



# JohnHany的博客

编程 手绘 文字

[首页](#) [技术文章](#) [展示](#) [随笔](#) [关于我](#)



累积阅读

464083

分类目录

展示 (15)

CG (6)

手绘 (5)

摄影 (4)

技术文章 (33)

Android (14)

## 在Android Studio上进行 OpenCV 3.1开发

发布于 2016年1月27日 作者: John Hany

19,772次阅读

2016.07.08更新：增加Android Studio 2.1.2的配置方法

自从Android Studio发布以后，Google已不再推荐开发者使用Eclipse+ADT的方式进行Android的开发了。而且，随着开发脚步的前进，Android Studio提供了越来越多的Eclipse所没有的优秀特性。所以，忍痛放弃老工具，学习新工具才是明智之选。

Android Studio 2发布之后，增加了若干新功能。相比较1代版本，2.x最主要的变化在于增加了Instant Run功能，免去了每次

C/C++ (2)  
HTML5 (2)  
Java (1)  
图像处理 (19)  
应用数学 (4)  
模式识别 (4)  
计算机图形学 (5)  
随笔 (2)

## 最多阅读

---

Windows7+VS2010下OpenCV环境配置 - 74,408 次阅读  
Java+MySQL实现网络爬虫程序 - 50,889 次阅读  
Windows7+VS2012下64位OpenCV3.0+CUDA7.5的编译和部署 - 30,647 次阅读  
关于我 - 24,985 次阅读  
Android上使用OpenCV处理图像 - 23,848 次阅读  
Android Studio 2上利用NDK进行OpenCV 3.1开发 - 20,641 次阅读

修改代码之后都要重新编译安装的过程，不过需要minSdkVersion设置为15以上，而在21以上时性能最好；另一个变化就是Android模拟器的速度提升明显。所以如果你使用的是Android Studio 2，而且对模拟器的功能的丰富性要求不太高的话，可以略过Android模拟器|Genymotion一节，直接使用默认的AVD Manager建立模拟器。

注：本文涉及的某些网站的访问与工具的下载在国内是受限的。在没有解决网络连接问题的情况下进行环境配置的方法，本文并不涉及。

## 索引

—[环境配置](#)

—[SDK Manager](#)

—[Android Studio欢迎界面](#)

—[切换深色主题](#)

—[更改编辑器字体](#)

—[Android模拟器](#)

—[AVD Manager](#)

—[HAXM](#)

OpenCV实现基于傅里叶变换的旋转  
文本校正 - 19,939 次阅读

在Android Studio上进行OpenCV 3.1  
开发 - 19,772 次阅读

Tracking-Learning-Detection原理分  
析 - 16,937 次阅读

水面的简单渲染 – Gerstner波 -  
15,785 次阅读

## 最新评论

---

John Hany 在 Java+MySQL实现网络  
爬虫程序  
httpget.java文件里的responseBody就  
是 ...

251013371@qq.com 在 Java+MySQL  
实现网络爬虫程序  
如果能把查到的网页内容保存到本  
地，供以后在本地使用，要怎么办  
...

John Hany 在 在Android Studio上进  
行OpenCV 3.1开发  
你好，android版本的opencv也是由  
原本为PC设计的 ...

—[Genymotion](#)

—[为项目配置OpenCV](#)

—[建立Android项目](#)

—[导入OpenCV](#)

—[build.gradle](#)

—[将Android项目与OpenCV整合](#)

—[OpenCV Manager](#)

## 开发环境：

Windows 10 x64 专业版（Windows 7 x64 旗舰版也验证可  
行）

Android Studio 1.5.1（Gradle版本2.8）、Android Studio  
2.1.2（Gradle版本2.10）

JDK 1.8.0

Android 6.0（API 23）

OpenCV 3.1.0 Android SDK

冰淇淋 在 在Android Studio上进行OpenCV 3.1开发

博主，你好，最近在忙着用opencv做一个拍照解数独的app ...

John Hany 在 Ubuntu 16.04下为Android编译OpenCV 3.2.0 Manager

我当时编译时没有包含BRIEF算法，我估计按照文章里相似的方 ...

夜de1 在 Ubuntu 16.04下为Android编译OpenCV 3.2.0 Manager

楼主有解锁brief的OpenCV Android SDK吗 ...

John Hany 在 Android Studio 2.3利用CMAKE进行OpenCV 3.2的NDK开发

确实，很多问题主要是版本过老引起的，但也有一些明显的代码错误 ...

matrixth 在 Android Studio 2.3利用CMAKE进行OpenCV 3.2的NDK开发哦哦。。我刚刚用opencv2终于弄出来了--没想到博主大 ...

## 文章归档

---

可能会用到的工具：

Genymotion（一个第三方的Android模拟器）

## 环境配置：

1.首先要保证计算机上有Java环境。如果没有的话请参考其他文章（比如《[Windows下Android开发环境的配置](#)》的第一部分）进行Java开发环境的配置。

2.在<http://developer.android.com/sdk/index.html>下载最新稳定版Android Studio。从这个链接下载的Android Studio可能并不是最新版，我这里给出1.5.1版本的下载链接：<https://dl.google.com/dl/android/studio/ide-zips/1.5.1.0/android-studio-ide-141.2456560-windows.zip>。如果下载到的文件名称是以“android-studio-bundle”开头的，说明该安装包不只包含了Android Studio本身，还包括了最新版本的Android SDK以及相应的模拟器系统镜像；如果文件名称是以“android-studio-ide”开头的，则只包含Android Studio本身。

~~虽然Android Studio 2已有预览版，但是每一次更新都会对工作流程产生一些改变，所以目前还是推荐使用Android Studio 1.5.1。~~Android Studio 1.x与Android Studio 2.x对OpenCV Java API的配置方法基本相同，区别仅在于一些插件的版本和界面略有不同，下文介绍的步骤对两种版本都适用，差别之处会有明显的标记。值得一提的是，不同版本的Android Studio相互之间并不冲

2017年八月 (5)

2017年七月 (1)

2016年七月 (5)

2016年五月 (1)

2016年四月 (1)

2016年一月 (1)

2015年十月 (1)

2015年三月 (1)

2015年一月 (1)

2014年七月 (1)

2014年六月 (1)

2014年五月 (2)

2014年四月 (1)

2014年三月 (1)

2014年二月 (1)

2014年一月 (1)

2013年十二月 (1)

2013年十一月 (24)

突，你甚至可以在桌面上同时打开多个不同版本的Android Studio。

在安装过程中会询问Android Studio安装位置与Android SDK的安装位置，假设SDK的位置选择在了C:\dev-lib\android-sdk。

安装完成之后，第一次运行Android Studio时，软件会自动连接Google服务器下载最新的Android SDK。这一步可以跳过，稍后我们自己打开Android SDK Manager来安装所需的文件。

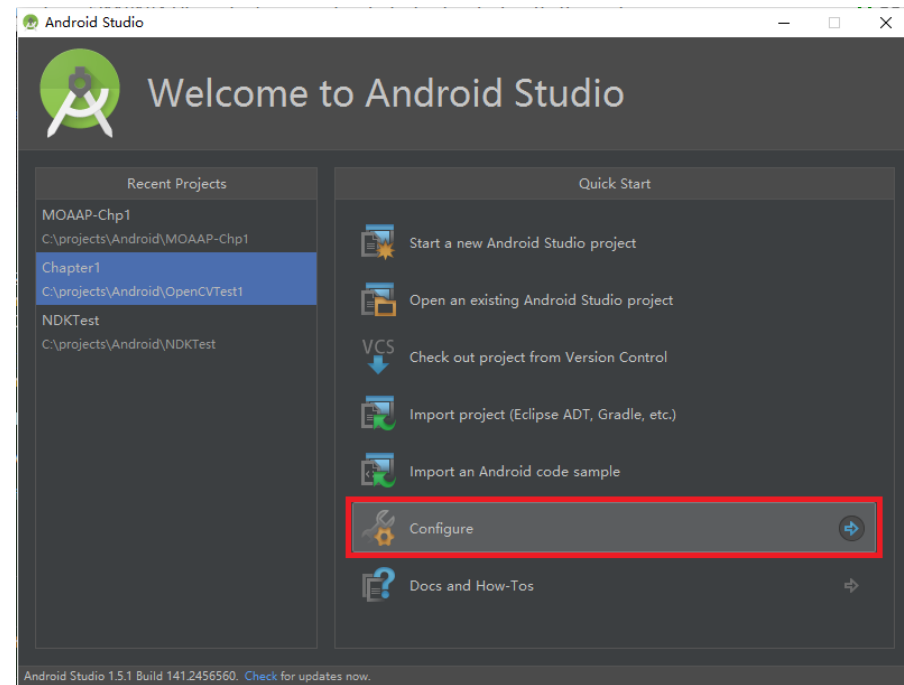
3.在<http://sourceforge.net/projects/opencvlibrary/files/opencv-android/3.1.0/OpenCV-3.1.0-android-sdk.zip/download>下载OpenCV 3.1.0 Android SDK，解压到某个不限制读写权限的目录下，比如C:\dev-lib\OpenCV-android-sdk。

4.打开Android Studio。第一次运行时会自动下载并安装最新的Android SDK文件，这里可以取消，直接打开Android Studio。此时会弹出一个欢迎界面。为了安装必要的SDK文件，点击Configure：

