

中控指静脉仪 SDK 开发指南

版本：2.0

日期：2017 年 6 月

中控指静脉仪 SDK 开发指南

Copyright ©ZKTeco Inc.2017 All rights reserved.

Release history

日期	版本	备注
2016 年 3 月 20 日	1.0	发布版
2017 年 6 月 04 日	2.0	接口全面更新

目录

1 概述	5
2 隐私权说明.....	5
3 系统需求.....	5
4 安装部署.....	5
5 SDK 接口说明	5
5.1 类型定义.....	6
5.1.1 定义常量.....	6
5.2 接口说明.....	7
5.2.1 ZKFingerVein_Init	7
5.2.2 ZKFingerVein_Terminate	7
5.2.3 ZKFingerVein_GetDeviceCount	7
5.2.4 ZKFingerVein_OpenDevice	8
5.2.5 ZKFingerVein_CloseDevice	8
5.2.6 ZKFingerVein_SetParameter.....	9
5.2.7 ZKFingerVein_GetParameter	9
5.2.8 ZKFingerVein_CaptureFingerVeinImageAndTemplate	10
5.2.9 ZKFingerVein_CaptureFingerVeinImage.....	12
5.2.10 ZKFingerVein_Verify	12
5.2.11 ZKFingerVein_DBInit.....	13
5.2.12 ZKFingerVein_DBFree.....	14
5.2.13 ZKFingerVein_DBAdd.....	14
5.2.14 ZKFingerVein_DBel	15
5.2.15 ZKFingerVein_DBClear.....	16
5.2.16 ZKFingerVein_DBCount	16
5.2.17 ZKFingerVein_DBIdentify.....	17

5.2.18 ZKFingerVein_DBHybridIdentify.....	18
5.2.19 ZKFingerVein_MergeFP	19
5.2.20 ZKFingerVein_SetThreshold	20
5.2.21 ZKFingerVein_GetThreshold.....	20
5.2.22 ZKFingerVein_GetSDKVersion	21
6 附录	22
6.1 附录 1	22
6.2 附录 2	22
6.3 附录 3	22

