YDLIDAR

G2
USER MANUAL

Shenzhen EAI Technology Co.,Ltd.

CONTENTS

1 YDLIDAR G2 LIDAR DEVELOPMENT KIT

The development kit of YDLIDAR G2 lidar (hereinafter referred to as G2) is an accessory tool provided for performance evaluation and early development of the G2. Through the G2 development kit, and with the evaluation software, users can observe point cloud data scanned by G2 on your environment or development on the SDK.

1.1 Development Kit

The G2 development kit has the following components：

G2 Lidar   USB Type-C Cable   USB Adapter Board

FIG 1 YDLIDAR G2 DEVELOPMENT KIT

CHART 1 YDLIDAR G2 LIDAR DEVELOPMENT KIT DESCRIPTION

Item	Qty.	Description
G2 lidar	1	Standard version of the G2 Lidar, with internal integrated motor drive, which can realize motor stall control and motor control
USB Type-C Cable	1	Use with USB adapter board to connect G2 and PC. USB cable is both a power supply cable and a data cable
USB Adapter Board	1	USB to UART, convenient for G2 and PC to quickly interconnect. Meanwhile, it supports serial port DTR signal to control G2 motor rotation and stop. In addition, a Micro USB power interface (PWR) for auxiliary power supply is provided

Note: USB Adapter board has two USB TYPE C interface：USB_DATA、USB_PWR.

USB_DATA: Data power supply multiplexing interface. In most cases, this interface can be used to meet power and communication requirements.

USB_PWR: Auxiliary power supply interface. The USB interface of some development platforms has weak current drive capability. At this time, auxiliary power supply can be used.

2 USAGE UNDER WINDOWS

2.1 Device Connection

When G2 is evaluated and developed under windows, G2 and PC need to be interconnected. The specific process is as follows:
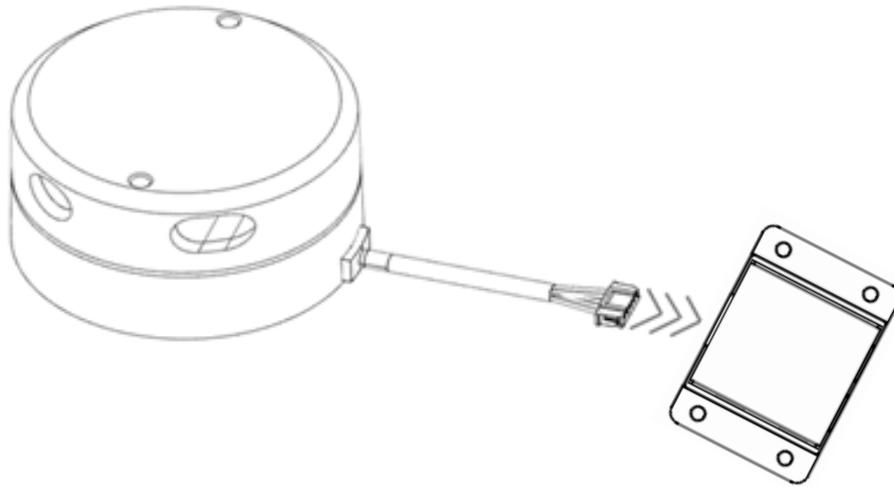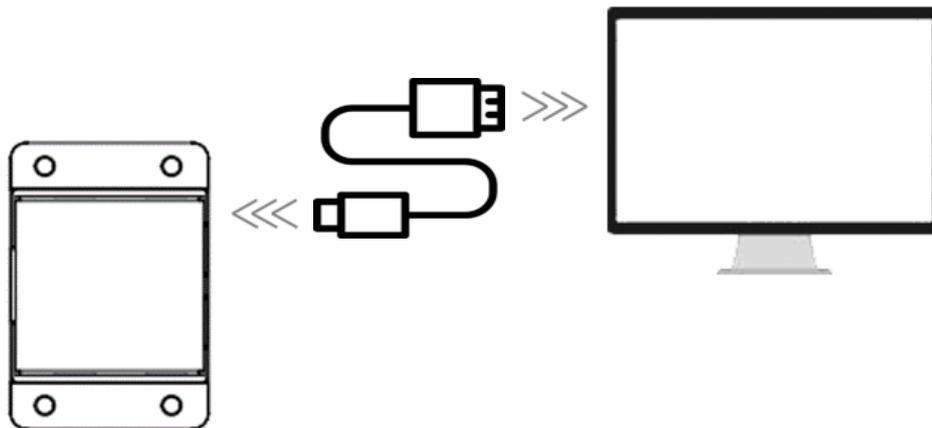

FIG 2 YDLIDAR G2 DEVICE CONNECTION STEP 1

FIG 3 YDLIDAR G2 DEVICE CONNECTION STEP 2

Connect the adapter board with G2 first, then connect the USB cable to the USB port of the adapter board and the PC. Note that the Type-C interface of the USB cable is connected to the USB_DATA of the USB interface board, and the idle mode is used after G2 is powered on. The motor does not rotate.

