AICamera (Eeye-UVC1) Android SDK API Reference

WDS<info@wd-s.com>

Version：V1. 1

Release Date：2020-5-11

Revision History

Date	Revison	Author	Description
2020-1-20	V1.0	WDS	Initial Release Version
2020-5-11	V1.1	WDS	Modify Top page Eeye Logo

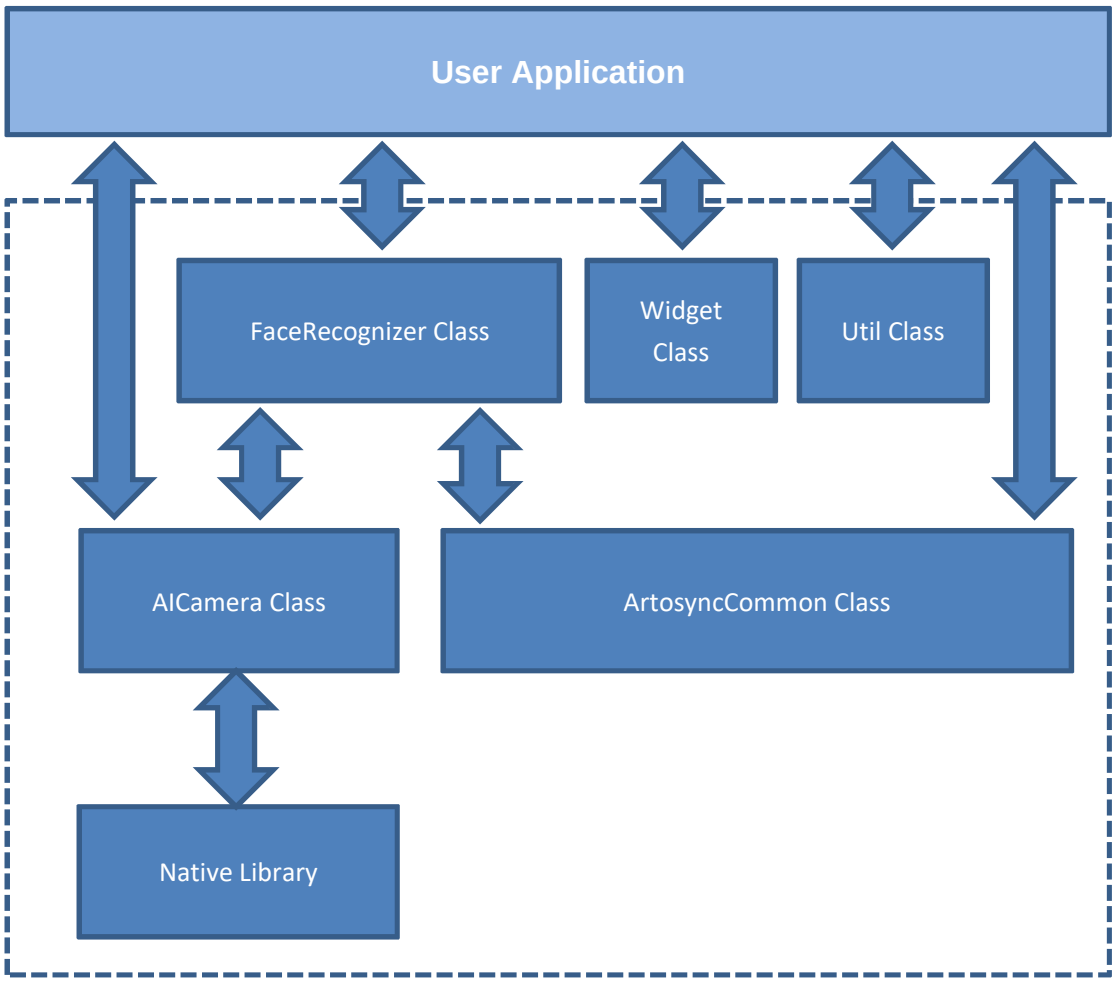
Contents

1. Overview

1.1 Introduction

AICamera Android SDK(hereafter referred to as SDK), encapsulates the underlying operation of parameter configuration and data read/write to the AICamera Device.By Using the APIs provided by the SDK, users can obtain frame and face recognition data from AICamera easily. Since most of users just need data obtained, and more concern about the business level of the application, the SDK can improve this situation and fast the development of the application.

