ArcSoft Gender Estimation

开发指导文档

ArcSoft Corporation
46601 Fremont Blvd.
Fremont, CA 94538
http://www.arcsoft.com

Trademark or Service Mark Information

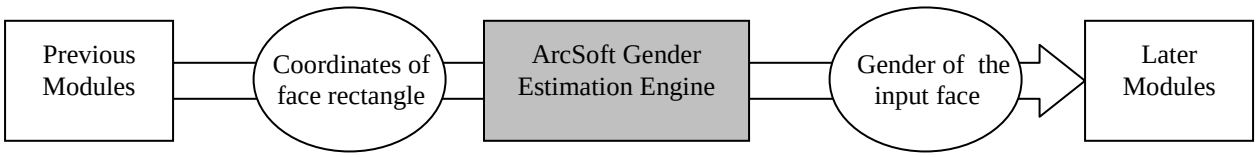
ArcSoft Inc. and ArcWare are registered trademarks of ArcSoft Inc.

Other product and company names mentioned herein may be trademarks and/or service marks of their respective owners. The absence of a trademark or service mark from this list does not constitute a waiver of ArcSoft Inc.'s trademark or other intellectual property rights concerning that trademark or service mark.

The information contained in this document is for discussion purposes only. None of the information herein shall be interpreted as an offer or promise to any of the substance herein nor as an agreement to contract or license, or as an implication of a transfer of rights. Any and all terms herein are subject to change at the discretion of ArcSoft. Copying, distributing, transferring or any other reproduction of these documents or the information contained herein is expressly prohibited, unless such activity is expressly permitted by an authorized representative of ArcSoft, Inc.

Chapter 1: 概述

虹软性别检测引擎工作流程图.



1.1. 运行环境

- iOS armv7, arm64

1.2. 系统要求

- 支持 iOS 8.x 及以上

1.3. 依赖库

- 虹软平台库

注: 请把虹软平台库的头文件 (在 SDK 包的“platform”目录下)放入您的开发工程里面。

Chapter 2: 结构与常量

2.1. 基本类型

所有基本类型在平台库中有定义。 定义规则是在 ANSIC 中的基本类型前加上字母“M”同时将类型的第一个字母改成大写。例如“long” 被定义成“MLong”

2.2. 数据结构

2.2.1. ASGE_FSDK_Version

描述

SDK 版本信息

定义

```
typedef struct{
	MInt32 lCodebase;
	MInt32 lMajor;
	MInt32 lMinor;
	MInt32 lBuild;
	MPChar Version;
	MPChar BuildDate;
	MPChar CopyRight;
} ASGE_FSDK_Version;
```

成员变量

lCodebase	代码库版本号
lMajor	主版本号
lMinor	次版本号
lBuild	编译版本号，递增
Version	字符串形式的版本号
BuildDate	编译时间
CopyRight	版权信息

2.2.2. ASGE_FSDK_GENDERRESULT

描述

定义性别检测结果信息

定义

```
typedef struct{
       MInt32        *pGenderResultArray;
       MInt32        lFaceNumber;
} ASGE_FSDK_GENDERRESULT, *LPASGE_FSDK_GENDERRESULT;
```

成员变量

pGenderResultArray	检测出的性别结果数组
lFaceNumber	检测出的性别结果个数

2.2.3. ASGE _FSDK_GENDERFACEINPUT

描述

定义脸部信息

定义

```
Typedef struct{
       MRECT         *pFaceRectArray;
       MInt32        *pFaceOrientArray;
       MInt32        lFaceNumber;
} ASGE_FSDK_GENDERFACEINPUT, *LPASGE_FSDK_GENDERFACEINPUT;
```

成员变量

pFaceRectArray	人脸框信息数组
pFaceOrientArray	输入的人脸角度数组。
lFaceNumber	人脸个数

2.3. 枚举

2.3.1. ASGE_FSDK_GenderOrientCode

描述

定义基于逆时针方向的人脸角度

定义

```
enum ASGE_FSDK_GenderOrientCode {
              ASGE_FSDK_FOC_Gender_0     = 0x1,
              ASGE_FSDK_FOC_Gender_90    = 0x2,
```

```
    ASGE_FSDK_FOC_Gender_270   = 0x3,
    ASGE_FSDK_FOC_Gender_180   = 0x4,
    ASGE_FSDK_FOC_Gender_30    = 0x5,
    ASGE_FSDK_FOC_Gender_60    = 0x6,
    ASGE_FSDK_FOC_Gender_120   = 0x7,
    ASGE_FSDK_FOC_Gender_150   = 0x8,
    ASGE_FSDK_FOC_Gender_210   = 0x9,
    ASGE_FSDK_FOC_Gender_240   = 0xa,
    ASGE_FSDK_FOC_Gender_300   = 0xb,
    ASGE_FSDK_FOC_Gender_330   = 0xc
};
```

