YDLIDAR SDK Build Status Build status codebeat C

Introduction

YDLIDAR(https://www.ydlidar.com/) series is a set of high-performance and low-cost LIDAR sensors, which is the perfect sensor of 2D SLAM, 3D reconstruction, multi-touch, and safety applications.

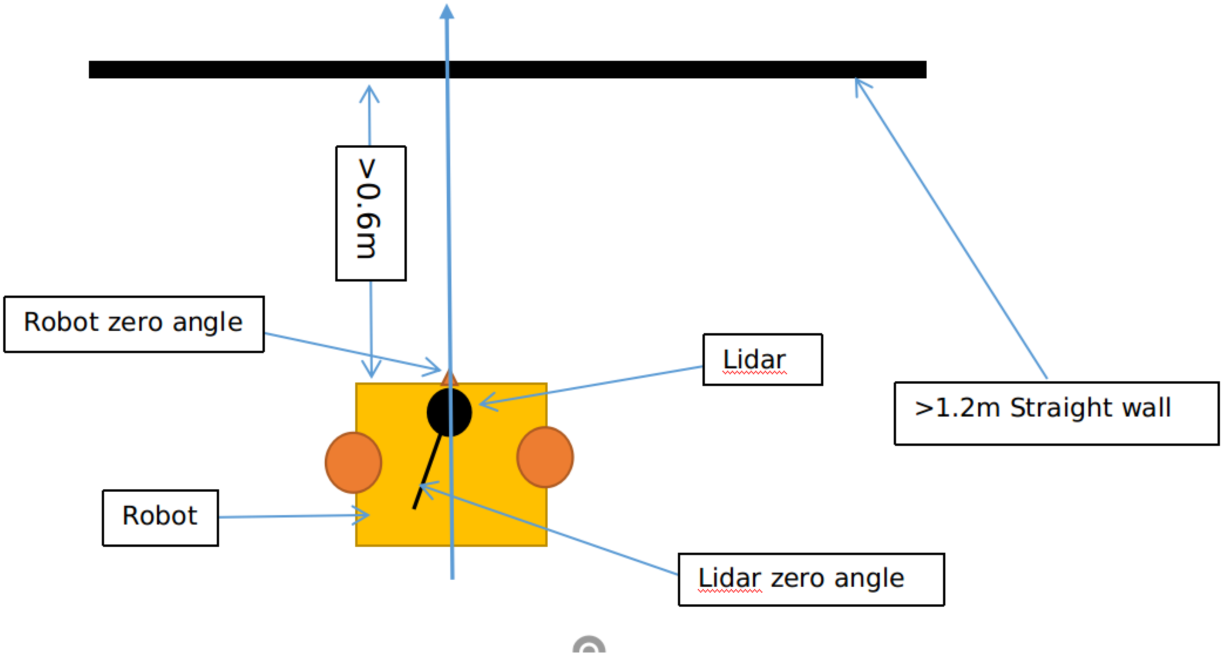
If you are using ROS (Robot Operating System), please use our open-source ROS Driver .

Release Notes

Title	Version	Data
SDK	2.1.1	2020-03-06

- [feat] SDK exceptional case of false generation, the specific holding time from “few” seconds to 30 seconds.
- [fixed] fixed local_scan size.
- [feat] the timeout is greater than the default value before the exception message is printed.