1.2 Framework



SDK divides into 3 layers from bottom to top：

- Layer 1：Native Library

  The native code is encoded by C++, it encapsulates the communication details with AICamera, only be called by the AICamera class and can't be called by the

user application.

- Layer 2 : AICamera class and ArtosyncCommon class

  These two classes are the core of SDK, they are encoded with Java.AICamera class encapsulates the Native Library in advance, mainly used to get data from AICamera device; ArtosyncCommon class provides some common static methods, mainly used to parse data get from AICamera. If the users grasp programming, they can build some advance classes base on these two classes.

- Layer 2：Advance classes and auxiliary classes

  The Advance classes are high-level classes developed base on the classes of the layer 2, they are class sets aim to help users to implement their application quickly, they include FaceRecognizer class that help to do face recognition simply, some widgets commonly used in face recognition application and some auxiliary methods.

2. API Reference

2.1 AICamera Class

2.1.1 Instantiate an AICamera object

Users can use the constructor function of the AICamera class to instantiate an object :

```
AICamera camera = new AICamera(“/dev/video0”);
```

Call the open static method of AICamera can also return an AICamera object：

```
AICamera camera = AICamera.open("/dev/video0");
```

2.1.2 CaptureCallback Interface

CaptureCallback interface mainly used to notify the user of the AICamera class when the frame data and face recognition data arrived. Before calling the startCapture method, users should call setCaptureCallback and pass the instantiation of the CaptureCallback to the AICamera class, and overwrite onJpegFrame and onDataFrame methods of the CaptureCallback Interface, and deal frame and face recognition in these two callback functions.

2.1.3 setCaptureCallback method

Pass the CaptureCallback instantiation to AICamera class to notify the user of the

AICamera class when the frame data and face recognition data arrived.

2.1.4 startCapture method

Notify AICamera device to start capturing data. If any exception，it will throw IOException.

2.1.5 stopCapture method

Paring use with startCapture method, and notify the AICamera device to stop capturing data. If any exception，it will throw IOException.

2.1.6 static method getCameraDeviceNodes

Get all device nodes before AICamera object instantiation.

2.1.7 release method

Stop AICamera device, release the IO resources and memories occupied by the SDK.

2.2 ArtosyncCommon class

ArtosyncCommon class provides static methods used to parse the face recognition data pass from the AICamera Device.

2.2.1 checkPacket method

Check the magics and verify the checksum of the face recognition data.

2.2.2 getFacePacketHeader method

Get the header of face recognition data, rarely used by users.

2.2.3 getFaceSegmentList method

Return the face data as a List, users use the data return by this method and do their application in advance.

3. Use the SDK in your project

3.1 Obtain SDK

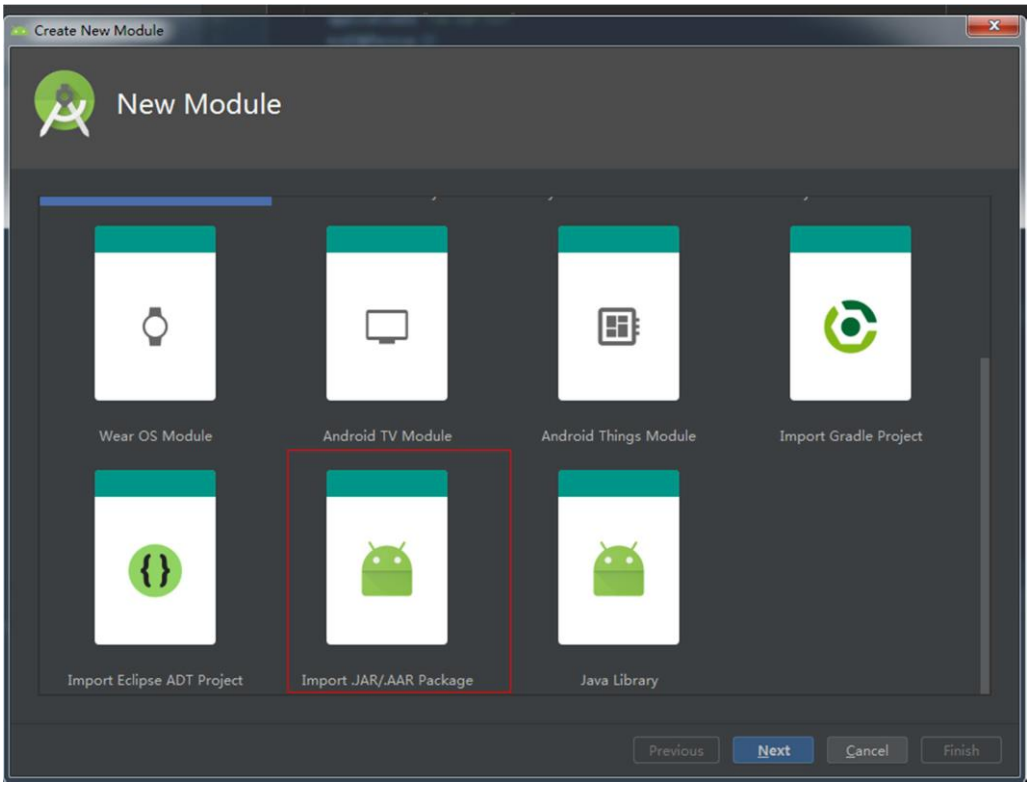
Please contact our customer service.

