LIPSedge^{TM} L210u / L215u

3D Camera

SDK V1.0.0.0

User Guide

Although every effort has been made to ensure the accuracy of this manual, errors and inconsistencies may remain. The manufacturer assumes no liability resulting from errors or omission in this manual, even if damages arise from the use of the information. All contents are subject to constant revision to improve its reliability and may be changed without prior notice.

December 2022

Contents

LIPSedge^{TM} L210u / L215u 3D Camera

SDK User's Guide

Welcome to **LIPSedge^{TM} L210u/L215u 3D Camera SDK User's Guide**! This document provides a step-by-step guide to use your LIPSedge^{TM} L210u/L215u camera and set up its development platform on your PC / laptop.

Regulatory Compliance

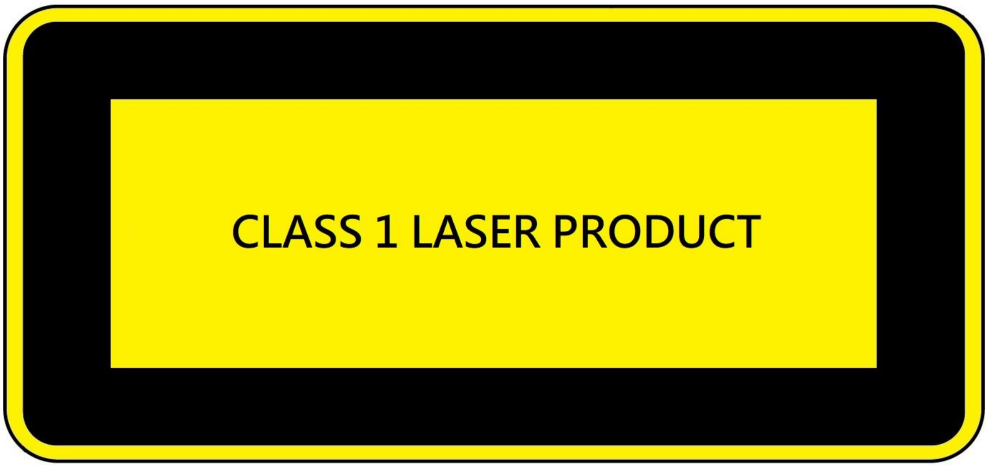
FCC Compliance

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Caution: Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

FCC Label Notice



CLASS 1 LASER PRODUCT

This device complies with part 15 of the FCC Rules. Operation is subject to the following conditions:

(1) This device may not cause harmful interference, and
(2) this device must accept any interference received, including interference that may cause undesired operation.

CE Compliance

This is a Class B product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

RoHS Compliance

All lead-free products offered by the company comply with the requirements of the European law on the Restriction of Hazardous Substances (RoHS) directive, which means our manufacture processes and products are strictly “lead-free” and without the hazardous substances cited in the directive.

Notes for Programmers

We recommend the following system setup for LIPSedge^{TM} L210u / L215u:

Recommended Hardware

- USB 2.0 / 3.2 port

Recommended Systems

- Windows 10 64-bit System
- Ubuntu 18.04 / 20.04 LTS System

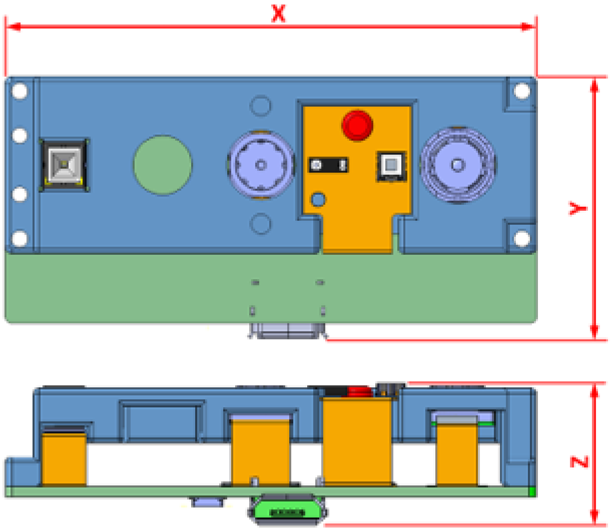
1. Overview

The LIPSedge^{TM} L210u / L215u 3D structured light camera come with high depth precision (≤ +/-0.3% @100cm), full-HD RGB resolution, and 940nm wavelength VCSEL light source, suitable for both indoor and outdoor environment. The camera supports high precision and short-range 3D sensing applications such as 3D scan, 3D measurement, visual defect inspection et cetera.

LIPSedge^{TM} L210 (Module only)

LIPSedge^{TM} L215u (with housing)

1.1 Dimension



Dimension					
	Min	Nom	Max	Tolerance	Unit
X	53.8	54	54.2	±0.2	mm
Y	26.3	26.6	26.9	±0.3	
Z	13.9	14.3	14.7	±0.4	

1.2 Packing List (L215u only)

The packing list serves as a reference for users to check the package content. If anything was missing, contact info@lips-hci.com. For technical specification, refer to LIPSdege^{TM} L215u datasheet.

- LIPSedge^{TM} L215u camera

- USB 2.0 Y-Cable

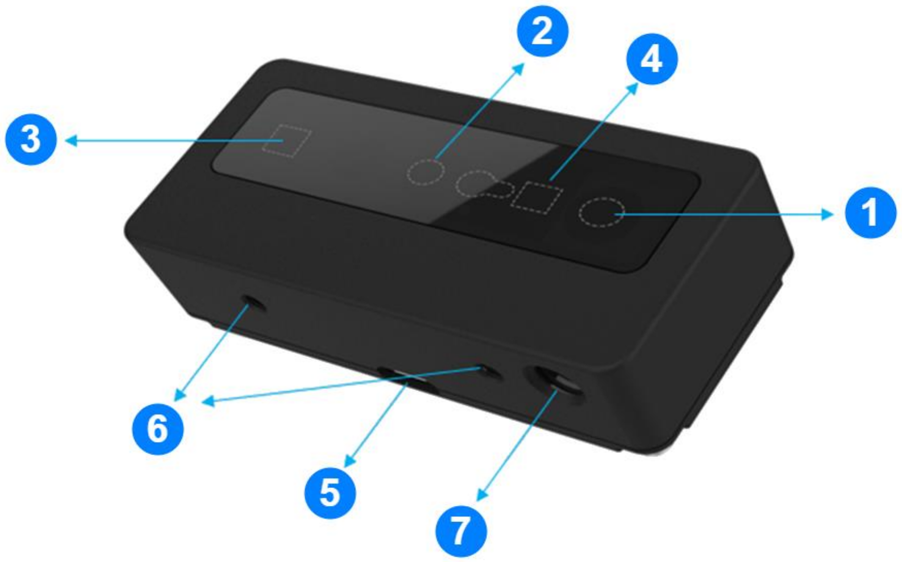
- LIPSedge^{TM} L series User's Guide

1.3 Camera Accessories

The following hardware accessories facilitates the installation process but are **NOT** included in the package. You need to prepare these accessories on your own.

- Camera tripod
- PC / Laptop (For specification. Refer to *Notes for Programmers*)

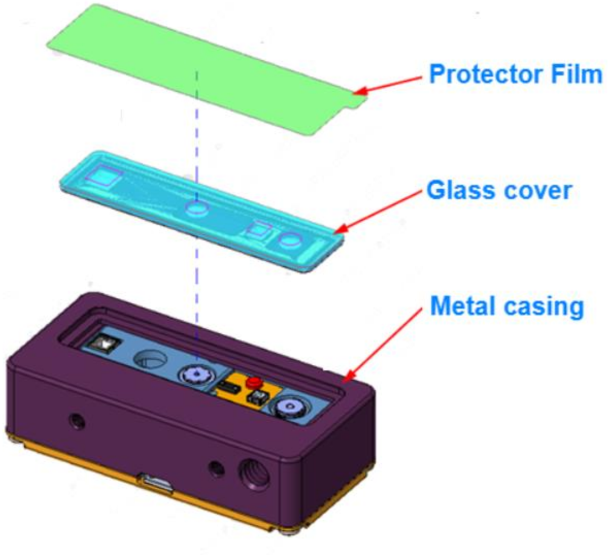
1.4 Hardware Features



No.	Key Component	Description
1.	**NIR sensor**	Receives the IR image
2.	**RGB sensor**	Receives the RGB image
3.	**NIR illuminator**	Projects NIR structured dots light
4.	**Flood Projector**	Emits uniform, flat IR light to enhance lightness
5.	**Micro USB Port**	Provides power input and data output to the host PC / laptop
6.	**M3 * 4 mm * 2**	Major mount holes
7.	**Tripod mount**	Mounts the camera onto the tripod

2. Hardware Installation

WARNING: A thin layer of film is applied to the camera lens for protective purposes when shipped. Remove the film **ONLY** when you are ready to use the camera. Keeping the thin film while streaming may cause the image quality to decrease.

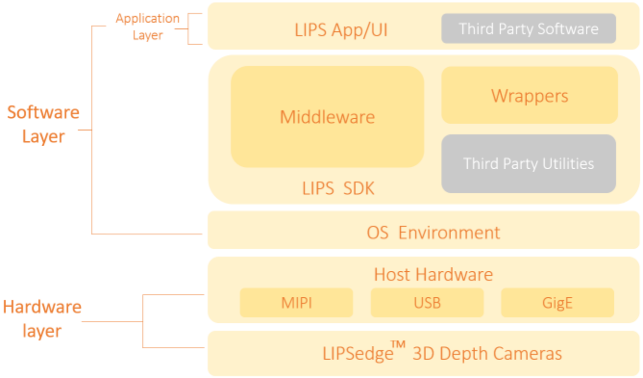


1. Plug the **Micro B** end of the USB-Y cable to the LIPS Camera.

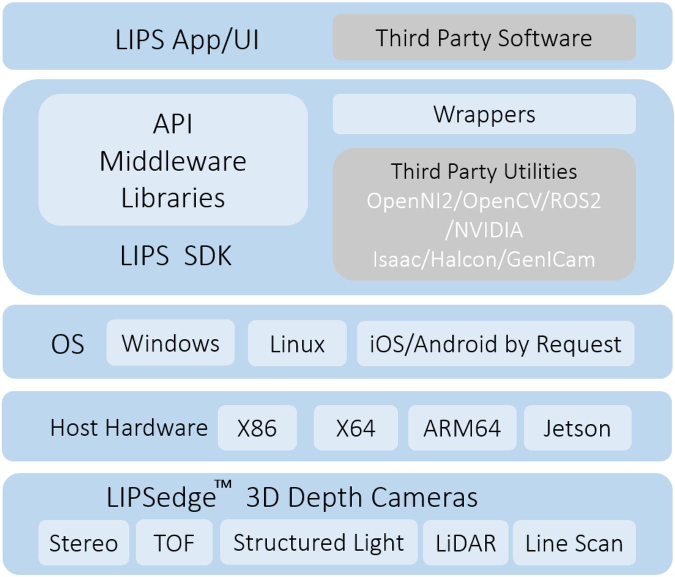
2. Plug both **Type-A end** of the USB-Y cable to your PC / laptop.

3. System Architecture

LIPS 3D camera / SDK offers a system for developing depth-sensing applications. As the LIPS system architecture illustrates, the system is comprised of the hardware layer and the software layer. The hardware layer oversees data capture, transfer, and processes. In the software layer, the captured data is fetched by the LIPS SDK (Software Development Kit) on the OS environment. Depending on the project complexity, wrappers and third-party utilities may be engaged before the data is eventually presented in the application layer for business applications.



The core of the system, the LIPS SDK, is comparable to a toolbox full of software modules comprised of middleware, libraries, wrappers and API, and miscellaneous programming languages / platforms for application development. With extensive wrapper support, LIPS SDK enables developers to access bottom layer data with APIs, thus eliminating the hassle of changing third-party functions. The result is a highly effective project scoping, monitoring and execution workflow compatible with the fast-pacing AIoT market and machine vision demands.



Windows

4. SDK Installation

LIPS Corp. releases the versions of LIPS SDK for specific systems on LIPS Corp. official websites. Download and install the LIPS SDK according to your host platform.

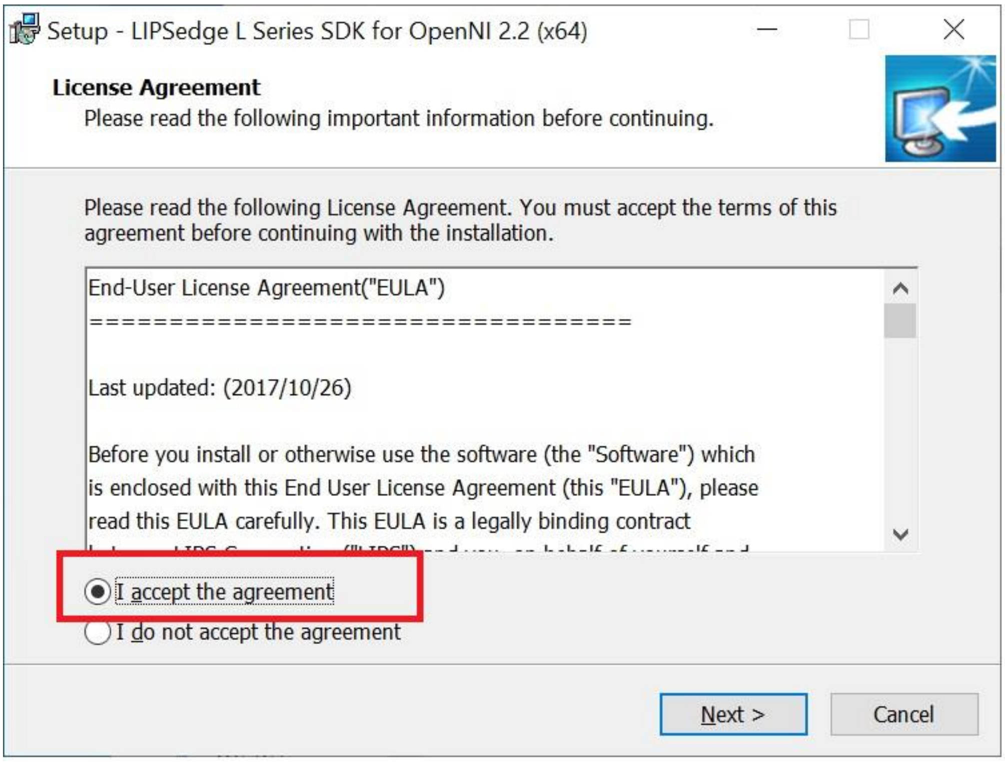
1. Go to our website: https://www.lips-hci.com/developer-documentation.

