# Android Studio中使用OpenCV的方法

在Android Studio中使用OpenCV摸索尝试了很久，网上的似乎都不靠谱，虽然完全照做，仍然各种错误，简直让人奔溃，几乎放弃，终于在YouTube上找到一个配制教程才得以成功，记录如下，以供以后参考。

## 软件环境

JDK：jdk\_8u101\_windows\_i586\_8.0.1010.13

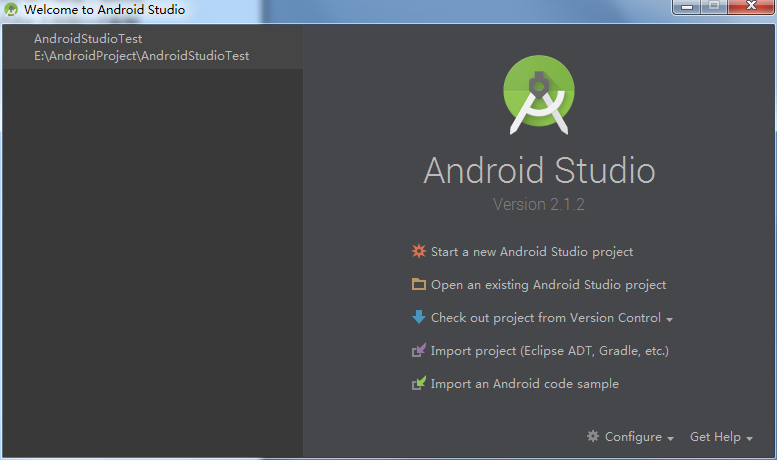
Android Studio: android\_studio\_2.1.0.0