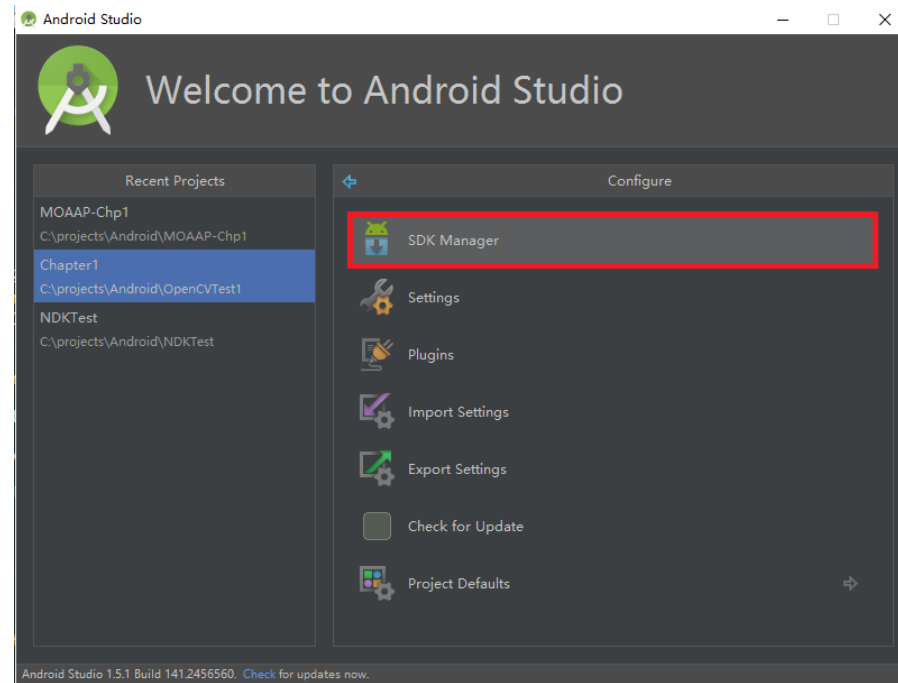
## 常用标签


---

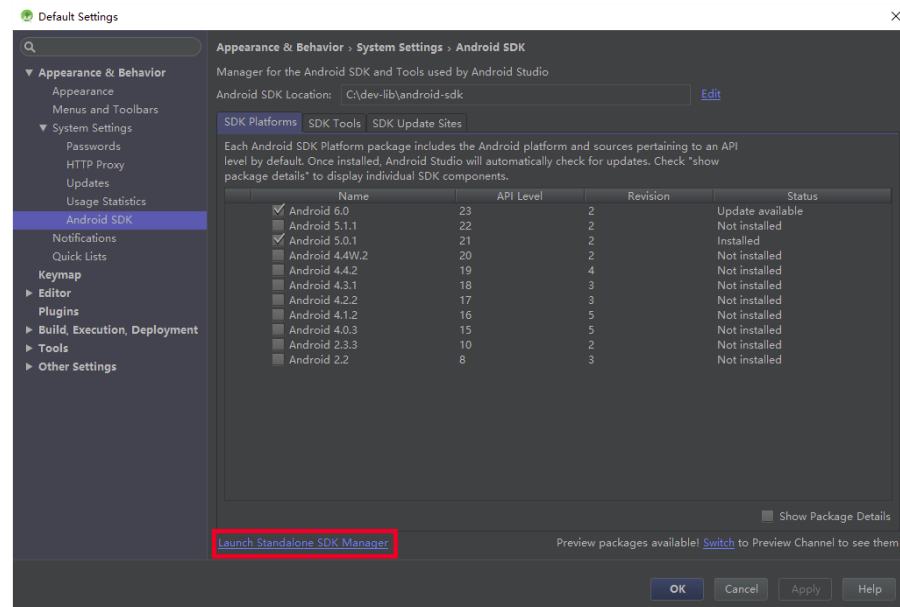
android c++ cloudforge cocos-2d css  
eclipse GLSL html5 Java javascript maya  
MySQL ndk **opencv** opengl php ps  
sift surf svn VS wordpress 临摹 交互 噪声 图  
壁纸 **建模 手绘 摄影** 文本处理 机器学习  
**习** 深度学习 渲染 滤波 爬虫 **环境配置**  
短篇 算法 素描 网络 视频追踪 计算几何 随机  
随笔



再选择SDK Manager：



在弹出的窗口中点击下方的Launch Standalone SDK Manager（如果以后想随时打开这个界面，只需要点击Android Studio界面上方的按钮）：

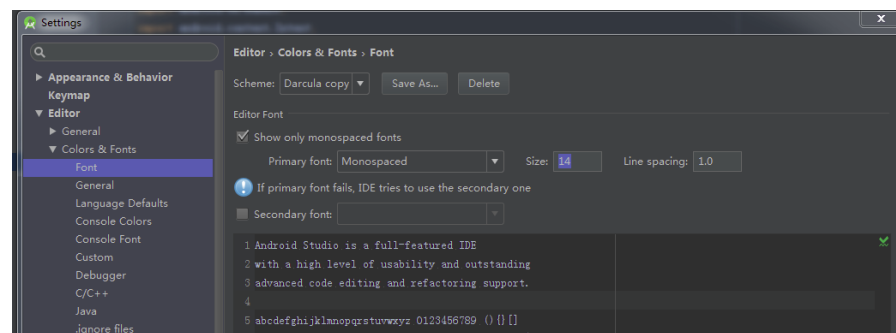


一旦建立项目之后，Android Studio每次都会默认打开最近使用的的项目。如果想每次打开Android Studio时都弹出这个欢迎界面，需要依次打开File->Settings...->Appearance & Behaviour->System Settings，取消勾选Reopen last project on startup。（参考：<http://stackoverflow.com/questions/21181847/how-to-force-android-studio-to-start-with-the-welcome-screen>。）

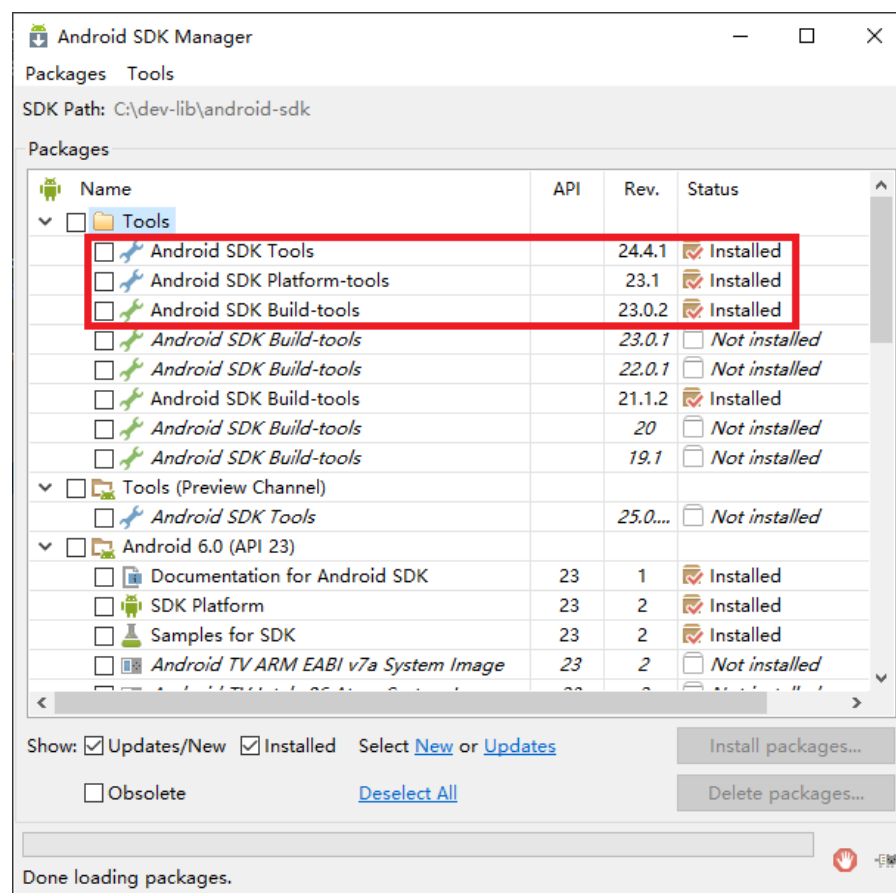
另外，我是在暗色主题下截取的图片。切换Android Studio主题的方法是，依次选择File->Settings...->Appearance & Behaviour->Appearance，在Theme后的下拉菜单中选择Darcula。

编辑器默认的字体比较小，调节字体的方法如下：首先依次选择File->Settings...->Editor->Colors & Fonts->Font，默认的主题是无法直接修改的，所以要先点击一下Save As...，新建一个相同的主题，然后调整Size值，如下图所示：

