Mälardalen University
School of Innovation Design and Engineering
Västerås, Sweden

GUI USER MANUAL

13th December 2023

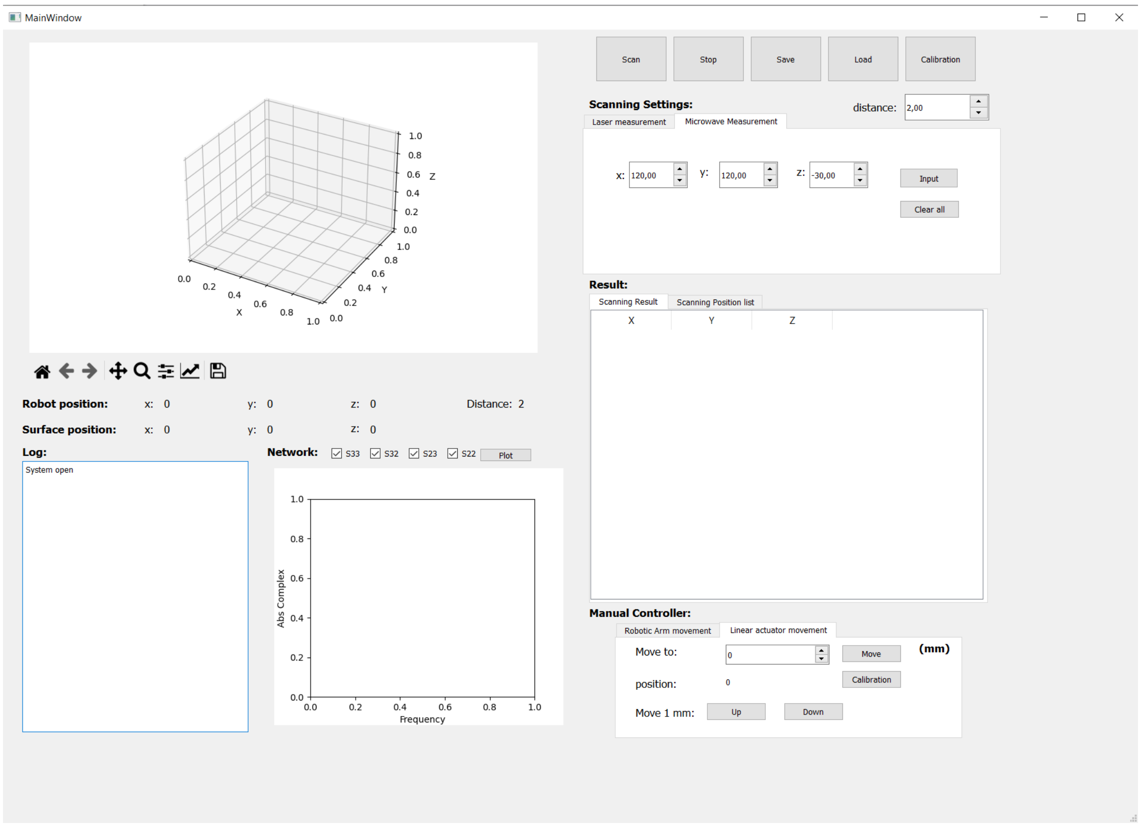
Contents

1. GUI overview

This GUI is not dynamic adjustable. Therefore, you need a big screen for display all functions.

Run MainProgram.py for run the GUI.

It is able to perform cylindrical scan. When you use halve sphere scan, your robot will have a collision with linear actuator. Same problem in microwave scanning when you want to scan the point on the bottom.



Scanning Settings: distance: 2,00

x: 120,00 y: 120,00 z: -30,00

Robot position: x: 0 y: 0 z: 0 Distance: 2

Surface position: x: 0 y: 0 z: 0

Log: System open

Network: ☑ S33 ☑ S32 ☑ S23 ☑ S22

Move to: 0

position: 0

Move 1 mm:

2. Scan Button

Perform scanning based on which tab is activating in Scanning Settings when user press the scan button. When scanning, other buttons about robot arm will be disable expect stop button. The result will be save in the Scanning result table in Result and plot in the Scanning viewer.

