DFX系列相机接口文档

目录

[DFX系列相机接口文档 1](#_Toc15448)

[版本记录 3](#_Toc23048)

[接口描述 5](#_Toc23747)

[1. Connect 5](#_Toc18269)

[2. Disconnect 5](#_Toc23707)

[3. Get camera resolution 5](#_Toc21875)

[4. Capture camera data 5](#_Toc23077)

[5. Get Depth Data 6](#_Toc29581)

[6. Get HeightMap Data 6](#_Toc4428)

[7. Get Brightness Data 6](#_Toc6360)

[8. Get Standard Plane Param 6](#_Toc17355)

[9. Get Height Map Data Base Param 7](#_Toc4700)

[10. Get Pointcloud Data 7](#_Toc15324)

[11. Get calibration parameters 7](#_Toc13686)

[12. Set Led Current parameters 7](#_Toc1352)

[13. Get Led Current parameters 7](#_Toc17188)

[14. Set Hdr parameters 8](#_Toc6150)

[15. Get Hdr parameters 8](#_Toc27306)

[16. Set Standard Plane parameters 8](#_Toc18641)

[17. Get Standard Plane parameters 8](#_Toc3668)

[18. Set Camera Exposure Parameters 9](#_Toc14165)

[19. Get Camera Exposure Parameters 9](#_Toc17254)

[20. Set Mixed Hdr Parameters 9](#_Toc28391)

[21. Get Mixed Hdr Parameters 9](#_Toc3707)

[22. Set Camera Confidence Parameters 10](#_Toc21991)

[23. Get Camera Confidence Parameters 10](#_Toc10206)

[24. Set Camera Gain Parameters 10](#_Toc10551)

[25. Get Camera Gain Parameters 10](#_Toc12194)

[26. Set Pointcloud Smoothing Parameters 11](#_Toc27760)

[27. Get Pointcloud Smoothing Parameters 11](#_Toc14212)

[28. Set Radius Filter Parameters 11](#_Toc15550)

[29. Get Radius Filter Parameters 11](#_Toc30943)

[30. Set Outlier Filter Parameters 12](#_Toc9089)

[31. Get Outlier Filter Parameters 12](#_Toc32567)

[32. Update Device List 12](#_Toc12080)

[33. Get All Device Base Info 12](#_Toc5159)

[例程 13](#_Toc3083)

[错误码 13](#_Toc24503)

# 版本记录

|  |  |  |  |
| --- | --- | --- | --- |
| **时间** | **版本** | **作者** | **描述** |
| 2022.02.21 | V1.0.1 | 张观锦 | 完成9个基本函数 |
| 2022.03.18 | V1.0.2 | 张观锦 | 完成17个基本函数 |
| 2022.05.13 | V1.0.3 | 张观锦 | 完成19个基本函数 |
| 2022.06.16 | V1.0.4 | 张观锦 | 完成27个基本函数 |
| 2022.08.05 | V1.0.5 | 张观锦 | 完成29个基本函数 |
| 2022.09.09 | V1.0.6 | 张观锦 | 完成33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）：1（为单曝光）、大于1为多曝光（曝光次数由hdr参数决定）。

//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.

## Set Mixed 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 Mixed 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 Camera Confidence Parameters

int DfSetParamCameraConfidence(float confidence);

//函数名： DfSetParamCameraConfidence

//功能： 设置相机曝光时间

//输入参数：confidence(相机置信度)

//输出参数： 无

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

## Get Camera Confidence Parameters

int DfGetParamCameraConfidence(float& confidence);

//函数名： DfGetParamCameraConfidence

//功能： 获取相机曝光时间

//输入参数： 无

//输出参数：confidence(相机置信度)

//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 Gain Parameters

int DfSetParamCameraGain(float gain);

//函数名： DfSetParamCameraGain

//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表示获取数据成功;返回-1表示采集数据失败.

## Get Camera Gain Parameters

int DfGetParamCameraGain(float& gain);

//函数名： DfGetParamCameraGain

//功能： 获取相机增益

//输入参数： 无

//输出参数：gain(相机增益)

//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 Pointcloud Smoothing Parameters

int DfSetParamSmoothing(int smoothing);

//函数名： DfSetParamSmoothing

//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表示获取数据成功;返回-1表示采集数据失败.

## Get Pointcloud Smoothing Parameters

int DfGetParamSmoothing(int& smoothing);

//函数名： DfGetParamSmoothing

//功能： 设置点云平滑参数

//输入参数：无

//输出参数：smoothing(0:关、1-5:平滑程度由低到高)

//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 Radius Filter Parameters

int DfSetParamRadiusFilter(int use,float radius,int num);

//函数名： DfSetParamRadiusFilter

//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表示获取数据成功;返回-1表示采集数据失败.

## Get Radius Filter Parameters

int DfGetParamRadiusFilter(int& use, float& radius, int& num);

//函数名： DfGetParamRadiusFilter

//功能： 获取点云半径滤波参数

//输入参数：无

//输出参数：use(开关：1开、0关)、radius(半径）、num（有效点）

//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 Outlier Filter Parameters

int DfSetParamOutlierFilter(float threshold);

//函数名： DfSetParamOutlierFilter

//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表示获取数据成功;返回-1表示采集数据失败.

## Get Outlier Filter Parameters

int DfGetParamOutlierFilter(float& threshold);

//函数名： DfGetParamOutlierFilter

//功能： 获取外点过滤阈值

//输入参数： 无

//输出参数：threshold(阈值0-100)

//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.

## Update Device List

int DfUpdateDeviceList(int& device\_num);

//函数名： DfUpdateDeviceList

//功能： 获取可连接设备数

//输入参数： device\_num(设备数)

//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.

## Get All Device Base Info

int DfGetAllDeviceBaseInfo(DeviceBaseInfo\* pDeviceInfo, int\* pBufferSize);

//函数名： DfGetAllDeviceBaseInfo

//功能： 获取设备基本信息

//输入参数： pDeviceInfo(设备信息)、pBufferSize（设备结构体内存尺寸）

//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.

# 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];

};

//设备基本信息结构体

struct DeviceBaseInfo

{

//相机内参

char mac[64];

//相机外参

char ip[64];

};

# 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 |
|  |  |