The drive current of the USB interface of some development platforms or PC is not sufficient. G2 need to be connected to the auxiliary power supply of +5V, otherwise the lidar will be abnormal.
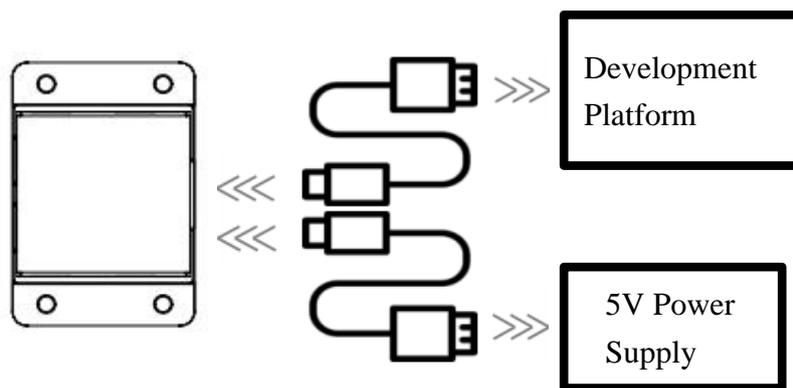
FIG 4 YDLIDAR G2 AUXILIARY POWER SUPPLY

2.2 Driver Installation

To evaluate and develop the G2 under Windows, you need to install the serial port driver of the USB adapter board. The USB adapter board of this kit adopts CP2102 chip to realize serial port (UART) to USB signal conversion. Its driver can be downloaded from our official website or downloaded from the official website of Silicon Labs.

https://www.ydlidar.com/dowfile.html?id=97

http://cn.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

After decompressing the driver package, run the CP2102's Windows driver installation file (exe file under CP210x_VCP_Windows). Please select the 32-bit version (x86) or 64-bit version (x64) installation program according to the version of the windows operating system.

FIG 5 YDLIDAR G2 DRIVER VERSION SELECTION

Double-click the exe file and follow the prompts to install it.

FIG 6 YDLIDAR G2 DRIVER INSTALLING

After the installation is complete, you can right-click on [My Computer] and select [Properties]. On the open [System] screen, select [Device Manager] from the left menu to access the [Device Manager].

Expand [Port] to see the serial port name corresponding to the identified USB adapter, that is, the driver installation is successful. The following figure shows COM3. (Note that the port must be checked in case of G2 and PC interconnection).
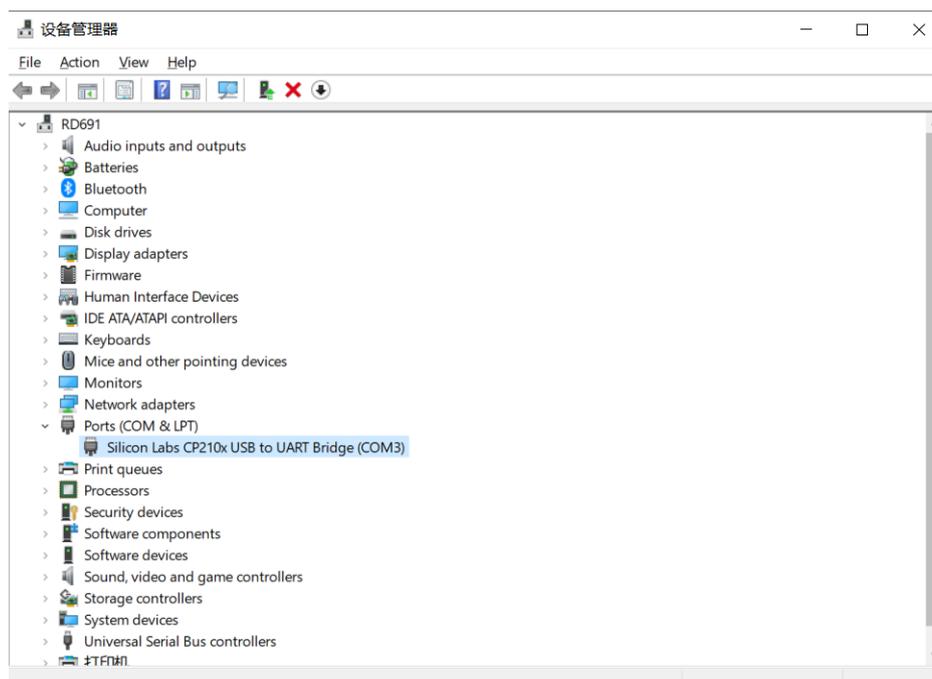

FIG 7 YDLIDAR G2 DRIVER INSTALLATION CHECK

2.3 Evaluation Software Usage

YDLIDAR provides G2 real-time scanning point cloud data visualization software LidarViewer. Using this software, users can intuitively observe the scanning effect of G2. YDLIDAR provides G2 real-time point cloud data and real-time scanning frequency. At the same time, G2 version information can be read, and the scanned data can be saved offline to an external file for further analysis. Visualization software download link: https://www.ydlidar.com/Public/upload/download/TOOL.zip

Before using the YDLIDAR software, make sure that the G2 USB adapter board serial port driver is installed successfully, and interconnect the G2 with the USB port of the PC. Run the evaluation software: LidarViewer.exe, select the corresponding serial port number and model number. Meanwhile, users could choose language on the top right corner.
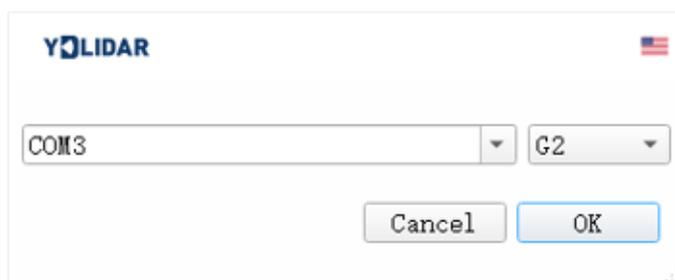

