**Shining 3D EinStar SDK User Manual**

**V1.5.0**

**Shining 3D Tech Co., Ltd.**

|  |  |  |
| --- | --- | --- |
| Revision History | | |
| Version | Date | Revision Content |
| V1.0.0.0 | 2022-9-20 | First release. |
| V1.0.0.1 | 2023-3-6 | New sdk interface |
| V1.0.0.2 | 2023-3-13 | Add calibration tool |
| V1.0.0.3 | 2023-3-17 | Add mesh data interface |
| V1.0.0.4 | 2023-3-19 | Add device button event interface |
| V1.0.0.5 | 2023-3-30 | 1)Add get brightness interface.2)Add set work range interface.3)Repair get brightness range failed.4)Remove longpress\_plus and longpress\_sub funtion. |
| V1.0.0.6 | 20230713 | 1. Delete marker align type . 2. Delete auto align type. 3. Delete turtable align type. 4. Delete code points align type. 5. Add plane check interface. 6. Add get current marker interface. 7. Add get whole marker interface. |

[1 SDK Description 3](#_Toc606)

[1.1 Development Related Files 3](#_Toc23711)

[SDK List 3](#_Toc3785)

[Operation Requirements 3](#_Toc7284)

[1.2 Main Software Components 3](#_Toc4648)

[List of Documents 3](#_Toc29535)

[Operation Requirements 4](#_Toc28023)

[1.3 SDK Header File Description 4](#_Toc14314)

[1.4 Using the SDK Demo 4](#_Toc14474)

[1.5 SDK Known Issues 6](#_Toc14381)

[2 Data Structures and Macro Definitions 7](#_Toc4204)

[SN3D\_IMAGE\_DATA 7](#_Toc12074)

[SN3D\_POINT\_DATA 7](#_Toc5548)

[SN3D\_INCREASE\_POINT\_CLOUD 8](#_Toc11878)

[SN3D\_CAMERA\_POSITION 9](#_Toc550)

[SN3D\_TRI\_FACE 10](#_Toc25870)

[SN3D\_TRI\_MESH 11](#_Toc14860)

[SN3D\_CLOUD\_POINT 12](#_Toc14685)

[3 Return Value Definition 13](#_Toc6714)

[4 Alignment Type 14](#_Toc3620)

[5 Callback Function Declaration 15](#_Toc5654)

[Sn3DScanServiceWatcherCallBack 15](#_Toc7585)

[Sn3DWholePointCloudCallBack 16](#_Toc2445)

[Sn3DCurrentPointCloudCallBack 16](#_Toc23564)

[Sn3DIncreasePointCloudCallBack 16](#_Toc2036)

[Sn3DCameraPositionCallBack 17](#_Toc10111)

[Sn3DTrackLostStatusCallBack 17](#_Toc30407)

[Sn3DScanDistCallBack 18](#_Toc30754)

[6 Workflow 19](#_Toc31655)

[7 API Definition 20](#_Toc6856)

[7.1 Base Function Interface 20](#_Toc16133)

[Sn3DInitialize 20](#_Toc4446)

[Sn3DRelease 21](#_Toc20823)

[Sn3DInitialLog 22](#_Toc20437)

[Sn3DSaveData 23](#_Toc11983)

[7.2 Device Control Interface 24](#_Toc376)

[Sn3DCheckDevice 24](#_Toc32231)

[Sn3DGetBrightnessRange 25](#_Toc8647)

[Sn3DSetBrightness 26](#_Toc18249)

[Sn3DIntallGetImagasCallBack 27](#_Toc19791)

[Sn3DSetCurrentPointCloudCallBack 27](#_Toc6605)

[Sn3DSetIncreasePointCloudCallBack 27](#_Toc4627)

[Sn3DSetCameraPositionCallBack 28](#_Toc25517)

[Sn3DSetTrackLostStatusCallBack 28](#_Toc1163)

[Sn3DSetScanDistCallBack 28](#_Toc21025)

[7.3 The Function Interface of Scan 29](#_Toc17727)

[Sn3DEnterScan 29](#_Toc21240)

[Sn3DOpenOrCreateSolution 30](#_Toc17537)

[Sn3DNewProject 30](#_Toc28659)

[Sn3DOpenProject 31](#_Toc15590)

[Sn3DPreviewScan 32](#_Toc14820)

[Sn3DStartScan 33](#_Toc25176)

[Sn3DPauseScan 34](#_Toc27106)

[Sn3DEndScan 35](#_Toc29442)

[Sn3DCleanScan 36](#_Toc20759)

[Sn3DExitScan 37](#_Toc106)

[Sn3DSetEnablePseudoColor 38](#_Toc13290)

[7.4 Packaging Function Interface 39](#_Toc14955)

[Sn3DMesh 39](#_Toc21402)

# 1 SDK Description

In the manual, EinStar is referred to as “the Scanner”, the software client is referred to as “the Client” and “scanservice.exe” is referred to as “the Service.”

## **1.1 Development Related Files**

### **SDK List**

|  |  |
| --- | --- |
| Item | Description |
| Shining 3D EinStar SDK User Manual | Software interfaces description manual. |
| Include | Header files for ZMQ development;  API header files helping analyze shared memory. |
| Lib | Lib files For ZMQ development;  API lib file helping analyze shared memory. |
| Bin | Software execution package. |
| ScannerDemo | SDK demo software source code. |
| calibration\_tool | Calibration tool |

### **Operation Requirements**

1, Use vs2017+Qt5.12.8

2, The output of ScannerDemo.exe after compilation is in the bin file (the demo install will copy it).

3, Copy zmq\_sdk\_config.ini to the working directory and modify isRemote=1 (demo install will copy it).

4, Copy the sn3d\_smc.dll output in the bin file (the demo install will copy it).

5, Copy the sn3de10sdk.dll output in the bin file (the demo install will copy it).

## **1.2 Main Software Components**

### **List of Documents**

|  |  |  |
| --- | --- | --- |
| Item | Name | Description |
| Bin\zmq\_sdk\_config.ini | SDK configuration file | isRemote: 0 does not support SDK access  1 Support SDK access  Pub: public messaging port,  Default value 11398  Rep: Request messaging port  Default value 11399 |
| Bin\scanservice.exe | Scan service | Scans and processes data. |
| ScannerDemo.exe | Scan client demo | Call scanner service to scan. |
| Bin\sn3dE10SDK.dll | SDK link library |  |

### **Operation Requirements**

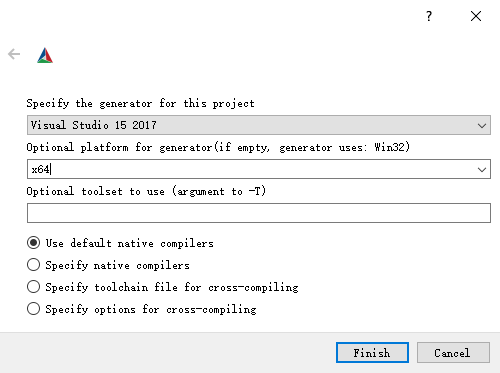
Modify interface in zmq\_sdk\_config.ini when communication port 11398 or 11399 is used by other software in your PC.

## **1.3 SDK Header File Description**

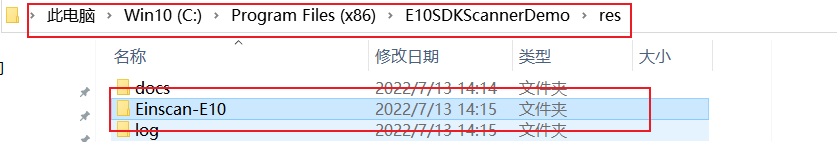
|  |  |
| --- | --- |
| **File Name** | **Description** |
| Include\\Sn3DE10Sdk.h | Contains sdk interfaces declaration |
| Include\\Sn3DErrorCode.h | Contains error codes information |
| Include\\Sn3DPublic.h | Custom structure included |
| Include\\PublicClass.h | Public definitions |

## **1.4 Using the SDK Demo**

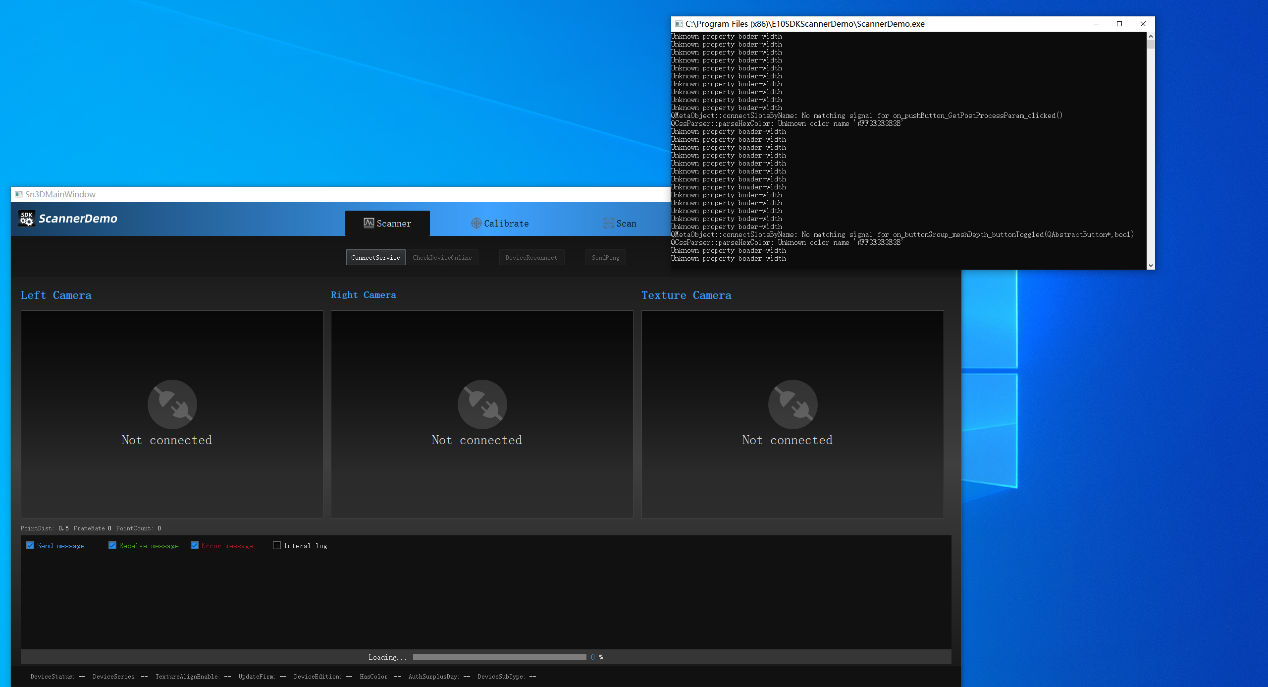
1, Install CMake3.17.2 and above, install QT version 5.12.8;

