

INTSIG BankCard OCR iOS SDK Developer Guide

V2.0.0.20170518

Menu

1	Instruction Of Bank Card Scanning SDK.....	2
1.1	Function brief introduction.....	2
2	Bank card SDK Menu	2
3	Bank card preview recognition SDK Integrated description	3
3.1	Call SDK Camera Module API Recognition	3
3.2	Call SDK Custom Camera API Recognition	3
4	SDK Configuration Description	3
5	Bank Card Preview Recognition SDK Call Description	4
5.1	Call SDK Camera Module API Recognition	4
5.2	Call the SDK Camera Module API.....	9
5.3	The description of API returned error code for BankCard preview recognition	12
5.4	BankCard preview recognition picture requirements.....	13
6	Sample code of callback	13
6.1	Call the SDK Camera Module API to recognition	13
6.2	Call the SDK method API to recognize	14
7	Common problems.....	14

1 Instruction Of Bank Card Scanning SDK

1.1 Function brief introduction

With the world class OCR technology from IntSig Information, we can do the scanning on bank card and credit card, return the card number, card holder name, and expire date, issuer and card type information. Reduce the time for manually typing and let users save time. Developers could integrate our bank card scanning SDK with the app to have the bank card scanning and recognizing feature.

The bank card recognition Mini and Pro version of the feature list are as follows:

Feature List	Pro	Mini
1. BankCardType	Support	Support
2. CreditCardType	Support	Support
3. CardNumber	Support	Support
4. ExpiryDate	Support	No Support
5. HolderName	Support	No Support
6. BankName	Support	Support
7. Bank ID	Support	Support
8. TrimmingImage	Support	Support
9. CardRotate	Support	Support
10. BankNumberImage	Support	Support

2 Bank card SDK Menu

1. Demo	— INTSIG OCR SDK Demo
---------	-----------------------

2.Documentaions	— INTSIG OCR SDK Document
3.Release	— INTSIG OCR SDK Core Engine Recognition

3 Bank card preview recognition SDK Integrated description

Please refer to the demo project "ISBankCardCameraDemo" and "ISBankCardDemo" 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:

3.1 Call SDK Camera Module API Recognition

The camera module is boxed as `ISOpenSDKCameraViewController` in the SDK, it's used to Scan&Recognize a bankcard. The definition is located in `<ISOpenSDKFoundation.framework/ISOpenSDKCameraViewController.h>`, and below are the details. Please refer to the "ISBankCardCameraDemo" project.

3.2 Call SDK Custom Camera API Recognition

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. Please refer to the "ISBankCardDemo" project.

4 SDK Configuration Description

- Add `ISBankCard.framework` and `ISOpenSDKFoundation.framework` to your project. (You can find the two frameworks in the release folder)
- 在Add "`libstdc++.tbd`" and "`AVFoundation.framework`" and "`libc++.tbd`" to "BuildPhases->Link Binary With Libraries" ;
- Add "`-ObjC`" flag in BuildSettings->Other Linker Flags.
- BanCard SDK requires iOS 7.0 or later version.

5 Bank Card Preview Recognition SDK Call Description

Please refer to the demo project "ISBankCardCameraDemo" and "ISBankCardDemo" 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:

5.1 Call SDK Camera Module API Recognition

The camera module is boxed as `ISOpenSDKCameraViewController` in the SDK, it's used to Scan&Recognize a bankcard. The definition is located in `<ISOpenSDKFoundation.framework/ISOpenSDKCameraViewController.h>`, and below are the details

Camera module "ISOpenSDKCameraViewController" parameters:

Parameters	Description
<code>needShowBackButton</code>	Whether show the "Back" button on the Camera View
<code>shouldHighlightCorners</code>	High light the Corner when detect the border of the back card
<code>coverView</code>	Cover View, default to be null, which means cover view not show
<code>delegate</code>	Camera Delegate
<code>customInfo</code>	The note words on the Camera view, can be changed
<code>frameInterval</code>	Camera module recognition frame number interval (CMRFNI). System default minimum frame number per second/6; I.E when the minimum frame number per second are 30, then CMRFNI= 30/6=5
<code>continousRecognize</code>	Whether continue recognize next frame and return the result. Default is NO. It will call the callback of recognition if set to YES

1) Initial the Camera module "ISOpenSDKCameraViewController" .

```
-(ISOpenSDKCameraViewController *)cameraViewControllerWithAppkey:(NSString *)appKey
subAppkey:(NSString *)subAppKey
```

Parameters:

appKey	The key granted to you
subAppKey	The subAppKey is a reserved key for future usage, you can use nil for now.

