

Python Interface Development User Manual



Version: V3.0

Date: 2025-7-25

Author: Haitang

Content

安装方式	1
class SPARROW	1
1.captureData	1
2.connect	1
3.disconnect	1
4.getBrightnessData	2
5.getCalibrationParam	2
6.getCameraChannels	2
7.getCameraResolution	2
8.getColorBrightnessData	2
9.getDepthData	3
10.getHeightMapDataBaseParam	3
11.getPointcloudData	3
12.getUndistortBrightnessData	3
13.getUndistortColorBrightnessData	3
14.getUndistortDepthData	4
15.setCaptureEngine	4
16.setParamBrightnessExposureModel	4
17.setParamBrightnessGain	4
18.setParamBrightnessHdrExposure	4
19.setParamCameraConfidence	5
20.setParamCameraExposure	5
21.setParamCameraGain	5
22.setParamDepthFilter	5
23.setParamGenerateBrightness	6
24.setParamGrayRectify	6
25.setParamLedCurrent	6
26.setParamMixedHdr	6
27.setParamMultipleExposureModel	7
28.setParamOutlierFilter	7
29.setParamRadiusFilter	7
30.setParamReflectFilter	7
31.setParamRepetitionExposureNum	7
32.setParamSmoothing	8
33.captureBrightnessData	8
class CalibrationParam	8
class Color	8
class Engine	9
Functions	9
1.get_distortion	9
2.get_extrinsic	9
3.get_intrinsic	9

本文档介绍了 SPARROW 相机的 Python SDK 接口，包含主要类” SPARROW” 的使用方法及提供的功能接口说明，旨在帮助开发者快速接入并控制相机，获取图像数据与标定信息等。

安装方式

如需要 Python3.10 版本可在 lib 包中通过 pip 进行安装

安装：python -m pip install sparrow-3.0-cp310-cp310-win_amd64.whl

其他版本安装同上。

使用：

```
from sparrow import sparrow
```

```
camera=sparrow.SPARROW()
```

```
camera.connect("127.0.0.1")
```

详情请查看 example.py 例程

class SPARROW

1.captureData

```
def captureData(self, arg0, arg1)
```

//功能： 采集一帧数据并阻塞至返回状态

//输入参数： exposure_num（曝光次数）： 设置值为 1 为单曝光，大于 1 为多曝光模式（具体参数在相机 gui 中设置）。

//输出参数： timestamp(时间戳)

//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.

2.connect

```
def connect(self, arg0)
```

//功能： 连接相机

//输入参数： camera_id（相机 ip 地址）

//输出参数： 无

//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.

3.disconnect

```
def disconnect(self, arg0)
```

//功能： 断开相机连接

//输入参数： camera_id（相机 ip 地址）

//输出参数： 无

//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.

4.getBrightnessData

```
def getBrightnessData(self, arg0, numpy_uint8=None)
//功能： 获取亮度图
//输入参数： 无
//输出参数： brightness(亮度图)
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

5.getCalibrationParam

```
def getCalibrationParam(self, arg0)
//功能： 获取相机标定参数
//输入参数： 无
//输出参数： calibration_param（相机标定参数结构体）
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

6.getCameraChannels

```
def getCameraChannels(self, arg0, numpy_int32=None)
//功能： 获取相机图像通道数
//输入参数： 无
//输出参数： channels(通道数)
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

7.getCameraResolution

```
def getCameraResolution(self, arg0, numpy_int32=None, *args, **kwargs)
//功能： 获取相机分辨率
//输入参数： 无
//输出参数： width(图像宽)、height(图像高)
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

8.getColorBrightnessData

```
def getColorBrightnessData(self, arg0, numpy_uint8=None, *args, **kwargs)
//功能： 获取彩色亮度图
//输入参数： 无
//输出参数： brightness(亮度图), color(亮度图颜色类型)
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

9.getDepthData

```
def getDepthData(self, arg0, numpy_float32=None)
//功能： 获取深度图
//输入参数： 无
//输出参数： depth(深度图)
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

10.getHeightMapDataBaseParam

```
def getHeightMapDataBaseParam(self, arg0, numpy_float32=None, *args, **kwargs)
//功能： 获取校正到基准平面的高度映射图
//输入参数： R(旋转矩阵)、T(平移矩阵)
//输出参数： height_map(高度映射图)
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

11.getPointcloudData

```
def getPointcloudData(self, arg0, numpy_float32=None)
//功能： 获取点云
//输入参数： 无
//输出参数： point_cloud(点云)
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

12.getUndistortBrightnessData

```
def getUndistortBrightnessData(self, arg0, numpy_uint8=None)
//功能： 获取去畸变亮度图
//输入参数： 无
//输出参数： undistort_brightness(亮度图)
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

13.getUndistortColorBrightnessData

```
def getUndistortColorBrightnessData(self, arg0, numpy_uint8=None, *args, **kwargs)
//功能： 获取去畸变后的彩色亮度图
//输入参数： 无
//输出参数： brightness(亮度图)
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

14.getUndistortDepthData

```
def getUndistortDepthData(self, arg0, numpy_float32=None)
//功能： 获取去畸变深度图
//输入参数： 无
//输出参数： undistort_depth(深度图)
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

15.setCaptureEngine

```
def setCaptureEngine(self, arg0)
//功能： 设置采集引擎
//输入参数： engine
//输出参数： 无
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

16.setParamBrightnessExposureModel

```
def setParamBrightnessExposureModel(self, arg0)
//功能： 设置亮度图曝光模式
//输入参数： model（1：单曝光、2：曝光融合）
//输出参数： 无
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

17.setParamBrightnessGain