3.2 Import SDK Development kit

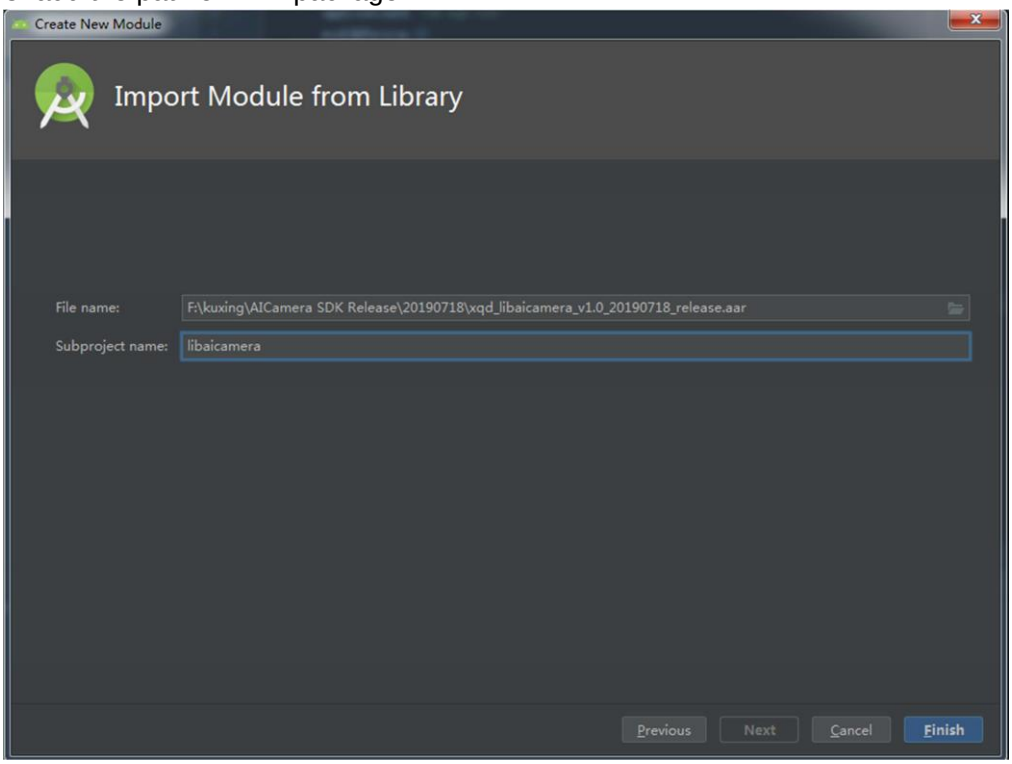
Our SDK is provided as an aar package, the following steps will show you how to import the SDK package in Android Studio:

1. Create a new project, and create a new Module in this project:

2. select “Import JAR/AAR Package”



3. add the path of AAR package



4. press F4, add module dependencies, and select the libaicamera added before: