



MI FDFR 軟體發展參考

修訂日期	版本	作者	說明
2016-01-13	V1.12		

版權所有 © 深圳市晨星半導體有限公司 2016。保留一切權利。

非經本公司書面許可，任何單位和個人不得擅自摘抄、複製本文檔內容的部分或全部，並不得以任何形式傳播。

注意

您購買的產品、服務或特性等應受晨星半導體有限公司商業合同和條款的約束，本文檔中描述的全部或部分產品，服務或特性可能不在您的購買或使用範圍之內。除非合同另有約定，本文檔僅作為使用指導，本文檔中的所有陳述，資訊和建議不構成任何明示或暗示的擔保。

晨星軟體研發(深圳)有限公司

地址：深圳市南山區大沖一路 18 號華潤置地大廈 E 座 9-13 樓

電話：0755-88860611

前言

本文為使用 FDFR 進行開發的程式師而寫，目的是供您在開發過程中查閱 FDFR 軟體發展包的各種參考資訊，包括 API、標頭檔、錯誤碼等。

本文檔描述 FDFR 軟體的各個 API 的使用方法，以及相關資料結構和錯誤碼。

讀者對象

本文檔主要適用於以下工程師

- 技術支援工程師。
- 軟體發展工程師。

前言	錯誤! 尚未定義書籤。
1 概述	錯誤! 尚未定義書籤。
2 API 参考	錯誤! 尚未定義書籤。
2.1 API 概述	錯誤! 尚未定義書籤。
2.2 API 列表	錯誤! 尚未定義書籤。
MI_FD_Init.....	錯誤! 尚未定義書籤。
MI_FD_Uninit.....	錯誤! 尚未定義書籤。
MI_FD_EnableFD.....	錯誤! 尚未定義書籤。
MI_FD_SetOption.....	錯誤! 尚未定義書籤。
MI_FD_Run.....	錯誤! 尚未定義書籤。
MI_FD_GetFaceInfo.....	錯誤! 尚未定義書籤。
MI_FR_Init.....	錯誤! 尚未定義書籤。
MI_FR_Uninit	錯誤! 尚未定義書籤。
MI_FR_EnableFR	錯誤! 尚未定義書籤。
MI_FR_GetFeatureSizes	錯誤! 尚未定義書籤。
MI_FR_CalFeatureFromImg	錯誤! 尚未定義書籤。
MI_FR_CalFeatureFromRawY.....	錯誤! 尚未定義書籤。
MI_FR_GetFeatureData.....	錯誤! 尚未定義書籤。
MI_FR_SetFeatureData	錯誤! 尚未定義書籤。
MI_FR_GetFRInfo	錯誤! 尚未定義書籤。
3 数据类型.....	錯誤! 尚未定義書籤。
FD_HANDLE	錯誤! 尚未定義書籤。
FR_HANDLE	錯誤! 尚未定義書籤。
FDOption_e	錯誤! 尚未定義書籤。
FDDetectMode_e	錯誤! 尚未定義書籤。
FacePos_t	錯誤! 尚未定義書籤。
FRFaceInfo_t.....	錯誤! 尚未定義書籤。
4 错误码	錯誤! 尚未定義書籤。
5 流程	錯誤! 尚未定義書籤。
6 举例	錯誤! 尚未定義書籤。

1 概述

臉部識別(Face Recognition)功能用於在攝像頭獲取的視頻中識別資料庫中的臉部特徵的場景。此功能可用于安防監控中。

MI FDFR API 需要滿足下面幾個條件才能正常工作:

- 1, 系統中有人臉檢測庫 libfd.so 。
- 2, 系統時間需要大於 2015 年。
- 3, 使用 FDFR api 的應用目錄下面需要有 FDFR 所需的資料庫資料夾 ROMfiles 。
- 使用 FD api 需要有 ROMfiles\arm\det_hs.bin 文件。
- 使用 FR api 需要有 ROMfiles\arm\dic_part.bin 和 ROMfiles\arm\fea_ha2.bin 文件。
- 4, 輸入的 yuv 資料的解析度和 FDFR api 設置的解析度參數一致。