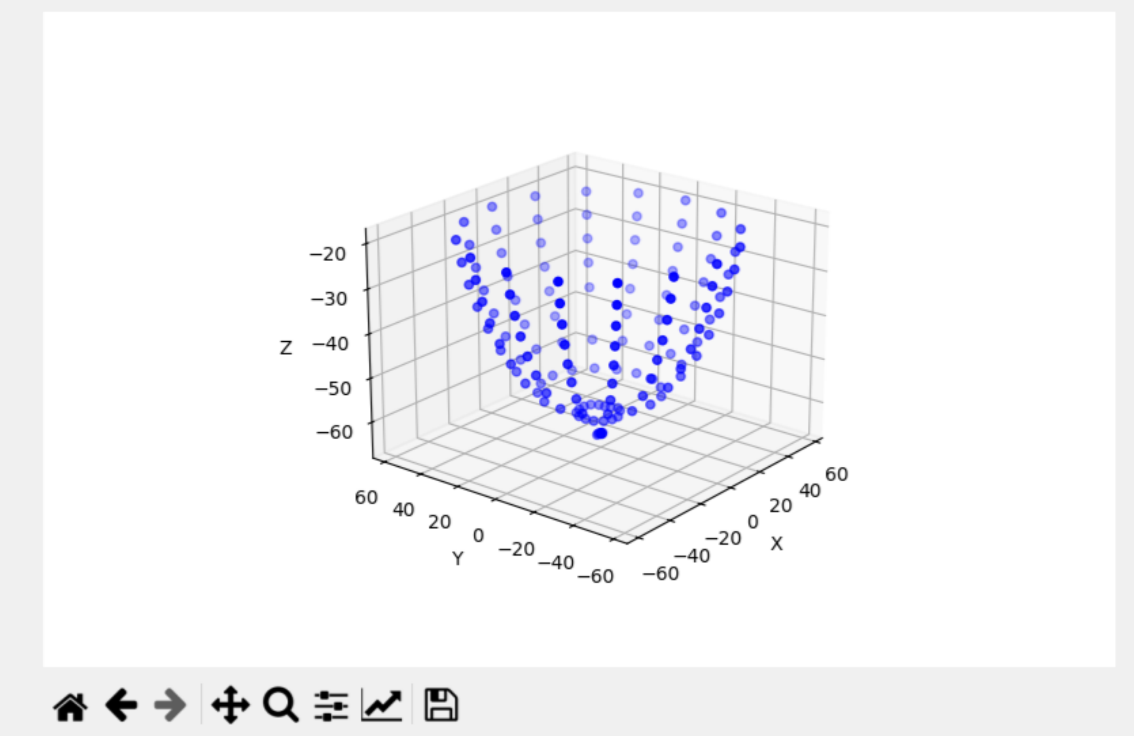
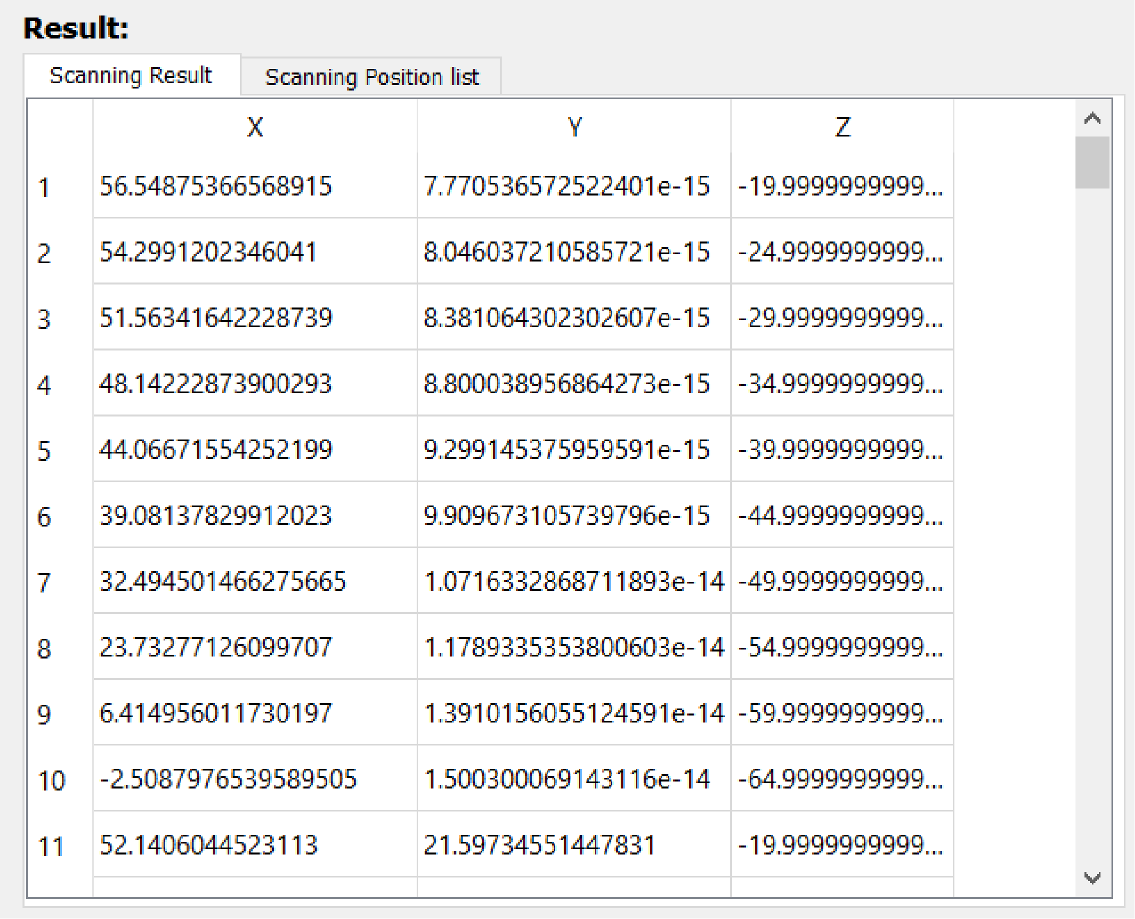


Figure 1: Scanning viewer



Result:

	X	Y	Z
1	56.54875366568915	7.770536572522401e-15	-19.9999999999...
2	54.2991202346041	8.046037210585721e-15	-24.9999999999...
3	51.56341642228739	8.381064302302607e-15	-29.9999999999...
4	48.14222873900293	8.800038956864273e-15	-34.9999999999...
5	44.06671554252199	9.299145375959591e-15	-39.9999999999...
6	39.08137829912023	9.909673105739796e-15	-44.9999999999...
7	32.494501466275665	1.0716332868711893e-14	-49.9999999999...
8	23.73277126099707	1.1789335353800603e-14	-54.9999999999...
9	6.414956011730197	1.3910156055124591e-14	-59.9999999999...
10	-2.5087976539589505	1.500300069143116e-14	-64.9999999999...
11	52.1406044523113	21.59734551447831	-19.9999999999...