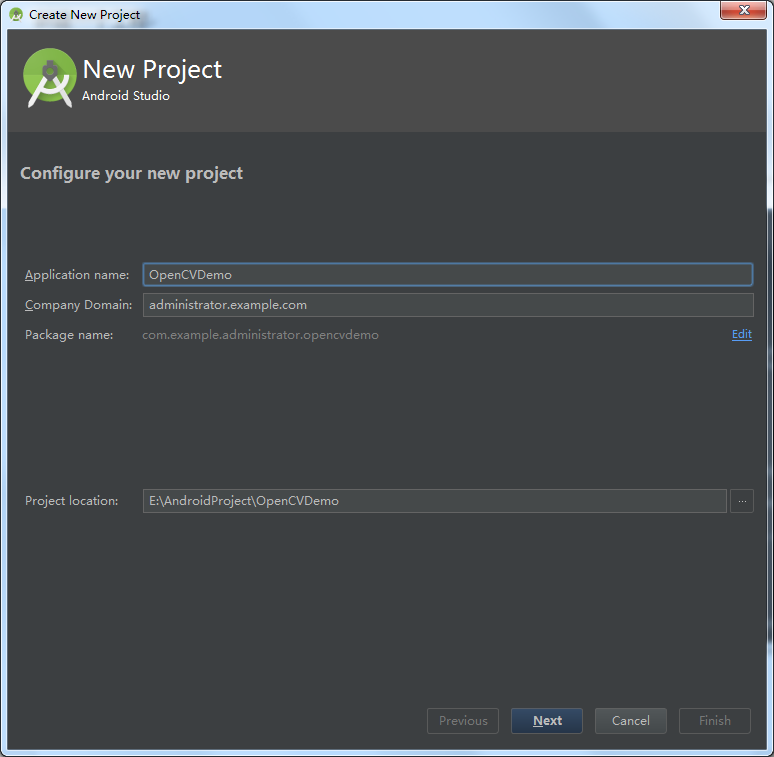
OpenCV：OpenCV-3.1.0-android-sdk

各种安装过程就不细说了，不过配置好NDK。

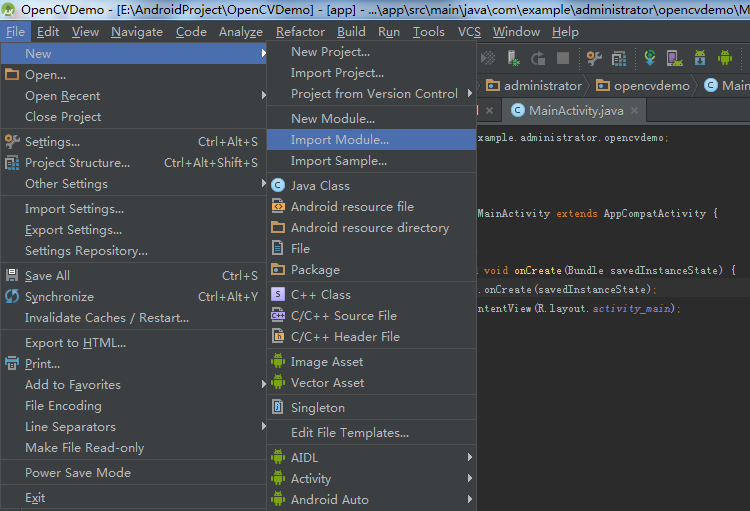
## 具体步骤

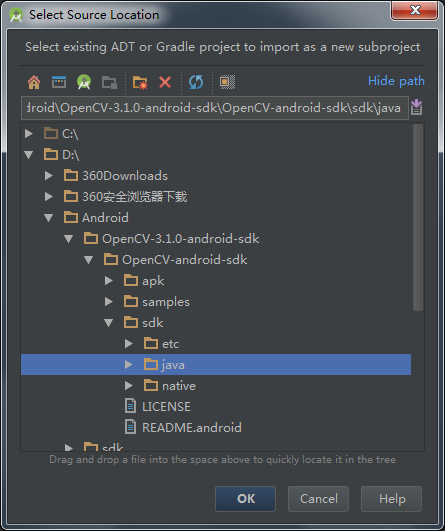
新建一个工程，名字叫OpenCVDemo



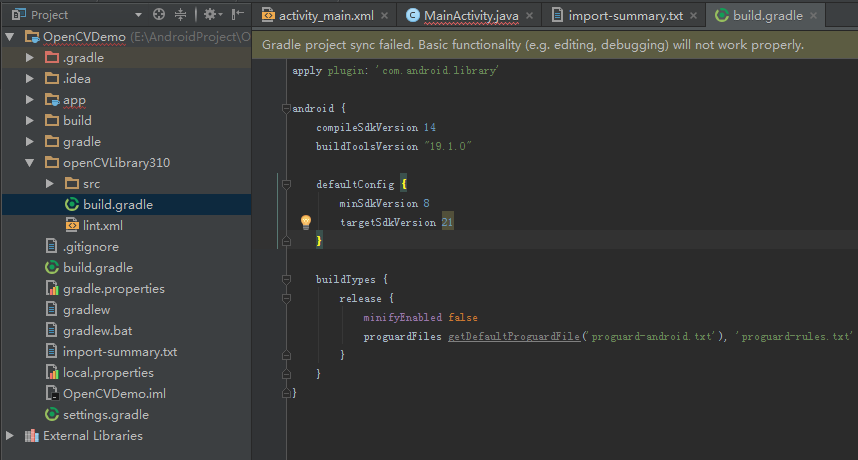


导入OpenCV模块：

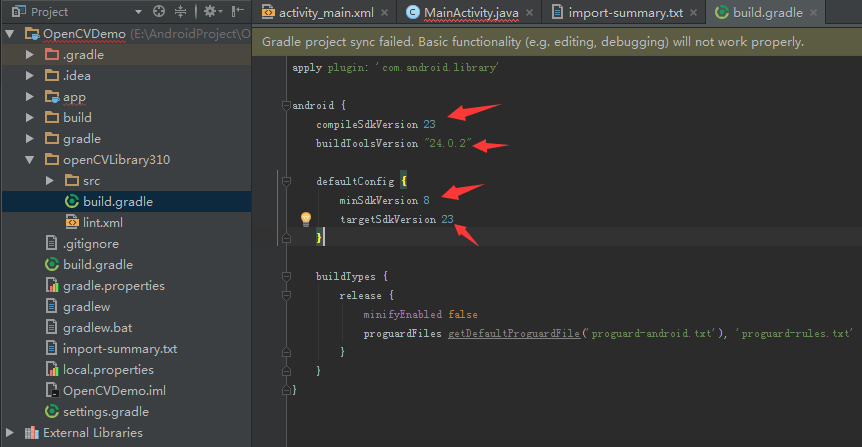




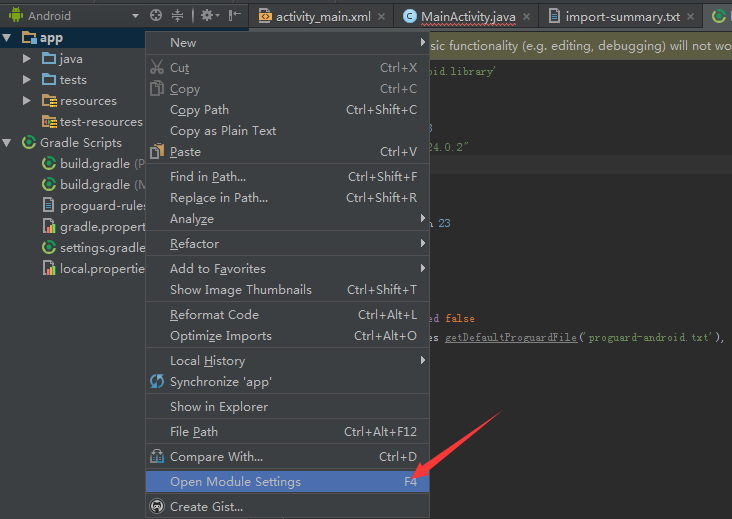
修改OpenCVLibrary310下的build.gradle文件，修改前：

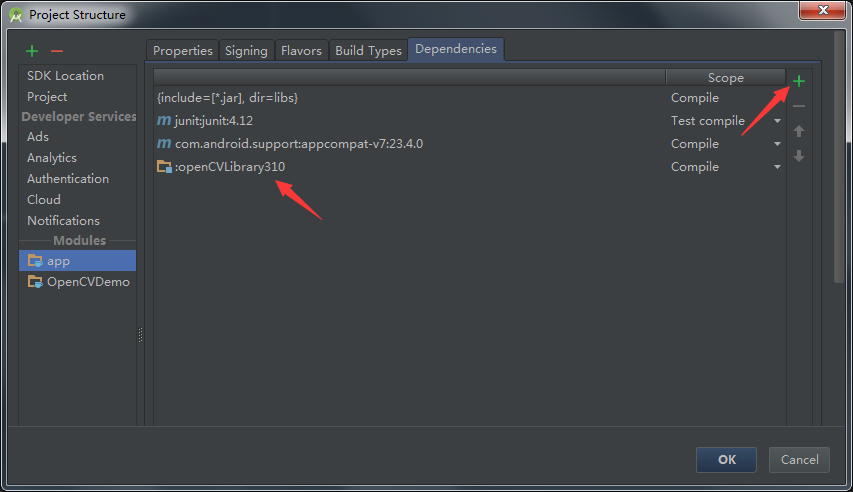


修改后：

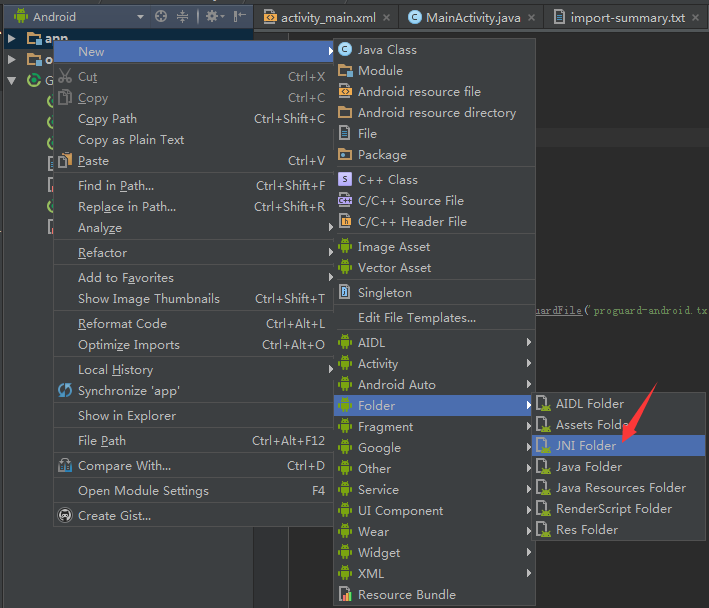


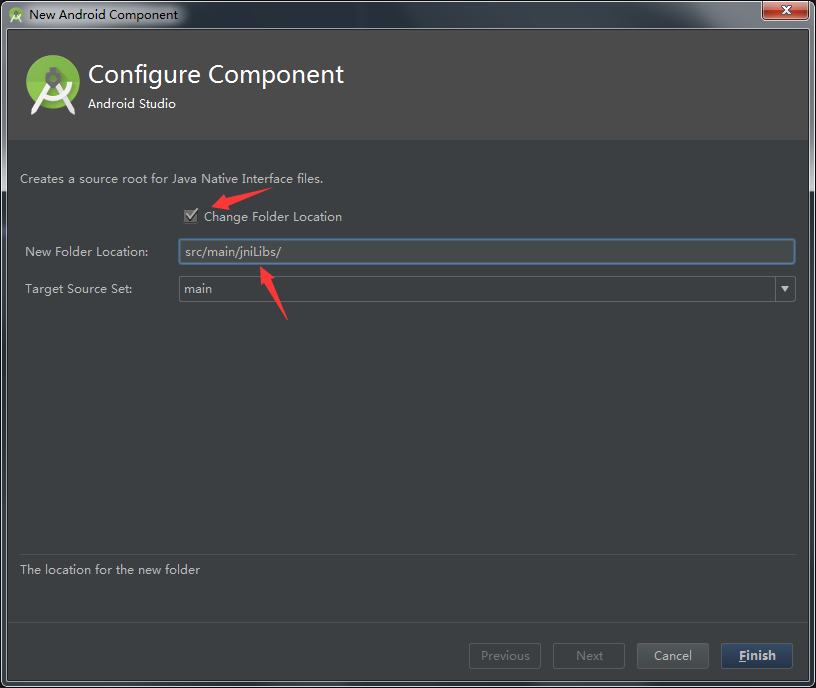
添加依赖：



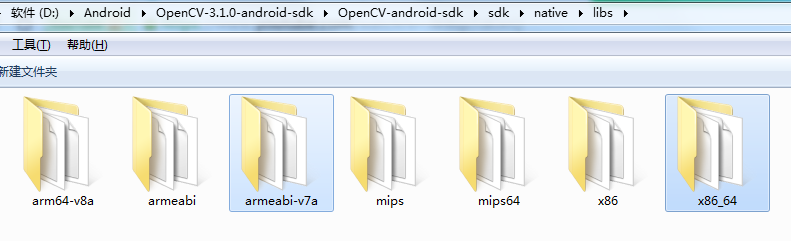


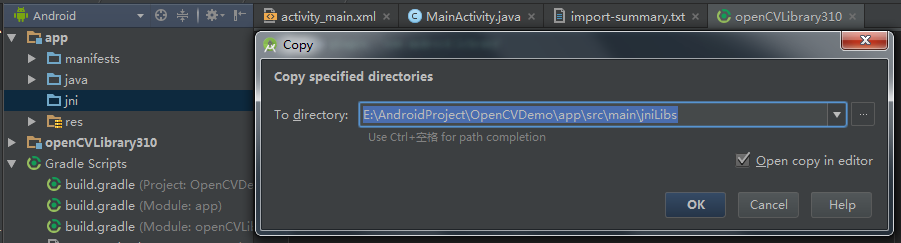
创建JniLibs目录：



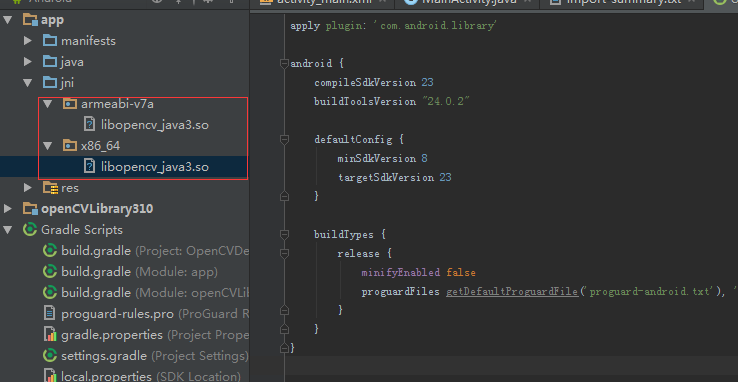


将OpenCV-android-sdk/native/libs目录下的两个文件夹armeabi-v7a和x86\_64拷贝到JniLibs目录下：



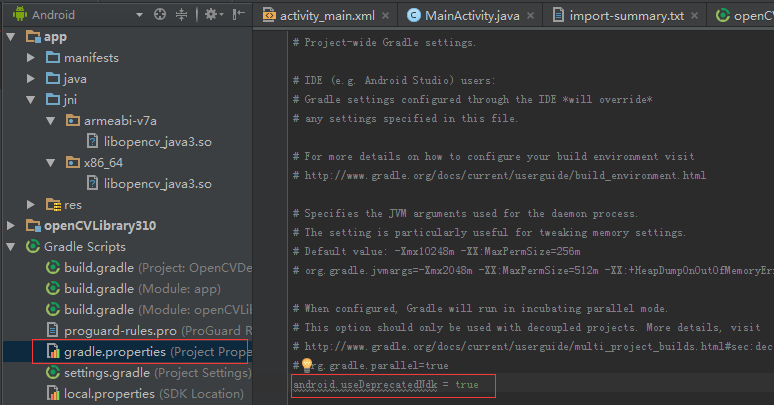


