CE30-C Solid State Array LiDAR

Operation Manual

Benewake (Beijing) Co., Ltd

Table of Contents

1. Introduction

CE30-C is an infrared LiDAR developed on the basis of ToF principle. Compared with scanner LiDAR, CE30-C does not have any rotating components in the interior, which could ensure the reliability of long-time work and wider vertical angle.

1. Shell

4. Power/CANBUS port (M8 aerial socket)

2. Reception panel (working area, cannot be covered)

5. Ethernet port (M8 aerial socket)

3. Emitting panel (working area, cannot be covered)
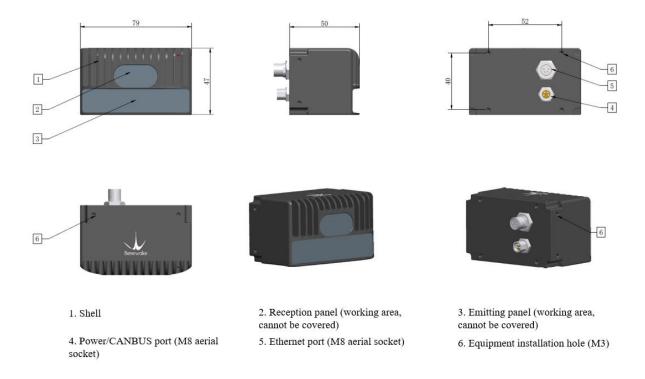
6. Equipment installation hole (M3)

Figure 1 Outline Dimension Drawing of DE-LIADAR CE30

2. Accuracy Test

Figure 2 Accuracy test environment and devices

The test is made under the condition of no ambient light. 20 minutes warm-up time is required before test to make CE30-C reach its stable operating temperature. The CE30-C is mounted on an automatic test car that moves along a guide rail and has a displacement accuracy of 1 cm. CE30-C detects the white board of 90% reflectivity, collects 50 frames of data every 10cm. The center 24*24 pixels area's data is selected for statistical analysis.

CE30-C's accuracy is defined as the difference between the average value of a 24*24 pixels area in the center of the field of view and the true distance value.
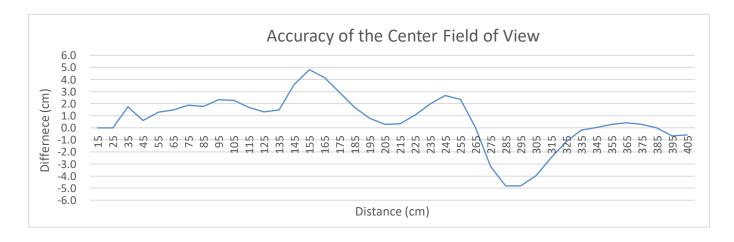

*Figure 3 Accuracy of the center 24*24 pixels area*

CE30-C's repeatability is defined as the standard deviation of the average value of a 24*24 pixels area in the center of
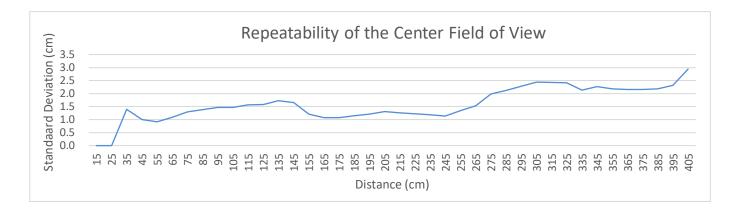
the field of view.