2, Build solutions using cmake, Visual Studio 15 2017 + QT 5.12.8;

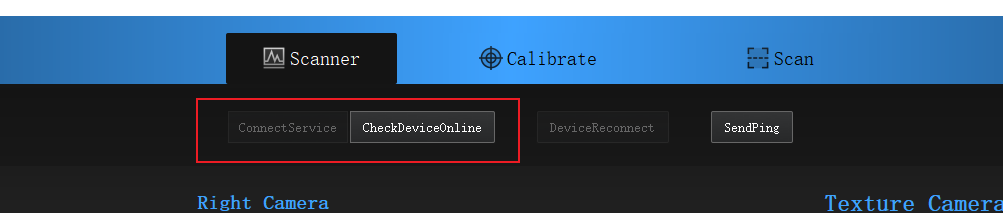
3, vs 2017 builds solutions under release x64

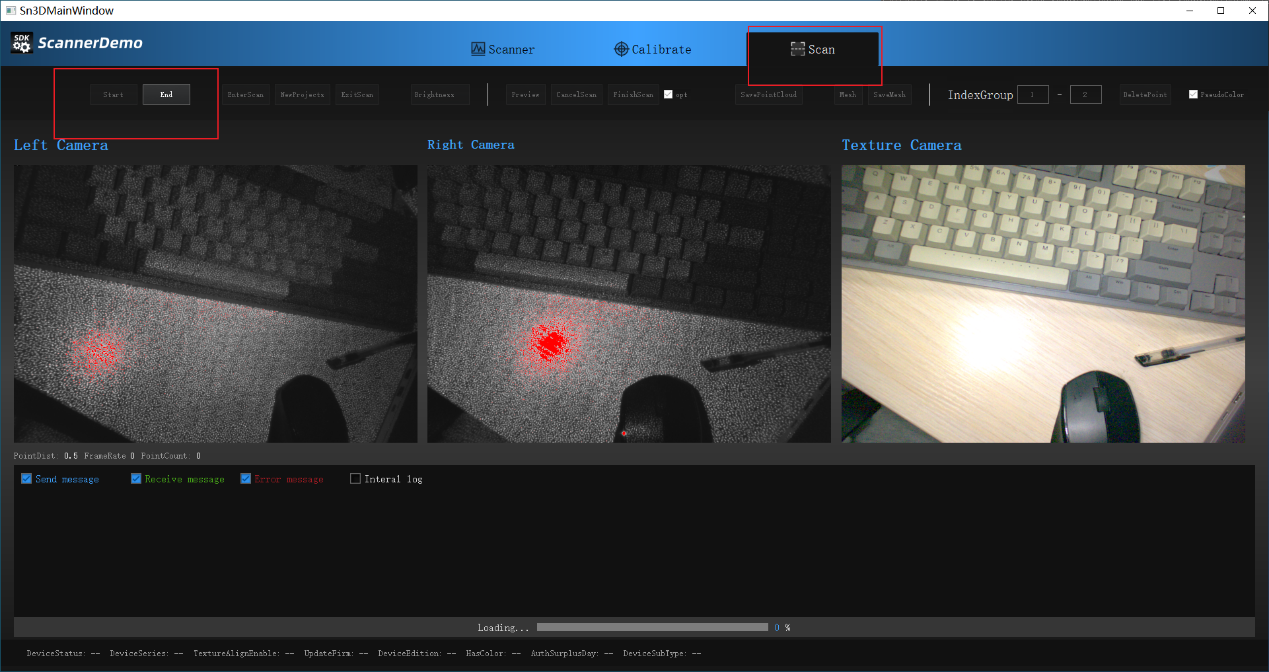
4, Copy ScannerDemo.exe, zmq\_sdk\_config.ini and all dynamic libraries under the folder bin to the root directory of the EinScan release software

5, Execute scannerDemo.exe under the root directory of release software.



6, Execute connectService (sdk initialization) and connectdevice (connection device) in turn.



7, Go to the scanner tab page, click start (start scan) and end (generate data).

## **Using the calibration tool**

1) Copy calibration\_tool.exe to the distribution software directory;

2) Connect the device, start it directly, and calibrate according to the prompt;

3) After the calibration calculation is completed, the calibration file will be generated in the directory res/Einscan-E10/200x150;

Note: When starting the calibration program, ensure that the ple file exists and the device is connected normally, otherwise it cannot be calibrated normally.

## **1.5 SDK Known Issues**

Garbled code for new construction projects under the Chinese path.

# 2 Data Structures and Macro Definitions

### SN3D\_IMAGE\_DATA

Data structure

Image data

|  |
| --- |
| typedef struct tag SN3D\_IMAGE\_DATA  {  int width;  int height;  int channel;  int length;  unsigned char\* data;  } SN3D\_ IMAGE\_ DATA, \*LPSN3D\_ IMAGE\_ DATA; |

Members

width

The width of the image;

height

The height of the image;

channel

The number of image channels;

length

image data length = width\*height;

data

Image data.

Remarks

### SN3D\_POINT\_DATA

Data structure

Point data

|  |
| --- |
| typedef struct  {  float x; //X coordinate value  float y;//Y coordinate value  float z;//Z coordinate value  }Sn3dPointData, \*LPSn3dPointData; |

Members

x

X coordinate.

y

Y coordinate.

z

Z coordinate.