删除两个目录下的\*.a文件，只保留.so文件：



在gradle.properties中添加以下代码：

android.useDeprecatedNdk = true



在MainActivity.java中添加以下代码：

private static final String TAG ="MainActivity";

static {

if(!OpenCVLoader.initDebug())

{

Log.i(TAG,"OpenCV not loaded!");

}

else

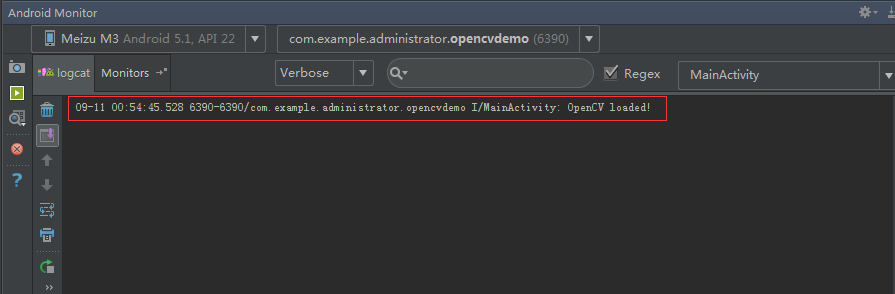
{

Log.i(TAG,"OpenCV loaded!");

}

}

编译运行：



说明OpenCV配置正确！成功了！！！！

## 例程

布局文件：

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 android:orientation="vertical"  
 tools:context="com.example.administrator.opencvdemo.MainActivity">  
  
  
 <ImageView  
 android:id="@+id/img"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:layout\_centerInParent="true"  
 android:background="@drawable/gril"/>  
  
 <Button  
