

# YDLIDAR TEA USER MANUAL



# CONTENTS

<b>1</b>	<b>OPERATION UNDER WINDOWS</b>	<b>1</b>
1.1	Device Connection	1
1.2	Network setting	1
1.3	How to Use EaiLidarTest	2
1.3.1	Start Scanning	4
1.3.2	Data Storage	5
1.3.3	Playback and Recording display Mean and Standard Deviation	6
1.3.4	Wave Filtering	6
1.3.5	Network Configuration	7
<b>2</b>	<b>LINUX ROS OPERATION</b>	<b>7</b>
2.1	Device Connect	8
2.2	Compile and Install YDLidar-SDK	8
2.3	ROS Driver Installation	8
2.4	Run ydlidar_ros_driver	9
2.5	RVIZ Viewing Scan Results	9
2.6	Modify Scan Angle	10
<b>3</b>	<b>REVISE</b>	<b>12</b>

# 1 OPERATION UNDER WINDOWS

## 1.1 Device Connection

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

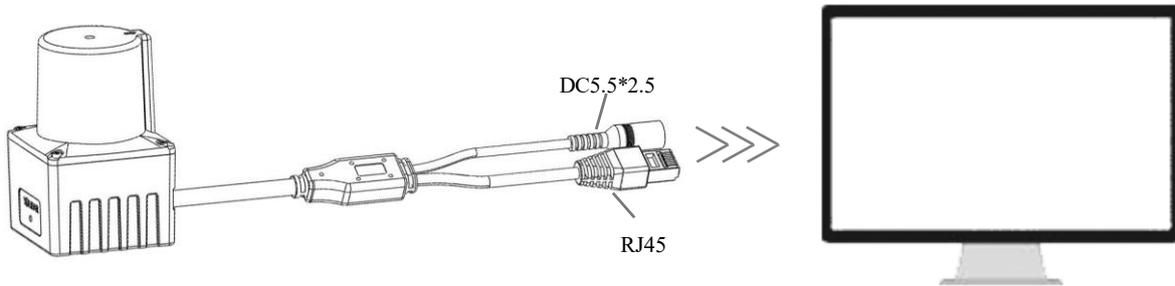


FIG 1 YDLIDAR TEA DEVICE CONNECTION

Connect the RJ45 port on one end of the device cable to the network port on the PC, set the device and PC to be on the same network segment. The DC port is connected to the power socket through the 12V-1A power adapter.

## 1.2 Network setting

TEA Initial network parameters have been set before delivery:

IP address: 192.168.0.11

Subnet mask: 255.255.255.0

When connecting to the lidar for the first time, the network parameters of the PC and lidar must be set in the same network segment and do not conflict with each other. As follows:

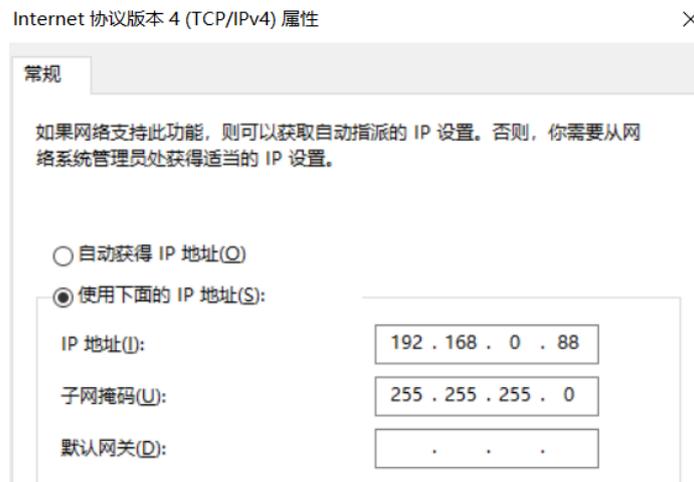


FIG 2 PC NETWORK SETTING

When using the visualization software EaiLidarTest to check the point cloud data, the user need to penetrate the firewall of the operating system to obtain the data. It is recommended to add the

software to the list of allowed software in the firewall. Operation method: “Control Panel” - “System and Security” - “Allow applications to pass through the firewall”, allowing EaiLidarTest to pass through the firewall.

### 1.3 How to Use EaiLidarTest

YDLIDAR provides EaiLidarTest, a point cloud data visualization software for TEA real-time scanning. Users can use this software to visually observe the TEA scanning map: YDLIDAR provides TEA real-time point cloud data and real-time scanning frequency. At the same time, the scanned data can be saved offline to an external file for further analysis.

Run the evaluation software: EaiLidarTest.exe, select the corresponding model number, click “Login”. Meanwhile, users could choose language based on our personal situation (“Help” in the upper left corner).

