

isc-dpl Manual

【ご注意】

1. 本マニュアルの内容の一部または全部を無断転載することは禁止されています
2. 本マニュアルの内容に関しては将来予告なしに変更することがあります
3. 本マニュアルの内容について万全を期して作成しております、万一ご不審な点や誤り、記載漏れなどお気づきのことがございましたら、ご連絡ください
4. 運用した結果の影響に関しては、3. 項にかかわらず責任を負いかねますのでご了承ください

Copyright 2023 ITD Lab 株式会社

本マニュアルで使用されている各会社名、各製品名は各社の商標あるいは登録商標です

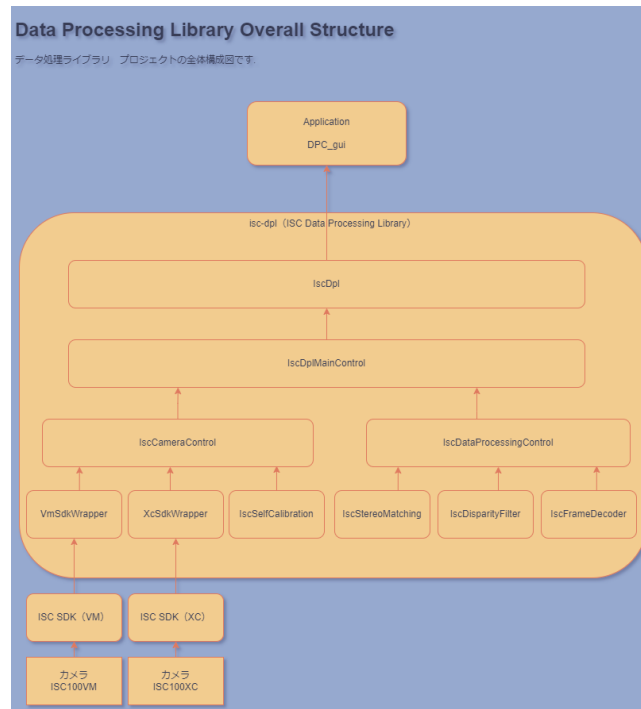
目次

1. 概要.....	3
2. IscDpl.....	5
2. 1 初期化と終了.....	7
2. 2 カメラ 開始とデータ取得.....	8
2. 3 データ処理ライブラリ データ取得.....	9
2. 4 カメラ 固有パラメータ取得、設定.....	10
2. 5 カメラ 制御パラメータ設定.....	10
3.設定ファイル.....	11
改版履歴.....	12

1. 概要

isc-dpl(ISC Data Processing Library)の説明書である。

isc-dpl は、データ処理ライブラリ及びその周辺モジュールによって構成されている。
全体の構成を下図に示す。



【データ処理ライブラリ】

- ・ IscDataProcessingControl：データ処理ライブラリの呼び出しを行う
 - ・ IscStereoMatching：ステレオマッチングを行う
 - ・ IscDisparityFilter：視差の平均化、補間処理を行う
 - ・ IscFrameDecoder：フレームデータの展開を行う補助ライブラリ
 - ・ IscSelfCalibration：ソフトウェアによるステレオ平行化を行う

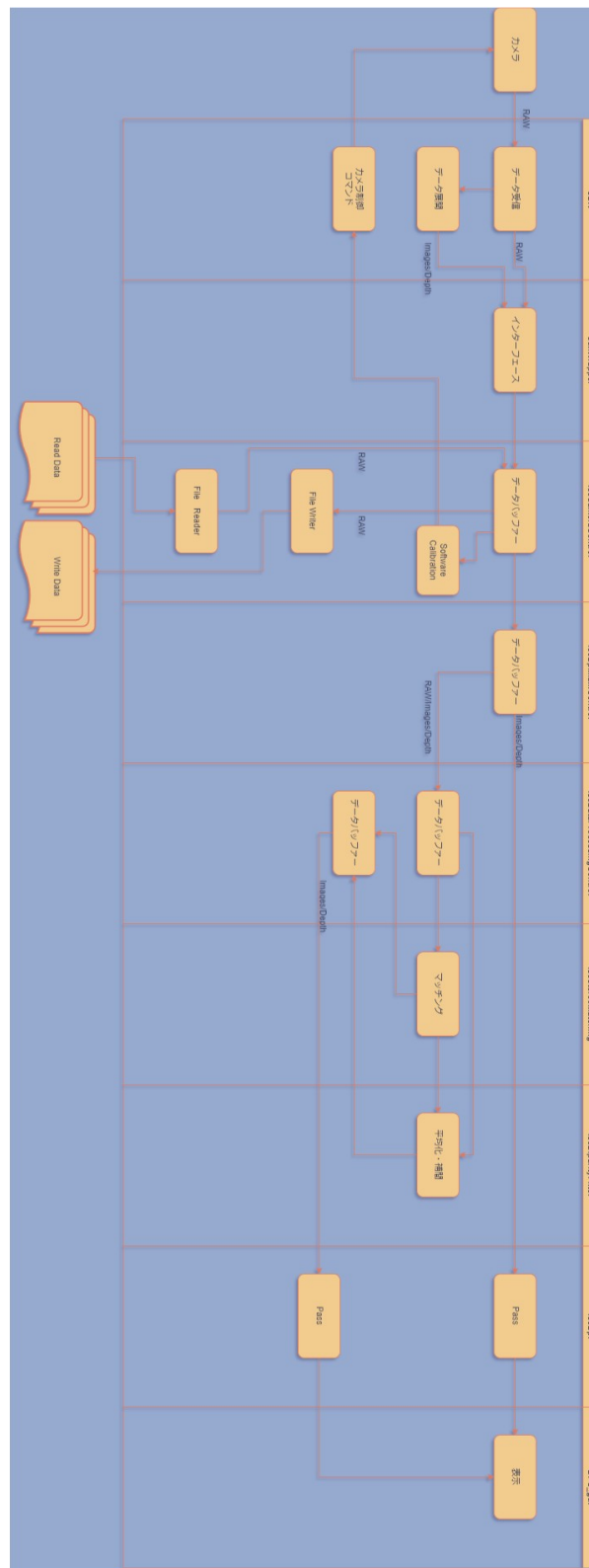
【周辺モジュール】

- ・ IscDpl：インターフェース用 DLL である
- ・ IscDplMainControl：全体の制御、データの受け渡しを行う
- ・ IscCameraControl：カメラの制御及びカメラデータのファイル読み書きを行う
 - ・ VmSdkWrapper / XcSdkWrapper：SDK インターフェースである

【カメラ SDK】

- ・ ISC SDK(VM/XC)：それぞれのカメラに対応した SDK である

データフローを以下に示す。



2. IscDpl

インターフェースを提供する Class である。

関数の一覧を下図に示す。

カテゴリ	API	説明
初期化/終了	Initialize	モジュールの初期化を行う
	Terminate	モジュールの使用を終了する
カメラ制御	DeviceOptionIsImplemented	機能の有無を取得する
	DeviceOptionIsReadable	パラメータの取得の可否を取得する
	DeviceOptionIsWritable	パラメータの設定の可否を取得する
	DeviceGetOptionMin	パラメータの設定可能な最小値
	DeviceGetOptionMax	パラメータの設定可能な最大値
	DeviceGetOptionInc	パラメータの最小増減値を取得する
	DeviceGetOption	パラメータを取得する
	DeviceSetOption	パラメータを設定する
	Start	カメラ又はファイルからの取り込みを開始する
	Stop	取り込みを停止する
	GetGrabMode	現在の取り込みモードを取得する
カメラデータ	InitializeIsclangeinfo	取得用バッファの初期化
	ReleaseIsclangeinfo	取得用バッファの解放
	GetCameraData	データの取得
	GetFileInformation	ファイル読み込み時のファイルヘッダー取得
距離情報取得	GetPositionDepth	指定位置の距離を取得
	GetPosition3D	指定位置の 3D 情報を取得
	GetAreaStatistics	指定領域の幅、高さ、距離を取得
データ処理モジュール制御	GetTotalModuleCount	モジュールの数を取得
	GetModuleNameByIndex	指定番号のモジュール名を取得
	GetDataProcModuleParameter	モジュールのパラメータを取得
	SetDataProcModuleParameter	モジュールのパラメータを設定
	GetParameterFileName	モジュールのパラメータファイル名を取得