2. On the left menu, select the model you needed. A dropdown link appears.

3. Select **SDK Release > SDK > SDK**, download the installation file according to your operating system.

3. Unzip the downloaded file and double click the **Setup** program.

4. Accept the **License Agreement**, click **Next** and click **Install**. The installation process starts.



Setup - LIPSedge L Series SDK for OpenNI 2.2 (x64)

License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

End-User License Agreement("EULA")

==================================

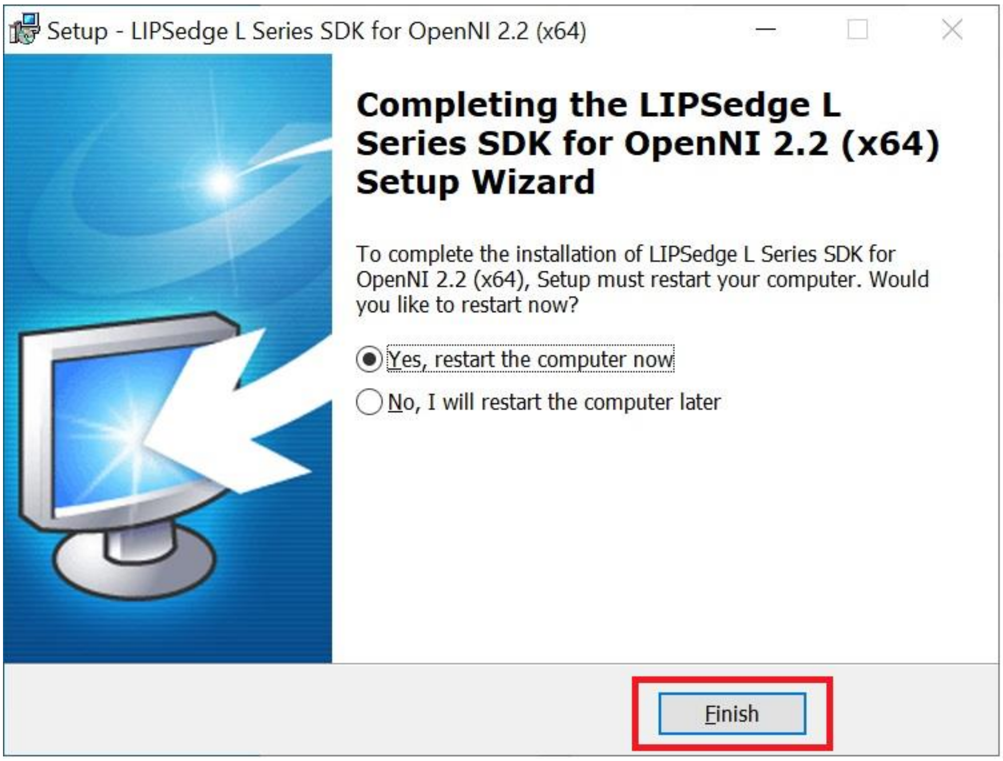
Last updated: (2017/10/26)

Before you install or otherwise use the software (the "Software") which is enclosed with this End User License Agreement (this "EULA"), please read this EULA carefully. This EULA is a legally binding contract

● I accept the agreement

○ I do not accept the agreement

5. Click **Finish** and restart your PC / Laptop.



Setup - LIPSedge L Series SDK for OpenNI 2.2 (x64)

Completing the LIPSedge L Series SDK for OpenNI 2.2 (x64) Setup Wizard

To complete the installation of LIPSedge L Series SDK for OpenNI 2.2 (x64), Setup must restart your computer. Would you like to restart now?

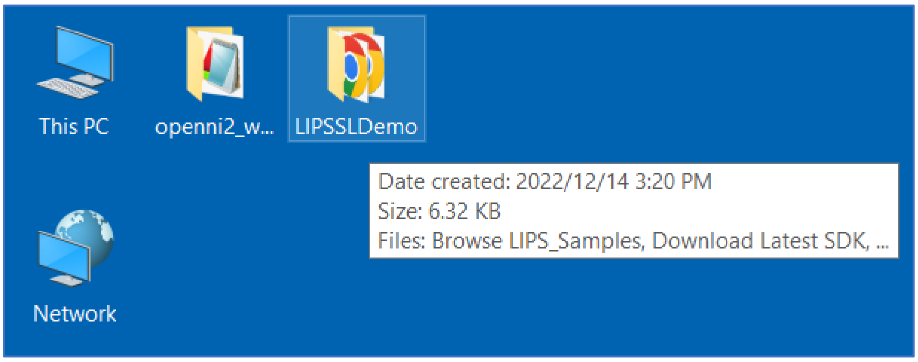
● Yes, restart the computer now

○ No, I will restart the computer later

5. NiViewer

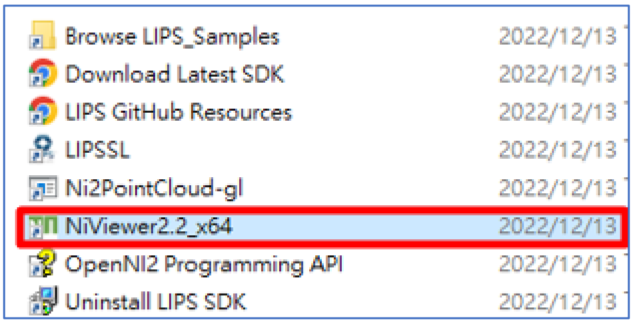
After the LIPS SDK is installed, a new **LIPSSLDemo** folder, in which the **NiViewer** is contained, can be found on your desktop. With NiViewer, you can access the camera image and status. Note that your NiViewer is stored in different locations on your PC / laptop depending on your operating system.

1. On your desktop, click the **LIPSSLDemo** folder.



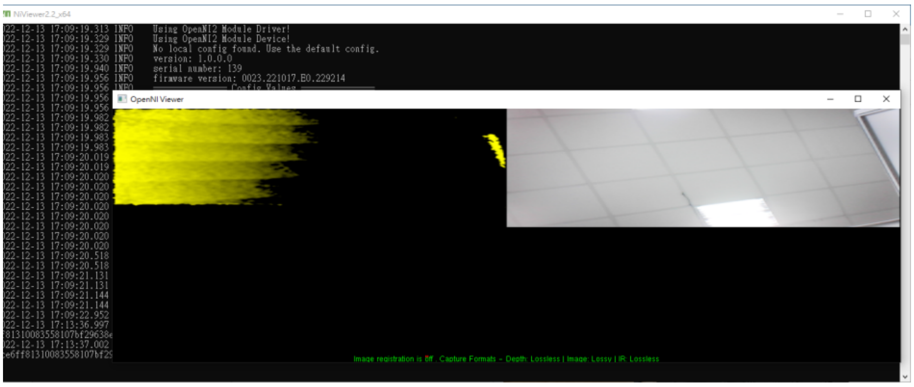
Date created: 2022/12/14 3:20 PM
Size: 6.32 KB
Files: Browse LIPS_Samples, Download Latest SDK, ...

2. Open **NiViewer2.2_x64.exe**.



Browse LIPS_Samples	2022/12/13
Download Latest SDK	2022/12/13
LIPS GitHub Resources	2022/12/13
LIPSSL	2022/12/13
Ni2PointCloud-gl	2022/12/13
NiViewer2.2_x64	2022/12/13
OpenNI2 Programming API	2022/12/13
Uninstall LIPS SDK	2022/12/13

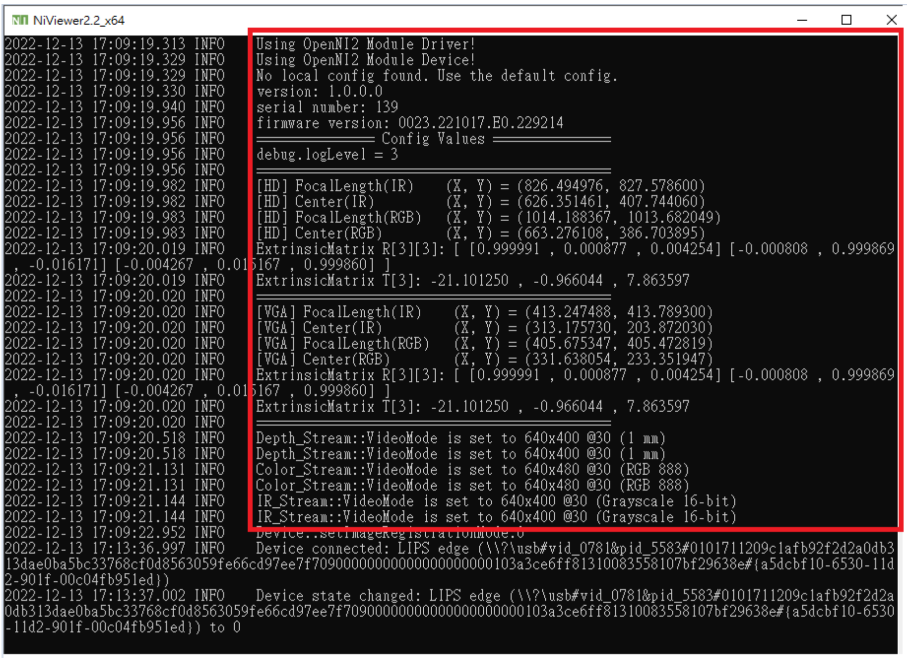
3. Upon activation, a **Status Window** and a **Viewer Window** pop up.



```
022-12-13 17:09:19.313 INFO    Using OpenNI2 Module Driver!
022-12-13 17:09:19.329 INFO    Using OpenNI2 Module Device!
022-12-13 17:09:19.329 INFO    No local config found. Use the default config.
022-12-13 17:09:19.330 INFO    version: 1.0.0.0
022-12-13 17:09:19.940 INFO    serial number: 139
022-12-13 17:09:19.956 INFO    firmware version: 0023.221017.E0.229214
022-12-13 17:09:19.956 INFO    ────────── Config Values ──────────
022-12-13 17:09:19.956
022-12-13 17:09:19.956
022-12-13 17:09:19.982
022-12-13 17:09:19.982
022-12-13 17:09:19.983
022-12-13 17:09:19.983
022-12-13 17:09:20.019
022-12-13 17:09:20.019
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.020
022-12-13 17:09:20.518
022-12-13 17:09:20.518
022-12-13 17:09:21.131
022-12-13 17:09:21.131
022-12-13 17:09:21.144
022-12-13 17:09:21.144
022-12-13 17:09:22.952
022-12-13 17:13:36.997
f81310083558107bf29638e
022-12-13 17:13:37.002
ce6ff81310083558107bf29
```

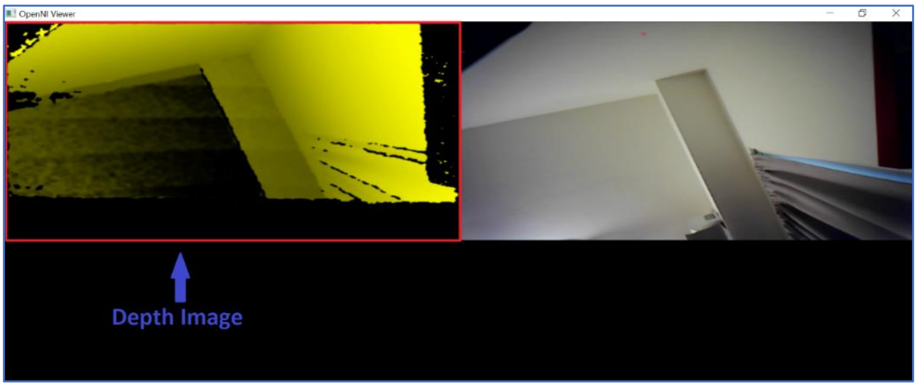
Image registration is off . Capture Formats - Depth: Lossless | Image: Lossy | IR: Lossless

4. On the **Status Window**, you can view the camera information, such as SDK build version, SDK time stamp, camera ID, status.

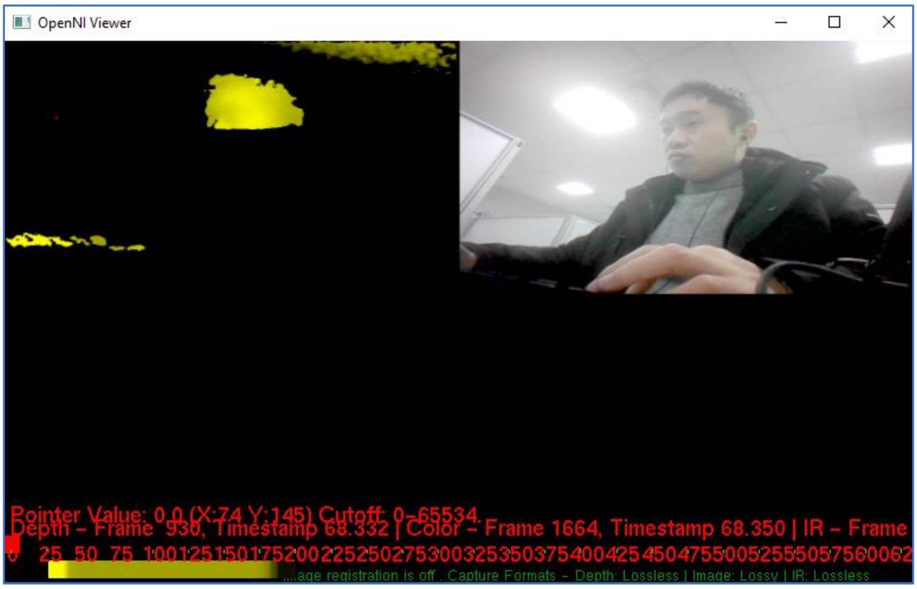


```
2022-12-13 17:09:19.313 INFO    Using OpenNI2 Module Driver!
2022-12-13 17:09:19.329 INFO    Using OpenNI2 Module Device!
2022-12-13 17:09:19.329 INFO    No local config found. Use the default config.
2022-12-13 17:09:19.330 INFO    version: 1.0.0.0
2022-12-13 17:09:19.940 INFO    serial number: 139
2022-12-13 17:09:19.956 INFO    firmware version: 0023.221017.E0.229214
2022-12-13 17:09:19.956 INFO    ═══════════════ Config Values ═══════════════
2022-12-13 17:09:19.956 INFO    debug.logLevel = 3
2022-12-13 17:09:19.956 INFO    ══════════════════════════════════════════════
2022-12-13 17:09:19.982 INFO    [HD] FocalLength(IR)    (X, Y) = (826.494976, 827.578600)
2022-12-13 17:09:19.982 INFO    [HD] Center(IR)         (X, Y) = (626.351461, 407.744060)
2022-12-13 17:09:19.983 INFO    [HD] FocalLength(RGB)   (X, Y) = (1014.188367, 1013.682049)
2022-12-13 17:09:19.983 INFO    [HD] Center(RGB)        (X, Y) = (663.276108, 386.703895)
2022-12-13 17:09:20.019 INFO    ExtrinsicMatrix R[3][3]: [ [0.999991 , 0.000877 , 0.004254] [-0.000808 , 0.999869
 , -0.016171] [-0.004267 , 0.015167 , 0.999860] ]
2022-12-13 17:09:20.019 INFO    ExtrinsicMatrix T[3]: -21.101250 , -0.966044 , 7.863597
2022-12-13 17:09:20.020 INFO    ══════════════════════════════════════════════
2022-12-13 17:09:20.020 INFO    [VGA] FocalLength(IR)    (X, Y) = (413.247488, 413.789300)
2022-12-13 17:09:20.020 INFO    [VGA] Center(IR)         (X, Y) = (313.175730, 203.872030)
2022-12-13 17:09:20.020 INFO    [VGA] FocalLength(RGB)   (X, Y) = (405.675347, 405.472819)
2022-12-13 17:09:20.020 INFO    [VGA] Center(RGB)        (X, Y) = (331.638054, 233.351947)
2022-12-13 17:09:20.020 INFO    ExtrinsicMatrix R[3][3]: [ [0.999991 , 0.000877 , 0.004254] [-0.000808 , 0.999869
 , -0.016171] [-0.004267 , 0.015167 , 0.999860] ]
2022-12-13 17:09:20.020 INFO    ExtrinsicMatrix T[3]: -21.101250 , -0.966044 , 7.863597
2022-12-13 17:09:20.020 INFO    ══════════════════════════════════════════════
2022-12-13 17:09:20.518 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)
2022-12-13 17:09:20.518 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)
2022-12-13 17:09:21.131 INFO    Color_Stream::VideoMode is set to 640x480 @30 (RGB 888)
2022-12-13 17:09:21.131 INFO    Color_Stream::VideoMode is set to 640x480 @30 (RGB 888)
2022-12-13 17:09:21.144 INFO    IR_Stream::VideoMode is set to 640x400 @30 (Grayscale 16-bit)
2022-12-13 17:09:21.144 INFO    IR_Stream::VideoMode is set to 640x400 @30 (Grayscale 16-bit)
2022-12-13 17:09:22.952 INFO    Device::setImageRegistrationMode:0
2022-12-13 17:13:36.997 INFO    Device connected: LIPS edge (\\?\usb#vid_0781&pid_5583#0101711209c1afb92f2d2a0db3
13dae0ba5bc33768cf0d8563059fe66cd97ee7f709000000000000000000000103a3ce6ff81310083558107bf29638e#{a5dcbf10-6530-11d
2-901f-00c04fb951ed})
2022-12-13 17:13:37.002 INFO    Device state changed: LIPS edge (\\?\usb#vid_0781&pid_5583#0101711209c1afb92f2d2a
0db313dae0ba5bc33768cf0d8563059fe66cd97ee7f709000000000000000000000103a3ce6ff81310083558107bf29638e#{a5dcbf10-6530
-11d2-901f-00c04fb951ed}) to 0
```