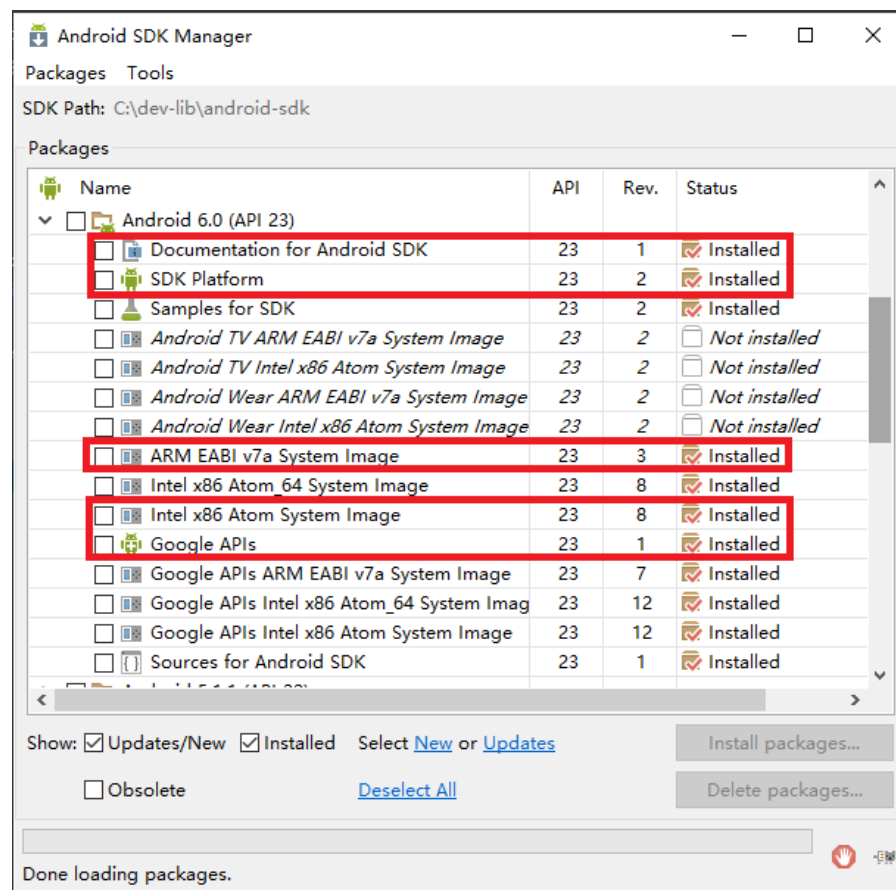




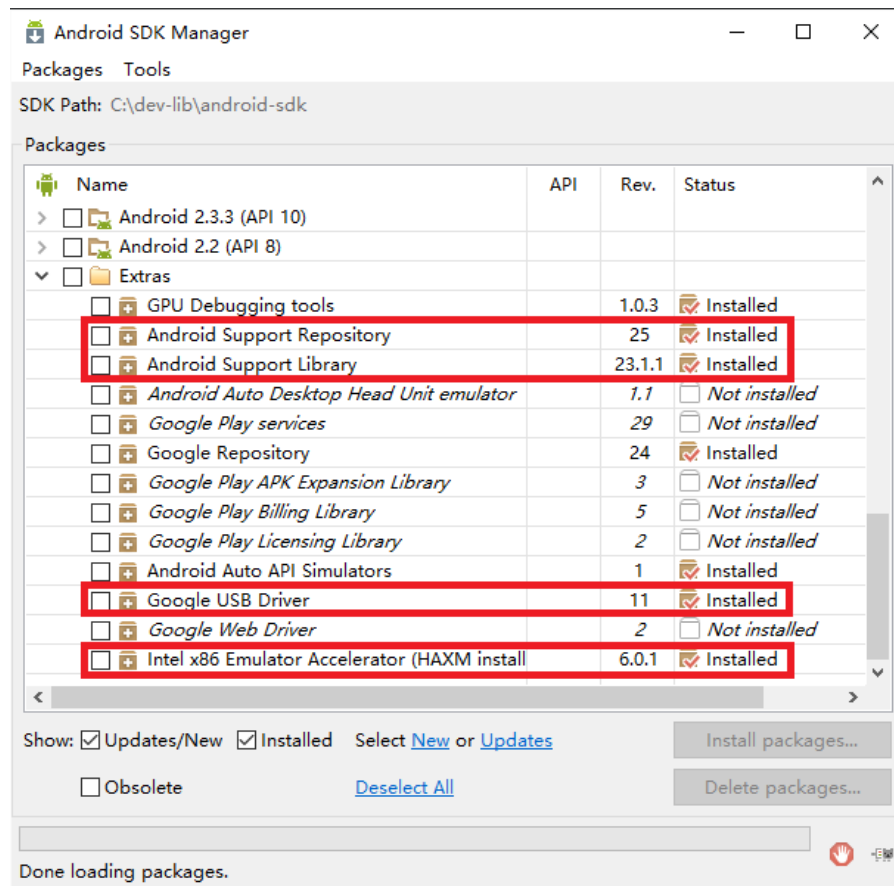
在弹出的Android SDK Manager界面中，选择以下若干项：



因为我同时也安装了API 21的SDK，所以我也勾选了Build-tools 21.1.2。



其他包如Samples for SDK与Sources for Android SDK是可选的。虽然平时采用x86架构的手机并不多见，但我们安装x86系统镜像主要是为了模拟器用。用Android Studio自带的AVD Manager创建虚拟设备时，它会推荐你使用带有Google API的镜像，所以我把所有镜像都安装了。



这里的HAXM驱动结合上一图中勾选的x86系统镜像，在大多数Intel CPU上可以明显提高模拟器的运行速度。其他包是可选的。

选择完毕后，点击Install packages...就可以开始安装了。安装过程十分耗时，而且如果网络不稳定的话还会经常断开连接，所以这个过程很有耐心。

安装完毕后，关闭Android SDK Manager。接下来就可以创建虚拟设备了。

# Android模拟器

## 1.AVD Manager

这是Android SDK自带的Android模拟器，也是Android Studio默认的模拟器。

如果你之前安装了HAXM驱动，打开C:\dev-lib\android-sdk\extras\intel\Hardware\_Accelerated\_Execution\_Manager，准备安装该驱动。先打开命令提示符，把haxm\_check.exe拖动到命令提示符窗口内，按回车，检查你的CPU是否支持VT和NX：



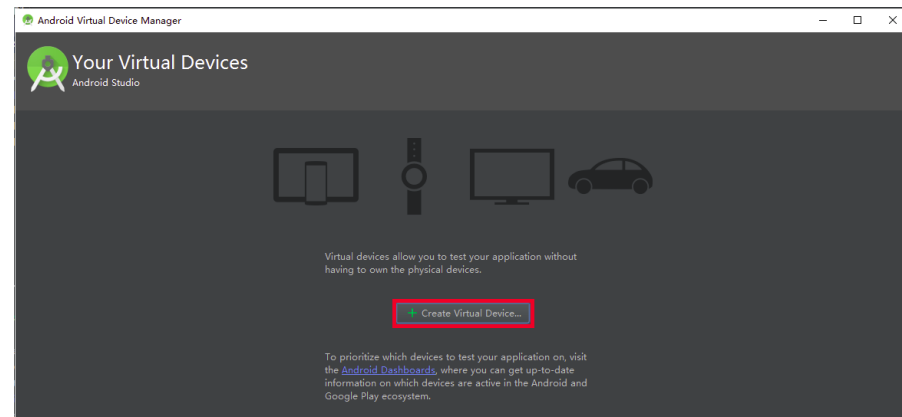
```
命令提示符
Microsoft Windows [版本 10.0.10586]
(c) 2015 Microsoft Corporation. 保留所有权利。

C:\Users\john>C:\dev-lib\android-sdk\extras\intel\Hardware_Accelerated_Execution_Manager\haxm_check.exe
VT support -- yes
NX support -- yes

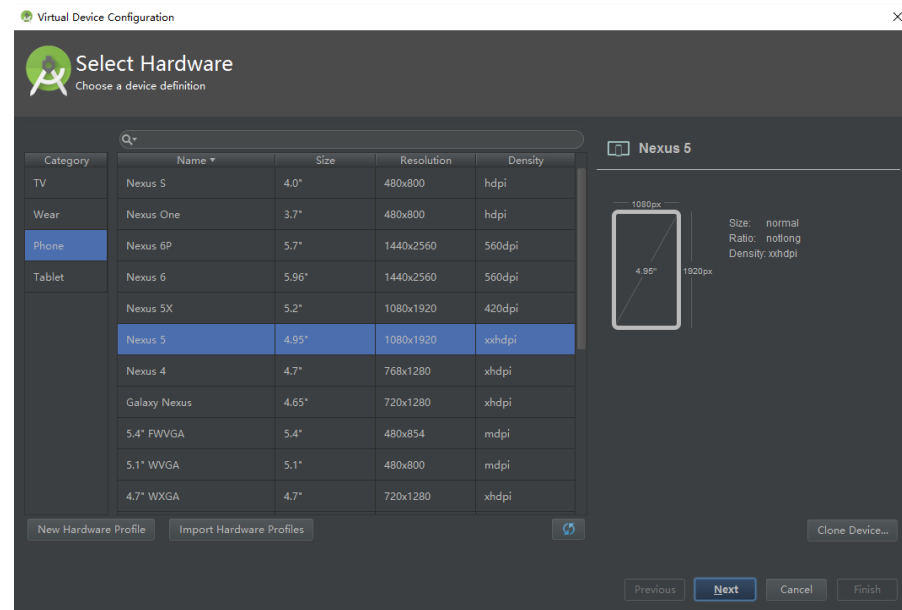
C:\Users\john>
```

如果二者都为Yes，双击intelhaxm-android.exe，安装HAXM。在安装过程中会询问为模拟器占用多少内存用来优化速度，采用默认值就好。

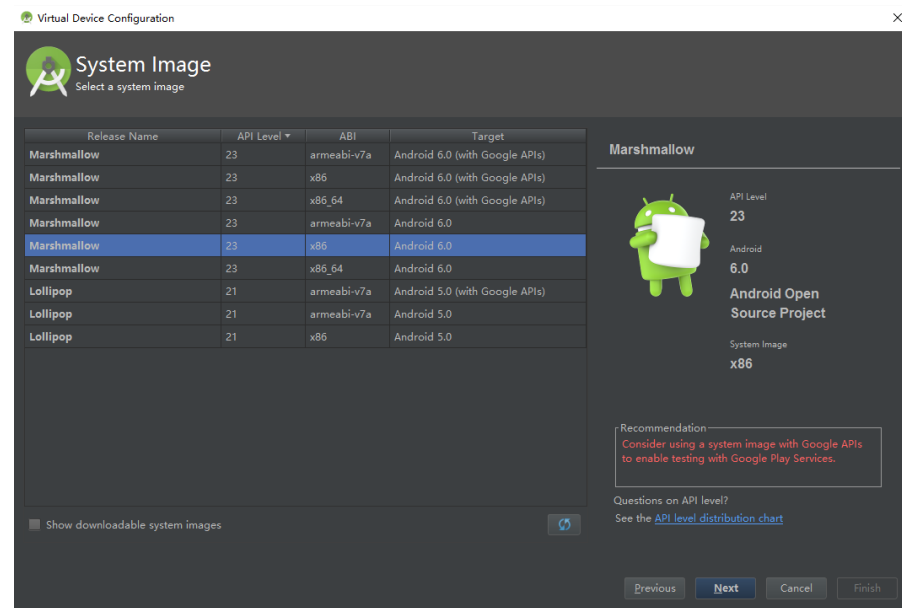
点击Android Studio上方的按钮，打开Android Virtual Device Manager。点击Create Virtual Device...：