2 API 參考

2.1 API 概述

- [MI_FD_Init](#):初始化 FD 庫。
- [MI_FD_Uninit](#) : 退出庫函數,釋放記憶體。
- [MI_FD_EnableFD](#) : 使能臉部檢測。
- [MI_FD_SetOption](#) : 設置臉部識別參數。
- [MI_FD_Run](#) : 開始處理脸部識別圖像幀。
- [MI_FD_GetFaceInfo](#) : 獲取脸部檢測結果。
- [MI_FR_Init](#): 初始化 FR 庫。
- [MI_FR_Uninit](#): 退出庫函數,釋放記憶體。
- [MI_FR_EnableFR](#): 使能脸部識別。
- [MI_FR_GetFeatureSizes](#): 獲取人臉特徵資料的大小。
- [MI_FR_CalFeatureFromImg](#): 從 pgm 格式的圖片生成人臉特徵。
- [MI_FR_CalFeatureFromRawY](#): 從圖像 YUV 中的 y 分量資料生成人臉特徵。
- [MI_FR_GetFeatureData](#): 獲取資料庫中的人臉特徵。
- [MI_FR_SetFeatureData](#) : 設置人臉特徵資訊到資料庫。
- [MI_FR_GetFRInfo](#): 獲取人臉識別的結果資訊。

2.2 API 列表

MI_FD_Init

【說明】

初始化 FD 庫。

【語法】

FD_HANDLE MI_FD_Init(S32 inImgW, S32 inImgH) ;

【參數】

參數描述	說明
inImgW	輸入源寬
inImgH	輸入源高

【返回值】

返回值	說明
FD_HANDLE	FD 的 handle 控制碼

【需求】

●標頭檔：mi_fd.h

●庫文件：libmi.so

【注意】

減少 FD 演算法的 cpu 消耗，可降低輸入源的解析度，源圖像的寬高推薦值為 320、180

【舉例】

無。

MI_FD_Uninit

【說明】

退出庫函數，釋放記憶體。

【語法】

void MI_FD_Uninit([FD_HANDLE](#) fdHandle);

【參數】

參數描述	說明
fdHandle	FD 的 handle 控制碼

【返回值】

無。

【需求】

●標頭檔：mi_fd.h

●庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FD_EnableFD

【說明】

使能臉部檢測。

【語法】

MI_RET MI_FD_EnableFD([FD_HANDLE](#) fdHandle, S32 bEnable) ;

【參數】

參數描述	說明
fdHandle	FD 的 handle 控制碼
bEnable	使能標誌

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_FD_ENABLE_ERROR	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FD_SetOption

【說明】

設置臉部識別參數。

【語法】

MI_RET MI_FD_SetOption([FD_HANDLE](#) fdHandle, [FDOption_e](#) opt, S32 val) ;

【參數】

參數描述	說明
fdHandle	FD 的 handle 控制碼
opt	參數選項
val	參數值

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_INVALID_PARAMETER	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

[FDOption_e](#) 分別對應的 val 取值範圍：

FD_OPTION_DETECT_MODE : val 取 [FDDetectMode_e](#) 中的一個枚舉值

FD_OPTION_FACE_WIDTH : val 需要大於 20

FD_OPTION_PARTIAL_WIDTH : val 需要大於 20

FD_OPTION_FACE_DIRECTION : val 對應的含義如下：

0：可檢測人臉對應攝像頭是逆時針旋轉 0 度的方向

- 1：可檢測人臉對應攝像頭是逆時針旋轉 90 度的方向
- 2：可檢測人臉對應攝像頭是逆時針旋轉 0 和 90 度的方向
- 3：可檢測人臉對應攝像頭是逆時針旋轉 180 度的方向
- 4：可檢測人臉對應攝像頭是逆時針旋轉 0 和 180 度的方向
- 5：可檢測人臉對應攝像頭是逆時針旋轉 90 和 180 度的方向
- 6：可檢測人臉對應攝像頭是逆時針旋轉 0、90 和 180 度的方向
- 7：可檢測人臉對應攝像頭是逆時針旋轉 270 度的方向
- 8：可檢測人臉對應攝像頭是逆時針旋轉 0 和 270 度的方向
- 9：可檢測人臉對應攝像頭是逆時針旋轉 90 和 270 度的方向
- 10：可檢測人臉對應攝像頭是逆時針旋轉 0、90 和 270 度的方向
- 11：可檢測人臉對應攝像頭是逆時針旋轉 180 和 270 度的方向
- 12：可檢測人臉對應攝像頭是逆時針旋轉 0、180 和 270 度的方向
- 13：可檢測人臉對應攝像頭是逆時針旋轉 90、180 和 270 度的方向
- 14：可檢測人臉對應攝像頭是逆時針旋轉 0、90、180 和 270 度的方向

