



文档名称：运动目标识别算法 API 参考手册

文档版本号：V1.0

发布日期：2017/8/22

版权所有 © 珠海全志科技股份有限公司 2017。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



、全志和其他全志商标均为珠海全志科技股份有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受全志公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，全志公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保

前言

概述

本手册将对运动目标识别 API 接口进行介绍，为正确使用运动目标识别算法库请阅读本手册。

产品版本

产品名称	产品版本
V5-100	V1.0

读者对象

本文档（本指南）主要适用于以下工程师：

- 电子产品设计开发人员

修订记录

版本号	修订日期	修订内容
V1.0	20170822	正式发布版本

目 录

1. 概述.....	1
1.1 功能描述.....	1
1.2 API 函数参考域.....	2
1.3 数据类型参考域.....	2
1.4 相关文档.....	3
2. API 参考.....	4
2.1 运动目标识别 API 参考.....	4
2.1.1 AW_AI_CVE_DTCA_GetAlgoVersion.....	5
2.1.2 AW_AI_CVE_DTCA_Init.....	5
2.1.3 AW_AI_CVE_DTCA_UnInit.....	6
2.1.4 AW_AI_CVE_DTCA_Process.....	7
2.1.5 AW_AI_CVE_DTCA_GetLastError.....	8
2.1.6 AW_AI_CVE_DTCA_GetSceneVersion.....	9
2.1.7 AW_AI_CVE_DTCA_PacketParamFile.....	9
2.1.8 AW_AI_CVE_DTCA_PacketParamString.....	10
2.1.9 AW_AI_CVE_DTCA_ParseParamFile.....	11
2.1.10 AW_AI_CVE_DTCA_ParseParamString.....	12
2.1.11 AW_AI_CVE_CLBR_GetSceneVersion.....	13
2.1.12 AW_AI_CVE_CLBR_PacketParamFile.....	13
2.1.13 AW_AI_CVE_CLBR_PacketParamString.....	14
2.1.14 AW_AI_CVE_CLBR_ParseParamFile.....	15
2.1.15 AW_AI_CVE_CLBR_ParseParamString.....	16
2.2 数据类型.....	17
2.2.1 运动目标识别算法结果数据类型.....	17
2.2.2 运动目标识别算法配置数据类型.....	22
2.2.3 景深标定算法配置数据类型.....	40
2.3 错误码.....	43
3. 开发应用.....	44
3.1 调用流程.....	45
3.2 实例介绍.....	46
4. 算法资源需求指标.....	47
4.1 算法资源需求.....	47

1. 概述

CVE 移动目标侦测模块可以有效检测车辆、行人、动物等移动物体，针对不同检测目的可以支持周界入侵、警戒线、逆行、非法停车等十几项子功能。具有功能全面，检测精度高、速度快、CPU 资源占用率低等特点。

本章描述内容如下表所示：

标题	内容
1.1 功能描述	描述运动目标检测基本功能
1.1 API 函数参考域	描述 API 参考域。
1.2 数据类型参考域	描述数据类型的参考域。
1.3 相关文档	介绍本指南相关的文档。

表 1-1 本章内容说明

1.1 功能描述

- 支持最小 8x8 像素的小目标检测
- 最大支持 720P 图像
- D1 分辨率，达到 10fps
- 单模块运行，硬件功耗小于 100mw
- 包含如下表所述子功能：

移动目标侦测功能列表	是否支持
周界入侵	支持
警戒线	支持
双警戒线	支持
逆行	支持
非正常速度	支持
徘徊	支持
多人徘徊	支持
非法停车	支持
物品遗留	支持
物品移走	支持

1.2 API 函数参考域

本文档对 API 参考信息使用以下域来描述，具体信息如表 1-2 所示。

参考域	作用
描述	描述 API 的功能。
语法	描述 API 的语法样式。
参数	列出 API 的参数、参数说明及其属性。
返回值	列出 API 的返回值及其返回值说明。
错误码	列出 API 的错误码及其错误码说明。
需求	列出本 API 要包含的头文件和 API 编译时要链接的库文件。
注意	使用 API 时应注意的事项。
举例	使用 API 的实例
相关主题	同本 API 的其他相关信息。

表 1-2 API 函数参考域说明

1.3 数据类型参考域

本文档对数据类型参考信息使用以下域来描述，具体信息如表 1-3 所示。

参考域	作用
说明	简单描述结构体所实现的功能。
定义	列出结构体的定义。
成员	列出数据结构的成员及含义。
注意事项	列出使用数据类型时应注意的事项。
相关数据类型和接口	列出与本数据类型相关联的其他数据类型和接口。
需求	列出本 API 要包含的头文件和 API 编译时要链接的库文件。
注意	使用 API 时应注意的事项。
举例	使用 API 的实例
相关主题	同本 API 的其他相关信息。

表 1-3 数据类型参考域说明

1.4 相关文档

- 《运动目标识别用户手册》

2. API 参考

关于本章

本章描述内容如下表所示：

标题	内容
2.1 API 参考	描述运动目标识别算法 API 函数参考
2.2 数据类型	描述运动目标识别算法相关数据类型

表 2-1 本章内容说明

2.1 运动目标识别 API 参考

运动目标识别函数列表	功能描述
AW_AI_CVE_DTCA_GetAlgoVersion	查询算法版本
AW_AI_CVE_DTCA_Init	创建算法模块并初始化
AW_AI_CVE_DTCA_UnInit	终止算法模块并释放内存
AW_AI_CVE_DTCA_Process	处理一幅图像，并获取处理结果
AW_AI_CVE_DTCA_GetLastError	获取最后的错误码结果
AW_AI_CVE_DTCA_GetSceneVersion	查询目标识别算法参数配置版本
AW_AI_CVE_DTCA_PacketParamFile	将目标识别算法处理参数打包为 XML 文件
AW_AI_CVE_DTCA_PacketParamString	将目标识别算法处理参数打包为 XML 字符串
AW_AI_CVE_DTCA_ParseParamFile	从 XML 文件解析目标识别算法处理参数
AW_AI_CVE_DTCA_ParseParamString	从 XML 字符串解析目标识别算法处理参数
AW_AI_CVE_CLBR_GetSceneVersion	查询景深标定算法参数配置版本
AW_AI_CVE_CLBR_PacketParamFile	将景深标定算法处理参数打包为 XML 文件
AW_AI_CVE_CLBR_PacketParamString	将景深标定算法处理参数打包为 XML 字符串

tring	XML 字符串
AW_AI_CVE_CLBR_ParseParamFile	从 XML 文件解析景深标定算法处理参数
AW_AI_CVE_CLBR_ParseParamString	从 XML 字符串解析景深标定算法处理参数