Figure 2: Scanning result table in Result

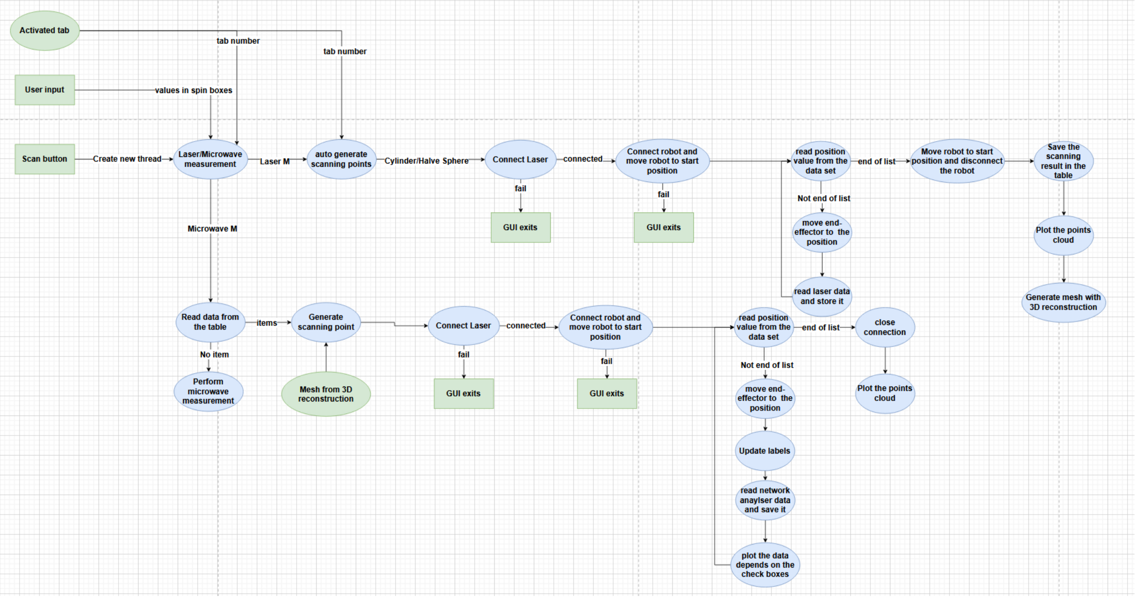

Activated tab
tab number
tab number
User input
values in spin boxes
Scan button
Create new thread
Laser/Microwave measurement
Laser M
auto generate scanning points
Cylinder/Halve Sphere
Connect Laser
connected
Connect robot and move robot to start position
read position value from the data set
end of list
Move robot to start position and disconnect the robot
Save the scanning result in the table
fail
GUI exits
fail
GUI exits
Not end of list
move end-effector to the position
read laser data and store it
Plot the points cloud
Generate mesh with 3D reconstruction
Microwave M
Read data from the table
items
Generate scanning point
Connect Laser
connected
Connect robot and move robot to start position
read position value from the data set
end of list
close connection
No item
Perform microwave measurement
Mesh from 3D reconstruction
fail
GUI exits
fail
GUI exits
Not end of list
move end-effector to the position
Plot the points cloud
Update labels
read network anaylser data and save it
plot the data depends on the check boxes

Figure 3: Scan button process. Blue color means the functions work in other thread than the main thread.

2.1. Laser scanning

When this tab is activating, the robot will preform a scanning by auto generate scanning points from values in spin boxes.

Meaning of spin boxes:

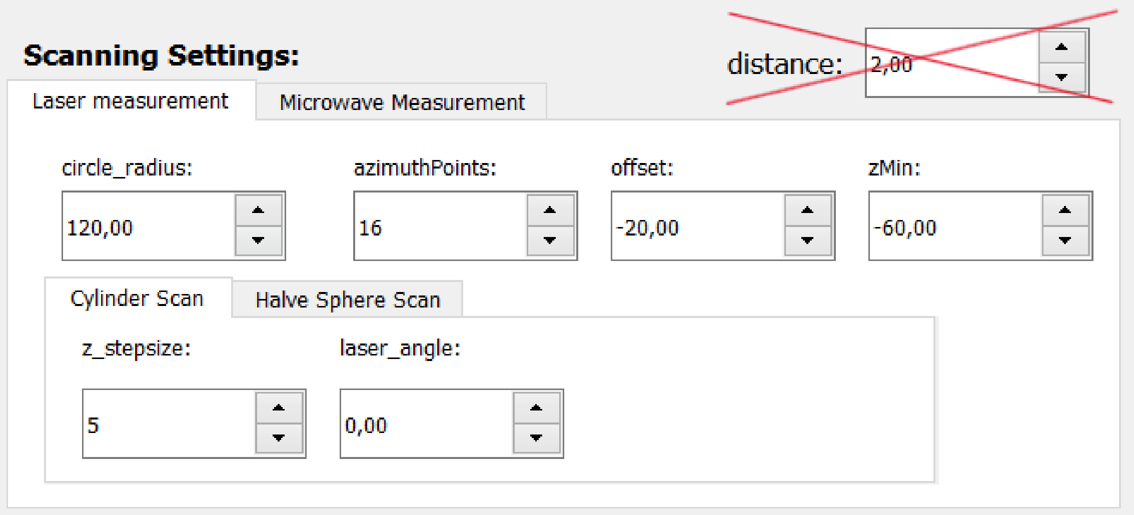
- circle_radius = Radius of the cylinder
- azimuthPoints = Number of points in the azimuth angle
- zMin = Lowest point of the cylinder
- offset = Offset in the z-axis