Return Value:

ISOpenSDKCamera ViewController	The instance of the Camera module
-----------------------------------	-----------------------------------

2) After the Camera module is initiated, it will return 2 callbacks

a) Camera Permission error:

```
-(void)accessCameraDidFailed
```

If the iOS device is not granted the camera permission, then you can deal with the error case in this callback.

b) The result of Camera module initial returned in the following callback:

```
-(void)constructResourcesDidFailedWithStatusCode:(ISOpenSDKStatus)status
```

The camera module `ISOpenSDKCameraViewController` class member variable is described below:

Parameters	Description	
ISOpenSDKStatusUnauthorized = -1	Code	SDK without authorization
	Description:	
ISOpenSDKStatusUnreachable = 204	Solution:	Please contact our technical support to provide the correct AppKey
ISOpenSDKStatusSuccess = 0	SDK authorization success	
ISOpenSDKStatusUnreachable = 204	Code	SDK authorization status is unreachable. When use at first time SDK requires an internet connection to be authorized
	Description:	
ISOpenSDKStatusUnreachable = 204	Solution:	1. Check whether the current 2G, 3G, 4G, and WIFI can access the Internet properly;

		<p>2. please reboot your phone or switch on the fly mode after accessing the internet;</p> <p>3. If 1 and 2 operations cannot be verified, please contact our technical support for resolution</p>
ISOpenSDKStatusDeviceIDError = 100	Code Description:	Device ID error.
	Solution:	Please contact our technical support to provide the correct AppKey
ISOpenSDKStatusAppIDError = 101	Code Description:	Bundle ID error. It's inconsistent with one bound to your granted APP Key.
	Solution:	Please check whether TARGETS->General->Bundle Identifier is consistent with the AppKey we offer to you
ISOpenSDKStatusAppKeyError = 102	Code Description:	APP Key error. APP Key is case sensitive.
	Solution:	Please check whether the AppKey is supplied by our company
ISOpenSDKStatusAuthExpiredError = 103	Code Description:	APP Key is expired.
	Solution:	Please contact our technical support to provide the correct AppKey
ISOpenSDKStatusDeviceCappedError = 104	Code Description:	Granted device number is exceeded.
	Solution:	Please contact our technical support to provide the correct AppKey

ISOpenSDKStatusDetectC appError = 105	Code Description:	Granted recognition times are exceeded.
	Solution:	Please contact us for technical support to resolve
ISOpenSDKStatusSubAppK eyError = 106	Code Description:	Sub APP Key error.
	Solution:	Please contact us for technical support to resolve
ISOpenSDKStatusUnsuppo rtedAuthError = 202	Code Description:	Unknown issue.
	Solution:	Please contact us for technical support to resolve
ISOpenSDKStatusAuthori zeInfoError = 203	Code Description:	Server Error. When first online authorization, it' s failed due to server error.
	Solution:	Please contact us for technical support to resolve
ISOpenSDKStatusConstru ctResourceError = 205	Code Description:	Construct Resource Error.
	Solution:	Please contact us for technical support to resolve
ISOpenSDKStatusVersion Error = 206	Code Description:	SDK version error
	Solution:	Please contact us for technical support to resolve

After entering the camera shooting module, in the shooting recognition process, there will be the following three callbacks:

- a) The callback of detecting back card' s border:
- b) The callback of recognition result:
- c) The callback of the “Back” button:

a) The callback of detecting back card' s border:

```
-(void)cameraViewController:(UIViewController *)viewController
didFinishDetectCardWithResult:(int)result
borderPoints:(NSArray *)borderPoints
```

Parameters:

viewController	Camera View
result	Result>0, which indicates that the edge detection is successful
borderPoints	4 angle points of the bank card border, it includes 4 NSValue objects. You can get the corresponding "CGPoint" value by calling "CGPointValue". Clockwise, left top corner/right top corner/right bottom corner/left bottom corner.

b) The callback of recognition result:

```
-(void)cameraViewController:(UIViewController *)viewController
didFinishRecognizeCard:(NSDictionary *)resultInfo
cardSDKType:(ISOpenPreviewSDKType) sdkType
```