```
def setParamBrightnessGain(self, arg0)
//功能： 设置亮度图增益
//输入参数： gain(亮度图增益)
//输出参数： 无
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

18.setParamBrightnessHdrExposure

```
def setParamBrightnessHdrExposure(self, arg0, arg1, numpy_int32=None)
//功能： 设置亮度图多曝光参数（最大曝光次数为 10 次）
//输入参数： num（曝光次数）、exposure_param[6]（6 个曝光参数、前 num 个有效）
//输出参数： 无
//返回值： 类型（int）:返回 0 表示连接成功;返回-1 表示连接失败.
```

19.setParamCameraConfidence

```
def setParamCameraConfidence(self, arg0)
//功能： 设置相机置信度
//输入参数： confidence(相机置信度)
//输出参数： 无
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

20.setParamCameraExposure

```
def setParamCameraExposure(self, arg0)
//功能： 设置相机曝光时间
//输入参数： exposure(相机曝光时间)
//输出参数： 无
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

21.setParamCameraGain

```
def setParamCameraGain(self, arg0)
//功能： 设置相机增益
//输入参数： gain(相机增益)
//输出参数： 无
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

22.setParamDepthFilter

```
def setParamDepthFilter(self, arg0, arg1)
//功能： 设置深度图滤波参数
//输入参数： use(开关： 1 开、 0 关)、depth_filterthreshold(深度图在 1000mm 距离过滤的噪声阈值)
//输出参数： 无
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

23.setParamGenerateBrightness

```
def setParamGenerateBrightness(self, arg0, arg1)
//功能： 设置生成亮度图参数
//输入参数： model(1:与条纹图同步连续曝光、2:单独发光曝光、3:不发光单独曝光)、exposure(亮度图曝光时间)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

24.setParamGrayRectify

```
def setParamGrayRectify(self, arg0, arg1, arg2)
//功能： 设置点云灰度补偿参数
//输入参数： use(开关:1开、0关)、radius(半径:3、5、7、9)、sigma(补偿强度,范围0-100)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

25.setParamLedCurrent

```
def setParamLedCurrent(self, arg0)
//功能： 设置投影亮度
//输入参数： 亮度值
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

26.setParamMixedHdr

```
def setParamMixedHdr(self, arg0, arg1, numpy_int32=None, *args, **kwargs)
//功能： 设置混合多曝光参数(最大曝光次数为6次)
//输入参数： num(曝光次数)、exposure_param[6](6个曝光参数、前num个有效)、led_param[6](6个led亮度参数、前num个有效)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

27.setParamMultipleExposureModel

```
def setParamMultipleExposureModel(self, arg0)
//功能： 设置多曝光模式
//输入参数： model(1: HDR(默认值)、2: 重复曝光)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

28.setParamOutlierFilter

```
def setParamOutlierFilter(self, arg0)
//功能： 设置外点过滤阈值
//输入参数： threshold(阈值 0-100)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

29.setParamRadiusFilter

```
def setParamRadiusFilter(self, arg0, arg1, arg2)
//功能： 设置点云半径滤波参数
//输入参数： use(开关: 1开、0关)、radius(半径)、num(有效点)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

30.setParamReflectFilter

```
def setParamReflectFilter(self, arg0, arg1)
//功能： 设置反射滤波参数
//输入参数： use(开关: 1开、0关)、param_b(过滤系数: 范围 0-100)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

31.setParamRepetitionExposureNum

```
def setParamRepetitionExposureNum(self, arg0)
//功能： 设置重复曝光数
//输入参数： num(2-10)
//输出参数： 无
//返回值： 类型(int):返回0表示连接成功;返回-1表示连接失败.
```

32.setParamSmoothing

```
def setParamSmoothing(self, arg0)
//功能： 设置点云平滑参数
//输入参数： smoothing (0:关、1-5:平滑程度由低到高)
//输出参数： 无
//返回值： 类型 (int) :返回 0 表示连接成功;返回-1 表示连接失败.
```

33.captureBrightnessData

```
//功能： 获取亮度图
//输入参数： color (图像颜色类型)
//输出参数： brightness (亮度图)
//返回值： 类型 (int) :返回 0 表示获取数据成功;返回-1 表示采集数据失败.
```

#相机参数类

class CalibrationParam:

```
def __init__(self):
    distortion = property(lambda self: object(), lambda self, v: None, lambda self: None)
    extrinsic = property(lambda self: object(), lambda self, v: None, lambda self: None)
    intrinsic = property(lambda self: object(), lambda self, v: None, lambda self: None)
//功能： 可配合 SPARROW 中 getCalibrationParam 获取到不同相机的参数结构体
```

class Color

```
class Color(__pybind11_builtins.pybind11_object):
    Rgb
    Bgr
    Bayer
    Gray
//功能： 可配合 SPARROW 接口设置获取 2D 图像数据的类型
```

class Engine

```
class Engine(__pybind11_builtins.pybind11_object):
```

```
    Normal
```

```
    Black
```

```
    Reflect
```

```
//功能： 设置相机的不同引擎
```

Functions

1.get_distortion

```
def get_distortion(calibration_param)
```

```
//功能： 获取参数结构体中的畸变参数
```

2.get_extrinsic

```
def get_extrinsic(calibration_param)
```

```
//功能： 获取参数结构体中的外参数
```

3.get_intrinsic

```
def get_intrinsic(calibration_param)
```

```
//功能： 获取参数结构体中的内参数
```