	ReloadParameterFromFile	パラメータをファイルからリロードする
データ処理モジュールデータ	InitializeIscDataProcResultData	取得用バッファの初期化
	ReleaseIscDataProcResultData	取得用バッファの解放
	GetDataProcModuleData	データの取得

2. 1 初期化と終了

(1) 初期化

```
ns_isc_dpl::IscDpl* isc_dpl_ = new ns_isc_dpl::IscDpl;

// 初期化パラメータ設定
IscDplConfiguration isc_dpl_configuration_ = {};
swprintf_s(isc_dpl_configuration_.configuration_file_path, L"%s", _T("c:¥¥isc-dpl¥¥bin"));
swprintf_s(isc_dpl_configuration_.log_file_path, L"%s", _T("c:¥¥isc-dpl¥¥log"));
isc_dpl_configuration_.log_level = 0;
isc_dpl_configuration_.enabled_camera = true;

const int camera_model = 1
IscCameraModel isc_camera_model = IscCameraModel::kUnknown;
switch (camera_model) {
case 0:isc_camera_model = IscCameraModel::kVM; break;
case 1:isc_camera_model = IscCameraModel::kXC; break;
case 2:isc_camera_model = IscCameraModel::k4K; break;
case 3:isc_camera_model = IscCameraModel::k4KA; break;
case 4:isc_camera_model = IscCameraModel::k4KJ; break;
}
isc_dpl_configuration_.isc_camera_model = isc_camera_model;

swprintf_s(isc_dpl_configuration_.save_image_path, L"%s", _T("c:¥¥isc-dpl¥¥data"));
swprintf_s(isc_dpl_configuration_.load_image_path, L"%s", _T("c:¥¥isc-dpl¥¥data"));

isc_dpl_configuration_.enabled_data_proc_module = true;

DPL_RESULT dpl_result = isc_dpl_>Initialize(&isc_dpl_configuration_);
```

(2) 終了

```
if (isc_dpl_ != nullptr) {
    isc_dpl_>Terminate();
    delete isc_dpl_;
    isc_dpl_ = nullptr;
}
```

2. 2 カメラ 開始とデータ取得

```
// バッファ初期化
IsclmageInfo isc_image_info = {};
isc_dpl_ ->Initializelsclmageinfo(&isc_image_info);

// 開始パラメータ初期化
IscGrabStartMode isc_grab_start_mode = {};
isc_start_mode.isc_grab_start_mode.isc_grab_mode = IscGrabMode::kParallax;
isc_start_mode.isc_grab_start_mode.isc_grab_color_mode = IscGrabColorMode::kColorOFF;
isc_start_mode.isc_grab_start_mode.isc_get_raw_mode = IscGetModeRaw::kRawOff;
isc_start_mode.isc_grab_start_mode.isc_get_color_mode = IscGetModeColor::kBGR;
isc_start_mode.isc_grab_start_mode.isc_record_mode = IscRecordMode::kRecordOff;
isc_start_mode.isc_grab_start_mode.isc_play_mode = IscPlayMode::kPlayOff;
isc_start_mode.isc_grab_start_mode.isc_play_mode_parameter.interval = 30;
memset(isc_start_mode.isc_grab_start_mode.isc_play_mode_parameter.play_file_name, 0,
sizeof(isc_start_mode.isc_grab_start_mode.isc_play_mode_parameter.play_file_name));
isc_start_mode.isc_grab_start_mode.isc_get_mode.wait_time = 100;

isc_start_mode.isc_dataproc_start_mode.enabled_block_matching = true;
isc_start_mode.isc_dataproc_start_mode.enabled_frame_decoder = true;

DPL_RESULT dpl_result = isc_dpl_ ->Start(&isc_control.isc_start_mode);
if (dpl_result == DPC_E_OK) {
}
else {
}

// データ取得 取得したら停止
while(true){
    // データ取得
    DPL_RESULT dpl_result = isc_dpl_ ->GetCameraData(&isc_image_info);
    if (dpl_result != DPC_E_OK) {
    }
    else{
        break;
    }
}

// 停止
DPL_RESULT dpl_result = isc_dpl_ ->Stop();

// バッファ解放
isc_dpl_ ->Releaelsclmageinfo(&isc_image_info);
```


