

IntSig Information BankCard OCR

Android SDK Developer Guide

v1.3.1

Content

1. BankCard OCR SDK introduction	2
2. BankCard OCR Android SDK development guide	2
1. Invoking the build-in camera module from SDK to recognize	2
2. Invoking the SDK methods to recognize	2
3. SDK development prerequisites	2
4. SDK interfaces introduction	3
1. Invoking the SDK camera module recognition interface	4
2. Invoking the SDK methods recognition interface	6
5. SDK maintenance and troubleshooting	10

1. BankCard OCR SDK introduction

With the world leading OCR technology from IntSig Information, the Bankcard OCR SDK is able to detect the bankcard via camera preview mode and return the card number, cardholder name, expiry date and card type information, saving a lot of trouble when typing it manually. To provide better on-premises bankcard recognition experience, IntSig Information supports bankcard OCR SDKs for both Android and iOS platforms. Developers or third parties only need to integrate our bankcard OCR SDK with the application, and it will provide the local mobile bankcard OCR function.

2. BankCard OCR Android SDK development guide

We provide the bankcard OCR .so library files and the Java JNI interfaces, developers could do the fast and easy integration with the current app or project. The bankcard OCR SDK supports the following 2 invoking methods:

1. Invoking the build-in camera module from SDK to recognize

Via Intent calling to start the camera module from SDK and recognize the bankcard, return the recognition result to third party app, the entire bankcard capturing and recognizing process is encapsulated in the SDK camera module. If your app or project does not have any situations or requirements regarding the customization of camera module, you can use this integration solution.

2. Invoking the SDK methods to recognize

You can develop your own camera module with the specific requirement, and invoke the SDK in the camera module with the bankcard OCR SDK methods to do the recognition. For developers who need to customize the camera module in the app, this invoking method is recommended.

3. SDK development prerequisites

SDK development steps:

1. Before using or integrating with passport OCR SDK, please copy SDK path `libs\ccr_android_sdk.jar` and “armeabi-v7a” folder (the folders contain the local OCR so library files) to your current Android project path `libs\`, shown as figure 1

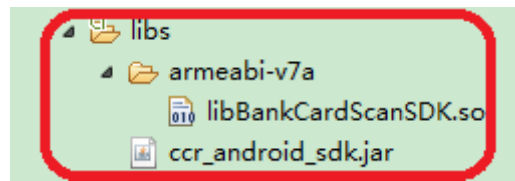


Figure 1

Note:

- 1) The `libBankCardScanSDK.so` file in “armeabi-v7a” folder supports ARM and ARM-V7 CPU. If the app only supports ARM architecture CPU, you can copy the “libBankCardScan.so” file to “armeabi” folder and only keep one “armeabi” folder in the `libs\` directory.
- 2) Please do not try to modify the file name “libBankCardScanSDK.so”, or you will get the SDK loading failure issue.

2. Declare the following permissions in the `AndroidManifest.xml` in your android project.

```
<uses-permission
android:name="android.permission.INTERNET"/>
<uses-permission
android:name="android.permission.CAMERA"/>
<uses-permission
android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

3. The BankCard OCR SDK supports Android 2.3 or higher version.
4. When doing the code proguard, please do not do proguard for any classes in the SDK. You can add this code in your Proguard file:

```
-keep class com.intsig.** {*,}
```

5. Now we can invoke the SDK from your own app.

4. SDK interfaces introduction

Please refer to the demo project “CCRSdk-Demo” in the SDK to learn how to integrate with the Bankcard OCR SDK, the demo project contains all the code examples. The two SDK invoking methods show as below:

1. Invoking the SDK camera module recognition interface

a) Start the Camera Capturing module:

```
//Call the SDK camera module method ISCardScanActivity via Intent to do the recognition
Intent intent = new Intent(this, ISCardScanActivity.class);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_COLOR_MATCH, 0xffff0000);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_COLOR_NORMAL, 0xff00ff00);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_TIPS, "Please put the bankcard in the frame");
intent.putExtra(ISCardScanActivity.EXTRA_KEY_APP_KEY, APP_KEY);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_GET_NUMBER_IMG, true);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_ORIENTATION,
ISCardScanActivity.ORIENTATION_HORIZONTAL);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_GET_TRIMMED_IMG, "/sdcard/trimmedcard.jpg");
intent.putExtra(ISCardScanActivity.EXTRA_KEY_GET_ORIGINAL_IMG, "/sdcard/originlcard.jpg");
intent.putExtra(ISCardScanActivity.EXTRA_KEY_SHOW_CLOSE,true);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_BOOL_FLAG_SECURE,true);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_TIPS_FONT_SIZE,23);
intent.putExtra(ISCardScanActivity.EXTRA_KEY_TIPS_FONT_COLOR,Color.WHITE);
startActivityForResult(intent, REQ_CODE_CAPTURE);
```

Intent Parameters Descriptions:

Parameter	Description
ISCardScanActivity. class	The camera module in SDK ISCardScanActivity class
ISCardScanActivity. EXTRA_KEY_COLOR_MATCH	Assign the color of the detecting rectangle when getting the bankcard in camera module in SDK, can be customized
ISCardScanActivity. EXTRA_KEY_COLOR_NORMAL	Assign the color of the detecting rectangle, can be customized
ISCardScanActivity. EXTRA_KEY_TIPS	Assign the UI tips string in camera module in SDK, can be customized
ISCardScanActivity. EXTRA_KEY_APP_KEY	The APP_KEY granted by IntSig Information.
ISCardScanActivity. EXTRA_KEY_GET_NUMBER_IMG	Assign the value for whether return the bankcard card number image, true for return and false for not return.
ISCardScanActivity. EXTRA_KEY_GET_TRIMMED_IMG	Define the bankcard trimmed image saving path
ISCardScanActivity. EXTRA_KEY_GET_ORIGINAL_IMG	Define the original bankcard image saving path.

ISCardScanActivity. EXTRA_KEY_ORIENTATION	Define the camera UI in landscape or portrait orientation. ISCardScanActivity.ORTIENTATION_VERTICAL means portrait orientation camera; ISCardScanActivity.ORTIENTATION_HORIZONTAL means landscape orientation camera
REQ_CODE_CAPTURE	The requestCode from onActivityResult
ISCardScanActivity. EXTRA_KEY_SHOW_CLOSE	Show or hide the close button
ISCardScanActivity. EXTRA_KEY_BOOL_FLAG_SECURE	Default is false:can screenshot Set True:can not screenshot
ISCardScanActivity. EXTRA_KEY_TIPS_FONT_SIZE	Set the title font size,must set float type
ISCardScanActivity. EXTRA_KEY_TIPS_FONT_COLOR	Set the title font color,must set float type

b) Get the bankcard recognition result:

Via the returned Intent data object from onActivityResult() to get the bankcard recognition result, shows as below:

```
ResultData result = (ResultData)
data.getSerializableExtra(ISCardScanActivity.EXTRA\_KEY\_RESULT)
Bitmap bmp = (Bitmap)
data.getParcelableExtra(ISCardScanActivity.EXTRA\_KEY\_GET\_NUMBER\_IMG);
ISCardScanActivity.EXTRA\_KEY\_RESULT
This is the Intent KEY value of the getting bankcard recognition result ResultData objects.
ISCardScanActivity.EXTRA\_KEY\_GET\_NUMBER\_IMG
This is the Intent KEY value of the getting bankcard card number image.
```

ResultData for getting bankcard recognition result, the description of return value:

ResultData Recognition Result	Description
getCardNumber()	Get the recognition result for bankcard card number
getCardHolderName()	Get the recognition result for bankcard cardholder

	name
getCardValidThru()	Get the recognition result for bankcard expiry date
getCardInsName()	Get the recognition result for bankcard issuer name
getBankCardType()	Get the recognition result for bankcard card type, please refer to the bank card type description
getCreditCardType()	Get the recognition result for bankcard credit card type, please refer to the credit card type description
getCode()	If the getCode() value is greater than 0, it shows the recognition succeeds, or the recognition fails.

Bankcard card type description:

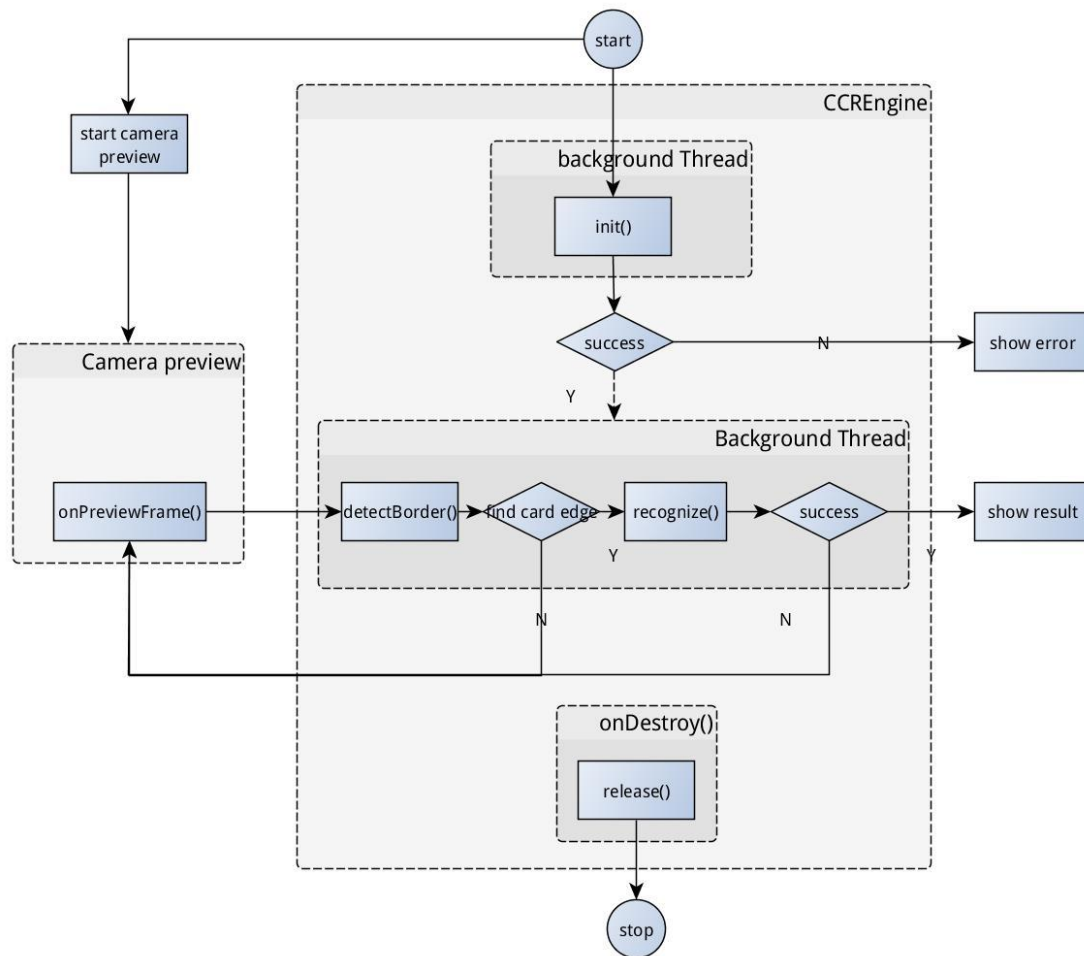
Bankcard type	Description
CCREngine.CCR_TYPE_CREDIT_CARD	credit card
CCREngine.CCR_TYPE_DEBIT_CARD	debit card
CCREngine.CCR_TYPE_QUASI_CREDIT_CARD	semi-credit card

Credit card type description:

Credit card type	Description
CCREngine.CCR_CARD_TYPE_VISA	VISA credit card
CCREngine.CCR_CARD_TYPE_MASTER	MasterCard credit card
CCREngine.CCR_CARD_TYPE_MAESTRO	Maestro credit card
CCREngine.CCR_CARD_TYPE_AMEX	American Express credit card
CCREngine.CCR_CARD_TYPE_DINERS	Diners Club credit card
CCREngine.CCR_CARD_TYPE_DISCOVER	Discover credit card
CCREngine.CCR_CARD_TYPE_JCB	JCB credit card
CCREngine.CCR_CARD_TYPE_CHINA_UNIONPAY	UnionPay credit card

2. Invoking the SDK methods recognition interface

The SDK methods invoking flowchart:



SDK interfaces invoking description:

1. int CCREngine.init(Context context, String appKey)

The method is used for “verify the license authentication and do the initialization for bankcard OCR engine”. It needs to be executed in thread, so please calling it when doing the initialization, before invoking the bankcard SDK recognition methods.

Description:

Parameter	Description
context	The application context from third party apps.
appKey	The APP_KEY is provided by IntSig Information, it binds the third party app’s Android package name and signature information

CCREngine.init() method returning int value description:

Int return value	Description
0	initialization succeeds

101	Package name is incorrect, it does not match the APP package name bond to the APP_KEY.
102	The inputted APP_KEY is incorrect
103	The APP-KEY has expired
104	The APP_KEY reaches the devices number limitation
201	The signature does not match the one bond to the APP_KEY
202	Other unknown issues, such as Initialization fails.
203	Server Error. When the first time SDK tries to connect to server to verify the license, it fails due to the server error.
204	Network Error. When the first time SDK tries to connect to server for authentication, it fails due to the internet connection