FIG 8 YDLIDAR G2 EVALUATION SOFTWARE

If there is no G2 model, please refer to the following parameter to add G2 model:

FIG 9 YDLIDAR G2 PARAMETER CONFIGURATION
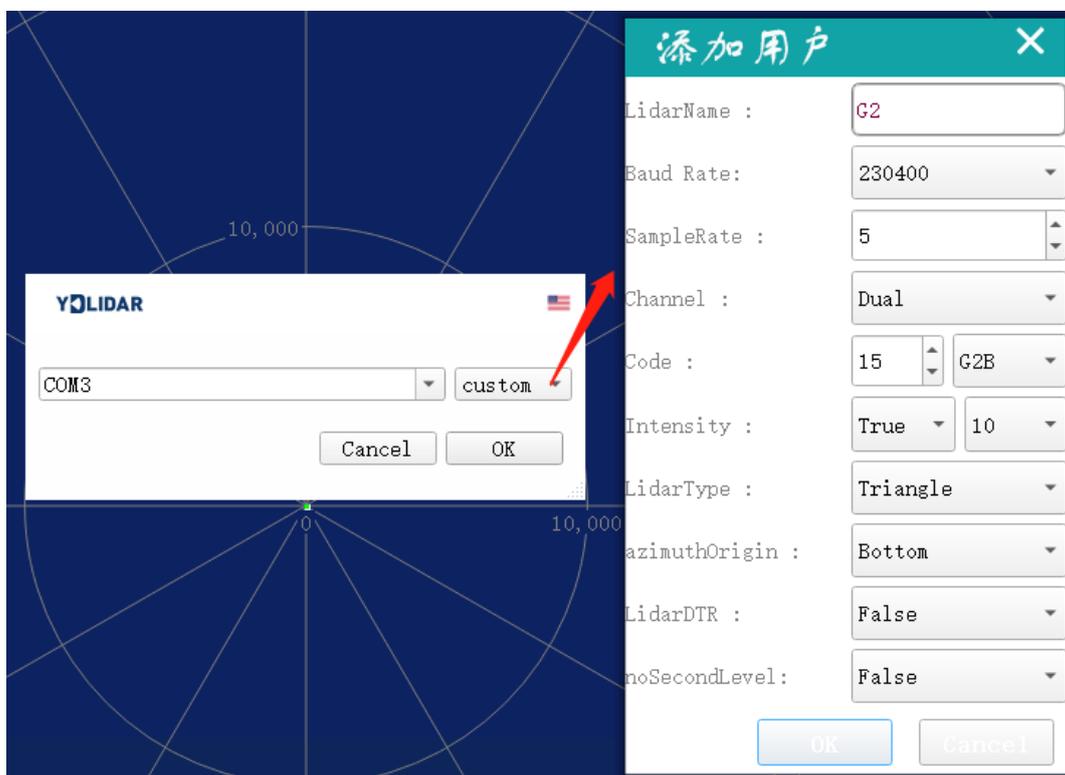
If the connection is correct, you will see the following screen：
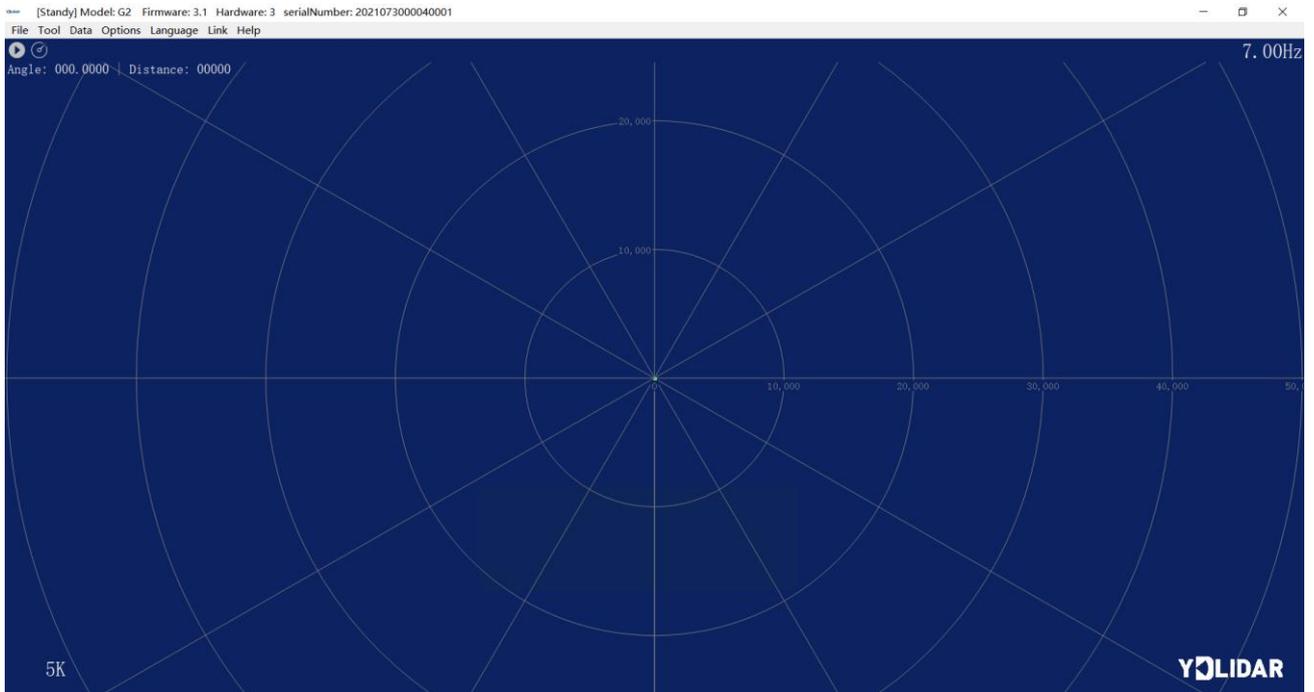