1 概述

欢迎使用中控指静脉仪 SDK，在使用前请您先仔细阅读本手册，以便您能更快地掌握并使用中控指静脉仪 SDK。

2 隐私权说明

本公司将本软件程序的使用权授予您，但您必须向本公司作出如下保证：不在本协议规定的条款之外，使用、拷贝、修改、租赁或转让本系统获取其中的任一部分。

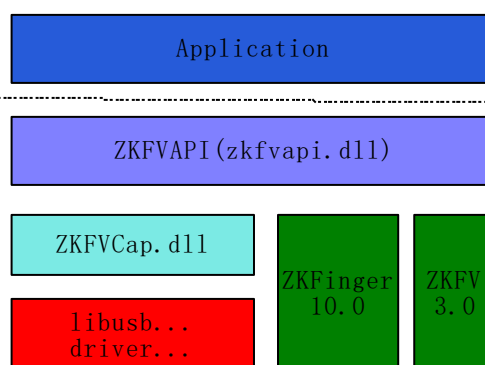
3 系统需求

- 1) 操作系统：Windows XP 及以上操作系统
- 2) 适用开发语言 C++、C#、VB、Delphi 等

4 安装部署

- 1) 安装：driver 目录为设备驱动；Bin\x86 为 32 位动态库文件；Bin\x64 为 64 位动态库文件。

5 SDK 接口说明



5.1 类型定义

5.1.1 定义常量

- 1) 模板最大长度

[定义] `#define MAX_TEMPLATE_SIZE 2048`

- 2) 登记模板最大指静脉数

[定义] `#define MAX_FVTEMPLATE_COUNT 6`

- 3) 指纹 1:1 阈值参数代码

[定义] `#define FP_THRESHOLD_CODE 1`

- 4) 指纹 1:N 阈值参数代码

[定义] `#define FP_MTHRESHOLD_CODE 2`

- 5) 指静脉 1:1 阈值参数代码

[定义] `#define FV_THRESHOLD_CODE 3`

- 6) 指静脉 1:N 阈值参数代码

[定义] `#define FV_MTHRESHOLD_CODE 4`

- 7) 生物识别类型:指纹

[定义] `#define BIO_TYPE_FP 0`

- 8) 生物识别类型:指静脉

[定义] `#define BIO_TYPE_FV 1`

- 9) 混合比对模式:普通模式-高通过率

[定义] `#define IDENTIFY_MODE_ANY 0`

- 10) 混合比对模式:防假模式

[定义] `#define IDENTIFY_MODE_FAKE 1`

- 11) 混合比对模式:安全模式

[定义] `#define IDENTIFY_MODE_BOTH 2`

5.2 接口说明

5.2.1 ZKFingerVein_Init

[函数]

```
int APICALL ZKFingerVein_Init();
```

[功能]

初始化资源，首先调用此接口

[参数]

无

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.2 ZKFingerVein_Terminate

[函数]

```
int APICALL ZKFingerVein_Terminate();
```

[功能]

释放资源

[参数]

无

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.3 ZKFingerVein_GetDeviceCount

[函数]

```
int APICALL ZKFingerVein_GetDeviceCount();
```

[功能]

获取设备数

[参数]

无

[返回值]

≥ 0 表示设备数

< 0 表示调用失败(见附录 3)

5.2.4 ZKFingerVein_OpenDevice

[函数]

```
void* APICALL ZKFingerVein_OpenDevice(int index);
```

[功能]

打开设备

[参数]

index

设备索引

[返回]

设备操作实例句柄

5.2.5 ZKFingerVein_CloseDevice

[函数]

```
int APICALL ZKFingerVein_CloseDevice(void* hDevice);
```

[函数]

关闭设备

[参数]

hDevice

设备操作实例句柄

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.6 ZKFingerVein_SetParameter

[函数]

```
int APICALL ZKFingerVein_SetParameter(void* hDevice, int nParamCode, unsigned char* paramValue, unsigned int cbParamValue);
```

[功能]

设置采集器参数

[参数]

hDevice

设备操作实例句柄

nParamCode

参数代码

paramValue

参数值

cbParamValue

参数数据长度

[返回值]

0 表示成功

其他表示失败(见附录 3)

[其他]

暂未启用，预留接口

5.2.7 ZKFingerVein_GetParameter

[函数]

```
int APICALL ZKFVM_GetParameter(void* hDevice, int nParamCode, unsigned char*
```

```
paramValue, unsigned int* cbParamValue);
```

[功能]

获取采集器参数

[参数]

hDevice

设备操作实例句柄

nParamCode

参数代码

paramValue[out]

返回参数值

cbParamValue[in/out]

in:paramValue 分配内存大小

out:返回参数值数据大小

[返回值]

0 表示成功

其他表示失败(见附录 3)

[其他]

暂未启用，预留接口

5.2.8 ZKFingerVein_CaptureFingerVeinImageAndTemplate

[函数]

```
int APICALL ZKFingerVein_CaptureFingerVeinImageAndTemplate(void* hDevice, unsigned char* fpImage, int* cbFPImage, unsigned char* fvImage, int* cbFVImage, unsigned char* fpTemplate, unsigned int* cbfpTemplate, unsigned char* fvTemplate, unsigned int* cbfvTemplate);
```

[功能]

采集图像和模板

[参数]

hDevice

设备操作实例句柄

fpImage[out]

返回指纹图像

cbFPImage[in/out]

in:fpImage 预分配内存大小(\geq fpWidth*fpHeight Bytes)

out:实际返回图像数据大小 (0 表示指纹采集失败)

fvImage[out]

返回指静脉图像

cbFVImage [in/out]

in:fvImage 预分配内存大小(\geq fvWidth*fvHeight Bytes)

out:实际返回图像数据大小

fpTemplate [out]

返回指纹模板

cbfpTemplate[in/out]

in:预分配 fpTemplate 内存大小, 建议值 MAX_TEMPLATE_SIZE (2048)

out:实际返回指纹模板数据大小

fvTemplate [out]

返回指静脉模板

cbfvTemplate [in/out]

in:预分配 fvTemplate 内存大小, 建议值 MAX_TEMPLATE_SIZE (2048)

out:实际返回指静脉模板数据大小

[返回值]

