



版本: 1.0.1 2011 年 12 月

运动检测库接口说明

声 明

本手册的版权归安凯技术公司所有，受相关法律法规的保护。未经安凯技术公司的事先书面许可，任何人不得复制、传播本手册的内容。

本手册所涉及的知识产权归属安凯技术公司所有（或经合作商授权许可使用），任何人不得侵犯。

本手册不对包括但不限于下列事项担保：适销性、特殊用途的适用性；实施该用途不会侵害第三方的知识产权等权利。

安凯技术公司不对由使用本手册或执行本手册内容而带来的任何损害负责。

本手册是按当前的状态提供参考，随附产品或本书内容如有更改，恕不另行通知。

联 系 方 式

安凯（广州）微电子有限公司

地址：广州科学城科学大道 182 号创新大厦 C1 区 3 楼

电话: (86)-20-3221 9000

传真: (86)-20-3221 9258

邮编: 510663

销售热线:

(86)-20-3221 9499

电子邮箱:

sales@anyka.com

主页:

<http://www.anyka.com>

版本变更说明

以下表格对于本文档的版本变更做一个简要的说明。版本变更仅限于技术内容的变更，不包括版式、格式、句法等的变更。

版本	说明	完成日期
V1.0.1	正式发布	2011/12/22

运动检测库接口说明与运动检测库对应的关系

Version	The corresponding motion detector lib
V1.0.1	V1.1.01
V0.0.2	V0.1.0

目录

1	模块介绍	4
1.1	功能概述	4
1.2	与其它模块关系	4
2	相关文档	4
3	集成指南	5
3.1	层次结构	5
3.2	集成步骤	5
3.2.1	获得运动检测库相关文件	5
3.2.2	实现运动检测库依赖的系统API	5
3.2.3	集成链接测试	5
3.2.4	联合调试	5
4	接口说明	6
4.1	数据结构/格式	6
4.2	接口函数列表	8
4.2.1	Motion_Detector_GetVersion	8
4.2.2	Motion_Detector_Open	8
4.2.3	Motion_Detector_Close	9
4.2.4	Motion_Detector_Handle	9
4.2.5	Motion_Detector_SetRatio	9
4.2.6	Motion_Detector_SetDimension	10
4.2.7	Motion_Detector_GetAvgLum	10
4.3	典型调用范例	11
5	依赖接口说明	13
6	常见问题	13

1 模块介绍

1.1 功能概述

本模块实现的功能为检测是否有运动物体出现。

注意：本模块仅通过对比两帧图像检测是否运动物体出现，灵敏度可通过 T_MOTION_DETECTOR_RATIO 和初始化时的阈值 m_uThreshold 调节。

T_MOTION_DETECTOR_RATIO 结构体中的各值表示运动像素点数与区域总像素的比值，以千分比为单位；设置的比值越小，检测的灵敏度越高，但如果设置太小可能会导致误判。

初始化时的阈值 m_uThreshold 越小灵敏度也越高，但如果设置太小同样可能会导致误判。

1.2 与其它模块关系

本模块依赖于以下模块：

- 内存管理模块
- 系统功能模块

本模块需要调用以上模块相关接口进行内存分配释放等，具体见第 5 章“依赖接口说明”。

2 相关文档

《音视频库总体使用说明》

3 集成指南

3.1 层次结构

运动检测模块是一个独立的模块。

3.2 集成步骤

3.2.1 获得运动检测库相关文件

运动检测库包括库二进制文件、头文件和相关文档。

3.2.2 实现运动检测库依赖的系统API

请仔细阅读本文的第五部分，准备好运动检测库所依赖的系统 API。

3.2.3 集成链接测试

把运动检测库、依赖的系统 API 实现和目标系统进行链接。

3.2.4 联合调试

在目标系统上联合调试一般分成以下步骤，调试和使用运动检测库常见问题请参考本文的相关部分。调试运动检测，确定模块基本正常运作。

4 接口说明

4.1 数据结构/格式

```
typedef T_VOID (*MEDIALIB_CALLBACK_FUN_PRINTF)(T_pCSTR format, ...);
```

```
typedef T_pVOID (*MEDIALIB_CALLBACK_FUN_MALLOC)(T_U32 size);
```

```
typedef T_VOID (*MEDIALIB_CALLBACK_FUN_FREE)(T_pVOID mem);
```

```
typedef struct
```

```
{
    MEDIALIB_CALLBACK_FUN_PRINTF    m_FunPrintf;
    MEDIALIB_CALLBACK_FUN_MALLOC    m_FunMalloc;
    MEDIALIB_CALLBACK_FUN_FREE      m_FunFree;
}T_MOTION_DETECTOR_CB;
```