2.1..1 Cylinder Scan

When this tab is activating, the robot will preform a cylindrical scanning by auto generate scanning points.

Meaning of spin boxes:

- z_stepsize = Number of mm between each z-plane
- laser_angle = The angle of the end effector to point the laser (between +90 and -90)



Scanning Settings:

~~distance: 2,00~~

circle_radius: 120,00

azimuthPoints: 16

offset: -20,00

zMin: -60,00

z_stepsize: 5

laser_angle: 0,00

Figure 4: A cylindrical scanning setting

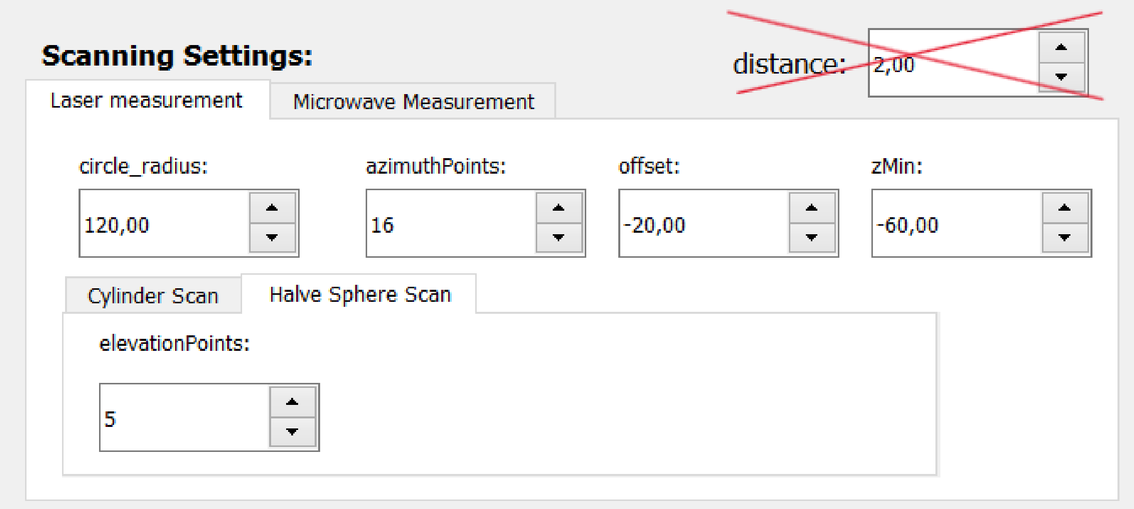
2.1..2 Halve Sphere Scan

Warning: Don't use this function when there is a linear actuator on the bottom of the object.

When this tab is activating, the robot will preform a halve spherical scanning by auto generate scanning points.

Meaning of spin boxes:

- elevationPoints = Number of points in the elevation plane



~~distance: 2,00~~

circle_radius: 120,00

azimuthPoints: 16

offset: -20,00

zMin: -60,00

elevationPoints: 5

Figure 5: A halve spherical scanning setting

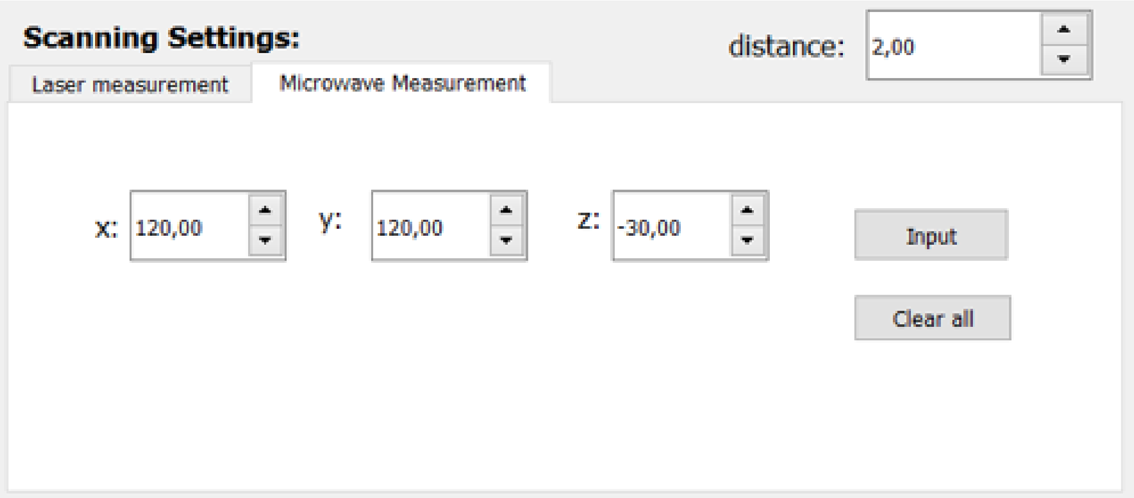
2.2. Microwave measurement

Warning: When you want to scan the point on the bottom, collision with linear actuator happen.