0 表示成功

其他表示失败(见附录 3)

[其他]

5.2.9 ZKFingerVein_CaptureFingerVeinImage

[函数]

```
int APICALL ZKFingerVein_CaptureFingerVeinImage(void* hDevice, unsigned char* fpImage, int* cbFPImage, unsigned char* fvImage, int* cbFVImage);
```

[功能]

采集图像

[参数]

hDevice

设备操作实例句柄

fpImage[out]

返回指纹图像

cbFPImage[in/out]

in:fpImage 预分配内存大小(\geq fpWidth*fpHeight Bytes)

out:实际返回图像数据大小 (0 表示指纹采集失败)

fvImage[out]

返回指静脉图像

cbFVImage[in/out]

in:fvImage 预分配内存大小(\geq fvWidth*fvHeight Bytes)

out:实际返回图像数据大小

[返回值]

0 表示成功

其他表示失败(见附录 3)

[其他]

5.2.10 ZKFingerVein_Verify

[函数]

```
int APICALL ZKFingerVein_Verify(void* handle, int type, unsigned char* tmp1, unsigned
int cbTmp1, unsigned char* tmp2, unsigned int cbTmp2);
```

[功能]

比对两枚指纹是否匹配

[参数]

handle

设备操作实例句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

tmp1

比对模板 1

cbTmp1

模板 1 数据长度

tmp2

比对模板 2

cbTmp2

模板 2 数据长度

[返回值]

≥ 0 比对分数

< 0 错误 (见附录 3)

5.2.11 ZKFingerVein_DBInit

[函数]

```
void* APICALL ZKFingerVein_DBInit(void* handle);
```

[功能]

初始化并创建 1: N 缓冲区

[参数]

handle

设备操作实例指针

[返回值]

缓冲区句柄

5.2.12 ZKFingerVein_DBFree

[函数]

```
int APICALL ZKFingerVein_DBFree(void* dbHandle);
```

[功能]

释放算法缓冲区

[参数]

缓冲区句柄

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.13 ZKFingerVein_DBAdd

[函数]

```
int APICALL ZKFingerVein_DBAdd(void* dbHandle, int type, char* fvid, unsigned char**  
tmp, int count);
```

[功能]

添加登记模板到缓冲区

[参数]

hDBCACHE

缓冲区句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

fvid

指静脉 ID (<64Bytes 字符串)

tmp

登记模板

count

模板数

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.14 ZKFingerVein_DBDel

[函数]

```
int APICALL ZKFingerVein_DBDel(void* dbHandle, int type, char* fvid);
```

[功能]

从缓冲区删除指定指静脉 ID 的登记模板

[参数]

hDBCach

缓冲区句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

fvid

指静脉 ID

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.15 ZKFingerVein_DBClear

[函数]

```
int APICALL ZKFingerVein_DBClear(void* dbHandle, int type);
```

[功能]

清空缓冲区

[参数]

hDBCach

缓冲区句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.16 ZKFingerVein_DBCount

[函数]

```
int APICALL ZKFingerVein_DBCount(void* dbHandle, int type);
```

[功能]

获取缓冲区模板数

[参数]

hDBCach

缓冲区句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

[返回值]

>=0 表示模板数

其他表示失败(见附录 3)

5.2.17 ZKFingerVein_DBIdentify

[函数]

```
int APICALL ZKFingerVein_DBIdentify(void* hDBCach, unsigned char* tmp, unsigned int  
cbTmp, char* fvid, unsigned int* score);
```

[功能]

指纹/指静脉 1:N 识别

[参数]

hDBCach

缓冲区句柄

type

BIO_TYPE_FP(0): 指纹

BIO_TYPE_FV(0): 指静脉

tmp

指纹/指静脉模板

cbTmp

模板数据长度

fvid[out]

返回指静脉 ID

score[out]

返回比对分数

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.18 ZKFingerVein_DBHybridIdentify

[函数]

```
int APICALL ZKFVM_NormalHybridIdentify(void* hDBCach, int mode, unsigned char* fpTemplate, unsigned int cbfpTemplate, unsigned char* fvTemplate, unsigned int cbfvTemplate, char* fvid, unsigned int* score);
```