 android:id="@+id/btn"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/img"  
 android:layout\_centerHorizontal="true"  
 android:text="To Gray"/>"  
</LinearLayout>

MainActivity.java文件：

package com.example.administrator.opencvdemo;  
  
import android.graphics.Bitmap;  
import android.graphics.BitmapFactory;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
  
import org.opencv.android.OpenCVLoader;  
import org.opencv.android.Utils;  
import org.opencv.core.Mat;  
import org.opencv.imgproc.Imgproc;  
  
public class MainActivity extends AppCompatActivity {  
 private static final String *TAG* ="MainActivity";  
 static {  
 if(!OpenCVLoader.*initDebug*())  
 {  
 Log.*i*(*TAG*,"OpenCV not loaded!");  
 }  
 else  
 {  
 Log.*i*(*TAG*,"OpenCV loaded!");  
 }  
 }  
 private Button btn;  
 private ImageView img;  
 private Bitmap srcBitmap;  
 private Bitmap grayBitmap;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 img = (ImageView)findViewById(R.id.*img*);  
 btn = (Button)findViewById(R.id.*btn*);  
 btn.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 procSrc2Gray();  
 img.setImageBitmap(grayBitmap);  
 }  
 });  
 }  
 public void procSrc2Gray(){  
 Mat rgbMat = new Mat();  
 Mat grayMat = new Mat();  
 srcBitmap = BitmapFactory.*decodeResource*(getResources(), R.drawable.*gril*);  
 grayBitmap = Bitmap.*createBitmap*(srcBitmap.getWidth(), srcBitmap.getHeight(), Bitmap.Config.*RGB\_565*);  
 Utils.*bitmapToMat*(srcBitmap, rgbMat);  
 Imgproc.*cvtColor*(rgbMat, grayMat, Imgproc.*COLOR\_RGB2GRAY*);  
 Utils.*matToBitmap*(grayMat, grayBitmap);  
 }  
}