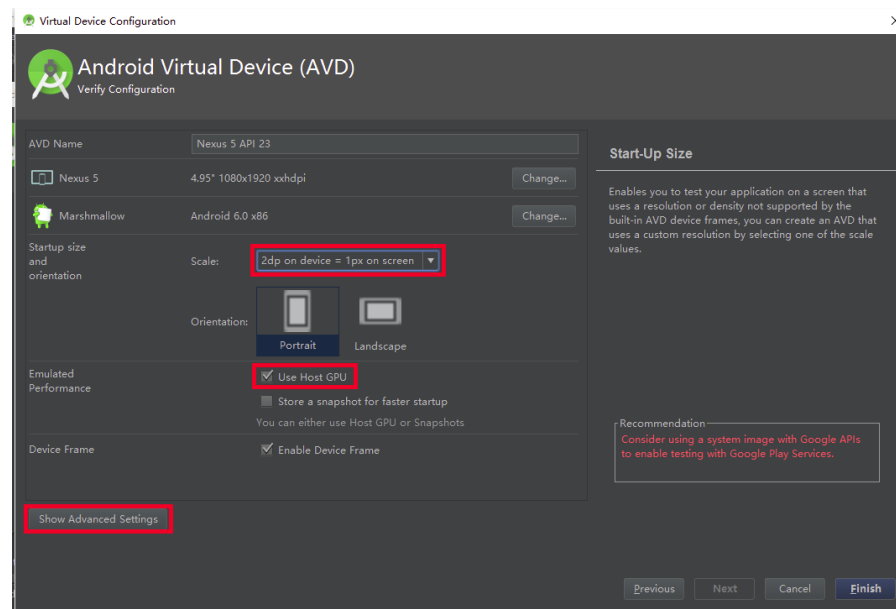
第二步选择虚拟设备的外观与尺寸，这里采用默认Nexus 5：



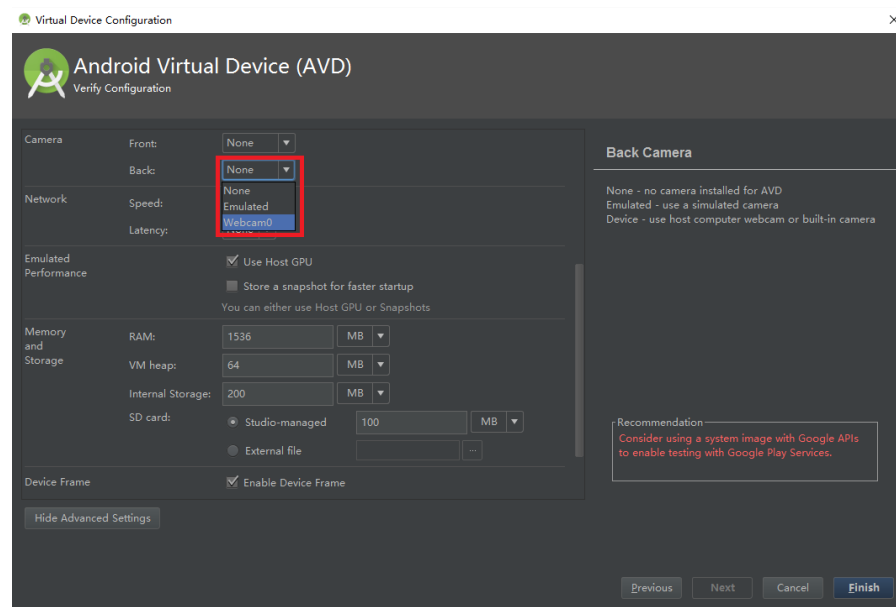
下一步，选择模拟器的Android版本。这里我选择API 23，ABI x86的：



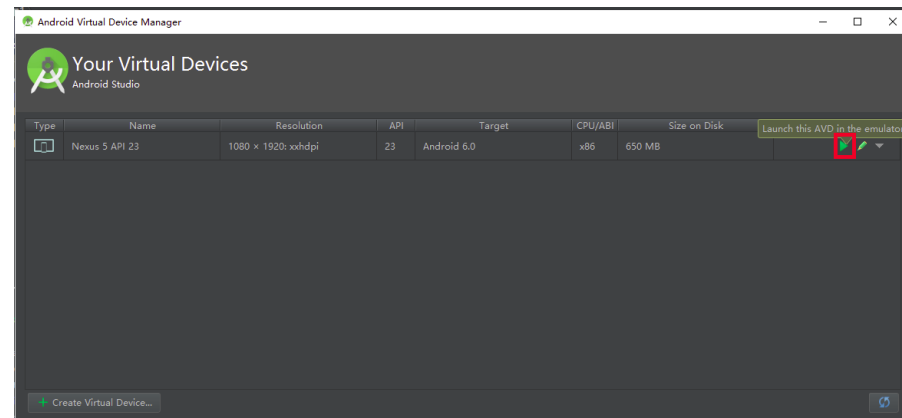
再点击Next，设置虚拟设备的详细属性。如果之前安装了HAXM，这里就要勾选Use Host GPU，以提高模拟器的运行速度。在Android Studio 2.1.2中，该项显示为Graphics，如果你的显卡支持OpenGL，可以选择Hardware – GLES 2.0。还有需要注意的是，由于虚拟设备本身屏幕分辨率就比较高（一般为1080×1920以上），我们还要根据计算机的实际屏幕分辨率来设置Scale，我一般选择2dp on device=1px on screen，相当于把虚拟设备的屏幕长宽缩小了一半。如果是在笔记本上进行开发，尤其是Orientation采用Portrait（竖屏模式）时，这个比例还要调高一些：



点击左下角的Show Advanced Settings，可以设置更详细的参数。在开发图像处理应用时，最好为虚拟设备添加一个摄像头：



点击Finish，虚拟设备就创建好了。在AVD Manager界面中点击对应虚拟设备后面的三角形按钮，就可以启动模拟器，打开相应的虚拟设备：



Android Studio自带模拟器的一个很大的弊端是，运行缓慢。甚至在我的年老体衰的笔记本电脑上根本无法启动。在使用这个模拟器时，要注意机器的空闲内存、HAXM占用内存与虚拟设备RAM之间的协调。否则即使安装了HAXM，也可能由于内存不足的原因无法开启加速模式，导致运行速度更慢。

## 2.Genymotion


Genymotion是Genymobile开发的一款Android模拟器，运行速度比Android SDK自带的模拟器快很多，而且提供了包括网络、摄像头、电池电量、信号强度、GPS等参数的模拟。在某些情况下，它的运行速度甚至比实际设备还要快。Genymotion的官方网站是<https://www.genymotion.com/>，需要注册一个账户才能下载，选择personal use就能免费使用大部分功能。

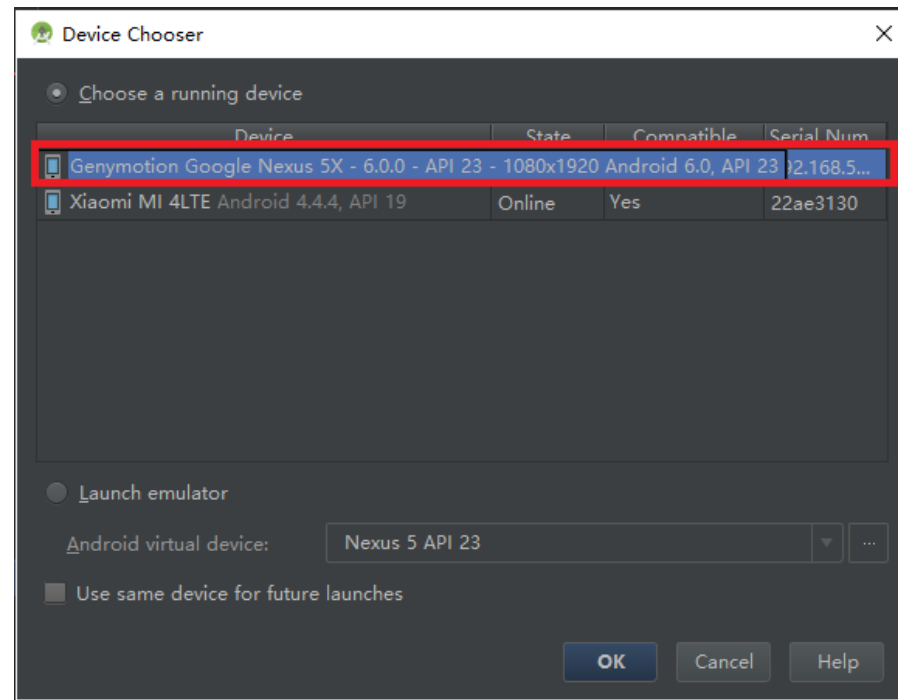


下载并安装好Genymotion之后，打开软件，点击Settings->ADB->Use custom Android SDK tools，然后选择之前下载的Android SDK目录，比如C:/dev-lib/android-sdk。如果下方显示一个对号，则说明Android SDK配置成功，无需为Genymotion单独下载一个Android SDK了。