FIG 10 POINTCLOUD VIEWER INTERFACE

2.3.1 Start Scanning

Click ▶ to start scanning and display the environment point cloud, the upper left corner displays the angle & distance information of the red line position (unit: mm). Click ⏸ to stop it, as shown below:
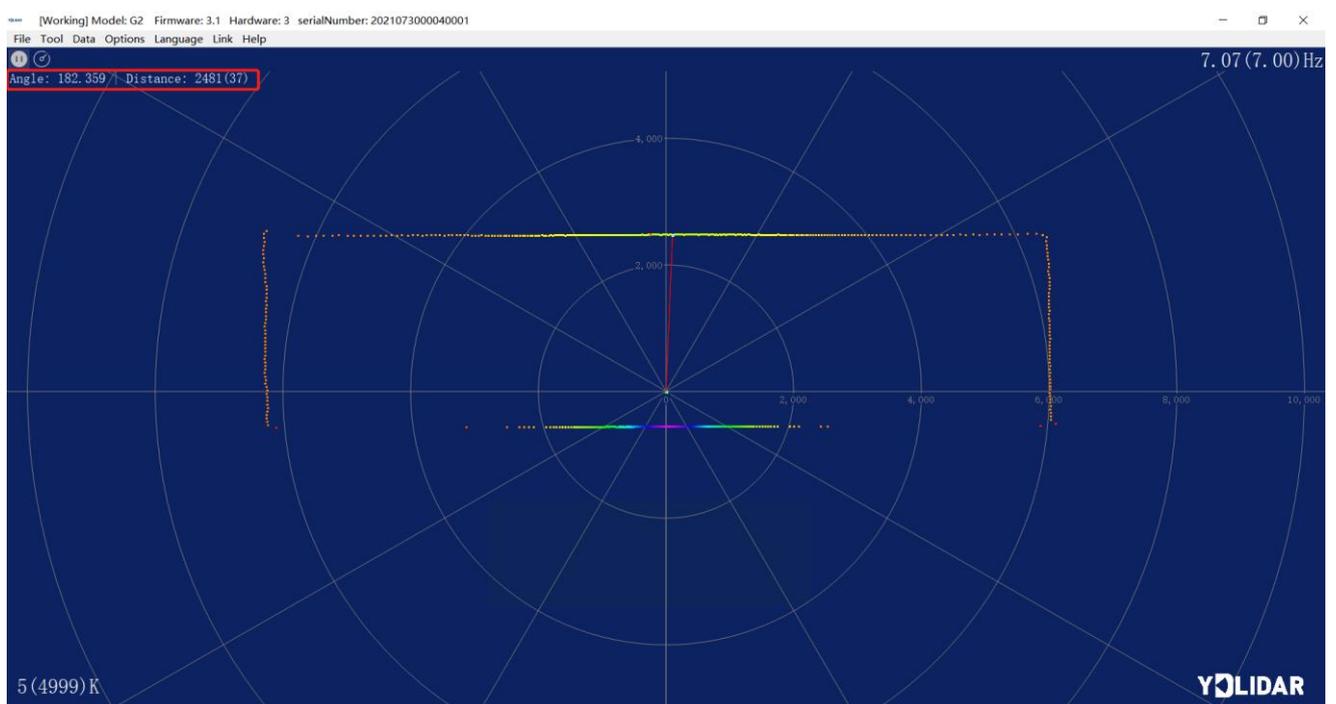

FIG 11 LIDAR SCANNING POINT CLOUD DISPLAY

2.3.2 Data Storage

During lidar scanning, click [File] in the main menu, select [Export to Excel], and save point cloud data according to the prompts. Then the system will save the point cloud information scanned in a circle in Excel format.

FIG 12 SAVE DATA

2.3.3 Scan Frequency

⊘is used to adjust the scanning frequency (motor speed) of the lidar. Click this control, the system will pop up a frequency setting bar, which can be automatically adjusted according to requirements. When the lidar is in the scan state, you need to click the scan control again after adjusting the scan frequency. The speed can be adjusted by dragging the control, the control of scanning frequency is shown in the figure below

FIG 13 SCAN FREQUENCY ADJUSTMENT

2.3.4 Display Mean and Standard Deviation

Click [Tools] in the main menu, then select [Mean And STD] - [View].