表 2-2 运动目标识别 API 函数列表

2.1.1 AW_AI_CVE_DTCA_GetAlgoVersion

【描述】

查询运动目标识别算法版本。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_GetAlgoVersion( AW_CHAR *pVersion );
```

【参数】

参数名称	描述	输入/输出
pVersion	版本号字符串	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功
AW_STATUS_ERROR	失败

【需求】

头文件: aw_ai_cve_dtca_interface.h

【注意】

无

【举例】

无

【相关主题】

无

2.1.2 AW_AI_CVE_DTCA_Init

【描述】

创建算法模块并初始化。

【语法】

```
AW_HANDLE AW_AI_CVE_DTCA_Init( AW_AI_CVE_DTCA_PARAM *pstParamDTCA,  
AW_AI_CVE_CLBR_PARAM *pstParamCLBR );
```

【参数】

参数名称	描述	输入/输出
pstParamDTCA	目标识别算法参数	输入
pstParamCLBR	景深标定算法参数	输入

【返回值】

返回值	描述
AW_HANDLE	算法模块句柄

【错误码】

接口返回值	描述
AW_STATUS_OK	成功
AW_STATUS_ERROR	失败

【需求】

头文件：aw_ai_cve_dtca_interface.h

【注意】

初始化输入宽度和高度参数需要与 Process 处理图像的宽度和高度保持一致；
宽度和高度支持 D1、CIF、QCIF 三种分辨率格式，支持 N 和 P 制式，建议使用 QCIF；
支持自定义宽度和高度；
如返回 NULL，请调用 AW_AI_CVE_DTCA_GetLastError 获取错误码

【举例】

无

【相关主题】

无

2.1.3 AW_AI_CVE_DTCA_UnInit

【描述】

终止算法模块并释放内存。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_UnInit (AW_HANDLE hModule,  
AW_AI_CVE_DTCA_RESULT *pstResult );
```

【参数】

参数名称	描述	输入/输出
hModule	算法模块句柄	输入
pstResult	算法处理结果	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_interface.h

【注意】

必须在 AW_AI_CVE_DTCA_Init 调用成功后调用。

如返回失败请调用 AW_AI_CVE_DTCA_GetLastError 获取错误码

【举例】

无

【相关主题】

无

2.1.4 AW_AI_CVE_DTCA_Process

【描述】

处理一幅图像，并获取处理结果。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_Process( AW_HANDLE hModule, AW_IMAGE_S
*pstImage, AW_U32 u32TimeStamp, AW_AI_CVE_DTCA_RESULT *pstResult );
```

【参数】

参数名称	描述	输入/输出
hModule	算法模块句柄	输入
pstImage	YUV 彩色图像	输入
u32TimeSta	以帧为单位的时间戳	输入

mp		
pstResult	算法处理结果	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_SKIP	跳帧。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_interface.h

【注意】

pstImage 需要事先分配好内存空间；

pstImage 的宽度和高度需要与 AW_AI_CVE_DTCA_Init 的宽度和高度保持一致；

如返回失败请调用 AW_AI_CVE_DTCA_GetLastError 获取错误码；

【举例】

无

【相关主题】

无

2.1.5 AW_AI_CVE_DTCA_GetLastError

【描述】

获取最后的错误码结果。

【语法】

```
AW_ERROR_CODE_E AW_AI_CVE_DTCA_GetLastError( AW_HANDLE *hModule );
```

【参数】

参数名称	描述	输入/输出
hModule	算法模块句柄	输入

【返回值】

返回值	描述
-----	----

AW_ERROR_CODE_E

算法错误码

【需求】

头文件：aw_ai_cve_dtca_interface.h

【注意】

1、错误码信息详见 3.3 节。【举例】

【相关主题】

无

2.1.6 AW_AI_CVE_DTCA_GetSceneVersion

【描述】

查询目标识别算法参数配置版本。

【语法】

AW_STATUS_E CVE_DTCA_GetSceneVersion(AW_CHAR *pVersion);

【参数】

参数名称	描述	输入/输出
pVersion	版本号字符串	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_config.h

【注意】

无。

【举例】

无

【相关主题】

无

2.1.7 AW_AI_CVE_DTCA_PacketParamFile

【描述】

版权所有 侵权必究

Copyright © by Allwinner. All rights reserved

将目标识别算法处理参数打包为 XML 文件。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_PacketParamFile( AW_AI_CVE_DTCA_PARAM_ALGO
*pParams, AW_CHAR *pFilNam, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入/输出
pParams	参数数据	输入
pFilNam	参数文件名	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_config.h

【注意】

无。

【举例】

无

【相关主题】

无

2.1.8 AW_AI_CVE_DTCA_PacketParamString

【描述】

将目标识别算法处理参数打包为 XML 字符串。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_PacketParamString( AW_AI_CVE_DTCA_PARAM_ALGO
*pParams, AW_CHAR *pBuffer, AW_S32 nBufLen, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入/输出
pParams	参数数据	输入

pBuffer	字符串缓冲区	输出
nBufLen	缓冲区长度	输入

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_config.h

【注意】

无。

【举例】

无

【相关主题】

无

2.1.9 AW_AI_CVE_DTCA_ParseParamFile

【描述】

从 XML 文件解析目标识别算法处理参数。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_ParseParamFile( AW_CHAR *pFilNam,
AW_AI_CVE_DTCA_PARAM_ALGO *pParams, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入/输出
pFilNam	参数文件名	输入
pParams	参数数据	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_config.h

【注意】

无

【举例】

无

【相关主题】

无

无

2.1.10 AW_AI_CVE_DTCA_ParseParamString

【描述】

从 XML 字符串解析目标识别算法处理参数。

【语法】

```
AW_STATUS_E AW_AI_CVE_DTCA_ParseParamString( AW_CHAR *pXmlString,
AW_AI_CVE_DTCA_PARAM_ALGO *pParams, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入/输出
pXmlString	XML 字符串	输入
pParams	参数数据	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_dtca_config.h

【注意】

无

【举例】

无

【相关主题】

无

2.1.11 AW_AI_CVE_CLBR_GetSceneVersion

【描述】

查询景深标定算法参数配置版本。

【语法】

```
AW_STATUS_E AW_AI_CVE_CLBR_GetSceneVersion(AW_CHAR *pVersion );
```

【参数】

参数名称	描述	输入/输出
pVersion	版本号字符串	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_clbr_config.h

【注意】

无。

【举例】

无

【相关主题】

无

2.1.12 AW_AI_CVE_CLBR_PacketParamFile

【描述】

将景深标定算法处理参数打包为 XML 文件。

【语法】

AW_STATUS_E AW_AI_CVE_CLBR_PacketParamFile(AW_AI_CVE_CLBR_PARAM *pParams,
AW_CHAR *pFilNam, AW_PVOID pMemMgr);

【参数】

参数名称	描述	输入/输出
pParams	参数数据	输入
pFilNam	参数文件名	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_clbr_config.h

【注意】

无

【举例】

无

【相关主题】

无

2.1.13 AW_AI_CVE_CLBR_PacketParamString

【描述】

将景深标定算法处理参数打包为 XML 字符串。

【语法】

AW_STATUS_E AW_AI_CVE_CLBR_PacketParamString(AW_AI_CVE_CLBR_PARAM
*pParams, char *pBuffer, AW_S32 nBufLen, AW_PVOID pMemMgr);