5. On the **Viewer Window**, the **Depth & RGB images** are displayed on screen

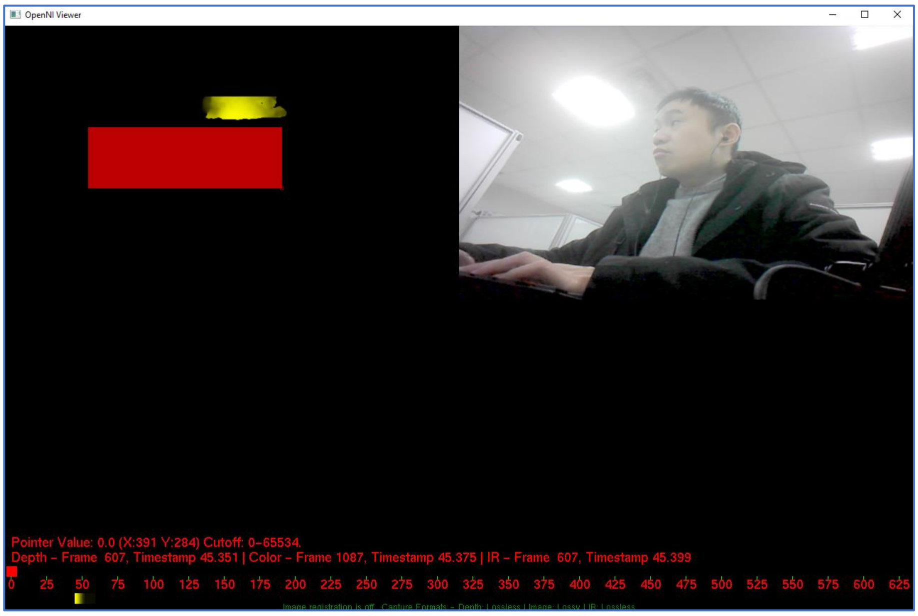


6. Press **p** to activate the **point mode**. The meter below shows the depth data of the given point marked by the indicator.



```
Pointer Value: 0.0 (X:74 Y:145) Cutoff: 0-65534.
Depth - Frame  930, Timestamp 68.332 | Color - Frame 1664, Timestamp 68.350 | IR - Frame
....age registration is off . Capture Formats - Depth: Lossless | Image: Lossy | IR: Lossless
```

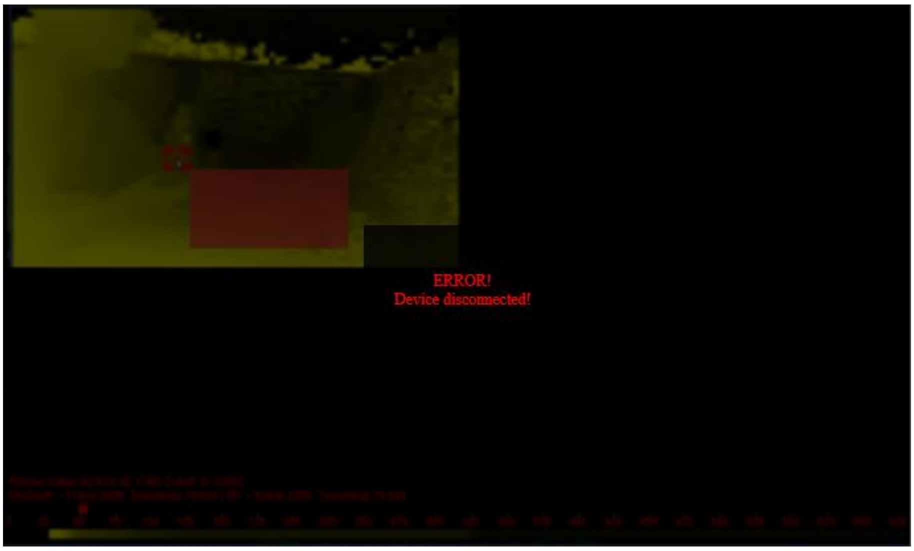
7. Optionally choose an area to display by dragging a cropping area with your cursor. The rest of the areas will be masked from displaying.



```
Pointer Value: 0.0 (X:391 Y:284) Cutoff: 0-65534.
Depth - Frame  607, Timestamp 45.351 | Color - Frame 1087, Timestamp 45.375 | IR - Frame  607, Timestamp 45.399
Image registration is off . Capture Formats - Depth: Lossless | Image: Lossy | IR: Lossless
```

8. Optionally press **F** to maximize / minimize the viewer's window.

Note: If for any reason the USB cable is forcibly removed while NiViewer is functioning, an **ERROR! Device disconnected** message appears on the screen. Please close NiViewer window and try again.



ERROR!
Device disconnected!

[NiViewer Settings]

Right-click on the NiViewer window to access the following image settings.

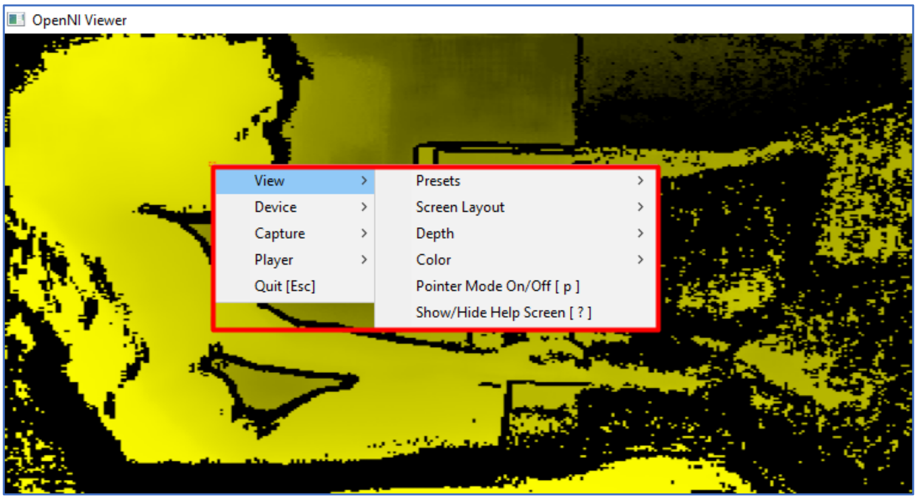


Figure 4-8

View

- **Presets**: Displays the depth / color images in 11 styles. By default, depth / color images are displayed **Side By Side**.
- **Screen Layout**: Selects the arrangement for multiple camera images between **Side by Side** or **Overlay**. With **Side By Side** layout, depth / color images are displayed side by side. With **Overlay**, depth / color images are merged and displayed simultaneously.

- **Depth**: Displays the depth images in various styles or turns the depth image **Off**.
- **Color**: Select **Depth Masked Color** to display color images masked with depth grains or select **Normal** to display regular color images. Select **Off** to turns the depth image off.
- **Pointer Mode On / Off [p]**: Enables **Pointer Mode** to display depth value of a given point on camera images with a red dot.
- **Show / Hide Help Screen [?]**: Shows the **Help Screen** which contains the description of keyboard shortcuts with their functions.

<u>**Device**</u>

- **Streams**: Adjusts the image streaming settings of **depth**, **Color,** and **IR** images. Be default, IR images are overlapped with color images. Turn the color images off to display IR images.
  - **Video Mode**:

    For **depth images**, select the grain size for each resolution. You can choose between **1 mm** and **100 µm**. For **color images**, select the image resolution between **640 X 480 30FPS (RGB888)** and **1280 X 800 15 FPS (RGB888)**.

    For **IR images**, select the **image resolution** and **frame per second** to be displayed. When the image resolution is changed, users can confirm the change in the Status Window message.
  - **Mirror**: Flips the streaming image horizontally.
  - **Reset Cropping**: Reset the cropping area previously assigned.
- **Registration**: Overlays the depth image onto the color image or turn the image overlap off.
- **Mirror All [m]**: Flips the streaming image horizontally for every stream.

<u>**Capture**</u>

- **Depth Capturing**: Assign the image format for depth image stream when recording images.
- **Image Capturing**: Assign the image format for color image stream when recording images. Select **Lossless** to record the image at uncompressed quality or **Lossy** for compressed quality.
- **IR Capturing**: Assign the image format for IR image stream when recording images.
- **Browse**: Locate the recordings of camera images in your PC / laptop.
- **Start [s]**: Assign a path for saving the recording files and start recording images.
- **Start (5 sec delay) [d]**: Starts to record images 5 seconds upon clicking this function.

- **Restart**: Plays the recorded image again.
- **Stop [x]**: Stops recording camera image.

Player

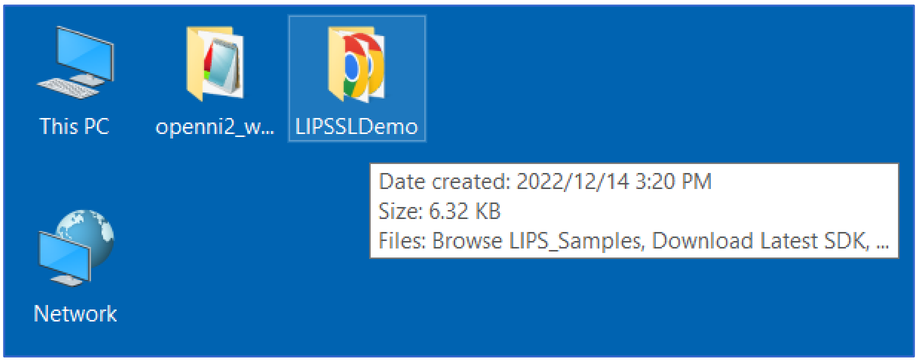
- **Pause / Resume [Space]:** Pauses the playback of the recording or resumes playback.
- **Read one Frame [;]**: This function appears only when the playback is paused. Click to load the playback image of the next frame.

6. Other Test Applications

Besides the NiViewer program, LIPS Corp. also provide other testing tools for developers to access the internal parameters of the camera.

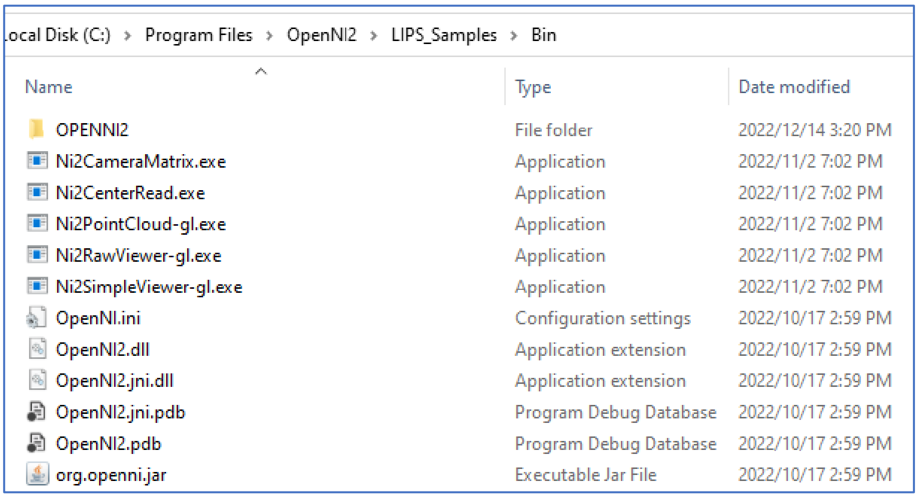
To access the test applications:

1. On your desktop, click the **LIPSSLDemo** folder.



Date created: 2022/12/14 3:20 PM
Size: 6.32 KB
Files: Browse LIPS_Samples, Download Latest SDK, ...

2. Clink **Browse LIPS_Samples** to move to the path below to find more test Applications C:\Program Files\OpenNI2\LIPS_Samples\Bin. You can find other test applications and their source code in the destination.