Parameters:

viewController	Camera View
resultInfo	<p>If the returned "cardInfo" is not null, then means recognize successfully, and retrieve the card information via the following Key. kBankCardType: Type of the bank card, you can have a reference view on the "ISBankCardType" part to get the details.</p> <p>kCreditCardType: Type of the credit card, such as VISA or MASTER; Please refer to "ISCreditCardType"</p> <p>kCardNumber: Card number.</p> <p>kBankName: issue bank name.</p> <p>kBankID: issue bank ID.</p> <p>kCardNumberRect: The rectangular position of card number on cropped card image.</p> <p>kCardNumberOriginalBorderPoint: The rectangular position of card number on the original card image without being cropped. kBankCardNumberImage: Screenshot of the card number.</p> <p>kOpenSDKCardResultTypeImage: Cropped card image.</p> <p>kOpenSDKCardResultTypeImage: cropped image of bank card. kOpenSDKCardResultTypeOriginImage: the original bank card image. kOpenSDKCardResultTypeCardRotate: the</p>

	original bank card image rotates degree, support 0, 90, 180, 270
sdkType	The type of SDK, currently only return "ISOpenPreviewSDKTypeBankCardReader"

c) The callback of the "Back" button:

```
-(void)cameraViewController:(UIViewController *)viewController
    didClickCancelButton:(id) sender
```

Parameters:

viewController	Camera View
sender	Close View Button

The camera module SDK detailed method of use can be found in the ISBankCardCameraDemo sample project.

5.2 Call the SDK Camera Module API

Directly call the recognition interface in the SDK, the main call steps are as follows:

- 1) Initialize the SDK resource;
- 2) BankCard video stream frame data recognition;
- 3) Release the SDK resources;

The specific interface is defined in <ISBankCardDemoSDK/
ISBankCardController.h >

Add the followings in the SDK or .pch file

```
#import < ISBankCard / ISBankCard.h >
#import <ISOpenSDKFoundation/ISOpenSDKFoundation.h>]
```

1) Initialize the SDK resource

```
-(void)constructResourcesWithAppKey:(NSString *)appKey
    subAppkey:(NSString *)subAppKey

finishHandler:( ConstructResourcesFinishHandler)handler
```

The parameters are described below;

appKey	The APP_KEY authorized by IntSig
--------	----------------------------------

subAppKey	SubAppKey is prepared for future expansion, you can keep it blank now
handler	Callback after initialization, the returned result is defined in ISOpenSDKStatus

When you use the SDK for the first time, it may take some time to callback the interface for network authorization. If the license has been successfully authorized, the API will immediately return and update the authorization status at the back end in *ISOpenSDKStatus.h*

2) BankCard video stream frame data recognition:

```
(ISOpenSDKStatus)detectCardWithOutputSampleBuffer:( CMSampleBufferRef)sampleBuffer
                                     cardRect:(CGRect)rect
                                     detectCardFinishHandler:(DetectCardFinishHandler)detectCardFinishHandler
recognizeCardFinishHandler:(RecognizeCardFinishHandler)recognizeFinishHandler
```

The parameters are described below 下:

sampleBuffer	Video stream buffer data
rect	The position of the passport in the video stream image. Note that you need to convert the coordinates of the rectangle from the camera preview layer to the actual video stream.
detectCardFinishHandler	Callback of the edge detection. If the return value is greater than 0, it indicates that the passport is successfully detected. BorderPointsArray represents the detected passport boundary point, contains four NSValue object, you can use the CGPointValue method to obtain the corresponding CGPoint value. The order is clockwise, top left, top right, bottom right, bottom left.
recognizeFinishHandler	BankCard recognition callback If the return value cardInfo is not empty, it indicates that the bankcard is successfully recognized. The following key gets the corresponding recognition result information; kBankCardType:Bank card type, please refer to the “ ISBankCardType ” definition; kCreditCardType: Credit card type; please refer to the “ ISCreditCardType ” definition

	<p>kCardNumber: Card number;</p> <p>kExpiryDate: Expiry date;</p> <p>kHolderName: Holder name;</p> <p>kBankName: Issue bank name;</p> <p>kBankID: Issue Bank ID</p> <p>kCardNumberRect: The rectangular position of card number on cropped card image;</p> <p>kCardNumberOriginalBorderPoint: The rectangular position of card number on the original card image without being cropped;</p> <p>kBankNumberImage: ; Bank card screenshot of the card number</p> <p>kOpenSDKCardResultTypeImage: bank card cropped card image;</p> <p>kOpenSDKCardResultTypeOriginImage: Original bank card image;</p> <p>kOpenSDKCardResultTypeCardRotate: The original bank card image rotates degree, support (0, 90, 180, 270)</p>
--	--

Make sure to call the *constructResourcesWithAppKey: subAppkey* first before calling the interface. The interface will only be recognized if the returned value is *ISOpenSDKStatusSuccess*. The recognized callback is *RecognizeCardFinishHandler* whose parameter *cardInfo* contains the recognition result