When this tab is activating, the robot will preform a scanning by reading the values in the Scanning Position list. If there is no item in the table, then it will start microwave measurement without moving. After measurement, robot position data will be display in Robot position and surface position labels. Also, the result of the network analyser will be display in the network viewer.

Buttons:

- Input = Read the values in the x, y, z spin box. Recalculate the coordinates to a point on the surface of the object. And print the point in the table and scanning viewer. (Warning: If there are no mesh when you pressed the button, the GUI will be crashed. Make sure you start a laser scanning or load a CSV file before the action.)
- Clear all = Remove all items in the Scanning Position list table.
- distance spin box = how far between the surface of object and the antenna in mm.



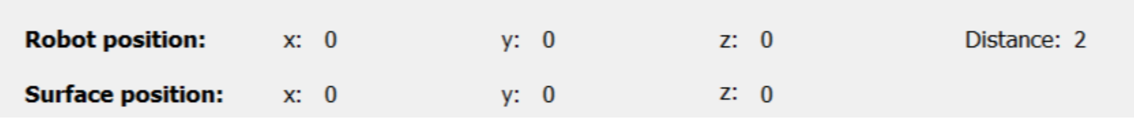
distance: 2,00

x: 120,00

y: 120,00

z: -30,00

Figure 6: When microwave measurement tab is activating

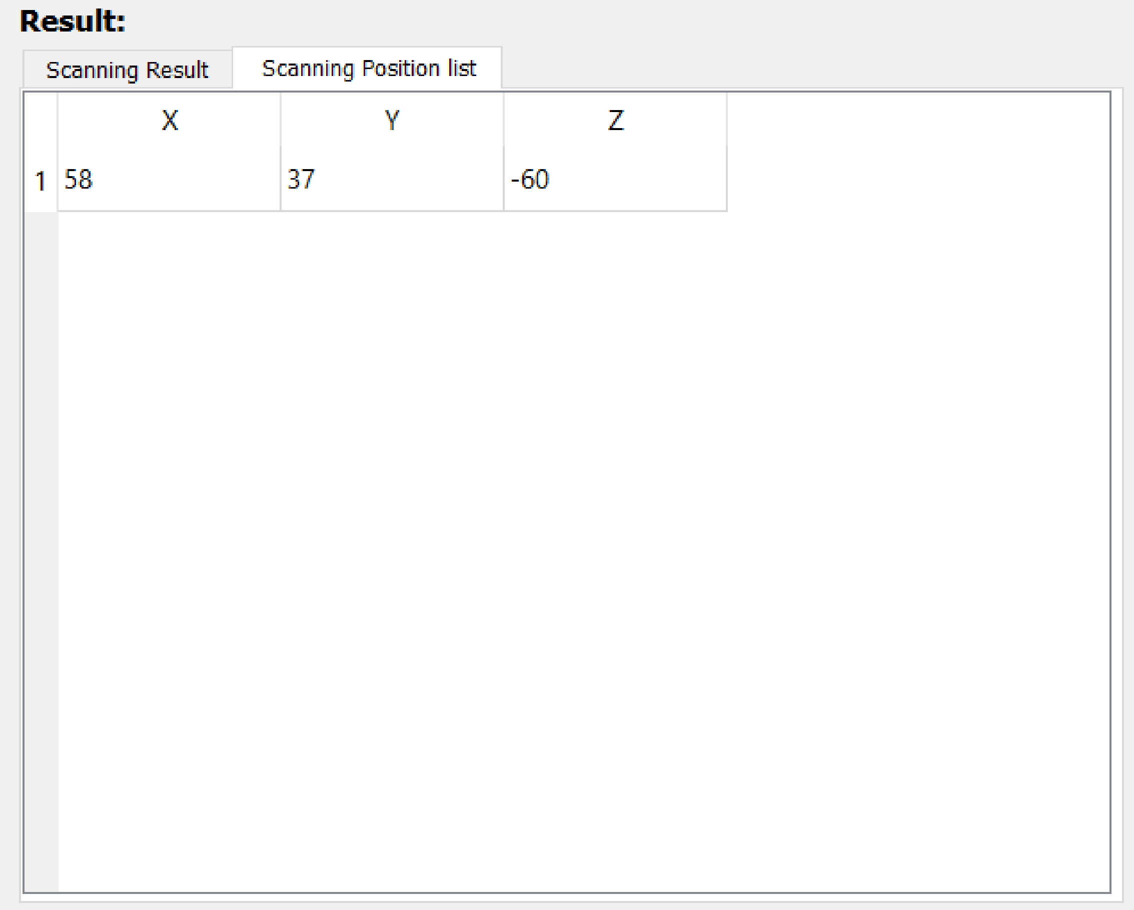


Robot position:	x: 0	y: 0	z: 0	Distance: 2
Surface position:	x: 0	y: 0	z: 0	

Figure 7: Latest scan point will display in labels if you scan the object with manual inputs.

Note:

- You are able to change the value in the table by double click the value.
- This scanning using the mesh data from 3D reconstruction. The mesh data generated by using values in results. There, you must scan the object or load data from csv using Load button.