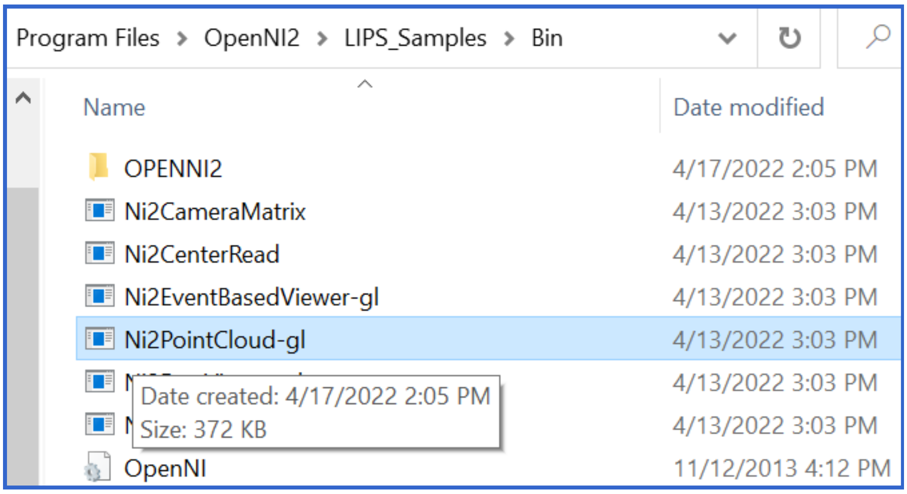
.ocal Disk (C:) › Program Files › OpenNI2 › LIPS_Samples › Bin

Name	Type	Date modified
OPENNI2	File folder	2022/12/14 3:20 PM
Ni2CameraMatrix.exe	Application	2022/11/2 7:02 PM
Ni2CenterRead.exe	Application	2022/11/2 7:02 PM
Ni2PointCloud-gl.exe	Application	2022/11/2 7:02 PM
Ni2RawViewer-gl.exe	Application	2022/11/2 7:02 PM
Ni2SimpleViewer-gl.exe	Application	2022/11/2 7:02 PM
OpenNI.ini	Configuration settings	2022/10/17 2:59 PM
OpenNI2.dll	Application extension	2022/10/17 2:59 PM
OpenNI2.jni.dll	Application extension	2022/10/17 2:59 PM
OpenNI2.jni.pdb	Program Debug Database	2022/10/17 2:59 PM
OpenNI2.pdb	Program Debug Database	2022/10/17 2:59 PM
org.openni.jar	Executable Jar File	2022/10/17 2:59 PM

6.1 Ni2PointCloud

You can view point cloud images through **Ni2PointCloud.exe**. To access Ni2PointCloud, refer to *6. Other Test Applications.*

1. In **LIPS_Samples/Bin**, click Ni2PointCloud-gl.exe.

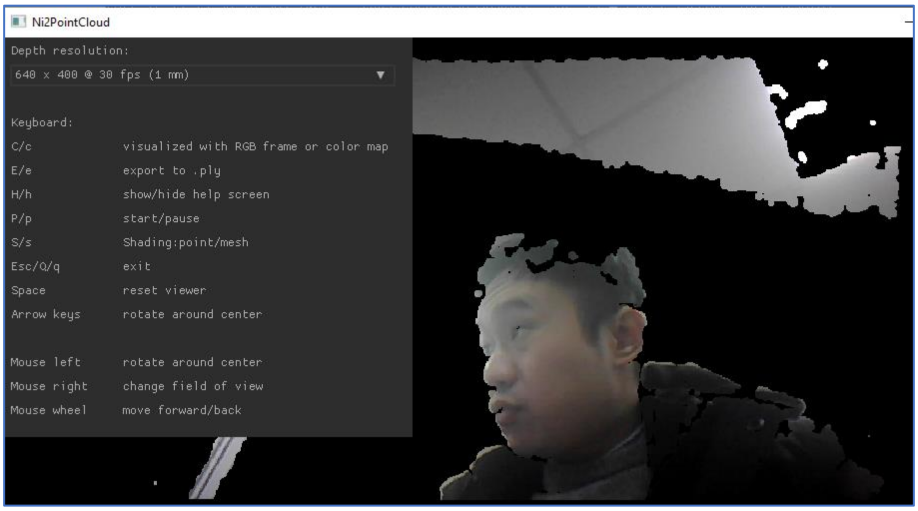


Program Files › OpenNI2 › LIPS_Samples › Bin

Name	Date modified
OPENNI2	4/17/2022 2:05 PM
Ni2CameraMatrix	4/13/2022 3:03 PM
Ni2CenterRead	4/13/2022 3:03 PM
Ni2EventBasedViewer-gl	4/13/2022 3:03 PM
Ni2PointCloud-gl	4/13/2022 3:03 PM
N	4/13/2022 3:03 PM
N	4/13/2022 3:03 PM
OpenNI	11/12/2013 4:12 PM

Date created: 4/17/2022 2:05 PM
Size: 372 KB

2. You can view point cloud images on live.

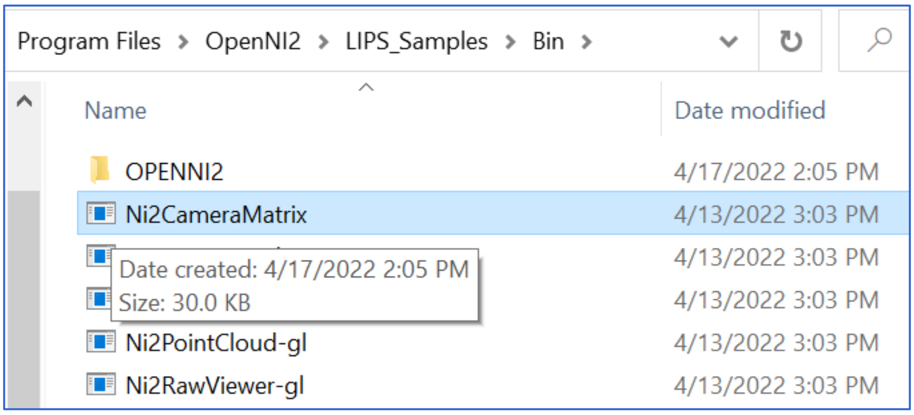


```
Ni2PointCloud

Depth resolution:
640 x 400 @ 30 fps (1 mm)                    ▼

Keyboard:
C/c            visualized with RGB frame or color map
E/e            export to .ply
H/h            show/hide help screen
P/p            start/pause
S/s            Shading:point/mesh
Esc/Q/q        exit
Space          reset viewer
Arrow keys     rotate around center

Mouse left     rotate around center
Mouse right    change field of view
Mouse wheel    move forward/back
```

6.2 Ni2CameraMatrix

You can access the camera’s intrinsic& extrinsic matrix through **Ni2CameraMatrix.exe** for mapping color / depth images. To access Ni2CameraMatrix, refer to *6. Other Test Applications*.

1. In **LIPS_Samples/Bin**, click **Ni2CameraMatrix.exe**.

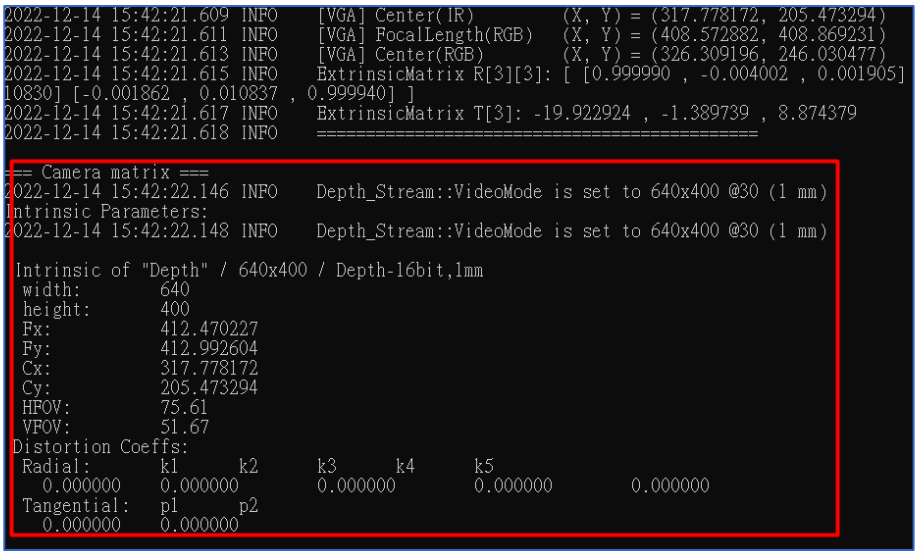


Program Files › OpenNI2 › LIPS_Samples › Bin ›

Name	Date modified
OPENNI2	4/17/2022 2:05 PM
Ni2CameraMatrix	4/13/2022 3:03 PM
	4/13/2022 3:03 PM
	4/13/2022 3:03 PM
Ni2PointCloud-gl	4/13/2022 3:03 PM
Ni2RawViewer-gl	4/13/2022 3:03 PM

Date created: 4/17/2022 2:05 PM
Size: 30.0 KB

2. A window containing the camera’s intrinsic & extrinsic matrix pops up.

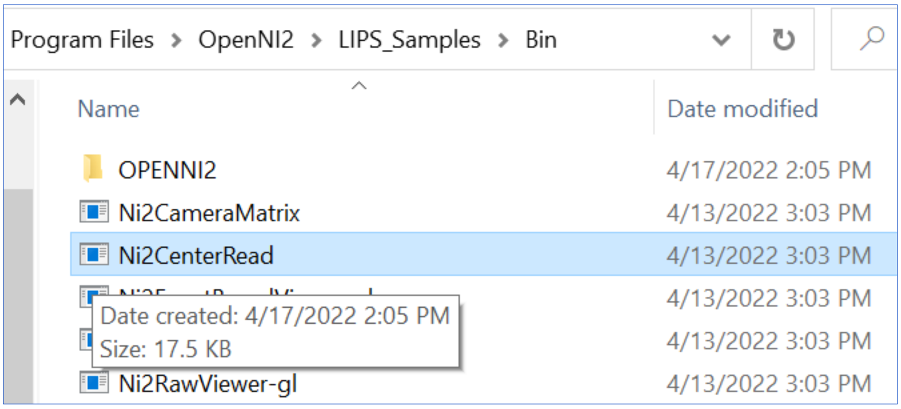


```
2022-12-14 15:42:21.609 INFO    [VGA] Center(IR)          (X, Y) = (317.778172, 205.473294)
2022-12-14 15:42:21.611 INFO    [VGA] FocalLength(RGB)    (X, Y) = (408.572882, 408.869231)
2022-12-14 15:42:21.613 INFO    [VGA] Center(RGB)         (X, Y) = (326.309196, 246.030477)
2022-12-14 15:42:21.615 INFO    ExtrinsicMatrix R[3][3]: [ [0.999990 , -0.004002 , 0.001905]
10830] [-0.001862 , 0.010837 , 0.999940] ]
2022-12-14 15:42:21.617 INFO    ExtrinsicMatrix T[3]: -19.922924 , -1.389739 , 8.874379
2022-12-14 15:42:21.618 INFO    ==============================================

=== Camera matrix ===
2022-12-14 15:42:22.146 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)
Intrinsic Parameters:
2022-12-14 15:42:22.148 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)

 Intrinsic of "Depth" / 640x400 / Depth-16bit,1mm
  width:        640
  height:       400
  Fx:           412.470227
  Fy:           412.992604
  Cx:           317.778172
  Cy:           205.473294
  HFOV:         75.61
  VFOV:         51.67
 Distortion Coeffs:
  Radial:       k1      k2      k3      k4      k5
    0.000000    0.000000        0.000000        0.000000        0.000000
  Tangential:   p1      p2
    0.000000    0.000000
```

6.3 Ni2CenterRead

You can access the depth value of center point through **Ni2CenterRead.exe**. To access Ni2CenterRead, refer to *6. Other Test Applications*.

1. In **LIPS_Samples/Bin**, click **Ni2CenterRead.exe**.

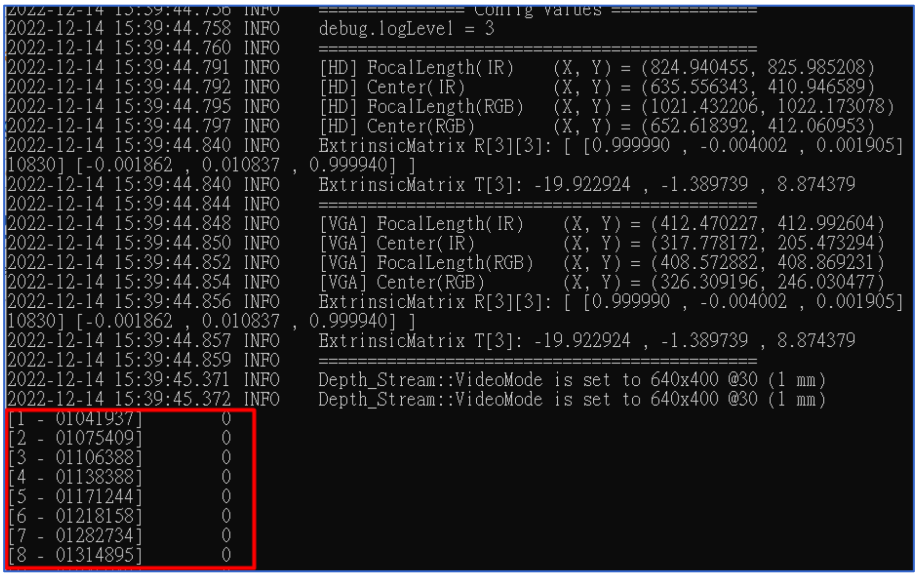


Program Files > OpenNI2 > LIPS_Samples > Bin

Name	Date modified
OPENNI2	4/17/2022 2:05 PM
Ni2CameraMatrix	4/13/2022 3:03 PM
Ni2CenterRead	4/13/2022 3:03 PM
[illegible]	4/13/2022 3:03 PM
[illegible]	4/13/2022 3:03 PM
Ni2RawViewer-gl	4/13/2022 3:03 PM

Date created: 4/17/2022 2:05 PM
Size: 17.5 KB

2. A window containing the coordinates of the depth value of center point pops up.



```
2022-12-14 15:39:44.756 INFO    =============== Config Values ===============
2022-12-14 15:39:44.758 INFO    debug.logLevel = 3
2022-12-14 15:39:44.760 INFO    =============================================
2022-12-14 15:39:44.791 INFO    [HD] FocalLength(IR)    (X, Y) = (824.940455, 825.985208)
2022-12-14 15:39:44.792 INFO    [HD] Center(IR)         (X, Y) = (635.556343, 410.946589)
2022-12-14 15:39:44.795 INFO    [HD] FocalLength(RGB)   (X, Y) = (1021.432206, 1022.173078)
2022-12-14 15:39:44.797 INFO    [HD] Center(RGB)        (X, Y) = (652.618392, 412.060953)
2022-12-14 15:39:44.840 INFO    ExtrinsicMatrix R[3][3]: [ [0.999990 , -0.004002 , 0.001905]
10830] [-0.001862 , 0.010837 , 0.999940] ]
2022-12-14 15:39:44.840 INFO    ExtrinsicMatrix T[3]: -19.922924 , -1.389739 , 8.874379
2022-12-14 15:39:44.844 INFO    =============================================
2022-12-14 15:39:44.848 INFO    [VGA] FocalLength(IR)    (X, Y) = (412.470227, 412.992604)
2022-12-14 15:39:44.850 INFO    [VGA] Center(IR)         (X, Y) = (317.778172, 205.473294)
2022-12-14 15:39:44.852 INFO    [VGA] FocalLength(RGB)   (X, Y) = (408.572882, 408.869231)
2022-12-14 15:39:44.854 INFO    [VGA] Center(RGB)        (X, Y) = (326.309196, 246.030477)
2022-12-14 15:39:44.856 INFO    ExtrinsicMatrix R[3][3]: [ [0.999990 , -0.004002 , 0.001905]
10830] [-0.001862 , 0.010837 , 0.999940] ]
2022-12-14 15:39:44.857 INFO    ExtrinsicMatrix T[3]: -19.922924 , -1.389739 , 8.874379
2022-12-14 15:39:44.859 INFO    =============================================
2022-12-14 15:39:45.371 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)
2022-12-14 15:39:45.372 INFO    Depth_Stream::VideoMode is set to 640x400 @30 (1 mm)
[1 - 01041937]        0
[2 - 01075409]        0
[3 - 01106388]        0
[4 - 01138388]        0
[5 - 01171244]        0
[6 - 01218158]        0
[7 - 01282734]        0
[8 - 01314895]        0
```

7. Compilation Requirements

LIPS Corp. provides the platform-based **source code packages** of computer vision functions which developers can modify and compile according to individual development needs.