Figure 4 Repeatability if the center 24*24 pixels area

3. Operation of Windows Display Program

This display program is used for processing and displaying of output data from CE30-C LiDAR in Windows operating system. At first, you should get the display program *CE30_Display.exe* from official website or technical support:

Figure 5 Windows Display Program
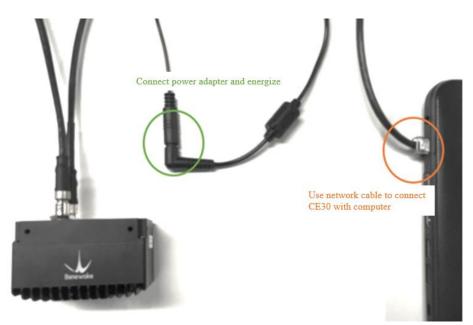
3.1. Connection

Figure 6 Illustration for Connection of Components

Connect power supply cable and Ethernet cable and set IP address and subnet mask under the directory *Control*

www.benewake.com

Panel\Network and Internet\Network and Internet Connections as shown in Figure 7 below.

Figure 7 Network Setting

To check whether connect is set correctly, start a Command Prompt via Run dialog window and execute command:

ping 192.168.1.80

If the information as shown in Figure 8 appears, this indicates LiDAR is connected correctly and working.

Figure 8 Check of Data Connection

Notes: 192.168.1.80 is the default IP address of CE30-C LiDAR. If users changed IP address in use, the steps above should also have corresponding changes. For example, we changed LiDAR's IP address to 192.168.2.1, so we should change the IP address of the device that connected to LiDAR to 192.168.2.xxx in Figure 7.

3.2. Display Program Windows

Double click *CE30_Display.exe* to run display program. A message will show in console that ask for LiDAR's IP address, shown in Figure 9. The default IP address is 192.168.1.80. After entering the IP address, three windows as shown in Figure 10 will pop up: Message Prompt Window, Undistorted Depth Image Window and Point Cloud Window.

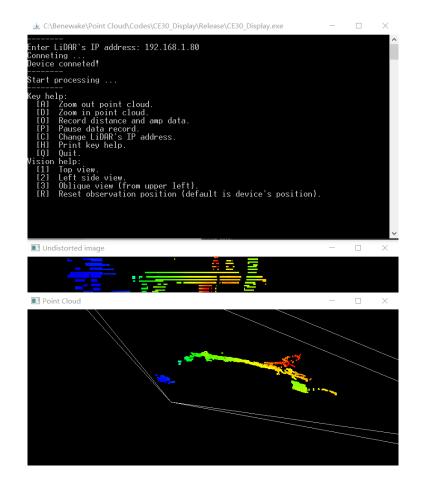*Figure 9 Message ask for LiDAR's IP address*

Figure 10 Windows of Upper Computer, From Top to Bottom: Message Prompt Window, Depth Image Window and Point Cloud Window

3.2.1. Message Prompt Window

This window is used to display LiDAR connection information, Key Help for shortcut keys and other running information.

www.benewake.com

3.2.2. Depth Image Window

Display program outputs undistorted depth image (Undistorted Image). The resolution is 660×24 and the field of view is $132 \times 9°$.

Click depth image in both windows, the depth value (in centimeter) of the pixel corresponding to current cursor position will be displayed in the message prompt window, as shown in Figure 11.

Figure 11 Display of Depth Value at Pixel Position

3.2.3. Point Cloud Window

Point Cloud Window shows the point cloud image obtained based on the depth image and projection. In this window, hold left mouse button and drag to switch viewing direction of point cloud image; use mouse wheel to zoom point cloud window; use right mouse button to reset observation position of the point cloud window to default angle.
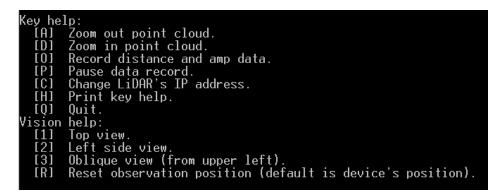
3.2.4. Key Help

Figure 12 Key Help

When launching the program, Key Help for shortcut keys will be printed in the Message Prompt Window, as shown in Figure 12. Controls by shortcut keys consist of two parts:

1. General control: control and enabling/disabling of general functions of the program.
   a) Shortcut A   To zoom out point cloud image
   b) Shortcut D   To zoom in point cloud image
   c) Shortcut O   To record original & undistorted depth image data and amplitude data. Save the data under the directory of the program
   d) Shortcut P   To pause data recording
   e) Shortcut C   To change LiDAR's IP address
   f) Shortcut H   To reprint this Key Help messages in message prompt window.
   g) Shortcut Q   To deactivate LiDAR and disconnect the program with LiDAR
2. Vision Help:
   a) Shortcut 1   To adjust point cloud image to top view
   b) Shortcut 2   To adjust point cloud image to left side view
   c) Shortcut 3   To adjust point cloud image to oblique view from upper left
   d) Shortcut X   To reset observation position to default (default is the position of LiDAR)

www.benewake.com

Notes: shortcut keys will come into effect only in Original/Undistorted Image Window or Point Cloud Window. When in Message Prompt Window, they will function to input into program.

3.3. Deactivation

Procedures for normal deactivation of LiDAR and display program are:

1. Press "Shortcut Q" in Original/Undistorted Image or Point Cloud Window. The prompt message for deactivation of LiDAR and disconnection will appear on Message Prompt Window.
2. Press "Enter" key in Message Prompt Window to quit program.

Noted that LiDAR will remain operating and connected with PC terminal if the display program is directly turned off without using "Shortcut Q". If the display program is rebooted in this case, it will not operate normally as LiDAR will not respond to its request for connection and activation. In that case, the LiDAR needs to be directly power off, followed by reconnection and reboot of the display program.

4. Indicator

Figure 13 Position of Indicator

1. Blue light: ready state, ready for connection and running
2. Blue flash: running state
3. Red flash: missing of relevant running files.
4. Red light: fatal error (signal abnormal, interface communication abnormal, I^2C abnormal, etc.)
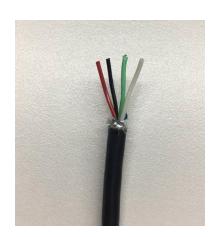
5. Line Sequence

Figure 14 Line Sequence of Power Supply Port: Red - Positive Pole of Power Supply, Black - Negative Pole of Power Supply

Attention!

1. Current of power adapter shall be above 2A.
2. During energization of LiDAR, there is merely a slim chance of prolonged starting time. If LiDAR is not started after 2 minutes, it is recommended to deenergize and reboot it.
3. Please make sure to disconnect power supply before other connections.
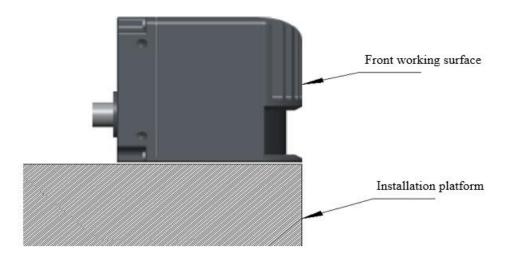
6. Installation Schematics

Figure 15 Recommendations on LiDAR Installation Location. The front working surface of LiDAR should exceed or at least be parallel with the installation platform; otherwise, there may be certain interference, further reducing the data accuracy

www.benewake.com

7. Introduction of SDK

7.1. Descriptions of SDK

SDK is mainly used to develop applications of this LiDAR on Linux or Windows system by using C/C++, of which source codes can be referenced for migration & development of applications on other platforms.

1) SDK consists of following file or folders:

   - samples: examples of using SDK, including makefile (Linux) or Visual Studio solution files (Windows);
   - sources: source codes of SDK;
   - README: operation instructions as well as other relevant information and instructions of SDK.

2) Globel variables in SDK:

   - unsigned short **gRawDistMatrix[]**: original distance data obtained from LiDAR, arranged in form of one-dimensional array;
   - unsigned short **gRawAmpMatrix[]**: original quality data obtained from LiDAR, arranged in form of one-dimensional array;
   - float **gCameraMatrix[9]**: internal parameters matrix of LiDAR, arranged in form of one-dimensional array and in the sequence {f/dx, skew, u0, 0, f/dy, v0, 0, 0, 1};
   - float **gDistCoffs[4]**: distortion parameters of LiDAR, arranged in the sequence of {k1, k2, k3, k4}.