【参数】

参数名称	描述	输入/输出
pParams	参数数据	输入
pBuffer	字符串缓冲区	输出

nBufLen	缓冲区长度	输入
---------	-------	----

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件: aw_ai_cve_clbr_config.h

【注意】

无。

【举例】

无

【相关主题】

无

2.1.14 AW_AI_CVE_CLBR_ParseParamFile

【描述】

从 XML 文件解析景深标定算法处理参数。

【语法】

```
AW_STATUS_E AW_AI_CVE_CLBR_ParseParamFile( AW_CHAR *pFilNam,
AW_AI_CVE_CLBR_PARAM *pParams, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入/输出
pFilNam	参数文件名	输入
pParams	参数数据	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
-------	----

AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_clbr_config.h

【注意】

无

【举例】

无

【相关主题】

无

2.1.15 AW_AI_CVE_CLBR_ParseParamString

【描述】

从 XML 字符串解析景深标定算法处理参数。

【语法】

```
AW_STATUS_E AW_AI_CVE_CLBR_ParseParamString( AW_CHAR *pXmlString,
CVE_CLBR_PARAM *pParams, AW_PVOID pMemMgr );
```

【参数】

参数名称	描述	输入 / 输出
pXmlString	XML 字符串	输入
pParams	参数数据	输出

【返回值】

返回值	描述
AW_STATUS_E	处理状态

【错误码】

接口返回值	描述
AW_STATUS_OK	成功。
AW_STATUS_ERROR	失败。

【需求】

头文件：aw_ai_cve_clbr_config.h

【注意】

无

【举例】

无

【相关主题】

无

2.2 数据类型

2.2.1 运动目标识别算法结果数据类型

数据结构列表	功能描述
AW_AI_CVE_DTCA_RESULT	算法处理结果
AW_TARGET_SET_S	目标集合
AW_TARGET_S	目标数据
AW_TGT_TYPE_E	目标类型
AW_TGT_TRAJECT_S	目标轨迹
AW_EVENT_SET_S	事件集合
AW_EVENT_S	事件数据
AW_EVT_TYPE	事件类型
AW_EVT_STATUS	事件状态

AW_AI_CVE_DTCA_RESULT

结构定义			
<pre>typedef struct { AW_TARGET_SET_S stTargetSet; AW_EVENT_SET_S stEventSet; } AW_AI_CVE_DTCA_RESULT;</pre>			
结构说明			
变量	名称	含义	取值范围
stTargetSet	目标集合	目标结果数据	参考 AW_TARGET_SET_S

stEventSet	事件集合	事件结果数据	参 考 AW_EVENT_SET_S
------------	------	--------	-----------------------

AW_TARGET_SET

结构定义			
<pre>typedef struct { AW_S32 s32TargetNum; AW_TARGET_S astTargets[AW_MAX_TGT_NUM]; } AW_TARGET_SET;</pre>			
结构说明			
变量	名称	含义	取值范围
s32TargetNum	目标数目	有效的目标数目	0~64
astTargets	目标数组	用于存储目标数据	参考 AW_TARGET_S

AW_TARGET_S

结构定义			
<pre>typedef struct { AW_U32 u32ID; AW_S32 u32Type; AW_POINT_S stPoint; AW_RECT_S stRect; AW_S32 s32AreaPix; AW_S32 s32Area; AW_S32 s32Direct; AW_FLOAT fSpeed; AW_TGT_TRAJECT_S stTraject } AW_TARGET_S;</pre>			
结构说明			
变量	名称	含义	取值范围
u32ID	目标 ID	目标身份标识	32 位整数
u32Type	目标类型	目标的类型	参 考 AW_TGT_TYPE_E
stPoint	目标位置	目标质心的图像坐标	图像像素坐标

stRect	目标矩形框	目标的外接矩形框	图像像素坐标
s32AreaPlx	目标面积 (像素)	目标的像素面积	单位: 像素
s32Area	目标面积 (实际)	目标的实际面积	单位: 平方厘米
s32Direct	运动方向	目标的运动方向	单位: 角度
fSpeed	运动速度	目标的运动速度	单位: 千米/小时
stTraject	目标轨迹	目标的运动路径	参考 AW_TGT_TRAJECT_S

AW_TGT_TYPE_E

结构定义			
<pre>typedef enum { AW_TGT_TYPE_UNKNOWN = 0, AW_TGT_TYPE_HUMAN = 1, AW_TGT_TYPE_VEHICLE = 2 } AW_TGT_TYPE;</pre>			
结构说明			
变量	名称	含义	取值范围
AW_TGT_TYPE_HUMAN	人	目标类型为车	1
AW_TGT_TYPE_VEHICLE	车	目标类型为车	2

AW_TGT_TRAJECT

结构定义			
<pre>typedef struct { AW_S32 length; AW_POINT_S points[AW_MAX_TRAJECT_LEN]; } AW_TGT_TRAJECT_S;</pre>			
结构说明			
变量	名称	含义	取值范围
length	轨迹点数	有效的轨迹点数目	1~40

	目		
points	轨迹点数组	用于存储轨迹点数组	图像像素坐标

AW_EVENT_SET_S

结构定义			
<pre>typedef struct { AW_S32 s32EventNum; AW_EVENT_S astEvents[AW_MAX_EVT_NUM]; } AW_EVENT_SET;</pre>			
结构说明			
变量	名称	含义	取值范围
s32EventNum	事件数目	有效的事件数目	0~128
astEvents	事件数组	用于存储事件数据	参考 AW_EVENT_S

AW_EVENT_S

结构定义			
<pre>typedef struct { AW_U32 u32Type; AW_U32 u32ID; AW_U32 u32Status; AW_U32 u32Zone; AW_U32 u32TgtID; AW_U8 u8Data[16]; } AW_EVENT_S;</pre>			
结构说明			
变量	名称	含义	取值范围
u32Type	事件类型	事件所属类型	参考 AW_AI_CVE_EVT_TYPE_E
u32ID	事件标识	事件身份标识	32 位整数
u32Status	事件状态	事件所处状态	参考 AW_EVT_STATUS

u32Zone	事件发生区域	发生此事件的区域编号	32 位整数
u32TgtID	目标标识	关联目标的标识	32 位整数
u8Data	事件私有数据	与类型相关的事件数据	

AW_AI_CVE_EVT_TYPE_E

结构定义			