Remarks

### SN3D\_INCREASE\_POINT\_CLOUD

Data structure

Point data

|  |
| --- |
| typedef struct  {  int vertex\_count; //The number of vertex .  LPSn3dPointData vertex\_data; //The data of vertex.  int norma\_count; //The number of vertex normal.  LPSn3dPointData norma\_data; //The data of vertex normal.  int vertex\_color\_count; //The number of vertex color.  LPSn3dPointData vertex\_color\_data; //The data of vertex color.  int index\_count; //The number of index  unsigned int\* index; //The data of index  }Sn3DIncreasePointCloud,\*LPSn3dIncreasePointCloud; |

**Members**

vertex\_count

Vertices number

vertex\_data

Vertices data

norma\_count

Normal number

norma\_data

Normal data

vertex\_ color \_count

Vertices color number

vertex\_color\_data

Vertices color data

index\_count

The index number of point

index

The index data of point

**Remarks**

### SN3D\_CAMERA\_POSITION

|  |
| --- |
| typedef struct  {  QVector3D position;  QVector3D center;  QVector3D up;  }Sn3DCameraPosition,\*LPSn3dCameraPosition; |

**Members**

position

Camera Position

center

Center of view

up

Normal of up

**Remarks**

### SN3D\_FACE\_ID

Triangular patch information

|  |
| --- |
| typedef struct  {  int x;  int y;  int z;  }Sn3DFaceId, \*LPSn3dFaceId; |

**Members**

x, y, z: index of face

**Remarks**

### SN3D\_VEC2F

|  |
| --- |
| typedef struct  {  float x;  float y;  }Sn3DVec2F, \*LPSn3dVec2F; |

**Members**

x, y: coordinate of texture

**Remarks**

### SN3D\_IMAGE

|  |
| --- |
| typedef struct  {  int width;  int height;  int channel;  uchar\* data;  //RGB  }Sn3DImage; |

**Members**

width: width of image

height: height of image

channel: channel of image

data: data of image

**Remarks**

### SN3D\_MESHDATA

|  |
| --- |
| typedef struct  {  int meshpoint\_count;  LPSn3dPointData meshpoint;  int meshnormal\_count;  LPSn3dPointData meshnormal;  int meshtrifaceid\_count;  LPSn3dFaceId meshtrifaceid;  int meshtextureid\_count;  LPSn3dId meshtextureid;  int textureUV\_count;  LPSn3dVec2F textureUV;  Sn3DImage image;  }Sn3DMeshData, \*LPSn3dMeshData |

**Members**

meshpoint\_coun: num of point cloud

meshpoint: data of point cloud

meshnormal\_count: num of normal

meshnormal: data of normal

meshtrifaceid\_count: num of point index

meshtrifaceid: data of point index

meshtextureid\_count: num of textureimage

meshtextureid: data of texture image

textureUV\_count: num of uv

textureUV:data of uv

image: texture image

**Remarks**

### SN3D\_CLOUD\_POINT

Data structure

Point cloud data

|  |
| --- |
| typedef struct  {  int vertex\_count; //The number of vertex .  LPSn3dPointData vertex\_data; //The data of vertex.  int norma\_count; //The number of vertex normal.  LPSn3dPointData norma\_data; //The data of vertex normal.  int vertex\_color\_count; //The number of vertex color.  LPSn3dPointData vertex\_color\_data; //The data of vertex color.  }Sn3DPointCloud,\*LPSn3dPointCloud; |

Members

vertex\_count

Number of vertex data.

vertex\_data

Vertex data.

norma\_count

Number of vertex normal data.

norma\_data

Vertex normal data.

vertex\_ color \_count

The number of vertex color data.

vertex\_color\_data

Vertex color data.

Remarks

### SN3D\_MARKER\_DATA

|  |
| --- |
| struct MarkerInfo  {  Sn3dPointData coord; // data of marker  Sn3dPointData normal; // data of noraml  double diameter; // diameter  int codeId; //marker id (<=0 means common marker)  };  typedef struct {  int marker\_count; //marker count  MarkerInfo\* markerList; //marker list  }Sn3DMarkerData, \*LPSn3DMakerData; |

Members

Information of single marker：

coord: marker data.

normal: marker normal data.

diameter: marker diameter.

codeId: marker id (<=0 means common marker).

Information of marker list:

marker\_count: number of markers.

markerList: list of markers.

Remarks

# 3 Return Value Definition

|  |  |  |
| --- | --- | --- |
| **Macro definition** | **Value** | **Description** |
| **EC\_SUCCESS** | 0 | No error |
| **EC\_NOTINITIALIZED** | 1 | No initialization |
| **EC\_INITIALIZEFIAILZED** | 2 | Initialization failed. |
| **EC\_AlREADYINITIALIZED** | 3 | Initialization repeat. |
| **EC\_SAVEFAILED\_TYPEERROR** | 4 | Format error，failed to save files. |
| **EC\_CHECKDEVICEFAILED\_PLENOTRIGHT** | 5 | PLE error, failed to connect device. |
| **EC\_CHECKDEVICEFAILED\_NODEVICEFOUND** | 6 | Device no found, connection failed. |
| **EC\_OPENORCREATSLNFAILED** | 7 | Failed to create or open project. |
| **EC\_ENTERSCANFAILED** | 8 | Failed to enter scan mode. |
| **EC\_SCANFAILED** | 9 | Scanning failed. |
| **EC\_ENDSCANFAILED** | 10 | Generating failed. |
| **EC\_MESHFAILED** | 11 | Meshing failed. |
| **EC\_EXITSCANFAILED** | 12 | Quitting scan mode failed. |
| **EC\_CANCELSCANFAILED** | 13 | Canceling scan failed. |
| **EC\_SAVEFAILED** | 14 | Saving failed. |
| **EC\_CREATNEWPROJECTFAILED** | 16 | New project failed. |
| **EC\_WORKRANGE** | 17 | Set faied |
| **EC\_PROCESSERROR** | 268435456 | Call process error 0x10000000 |