ISBankCardType Enumeration type has the following meaning

ISBankCardType parameters list	Description
ISBankCardTypeNone	No bank card type
ISBankCardTypeCreditCard	Credit card
ISBankCardTypeDebitCard	Debit card
ISBankCardTypeQuasiCreditCard =	Quasi credit card

ISCreditCardType Enumeration type has the following meaning:

ISCreditCardType parameters list	Description
ISCreditCardTypeNone	No bank card type
ISCreditCardTypeVISA	Visa

ISCreditCardTypeMASTER	MasterCard
ISCreditCardTypeMAESTRO	Maestro
ISCreditCardTypeAMEX	American Express
ISCreditCardTypeDINERS	Diners Club
ISCreditCardTypeDISCOVER	Discover
ISCreditCardTypeJCB	JCB
ISCreditCardTypeUNIONPAY	China UNIONPAY

3) In the SDK, it also supports obtaining the issuing bank name and issuing agency ID through the card number, which is shown below:

```

-(ISOpenSDKStatus)recognizeInstitutionWithCardPrefix:(NSString
*)cardPrefix
recognizeInstitutionFinishHandler:(RecognizeInstitutionFinishHandler)r
ecognizeFinishHandler

```

Parameters:

cardPrefix	Bank card number 11 to total number
recognizeFinish Handler	<p>The callback of the issuing bank name and the issuer ID.</p> <p>If the return value, the name of the name and the number of the name, are not null, the card will be recognized as successful.</p> <p>institutionName: Issue bank name</p> <p>institutionID: Issue bank ID</p>

4) Release the SDK resource interface:

```

- (void) destructResources

```

The detaild usage method of BankCard recognition can be found in the “ISBankCardDemo” sample project

5.3 The description of API returned error code for BankCard preview recognition

Please refer to the” [The ISOpenSDKStatus parameter is described below](#)”: section in this document.

5.4 BankCard preview recognition picture requirements

1. If you use the camera module provided by IntSig, there is no need to consider the preview image pixel size
2. If you use the SDK API to customize the camera, the definition of the preview pixel cannot be less than 1280 * 720 pixels, it is recommended to use the device above the iPhone4S.

6 Sample code of callback

6.1 Call the SDK Camera Module API to recognition

Initialize the camera module, if you use the Intsig default the camera module, please use this method to initialize, after initialization, please implement "5.1 call SDK camera module API recognition" return method function

```
ISOpenSDKCameraViewController *cameraVC = [[ISBankCardController
sharedISOpenSDKController] cameraViewControllerWithAppkey:appKey
subAppkey:subAppkey needCompleteness:YES];
// Whether to display the return button in the upper right corner
cameraVC.needShowBackButton = YES;
// Whether to display the the flash button in the upper left corner
cameraVC.needShowflashLightButton = YES;
// Whether to highlight it after edge detection is successful
cameraVC.shouldHightlightCorners = YES;
// Custom text information
cameraVC.customInfo = @"Please put the passport into the recognition
box";
// Set the delegate to receive the callback after recognition
cameraVC.delegate = self;
// Whether to continue to return the result after recognition is done,
the default is NO. When set to Yes, the recognition result callback
method will be called each time the recognition is successful
cameraVC.continuousRecognize = YES;
// Open the log
cameraVC.debugMode = YES;
```

6.2 Call the SDK method API to recognize

For details please refer to the method in the “ViewController.m” file included in ISBankCard Demo

To customize the camera preview module via the System API, note the followings:

1) Implement in “- (void) viewDidLoad:(BOOL)animated”

```
- (void)constructResourcesWithAppKey:(NSString *)appKey
                        subAppkey:(NSString *)subAppKey
                        finishHandler:(ConstructResourcesFinishHandler) handler;
```

So that the OCR recognition engine is turned on

In the delegate method of *AVCaptureVideoDataOutputSampleBufferDelegate*, the *sampleBuffer* of each frame, the preview area rect is passed to the following:

```
(ISOpenSDKStatus)detectCardWithOutputSampleBuffer:(CMSampleBufferRef) sampleBuffer
                        cardRect:(CGRect)rect
                        detectCardFinishHandler:(DetectCardFinishHandler)detectCardFinishHandler
```

```
recognizeCardFinishHandler:(RecognizeCardFinishHandler)recognizeFinishHandler;
to perform preview recognize
```

2) It's necessary to release the OCR recognition resource method *destructResources* in *-(void)dealloc*.

7 Common problems

Please refer to the attachment “ 上海合合信息_iOS_OCR 识别 FAQ 说明文档_v1.0.2.20170808 ”