



中华人民共和国公共安全行业标准

GA 786—2008

十指指纹图像数据复现 JAVA 接口

JAVA API to call tenprint image decompression

2008-07-24 发布

2008-07-24 实施



中华人民共和国公安部 发布

前 言

本标准的全部技术内容为强制性。

本标准由公安部刑事侦查局提出。

本标准由全国刑事技术标准化技术委员会刑事信息分技术委员会(SAC/TC 179/SC 2)归口。

本标准起草单位:公安部刑事侦查局、江苏省公安厅刑事侦查局、航天科工集团二院二〇七所、北京北大高科指纹技术有限公司、北京东方金指科技有限公司。

本标准主要起草人:王瑛玮、周新民、苏月明、谢晓丹、王刚、许春光。

本标准委托公安部刑事侦查局负责解释。

十指指纹图像数据复现 JAVA 接口

1 范围

本标准规定了十指指纹图像数据复现 JAVA 接口。
本标准适用于十指指纹图像数据复现存储程序请求。

2 规范性引用文件

下列文件中的条款通过本标准的引用而成为本标准的条款。凡是注日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本标准,然而,鼓励根据本标准达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本标准。

- GA 773—2008 指纹自动识别系统术语
- GA 776—2008 指纹自动识别系统产品编码规则

3 术语和定义

GA 773—2008 确立的术语和定义适用于本标准。

4 格式要求

使用 JDK1.5 版本编译,使用 jar 进行打包。包名称采用如下的格式:fpt_dc××.jar,其中××为系统单位代码,应符合 GA 776—2008。

5 接口说明

包名(package):com. ga_casic. fptutil

5.1 接口名称

iImageDecoder

5.2 详细接口定义

Interface com. ga_casic. fptutil. iImageDecoder

Description:此接口为公安部各数据提供商转换厂商支持图片格式的公用接口,要求各指纹自动识别系统实现此接口的功能类的类名为<厂商的域名>. fptutil. ImageDecoder。

Operations

public InputStream	FPT_Decoder(Blob picblob) 转换厂商的对象为 InputStream 类型
public InputStream	FPT_Decoder(Blob picblob,String picfmt) 转换厂商的对象为 InputStream 类型
public String	getFactoryFullName() 取得厂商的名称全称
public String	getFactoryShrtName() 取得厂商的简要中文名称(6 字以下)
public boolean	isValidPicFmt(String picfmt) 是否为本厂商支持的图片格式
public int	SupportPicMaxLength() 厂商支持输出图片的最大长度

Operation Detail

FPT_Decoder

```
public InputStream FPT_Decoder(Blob picblob,String picfmt)
```

转换厂商的对象为 InputStream 类型

Throws:

InvalidBlobException

PictureLengthTooMaxException

CanNotSupportPicFmtExcetipion

Parameters:

picblob BLOB 对象

picfmt 指定图片格式后缀

Return:

指定格式的图片输入流

throws:

InvalidBlobException 无效的 Blob 对象

PictureLengthTooMaxException 图片过大

CanNotSupportPicFmtExcetipion 不支持指定图片类型转换

FPT_Decoder

```
public InputStream FPT_Decoder(Blob picblob)
```

转换厂商的对象为 InputStream 类型

Throws:

InvalidBlobException

PictureLengthTooMaxException

Parameters:

picblob BLOB 对象

Return:

InputStream 图片输入流

throws:

InvalidBlobException 无效的 Blob 对象

PictureLengthTooMaxException 图片过大

getFactoryFullName

```
public String getFactoryFullName()
```

取得厂商的名称全称

Return:

厂商的中文名称

getFactoryShtName

```
public String getFactoryShtName()
```

取得厂商的简要中文名称(6 字以下)

Return:

厂商的简要中文名称

isValidPicFmt

public boolean isValidPicFmt(String picfmt)

是否为本厂商支持的图片格式

Parameters:

picfmt 图片格式后缀

Return:

true/false

SupportPicMaxLength

public int SupportPicMaxLength()