FIG 14 YDLIDAR G2 DISPLAY MEAN AND STANDARD DEVIATION

Choose one according to your needs, move the mouse to the test position, right-click the pop-up menu, and select [Lock Mouse Tracking].
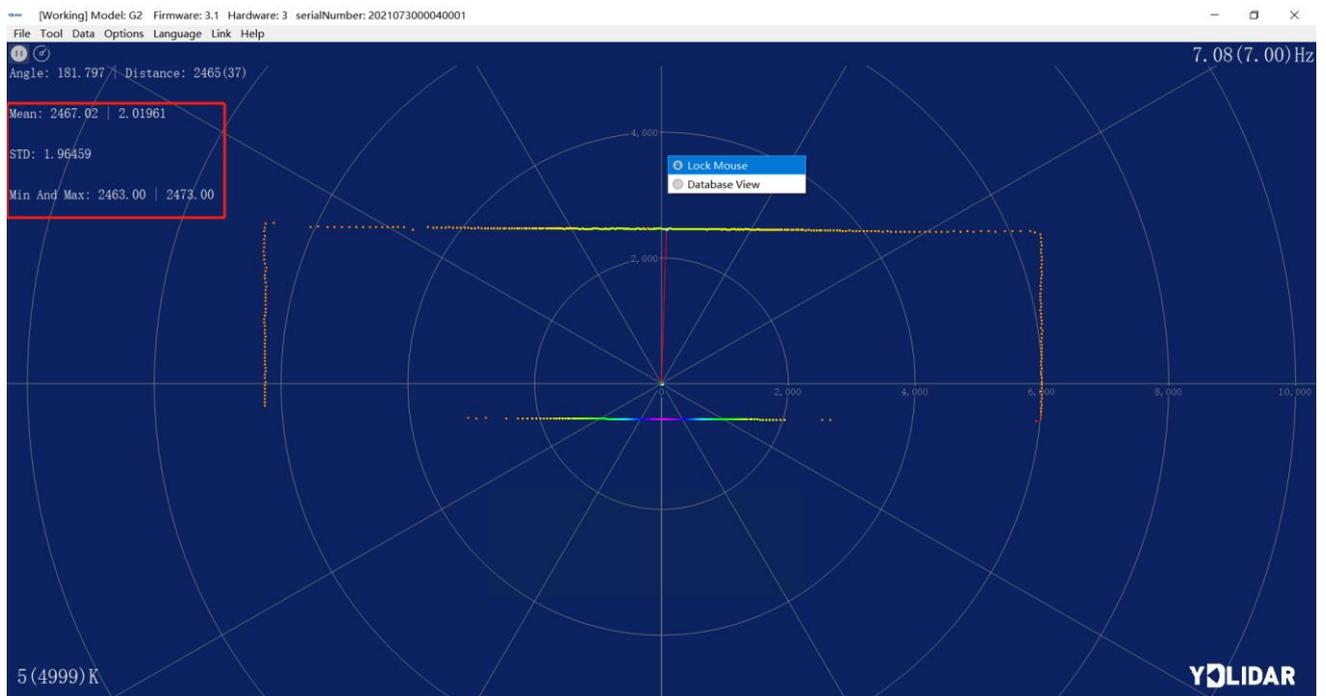
FIG 15 LOCK MOUSE TRACKING

2.3.5 Display Intensity Value

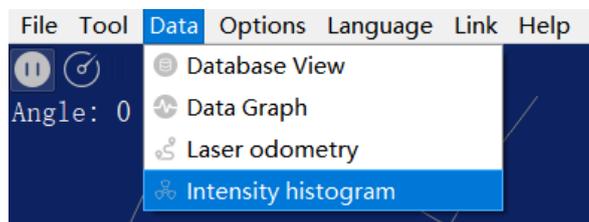Click [Data] in the main menu and select [Intensity histogram].

FIG 16 DISPLAY INTENSITY VALUE

The main window is displayed as follows, the intensity values of a total of 100 points on the left and right of the mouse lock position.
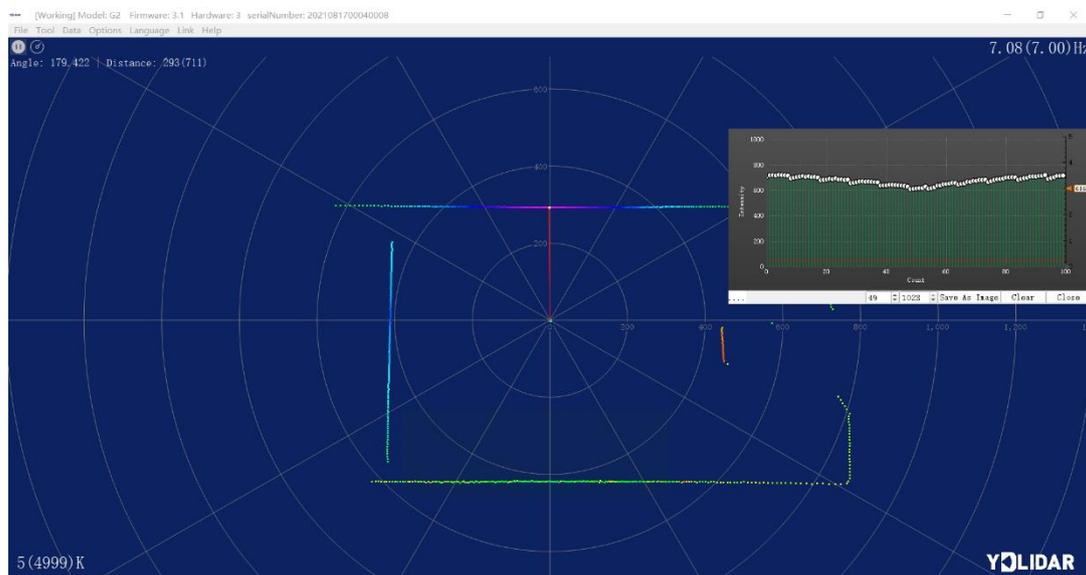
FIG 17 DISPLAY INTENSITY HISTOGRAM