3) Methods contained in SDK:

   - bool initDevice(int &_sd, char *_ip)

     Parameter:

     _sd   socket handle (the type will be SOCKET in Windows SDK)

     _ip   IP address of LiDAR

     In this method, *gCameraMatrix [9]*, *gDistCoffs[4]*, *mapX[]*, *mapY[]*, socket and TCP/IP connection are initialized. LiDAR is literally put into service by invoking such method.

   - bool getDistanceData(int _sd, unsigned short *_dst,   unsigned short *_amp)

     Parameter:

     _sd   socket handle (the type will be SOCKET in Windows SDK)

     _dst   data of undistorted depth image, the size should be 24[height] * 660[width]

     _amp   data of quality, size is equal to _dst, the size should be 24[height] * 660[width]

     In this method, original data from LiDAR is accessed and saved in *gRawDistMatrix[]*. The undistorted depth image is output from *_dist*. A frame of depth image (size 24*660) is output every time this method is called.

   - bool closeDevice(int &_sd)

     Parameter:

        _sd   socket handle (the type will be SOCKET in Windows SDK)

In this method, the LiDAR is deactivated and TCP/IP is disconnected.

- bool getPointCloud(unsigned short *_dist, float *_coordX, float *_coordY, float *_coordZ)

Parameter:

        _dist   distance data, get from *getDistanceData()*

        _coordX x-axis coordinates of point cloud, from left to right

        _coordY y-axis coordinates of point cloud, from bottom to top

        _coordZ z-axis coordinates of point cloud, from near to far

This method transform depth data from *getDistanceData()* to point cloud data. All params should have the same size of 24[height] * 660[width], and the same sequence number params construct a space point (_coordX[n], _coordY[n], _coordZ[n]). The data sequence is from field of view's left to right and then from top to bottom.

- bool changeIPAddress(int _sd, char *_newIP)

Parameter:

        _sd   socket handle (the type will be SOCKET in Windows SDK)
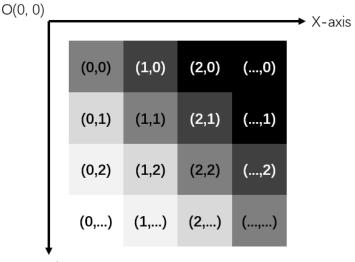
        _newIP   new IP address of LiDAR

Change LiDAR's IP address to _newIP. The LiDAR will reboot if succeed.

For details, please refer to sample program and codes in examples

7.2. Output Data

Figure 16 Illustration of Data Arrangement

A frame of depth data is outputted from the SDK with the resolution that have been set. The data is arranged in form of one-dimensional array. The sequence is from left to right and top to bottom, with the pixel in the upper left corner of the image as the origin. Output along X axis, then Y axis, the following output sequence will be generated as shown in the figure above:

(X,Y) : (0,0) (1,0) (2,0) (…,0) … (0,1) (1,1) (2,1) (…,1) … (0,2) (1,2) (2,2) (…,2) … (0,…) (1,…) (2,…) (…,…) …

For the distance from detected object to the plane of camera, it is expressed in cm by each pixel through the data format of 16-bit unsigned short int.
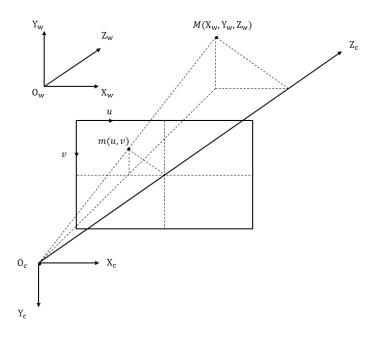