205	The APP package name or signature does not match the information bond to the APP_KEY

2. `int[]CCREngine.detectBorder(byte[] preview, int width, int height, int left, int top, int right, int bottom)`

This method is used to detect the edges of camera preview video frame image.

The Parameters descriptions show as below:

Parameter	Description
preview	The data of the camera preview video frame image
width	The width of the camera preview video frame image
height	The height of the camera preview video frame image
left	The top left X-axis coordinate of the preview rectangle
top	The top left Y-axis coordinate of the preview rectangle
right	The bottom right X-axis coordinate of the preview rectangle
bottom	The bottom right Y-axis coordinate of the preview rectangle

CCREngine.detectBorder() method return value description:

<code>int[]</code> return value	Description
<code>int[8]</code> value	The returned four coordinates of one bankcard image {x1,y1, x2,y2,x3,y3,x4,y4}, please refer

	to the demo project “CCRSdk-Demo” in SDK.
--	---

3. ResultData CCREngine.recognize(byte[] preview, int width, int height)

The bankcard image recognition method, needs to be run on background thread.

CCREngine.recognize() parameter description:

Parameter	Description
preview	Camera preview frame image data
width	The width of camera preview frame image
height	The height of camera preview frame image

If the return value of CCREngine.recognize() method is not null and the getCode() is greater than zero, it shows the recognition succeeds. If the return value is null or getCode() is smaller than zero, it shows the recognition fails.

ResultData description:

ResultData recognition result	Description
getCode()	If the value of getCode() is greater than zero, it shows recognition succeeds, or it fails.
getCardNumber()	Get the recognition result of bankcard card number
getCardHolderName()	Get the recognition result of bankcard cardholder name
getCardValidThru()	Get the recognition result of bankcard expiry date
getCardInsName()	Get the recognition result of bankcard issuer name
getBankCardType()	Get the recognition result for bankcard card type, please refer to the card type description
getCreditCardType()	Get the recognition result for bankcard credit card type, please refer to the credit card type description

4. CCREngine.release()

Release the memory and source used by bankcard OCR SDK or engine, you can call this method when the app exits. This method needs to be called with the method: CCREngine.init(), **only when the init() initialization succeeds, you can call the release()**.

Please refer to the project “CCRSdk-Demo” in our SDK to learn the detailed Bankcard OCR SDK sample code.

5. SDK maintenance and troubleshooting

1. SDK maintenance:

Please refer to the SDK developer guide document to learn how to integrate with our SDK. You can test your application or project scanning relevant function to ensure the SDK works fine.

2. SDK troubleshooting:

Test your application or our demo provided in the SDK, capture the clear and focus Bankcard image to do the scanning, check to see if the returned recognition results are correct. If the results are correct then indicating the SDK is running properly.

If the SDK could not return the results or return the incorrect results, please provide your testing image and the SDK log for further analyzing.