The Windows system requires a series of third-party programs for LIPSedge™ L210u / L215u programming. Before your development starts, refer to the following chapters to install the following third-party programs.

We recommend using the following application / compiler for LIPS SDK.

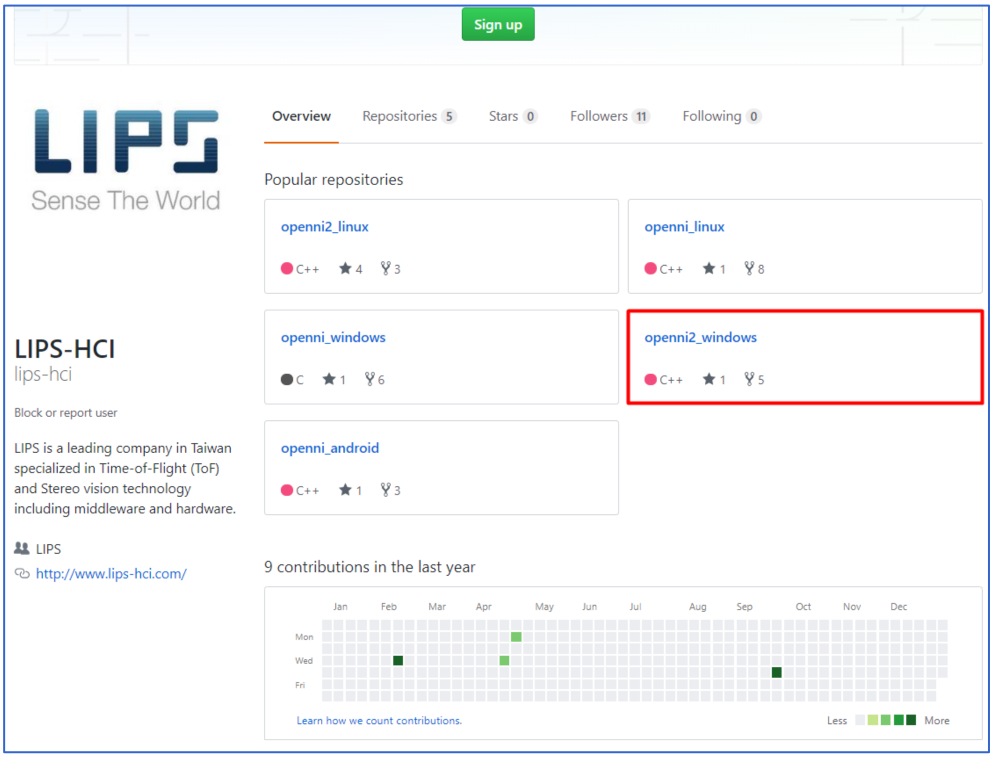
- Windows 10 64-bit system or later
- Microsoft Visual Studio 2015 or later
- OpenNI 2.2

Note: Due to compatibility issues, we recommend using Microsoft Visual Studio 2022 as the standard for developing your application.

7.1 Example Program Source Code Setup

LIPS Corp. provides the platform-based **source code packages** of computer vision functions. When downloading, select the Build Sample according to **the platform** and **OpenNI version**.

1. Select the build sample from https://github.com/lips-hci. Here, we use **openni2_windows** as an example.



Sign up

Overview | Repositories 5 | Stars 0 | Followers 11 | Following 0

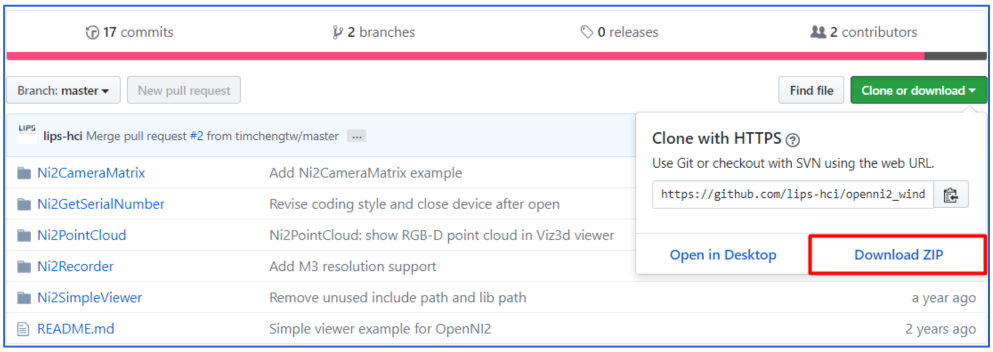
Popular repositories

openni2_linux — C++ ★ 4 ⑂ 3
openni_linux — C++ ★ 1 ⑂ 8
openni_windows — C ★ 1 ⑂ 6
openni2_windows — C++ ★ 1 ⑂ 5
openni_android — C++ ★ 1 ⑂ 3

LIPS-HCI
lips-hci
Block or report user
LIPS is a leading company in Taiwan specialized in Time-of-Flight (ToF) and Stereo vision technology including middleware and hardware.
LIPS
http://www.lips-hci.com/

9 contributions in the last year

2. Click **Clone or Download** > **Download ZIP**.



17 commits | 2 branches | 0 releases | 2 contributors

Branch: master ▾ | New pull request | Find file | Clone or download ▾

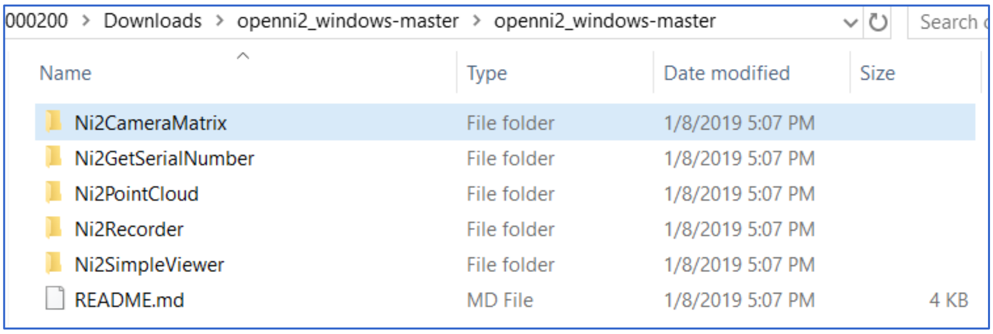
lips-hci Merge pull request #2 from timchengtw/master …

Ni2CameraMatrix — Add Ni2CameraMatrix example
Ni2GetSerialNumber — Revise coding style and close device after open
Ni2PointCloud — Ni2PointCloud: show RGB-D point cloud in Viz3d viewer
Ni2Recorder — Add M3 resolution support
Ni2SimpleViewer — Remove unused include path and lib path — a year ago
README.md — Simple viewer example for OpenNI2 — 2 years ago

Clone with HTTPS
Use Git or checkout with SVN using the web URL.
https://github.com/lips-hci/openni2_wind

Open in Desktop | Download ZIP

3. Right-click and extract the downloaded file. The **Build Samples** are accessible in the **openni2_windows-master** folder.



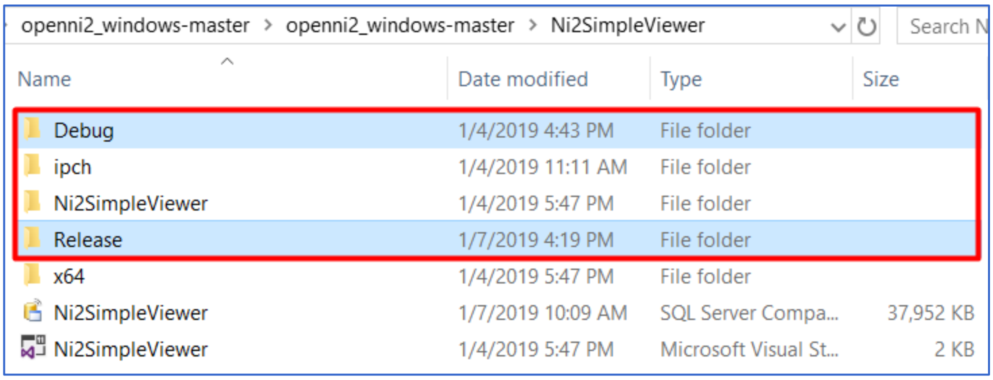
000200 > Downloads > openni2_windows-master > openni2_windows-master

Name	Type	Date modified	Size
Ni2CameraMatrix	File folder	1/8/2019 5:07 PM	
Ni2GetSerialNumber	File folder	1/8/2019 5:07 PM	
Ni2PointCloud	File folder	1/8/2019 5:07 PM	
Ni2Recorder	File folder	1/8/2019 5:07 PM	
Ni2SimpleViewer	File folder	1/8/2019 5:07 PM	
README.md	MD File	1/8/2019 5:07 PM	4 KB

Note:

1. The build samples are for demo purposes only, if you need further programmig assistance, contact LIPS Corp. at info@lips-hci.com.
2. We recommend startin with **Ni2GetSerialNumber** and take a trial with execution feedback from LIPSedge^{TM} L210u/L215u.

4. Assign a directory in your PC / laptop to store the sample application. If you choose to start developing your application by using the Build Samples, the sample application you released will be stored in the
Debug / Release folder under the original location of each
Build Samples.



openni2_windows-master > openni2_windows-master > Ni2SimpleViewer

Name	Date modified	Type	Size
Debug	1/4/2019 4:43 PM	File folder	
ipch	1/4/2019 11:11 AM	File folder	
Ni2SimpleViewer	1/4/2019 5:47 PM	File folder	
Release	1/7/2019 4:19 PM	File folder	
x64	1/4/2019 5:47 PM	File folder	
Ni2SimpleViewer	1/7/2019 10:09 AM	SQL Server Compa...	37,952 KB
Ni2SimpleViewer	1/4/2019 5:47 PM	Microsoft Visual St...	2 KB

Note: Visual Studio requires users to complete the release procedure at least once to generate the storage directory for your sample application. You can skip the procedure by simply creating **Debug / Release** folders in the storage directory.

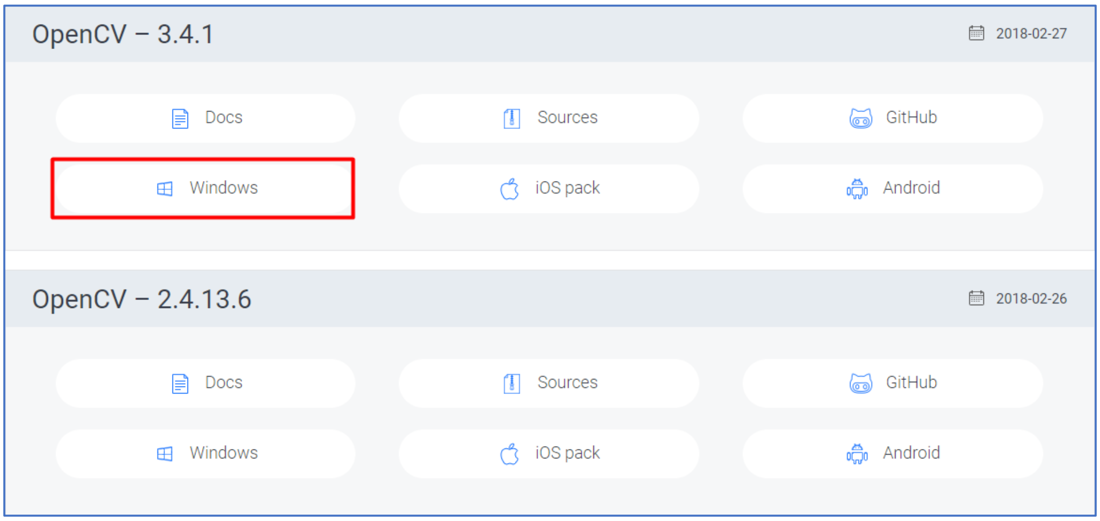
7.2 OpenCV Libraries Installation

LIPS SDK and build samples requires functions from **OpenCV Libraries**.

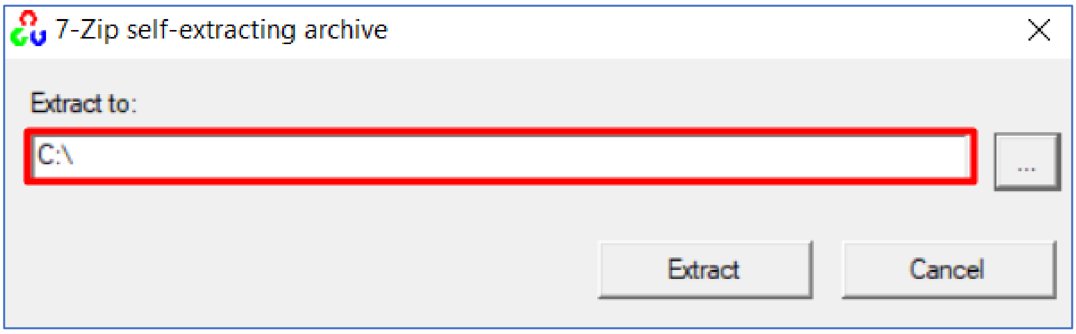
For LIPS SDK earlier than V1.0.0.0 or developers who plan to develop programs based on our build samples, install OpenCV libraries before your development starts.

For LIPS SDK on or later than V1.0.0.0, the OpenCV libraries are built into the SDK and thus there is no need for the installation.