[功能]

普通融合 1:N 识别(指纹, 指静脉一个识别通过即通过)

[参数]

hDBCach

缓冲区句柄

mode

IDENTIFY_MODE_ANY(0): 普通混合模式, 高通过率

IDENTIFY_MODE_FAKE(1): 防假混合模式, 以静脉作防假

IDENTIFY_MODE_BOTH(2): 安全混合模式, 高准确率

fpTemplate

指纹模板

cbfpTemplate

指纹模板数据长度

fvTemplate

指静脉模板

cbfvTemplate

指静脉模板数据长度

fvid [out]

返回指静脉 ID

score [out]

返回比对分数

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.19 ZKFingerVein_MergeFP

[函数]

```
int APICALL ZKFingerVein_MergeFP(void* handle, unsigned char** tmp, int count,
unsigned char* regTmp, int* cbRegTmp);
```

[功能]

指纹模板合并(多个模板取最好)

[参数]

hDBCach

缓冲区句柄

tmp

预登记指纹模板

count

指纹模板数

regTmp[out]

返回最好的指纹模板

cbRegTmp[in/out]

in:regTmp 分配内存长度, 建议预分配 2048 字节

out:返回模板数据长度

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.20 ZKFingerVein_SetThreshold

[函数]

```
int APICALL ZKFingerVein_SetThreshold(void* hDBCach, int code, int value);
```

[功能]

设置比对阈值

[参数]

hDBCach

缓冲区句柄

code

阈值参数代码(附录 2)

value

取值(0~100)

[返回值]

0 表示成功

其他表示失败(见附录 3)

[其他]

指纹比对阈值参考

1:1 35

1:N 55

指静脉比对阈值参考

1:1 72

1:N 82

5.2.21 ZKFingerVein_GetThreshold

[函数]

```
int APICALL ZKFingerVein_GetThreshold(void* hDBCach, int code, int* value);
```

[功能]

获取比对阈值

[参数]

hDBCach

缓冲区句柄

code

阈值参数代码(附录 2)

value[out]

返回阈值

[返回值]

0 表示成功

其他表示失败(见附录 3)

5.2.22 ZKFingerVein_GetSDKVersion

[函数]

```
int APICALL ZKFingerVein_GetSDKVersion(char* version, int *length);
```

[功能]

获取 SDK 版本

[参数]

version[in/out]

in:Version 预分配内存(>=10)

out: 实际返回数据

length[in/out]

in:预分配内存大小(>=10)

out: 实际返回数据大小

[返回值]

0 表示成功

其他表示失败(见附录 3)

6 附录

6.1 附录 1

参数代码描述

参数代码	参数	类型	描述
1	图像宽	int	低 16 位指纹图像宽度；高 16 位指静脉图像宽度
2	图像高	int	低 16 位指纹图像高度；高 16 位指静脉图像高度
1103	序列号	char*	最长 64 字节字符串

6.2 附录 2

比对阈值参数说明

参数代码	描述
1	1:1 指纹比对阈值
2	1:N 指纹比对阈值
3	1:1 指静脉比对阈值
4	1:N 指静脉比对阈值

6.3 附录 3

错误返回值描述

0	操作成功
1	已经初始化
-1	初始化算法库失败
-2	初始化采集库失败
-3	未连接设备
-4	接口暂不支持
-5	无效参数
-6	打开设备失败
-7	无效句柄
-8	取像失败
-9	提取指纹模板失败
-10	终端操作
-11	内存不足
-12	当前正在采集(设备繁忙)
-13	添加指纹模板到内存失败
-14	删除指纹模板失败
-15	添加指静脉模板失败
-16	删除指静脉模板失败
-17	操作失败(其他错误)
-18	取消采集
-19	提取指静脉模板失败
-20	比对指纹失败(登记过程中按不同的手指，或者按压不合理导致差异太大)
-21	比对指静脉失败(登记过程中按不同的手指，或者按压不合理导致差异太大)
-22	合并登记指纹模板失败
-23	指纹/指静脉 ID 重复