0.21 \quad n_1 = 4$$
$$\bar{Y} = 1.06 \quad S_2 = 0.11 \quad n_2 = 3$$
$$t_5 = 2.5766 \quad for \quad \alpha = 0.05$$
$$H_0 : \Delta_0 = \mu_1 - \mu_2 = 0$$

$$T = \frac{(\bar{X} - \bar{Y}) - \Delta_0}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2}}} = \frac{1.48 - 1.06}{\sqrt{\frac{0.21^2}{4} + \frac{0.11^2}{3}}} = 3.42 \tag{8.1}$$

$$3.42 > 2.5755$$

Bibliography

[1] Kanker.nl, “Overlevingscijfers van borstkanker.” Kanker.nl, march 2024. [Online]. Available: https://www.kanker.nl/kankersoorten/borstkanker/algemeen/overlevingscijfers-van-borstkanker

[2] NIBIB, “Mammography,” 14-03-2024. [Online]. Available: https://www.nibib.nih.gov/science-education/science-topics/mammography#:~:text=breast%20cancer%20screening%3F-,What%20is%20mammography%3F,cancer%20and%20other%20breast%20diseases.

[3] N. . NIV, “Borstkanker,” 2024. [Online]. Available: https://richtlijnendatabase.nl/richtlijn/borstkanker/primaire_diagnostiek/bi_rads.html

[4] A. Simorov, R. S. Otte, C. M. Kopietz, and D. Oleynikov, “Review of surgical robotics user interface: What is the best way to control robotic surgery?” 2012.

[5] H. Su, M. Zervas, G. A. Cole, C. Furlong, and G. S. Fischer, “Real-time mri-guided needle placement robot with integrated fiber optic force sensing,” 2011.

[6] S. B. Park, J. G. Kim, K. W. Lim, C. H. Yoon, D. J. Kim, H. S. Kang, and Y. H. Jo, “A magnetic resonance image-guided breast needle intervention robot system: overview and design considerations,” *International Journal of Computer Assisted Radiology and Surgery*, vol. 12, 2017.

[7] C. Gao, A. Farvardin, R. B. Grupp, M. Bakhtiarinejad, L. Ma, M. Thies, M. Unberath, R. H. Taylor, and M. Armand, “Fiducial-free 2d/3d registration for robot-assisted femoroplasty,” *IEEE Transactions on Medical Robotics and Bionics*, vol. 2, no. 3, pp. 437–446, 2020.

[8] A. C. Tsitos and M. Dagioglou, “Handling vision noise through robot motion control in a real-time teleoperation system,” in *2022 30th Mediterranean Conference on Control and Automation (MED)*, 2022, pp. 624–629.

[9] H. Healthcare, “Robotic excellence with the da vinci® surgical system.” 2 Cavendish Square: HCA Healthcare UK, 2023. [Online]. Available: https://www.hcahealthcare.co.uk/about-hca-uk/robotics-and-technology-at-hca/the-da-vinci-surgical-system

[10] I. Surgical, “Robotic-assisted surgery with da vinci systems,” 2024. [Online]. Available: https://www.intuitive.com/en-us/patients/da-vinci-robotic-surgery

[11] D. Connexion, “Spacemouse® compact.” 3Dconnexion, 21-03-2024. [Online]. Available: https://3dconnexion.com/uk/product/spacemouse-compact/

[12] PJRC, “Teensy® 4.1 development board.” PJRC, 21-03-2024. [Online]. Available: https://www.pjrc.com/store/teensy41.html

[13] “European medical device regulation (mdr).” West Lafayette, IN 47906: MED Institute, 2020. [Online]. Available: https://medinstitute.com/blog/european-medical-device-regulation-mdr/

[14] V. Papalouka, F. Kilburn-Toppin, M. Gaskarth, and F. Gilbert, “Mri-guided breast biopsy: a review of technique, indications, and radiological–pathological correlations,” *Clinical Radiology*, vol. 73, 2018.

[15] A. Patra and S. N. Keshava, “Biopsy with side-cutting coaxial needle-knowing the cutting length and throw length,” 2021.

[16] R. S. Woodworth, “Accuracy of voluntary movement.” *The Psychological Review: Monograph Supplements*, vol. 3, 1899. [Online]. Available: https://psycnet.apa.org/record/2011-13573-001

[17] N. Smyrnis, I. Evdokimidis, T. S. Constantinidis, and G. Kastrinakis, “Speed-accuracy tradeoff in the performance of pointing movements in different directions in two-dimensional space,” *Experimental Brain Research*, vol. 134, 2000.

[18] H. Pongrac, A. Reiter, B. Faerber, and M. F. Zaeh, “Evaluation of joystick sensitivity in telemanipulative microassembly,” 2007. [Online]. Available: https://ieeexplore.ieee.org/document/4371590

[19] W. contributors, “Computer mouse,” 2024, [Online; accessed 29-June-2024]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Computer_mouse&oldid=1229825984

[20] ElectronicWings, “Analog joystick guide with pinout & arduino interfacing,” 2024, [Online; accessed 29-June-2024]. [Online]. Available: https://www.electronicwings.com/sensors-modules/analog-joystick

[21] S. Kapoor, P. Arora, V. Kapoor, M. Jayachandran, and M. Tiwari, “Haptics - touchfeedback technology widening the horizon of medicine,” 2014.

[22] V. Dhat, N. Walker, and M. Cakmak, “Using 3d mice to control robot manipulators,” 2024. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3610977.3637486

[23] A. Apraiz, G. Lasa, and M. Mazmela, “Evaluation of user experience in human–robot interaction: A systematic literature review,” 2023. [Online]. Available: https://link.springer.com/article/10.1007/s12369-022-00957-z

[24] P. A. Lasota and J. A. Shah, “Analyzing the effects of human-aware motion planning on close-proximity human-robot collaboration,” *Human Factors*, vol. 57, 2015.

[25] J. R. Lewis, “Ibm computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use,” *International Journal of Human-Computer Interaction*, vol. 7, 1995.

[26] S. G. Hart and L. E. Staveland, “Development of nasa-tlx (task load index): Results of empirical and theoretical research,” *Advances in Psychology*, vol. 52, 1988.

[27] J. Brooke, “Sus -a quick and dirty usability scale usability and context,” *Usability evaluation in industry*, vol. 189, 1996.

[28] “System usability scale,” 2024. [Online]. Available: https://handbook.gitlab.com/handbook/product/ux/performance-indicators/system-usability-scale/#calculating-sus-scores

[29] L. Liberman, G. Mason, E. A. Morris, and D. D. Dershaw, “Does size matter? positive predictive value of mri-detected breast lesions as a function of lesion size,” *American Journal of Roentgenology*, vol. 186, 2006.

[30] A. Bangor, P. Kortum, J. M. J. of usability studies, and undefined 2009, “Determining what individual sus scores mean: Adding an adjective rating scale,” *uxpajournal.orgA Bangor, P Kortum, J MillerJournal of usability studies, 2009•uxpajournal.org*, vol. 4, 2009.