預設值為 0。可檢測的角度越多，運算量越大。

【舉例】

無。

MI_FD_Run

【說明】

開始處理臉部識別圖像幀。

【語法】

S32 MI_FD_Run([FD_HANDLE](#) hdl, const U8* yFrame)

【參數】

參數描述	說明
hdl	FD 的 handle 控制碼
yFrame	圖像 YUV 中的 y 分量資料

【返回值】

返回值	說明
-----	----

-1	Fail
Other	檢測到的臉部數

【需求】

●標頭檔：mi_fd.h

●庫文件：libmi.so

【注意】

減少 FD 演算法的 cpu 使用，可降低輸入 FD 演算法的 YUV 幀率，幀率推薦值為 3~5

【舉例】

無。

MI_FD_GetFaceInfo

【說明】

獲取臉部檢測結果。

【語法】

S32 MI_FD_GetFaceInfo([FD_HANDLE](#) hdl, [FacePos_t](#) **out_facepos)；

【參數】

參數描述	說明
hdl	FD 的 handle 控制碼
out_facepos	臉部位置結構體二級指標

【返回值】

返回值	說明
0	Fail
Other	檢測到的臉部數

【需求】

●標頭檔：mi_fd.h

●庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_Init

【說明】

初始化 FR 庫。

【語法】

FR_HANDLE MI_FR_Init(S32 inImgW, S32 inImgH)；

【參數】

參數描述	說明
inImgW	輸入源寬

inImgH	輸入源高
--------	------

【返回值】

返回值	說明
FR_HANDLE	FR 的 handle 控制碼

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

減少 FR 演算法的 cpu 消耗，可降低輸入源的解析度，源圖像的寬高推薦值為 320、180

【舉例】

無。

MI_FR_Uninit

【說明】

退出庫函數，釋放記憶體。

【語法】

void MI_FR_Uninit([FR_HANDLE](#) frHandle);

【參數】

參數描述	說明
frHandle	FR 的 handle 控制碼

【返回值】

無。

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_EnableFR

【說明】

使能臉部檢測。

【語法】

MI_RET MI_FR_EnableFR([FR_HANDLE](#) frHandle, S32 bEnable) ;

【參數】

參數描述	說明
frHandle	FR 的 handle 控制碼
bEnable	使能標誌

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_FR_ENABLE_ERROR	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_GetFeatureSizes

【說明】

獲取單個人臉特徵資料的大小。

【語法】

```
S32 MI_FR_GetFeatureSizes();
```

【參數】

無。

【返回值】

單個人臉特徵資料的大小。

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_CalFeatureFromImg

【說明】

從圖像 YUV 中的 y 分量資料生成人臉特徵。

【語法】

```
MI_RET MI_FR_CalFeatureFromImg(S8* filename, S32 store_idx);
```

【參數】

參數描述	說明
filename	圖片檔案名
store_idx	人臉特徵在資料庫的索引號

【返回值】

返回值	說明
MI_RET_SUCCESS	Success

MI_FDFR_RET_IN_IMAGE_ERROR	Fail
----------------------------	------

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_CalFeatureFromRawY

【說明】

從圖像 YUV 中的 y 分量資料生成人臉特徵。

【語法】

MI_RET MI_FR_CalFeatureFromRawY(U8* imgPtr, S32 width, S32 height, S32 store_idx);

【參數】

參數描述	說明
imgPtr	y 分量數據 buffer 的指標
width	圖像的寬度
height	圖像的高度
store_idx	人臉特徵在資料庫的索引號

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_FR_GET_FEATURE_DATA_ERR OR	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_GetFeatureData

【說明】

獲取資料庫中的人臉特徵。

【語法】

MI_RET MI_FR_GetFeatureData(S16 idx, S8* feat_data, S8* feat_name);

【參數】

參數描述	說明
idx	人臉特徵在資料庫的索引號
feat_data	人臉特徵資料
feat_name	人臉特徵的名字

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_FR_GET_FEATURE_DATA_ERR OR	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_SetFeatureData

【說明】

設置人臉特徵資訊到資料庫。

【語法】

```
MI_RET MI_FR_SetFeatureData(S16 idx, S8* feat_data, S8* feat_name);
```

【參數】

參數描述	說明
idx	人臉特徵在資料庫的索引號
feat_data	人臉特徵資料
feat_name	人臉特徵的名字

【返回值】

返回值	說明
MI_RET_SUCCESS	Success
MI_FDFR_RET_FR_SET_FEATURE_DATA_ERR OR	Fail

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

無。

【舉例】

無。

MI_FR_GetFRInfo

【說明】

獲取人臉識別的結果資訊。

【語法】

```
int32_t MI_FR_GetFRInfo(FR\_HANDLE frHandle, int fd_num, FRFaceInfo\_t *faceInfo);
```

【參數】

參數描述	說明
frHandle	FR 的 handle 控制碼
fd_num	最大可獲取的人臉數，即 faceInfo 指向的陣列大小
faceInfo	識別到的人臉資訊陣列

【返回值】

返回值	說明
Face num	Face num

【需求】

- 標頭檔：mi_fd.h
- 庫文件：libmi.so

【注意】

一次最多只能識別到 10 個人臉特徵，所以 fd_num 不能大於 10。

【舉例】

無。

3 資料類型

FD_HANDLE	FD 的 handle 控制碼
FR_HANDLE	FR 的 handle 控制碼
FDOption_e	FD 參數設置選項
FDDetectMode_e	人臉識別的 mode
FacePos_t	人臉識別位置參數結構
FRFaceInfo_t	人臉檢測資訊的結構體

FD_HANDLE

【說明】

FD 的 handle 控制碼

【定義】

```
typedef void* FD_HANDLE;
```

【參數】

無。

FR_HANDLE

【說明】

FR 的 handle 控制碼

【定義】

```
typedef void* FR_HANDLE;
```

【參數】

無。

FDOption_e

【說明】

FD 參數設置選項。

【定義】

```
typedef enum  
{  
    FD_OPTION_DETECT_MODE,  
    FD_OPTION_FACE_WIDTH,  
    FD_OPTION_PARTIAL_WIDTH,  
    FD_OPTION_FACE_DIRECTION ,  
    FD_OPTION_NUM  
}FDOption_e;
```

【參數】

參數描述	說明
FD_OPTION_DETECT_MODE	設置人臉檢測的模式
FD_OPTION_FACE_WIDTH	檢測與跟蹤模式中最小的人臉寬度
FD_OPTION_PARTIAL_WIDTH	局部檢測模式中最小的人臉寬度
FD_OPTION_FACE_DIRECTION	設置識別人臉方向的參數
FD_OPTION_NUM	選項最大數

FDDetectMode_e

【說明】

FD 檢測和追蹤的模式。

【定義】

```
Typedef enum  
{  
    FULL_MODE = 0,  
    TRACK_MODE,  
    PARTIAL_MODE,  
    NUM_MODE
```



```
}FDDetectMode_e;
```

【參數】

參數描述	說明
FULL_MODE	只檢測每一幀資料中的人臉
TRACK_MODE	只追蹤上一幀中檢測的人臉，僅當上一幀中沒有人臉才開始新的檢測
PARTIAL_MODE	追蹤上一幀中的人臉，並檢測其他區域中的人臉
NUM_MODE	選項最大數

FacePos_t

【說明】

人臉識別位置參數結構

【定義】

```
typedef struct
{
    S8    posx;
    S8    posy;
    S8    posw;
    S8    posh;
} FacePos_t;
```

【參數】

參數描述	說明
posx	X 座標位置
posy	Y 座標位置
posw	寬度
posh	高度

FRFaceInfo_t

【說明】

人臉檢測資訊結構體

【定義】

```
typedef struct
{
    S32 store_idx;
    U32 percentage;
    S8 name[32];
}FRFaceInfo_t;
```

【參數】

參數描述	說明
------	----

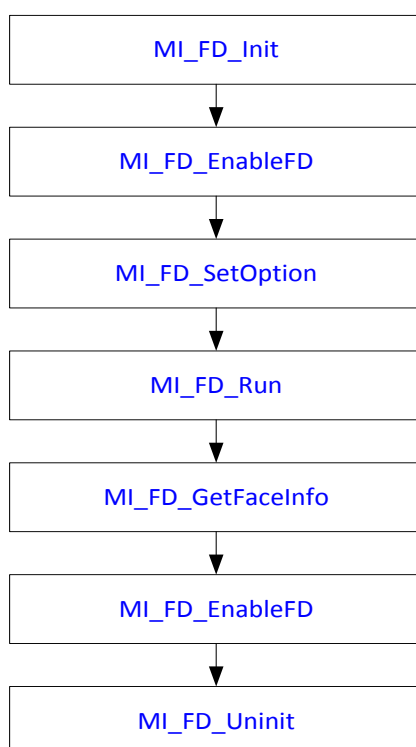
store_idx	人臉特徵在資料庫中的索引（-1 表示資料庫中沒有匹配的人臉特徵）
percentage	人臉特徵與資料庫特徵匹配的百分比
name	人臉特徵在資料庫中對應的名字

4 錯誤碼

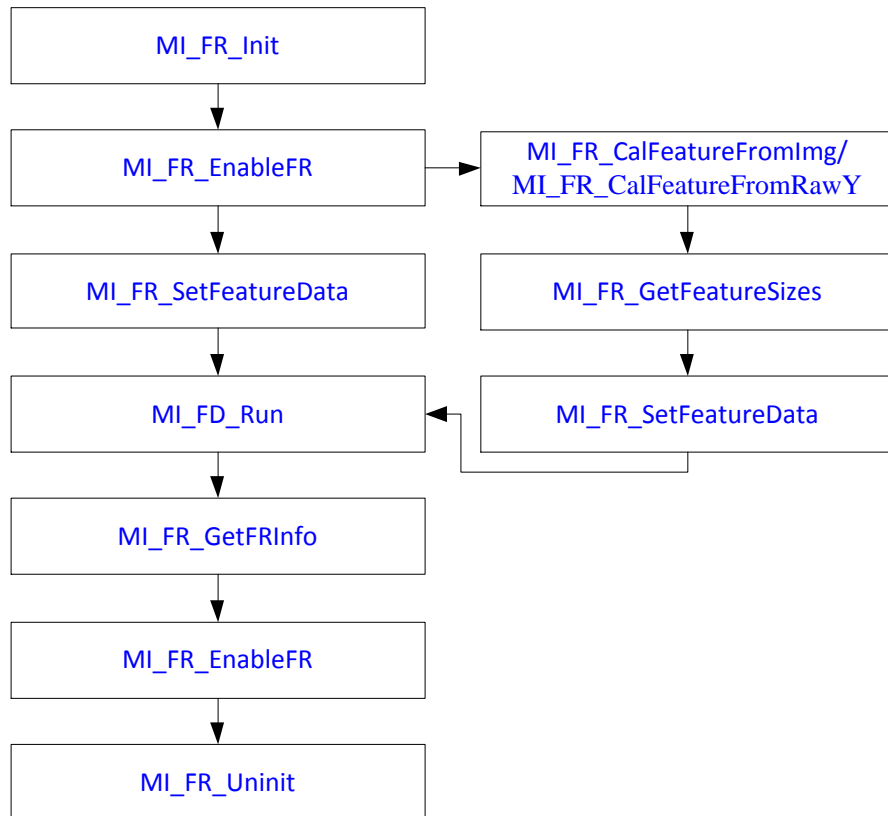
參數描述	說明
MI_RET_SUCCESS	Success
MI_RET_FAIL	Fail

5 流程

FD:



FR:



6 舉例

Sample code 舉例：\MI\sample\fd