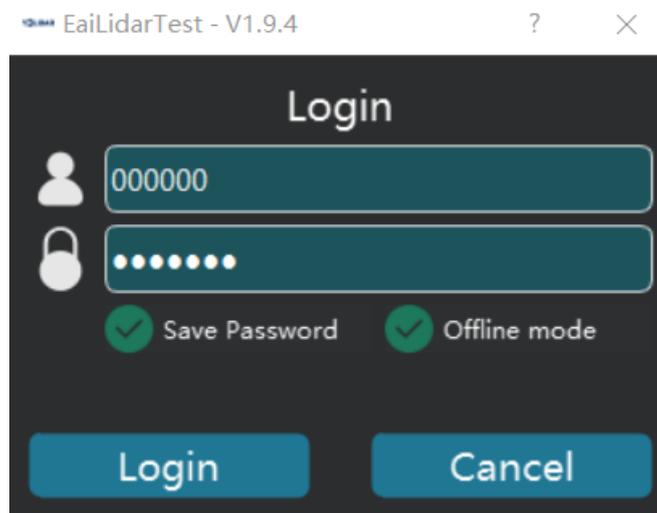


FIG 3 YDLIDAR TEA EVALUATION SOFTWARE

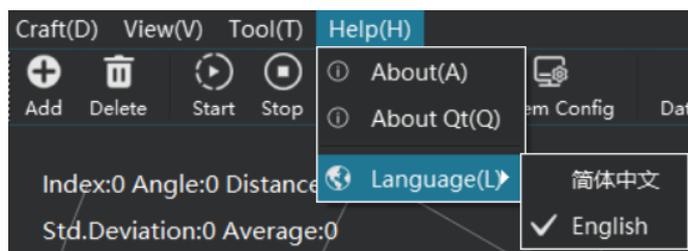


FIG 4 LANGUAGE SELECTION

System Configuration: Click “System Config” to view or modify lidar parameters in the pop-up window.

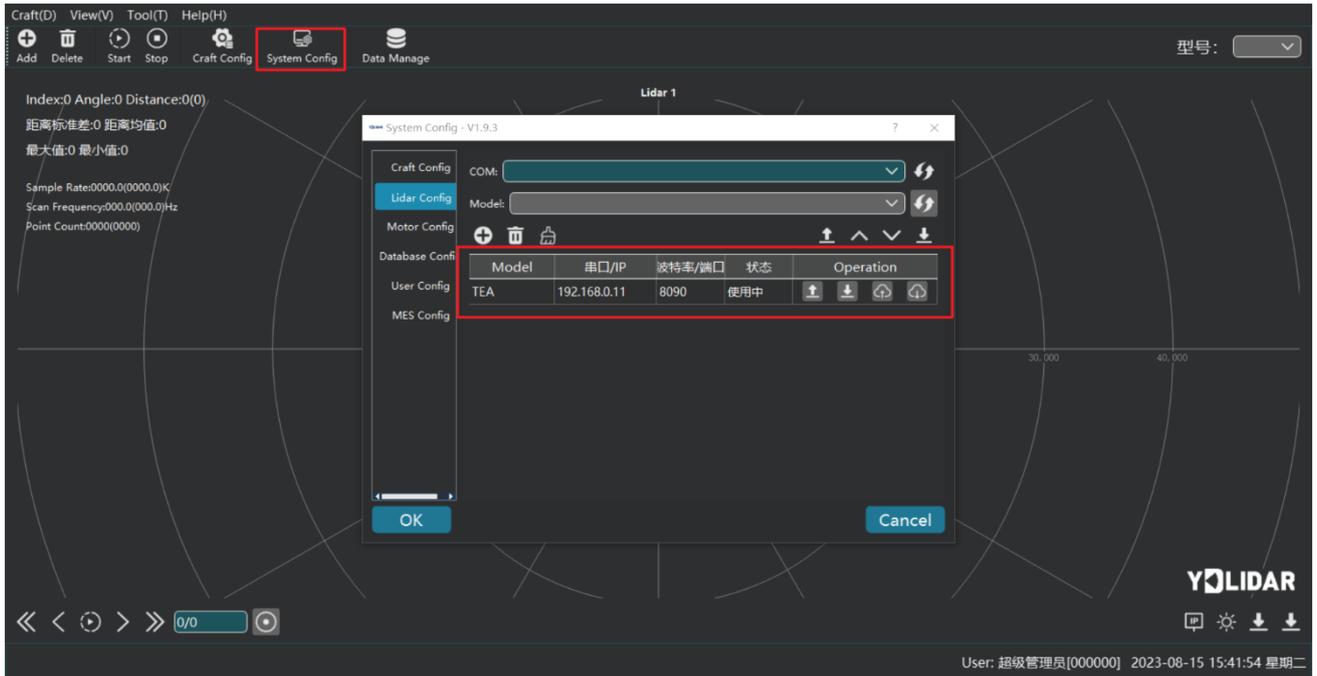


FIG 5 YDLIDAR TEA PARAMETER CONFIGURATION

Lidar parameters configuration: double click on red box area, and refer to the following figure for TEA parameters:

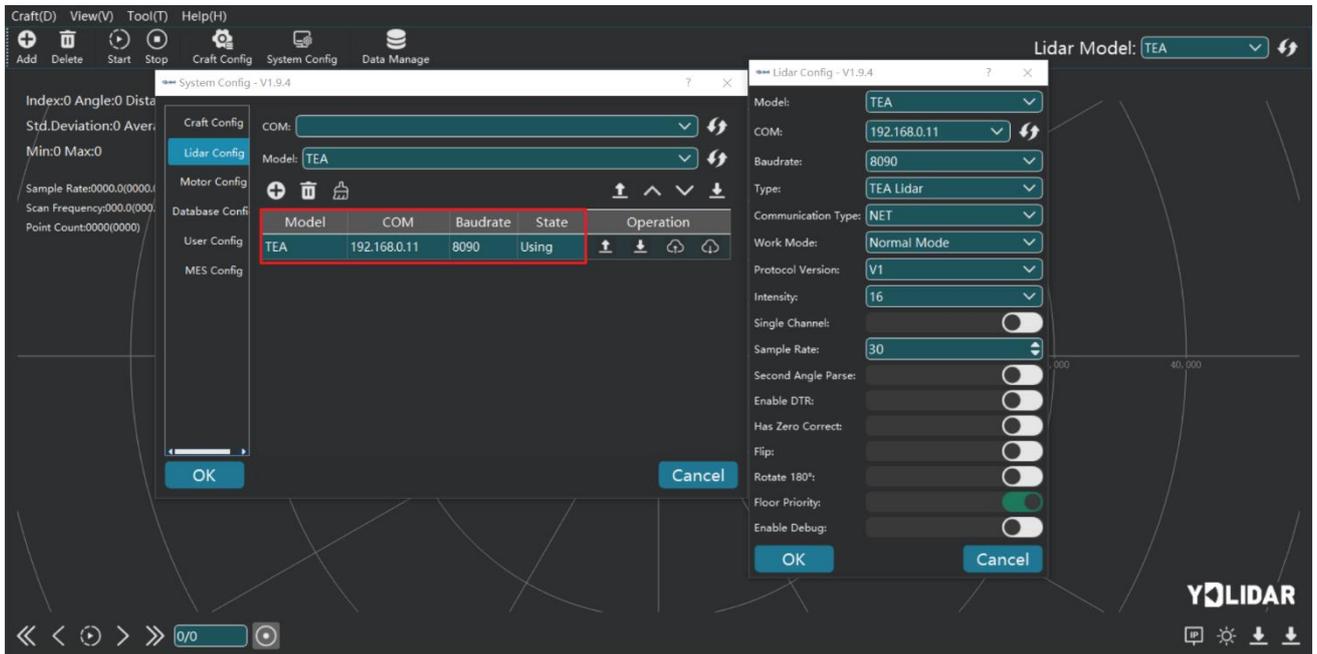


FIG 6 YDLIDAR TEA PARAMETERS CONFIGURATION

After confirmation, the client page is as follows:

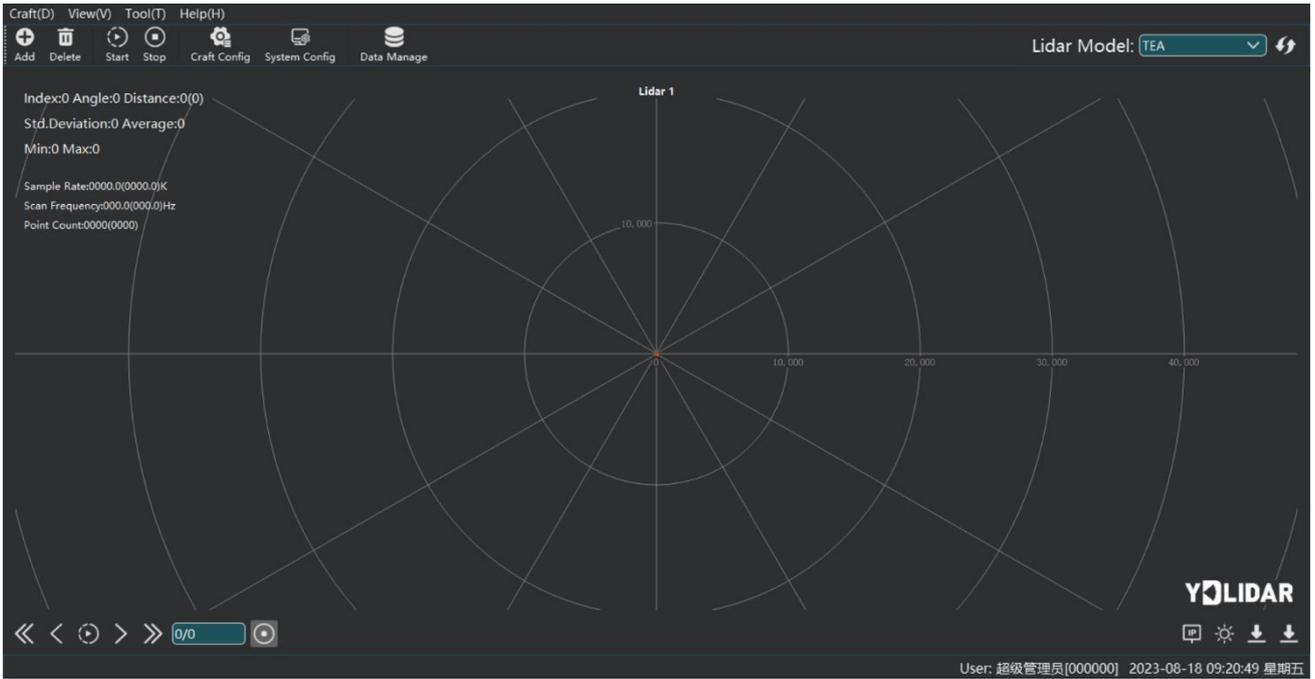


FIG 7 YDLIDAR CLIENT SOFTWARE INTERFACE

### 1.3.1 Start Scanning

Click "Start/Stop" in the Stopped state. Click "Start" button to start scanning and display the environment point cloud. The angle and distance information of the red line position will be displayed in the upper left corner (unit: mm). Click "Stop" button to stop it. as shown below:

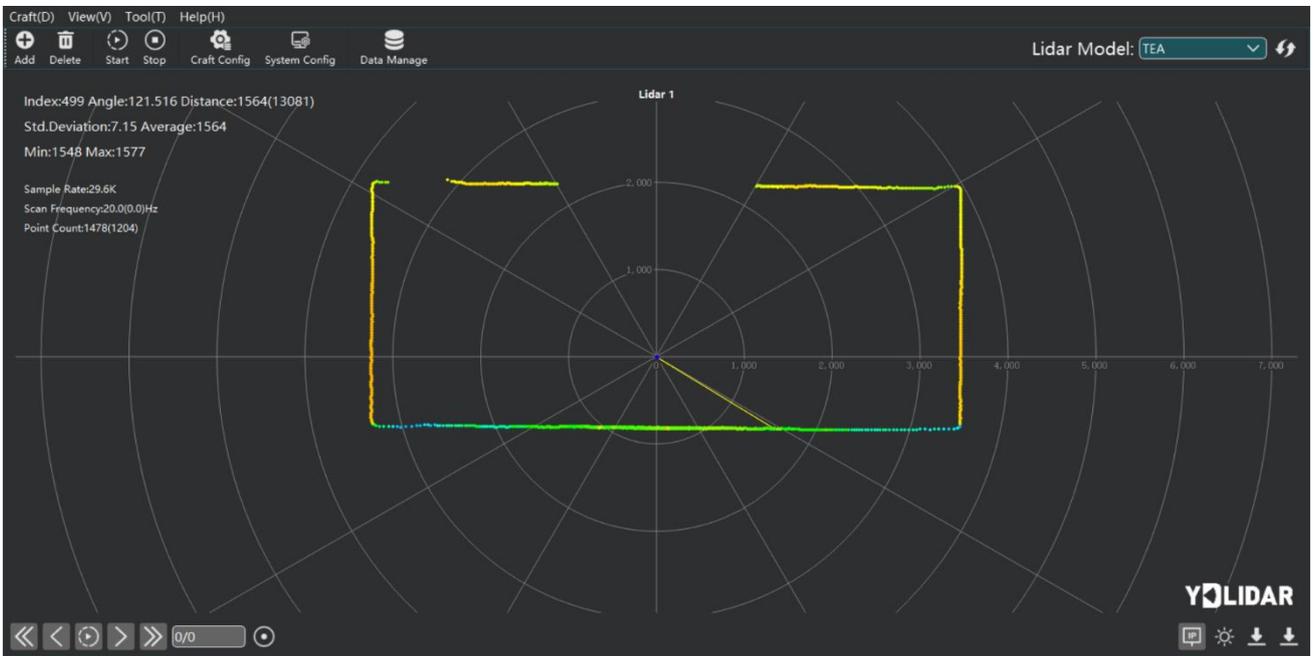


FIG 8 LIDAR SCANNING POINT CLOUD DISPLAY

Select one as needed, move the mouse to the testing position, right-click on the pop-up menu, and select “锁定光标”.