2.3.6 Play and Record

Click [Tools] in the main menu, then select [Record and Play].

FIG 18 RECORD AND PLAY

The main window is displayed ⏮▶⏭◉ as follows:

To record lidar data, click ◉ to start recording, and click ● to stop recording.

In non-scanning mode, click ▶ to start play.

The play process is as follows:
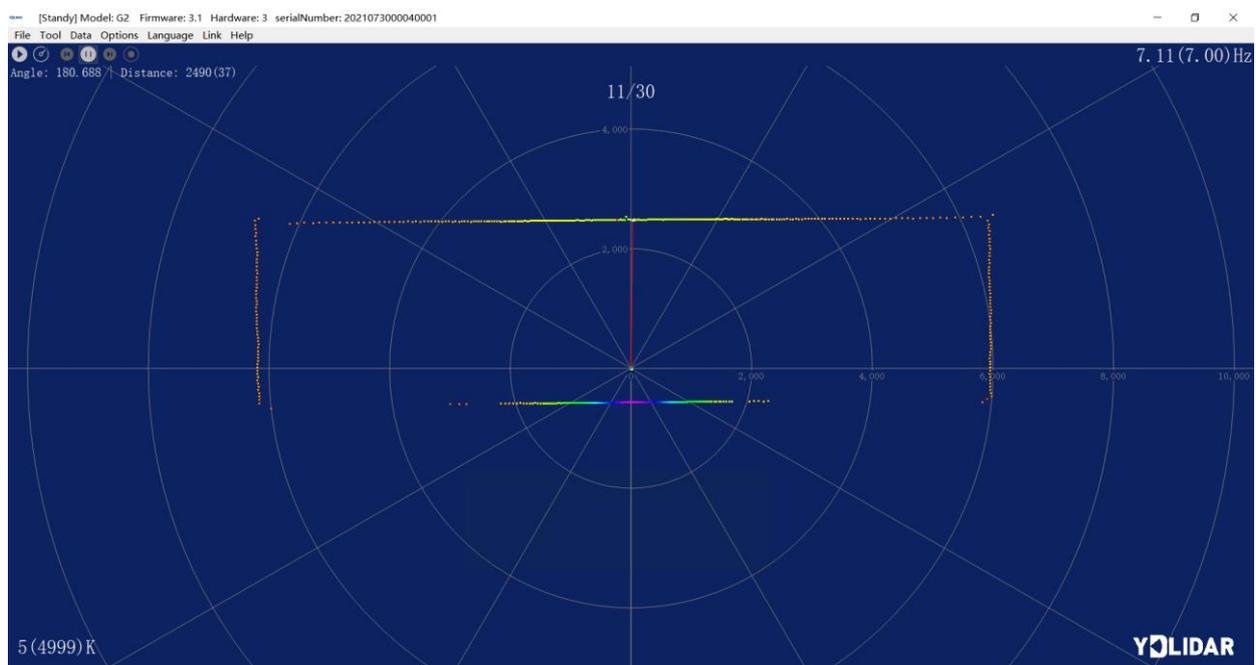
FIG 19 PLAY PROCESS

2.3.7 Debug

Click [Tools] in the main menu, and then select [DebugON] to output the raw lidar data to the "viewer_log.txt" and "viewer_log_err.txt" files.
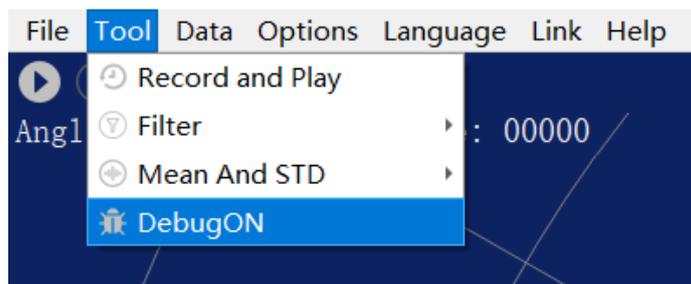
FIG 20 START DEBUGGING

2.3.8 Filter

Click [Tools] in the main menu, and then select [Filter] to add lidar data filtering algorithm.
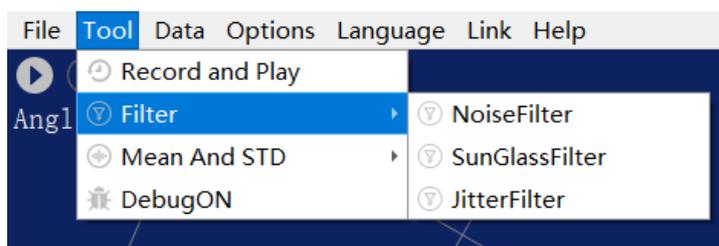
FIG 21 FILTER SETTING