厂商支持输出图片的最大长度

Return:

最大图片长度

接口定义源代码如下:ImageDecoder.java

package com.ga_casic.fptutil;

/**

* <p>Description:此接口为公安部各数据提供商转换厂商支持图片格的公用接口。

* /

import java.sql.Blob;

public interface ImageDecoder

{

//~ Methods

/**

* 转换厂商的对象为 InputStream 类型

* @param picblob BLOB 对象

* @return InputStream 图片输入流

* @throws InvalidBlobException 无效的 Blob 对象

* @throws PictureLengthTooMaxException 图片过大

* /

public java.io.InputStream FPT_Decoder(Blob picblob)

throws InvalidBlobException,

PictureLengthTooMaxException;

/**

* 转换厂商的对象为 InputStream 类型

* @param picblob BLOB 对象

* @param picfmt 指定图片格式后缀

* @return 指定格式的图片输入流

* @throws InvalidBlobException 无效的 Blob 对象

* @throws PictureLengthTooMaxException 图片过大

* @throws CanNotSupportPicFmtExcetipion 不支持指定图片类型转换

* /

```

public java.io.InputStream FPT_Decoder(Blob picblob,String picfmt)
                                throws InvalidBlobException,
                                PictureLengthTooMaxException,
                                CanNotSupportPicFmtExcetipion;

/* *
 * 是否为本厂商支持的图片格式
 * @param picfmt 图片格式后缀
 * @return true/false
 * /
public boolean isValidPicFmt(String picfmt);

/* *
 * 厂商支持输出图片的最大长度
 * @return 最大图片长度
 * /
public int SupportPicMaxLength();

/* *
 * 取得厂商的名称全称
 * @return 厂商的中文名称
 * /
public String getFactoryFullName();

/* *
 * 取得厂商的简要中文名称(6 字以下)
 * @return 厂商的简要中文名称
 * /
public String getFactoryShtName();
}

```

源代码: InvalidBlobException.java

```

package com.ga_casic.fptutil;
import java.io.*;
public class InvalidBlobException extends Exception
{
    //~ Instance fields -----
    private Throwable nestedThrowable=null;
    //~ Constructors -----

    public InvalidBlobException()
    {
        super();
    }
}

```

```

}

public InvalidBlobException(String msg)
{
    super(msg);
}

public InvalidBlobException(Throwable nestedThrowable)
{
    this.nestedThrowable=nestedThrowable;
}

public InvalidBlobException(String msg, Throwable nestedThrowable)
{
    super(msg);
    this.nestedThrowable=nestedThrowable;
}

//~ Methods -----

public void printStackTrace()
{
    super.printStackTrace();
    if(nestedThrowable != null) {
        nestedThrowable.printStackTrace();
    }
}

public void printStackTrace(PrintStream ps)
{
    super.printStackTrace(ps);
    if(nestedThrowable != null) {
        nestedThrowable.printStackTrace(ps);
    }
}

public void printStackTrace(PrintWriter pw)
{
    super.printStackTrace(pw);
    if(nestedThrowable != null) {
        nestedThrowable.printStackTrace(pw);
    }
}

```

源代码:CanNotSupportPicFmtExcetipion.java

```
package com.ga_casic.fptutil;
```

```
import java.io.*;
```

```
public class CanNotSupportPicFmtExcetipion extends Exception {
```

```
    private Throwable nestedThrowable=null;
```

```
    public CanNotSupportPicFmtExcetipion() {
        super();
    }
```

```
    public CanNotSupportPicFmtExcetipion(String msg) {
        super(msg);
    }
```

```
    public CanNotSupportPicFmtExcetipion(Throwable nestedThrowable)
        this.nestedThrowable=nestedThrowable;
    }
```

```
    public CanNotSupportPicFmtExcetipion(String msg,Throwable nestedThrowable) {
        super(msg);
        this.nestedThrowable=nestedThrowable;
    }
```

```
    public void printStackTrace() {
        super.printStackTrace();
        if(nestedThrowable != null) {
            nestedThrowable.printStackTrace();
        }
    }
```

```
    public void printStackTrace(PrintStream ps) {
        super.printStackTrace(ps);
        if(nestedThrowable != null) {
            nestedThrowable.printStackTrace(ps);
        }
    }
```

```
    public void printStackTrace(PrintWriter pw) {
        super.printStackTrace(pw);
        if(nestedThrowable != null) {
```

```

        nestedThrowable.printStackTrace(pw);
    }
}