	X	Y	Z
1	58	37	-60

Figure 8: Manual inputs in Scanning Position list

3. Stop Button

Stop the scanning, calibration or movement of robot.

Note:

- When you perform stop action, this means the thread of scanning or calibration will be terminate. But the command are still in the Yumi Controller. So the Yumi will perform actions until end of the command. Use the red stop button in the Yumi Controller if you need to stop immediately.

- Even you perform stop action, the connection between Raspberry Pi and Yumi controller is not disconnect. Therefore, you need to restart the GUI program to disconnect. Therefore, you are not able to use any function that move the Yumi after you pressed stop button.

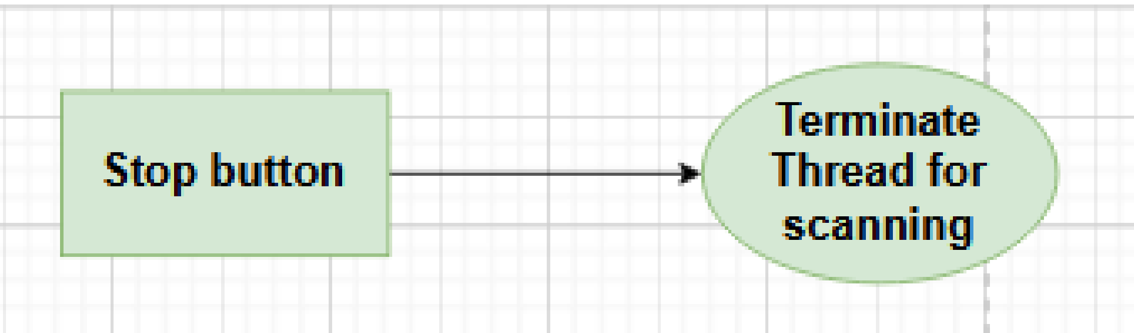


Figure 9: Stop button process. Blue color means the functions work in other thread than the main thread

4. Save Button

Save items in the Scanning Result table in Result as a csv file.

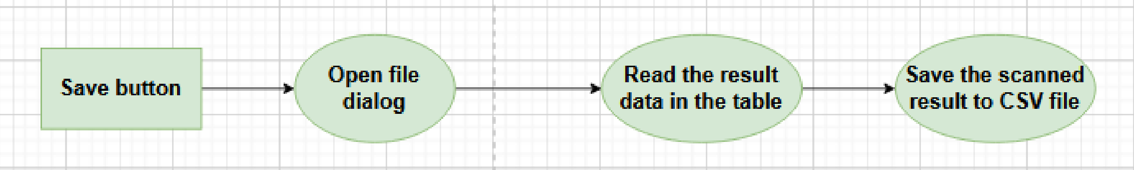

Save button
Open file dialog
Read the result data in the table
Save the scanned result to CSV file

Figure 10: Save button process

5. Load Button

Read data from a csv file and insert to the Scanning Result table. It will also plot the point cloud base on the data and perform a 3d reconstruction.

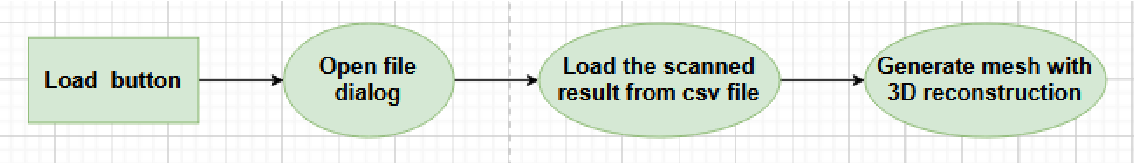

Load button
Open file dialog
Load the scanned result from csv file
Generate mesh with 3D reconstruction

Figure 11: Load button process

6. Calibration Button

Warning: It cause problem when robotic arm have too many cables.

Perform a calibration scanning and update the quaternions which are used for scan the object.

Note:

- The standard quaternion is [9.99954527e-01, 9.41712207e-03, 1.50357889e-03, 9.45543129e-06] in ClassGUI.py. This number is save in a csv file called quaternions.csv in GUI folder. After calibration, the value will be updated.

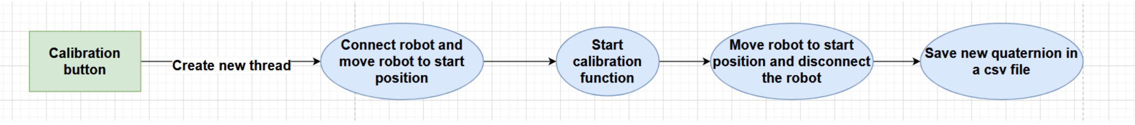

Calibration button
Create new thread
Connect robot and move robot to start position
Start calibration function
Move robot to start position and disconnect the robot
Save new quaternion in a csv file

Figure 12: Calibration button process. Blue color means the functions work in other thread than the main thread.

7. Manual Control

Functions for control the robotic arm and linear actuator manually.

7.1. robotic arm movement

The principal of robotic arm movement is to able perform small movement +-1 mm. But it always go back to the start position and then go to the specific position. Therefore, it is not good to use it. Use manual input scanning points instead. It also use same distance spin box same as the manual input scanning points.