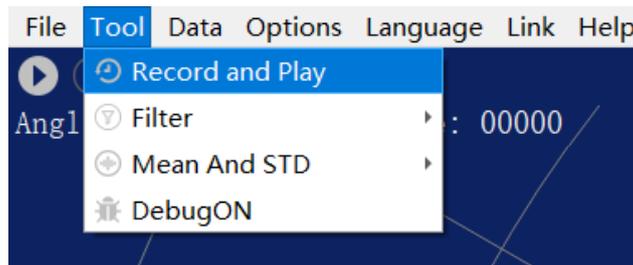Note: For more functions of LidarViewer, please click [Help], select [More Information], and learn more about how to use it.

3 LINUX ROS OPERATION

There are many Linux versions，this article only uses Ubuntu 18.04, Melodic version ROS as an example.

SDK driver address：

https://github.com/YDLIDAR/YDLidar-SDK

ROS driver address：

https://github.com/YDLIDAR/ydlidar_ros_driver

3.1 Device Connection

Under Linux, the G2 and PC interconnect processes are consistent with those under Windows. See Device Connection under Window.

3.2 Compile and Install YDLidar-SDK

ydlidar_ros_driver depends on the YDLidar -SDK library. If you have never installed the YDLidar -SDK library, or it has expired, you must first install the YDLidar-SDK library. If you have the latest version of YDLidar -SDK installed, please skip this step, then go to the next step.

```
$ git clone https://github.com/YDLIDAR/YDLidar-SDK.git

$ cd YDLidar-SDK/build

$ cmake ..

$ make

$ sudo make install
```

3.3 ROS Driver Installation

1) Cloning GitHub's ydlidar_ros_driver Package:

```
$ git clone https://github.com/YDLIDAR/ydlidar_ros_driver.git
ydlidar_ws/src/ydlidar_ros_driver
```

2) Build the ydlidar_ros_driver software package:

```
$ cd ydlidar_ws

$ catkin_make
```

3) Package environment Settings:

```
$ source ./devel/setup.sh
```

Note: Add permanent workspace environment variables. It will be very convenient if ROS environment variables are automatically added to your bash session every time you start a new shell:

```
$ echo "source ~/ydlidar_ws/devel/setup.bash" >> ~/.bashrc

$ source ~/.bashrc
```

4) Verify that your package path is set, echo the ROS_PACKAGE_PATH variable.

```
$ echo $ROS_PACKAGE_PATH
```

You should see something like this: /home/tony/ydlidar_ws/src:/opt/ros/melodic/share

5) Create Serial Port Alias [Optional]

```
$ chmod 0777 src/ydlidar_ros_driver/startup/*

$ sudo sh src/ydlidar_ros_driver/startup/initenv.sh
```

Note: After completing the previous operation, re-insert the LiDAR again.

3.4 Run the ydlidar_ros_driver

Run ydlidar_ros_driver with startup file, as shown below:

```
$ roslaunch ydlidar_ros_driver G2.launch
```

3.5 RVIZ View Scan Results

Run the launch file and open rviz to view the G2 scan results, as shown in the figure below:

```
$ roslaunch ydlidar_ros_driver lidar_view.launch
```

Note: G4 lidar is taken as an example by default. If other types of lidars are used, lidar.launch in the lidar_view.launch file needs to be changed to the corresponding **.launch file. (If G2 radar is used, it needs to be changed to G2.launch)
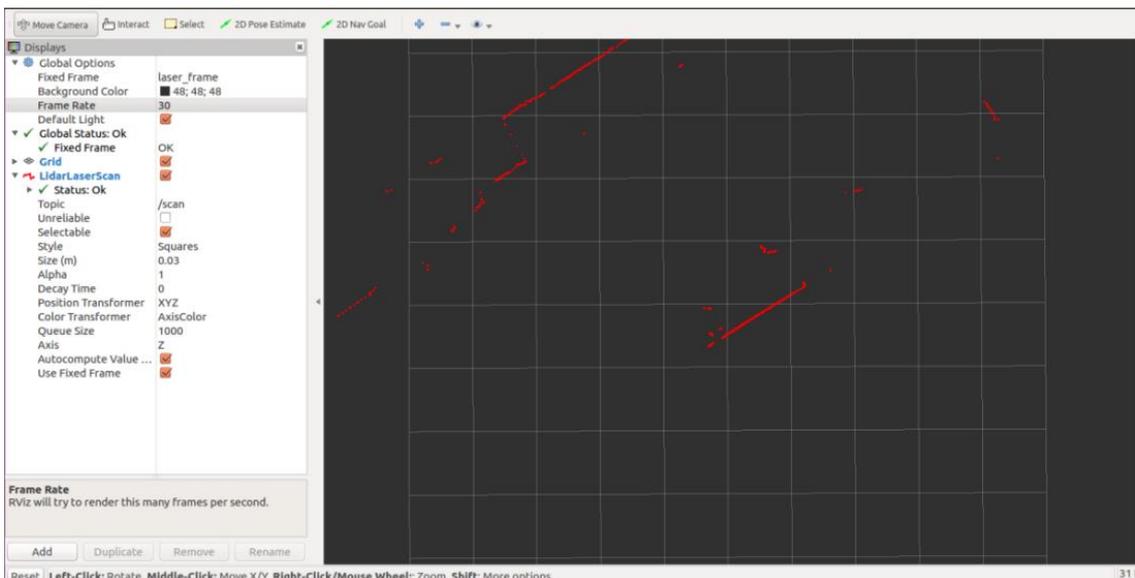