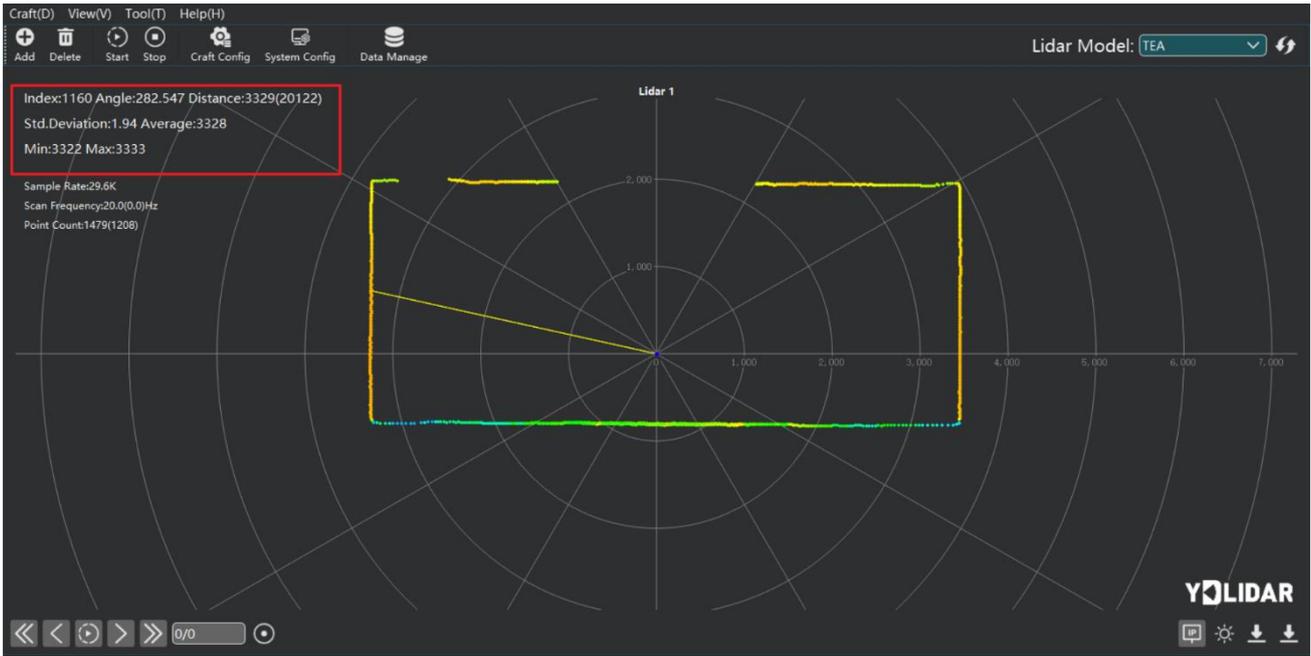


FIG 9 LOCKING MOUSE TRACKING

Move the mouse to the testing position, right-click the pop-up menu, and select “显示强光图”. The main window displays the intensity values of a total of 100 points on the left and right sides of the mouse lock position.