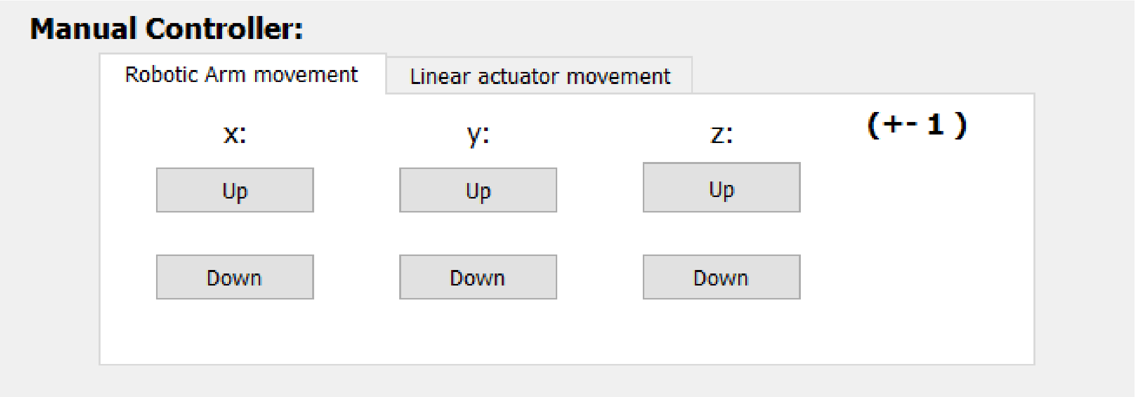


Figure 13: Manual robot controller

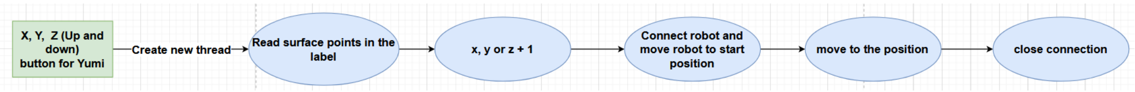

X, Y, Z (Up and down) button for Yumi
Create new thread
Read surface points in the label
x, y or z + 1
Connect robot and move robot to start position
move to the position
close connection

Figure 14: Robot control buttons process. Blue color means the functions work in other thread than the main thread.

7.2. Linear actuator movement

Functions for control the linear actuator. You are able to move the linear actuator +-1 mm or go to specific position by use Move button. Also, before you use the controller, you need to move the linear actuator down to bottom (the zero position), and then press calibration button in the tab to let the program know the linear actuator is on the zero position.

Buttons:

- Move = move to specific position using input in the spin box.
- Calibration = set the current value to 0
- Up = move linear actuator up 1mm
- Down = move linear actuator down 1mm

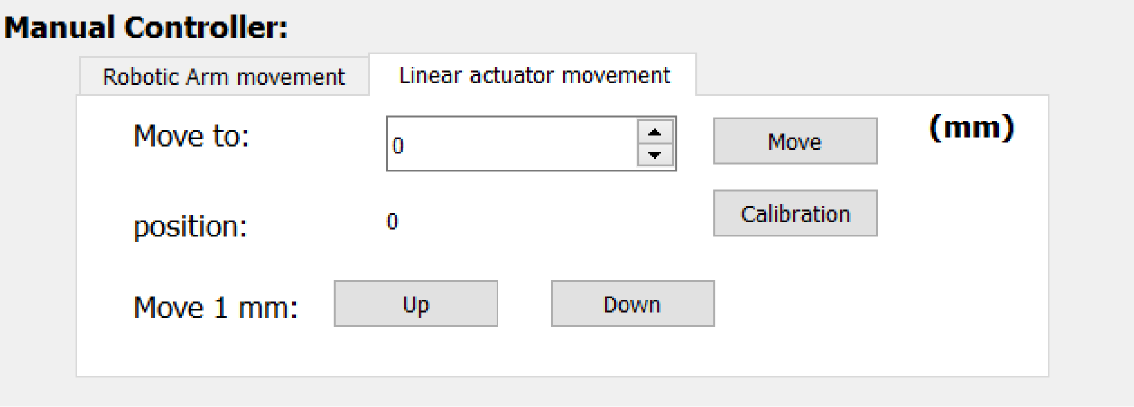


Figure 15: Linear actuator

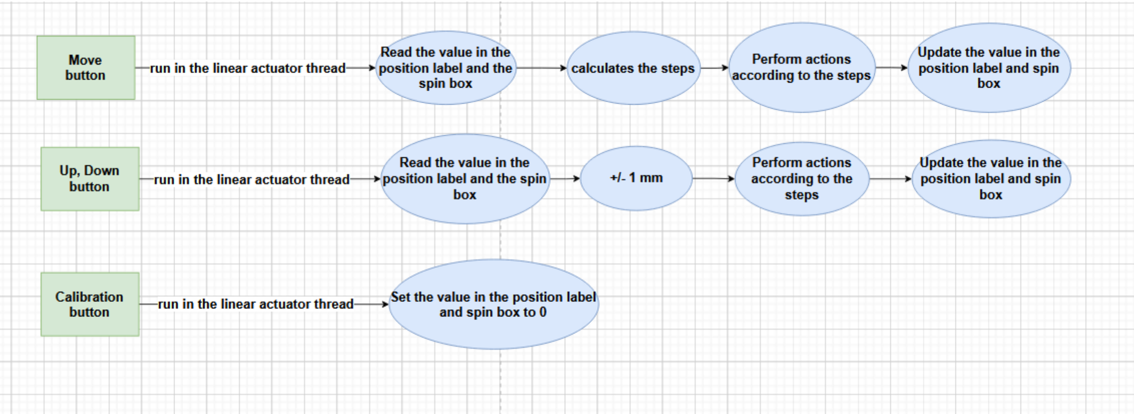

Move button
run in the linear actuator thread
Read the value in the position label and the spin box
calculates the steps
Perform actions according to the steps
Update the value in the position label and spin box
Up, Down button
run in the linear actuator thread
Read the value in the position label and the spin box
+/- 1 mm
Perform actions according to the steps
Update the value in the position label and spin box
Calibration button
run in the linear actuator thread
Set the value in the position label and spin box to 0