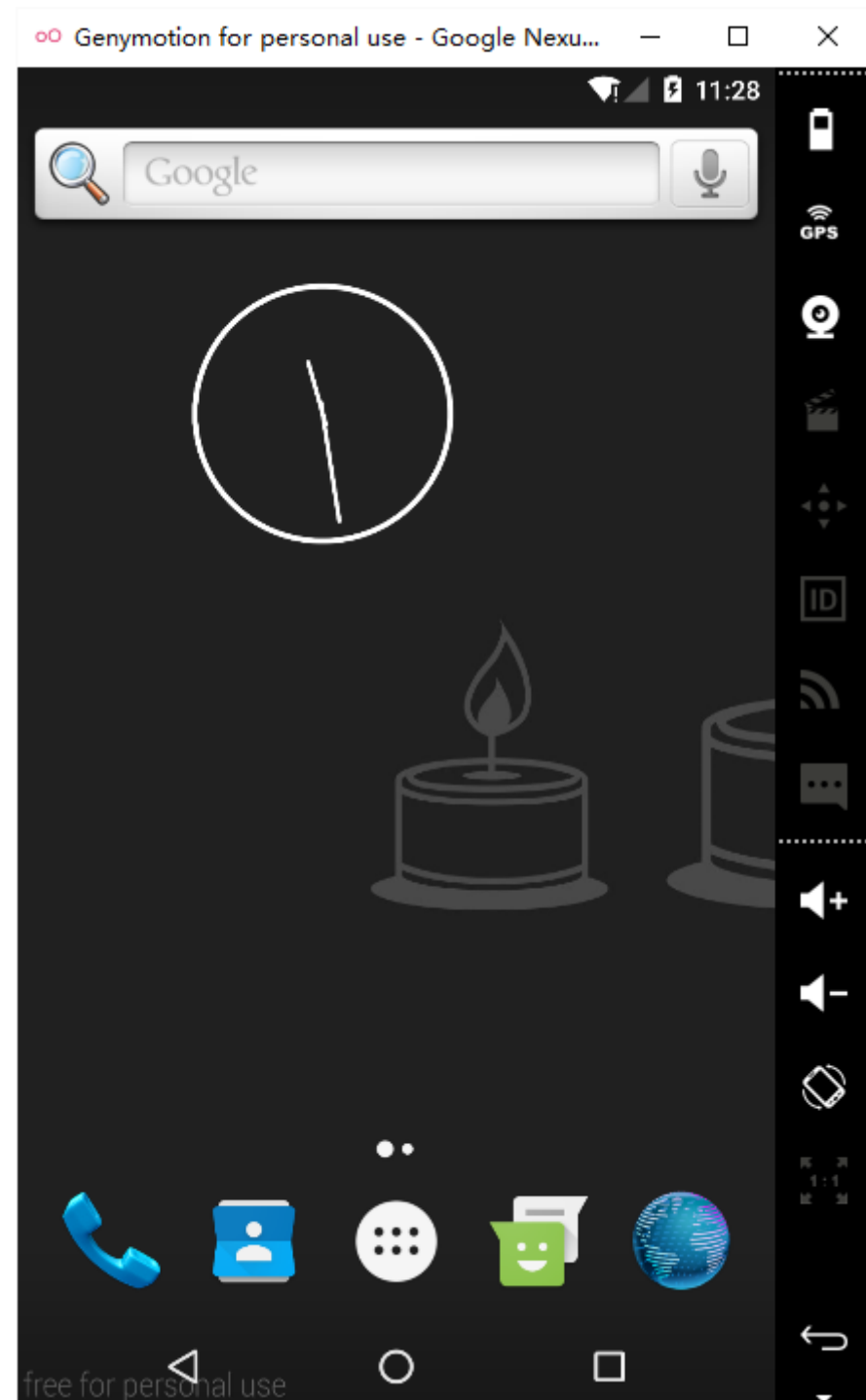
此外，还要为Android Studio安装一个Genymotion插件。打开Android Studio，依次点击File->Settings...->Plugins->Browse repositories...，在左上角的搜索框搜索“genymotion”，找到Genymotion，点击Instal plugin。安装完成之后，关闭并重新打开Android Studio。（参考：[https://docs.genymotion.com/pdf/PDF\\_Plugin\\_for\\_Android\\_Studio/Plugin-for-Android-Studio-1.0.7-Guide.pdf](https://docs.genymotion.com/pdf/PDF_Plugin_for_Android_Studio/Plugin-for-Android-Studio-1.0.7-Guide.pdf)）

点击Android Studio上方的按钮，就能打开Genymotion了。

用Genymotion创建虚拟设备的过程非常简单，这里就不赘述了。在需要调试应用时，只需要点击按钮，然后在弹出的Device Chooser窗口中选择Genymotion的虚拟设备即可：



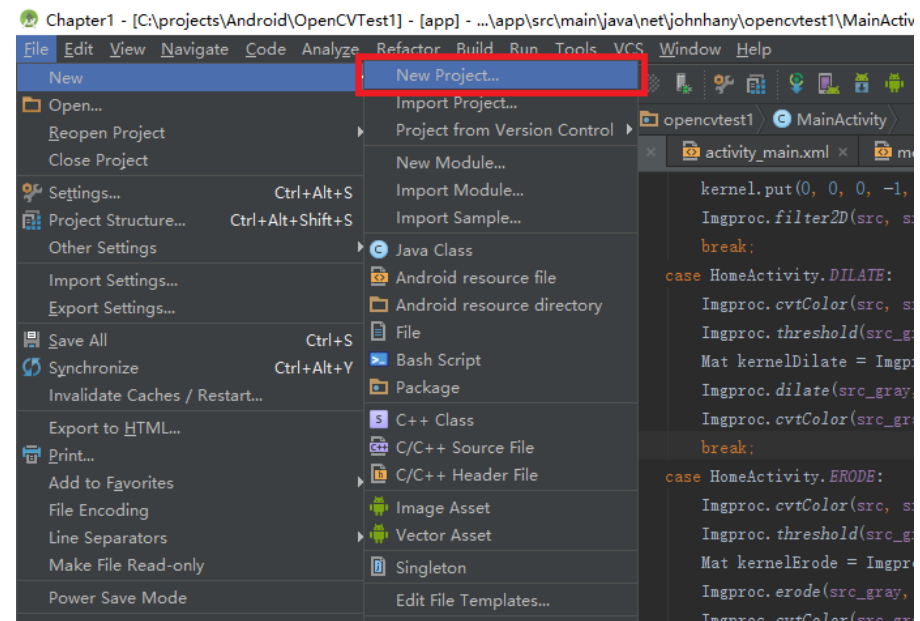
Genymotion模拟器运行时的效果如下：



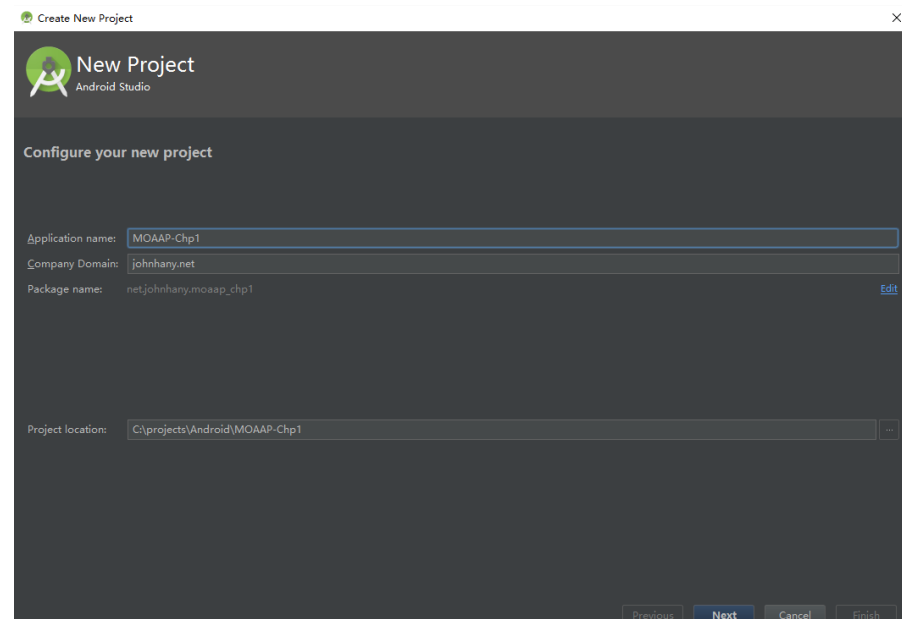
有时即使设置了自定义的Android SDK路径，新建虚拟设备时Genymotion还是会自己下载一个两百多MB的SDK。也许是同一API的SDK具体版本不同造成的，对此我也没有什么办法，好在下载速度不是太慢：)

## 为项目配置OpenCV

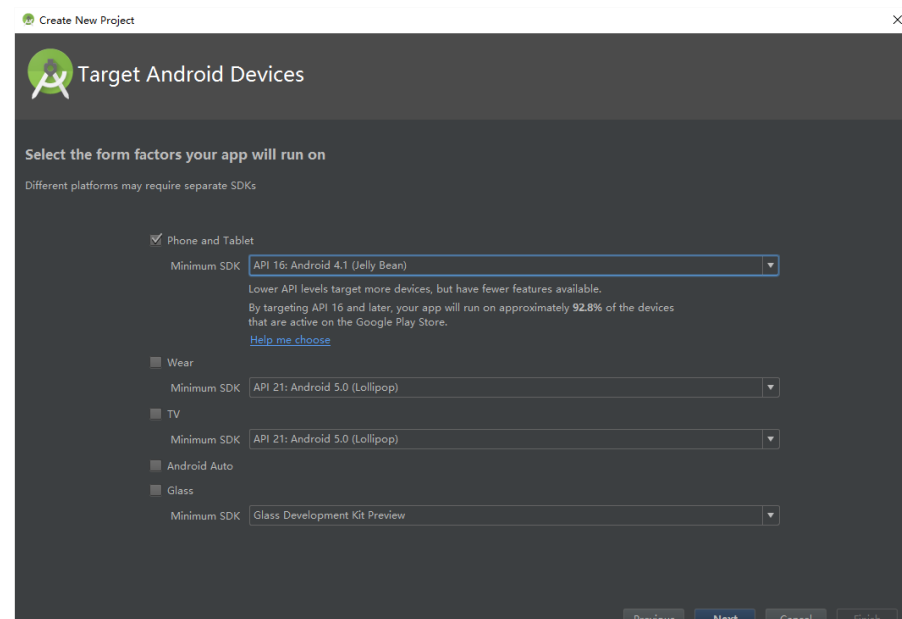
1. 建立Android项目。打开Android Studio，点击File->New Project...：