1. Download **OpenCV 3.4.1** from https://sourceforge.net/projects/opencvlibrary/files/opencv-win/3.4.1/opencv-3.4.1-vc14_vc15.exe



2. Click **opencv-3.4.1** and extract the file to the **root directory** on the C: drive of your local PC / laptop.

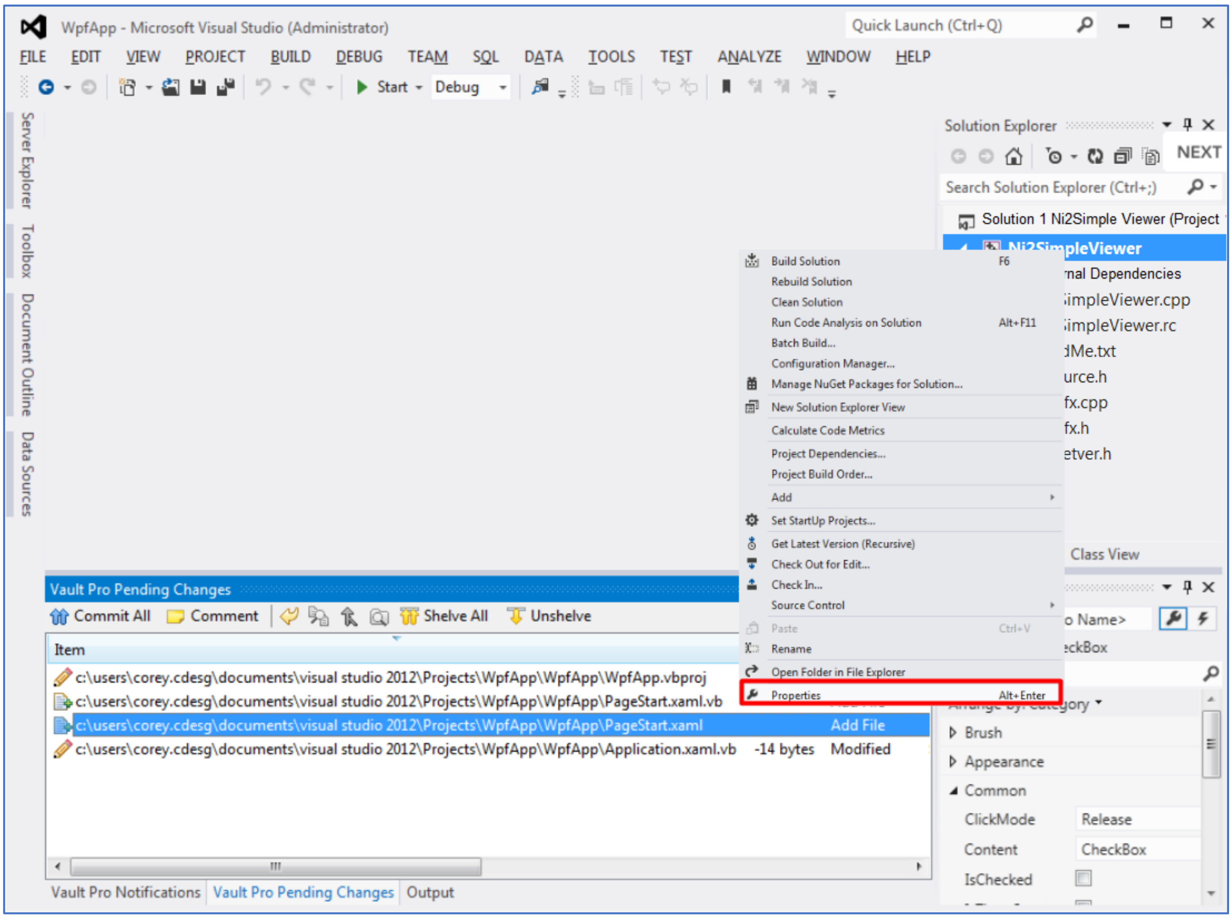


Extract to: C:\

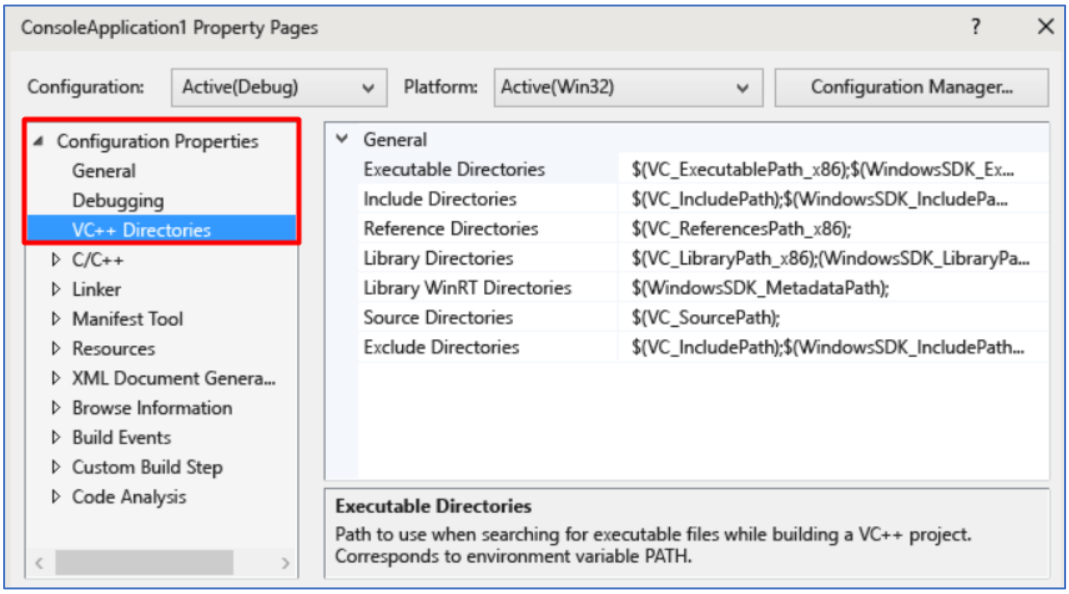
7.3 OpenNI Libraries Setup (Visual Studio)

Import **OpenNI libraries** to the Build Sample project in your Visual Studio. Otherwise, errors occur during the process of generating applications.

1. On your editor, right-click the Build Sample and select **Properties**.
   A window pops up.


WpfApp - Microsoft Visual Studio (Administrator)
Quick Launch (Ctrl+Q)
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL DATA TOOLS TEST ANALYZE WINDOW HELP
Start Debug
Server Explorer Toolbox Document Outline Data Sources
Solution Explorer
NEXT
Search Solution Explorer (Ctrl+;)
Solution 1 Ni2Simple Viewer (Project
Ni2SimpleViewer
rnal Dependencies
impleViewer.cpp
impleViewer.rc
dMe.txt
urce.h
fx.cpp
fx.h
etver.h
Build Solution F6
Rebuild Solution
Clean Solution
Run Code Analysis on Solution Alt+F11
Batch Build...
Configuration Manager...
Manage NuGet Packages for Solution...
New Solution Explorer View
Calculate Code Metrics
Project Dependencies...
Project Build Order...
Add
Set StartUp Projects...
Get Latest Version (Recursive)
Check Out for Edit...
Check In...
Source Control
Paste Ctrl+V
Rename
Open Folder in File Explorer
Properties Alt+Enter
Class View
o Name>
ckBox
Brush
Appearance
Common
ClickMode Release
Content CheckBox
IsChecked
Vault Pro Pending Changes
Commit All Comment Shelve All Unshelve
Item
c:\users\corey.cdesg\documents\visual studio 2012\Projects\WpfApp\WpfApp\WpfApp.vbproj
c:\users\corey.cdesg\documents\visual studio 2012\Projects\WpfApp\WpfApp\PageStart.xaml.vb
c:\users\corey.cdesg\documents\visual studio 2012\Projects\WpfApp\WpfApp\PageStart.xaml Add File
c:\users\corey.cdesg\documents\visual studio 2012\Projects\WpfApp\WpfApp\Application.xaml.vb -14 bytes Modified
Vault Pro Notifications Vault Pro Pending Changes Output

2. On the left menu, click **Configuring Properties**, and select **VC++ Directories**.



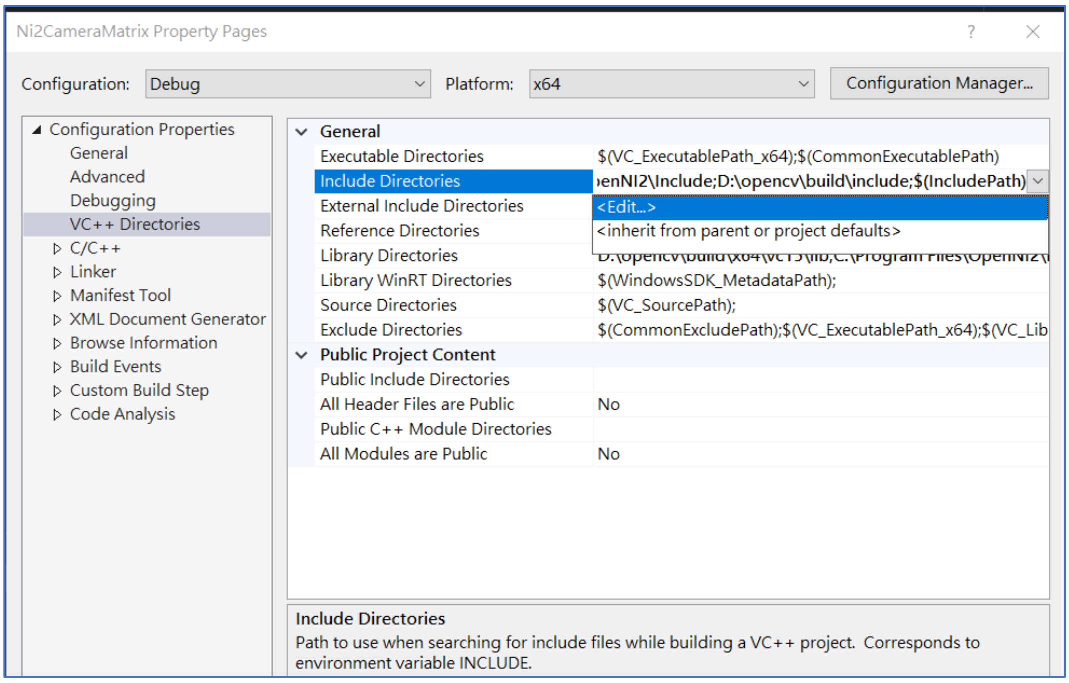
Configuration: Active(Debug)
Platform: Active(Win32)

General

Executable Directories	$(VC_ExecutablePath_x86);$(WindowsSDK_Ex...
Include Directories	$(VC_IncludePath);$(WindowsSDK_IncludePa...
Reference Directories	$(VC_ReferencesPath_x86);
Library Directories	$(VC_LibraryPath_x86);(WindowsSDK_LibraryPa...
Library WinRT Directories	$(WindowsSDK_MetadataPath);
Source Directories	$(VC_SourcePath);
Exclude Directories	$(VC_IncludePath);$(WindowsSDK_IncludePath...

Executable Directories
Path to use when searching for executable files while building a VC++ project. Corresponds to environment variable PATH.

3. On **Include Directories**, click the ⌄ sign and select **Edit**. A window pops up.



Configuration: Debug
Platform: x64

General

Executable Directories	$(VC_ExecutablePath_x64);$(CommonExecutablePath)
Include Directories	›enNI2\Include;D:\opencv\build\include;$(IncludePath)
External Include Directories	
Reference Directories	
Library Directories	D:\opencv\build\x64\vc15\lib;C:\Program Files\OpenNI2\
Library WinRT Directories	$(WindowsSDK_MetadataPath);
Source Directories	$(VC_SourcePath);
Exclude Directories	$(CommonExcludePath);$(VC_ExecutablePath_x64);$(VC_Lib

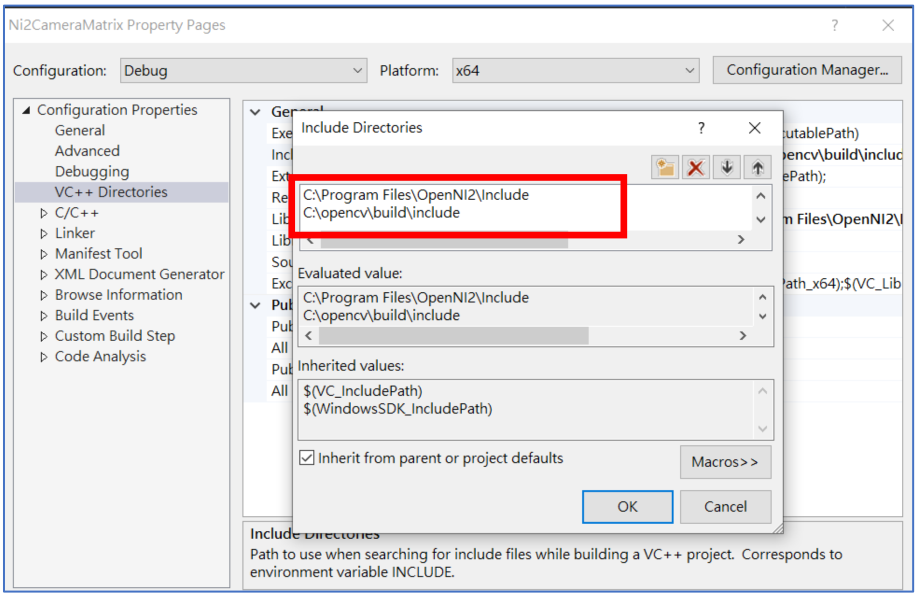
Public Project Content

Public Include Directories	
All Header Files are Public	No
Public C++ Module Directories	
All Modules are Public	No

Include Directories
Path to use when searching for include files while building a VC++ project. Corresponds to environment variable INCLUDE.

4. Click to add the following library paths to the **Include Directories**. The default locations of the library paths are:

Libraries	**Default Locations**
OpenNi2	C:\Program Files\OpenNI2\Include
OpenCV	C:\opencv\build\include



Ni2CameraMatrix Property Pages

Configuration: Debug Platform: x64

Include Directories

```
C:\Program Files\OpenNI2\Include
C:\opencv\build\include
```

Evaluated value:

```
C:\Program Files\OpenNI2\Include
C:\opencv\build\include
```

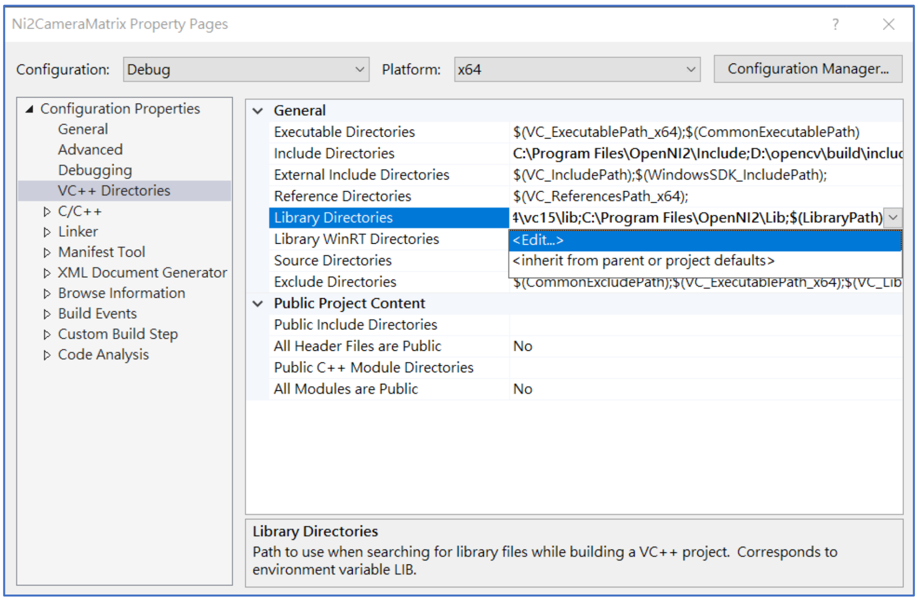
Inherited values:

```
$(VC_IncludePath)
$(WindowsSDK_IncludePath)
```

☑ Inherit from parent or project defaults

Include Directories
Path to use when searching for include files while building a VC++ project. Corresponds to environment variable INCLUDE.