<pre>typedef enum { AW_AI_CVEEVT_TYPE_UNKNOWN = 0000, AW_AI_CVEEVT_TYPE_PERIMETER = 1000, AW_AI_CVEEVT_TYPE_LOITER = 1001, AW_AI_CVEEVT_TYPE_ABNORMALVEL = 1002, AW_AI_CVEEVT_TYPE_CONVERSE = 1003, AW_AI_CVEEVT_TYPE_TRIPWIRE = 1004, AW_AI_CVEEVT_TYPE_MTRIPWIRE = 1005, AW_AI_CVEEVT_TYPE_OBJLEFT = 1006, AW_AI_CVEEVT_TYPE_OBJREMOVED = 1007, AW_AI_CVEEVT_TYPE_MLOITER = 1008, AW_AI_CVEEVT_TYPE_NOPARKING = 1009, AW_AI_CVEEVT_TYPE_SIGNAL_BAD = 2001 } AW_AI_CVE_EVT_TYPE_E;</pre>			
结构说明			
变量	名称	含义	取值范围
AW_AI_CVEEVT_TYPE_PERIMETER		周界入侵	1000
AW_AI_CVEEVT_TYPE_LOITER		徘徊	1001
AW_AI_CVEEVT_TYPE_ABNORMALVEL		非正常速度	1002
AW_AI_CVEEVT_TYPE_CONVERSE		逆行	1003
AW_AI_CVEEVT_TYPE_TRIPWIRE		绊线	1004
AW_AI_CVEEVT_TYPE_MTRIPWIRE		双绊线	1005

IRE			
AW_AI_CVEEVT_TYPE_OBJLEF T		物品遗留	1006
AW_AI_CVEEVT_TYPE_OBJREM OVED		物品移走	1007
AW_AI_CVEEVT_TYPE_MLOITE R		多人徘徊	1008
AW_AI_CVEEVT_TYPE_NOPARK ING		非法停车	1009
AW_AI_CVEEVT_TYPE_SIGNAL _BAD		视频信号异常	2001

AW_EVT_STATUS_E

结构定义			
<pre>typedef enum { AW_EVT_STATUS_NOSTA = 0, AW_EVT_STATUS_START = 1, AW_EVT_STATUS_END = 2 } AW_EVT_STATUS_E;</pre>			
结构说明			
变量	名称	含义	取值范围
AW_EVT_STATUS _NOSTA	无状态	无状态事件	0
AW_EVT_STATUS _START	开始	事件开始	1
AW_EVT_STATUS _END	结束	事件结束	2

2.2.2 运动目标识别算法配置数据类型

数据结构列表	功能描述
AW_AI_CVE_DTCA_PARAM	目标识别算法参数
AW_AI_CVE_DTCA_PARAM_I NOUT	算法 IO 参数
AW_AI_CVE_DTCA_PARAM_	算法处理参数

ALGO	
AW_AI_CVE_DTCA_PARAM_APP	应用参数
AW_AI_CVE_DTCA_PARAM_ADV	高级参数

AW_AI_CVE_DTCA_PARAM

结构定义			
<pre>typedef struct { AW_AI_CVE_DTCA_PARAM_INOUT cParamInout; AW_AI_CVE_DTCA_PARAM_ALGO cParamAlgo; } AW_AI_CVE_DTCA_PARAM;</pre>			
结构说明			
变量	名称	含义	取值范围
cParamInout	算法 IO 参数	与图像输入输出相关的参数。	参考 AW_AI_CVE_DTCA_PARAM_INOUT
cParamAlgo	算法处理参数	与算法配置相关的参数。	参考 AW_AI_CVE_DTCA_PARAM_ALGO

AW_AI_CVE_DTCA_PARAM_INOUT

结构定义			
<pre>typedef struct { AW_S32 s32Width; AW_S32 s32Height; AW_U32 u32FrameTime; } AW_AI_CVE_DTCA_PARAM_INOUT;</pre>			
结构说明			
变量	名称	含义	取值范围
s32Width	图像宽度	所配置视频图像水平分辨率	16~1920
s32Height	图像高度	所配置视频图像垂直分辨率	16~1080
u32FrameTime	图像帧时间间隔	图像帧的时间长度（单位：毫秒）	33, 40

AW_AI_CVE_DTCA_PARAM_ALGO

结构定义			
<pre>typedef struct { AW_AI_CVE_DTCA_PARAM_APP cParamApp; AW_S32 s32UseAdvParam; AW_AI_CVE_DTCA_PARAM_ADV cParamAdv; } AW_AI_CVE_DTCA_PARAM_ALGO;</pre>			
结构说明			
变量	名称	含义	取值范围
cParamApp	应用参数	用来设置分析功能、目标输出、视频异常检测、电子稳定等参数	参考 AW_AI_CVE_DTCA_PARAM_APP
s32UseAdvParam	高级参数启用开关	设置是否使用高级参数	0(否), 1(是); 缺省值: 0
cParamAdv	高级参数	算法高级参数配置	参考 AW_AI_CVE_DTCA_PARAM_ADV

AW_AI_CVE_DTCA_PARAM_ADV

结构定义			
<pre>typedef struct { // 背景建模 AW_S32 bgm_period; AW_S32 bgm_period_time; AW_S32 bgm_period_noise; AW_S32 sensi_high_noise; AW_S32 sensi_low_contrast; AW_S32 bgm_debug; AW_S32 bgm_reserved[2]; } AW_AI_CVE_DTCA_PARAM_ADV;</pre>			
结构说明			
变量	名称	含义	取值范围
背景建模			

bgm_multiple	是否启用 复杂背景建模	在复杂场景（树枝晃动、 水波荡漾等）下启用多模态背 景。	0(否), 1(是); 缺省值: 1
bgm_period	是否启用 周期运动背景 建模	在有往复运动的场景中 启用周期背景建模。	0(否), 1(是); 缺省值: 0
bgm_period_time	周期运动 时间	往复运动的周期时间。	1~60 秒; 缺省值: 15
bgm_period_noise	周期运动 噪声	在图像噪声较大时, 开启 此机制。	0(低), 1(高); 缺省值: 0
sensi_high_noise	是否高噪 声	在图像噪声较大时打开 此开关。	0(否), 1(是); 缺省值: 0
sensi_low_contrast	是否低对 比度	在图像对比度较低时打 开此开关。	0(否), 1(是); 缺省值: 0
bgm_debug	调试开关	算法内部调试开关	0(否), 1(是); 缺省值: 0
bgm_reserved	保留字段	保留字段	

AW_AI_CVE_DTCA_PARAM_APP

结构定义			
<pre>typedef struct { AW_S32 dim_w; AW_S32 dim_h; AW_AI_CVE_VANA_FUNCTION functions; AW_AI_CVE_PARAM_PERFORMANCE performance; } AW_AI_CVE_DTCA_PARAM_APP;</pre>			
结构说明			
变量	名称	含义	取值范围
dim_w	配置图像 的宽度	用户进行场景标定图像 的宽度	16~1920
dim_h	配置图像 的宽度	用户进行场景标定图像 的高度	16~1080
functions	分析功能 参数集合	用户设定分析功能相关 参数	参考 AW_AI_CVE_VANA_ FUNCTION

performa nce	性能参数	用于调节算法性能的参 数	参 考 AW_AI_CVE_PARA_PERFORM ANCE
-----------------	------	-----------------	--