```

源代码:InvalidBlobException.java

```
package com.ga_casic.fptutil;
```

```
import java.io.*;
```

```
public class InvalidBlobException extends Exception
```

```
{
```

```
//~ Instance fields -----
```

```
private Throwable nestedThrowable=null;
```

```
//~ Constructors-----
```

```
public InvalidBlobException()
```

```
{
```

```
    super();
```

```
}
```

```
public InvalidBlobException(String msg)
```

```
{
```

```
    super(msg);
```

```
}
```

```
public InvalidBlobException(Throwable nestedThrowable)
```

```
{
```

```
    this.nestedThrowable=nestedThrowable;
```

```
}
```

```
public InvalidBlobException(String msg,Throwable nestedThrowable)
```

```
{
```

```
    super(msg);
```

```
    this.nestedThrowable=nestedThrowable;
```

```
}
```

```
//~ Methods-----
```

```
public void printStackTrace()
```

```
{
```

```
    super.printStackTrace();
```

```
    if(nestedThrowable != null) {
```

```
        nestedThrowable.printStackTrace();
```

```
}
```

```

    }

    public void printStackTrace(PrintStream ps)
    {
        super.printStackTrace(ps);
        if(nestedThrowable != null) {
            nestedThrowable.printStackTrace(ps);
        }
    }

    public void printStackTrace(PrintWriter pw)
    {
        super.printStackTrace(pw);
        if(nestedThrowable != null) {
            nestedThrowable.printStackTrace(pw);
        }
    }
}

```

源代码:PictureLengthTooMaxException.java

```
package com.ga_casic.fptutil;
```

```

import java.io.*;

public class PictureLengthTooMaxException extends Exception {
    private Throwable nestedThrowable=null;
    public PictureLengthTooMaxException() {
        super();
    }

    public PictureLengthTooMaxException(String msg) {
        super(msg);
    }

    public PictureLengthTooMaxException(Throwable nestedThrowable) {
        this.nestedThrowable=nestedThrowable;
    }

    public PictureLengthTooMaxException(String msg,Throwable nestedThrowable) {
        super(msg);
        this.nestedThrowable=nestedThrowable;
    }

    public void printStackTrace() {

```

```
super.printStackTrace();  
if(nestedThrowable != null) {  
    nestedThrowable.printStackTrace();  
}  
}  
  
public void printStackTrace(PrintStream ps) {  
    super.printStackTrace(ps);  
    if(nestedThrowable != null) {  
        nestedThrowable.printStackTrace(ps);  
    }  
}  
  
public void printStackTrace(PrintWriter pw) {  
    super.printStackTrace(pw);  
    if(nestedThrowable != null) {  
        nestedThrowable.printStackTrace(pw);  
    }  
}  
}
```

中华人民共和国公共安全
行 业 标 准
十指指纹图像数据复现 JAVA 接口
GA 786—2008

*

中国标准出版社出版发行
北京复兴门外三里河北街16号
邮政编码:100045

网址 www.spc.net.cn

电话:68523946 68517548

中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销

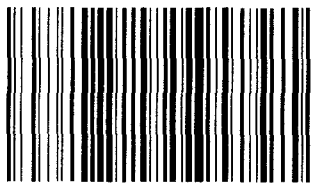
*

开本 880×1230 1/16 印张 1 字数 19 千字
2008 年 9 月第一版 2008 年 9 月第一次印刷

*

书号: 155066 · 2-19104 定价 16.00 元

如有印装差错 由本社发行中心调换
版权专有 侵权必究
举报电话:(010)68533533



GA 786—2008