# 4 Alignment Type

|  |  |  |
| --- | --- | --- |
| **Type definition** | **Value** | **Description** |
| **AT\_FEATURES** | 0 | Feature alignment |
| **AT\_HYBRID** | 2 | Hybrid alignment |
| **AT\_GLOBLE\_POINT** | 7 | Global markers alignment |
| **AT\_TEXTURE** | 8 | Texture alignment |

# 5 DeviceEvent Type

|  |  |  |
| --- | --- | --- |
| **枚举** | **值** | **含义** |
| **DE\_NULL** | -1 | NULL |
| **DE\_DOUBLECLICK** | 0 | Double Click |
| **DE\_CLICK** | 1 | Click |
| **DE\_PLUS** | 2 | Plus button click |
| **DE\_SUB** | 3 | Sub button click |
| **DE\_LONGPRESS** | 8 | Ok button long press |

# 5 Callback Function Declaration

### Sn3DScanServiceWatcherCallBack

Initialize the callback

Initialize the device setup callback.

|  |
| --- |
| **typedef void (CALLBACK \*Sn3DScanServiceWatcherCallBack)(int);** |

Remarks

Register the server monitors the callback and the server is notified of the exit.

### Sn3DWholePointCloudCallBack

Point cloud callback

Initialize the device setup callback

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DWholePointCloudCallBack)(LPSn3dPointCloud wholePointCloud);** |

Parameter

**wholePointCloud**

**[out] point cloud**

Remarks

1, Setting sn3dEndscan (end scan) or sn3dMesh (meshing) callback can get point cloud set at different stages.

2, The callback needs to be reset after setting a callback is successfully triggered.

### Sn3DCurrentPointCloudCallBack

Get current point cloud callback

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DCurrentPointCloudCallBack)(LPSn3dPointCloud currentPointCloud, void\* owner);** |

**Parameter**

**currentPointCloud**

**[out] current point cloud**

**owner**

**[out] the owner of callback function**

**Remarks**

1.Setting callback after Sn3DInitialize(initial) will get the point convergence of the current frame throughout the scan.

### Sn3DIncreasePointCloudCallBack

Get increase point cloud callback funtion

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DIncreasePointCloudCallBack)(LPSn3dIncreasePointCloud increasePointCloud, void\* owner);** |

**Parameter**

**increasePointCloud**

**[out] increase point cloud data**

**owner**

**[out] the owner of callback function**

**Remarks**

1.Setting callback after Sn3DInitialize(initial) will get the point convergence of the increase point throughout the scan。

### Sn3DCameraPositionCallBack

Get Camera Position callback function

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DCameraPositionCallBack)(LPSn3dCameraPosition cameraPosition, void\* owner);** |

**Parameter**

**cameraPosition**

**[out] The data of camrea position**

**owner**

**[out] the owner of callback function**

**Remarks**

1.Setting callback after Sn3DInitialize(initial) will get the point convergence of the camera position throughout the scan

### Sn3DTrackLostStatusCallBack

Get track lost status callback funtion

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DTrackLostStatusCallBack)(bool trackLostStatus, void\* owner);** |

**Parameter**

**trackLostStatus**

**[out] The status of track，true: Track lost；false: track success**

**owner**

**[out]** **the owner of callback function**

**Remarks**

1.Setting callback after Sn3DInitialize(initial) will get the point convergence of the track lost status throughout the scan

### Sn3DScanDistCallBack

Get scan distance gear callback funtion

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DScanDistCallBack)(double scanDist, void\* owner);** |

**Parameter**

**scanDist**

**[out] The distance gear of scan**

**owner**

**[out] the owner of callback function**

**Remarks**

1.Setting callback after Sn3DInitialize(initial) will get the point convergence of the distance gear throughout the scan

2.scanDist = -1:too near；scanDist = 100:too far；scanDist=-2，invalid；other (1~12)。

### Sn3DMeshDataCallBack

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DMeshDataCallBack)(LPSn3dMeshData meshData, void\* owner);** |

**Parameter**

**meshData**

**[out] mesh of data**

**owner**

**[out] the owner of callback function**

**Remarks**

.Setting callback after Sn3DEndScan will get the point convergence of the track lost status throughout the scan

### Sn3DDeviceEventCallBack

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DDeviceEventCallBack)(DeviceEvent event, void\* owner);** |

**Parameter**

**event**

**[out] type of button click**

**owner**

**[out] the owner of callback function**

**Remarks**

1. Setting callback after Sn3DInitialize(initial).

### Sn3DTooFlatStatusCallBack

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DTooFlatStatusCallBack)(bool tooFlatStatus, void\* owner);** |

**Parameter**

**tooFlatStatus**

**[out] check plane. The output will be true when object is plane.**

**owner**

**[out] the owner of callback function**

**Remarks**

### Sn3DCurrentMarkerCallBack

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DCurrentMarkerCallBack)(LPSn3DMakerData currentMarker, void\* owner);** |