结构名	T_MOTION_DETECTOR_CB	
定义概述	提供的回调函数	
参数说明	m_FunPrintf	提供的打印函数
	m_FunMalloc	提供的分配内存函数
	m_FunFree	提供的释放内存函数
注意事项		

```
typedef struct _T_MOTION_DETECTOR_RATIO
```

```
{
    T_U16 m_uFullRatio; //白色像素占整幅图像的比例，该比例，控制判断是否运
    动，以千分比的形式，范围[1, 300]

    T_U16 m_uAreaRatio1; //白色像素占区域图像的比例
    T_U16 m_uAreaRatio2;
    T_U16 m_uAreaRatio3;
    T_U16 m_uAreaRatio4;
    T_U16 m_uAreaRatio5;
    T_U16 m_uAreaRatio6;
```

```
T_U16 m_uAreaRatio7;

T_U16 m_uAreaRatio8;

T_U16 m_uAreaRatio9;

}T_MOTION_DETECTOR_RATIO;
```

结构名	T_MOTION_DETECTOR_RATIO	
定义概述	检测灵敏度设置	
参数说明	m_uFullRatio	白色像素占整幅图像的比例，该比例用于控制判断是否运动，以千分比的形式，范围[1, 300]
	m_uAreaRatio1	第一个区域白色像素占整幅图像的比例
	m_uAreaRatio2	第二个区域白色像素占整幅图像的比例

	m_uAreaRatio9	第九个区域白色像素占整幅图像的比例
注意事项	各区域比例以千分比的形式，范围[1, 300]	

```
typedef struct _T_MOTION_DETECTOR_OPEN_PARA
{
    T_U16 m_uWidth;           //图像的宽度
    T_U16 m_uHeight;          //图像的高度
    T_U16 m_uThreshold;        //判断该像素为运动前景的阈值，取值范围[10, 40]
    T_MOTION_DETECTOR_CB m_CBFunc; //callback functions
}T_MOTION_DETECTOR_OPEN_PARA;
```

结构名	T_MOTION_DETECTOR_OPEN_PARA	
定义概述	提供输入的参数	
参数说明	m_uWidth	图像宽度
	m_uHeight	图像高度
	m_uThreshold	判断该像素为运动前景的阈值，取值范围[10, 40]

结构名	T_MOTION_DETECTOR_OPEN_PARA	
	m_CBFunc	提供的回调函数
注意事项		

4.2 接口函数列表

4.2.1 Motion_Detector_GetVersion

原 型	const T_CHR *Motion_Detector_GetVersion(T_VOID)	
功能概述	获取运动检测库版本号	
参数说明		
返回值说明	返回版本号信息	
返回值说明	版本号字符串	
注意事项	无	
调用示例	Motion_Detector_GetVersion();	

4.2.2 Motion_Detector_Open

原 型	T_pVOID Motion_Detector_Open(T_MOTION_DETECTOR_OPEN_PARA *pDetector_open_para)	
功能概述	初始化检测结构体的相应变量，分配好相关的内存资源。	
参数说明	pDetector_open_para	见 T_MOTION_DETECTOR_OPEN_PARA 结构体定义
返回值说明	初始化后的结构体变量指针。	
返回值说明	非 AK_NULL	初始化结构体变量和分配资源成功。
	AK_NULL	初始化结构体变量或分配资源失败。
注意事项		
调用示例	见典型调用范例	

4.2.3 Motion_Detector_Close

原 型	T_BOOL Motion_Detector_Close(T_pVOID pDetector)	
功能概述	释放解码驱动模块占用的所有资源，对解码器所有数据变量设为 0.	
参数说明	pDetector	Motion_Detector_Open返回的指针。
返回值说明		
返回值说明		
注意事项	执行本函数后不能再进行任何解码的操作； 注意Motion_Detector_Close和Motion_Detector_Open要配对使用	
调用示例	见典型调用范例	

4.2.4 Motion_Detector_Handle

原 型	T_BOOL Motion_Detector_Handle(T_pVOID pDetector, T_U8 *pData1, T_U8 *pData2)	
功能概述	检测是否有运动物体出现	
参数说明	pDetector	Motion_Detector_Open返回的指针。
	pData1	图像 1 的首地址
	pData2	图像 2 的首地址
返回值说明	返回图像 2 相对于图像 1 有没有相对运动。	
返回值说明	AK_TRUE	有运动物体出现
	AK_FALSE	无运动物体出现
注意事项	采集时摄像头必须保持拍摄同一场景，如果摄像头本身有运动，会导致误判	
调用示例	见典型调用范例	

4.2.5 Motion_Detector_SetRatio

原 型	T_BOOL Motion_Detector_SetRatio(T_pVOID pDetector, T_MOTION_DETECTOR_RATIO *pRatio)
功能概述	设置检测的 ratio 参数组

原 型	T_BOOL Motion_Detector_SetRatio(T_pVOID pDetector, T_MOTION_DETECTOR_RATIO *pRatio)	
参数说明	pDetector	Motion_Detector_Open返回的指针。
	pRatio	见 T_MOTION_DETECTOR_RATIO
返回值说明	设置是否成功。	
返回值说明	AK_FALSE	设置失败
	AK_TRUE	设置成功
注意事项	Motion_Detector_Open 后，未调用 Motion_Detector_SetRatio 进行设置时，各比例值缺省值为千分之 125	
调用示例	见典型调用范例	