Figure 17 Projection Relation Diagram

www.benewake.com

Generally, the origin point O_c of camera coordinate system is considered to coincide with origin point O_w of world coordinate system. As such, a point m (u,v) in depth image can be transferred into 3D space point M (X_w,Y_w,Z_w) by following equations:

$$\begin{cases} X_w = Z_c \cdot (u - u_0) \cdot d_x/f \\ Y_w = Z_c \cdot (v_0 - v) \cdot d_y/f \\ Z_w = Z_c \end{cases}$$

Where

Z_c output value of each pixel

u_0 the x-coordiante of depth image's center

v_0 the y-coordiante of depth image's center

d_x size of pixel along x-axis

d_y size of pixel along y-axis

f focal distance of lens, d_x/f = 1 / gCameraMatrix[0], d_y/f = 1 / gCameraMatrix[4]

8. Cooler and Reference Design

In normal operation, the CE30 LiDAR has an average heat power consumption of around 5W. As shown below is the reference design of heat dissipation component of CE30 LiDAR. Users have the option to design heat dissipation component depending upon actual installation requirements and application scenario.
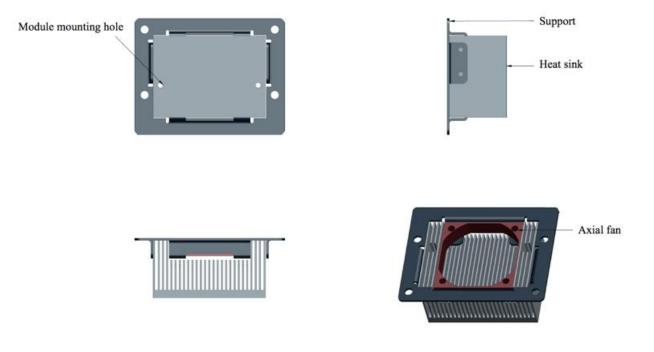

Figure 18 Optional cooler Components

9. Influence Factors of Measurement

9.1. Multi Optical Path

Based on ToF LiDAR principle, if there are multiple echo regions as shown in the figure below at the working height of the radar, the multi-path phenomenon will be triggered: the LiDAR receives the light returned by the path 1 and the path 2 at the same time, which may result in a larger measurement value.
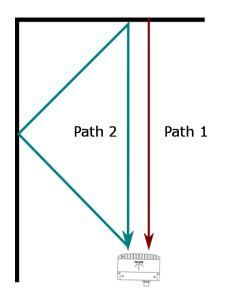

Figure 19 Multi optical path phenomenon

9.2. Stray Light

As shown below, when solid-state ToF LiDAR is working, in addition to the light reflected by the object 1, the light scattered by object 2 and object 3 that close to the LiDAR will enter the lens. Such stray light can lead to a deviation of the object 1's ranging.
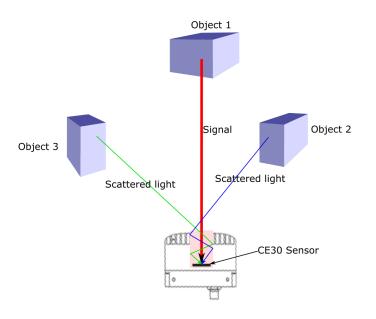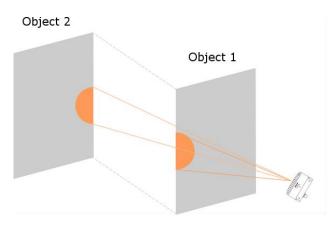
www.benewake.com

Figure 20 Illustration of stray light

9.3. Multi Distance Objects

Figure 21 Multi distance objects

The light radiated by the LiDAR is reflected by the object onto the sensor of the LiDAR. If some pixels receive signals from both front and rear obstacles at the same time, the output distance of this pixel may be the value between the two obstacles. The degree of deviation is related to the distance between the two obstacles and the material.

www.benewake.com