**Parameter**

**currentMarker**

**[out] current markers list.**

**owner**

**[out] the owner of callback function**

**Remarks**

### Sn3DWholeMarkerCallBack

|  |
| --- |
| **typedef void(CALLBACK \*Sn3DWholeMarkerCallBack)(LPSn3DMakerData wholeMarker, void\* owner);** |

**Parameter**

**wholeMarker**

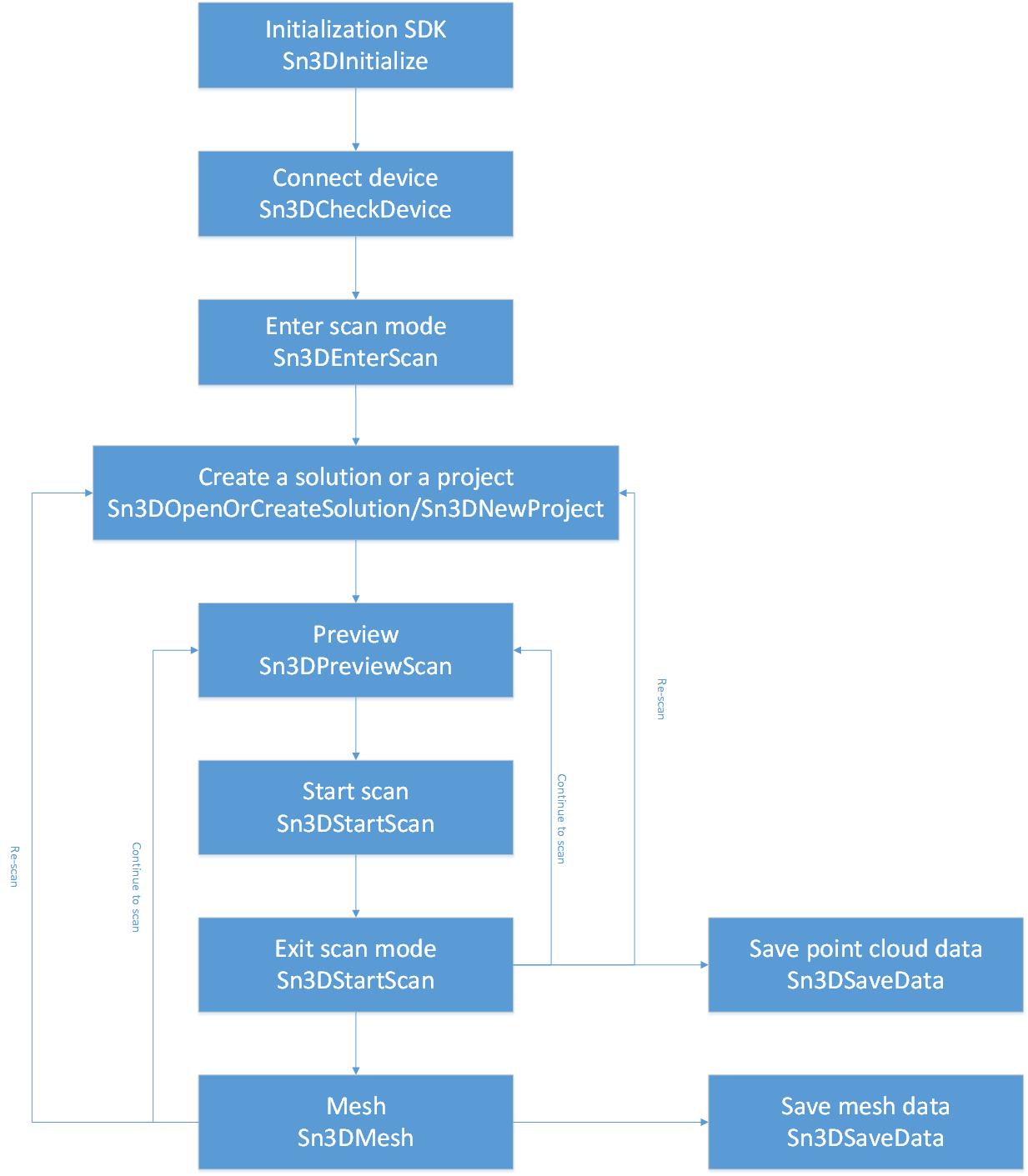
**[out] whole markers list.**

**owner**

**[out] the owner of callback function**

**Remarks**

# 6 Workflow



# 7 API Definition

## **7.1 Base Function Interface**

### Sn3DInitialize

Initialization SDK

|  |
| --- |
| **void\* Sn3DInitialize**(  Sn3DScanServiceWatctherCallBack watcher  ); |

Parameters

Sn3DScanServiceWatctherCallBack

[in] Registration service monitors the callback and receives notice when the server ended.

Return Values

For macro definition, refer to return values.

Remarks

1, Just initialize SDK once.

2, After initializing SDK, you can call Sn3DRelease to release.

### Sn3DRelease

Release SDK

|  |
| --- |
| **int Sn3DRelease**(); |

Return Values

For macro definition, refer to return values.

Remarks

The function for release the SDK resource.

### Sn3DInitialLog

Set up the SDK output log.

|  |
| --- |
| **int Sn3DInitialLog** (  char\* exeName,  char\* path  ); |

Parameters

exeName

[in] Call name of SDK proceeding.

path

[in] Set up directory of log in.

Return Values

Refer to error code.