成员变量

ASGE_FSDK_FOC_Gender_0	0 度
ASGE_FSDK_FOC_Gender_90	90 度
ASGE_FSDK_FOC_Gender_270	270 度
ASGE_FSDK_FOC_Gender_180	180 度
ASGE_FSDK_FOC_Gender_30	30 度
ASGE_FSDK_FOC_Gender_60	60 度
ASGE_FSDK_FOC_Gender_120	120 度
ASGE_FSDK_FOC_Gender_150	150 度
ASGE_FSDK_FOC_Gender_210	210 度
ASGE_FSDK_FOC_Gender_240	240 度
ASGE_FSDK_FOC_Gender_300	300 度
ASGE_FSDK_FOC_Gender_330	330 度

2.4. 支持的颜色格式

描述

颜色格式及其对齐规则

定义

ASVL_PAF_NV12	8-bit Y 层，之后是 8-bit 的 2x2 采样的 U 层和 V 层
ASVL_PAF_RGB24_B8G8R8	每个像素 8-bit B，8-bit R，8-bit R

Chapter 3: API 说明

3.1. ASGE_FSDK_InitGenderEngine

原型

```
MRESULT ASGE_FSDK_ InitGenderEngine (
       MPChar                    AppId,
       MPChar                    SDKKey,
       MByte                     *pMem,
       MInt32                    lMemSize
       MHandle                   *phEngine
);
```

描述

初始化性别检测引擎

参数

AppId	[in]	用户申请 SDK 时获取的 App Id
SDKKey	[in]	用户申请 SDK 时获取的 SDK Key
pMem	[in]	分配给引擎使用的内存地址
lMemSize	[in]	分配给引擎使用的内存大小
phEngine	[out]	性别检测引擎

返回值

成功返回 MOK，否则返回失败 code。失败 codes 如下所列:

MERR_INVALID_PARAM	参数输入非法
MERR_NO_MEMORY	内存不足

3.2. ASGE_FSDK_GenderEstimation_StaticImage

原型

```
MRESULT ASGE_FSDK_GenderEstimation_StaticImage (
       MHandle                                 hEngine,
       LPASVLOFFSCREEN                         pImginfo,
       LPASGE_FSDK_GENDERFACEINPUT             pFaceRes,
       LPASGE_FSDK_GENDERRESULT                pGenderRes,
);
```

描述

检测静态图片中人物的性别

参数

hEngine	[in]	性别检测引擎
pImginfo	[in]	输入图像信息
pFaceRes	[in]	输入的图像中人脸信息， 需要事先用人脸引擎检测出
pGenderRes	[out]	性别检测结果

返回值

成功返回 MOK，否则返回失败 code。失败 codes 如下所列:

MERR_INVALID_PARAM	参数输入非法
MERR_NO_MEMORY	内存不足

3.3. ASGE_FSDK_GenderEstimation_Preview

原型

```
MRESULT ASGE_FSDK_GenderEstimation_Preview (
      MHandle                                hEngine,
      LPASVLOFFSCREEN                        pImginfo,
      LPASGE_FSDK_GENDERFACEINPUT            pFaceRes,
      LPASGE_FSDK_GENDERRESULT               pGenderRes,
);
```

描述

检测动态视频中人物的性别

参数

hEngine	[in]	性别检测引擎
pImginfo	[in]	输入的图像数据
pFaceRes	[in]	已检测到的脸部信息
pGenderRes	[out]	性别检测结果

返回值

成功返回 MOK，否则返回失败 code。失败 codes 如下所列:

MERR_INVALID_PARAM	参数输入非法
MERR_NO_MEMORY	内存不足

3.4. ASGE_FSDK_UninitGenderEngine

原型

```
MRESULT ASGE_FSDK_UninitGenderEngine (
      MHandle      hEngine
);
```

描述

销毁引擎，释放相应资源

参数

hEngine	[in]	性别检测引擎

返回值

成功返回 MOK，否则返回失败 code。失败 codes 如下所列:

MERR_INVALID_PARAM　　参数输入非法

3.5. ASGE_FSDK_GetVersion

原型

```
const ASGE_FSDK_Version* ASGE_FSDK_GetVersion(
      MHandle      hEngine
);
```

描述

获取 SDK 版本信息

参数

hEngine	[in]	性别检测引擎

Chapter 4: Sample Codes

```
#include "ammem.h"
#include "merror.h"
#import <arcsoft_fsdk_gender_estimation/arcsoft_fsdk_gender_estimation.h>

#include <stdlib.h>


#define ARC_APP_ID                ""
#define ARC_GENDER_SDK_KEY        ""
#define ARC_GENDER_MAX_FACE_NUM   5
#define ARC_GENDER_MEM_SIZE       1024*1024*30

MRESULT doGenderEstimation()
{
    // init
    MVoid* pMemBuffer = MMemAlloc(MNull, ARC_GENDER_MEM_SIZE);
    MHandle hEngine = MNull;
    MRESULT mr = ASGE_FSDK_InitGenderEngine((MPChar)ARC_APP_ID,
(MPChar)ARC_GENDER_SDK_KEY, (MByte*)pMemBuffer,
ARC_GENDER_MEM_SIZE, &hEngine);
    if (MOK != mr) {
        // check the error code
    }

    ASGE_FSDK_GENDERFACEINPUT genderFaceInput = {0};
    genderFaceInput.lFaceNumber = 0;
    genderFaceInput.pFaceRectArray = (MRECT *)MMemAlloc(MNull,
sizeof(MRECT)*ARC_GENDER_MAX_FACE_NUM);
    genderFaceInput.pFaceOrientArray = (MInt32 *)MMemAlloc(MNull,
sizeof(MInt32)*ARC_GENDER_MAX_FACE_NUM);

    // estimation
    ASVLOFFSCREEN offScreenIn = {0};              // image data, replaced with your
data
    offScreenIn.u32PixelArrayFormat = ASVL_PAF_NV12;  // image format
    offScreenIn.i32Width = 1280;                  // image width
    offScreenIn.i32Height = 720;                  // image height
    offScreenIn.pi32Pitch[0] = offScreenIn.i32Width;  // pitch of plane 0, may not be equal
to width
    offScreenIn.pi32Pitch[1] = offScreenIn.i32Width;  // pitch of plane 1, may not be equal
to width
    offScreenIn.ppu8Plane[0] = MNull;             // data address of plane 0
    offScreenIn.ppu8Plane[1] = MNull;             // data address of plane 1

    genderFaceInput.lFaceNumber = 1;              // set face number from face detection
or face tracking result
```

```
    //genderFaceInput.pFaceRectArray               // set face rect array from face
detection or face tracking result
    //genderFaceInput.pFaceOrientArray             // set face orient array from face
detection or face tracking result

    ASGE_FSDK_GENDERRESULT genderResult = {0};
    MBool previewData = MTrue;                     // do preview data or static data gender
estimation
    if(previewData)
        mr = ASGE_FSDK_GenderEstimation_Preview(hEngine, &offScreenIn,
&genderFaceInput, &genderResult);
    else
        mr = ASGE_FSDK_GenderEstimation_StaticImage(hEngine, &offScreenIn,
&genderFaceInput, &genderResult);

    // unit
    if (genderFaceInput.pFaceRectArray)
    {
        MMemFree(MNull, genderFaceInput.pFaceRectArray);
        genderFaceInput.pFaceRectArray = MNull;
    }
    if (genderFaceInput.pFaceOrientArray)
    {
        MMemFree(MNull, genderFaceInput.pFaceOrientArray);
        genderFaceInput.pFaceOrientArray = MNull;
    }
    mr = ASGE_FSDK_UninitGenderEngine(hEngine);
    if(pMemBuffer != MNull)
    {
        MMemFree(MNull,pMemBuffer);
        pMemBuffer = MNull;
    }

    return mr;
}
```