FIG 22 YDLIDAR G2 RVIZ

3.6 Modify Scan Angle

   The scanning data seen by running the launch file is displayed by default with 360- degree data. To modify the display range, you need to modify the configuration parameters in the launch file. The specific operation is as follows:

1) Switch to the directory where the corresponding [launch file] is located, edit the file, and its content is as shown in the figure:

```
$ vim G2.launch
```

```
<launch>
  <node name="ydlidar_lidar_publisher"  pkg="ydlidar_ros_driver"  type="ydlidar_ros_driver_node" output="scree
n" respawn="false" >
    <!-- string property -->
    <param name="port"          type="string" value="/dev/ydlidar"/>
    <param name="frame_id"      type="string" value="laser_frame"/>
    <param name="ignore_array"     type="string" value=""/>

    <!-- int property -->
    <param name="baudrate"          type="int" value="512000"/>
    <!-- 0:TYPE_TOF, 1:TYPE_TRIANGLE, 2:TYPE_TOF_NET -->
    <param name="lidar_type"        type="int" value="0"/>
    <!-- 0:YDLIDAR_TYPE_SERIAL, 1:YDLIDAR_TYPE_TCP -->
    <param name="device_type"        type="int" value="0"/>
    <param name="sample_rate"        type="int" value="20"/>
    <param name="abnormal_check_count"          type="int" value="4"/>

    <!-- bool property -->
    <param name="resolution_fixed"    type="bool"   value="true"/>
    <param name="auto_reconnect"     type="bool"   value="true"/>
    <param name="reversion"       type="bool"   value="true"/>
    <param name="inverted"       type="bool"   value="true"/>
    <param name="isSingleChannel"     type="bool"   value="false"/>
    <param name="intensity"       type="bool"   value="false"/>
    <param name="support_motor_dtr"     type="bool"    value="false"/>
    <param name="invalid_range_is_inf"    type="bool"    value="false"/>

    <!-- float property -->
    <param name="angle_min"     type="double" value="-180" />
    <param name="angle_max"     type="double" value="180" />
    <param name="range_min"     type="double" value="0.01" />
    <param name="range_max"     type="double" value="50.0" />
    <param name="frequency"     type="double" value="10.0"/>
  </node>
  <node pkg="tf" type="static_transform_publisher" name="base_link_to_laser4"
    args="0.0 0.0 0.2 0.0 0.0 0.0 /base_footprint /laser_frame 40" />
</launch>

                                                            1,1            全部
```

FIG 23 G2.LAUNCH FILE

   Note: For more information about the file contents, please refer to

https://github.com/YDLIDAR/ydlidar_ros_driver#configure-ydlidar_ros_driver-internal-parameter

2) The G2 lidar coordinates follow the right-hand rule within ROS, with an angle range of [-180, 180]. "angle_min" is the start angle, and "angle_max" is the endangle. Thespecific scope needs to be modified according to actual use.
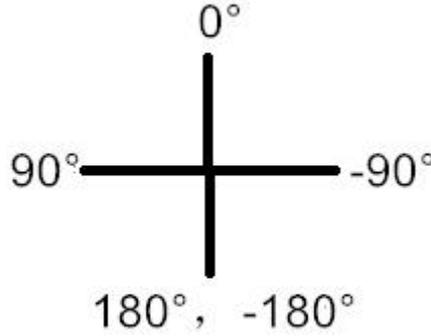

FIG 24 YDLIDAR G2 COORDINATES DEFINITION

4 CAUTION

4.1 Ambient Temperature

When the working environment temperature of G2 is too high or too low, it will affect the accuracy of the distance measuring system. It may also damage the structure of the scanning system and reduce the life of the G2 lidar. Avoid use in high temperature (>50 degrees Celsius) and low temperature (<0 degrees Celsius) conditions.

4.2 Ambient Lighting

The ideal working environment of G2 is indoor, and the indoor environment light (including no light) will not affect the work of G2. However, please avoid using strong light sources (such as high-power lasers) to directly illuminate the G2's vision system.

If you need to use it outdoors, please avoid the G2's vision system directly facing the sun, which may cause permanent damage to the vision system's photosensitive chip, which will invalidate the range finding.

The G2 standard version will cause interference in the distance measurement under the conditions of outdoor strong sunlight reflection, please pay attention to it.

4.3 Power Supply

During the development process, since the drive current of the USB interface of each platform or the USB interface of the computer may be too low to drive the G2, the external power supply of the +5V to the G2 needs to be provided through the USB_PWR interface of the USB interface board. It is not recommended to use mobile phone power bank because the voltage ripple of power bank is too large.

5 REVISE

Date	Version	Content
2017-12-05	1.0	Composing a first draft
2018-01-22	1.1	Add auxiliary power connection method, document description, configuration description, and power supply requirements
2018-04-03	1.2	Adapt to PointCloudViewer2.0 client
2021-08-02	1.3	Adapt LidarViewer client, update SDK and ROS tutorial

5 REVISE