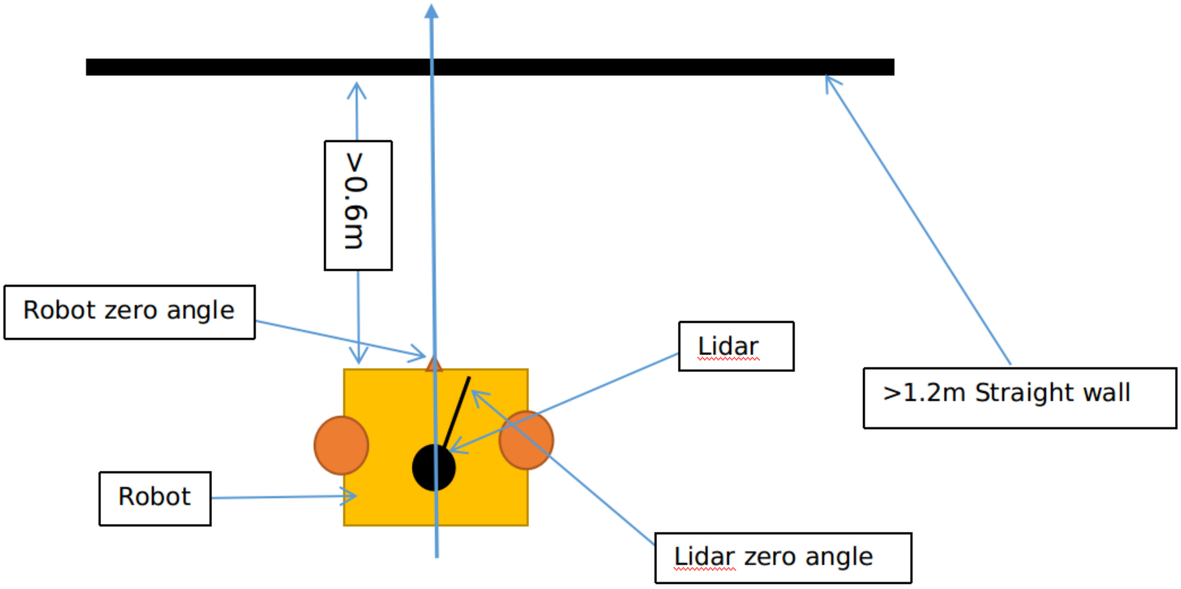
Zero Angle calibration



The Robot is put into the calibration tool required in the above figure to ensure that the zero angle direction of the robot is perpendicular to the wall. If the difference between the Lidar zero angle direction and the robot zero angle direction is greater than 90 degrees, the same as the figure above. the `setRobotLidarOpposite` in SDK is set to true.

```
CYdLidar laser;
laser.setRobotLidarOpposite(true);
```

otherwise, the difference is less than 90 degrees, the `setRobotLidarOpposite` in SDK is set to false.



```
CYdLidar laser;
laser.setRobotLidarOpposite(false);
```

How to start zero angle correction

SDK turns off correction model by default. If you want to start by execution the `setStartRobotAngleOffset` command once, SDK will start the correction of zero angle.

```
  CYdLidar laser;
  laser.setStartRobotAngleOffset();
```

`isRobotAngleOffsetCorrected` command will return true if the correction is complete.

```
  CYdLidar laser;
  if(laser.isRobotAngleOffsetCorrected()) {
        printf("Calibraion successful...\n");
  }
```

The calibration value will be saved in the file set by `setCalibrationFileName` command, you must set this value, Example:

```
  std::string calibration_filename = "Your's
path/LidarAngleCalibration.ini";
  CYdLidar laser;
  laser.setCalibrationFileName(calibration_filename);//Zero angle offset
filename
```

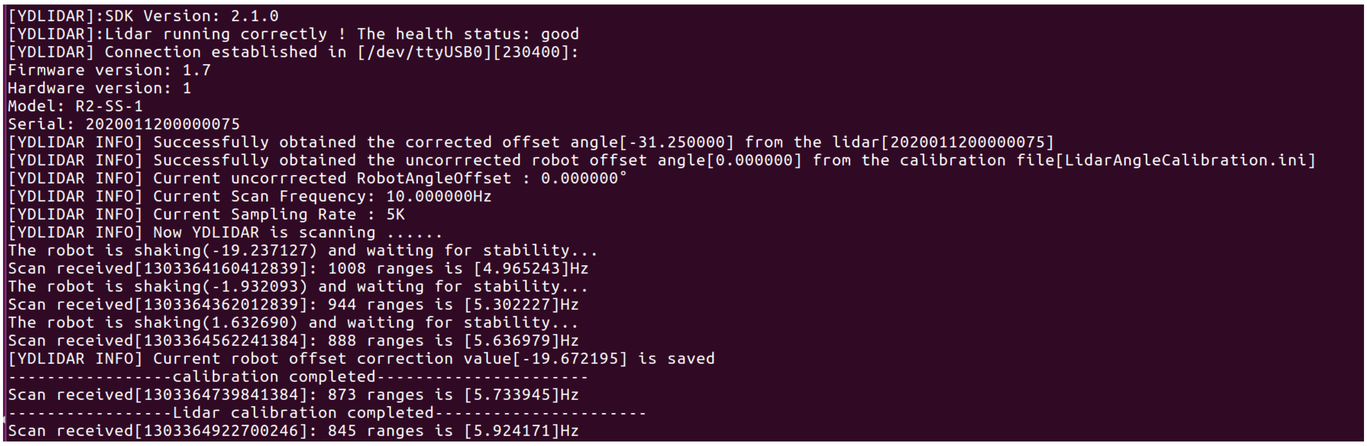
Note: The file format is an ini file,Requires .ini as extension.

After the calibration is successful, the next time the SDK starts, it will read the calibration value in the file, add this offset value to the Lidar data, and the output lidar data is already calibrated.

Note:

- 1. Zero Angle direction of the robot needs to be vertical to the wall.
- 2. `setCalibrationFileName` command file path and file name need to be set correctly.
- 3. `setRobotLidarOpposite` command parameters need to be set correctly.
- 4. calibration values are stored on the robot's internal memory area.
- 4. After the calibration is completed, the calibration model will be automatically closed, if you neeed to calibrate again, you can start the recalibration by `setStartRobotAngleOffset` command.