### Sn3DSaveData

Save data

|  |
| --- |
| **int Sn3DSaveData** (  **char\* absolutePrjName,**  **char\*savePath,**  **char\*saveType**  ); |

Parameters

**absolutePrjName**

[in] Complete path of project files.

**savePath**

[in] Path and name of the file required to be saved (without a suffix).

**saveType**

[in] The format of the file required to be saved (asc ply stl obj).

Return Values

Refer to error code.

## **7.2 Device Control Interface**

### Sn3DCheckDevice

Connection device.

|  |
| --- |
| **int Sn3DConnectDevice** () |

Return Values

For macro definition, refer to return values.

### Sn3DGetBrightnessRange

The brightness range

|  |
| --- |
| **SN3DSDKE10API int Sn3DGetBrightnessRange(int& min, int& max);** |

Parameters

min

[out] Reserved, the minimum brightness level is set to 0 (default). By default, this function is disable.

[out] Maximum level of brightness.

Return Values

Refer to error code for details.

Remarks

### Sn3DSetBrightness

Set the brightness

|  |
| --- |
| **int Sn3DSetBrightness** (  int brightness  ) |

Parameters

brightness

[in] Brightness level, the value range is determined by the value returned by Sn3DGetBrightnessRange,

the larger the value, the brighter.

Return Values

Refer to error code.

Remarks

### Sn3DIntallGetImagasCallBack

Get video stream

|  |
| --- |
| **int Sn3DIntallGetImagasCallBack** (  **Sn3DGetImagesCallBack imageCallback,**  **void\* owner,**  ); |

Parameters

**imageCallback**

[in] Set the callback function.

**owner**

[in] Pointer to the owner of the callback function.

### Sn3DSetCurrentPointCloudCallBack

|  |
| --- |
| **int Sn3DSetCurrentPointCloudCallBack** (  **Sn3DCurrentPointCloudCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

**CallbackFunc**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetIncreasePointCloudCallBack

|  |
| --- |
| **int Sn3DSetIncreasePointCloudCallBack**(  **Sn3DIncreasePointCloudCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

**CallbackFunc**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetCameraPositionCallBack

|  |
| --- |
| **int Sn3DSetCameraPositionCallBack**(  **Sn3DCameraPositionCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetTrackLostStatusCallBack

|  |
| --- |
| **int Sn3DSetTrackLostStatusCallBack**(  **Sn3DTrackLostStatusCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetScanDistCallBack

|  |
| --- |
| **int Sn3DSetScanDistCallBack**(  **Sn3DScanDistCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetDeviceEventCallBack

|  |
| --- |
| **int Sn3DSetDeviceEventCallBack**(  **Sn3DDeviceEventCallBack callBackFunc,**  **void\* owner,**  ); |

**Parameters**

[in] Set callback funtion

**owner**

[in] The owner of funciton

### Sn3DSetWorkRange

|  |
| --- |
| **int Sn3DSetWorkRange(float min, float max);** |

**Parameters**

**min**

[in] The minimum working distance，The defaut value is 160

**max**

[in] The maximum working distance，The defaut value is 600

**Return Values**

Error code

**Remarks**

**Use this function after enter scan and before exit scan.**

### Sn3DGetCurrentBrightness

|  |
| --- |
| **int Sn3DGetCurrentBrightness(int& currentBrightness)** |

**Parameters**

**currentBrightness**

[out] Current brightness value

**Return Values**

Error code

**Remarks**

### Sn3DSetTooFlatStatusCallBack

|  |
| --- |
| **int Sn3DSetTooFlatStatusCallBack(Sn3DTooFlatStatusCallBack callBackFunc, void\* owner)；** |

**Parameters**

**CallbackFunc**

[in] Set callback function

**owner**

[in] The owner of function

### Sn3DSetCurrentMarkerCallBack

|  |
| --- |
| **int Sn3DSetCurrentMarkerCallBack(Sn3DCurrentMarkerCallBack callBackFunc, void\* owner)；** |

**Parameters**

**CallbackFunc**

[in] Set callback function

**owner**

[in] The owner of function

### Sn3DSetWholeMarkerCallBack

|  |
| --- |
| **int Sn3DSetWholeMarkerCallBack(Sn3DWholeMarkerCallBack callBackFunc, void\* owner)；** |

**Parameters**

**CallbackFunc**

[in] Set callback function

**owner**

[in] The owner of function

## **7.3 The Function Interface of Scan**

### Sn3DEnterScan

Enter scan

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DEnterScan();** |

Parameters

Return Values

Error code, refer to macro definition of return values.

Remarks

1, This function can only be called after initializing the detection device online

2, Perform the function of initializing scan.

### Sn3DOpenOrCreateSolution

New solutions

|  |
| --- |
| **SN3DSDKE10API int Sn3DOpenOrCreateSolution(**  **const char\* slnDirPath,**  **bool isCreate,**  **int scanMode,**  **bool hasTexture,**  **float pointDis);** |

Parameters

slnDirPath

[in] Solution path

isCreate

[in] “true”- new, “false”-open

scanMode

[in] Scan mode, 0 portrait, 1 object, 2 smallobject. Reference: ScanMode

hasTexture

[in] “true”- texture, “false”-no texture

pointDis

[in] Point distance

Return Values

Error code refer to macro definition of return values.

Remarks

Call Sn3DOpenOrCreateSolution after Sn3DEnterScan.