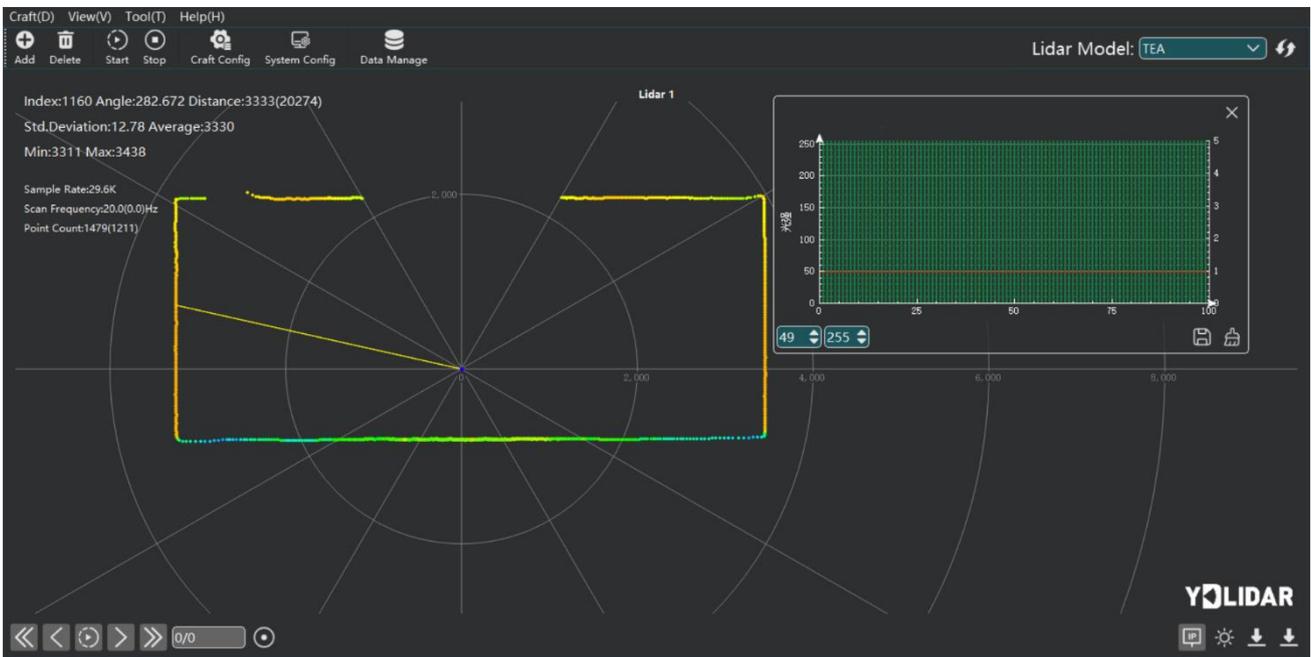


FIG 10 SHOWS THE LIGHT INTENSITY HISTOGRAM

### 1.3.2 Data Storage

During lidar scanning, click  in the Bottom right corner of the main window, select “Export data/Export data (sorted by angle)”, and save point cloud data according to the prompts. Then the system will save the point cloud information scanned in a circle.

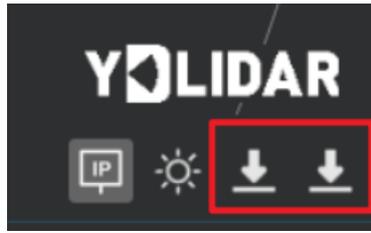


FIG 11 SAVE DATA

### 1.3.3 Playback and Recording display Mean and Standard Deviation

Select “Record/Stop” in the bottom left coner of main menu.

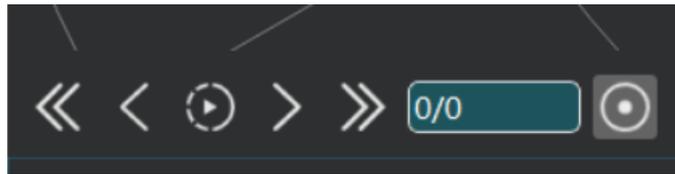


FIG 12 RECORD AND PLAY

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:

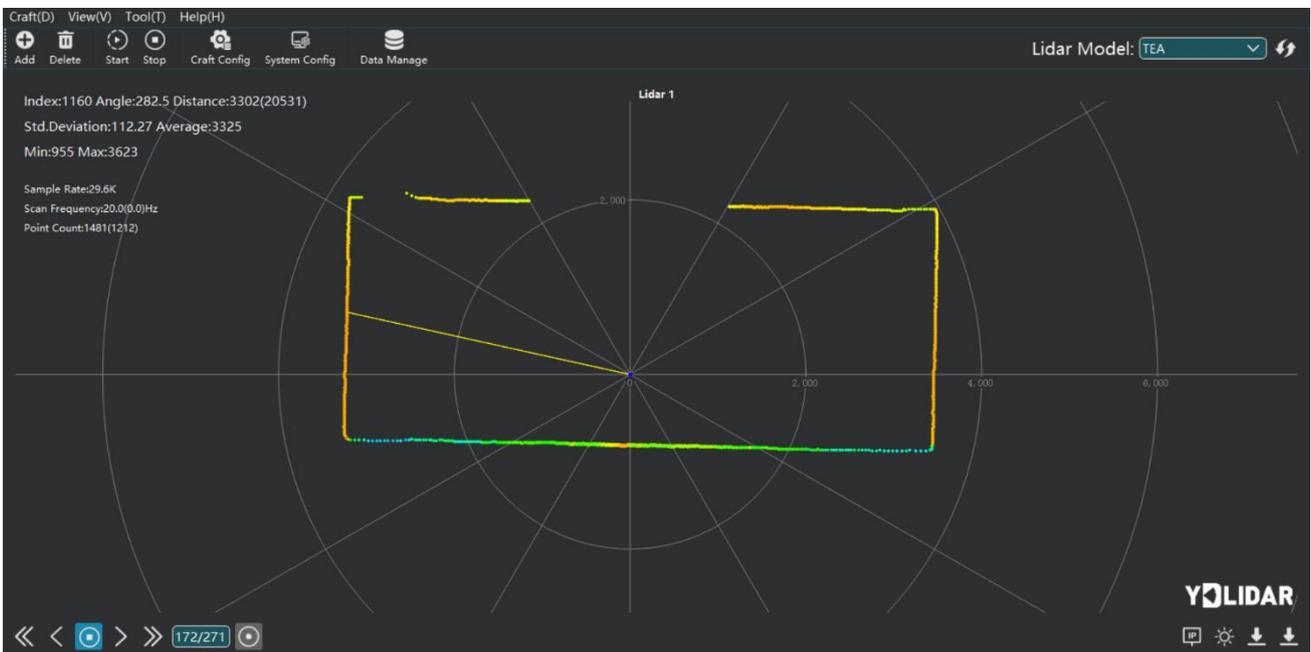


FIG 13 PLAY PROCESS

### 1.3.4 Wave Filtering

Click  on the bottom left corner of the main window to turn on the strong light filter and add a LiDAR data filtering algorithm