Result as follows:



```
[YDLIDAR]:SDK Version: 2.1.0
[YDLIDAR]:Lidar running correctly ! The health status: good
[YDLIDAR] Connection established in [/dev/ttyUSB0][230400]:
Firmware version: 1.7
Hardware version: 1
Model: R2-SS-1
Serial: 2020011200000075
[YDLIDAR INFO] Successfully obtained the corrected offset angle[-31.250000] from the lidar[2020011200000075]
[YDLIDAR INFO] Successfully obtained the uncorrrected robot offset angle[0.000000] from the calibration file[LidarAngleCalibration.ini]
[YDLIDAR INFO] Current uncorrrected RobotAngleOffset : 0.000000°
[YDLIDAR INFO] Current Scan Frequency: 10.000000Hz
[YDLIDAR INFO] Current Sampling Rate : 5K
[YDLIDAR INFO] Now YDLIDAR is scanning ......
The robot is shaking(-19.237127) and waiting for stability...
Scan received[1303364160412839]: 1008 ranges is [4.965243]Hz
The robot is shaking(-1.932093) and waiting for stability...
Scan received[1303364362012839]: 944 ranges is [5.302227]Hz
The robot is shaking(1.632690) and waiting for stability...
Scan received[1303364562241384]: 888 ranges is [5.636979]Hz
[YDLIDAR INFO] Current robot offset correction value[-19.672195] is saved
-----------------calibration completed---------------------
Scan received[1303364739841384]: 873 ranges is [5.733945]Hz
-----------------Lidar calibration completed---------------------
Scan received[1303364922700246]: 845 ranges is [5.924171]Hz
```

How to Set Rid Distortion Value.

SDK can filter custom angular area Lidar data,`setIgnoreArray` command use examples as follows:

- 1.filter 10 to 20 degrees and 50 to 60 degrees

```
CYdLidar laser;
std::vector<float> ignore_array;
// 10 to 20
ignore_array.push_back(10);
ignore_array.push_back(20);
//50 to 60
ignore_array.push_back(50);
ignore_array.push_back(60);
laser.setIgnoreArray(ignore_array);
```

- 2.filter 10 to 20 degrees, 180 to 200 degrees and 320 to 330 degrees.

```
CYdLidar laser;
std::vector<float> ignore_array;
// 10 to 20
ignore_array.push_back(10);
ignore_array.push_back(20);
//180 to 200
ignore_array.push_back(180);
ignore_array.push_back(200);
//320 to 330
ignore_array.push_back(320);
ignore_array.push_back(330);
laser.setIgnoreArray(ignore_array);
```

Filtering angles need to be entered in pairs.

Note:

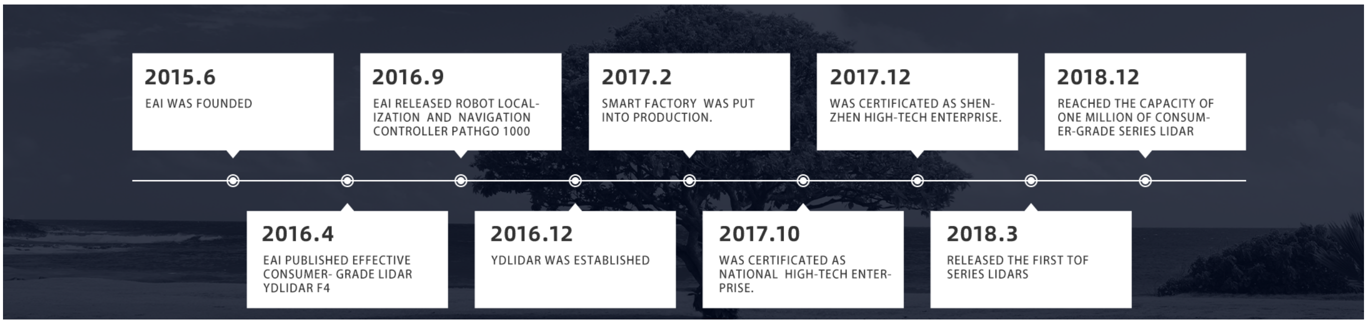
- The Input filter angles can be saved in the robot's internal memory, When starting sdk, the filter angles is read out from the file and input into the SDK. This applies to both the R2 and S2 Lidars.

Support

You can get support from YDLidar with the following methods:

- Send email to support@ydlidar.com with a clear description of your problem and your setup
- Github Issues

Contact EAI



2015.6 EAI WAS FOUNDED

2016.4 EAI PUBLISHED EFFECTIVE CONSUMER- GRADE LIDAR YDLIDAR F4

2016.9 EAI RELEASED ROBOT LOCALIZATION AND NAVIGATION CONTROLLER PATHGO 1000

2016.12 YDLIDAR WAS ESTABLISHED

2017.2 SMART FACTORY WAS PUT INTO PRODUCTION.

2017.10 WAS CERTIFICATED AS NATIONAL HIGH-TECH ENTERPRISE.

2017.12 WAS CERTIFICATED AS SHENZHEN HIGH-TECH ENTERPRISE.

2018.3 RELEASED THE FIRST TOF SERIES LIDARS

2018.12 REACHED THE CAPACITY OF ONE MILLION OF CONSUMER-GRADE SERIES LIDAR

If you have any extra questions, please feel free to contact us