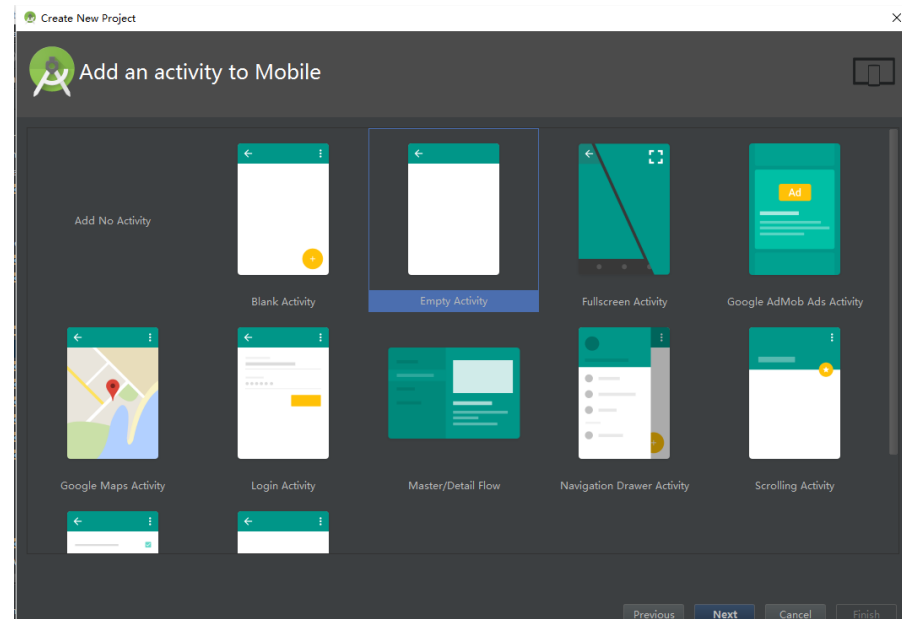
为项目设置应用名称、包名前缀和项目位置：



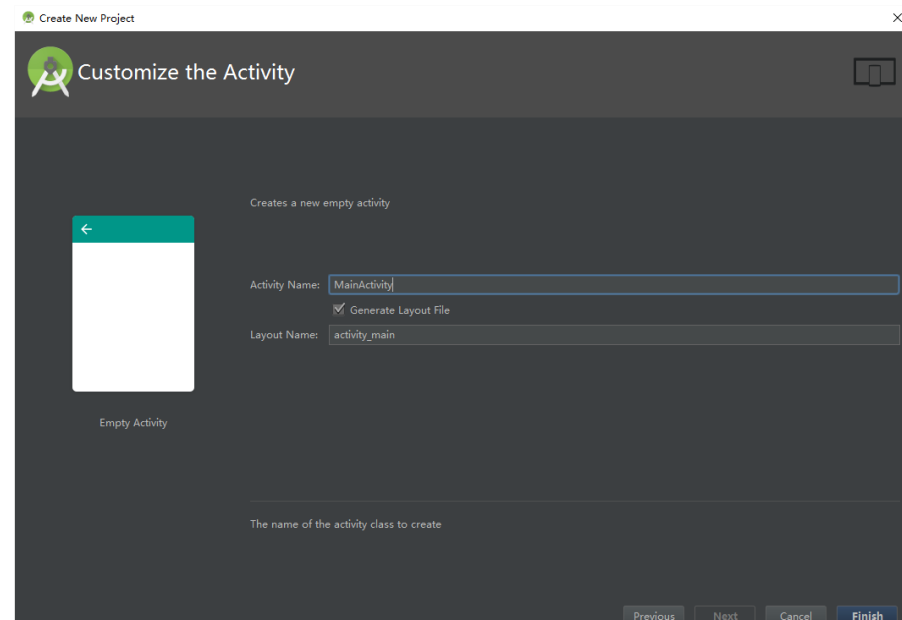
点击Next。设置应用支持的最低Android API级别，这里设置为API 16：



点击Next。选择默认Activity的外观，为了让界面尽量简洁，这里选择Empty Activity：

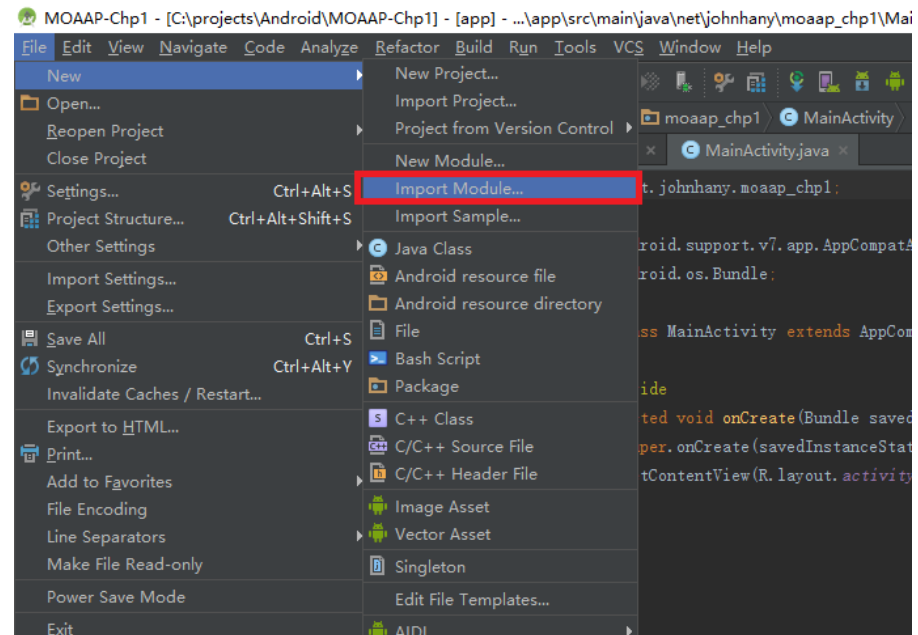


点击Next。设置Activity文件名称以及对应Layout文件名称，这里采用默认值：

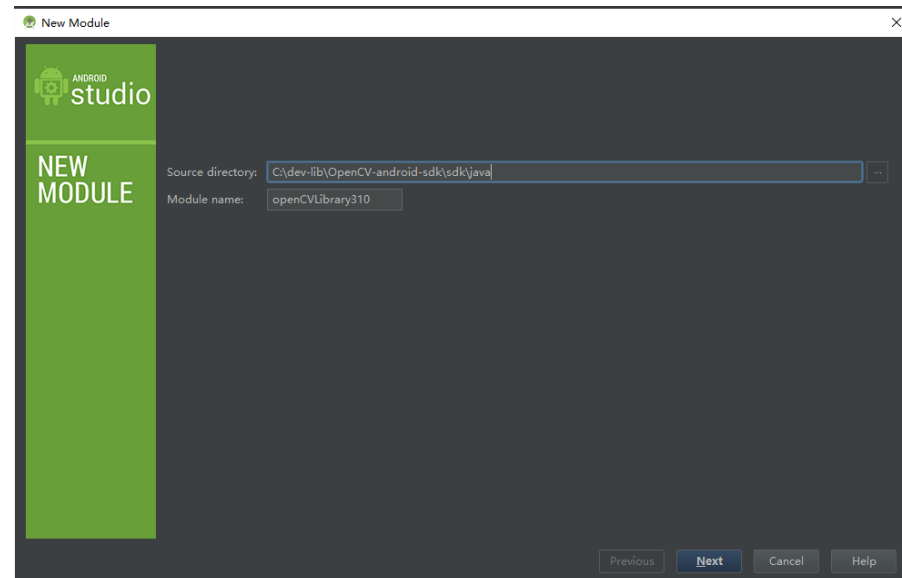


点击Finish。等待Android Studio载入相关文件、同步配置并自动编译之后，一个基本的Android项目就建立好了。

2. 导入OpenCV库。点击File->New->Import Module...：

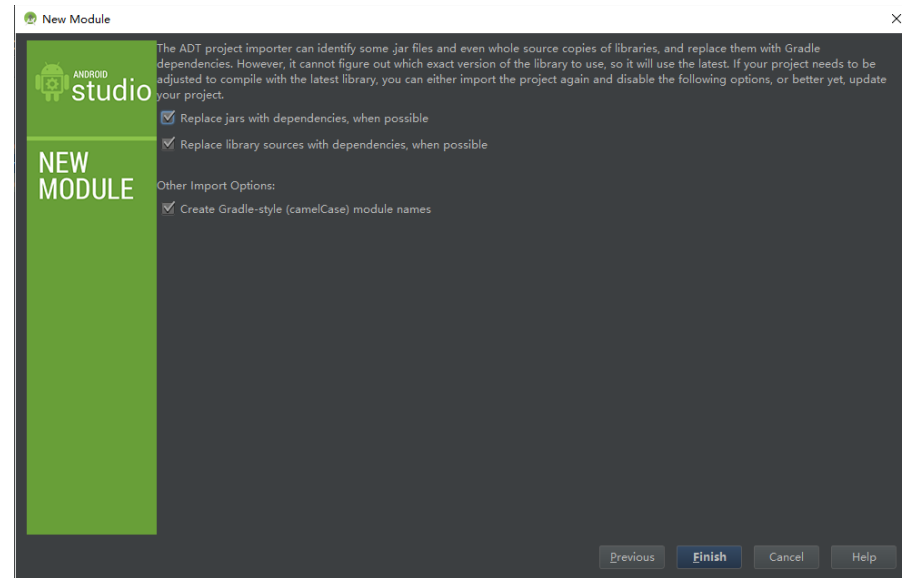


在弹出的窗口中找到OpenCV Android SDK下的sdk\java文件夹，比如C:\dev-lib\OpenCV-android-sdk\sdk\java：