5. On **Library Directories**, click the ⌄ sign and select **Edit**. A window pops up.



Ni2CameraMatrix Property Pages

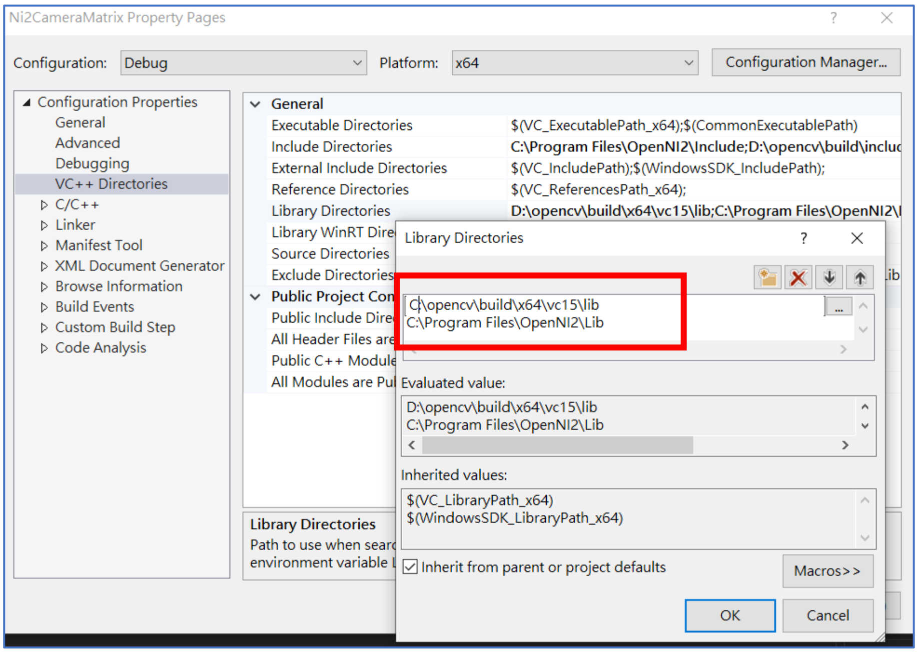
Configuration: Debug Platform: x64

General	
Executable Directories	$(VC_ExecutablePath_x64);$(CommonExecutablePath)
Include Directories	C:\Program Files\OpenNI2\Include;D:\opencv\build\incluc
External Include Directories	$(VC_IncludePath);$(WindowsSDK_IncludePath);
Reference Directories	$(VC_ReferencesPath_x64);
Library Directories	4\vc15\lib;C:\Program Files\OpenNI2\Lib;$(LibraryPath)
Library WinRT Directories	<Edit...> <inherit from parent or project defaults>
Source Directories	
Exclude Directories	$(CommonExcludePath);$(VC_ExecutablePath_x64);$(VC_Lib
Public Project Content	
Public Include Directories	
All Header Files are Public	No
Public C++ Module Directories	
All Modules are Public	No

Library Directories
Path to use when searching for library files while building a VC++ project. Corresponds to environment variable LIB.

6. Click to add the following library paths to the **Library Directories**. The default locations of the library paths are:

Libraries	Default Locations
OpenNi2	C:\Program Files\OpenNI2\Lib
OpenCV	C:\opencv\build\x64\vc15\lib



Ni2CameraMatrix Property Pages

Configuration: Debug   Platform: x64   Configuration Manager...

General
Executable Directories: $(VC_ExecutablePath_x64);$(CommonExecutablePath)
Include Directories: C:\Program Files\OpenNI2\Include;D:\opencv\build\incluc
External Include Directories: $(VC_IncludePath);$(WindowsSDK_IncludePath);
Reference Directories: $(VC_ReferencesPath_x64);
Library Directories: D:\opencv\build\x64\vc15\lib;C:\Program Files\OpenNI2\l
Library WinRT Dire
Source Directories
Exclude Directories
Public Project Con
Public Include Dire
All Header Files are
Public C++ Module
All Modules are Pul

Library Directories
Path to use when sear
environment variable

Library Directories

C:\opencv\build\x64\vc15\lib
C:\Program Files\OpenNI2\Lib

Evaluated value:

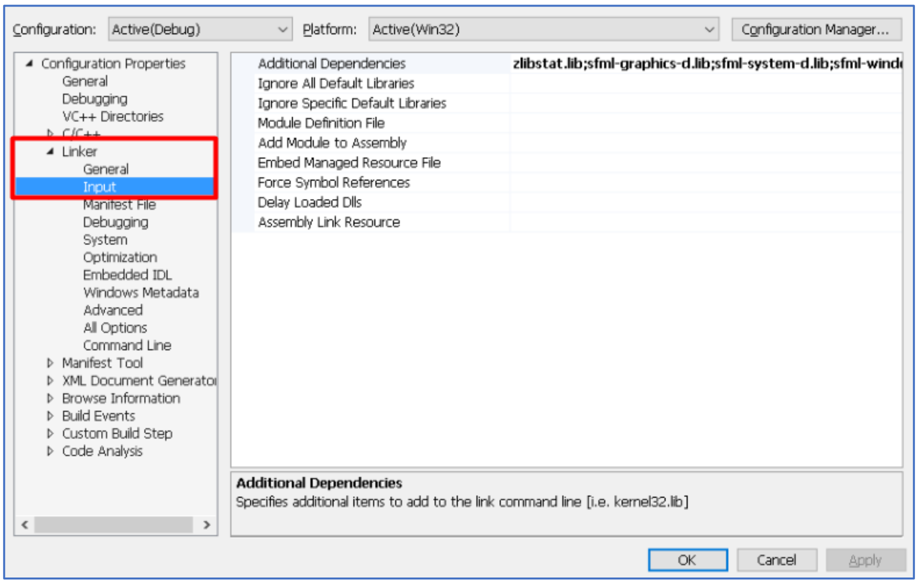
D:\opencv\build\x64\vc15\lib
C:\Program Files\OpenNI2\Lib

Inherited values:

$(VC_LibraryPath_x64)
$(WindowsSDK_LibraryPath_x64)

☑ Inherit from parent or project defaults

7. On the left menu, click **Linker**, and select **Input**.

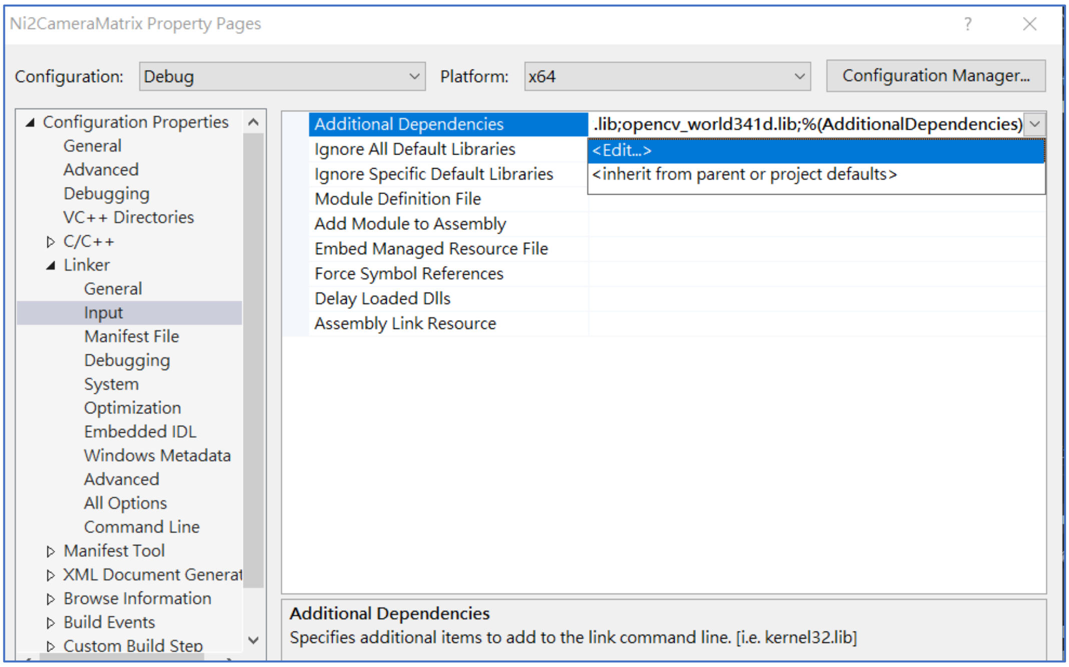


Configuration: Active(Debug)   Platform: Active(Win32)   Configuration Manager...

Additional Dependencies: zlibstat.lib;sfml-graphics-d.lib;sfml-system-d.lib;sfml-wind
Ignore All Default Libraries
Ignore Specific Default Libraries
Module Definition File
Add Module to Assembly
Embed Managed Resource File
Force Symbol References
Delay Loaded Dlls
Assembly Link Resource

Additional Dependencies
Specifies additional items to add to the link command line [i.e. kernel32.lib]

8. On **Additional Dependencies**, click the ⌄ sign and select **Edit**.
A window pops up.



Ni2CameraMatrix Property Pages

Configuration: Debug　Platform: x64　Configuration Manager...

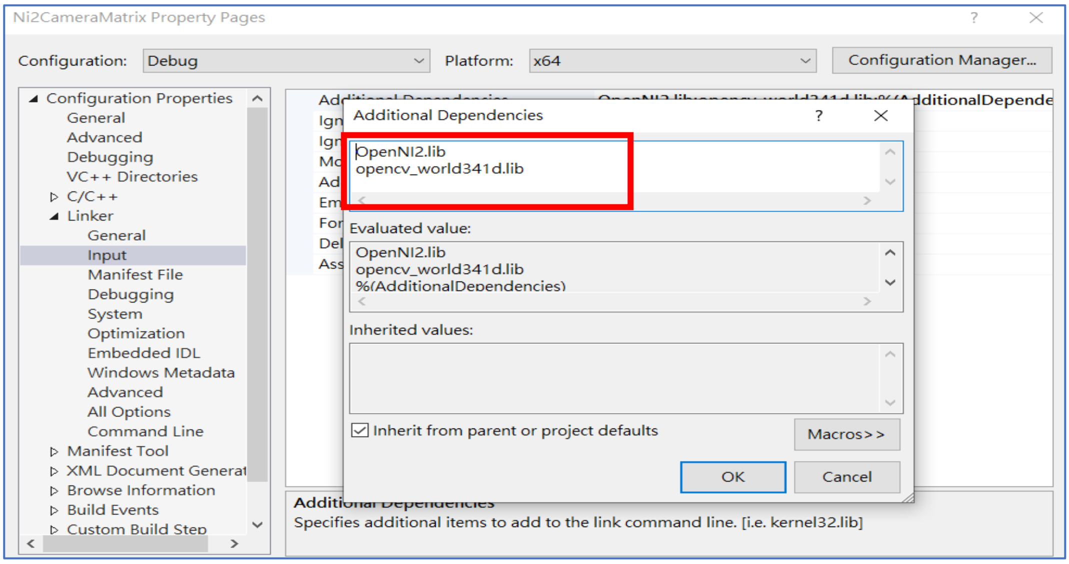
Additional Dependencies	.lib;opencv_world341d.lib;%(AdditionalDependencies)
Ignore All Default Libraries	<Edit...>
Ignore Specific Default Libraries	<inherit from parent or project defaults>
Module Definition File	
Add Module to Assembly	
Embed Managed Resource File	
Force Symbol References	
Delay Loaded Dlls	
Assembly Link Resource	

Additional Dependencies
Specifies additional items to add to the link command line. [i.e. kernel32.lib]

9. Click 📁 to add the following libraries to the

Additional Dependencies folder. The default locations of the library files are:

Open NI2 Library Files	**Default Locations**
OpenNi2.lib	C:\Program Files\OpenNI2\Lib
OpenCV Library Files	**Default Locations**
opencv_world341d.lib	C:\opencv\build\x64\vc15\lib



Ni2CameraMatrix Property Pages

Configuration: Debug　Platform: x64　Configuration Manager...

Additional Dependencies

```
OpenNI2.lib
opencv_world341d.lib
```

Evaluated value:

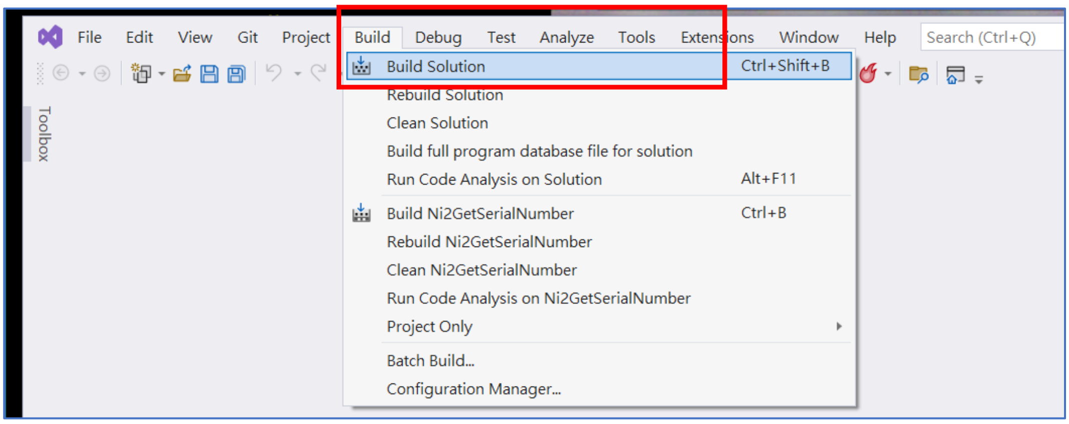
```
OpenNI2.lib
opencv_world341d.lib
%(AdditionalDependencies)
```

Inherited values:

☑ Inherit from parent or project defaults

Additional Dependencies
Specifies additional items to add to the link command line. [i.e. kernel32.lib]

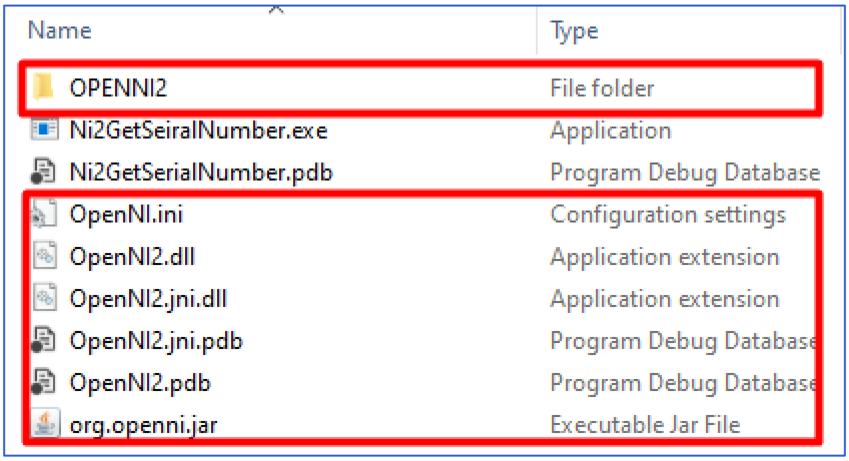
10. Start constructing codes and when the sample application is ready, select **Build** from the top bar and click **Build Solution**. Your sample application will be generated.


File
Edit
View
Git
Project
Build
Debug
Test
Analyze
Tools
Extensions
Window
Help
Search (Ctrl+Q)
Toolbox
Build Solution Ctrl+Shift+B
Rebuild Solution
Clean Solution
Build full program database file for solution
Run Code Analysis on Solution Alt+F11
Build Ni2GetSerialNumber Ctrl+B
Rebuild Ni2GetSerialNumber
Clean Ni2GetSerialNumber
Run Code Analysis on Ni2GetSerialNumber
Project Only
Batch Build...
Configuration Manager...