AW_AI_CVE_PARA_PERFORMANCE

结构定义			
<pre>typedef struct { AW_S32 do_istab; AW_S32 scene_mode; AW_S32 sensitivity; } AW_AI_CVE_PARA_PERFORMANCE;</pre>			
结构说明			
变量	名称	含义	取值范围
do_istab	启用图像电子 稳定	决定是否启用图像电子稳 定。	0(否), 1(是); 缺省值: 0
scene_mo de	场景模式	应用场景是室内还是室 外。	0(室外), 1(室内); 缺 省值: 0)
sensitivity	灵敏度	灵敏度越高, 目标输出越 快。	0: 高, 1: 中, 2: 低; 缺省值: 1

AW_AI_CVE_VANA_FUNCTION

结构定义			
<pre>typedef struct { AW_AI_CVE_VANA_PARAM_S defparam; AW_S32 num_used; AW_AI_CVE_VANA_FUNC funcs[AW_AI_CVE_VANA_FUNC_MAXNUM]; } AW_AI_CVE_VANA_FUNCTION;</pre>			
结构说明			
变量	名称	含义	取值范围
defpara m	缺省功能参数		参 考 AW_AI_CVE_VANA_PARA M_S
num_use d	有效功能数目	设定有效功能(事件规则) 的数目。	支持最大 32 个功能

funcs	分析功能数组	分析功能数组，存储设定相关功能的参数	参 考 AW_AI_CVE_VANA_FUNC
-------	--------	--------------------	----------------------------

AW_AI_CVE_VANA_FUNC

结构定义			
<pre>typedef struct { AW_S8 name[32]; AW_U32 enable; AW_U32 level; AW_U32 mode; AW_AI_CVE_VANA_PARAM param; } AW_AI_CVE_VANA_FUNC;</pre>			
结构说明			
变量	名称	含义	取值范围
name	规则名称	设定规则功能的名称	0(否), 1(是); 缺省值: 1
enable	是否使能	规则功能是否开启	0 (关闭), 1 (开启); 缺省值: 0
level	警报级别	设定规则警报级别	参 考 AW_AI_CVE_ALARM_LEVEL_E
mode	分析模式	规则启用分析的事件类型，对应算法具体应用功能。	参 考 AW_AI_CVE_DTCA_FUNCODE
param	分析参数	设定规则功能的分析参数	参 考 AW_AI_CVE_VANA_PARAM

AW_AI_CVE_ALARM_LEVEL_E

结构定义			
<pre>typedef enum { AW_AI_CVE_ALARM_LEVEL_ALARM = 0, AW_AI_CVE_ALARM_LEVEL_PREALARM = 1 } AW_AI_CVE_ALARM_LEVEL_E;</pre>			
结构说明			
变量	名称	含义	取值范围
AW_AI_CVE_ALARM_LEVEL_		报警	0

ALARM			
AW_AI_CVE_ALARM_LEVEL_ PREALARM		预警	1

AW_AI_CVE_DTCA_FUNCODE

结构定义			
<pre>typedef enum { AW_AI_CVE_DTCA_FUNC_PERIMETER = 0x00000001, AW_AI_CVE_DTCA_FUNC_LOITER = 0x00000002, AW_AI_CVE_DTCA_FUNC_CONVERSE = 0x00000008, AW_AI_CVE_DTCA_FUNC_TRIPWIRE = 0x00000010, AW_AI_CVE_DTCA_FUNC_MTRIPWIRE = 0x00000020, AW_AI_CVE_DTCA_FUNC_ABNORMALVEL = 0x00000004, AW_AI_CVE_DTCA_FUNC_MLOITER = 0x00010000, AW_AI_CVE_DTCA_FUNC_NOPARKING = 0x00020000, AW_AI_CVE_DTCA_FUNC_OBJLEFT = 0x00000040, AW_AI_CVE_DTCA_FUNC_OBJREMOVED = 0x00000080, } AW_AI_CVE_DTCA_FUNCODE;</pre>			
结构说明			
变量	名称	含义	取值范围
AW_AI_CVE_DTCA_FUNC_PERI METER		周界入侵	0x00000001
AW_AI_CVE_DTCA_FUNC_LOIT ER		徘徊	0x00000002
AW_AI_CVE_DTCA_FUNC_CON VERSE		逆行	0x00000008
AW_AI_CVE_DTCA_FUNC_TRIP WIRE		绊线	0x00000010
AW_AI_CVE_DTCA_FUNC_MTRI PWIRE		双绊线	0x00000020
AW_AI_CVE_DTCA_FUNC_ABN ORMALVEL		非正常速度	0x00000004
AW_AI_CVE_DTCA_FUNC_MLO ITER		多人徘徊	0x00010000

AW_AI_CVE_DTCA_FUNC_NOPARKING		非法停车	0x00020000
AW_AI_CVE_DTCA_FUNC_OBJECTLEFT		物品遗留	0x00000040
AW_AI_CVE_DTCA_FUNC_OBJECTREMOVED		物品移走	0x00000080

AW_AI_CVE_VANA_PARAM_S

结构定义			
<pre>typedef struct { AW_AI_CVE_VANA_PARAM_PERIMETER perimeter; AW_AI_CVE_VANA_PARAM_TRIPWIRE tripwire; AW_AI_CVE_VANA_PARAM_MTRIPWIRE mtripwire; AW_AI_CVE_VANA_PARAM_LOITER loiter; AW_AI_CVE_VANA_PARAM_CONVERSE converse; AW_AI_CVE_VANA_PARAM_ABNORMALVEL abnormalvel; AW_AI_CVE_VANA_PARAM_MLOITER mloiter; AW_AI_CVE_VANA_PARAM_NOPARKING noparking; AW_AI_CVE_VANA_PARAM_OBJECTLEFT objectleft; AW_AI_CVE_VANA_PARAM_OBJECTREMOVED objectremoved; } AW_AI_CVE_VANA_PARAM;</pre>			
结构说明			
变量	名称	含义	取值范围
perimeter	周界入侵	人或者车进入禁止区域后报警。	参考 AW_AI_CVE_VANA_PARAM_PERIMETER
tripwire	绊线	当运动目标以特定方向穿越设置的警戒线，则告警。	参考 AW_AI_CVE_VANA_PARAM_TRIPWIRE
mtripwire	双绊线	当运动目标的轨迹在设定时间内先后穿过两条警戒线，则告警。	参考 AW_AI_CVE_VANA_PARAM_MTRIPWIRE
loiter	徘徊	在告警监视区域内，单一目标或连续存在的目标滞	参考 AW_AI_CVE_VANA_PARAM_LO

		留时间过长报警。	ITER
converse	逆行	在设置的告警监视区，运动目标的行进轨迹与规定方向相违背，则告警。	参考 AW_AI_CVE_VANA_PARAM_CONVERSE
abnormalvel	非正常速度	在设置告警监视区，出现速度异常（过快、过慢等）的目标，则告警。	参考 AW_AI_CVE_VANA_PARAM_ABNORMALVE
mloiter	多人徘徊	在设置告警监视区内有符合条件的人数在徘徊，则告警。	参考 AW_AI_CVE_VANA_PARAM_MLOITER