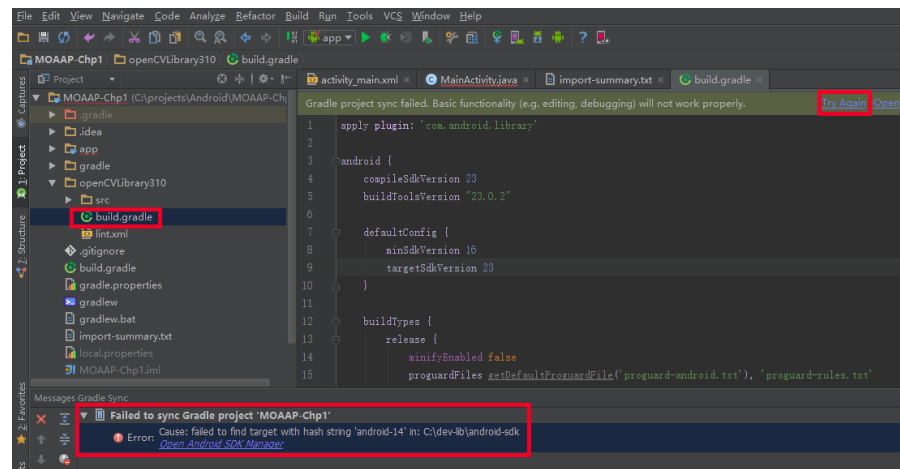




点击Next。保持默认选项不动，再点击Finish，完成OpenCV库的导入：



此时，Android Studio会自动更新项目配置。由于OpenCV库本身默认配置的原因，在这个过程中会产生很多错误（如下图最下边的红框所示）。这就需要我们修改OpenCVLibrary310\build.gradle文件，然后重新同步Gradle配置：




修改OpenCVLibrary310\build.gradle文件为如下内容：

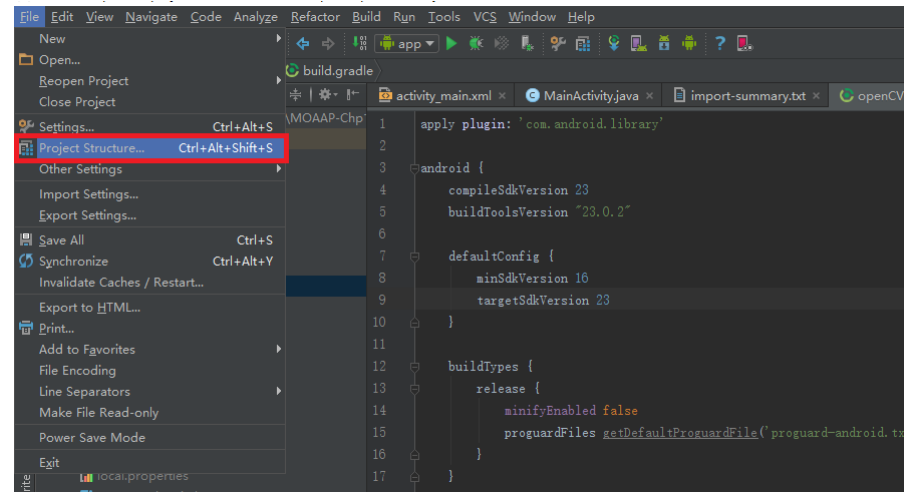
```


1  apply plugin: 'com.android.library'
2
3  android {
4      compileSdkVersion 23
5      buildToolsVersion "23.0.2"
6
7      defaultConfig {
8          minSdkVersion 16
9          targetSdkVersion 23
10     }
11
12     buildTypes {
13         release {
14             minifyEnabled false
15             proguardFiles getDefaultProguardFile('proguard
16         }
17     }
18 }

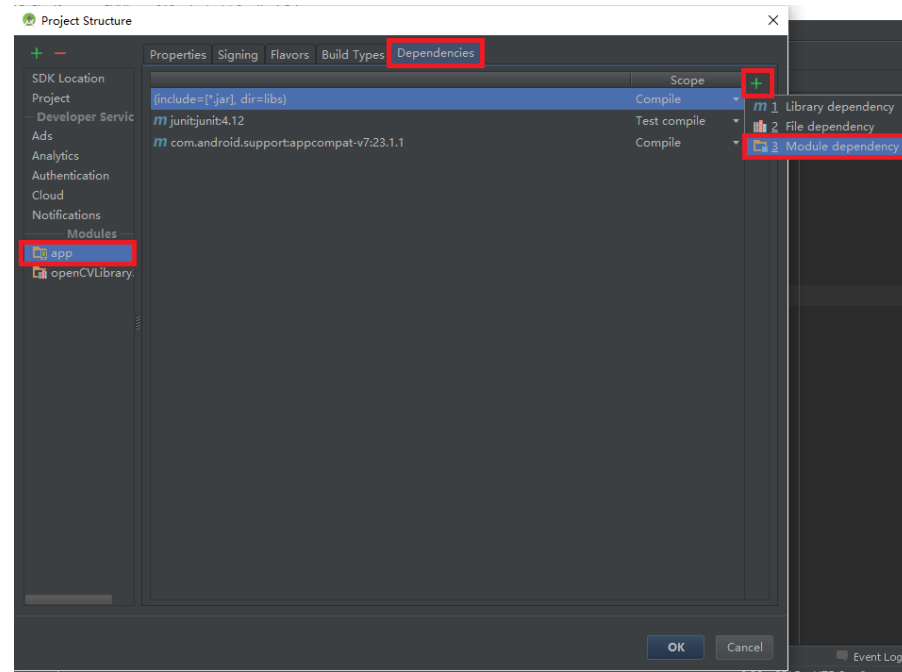
```

修改好后，点击上面图片中标示出的Try Again，或者点击Android Studio上方的按钮，同步项目的Gradle配置。同步完成后，就可以向Android项目中整合OpenCV模块了。

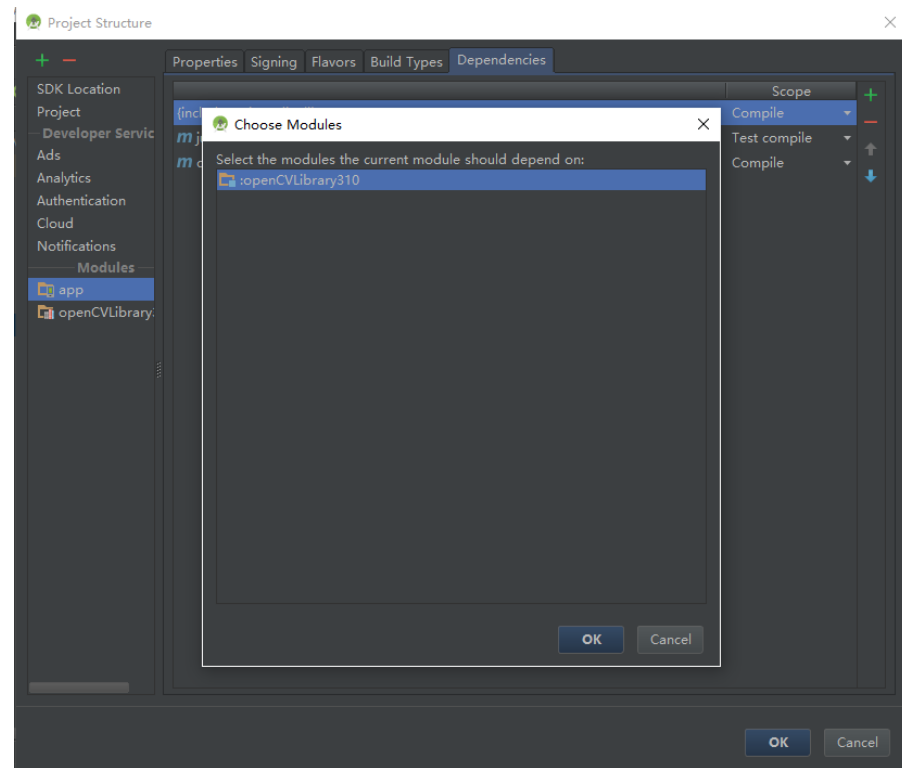
### 3.将Android项目与OpenCV整合。点击File->Project Structure... :