7.4 OpenNI Libraries Setup (Drivers / Starting Application)

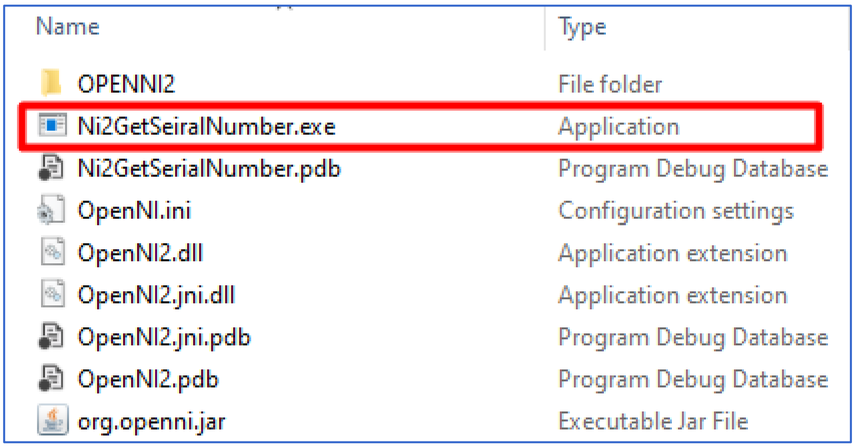
Move **OpenNI libraries** to the directory as the Build Samples call OpenNI functions.

1. Go to **Program Files > OpenNI2 > Redist**. Copy all the files in the Redist folder to the Debug / Release folder.



Name	Type
OPENNI2	File folder
Ni2GetSeiralNumber.exe	Application
Ni2GetSerialNumber.pdb	Program Debug Database
OpenNI.ini	Configuration settings
OpenNI2.dll	Application extension
OpenNI2.jni.dll	Application extension
OpenNI2.jni.pdb	Program Debug Database
OpenNI2.pdb	Program Debug Database
org.openni.jar	Executable Jar File

2. Click the sample application. A command prompt pops up.



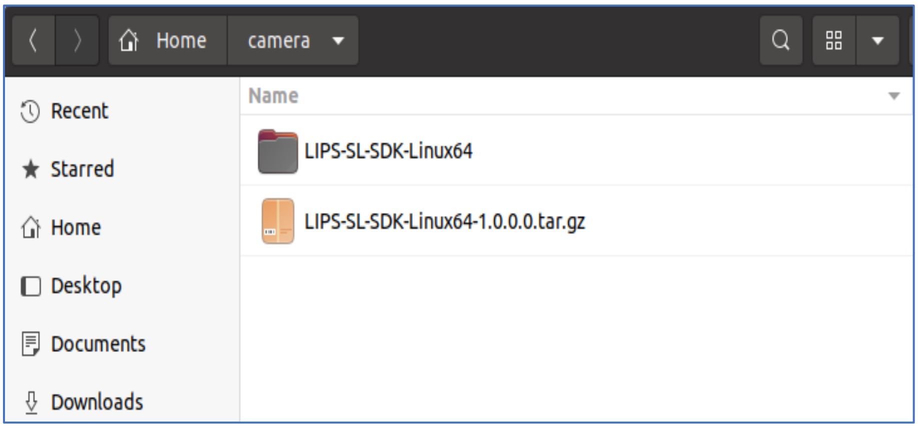
Name	Type
OPENNI2	File folder
Ni2GetSeiralNumber.exe	Application
Ni2GetSerialNumber.pdb	Program Debug Database
OpenNI.ini	Configuration settings
OpenNI2.dll	Application extension
OpenNI2.jni.dll	Application extension
OpenNI2.jni.pdb	Program Debug Database
OpenNI2.pdb	Program Debug Database
org.openni.jar	Executable Jar File

Linux

8. LIPS SDK Installation

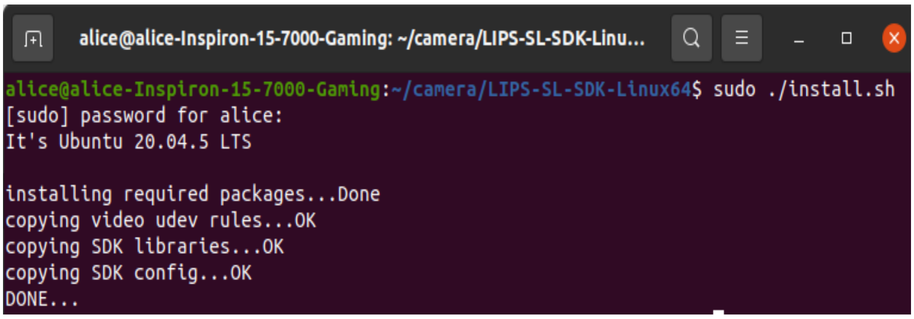
Refer to similar instructions for Windows to download the **LIPS SDK for Linux System** from our website.

1. After the download completes, unzip the **LIPS-SL-SDK-Linux64.**



2. Install the LIPS SDK.

```
sudo ./install.sh
```



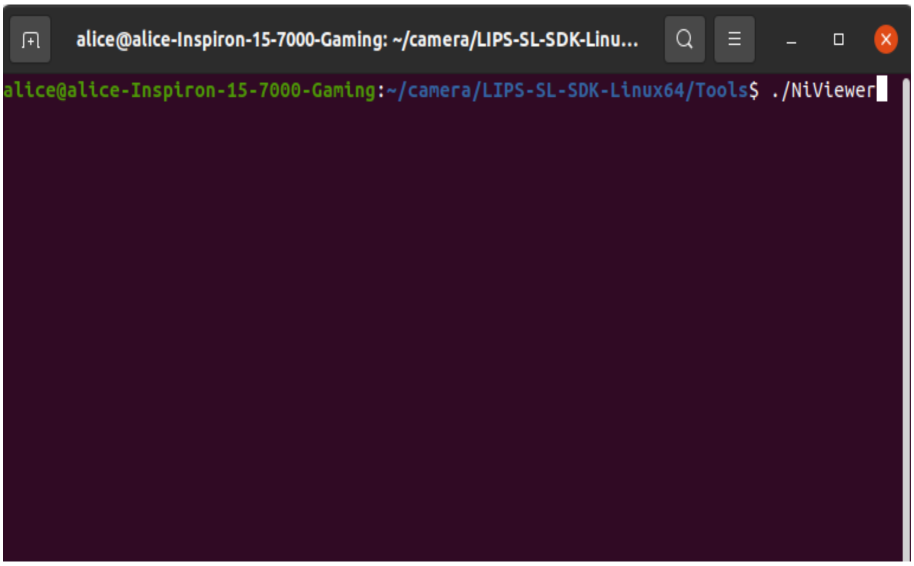
alice@alice-Inspiron-15-7000-Gaming: ~/camera/LIPS-SL-SDK-Linu...

```
alice@alice-Inspiron-15-7000-Gaming:~/camera/LIPS-SL-SDK-Linux64$ sudo ./install.sh
[sudo] password for alice:
It's Ubuntu 20.04.5 LTS

installing required packages...Done
copying video udev rules...OK
copying SDK libraries...OK
copying SDK config...OK
DONE...
```

9. NiViwer

Once the LIPS SDK is installed, you can access the camera image and status through NiViewer.

1. In the Terminal, go to the **Tools** folder. Type **sudo ./NiViewer**, then press **Enter**.



alice@alice-Inspiron-15-7000-Gaming: ~/camera/LIPS-SL-SDK-Linu...

```
alice@alice-Inspiron-15-7000-Gaming:~/camera/LIPS-SL-SDK-Linux64/Tools$ ./NiViewer
```

2. Upon successful activation, the **Depth & RGB image** will be displayed on screen.

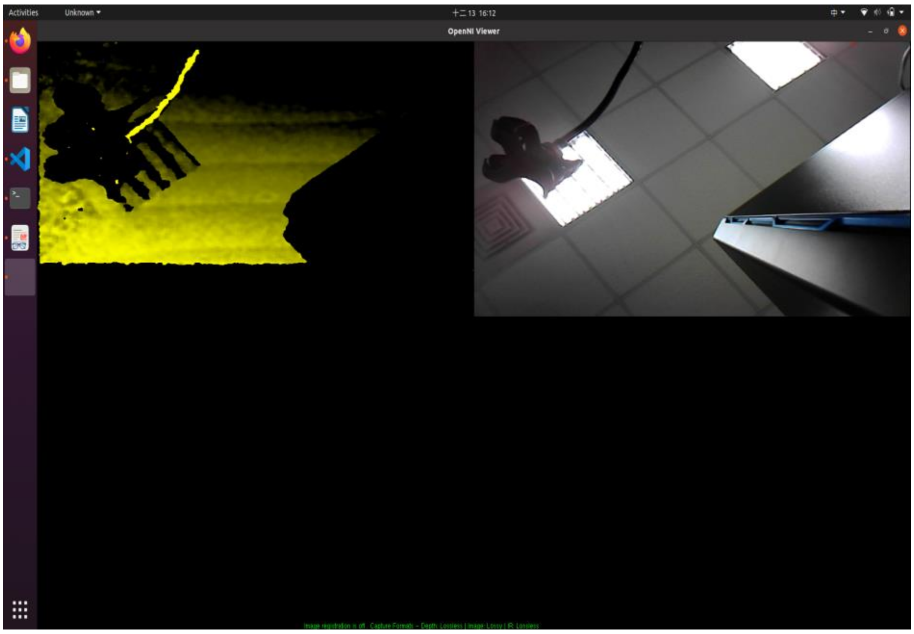
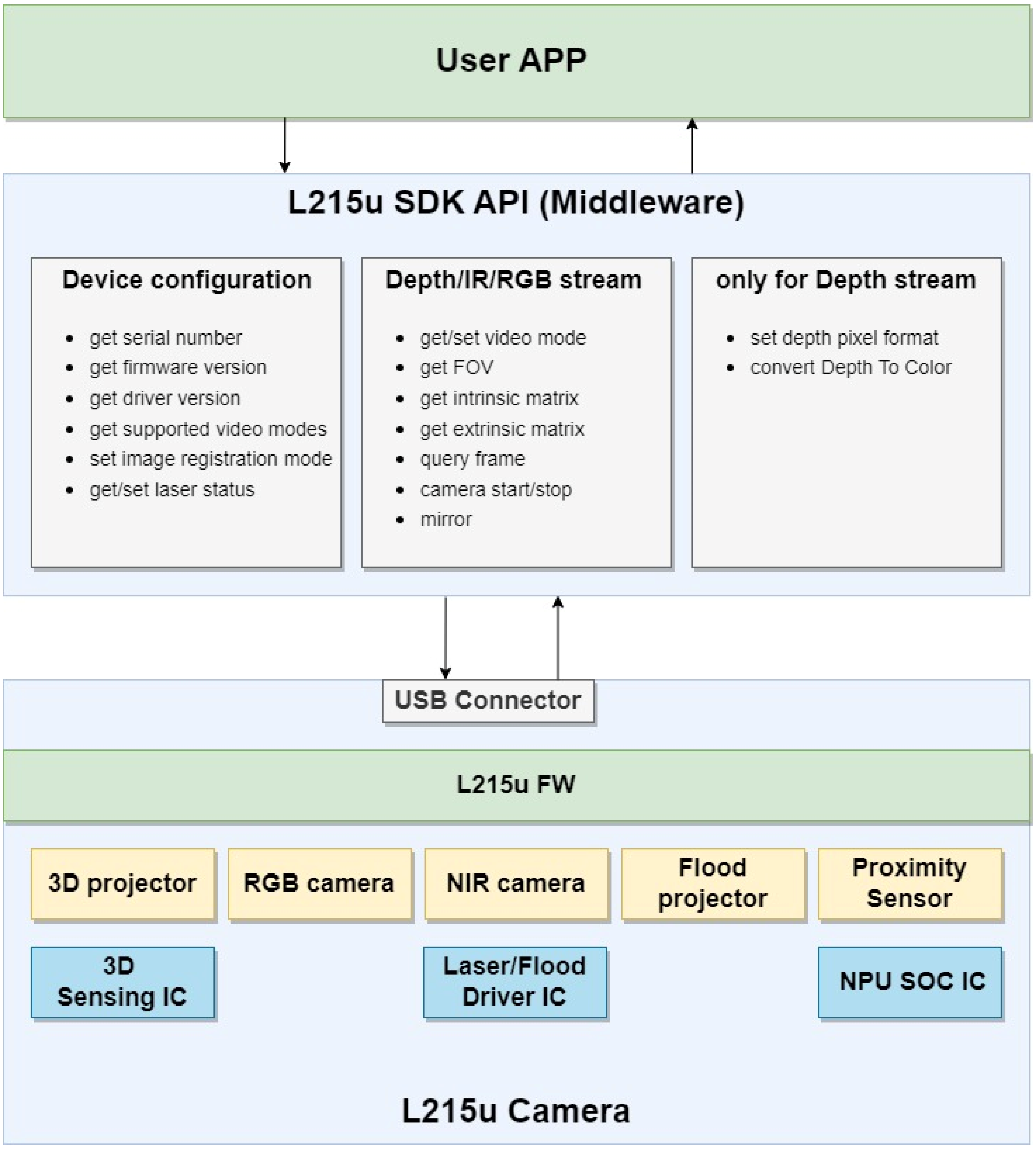


Image registration is off. Capture Formats – Depth: Lossless | Image: Lossy | IR: Lossless

More example applications are available at LIPS-SL-SDK-Linux64-1.0.0.0/LIPS-SL-SDK-Linux64/LIPS_Samples/Bin.

Appendix I APIs List for Programming

LIPS SDK supports OpenNI framework by which LIPSedge^{TM} L210u / L215u programs are based upon. The software architecture below exemplifies the process which the data are fetched.



Through OpenNI APIs, you can access basic data / functionalities necessary for programming purposes. LIPS Corp. also provides additional APIs for you to control illuminators (NIR and flood projectors) for power saving and thermal optimization

For basic OpenNI APIs, refer to OpenNI Programmer’s Guide.
For additional APIs, refer to the content below:

Additional API – illuminators control (OpenNI API extension)

Name	Language	Functions	System
Get laser status	C++	Get the status of the laser	Windows, Linux
Code			

openni::Status rc;
int mode;
rc =
devDevice.getProperty(LIPS_DEVICE_PROPERTY_LASER_ENABLE,
&mode);			

Name	**Language**	**Functions**	**System**
Set laser status	C++	Set to turn on/off the laser	Windows, Linux
Code			
rc = devDevice.setProperty(LIPS_DEVICE_PROPERTY_LASER_ENABLE, &mode, sizeof(int));			

For the functions to take effect, include the code into your project. Adjust the parameters according to your needs.

LIPS
Sense The World

LIPS CORPORATION

2F, No. 100, Ruiguag Road, Neihu District,
Taipei City 114, Taiwan
Tel.: + 886-2-8791-6998
Fax: +886-2-8791-8996
Official Website: https://www.lips-hci.com/
E-Mail: info@lips-hci.com