Figure 16: Linear actuator buttons process. Blue color means the functions work in other thread than the main thread.

8. Log

A text box that use for display messages.

9. 3D reconstruction

Because Raspberry pi does not does not support the draw function in open3D. Therefore, this functions is not available. But other functions like create mesh works. You can save the mesh to a .obj file by change the code in ClassGUI.py. And if you do, the 3D reconstruction data saves in a file called meshNew.obj.

10. Network viewer

A viewer for display the result of network analyser. It is also automatically save the result in the mw_data folder. Which result will be display base on which checkbox you had checked.

Buttons:

- plot = read a csv file in mw_data and plot it in the viewer

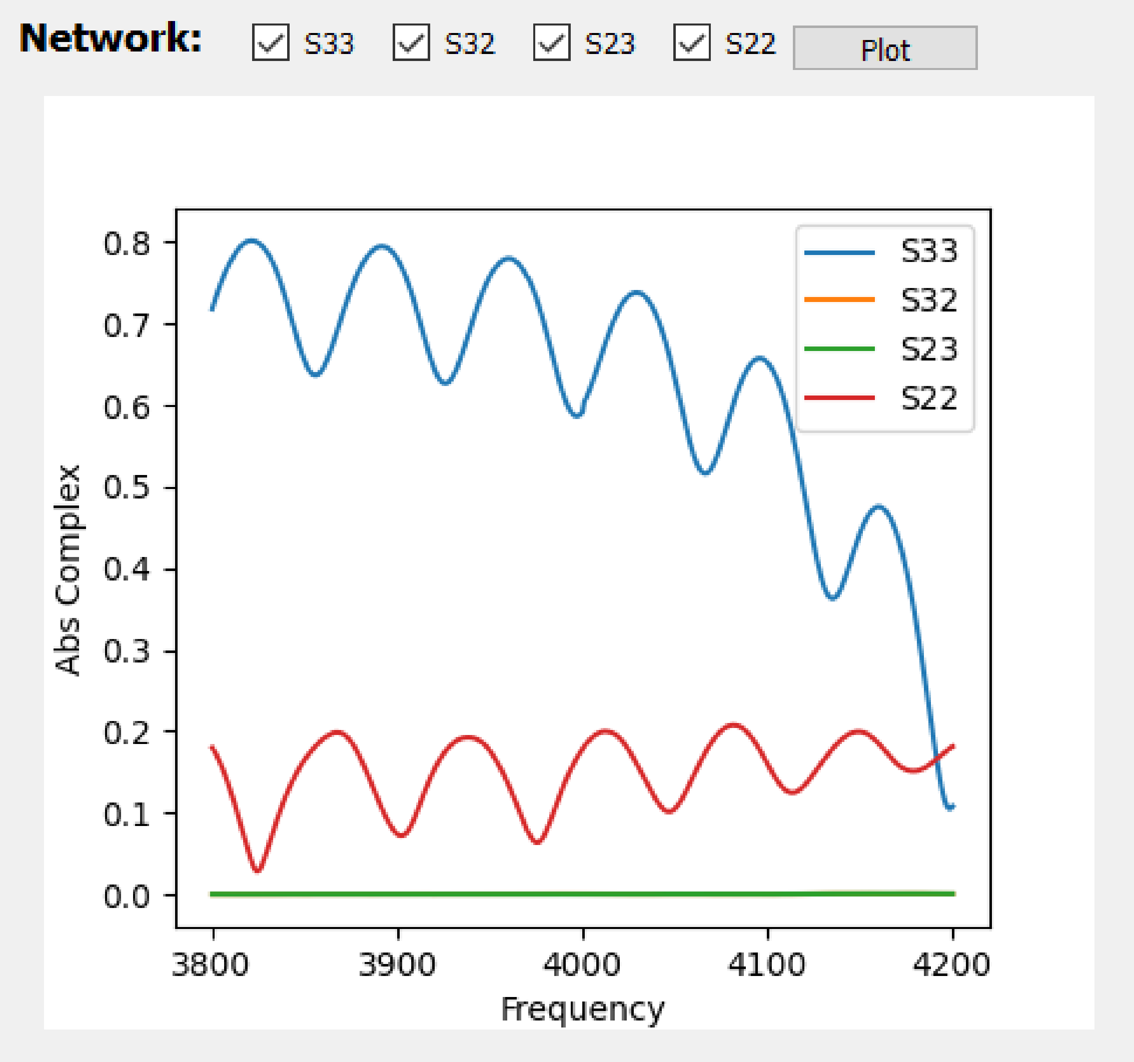


Figure 17: Network viewer