跨维3D相机DFX系列接口文档

目录

[跨维3D相机DFX系列接口文档 1](#_Toc17878)

[版本记录 3](#_Toc32282)

[接口描述 4](#_Toc28983)

[1. Connect 4](#_Toc274)

[2. Disconnect 4](#_Toc7968)

[3. Get camera resolution 4](#_Toc138)

[4. Capture camera data 4](#_Toc31804)

[5. Get Depth Data 5](#_Toc28995)

[6. Get HeightMap Data 5](#_Toc16906)

[7. Get Brightness Data 5](#_Toc18323)

[8. Get Standard Plane Param 5](#_Toc30988)

[9. Get Height Map Data Base Param 6](#_Toc24918)

[10. Get Pointcloud Data 6](#_Toc10635)

[11. Get calibration parameters 6](#_Toc10271)

[12. Set Led Current parameters 6](#_Toc22390)

[13. Get Led Current parameters 6](#_Toc30248)

[14. Set Hdr parameters 7](#_Toc18971)

[15. Get Hdr parameters 7](#_Toc23448)

[16. Set Standard Plane parameters 7](#_Toc26320)

[17. Get Standard Plane parameters 7](#_Toc3857)

[18. Set Camera Exposure Parameters 8](#_Toc24006)

[19. Get Camera Exposure Parameters 8](#_Toc27563)

[例程 8](#_Toc8864)

[错误码 9](#_Toc12228)

# 版本记录

|  |  |  |  |
| --- | --- | --- | --- |
| **时间** | **版本** | **作者** | **描述** |
| 2021.02.21 | V1.0.1 | 张观锦 | 完成9个基本函数 |
| 2021.03.18 | V1.0.2 | 张观锦 | 完成17个基本函数 |
| 2021.03.18 | V1.0.3 | 张观锦 | 完成17个基本函数 |

# 接口描述

## Connect

int DfConnect(const char\* camera\_id);

//函数名： DfConnect

//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.

## Disconnect

int DfDisconnect(const char\* camera\_id);

//函数名： DfDisconnect

//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表示断开成功;返回-1表示断开失败.

## Get camera resolution

int DfGetCameraResolution(int\* width, int\* height);

//函数名： DfGetCameraResolution

//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表示获取参数成功;返回-1表示获取参数失败。

## Capture camera data

int DfCaptureData(int exposure\_num, char\* &timestamp);

//函数名： DfCaptureData

//功能： 采集点云数据并阻塞至返回结果

//输入参数： exposure\_num（曝光次数）：可设置值为1、2、3（暂时不起作用）。

//输出参数： timestamp(时间戳)

//返回值： 类型（int）:返回0表示获取采集点云成功;返回-1表示采集点云失败。

## Get Depth Data

int DfGetDepthData(unsigned short\* depth);

//函数名： DfGetDepthData

//功能： 获取深度图数据

//输入参数：无

//输出参数： depth(深度图)

//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

参数是相应图像的数据指针，调用方需要根据DfGetCameraResolution返回的结果 提前提前申请好内存空间。数据在内存中将以H（行）/W（列）/C（通道）的顺序排列。

## Get HeightMap Data

//函数名： DfGetHeightMapData

//功能： 获取高度映射图数据

//输入参数：无

//输出参数： height\_map(高度映射图)

//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.参数是相应图像的数据指针，调用方需要根据DfGetCameraResolution返回的结果 提前提前申请好内存空间。数据在内存中将以H（行）/W（列）/C（通道）的顺序排列。

## Get Brightness Data

int DfGetBrightnessData(unsigned short\* depth);

//函数名： DfGetBrightnessData

//功能： 获取亮度图数据

//输入参数：无

//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。

参数是相应图像的数据指针，调用方需要根据DfGetCameraResolution返回的结果 提前提前申请好内存空间。数据在内存中将以H（行）/W（列）/C（通道）的顺序排列。

## Get Standard Plane Param

int DfGetStandardPlaneParam(float\* R,float\* T);

//函数名： DfGetStandardPlaneParam

//功能： 获取基准平面参数

//输入参数：无

//输出参数： R(旋转矩阵：3\*3)、T(平移矩阵：3\*1)

//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.

## Get Height Map Data Base Param

int DfGetHeightMapDataBaseParam(float\* R, float\* T, float\* height\_map);

//函数名： DfGetHeightMapDataBaseParam

//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表示获取数据成功;返回-1表示采集数据失败。

## Get Pointcloud Data

int DfGetPointcloudData(float\* point\_cloud);

//函数名： DfGetPointcloudData

//功能： 获取点云数据（深度图转点云）

//输入参数：无

//输出参数： point\_cloud(点云)

//返回值： 类型（int）:返回0表示获取数据成功;返回-1表示采集数据失败

参数是相应图像的数据指针，调用方需要根据DfGetCameraResolution返回的结果 提前提前申请好内存空间。数据在内存中将以H（行）/W（列）/C（通道）的顺序排列。

## Get calibration parameters

int DfGetCalibrationParam(struct CalibrationParam\* calibration\_param);

//函数名： DfGetCalibrationParam

//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表示获取标定参数成功;返回-1表示获取标定参数失败.

## Set Led Current parameters

int DfSetParamLedCurrent(int led);

//函数名： DfSetParamLedCurrent

//功能： 设置LED电流

//输入参数： led（电流值）

//输出参数： 无

//返回值： 类型（int）:返回0表示获取标定参数成功;返回-1表示获取标定参数失败.

## Get Led Current parameters

int DfGetParamLedCurrent(int led);

//函数名： DfGetParamLedCurrent

//功能： 设置LED电流

//输入参数： 无

//输出参数： led（电流值）

//返回值： 类型（int）:返回0表示获取标定参数成功;返回-1表示获取标定参数失败.

## Set Hdr parameters

int DfSetParamMixedHdr(int num, int exposure\_param[6], int led\_param[6]);

//函数名： DfSetParamMixedHdr

//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表示获取标定参数成功;返回-1表示获取标定参数失败.

## Get Hdr parameters

int DfGetParamMixedHdr(int& num, int exposure\_param[6], int led\_param[6]);

//函数名： DfGetParamMixedHdr

//功能： 获取混合多曝光参数（最大曝光次数为6次）

//输入参数： 无

//输出参数： 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）

//返回值： 类型（int）:返回0表示获取标定参数成功;返回-1表示获取标定参数失败.

## Set Standard Plane parameters

int DfSetParamStandardPlaneExternal(float\* R, float\* T);

//函数名： DfSetParamStandardPlaneExternal

//功能： 设置基准平面的外参

//输入参数：R(旋转矩阵：3\*3)、T(平移矩阵：3\*1)

//输出参数： 无

//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.

## Get Standard Plane parameters

int DfGetParamStandardPlaneExternal(float\* R, float\* T);

//函数名： DfGetParamStandardPlaneExternal

//功能： 获取基准平面的外参

//输入参数：无

//输出参数： R(旋转矩阵：3\*3)、T(平移矩阵：3\*1)

//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.

## Set Camera Exposure Parameters

int DfSetParamCameraExposure(float exposure);

//函数名： DfSetParamCameraExposure

//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表示获取数据成功;返回-1表示采集数据失败.

## Get Camera Exposure Parameters

int DfGetParamCameraExposure(float& exposure);

//函数名： DfGetParamCameraExposure

//功能： 获取相机曝光时间

//输入参数： 无

//输出参数：exposure(相机曝光时间)

//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.

# 例程

详细请看example.cpp

**属性说明**

//相机标定参数结构体

struct CalibrationParam

{

//相机内参

double intrinsic[3\*3];

//相机外参

double extrinsic[4\*4];

//相机畸变

//<k1,k2,p1,p2,k3,k4,k5,k6,s1,s2,s3,s4>暂时只使用5个畸变参数

double distortion[1\*12];

};

# 错误码

|  |  |
| --- | --- |
| 错误码 | 描述 |
| 0 | 成功 |
| -1 | 失败 |
| -2 | 未获取相机分辨率分配内存 |
|  |  |