Lab-03-Sensor Robotics

Students: Alexis GIBERT / Souha GHAZOUANI

# Goals

Scanning - reconstruction of a 3D surface with the help of a 2D laser scanner and a 2-axes accelerometer.

# Readings

laserscanner\_description\_\_\_URG\_04LX\_UG01.pdf

* It utilizes an infrared laser with a wavelength of 785nm.
* The scan area is a 240º semicircle with a maximum radius of 4000mm.
* It has an angular resolution of 0.36º and outputs distance measurements at every point (683 steps).
* Principle of distance measurement is based on calculating the phase difference, providing stable measurements with minimal influence from object color and reflectance.

laserscanner\_communication\_protocol\_\_\_URG\_04LX\_UG01.pdf

|  |  |
| --- | --- |
| Sensor rotates anti-clockwise when viewed from the top.  Detection Range (E) is the maximum angle scanned by the sensor.  Angular Resolution is defined as 360 degrees divided by the Slit Division (F).  Measurement points are called Steps, with Step 0 being the first measurement point and Step D being the last. | Une image contenant texte, diagramme, cercle, ligne  Description générée automatiquement |

The Three-Character Encoding technique is applied to express data having maximum length of 18 bits. Encoding is done by separating data into upper, middle and lower 6 bits and then 30H is added to convert them into ASCII characters.

Communication Format are predefined commands are exchanged between the sensor and host. Their commands have a specific format known as the communication format.

Une image contenant texte, capture d’écran, ligne, Police

Description générée automatiquementUne image contenant texte, capture d’écran, ligne, Police

Description générée automatiquement

The [MDMS-Command] is a command used for sensor data acquisition. Laser switches on/off automatically before/after the measurement

Une image contenant texte, Police, ligne, capture d’écran

Description générée automatiquementUne image contenant texte, Police, ligne, capture d’écran

Description générée automatiquement

Note: ‘00’ Status = Command received without any Error

accelerometer\_2axis\_\_\_Phidget\_1053.pdf

* Sensor: Uses ADXL320 to measure acceleration up to ±5g across two axes.
* Measurements: Can detect both static (tilt) and dynamic (movement) acceleration.
* Use static acceleration only to calculate the tilt angle. Ignore values beyond 1G during movement.
* Angle calculation formula: θ=arcsin(o/h), where o is the acceleration measured on an axis, and h = 1G is standard gravity.
* PhidgetAccelerometer API:
  + AxisCount(): Measures on 2 axes (horizontal and vertical axes)
  + Acceleration(int AxisIndex): Measures acceleration on a specific axis
  + AccelerationMax and AccelerationMin: Define the measurable acceleration limits.
  + AccelerationChangeTrigger: Sets the threshold for detecting acceleration changes.
* Events: OnAccelerationChange responds to significant acceleration changes.
* Focus on calibration and accuracy for correct measurements.

# About the measurements

We have measured various inclinations with the lidar, as shown in the photos below:

Une image contenant intérieur, personne, ordinateur, câble

Description générée automatiquementUne image contenant personne, outil, câble, intérieur

Description générée automatiquement

The measurement records provide us with .mat files in the following format:

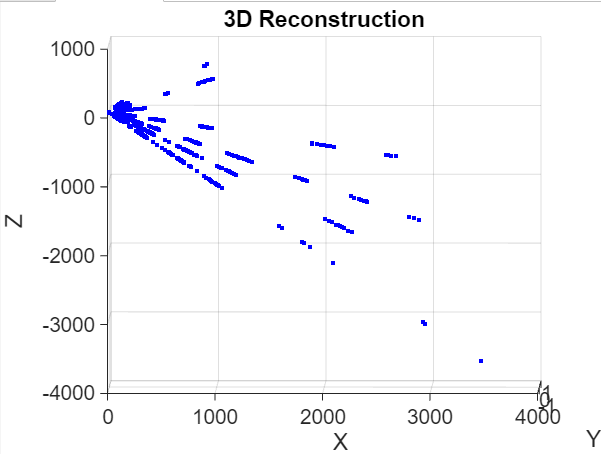
tilt\_angles = [1.633 -46.255]

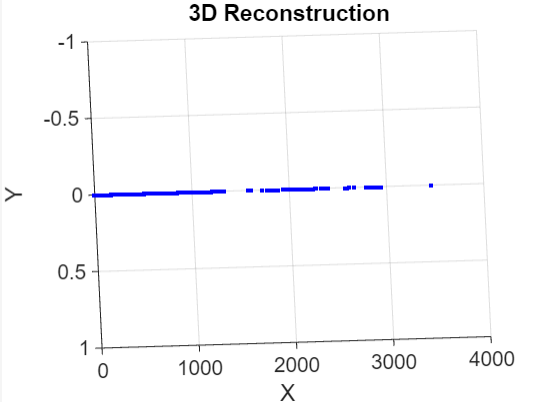
measurement\_data =[ 294 290 282 276 276 276 279 279 282 282 284 295 296 372 376 379 381 388 388 388 389 387 384 387 390 390 390 390 390 391 391 398 398 404 412 410 412 413 413 427 428 428

…

5006 2335 2285 1521 1478 1469 1462 1447 1438 1423 1394 1380 1364 1352 1345 1345 1315 1282 1110 1110 1110 3023 0 7 0 0 0 0 959 953 947 947 940 938 926]

# About the code

Une image contenant diagramme, texte, ligne, capture d’écran

Description générée automatiquement

*3D reconstruction in the ZX plane (in the left), in the isometric view (in the center) and in the YX plane (in the right)*

We attempted to reconstruct the 3D box in MATLAB using the following code, but we were only able to reconstruct the data in 2D under different tilt\_angles.

clear all

num\_files = 8; % Number of data files to merge

all\_points = []; % Initialize empty matrix to store all points

for i = 1:num\_files % Iterate through all data files

filename = sprintf('measurement\_data\_case1%d.mat', i);

data = load(filename);

phi = data.tilt\_angles(2); % Vertical angle

measurement\_data = data.measurement\_data;

% Convert measurements and angles into XYZ coordinates

Z = measurement\_data .\* sind(phi); % Using vertical angle for Z

X = measurement\_data .\* cosd(phi); % Using vertical angle for X

all\_points = [all\_points; [X', zeros(size(X')), Z']];

end

% Display 3D model

figure;

scatter3(all\_points(:,1), all\_points(:,2), all\_points(:,3), 'b.');

xlabel('X');

ylabel('Y');

zlabel('Z');

title('3D Reconstruction');

We thought about replacing the Y with measurement\_data to display the data in 3D, but an error occurs:

Error using [**horzcat**](matlab:matlab.lang.internal.introspective.errorDocCallback('horzcat'))Dimensions of arrays being concatenated are not consistent.

We also considered the possibility of developing an algorithm based on the relationship between the position relative to the number of data points (684), the lidar's viewing angle (240°), and the angular resolution (0.36º) to accurately visualize the data on the lidar plane. However, we opted for a simpler approach due to concerns about potential complexity.