noparking	非法停车	车辆停止在禁止区域，则告警。	参考 AW_AI_CVE_VANA_PARAM_NOPARKING
objectleft	物品遗留	在设置的告警监视区，出现新的且其位置稳定的物品，则触发告警。	参考 AW_AI_CVE_VANA_PARAM_OBJECTLEFT
objectremoved	物品移走	在设置的告警监视区内，物品的位置改变或是物品消失，则触发告警。	参考 AW_AI_CVE_VANA_PARAM_OBJECTREMOVED

AW_AI_CVE_VANA_PARAM

结构定义	
<pre>typedef union { AW_AI_CVE_VANA_PARAM_PERIMETER perimeter; AW_AI_CVE_VANA_PARAM_TRIPWIRE tripwire; AW_AI_CVE_VANA_PARAM_MTRIPWIRE mtripwire; AW_AI_CVE_VANA_PARAM_LOITER loiter; AW_AI_CVE_VANA_PARAM_CONVERSE converse; AW_AI_CVE_VANA_PARAM_ABNORMALVEL abnormalvel; AW_AI_CVE_VANA_PARAM_MLOITER mloiter; AW_AI_CVE_VANA_PARAM_NOPARKING noparking; AW_AI_CVE_VANA_PARAM_OBJECTLEFT objectleft; AW_AI_CVE_VANA_PARAM_OBJECTREMOVED objectremoved; } AW_AI_CVE_VANA_PARAM;</pre>	

结构说明			
变量	名称	含义	取值范围
perimeter	周界入侵	人或者车进入禁止区域后报警。	参考 AW_AI_CVE_VANA_PARAM_PERIMETER
tripwire	绊线	当运动目标以特定方向穿越设置的警戒线，则告警。	参考 AW_AI_CVE_VANA_PARAM_TRIPWIRE
mtripwire	双绊线	当运动目标的轨迹在设定时间内先后穿过两条警戒线，则告警。	参考 AW_AI_CVE_VANA_PARAM_MTRIPWIRE
loiter	徘徊	在告警监视区域内，单一目标或连续存在的目标滞留时间过长报警。	参考 AW_AI_CVE_VANA_PARAM_LOITER
converse	逆行	在设置的告警监视区，运动目标的行进轨迹与规定方向相违背，则告警。	参考 AW_AI_CVE_VANA_PARAM_CONVERSE
abnormalvel	非正常速度	在设置告警监视区，出现速度异常（过快、过慢等）的目标，则告警。	参考 AW_AI_CVE_VANA_PARAM_ABNORMALVE
mloiter	多人徘徊	在设置告警监视区内有符合条件的人数在徘徊，则告警。	参考 AW_AI_CVE_VANA_PARAM_MLOITER
noparking	非法停车	车辆停止在禁止区域，则告警。	参考 AW_AI_CVE_VANA_PARAM_NOPARKING
objectleft	物品遗留	在设置的告警监视区，出现新的且其位置稳定的物品，则触发告警。	参考 AW_AI_CVE_VANA_PARAM_OBJECTLEFT
objectremoved	物品移走	在设置的告警监视区内，物品的位置改变或是物品消失，则触发告警。	参考 AW_AI_CVE_VANA_PARAM_OBJECTREMOVED

AW_AI_CVE_VANA_PARAM_PERIMETER

结构定义

<pre>typedef struct { AW_POLYGON_S region; AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max; } AW_AI_CVE_VANA_PARAM_PERIMETER;</pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	在此区域内的目标产生报警	图像像素坐标
type_constrain	是否限定目标类型	满足类型约束的目标产生报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constrain	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000

AW_AI_CVE_VANA_PARAM_LOITER

结构定义			
<pre>typedef struct { AW_POLYGON_S region; AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max;</pre>			

<pre> AW_S32 time_min; AW_S32 analysis_path; } AW_AI_CVE_VANA_PARAM_LOITER; </pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	在此区域内的目标产生报警	图像像素坐标
type_constrain	是否限定目标类型	满足类型约束的目标产生报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constraint	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000
time_min	闲逛最短时间	目标在分析区域内, 闲逛持续的最短时间	5~60 秒, 缺省值: 10
analysis_path	是否启用路径判断	符合迂回路径条件的目标产生报警	0(否), 1(是); 缺省值: 1

AW_AI_CVE_FORBIDDEN_INTERVAL

结构定义			
<pre> typedef struct { AW_S32 forbidden; AW_S32 minimum; AW_S32 maximum; } AW_AI_CVE_FORBIDDEN_INTERVAL; </pre>			
结构说明			
变量	名称	含义	取值范围
forbidden	禁止标志	在禁止范围内报警, 在允许范围外报警。	0: 允许; 1: 禁止

minimum	最小值	取值-1时表示不作约束	整数
maximum	最大值	取值-1时表示不作约束	整数

AW_AI_CVE_VANA_PARAM_MLOITER

结构定义			
<pre>typedef struct { AW_POLYGON_S region; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max; AW_S32 time_min; AW_S32 analysis_path; AW_AI_CVE_FORBIDDEN_INTERVAL tgtnum_interval; } AW_AI_CVE_VANA_PARAM_MLOITER;</pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	在此区域内的目标产生报警	图像像素坐标
size_constrain	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000
time_min	徘徊最短时间	目标在分析区域内, 徘徊持续的最短时间	5~60 秒, 缺省值: 10
tgtnum_interval	禁止的人数范围	徘徊的人数在此范围内则报警	参考 AW_AI_CVE_FORBIDDEN_INTERVAL

AW_AI_CVE_VANA_PARAM_ABNORMALVEL

结构定义			
<pre>typedef struct { AW_POLYGON_S region;</pre>			