### Sn3DNewProject

New project

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DNewProject(**  **const char\* slnDirPath,**  **int scanMode,**  **bool hasTexture,**  **float pointDis,**  **int alignType,**  **const char\* globalMarkerPath);** |

Parameters

slnDirPath

[in] Solution Path

scanMode

[in] Scan mode, 0 portrait, 1 object, 2 smallobject. Reference: ScanMode

hasTexture

[in] “true”- texture, “false”-no texture

pointDis

[in] Point distance

alignType

[in] Alignment type. Reference: AlignType See appendix

globalMarkerPath

[in] Global markers path，no global markers path：set scan mode to empty.

Return Values

Error code refer to macro definition of return values.

Remarks

Call this function after call Sn3DOpenOrCreateSolution.

### Sn3DOpenProject

Open project

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DOpenProject(**  **const char\* projFileName);** |

Parameters

projFileName

[in] Project path

Return Values

Error code refer to macro definition of return values.

Remarks

Call this function after call Sn3DOpenOrCreateSolution

### Sn3DPreviewScan

Preview

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DPreviewScan();** |

Parameters

Return Values

Error code refer to macro definition of return values.

Remarks

1, Preview only rebuild the data not save the scan data to overall point cloud.

2, Call this function after you create or open project.

### Sn3DStartScan

Scan

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DStartScan();** |

Parameters

Return Values

Error code refer to macro definition of return values.

Remarks

1, Call this function after Sn3DPreviewScan.

2, Tracking, and fusion to the overall point cloud.

### Sn3DPauseScan

Pause Scan

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DPauseScan ();** |

Parameters

Return Values

Error code refer to macro definition of return values.

Remarks

Call this function after call Sn3DStartScan.

### Sn3DEndScan

End scan

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DEndScan(**  **bool globalOptimize,**  **double pointDist,**  **Sn3DWholePointCloudCallBack callback = nullptr);** |

Parameters

globalOptimize

[in] “true”-optimize , “false” -not optimized

pointDist

[in]setting point distance.

callback

[in]Set up the callback to get the point cloud data after finishing the scan.

Return Values

Error code refer to macro definition of return values.

Remarks

1, You can call Sn3DEndScan after Sn3DStartScan or Sn3DPauseScan.

2, Generate the overall point cloud data;

3, If you set the callback to get that stage to generate point cloud data, there are examples in demo.

### Sn3DCleanScan

Empty the scan data

|  |
| --- |
| **SN3DSDKE10API int Sn3DCancelScan(**  **bool isCancelCurrentProjectFramerMark);** |

Parameters

isCancelCurrentProjectFramerMark

[in] “true”- Empty the global markers information.

Return Values

Error code refer to macro definition of return values.

Remarks

Empty the currently scanned point cloud data.

### Sn3DExitScan

Quit scan mode

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DExitScan();** |

Parameters

Return Values

Error code refer to macro definition of return values.

Remarks

Quit scan mode.

### Sn3DSetEnablePseudoColor

Turn on pseudo function

|  |
| --- |
| **int Sn3DSetEnablePseudoColor**(  bool enable  ); |

Parameters

enable

[in] “true”-open, “false”-off

Return Values

Refer to error code.

## **7.4 Packaging Function Interface**

### Sn3DMesh

Generate mesh data

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DMesh(**  **int meshType,**  **int filterLevel,**  **int SmoothLevel,**  **float PointDis,**  **bool fillSmallHole,**  **double smellHolePerimeter,**  **bool fillMarkerHole,**  **double spikeSensitivity,**  **int faceLimit,**  **Sn3DWholePointCloudCallBack callback = nullptr);** |

Parameters

meshType.

[in] Reference: MeshType

filterLevel.

[in] 0-3 level, different levels of filtering degree.

SmoothLevel.

[in] 1-3 level, Setting up smooth level in filter.

PointDis.

[in]point distance.

fillSmallHole.

[in] “true”-Fill the small hole

smellHolePerimeter.

[in]Fill the circumference of the small hole

fillMarkerHole.

[in] “true”-filling markers.

spikeSensitivity.

[in] Remove the spikes

faceLimit.

[in] Maximum number of patch.

callback.

[in] setting up callback for getting point cloud data of meshed.

Return Values

Refer to error code.

Remarks

1, Calling Sn3DEndScan after ending scan.

2, Meshing data.

3, Setting up callback for getting overall point cloud set of meshed.

### Sn3DMeshEx

|  |
| --- |
| **SN3DSDKE10API int CALLMETHOD Sn3DMesh(**  **int meshType,**  **int filterLevel,**  **int SmoothLevel,**  **float PointDis,**  **bool fillSmallHole,**  **double smellHolePerimeter,**  **bool fillMarkerHole,**  **double spikeSensitivity,**  **int faceLimit,**  **void\* owner**  **Sn3DMeshDataCallBack callback = nullptr);** |

Parameters

meshType.

[in] Reference: MeshType

filterLevel.

[in] 0-3 level, different levels of filtering degree.

SmoothLevel.

[in] 1-3 level, Setting up smooth level in filter.

PointDis.

[in]point distance.

fillSmallHole.

[in] “true”-Fill the small hole

smellHolePerimeter.

[in]Fill the circumference of the small hole

fillMarkerHole.

[in] “true”-filling markers.

spikeSensitivity.

[in] Remove the spikes

faceLimit.

[in] Maximum number of patch.

Owner

[in] owner of callback

callback.

[in] setting up callback for getting mesh data of meshed.

Return Values

Refer to error code.

Remarks

1, Calling Sn3DEndScan after ending scan.

2, Meshing data.

3, Setting up callback for getting overall point cloud set of meshed.