2. 3 データ処理ライブラリ データ取得

```
// バッファ初期化
IscDataProcResultData isc_data_proc_result_data = {};
isc_dpl_>InitializelscDataProcResultData(&isc_data_proc_result_data);

*** カメラ取り込み開始 ***

// データ取得 取得したら停止
while(true){
    // データ取得

    // camera images
    DPL_RESULT dpl_result = isc_dpl_>GetCameraData(&isc_image_info);
    if (dpl_result != DPC_E_OK) {
        continue;
    }
    else{
        // deta processing result
        // カメラデータとはタイミングが違うため取得まで繰り返す
        dpl_result = isc_dpl_>GetDataProcModuleData(&isc_data_proc_result_data);
        if (dpl_result == DPC_E_OK) {
        }
        else{
            break;
        }
    }
}

*** カメラ取り込み停止 ***

// バッファ解放
isc_dpl_>ReleaelscDataProcResultData(&isc_data_proc_result_data);
```

2. 4 カメラ 固有パラメータ取得、設定

```
double b=0, bf=0, dinf=0;
// 基線長
isc_dpl_>DeviceGetOption(IscCameraInfo::kBaseLength, &b);
// BF
isc_dpl_>DeviceGetOption(IscCameraInfo::kBF, &bf);
// D_INF
isc_dpl_>DeviceGetOption(IscCameraInfo::kDINF, &dinf);
```

2. 5 カメラ 制御パラメータ設定

```
// Shutter Mode を取得
IscShutterMode shutter_mode = IscShutterMode::kManualShutter;
int ret = isc_dpl_>DeviceGetOption(IscCameraParameter::kShutterMode, &shutter_mode);

// Exposure 取得と設定
int read_value = 0;
ret = isc_dpl_>DeviceGetOption(IscCameraParameter::kExposure, &read_value);
read_value = +10;
ret = isc_dpl_>DeviceSetOption(IscCameraParameter::kExposure, read_value);
```

3.設定ファイル

各データ処理ライブラリに対応した設定ファイルは、以下である。

ライブラリ	設定ファイル
IscStereoMatching	StereoMatcingParameter_[*].ini
IscDisparityFilter	DisparityFilterParameter_[*].ini
IscFrameDecoder	FrameDecoderParameter_[*].ini
IscSelfCalibration	SelfCalibrationParameter_[*].ini

[*] : ISC カメラのタイプ

設定ファイルの詳細については、ファイル内コメント及び Library_Manual を参照

改版履歴

Rev	Date	Content
0.0.1	2023/4/30	初版発行
0.0.2	2023/7/31	ライブラリ 0.0.2 に対応
0.0.3	2023/10/30	ライブラリ 0.0.3 に対応
0.0.4	2023/11/1	誤記訂正
0.0.5	2023/11/30	タイトル及びファイル名を変更 タイトル：ISC DPL 説明書 ~ isc-dpl Manual ファイル名：ISC_DPL モジュール説明書 ~ isc-dpl_Manual

End of Document