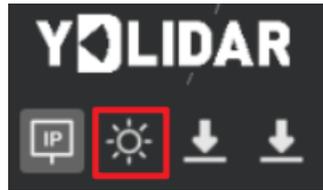


FIG 14 WAVE FILTERING SETTING

### 1.3.5 Network Configuration

Click  on the bottom right corner of the main window to configure the device network parameters, which can be used to set the device IP address, TCP port, and UDP port. After changing any of these configurations, you need to click the “Confirm” button to save the settings. After the device restarts, the settings will take effect.

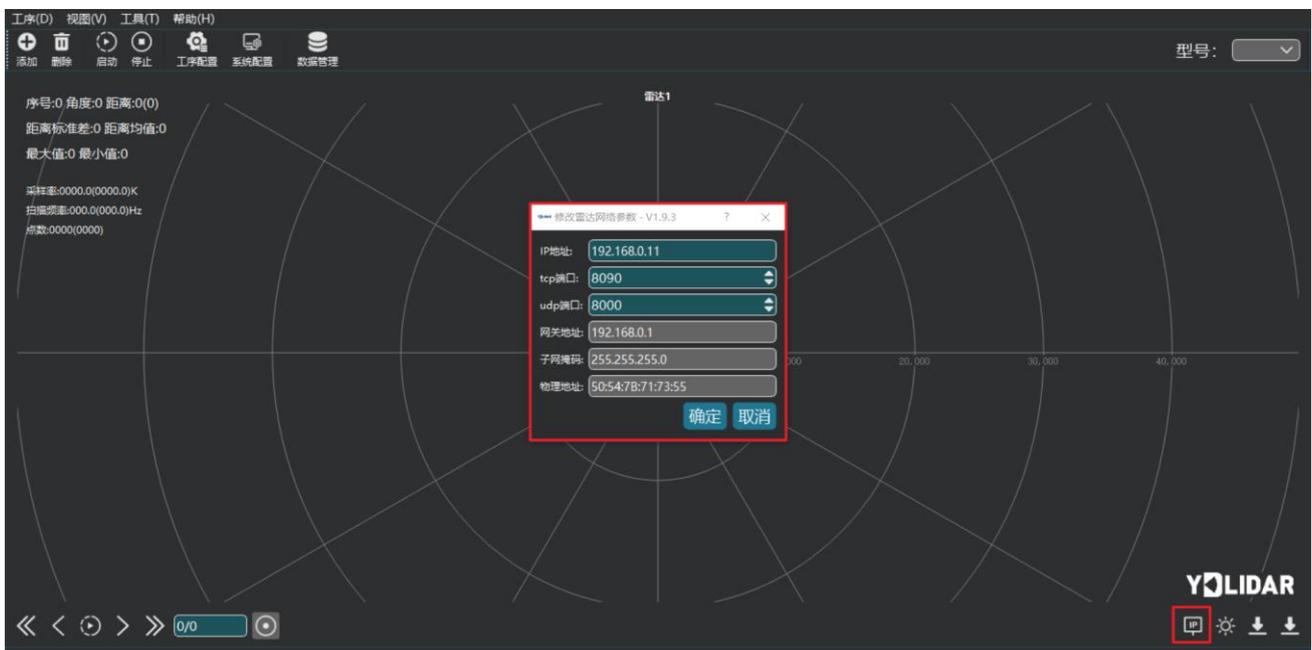


FIG 15 NETWORK CONFIGURATION

## 2 LINUX ROS OPERATION

There are many Linux distributions, and this article only takes Ubuntu 18.04 and Melodic version ROS as examples.

SDK driver address:

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

ROS driver address:

[https://github.com/YDLIDAR/ydlidar\\_ros\\_driver](https://github.com/YDLIDAR/ydlidar_ros_driver)

## 2.1 Device Connect

Under Linux, the interconnection process between TEA and PC is consistent with the operation under Windows, see [Device Connection](#) under Window.