4.2.6 Motion_Detector_SetDimension

原 型	T_BOOL Motion_Detector_SetDimension(T_pVOID pDetector, T_U16 uWidth, T_U16 uHeight)	
功能概述	设置检测的图像宽高	
参数说明	pDetector	Motion_Detector_Open返回的指针。
	pRatio	见 T_MOTION_DETECTOR_RATIO
返回值说明	设置是否成功。	
返回值说明	AK_FALSE	设置失败
	AK_TRUE	设置成功
注意事项	如果中途改变被检测的图像尺寸，通过该函数设置新的图像尺寸	
调用示例	Motion_Detector_SetDimension(pDetector, newWidth, newHeight);	

4.2.7 Motion_Detector_GetAvgLum

原 型	T_BOOL Motion_Detector_GetAvgLum(T_pVOID pDetector, T_U32 *pAvgLum1, T_U32 *pAvgLum2)	
功能概述	获取最近一次 Motion_Detector_Handle 中输入图像的亮度平均值	
参数说明	pDetector	Motion_Detector_Open返回的指针。
	pAvgLum1	Motion_Detector_Handle中图像 1 的亮度平均值

原 型	T_BOOL Motion_Detector_GetAvgLum(T_pVOID pDetector, T_U32 *pAvgLum1, T_U32 *pAvgLum2)	
	pAvgLum2	Motion_Detector_Handle 中图像 2 的亮度平均值
返回值说明	获取是否成功。	
返回值说明	AK_FALSE	获取失败
	AK_TRUE	获取成功
注意事项	如果Motion_Detector_Open之后从未调用过Motion_Detector_Handle, 两幅图像的亮度平均值输出为 0, 调用Motion_Detector_Handle后, 返回最近一次调用Motion_Detector_Handle时输入的图像的亮度平均值。 可以用于上层根据亮度平均值进行红外灯控制。	
调用示例	T_U32 AvgLum1, AvgLum2; Motion_Detector_Handle(pDetector, pData1, pData2); Motion_Detector_GetAvgLum(pDetector, &AvgLum1, &pAvgLum2);	

4.3 典型调用范例

T_VOID main(int argc, char* argv[])

{

T_MOTION_DETECTOR_OPEN_PARA detector_open_para;

T_VOID *hMdetector;

T_MOTION_DETECTOR_RATIO ratio;

T_U8 *pData1;

T_U8 *pData2;

T_U32 tickcount;

T_BOOLbMotionIsDetected = 0;

T_U32 timeDiv = 0;

init(); // initial file system, memory, camera, lcd and etc.

memset(&detector_open_para, sizeof(T_MOTION_DETECTOR_OPEN_PARA));

memset(&ratio, sizeof(T_MOTION_DETECTOR_RATIO));

```

detector_open_para.m_uWidth = 352;
detector_open_para.m_uHeight = 288;
detector_open_para.m_uThreshold = 35;

ratio.m_uFullRatio = 125;
ratio.m_uAreaRatio1 = 250;
ratio.m_uAreaRatio2 = 250;
ratio.m_uAreaRatio3 = 250;
ratio.m_uAreaRatio4 = 250;
ratio.m_uAreaRatio5 = 250;
ratio.m_uAreaRatio6 = 250;
ratio.m_uAreaRatio7 = 250;
ratio.m_uAreaRatio8 = 250;
ratio.m_uAreaRatio9 = 250;

hMdetector = Motion_Detector_Open(&detector_open_para);
if (AK_NULL == hMdetector)
{
    Motion_Detector_Close(hMdetector);
    return;
}

Motion_Detector_SetRatio(hMdetector, &ratio);

while (!bMotionIsDetected)
{
    pData1 = get_y_data();
    tickcount = get_system_time_ms();

    while(1)
    {
        if (get_system_time_ms() > (tickcount + 500))
    
```

```

        {
            break;
        }
    }

    pData2 = get_y_data();

    bMotionIsDetected = Motion_Detector_Handle(hMdetector, pData1, pData2);
    if (bMotionIsDetected)
    {
        start_record();//motion is detected, do something
        break;
    }
}
Motion_Detector_Close(hMdetector);
return;
}

```

5 依赖接口说明

本模块所依赖的外部接口主要有内存管理等一些运动检测库需要的功能接口，该类函数由目标平台实现。如申请内存，释放内存，清空内存内容，打印信息等等。

请参见《音视频库总体使用说明》。

6 常见问题

1) 无法检测到运动物体

可能是阈值（m_uThreshold）太大或比例（T_MOTION_DETECTOR_RATIO 结构体中各值）设置太大。

2) 没有运动物体检测出有

可能是阈值（m_uThreshold）太小或比例（T_MOTION_DETECTOR_RATIO 结构体中各值）设置太小。