<pre> AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max; AW_S32 velo_flag; AW_S32 velo_min; AW_S32 velo_max; } AW_AI_CVE_VANA_PARAM_ABNORMALVEL; </pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	在此区域内的目标产生报警	图像像素坐标
type_constrain	是否限定目标类型	满足类型约束的目标产生报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constrain	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000
velo_flag	限速标识	设定允许或者禁止速度范围的标识	0 (允许范围); 1 (禁止范围); 缺省值: 0
velo_min	最小运动速度	在检测区域目标的最小运动速度	0~1000 米/秒; 缺省值: 0
velo_max	最大运动速度	在检测区域目标的最大运动速度	0~1000 米/秒; 缺省值: 5
AW_AI_CVE_VANA_PARAM_CONVERSE			
结构定义			
<pre> typedef struct { </pre>			

<pre> AW_POLYGON_S region; AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max; AW_S32 direction; } AW_AI_CVE_VANA_PARAM_CONVERSE; </pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	在此区域内的目标产生报警	图像像素坐标
type_constrain	是否限定目标类型	满足类型约束的目标产生报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constrain	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000
direction	逆行方向角度	设定目标逆行的角度值	0~360 度

AW_AI_CVE_VANA_PARAM_TRIPWIRE

结构定义	
<pre> typedef struct { AW_AI_CVE_TRIPWIRE twline; AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; </pre>	

<pre> AW_S32 size_max; AW_S32 direction; } AW_AI_CVE_VANA_PARAM_TRIPWIRE; </pre>			
结构说明			
变量	名称	含义	取值范围
twline	绊线线段参数	用户设定绊线相关信息	参考 AW_AI_CVE_TRIPWIRE
type_constraint	是否限定目标类型	满足类型约束的目标产生报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constraint	是否限定目标尺寸	满足尺寸约束的目标产生报警	0(否), 1(是); 缺省值: 0
size_min	目标最小尺寸	报警目标的最小尺寸(单位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大尺寸	报警目标的最大尺寸(单位: 平方厘米)	0~1000000; 缺省值: 100000

AW_AI_CVE_TRIPWIRE

结构定义			
<pre> typedef struct { AW_S32 bidir; AW_S32 angle; AW_LINE_S line; } AW_AI_CVE_TRIPWIRE; </pre>			
结构说明			
变量	名称	含义	取值范围
bidir	绊线方向	绊线是否为双向绊线	0 (否), 1 (是); 缺省值: 0
angle	绊线角度	绊线禁止方向角度	0~360 度
line	绊线位置	绊线在图像中的位置	图像像素坐标

AW_AI_CVE_VANA_PARAM_MTRIPWIRE

结构定义			
------	--	--	--

<pre>typedef struct { AW_AI_CVE_MTRIPWIRE twline; AW_S32 type_constrain; AW_S32 type_mode; AW_S32 size_constrain; AW_S32 size_min; AW_S32 size_max; AW_S32 direction; } AW_AI_CVE_VANA_PARAM_MTRIPWIRE;</pre>			
结构说明			
变量	名称	含义	取值范围
twline	双绊线线 段参数	用户设定双绊线相关信息	参 考 AW_AI_CVE_MTRIPWIRE
type_constrain	是否限定 目标类型	满足类型约束的目标产生 报警	0(否), 1(是); 缺省值: 1
type_mode	类型限定 模式	目标类型限定的方式	0 (人或车), 1 (人), 2 (车); 缺省值: 0
size_constrain	是否限定 目标尺寸	满足尺寸约束的目标产生 报警	0(否), 1(是); 缺省值: 0
size_min	目标最小 尺寸	报警目标的最小尺寸(单 位: 平方厘米)	0~1000000; 缺省值: 1000
size_max	目标最大 尺寸	报警目标的最大尺寸(单 位: 平方厘米)	0~1000000; 缺省值: 100000

AW_AI_CVE_MTRIPWIRE

结构定义	
<pre>typedef struct { AW_S32 time_max; AW_S32 angles[2]; AW_LINE_S lines[2]; } AW_AI_CVE_MTRIPWIRE;</pre>	
结构说明	

变量	名称	含义	取值范围
time_max	时间间隔	穿越两条线的最长时间间隔	1~60 秒；缺省值：5
angles	绊线角度	绊线禁止方向角度	0~360 度
lines	绊线位置	绊线在图像中的位置	图像像素坐标

AW_AI_CVE_VANA_PARAM_NOPARKING

结构定义			
<pre>typedef struct { AW_POLYGON_S region; AW_S32 size_min; AW_S32 size_max; AW_S32 time_min; } AW_AI_CVE_VANA_PARAM_NOPARKING;</pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	设定的遗留检测区域	图像像素坐标
size_min	车辆最小尺寸	车辆的最小尺寸	0~1000000 平方厘米；缺省值：1000
size_max	车辆最大尺寸	车辆的最大尺寸	0~1000000 平方厘米；缺省值：1000000
time_min	车辆停留最短时间	车辆停留的最短时间	5~60 秒；缺省值：5

AW_AI_CVE_VANA_PARAM_OBJECTLEFT

结构定义			
<pre>typedef struct { AW_POLYGON_S region; AW_S32 size_min; AW_S32 size_max; AW_S32 time_min; } AW_AI_CVE_VANA_PARAM_OBJECTLEFT;</pre>			
结构说明			
变量	名称	含义	取值范围

region	分析区域	设定的遗留检测区域	图像像素坐标
size_min	物品最小尺寸	检测遗留物的最小尺寸	10~40000 平方厘米; 缺省值: 100
size_max	物品最大尺寸	检测遗留物的最大尺寸	10~40000 平方厘米; 缺省值: 10000
time_min	遗留最短时间	物品遗留的最短时间	5~60 秒; 缺省值: 5

AW_AI_CVE_VANA_PARAM_OBJECTMOVED

结构定义			