## 2.2 Compile and Install YDLidar-SDK

Ydlidar\_ros\_driver depends on the YDLidar-SDK library. If the user has never installed the YDLidar-SDK library, or it has expired, the user must first install the YDLidar-SDK library. If the user has 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
```

## 2.3 ROS Driver Installation

- 1) Cloning GitHub's ydlidar\_ros\_driver package:

```
$ git clone https://github.com/YDLIDAR/ydlidar\_ros\_driver.git
TIA_ros/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 a permanent workspace environment variable. 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
```

Users 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.

## 2.4 Run ydlidar\_ros\_driver

Run ydlidar\_ros\_driver with startup file, as shown below:

```
$ roslaunch ydlidar ros_driver TEA.launch
```

## 2.5 RVIZ Viewing Scan Results

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

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

Note: Take G4 as an example by default. If you use other lidars, change the lidar.launch file in lidar\_view.launch file to the corresponding \*.launch. (If using TEA lidar, change to TEA.launch)

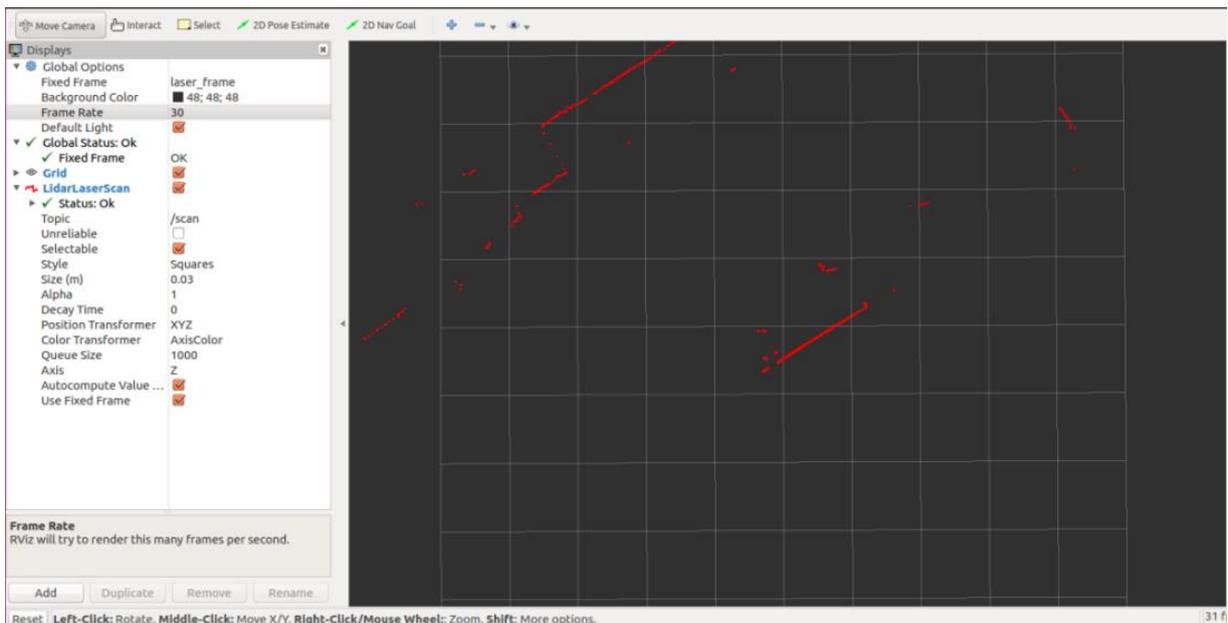


FIG 16 YDLIDAR TEA RVIZ

## 2.6 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) Go to the directory where the corresponding [launch file] is located and edit the file, as shown in the figure:

```
$ vim TEA.launch
```

```
<launch>
  <node name="ydlidar_lidar_publisher" pkg="ydlidar_ros_driver" type="ydlidar_ros_drive
r_node" output="screen" respawn="false" >
    <!-- string property -->
    <param name="port" type="string" value="192.168.0.11"/>
    <param name="frame_id" type="string" value="laser_frame"/>
    <param name="ignore_array" type="string" value=""/>

    <!-- int property -->
    <param name="baudrate" type="int" value="8090"/>
    <!-- 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="1"/>
    <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="true"/>
    <param name="support_motor_dtr" type="bool" value="false"/>
    <param name="invalid_range_is_inf" type="bool" value="false"/>
    <param name="point_cloud_preservative" 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="20.0" />
  </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>
```

FIG 17 TEA .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](https://github.com/YDLIDAR/ydlidar_ros_driver#configure-ydlidar-ros-driver-internal-parameter)

- 2) The TEA 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. The specific scope needs to be modified according to actual use.

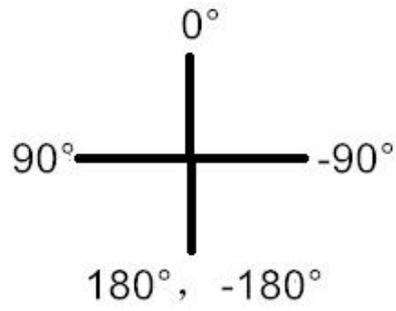


FIG 18 YDLIDAR TG COORDINATES DEFINITION

### 3 Revise

Date	Version	Content
2023-08-10	1.0	The 1st release