<pre>typedef struct { AW_POLYGON_S region; AW_S32 size_min; AW_S32 size_max; AW_S32 time_min; } AW_AI_CVE_VANA_PARAM_OBJECTREMOVED;</pre>			
结构说明			
变量	名称	含义	取值范围
region	分析区域	设定的遗留检测区域	图像像素坐标
size_min	物品最小尺寸	检测移走物的最小尺寸	10~40000 平方厘米; 缺省值: 100
size_max	物品最大尺寸	检测移走物的最大尺寸	10~40000 平方厘米; 缺省值: 10000
time_min	遗留最短时间	物品移走的最短时间	5~60 秒; 缺省值: 5

2.2.3 景深标定算法配置数据类型

数据结构列表	功能描述
AW_AI_CVE_CLBR_PARAM	景深标定参数
AW_AI_CVE_FDEPTH_COEF	景深系数
AW_AI_CVE_FDZ_LINES	标定线段集合

AW_AI_CVE_CLBR_PARAM

版权所有 侵权必究

Copyright © by Allwinner. All rights reserved

结构定义			
<pre>typedef struct { AW_S32 dim_w; AW_S32 dim_h; AW_AI_CVE_FDZ_LINES fdlines; AW_AI_CVE_FDEPTH_COEF coef; } AW_AI_CVE_CLBR_PARAM;</pre>			
结构说明			
变量	名称	含义	取值范围
img_w	标定图像 宽度	用户标定图像的宽度	16~1920
img_h	标定图像 高度	用户标定图像的高度	16~1080
fdlines	标定数据	标定景深线段集合	参 考 AW_AI_CVE_FDZ_LINES
coef	景深系数	换算的景深系数	参 考 AW_AI_CVE_FDEPTH_COEF

AW_AI_CVE_FDEPTH_COEF

结构定义			
<pre>typedef struct { float cx; float cy; float cf; } AW_AI_CVE_FDEPTH_COEF;</pre>			
结构说明			
变量	名称	含义	取值范围
cx	水平方向 系数	水平方向图像像素与实际 宽度换算的系数	浮点数
cy	垂直方向 系数	垂直方向图像像素与实际 高度换算的系数	浮点数
cf	偏置系数	景深换算的偏置系数	浮点数

AW_AI_CVE_FDZ_LINES

结构定义			
<pre>typedef struct { AW_S32 num_used; AW_AI_CVE_FDZ_LINE lines[AW_AI_CVE_MAXNUM_FDEPTH_LINE]; } AW_AI_CVE_FDZ_LINES;</pre>			
结构说明			
变量	名称	含义	取值范围
num_used	有效线段数目	用户标定景深时，有效的线段数目	3~8
lines	线段数组	用于存储用户标定景深，标定线段的数据	参考 AW_AI_CVE_FDZ_LINE

AW_AI_CVE_FDZ_LINE

结构定义			
<pre>typedef struct { AW_S32 type; AW_LINE_S ref_line; AW_S32 ref_length; } AW_AI_CVE_FDZ_LINE;</pre>			
结构说明			
变量	名称	含义	取值范围
type	线段类型	表示标定线段类型，是水平还是竖直	参考 AW_AI_CVE_FLINE_TYPE
ref_line	线段坐标	用于存储标定线段的坐标	图像像素坐标
ref_length	线段长度	标定线段对应实际的长度	单位：厘米

AW_AI_CVE_FDZ_TYPE

结构定义			
<pre>typedef enum { AW_AI_CVE_FLINE_HORI = 0, AW_AI_CVE_FLINE_VERT } AW_AI_CVE_FLINE_TYPE;</pre>			

结构说明			
变量	名称	含义	取值范围
AW_AI_CVE_FLI NE_HORI		水平线段	0
AW_AI_CVE_FLI NE_VERT		竖直线段	1

2.3 错误码

算法库 API 错误码如下表 2-1 所示：

错误代码	宏定义	描述
0x00000001	AW_AI_CVE_ERR_O K	算法运行正常，无错误
0xA00A000 1	AW_AI_CVE_ERR_N ULL_PTR	函数参数有空指针
0xA00A000 2	AW_AI_CVE_ERR_IL LEGAL_PARAM	参数超出合法范围
0xA00A000 3	AW_AI_CVE_ERR_IN VALID_ADDR	传递非法地址
0xA00A000 4	AW_AI_CVE_ERR_N OT_CONFIG	未进行参数配置
0xA00A000 5	AW_AI_CVE_ERR_N O_LISENCE	授权失败
0xA00A000 6	AW_AI_CVE_ERR_IN VALID_IMAGE	输入了无效的图像

表 2-3 算法库 API 错误

3. 开发应用

关于本章

本章描述内容如下表所示。

标题	内容
3.1 调用流程	介绍运动目标识别算法库的调用流程。
3.2 实例介绍	介绍运动目标识别算法的开发实例。

表 3-1 本章内容说明

AllwinnerTech

3.1 调用流程

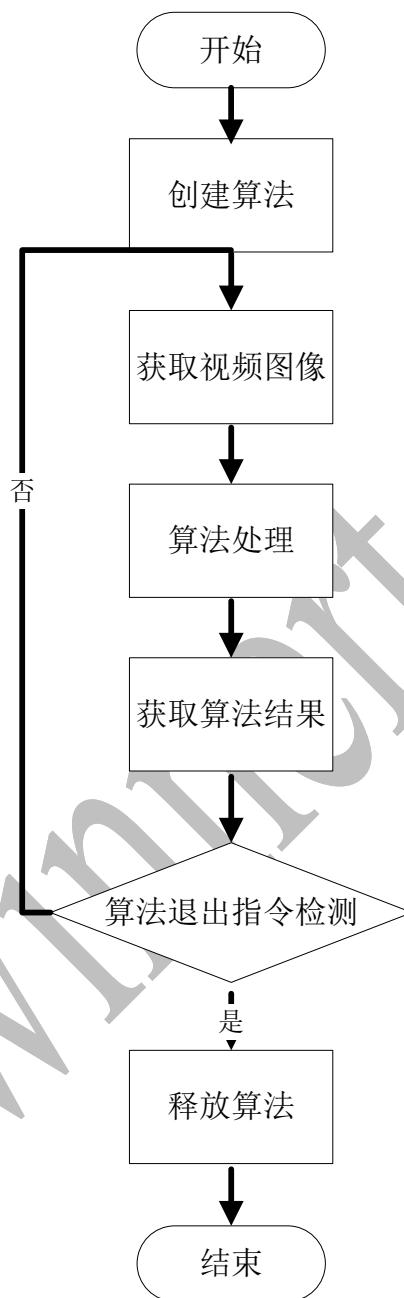


图 3-1 运动目标识别算法库的调用流程

3.2 实例介绍

本实例完成运动目标识别算法库的参数配置、接口 API 调用。

文件类型	文件列表		备注
接口头文件	aw_ai_common_type.h aw_ai_cve_base_type.h aw_ai_cve_clbr_config.h aw_ai_cve_clbr_interface.h aw_ai_cve_dtca_config.h aw_ai_cve_dtca_interface.h aw_ai_cve_dtca_result.h		
算法库文件	Linux ARM	libaw_cve_mod.a libclbr_cve.a libdtca_cve.a libdtcas_cve.a libzcvcore_cve.a libzcvxml_cve.a libzvabgfg_cve.a libzvabgfgs_cve.a libzvaclbr_cve.a libzvacore_cve.a libzvadtc_a_cve.a libzvadtcas_cve.a	
参数文件	params_cvedtca.xml, params_cveclbr.xml		
样例代码	sample_cve_MOD.c sample_vi2MOD.c		

表 3-2 运动目标识别算法实例相关文件列表及说明

4. 算法资源需求指标

关于本章

本章描述内容如下表所示。

标题	内容
4.1 算法资源需求	介绍算法所需的内存需求，代码空间需求，内存耗时指标。

表 4-1 本章内容说明

4.1 算法资源需求

下表表示算法库资源需求说明。

算法资源需求	资源占用大小			说明
内存需求	11M (CIF)			运动目标检测算法内存需求
代码空间需求	libaw_cve.a libclbr_cve.a libdtca_cve.a libdtcas_cve.a libzcvcore_cve.a libzcvxml_cve.a libzvabfg_cve.a libzvabfgfs_cve.a libzvaclbr_cve.a libzvacore_cve.a libzvadtc_cve.a libzvadtcas_cve.a	9 86k	9 86k	linuxARM 代码空间

表 4-2 运动目标检测算法资源需求说明

【参考代码】详见“sample_cve_MOD”

注意：应用时需要将 sample_cve_MOD 目录下的 DTCAParam 文件夹内的 xml 文件与生成可执行文件放置同一文件夹下，文件夹要有读写权限；同时在编译该例程时需要修改 middleware/config/mpp_config.mk 文件：MPPCFG_MOD = Y