在弹出的Project Structure窗口中选择左边的app，在右边选择Dependencies，然后单击  按钮，选择Module dependency：

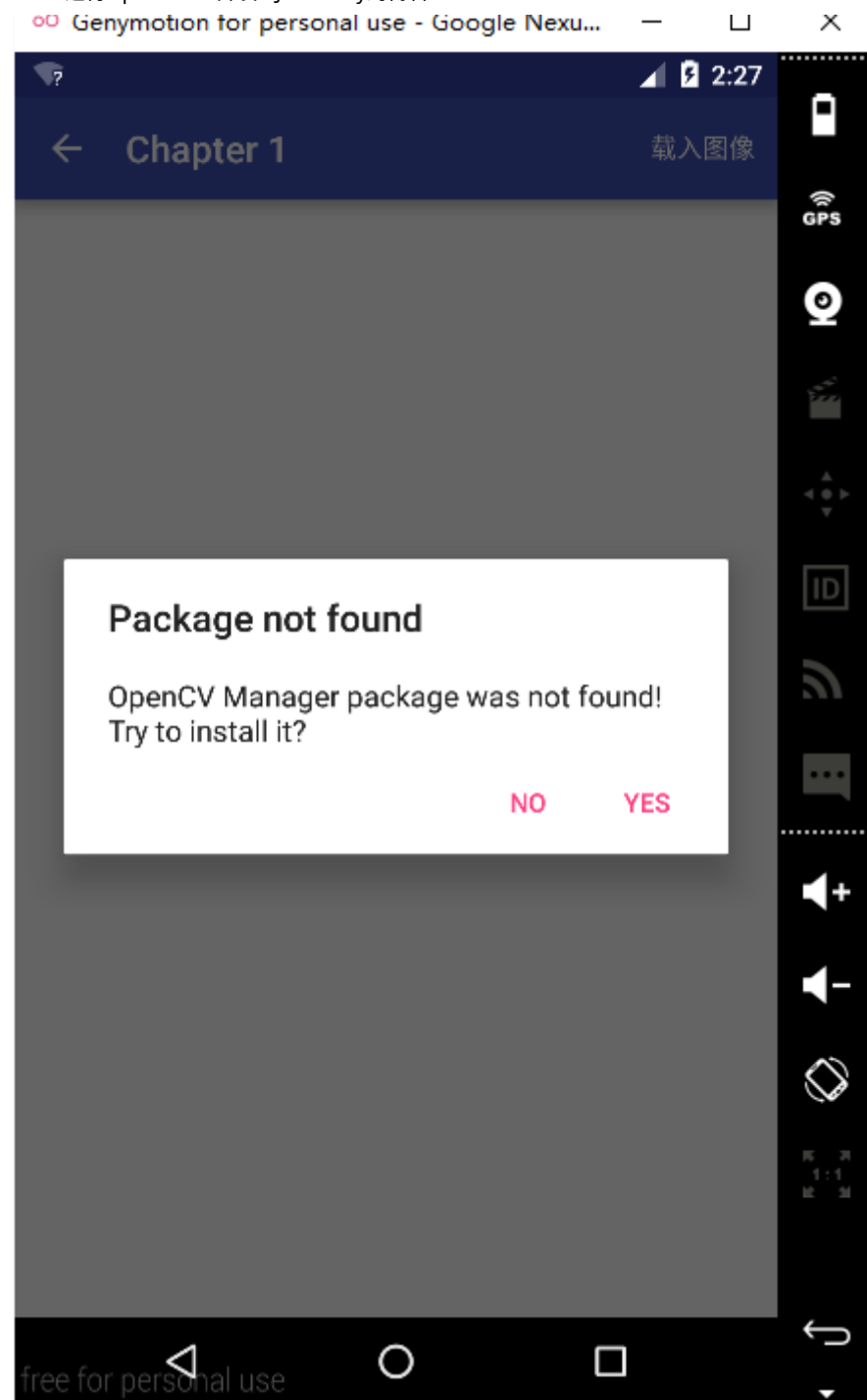


在弹出的窗口中选择OpenCVLibrary310，点击两次OK，保存并关闭两个窗口：



至此，你的Android项目就支持OpenCV的开发了！

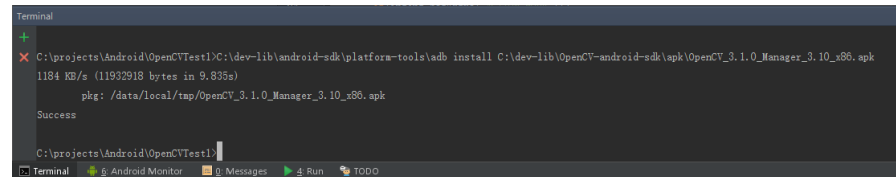
另外，在设备上调试APP之前，不要忘记先安装对应架构的OpenCV Manager，该软件包包含了所有OpenCV的库函数。否则会出现Package not found的错误：



OpenCV Manager位于OpenCV Android SDK的apk目录下，比如C:\dev-lib\OpenCV-android-sdk\apk，然后根据你的设备来选择安装包。比如，如果手机的CPU采用ARM v7架构，就要安装OpenCV\_3.1.0\_Manager\_3.10\_armeabi-v7a.apk；如果是x86架构，就要安装OpenCV\_3.1.0\_Manager\_3.10\_x86.apk。

对于实际设备，把OpenCV Manager的apk直接拷贝到设备安装即可；如果是虚拟设备，需要在命令提示符中用adb install命令来安装，比如：

```
1 | C:\dev-lib\android-sdk\platform-tools\adb install C:\dev-li
```



类别： [Android](#)，[图像处理](#)      标签： [android](#)，[opencv](#)，[环境配置](#)

Windows7+VS2012下64位  
OpenCV3.0+CUDA7.5的编译和  
部署

HanyNet – 卷积神经网络的初步  
实现

41 评论 在 "在Android Studio上进行OpenCV 3.1开发"

订阅评论

所有新评论

邮箱



加入讨论

排序: 最新|最旧



冰淇淋



博主，你好，最近在忙着用opencv做一个拍照解数独的app，在研究深入Android Opencv应用开发这本书，发现里面的知识还不够详细，能告诉我哪里能找到opencv-sdk-for android用法更详细的说明吗？

回复

2017年12月1日 21:22 ^



John Hany



你好，android版本的opencv也是由原本为PC设计的C++源码发展而来，单独介绍opencv android sdk的资料不是太多，建议你把opencv和android分开学习。opencv主要了解其C++版本，可以直接看官方文档（<https://docs.opencv.org/>），CSDN上也有很多用心的opencv教程。android方面有问题尽管在<https://stackoverflow.com/questions/tagged/android>搜索，基本上都有解决方法。



[← 回复](#)

🕒 2017年12月1日 22:09



## 深入OpenCV Android 应用开发 中文版 – 第 七章代码更新 – JohnHany的博客



发》。 [...]

[...] 关于在Android Studio上配置  
OpenCV开发环境的方法，请参考《在  
Android Studio上进行OpenCV 3.1开

[← 回复](#)

🕒 2017年8月6日 20:31



## 深入OpenCV Android 应用开发 中文版 – 第 六章代码更新 – JohnHany的博客



发》。 [...]

[...] 关于在Android Studio上配置  
OpenCV开发环境的方法，请参考《在  
Android Studio上进行OpenCV 3.1开

[← 回复](#)

🕒 2017年8月5日 14:34



## 深入OpenCV Android 应用开发 中文版 – 第 五章代码更新 – JohnHany的博客



发》。 [...]

[...] 关于在Android Studio上配置  
OpenCV开发环境的方法，请参考《在  
Android Studio上进行OpenCV 3.1开

[← 回复](#)

🕒 2017年8月4日 18:07



## 深入OpenCV Android 应用开发 中文版 – 第 四章代码更新 – JohnHany的博客



[...] 关于在Android Studio上配置OpenCV开发环境的方法，请参考《在Android Studio上进行OpenCV 3.1开

发》。 [...]

回复

2017年8月4日 12:54



## mysteryH



感谢楼主，满满干货。然而现在还是不知道怎么下手开发。想问下楼主，由于我这边是有份opencv的c++代码，想移植过去，我要怎么做？

回复

2017年3月15日 00:19 ^



## John Hany



多谢！因为现在android studio 2开始引入了新的建立ndk项目的方法，我暂时还没有更新上来，而且老方法也是很容易出错的。建议你先到官方文档了解一下，等我更新新的文章再欢迎你来指正！

回复


2017年3月18日 23:55



## chenzhikai



如何不使用那个manager来实现opencv的功能，求大神解答

 回复 2016年8月29日 11:06

深入OpenCV Android  
应用开发 中文版 – 第  
三章代码更新 –  
[JohnHany的博客](#)



[...] 关于在Android Studio上配置  
OpenCV 3.1开发环境的方法，请参考  
《在Android Studio上进行OpenCV 3.1开  
发》。 [...]

 回复 2016年7月29日 20:16

深入OpenCV Android  
应用开发 中文版 – 第  
二章代码更新 –  
[JohnHany的博客](#)



[...] 关于在Android Studio上配置  
OpenCV 3.1开发环境的方法，请参考  
《在Android Studio上进行OpenCV 3.1开  
发》。 [...]

 回复 2016年7月23日 22:08

深入OpenCV Android  
应用开发 中文版 – 第  
一章代码更新 –  
[JohnHany的博客](#)



[...] 关于在Android Studio上配置  
OpenCV 3.1开发环境的方法，请参考  
《在Android Studio上进行OpenCV 3.1开  
发》。 [...]

 回复 2016年7月11日 00:31



## Android Studio 2上整合NDK进行OpenCV

3.1开发 – JohnHany的 [...] 单纯为Android Studio配置OpenCV 3.1环境的方法请见《在Android Studio上进行OpenCV 3.1开发》。 [...]



回复

2016年7月9日 00:52



sergio



LZ又没在AS上配置JNI的教程 想用c++调opencv在as做开发 看过两个 试着配置都没有成功

回复

2016年6月7日 13:19 ^



John Hany



你好！JNI的教程刚刚更新，在首页就能看到，久等了

^\_^

回复

2016年7月9日 01:43





licvv



多谢分享，我现在也想在AS上使用opencv，但我不明白的是opencv是用c++写的，在安卓上是直接编译吗，还是转成Java代

码？

 回复 2016年5月17日 17:17 

John Hany



有两种方式：C++的代码可以通过NDK编译为可以被Java调用的.so库；OpenCV也提供了Java的接口，可以在Android上直接使用Java接口来进行处理。从性能的角度来讲，推荐采用NDK的方式。

 回复 2016年5月18日 21:43 

licvv



我就是对NDK编译这块比较困惑，我看的demo里对原来的opencv代码有改动，但又看不懂是怎么改的，楼主有没有教程之类的或是简单点的demo啊？

 回复 2016年5月19日 18:40 

John Hany



基础一些的样例代码还在整理，过一阵子就好了：)

 回复 2016年5月20日 00:10 



licvv



好的，博主辛苦了！

回复

🕒 2016年5月20日 09:32



Windows下Android开  
发环境的配置 –  
JohnHany的博客



[...] 注：使用Eclipse+ADT进行Android开发的方式已经过时，推荐阅读《在Android Studio上进行OpenCV 3.1开发》。 [...]

回复

🕒 2016年5月12日 12:56



wuwei



博主文章很有用，一直没配置好，看了你的文章立马解决了，不过我是android新手，能不能给出一点用android与opencv开发APP的实例呢，非常感谢！

回复

🕒 2016年5月10日 19:09 ^



John Hany



很高兴我的文章对你有所帮助！我正在整理一些Android OpenCV开发的样例代码，过一段时间会发出来。再次感谢你的支持！

[← 回复](#)

🕒 2016年5月11日 12:39 ^



wuwei



嗯嗯，那发表出来记得@我一下，一定膜拜，谢谢啦！

[← 回复](#)

🕒 2016年5月13日 09:15



coffee



您好，最近刚刚接触Android Opencv开发，环境配置和博主的基本一样，但在编写一个程序时报错 java.lang.UnsatisfiedLinkError: Native method not found: org.opencv.core.Mat.n\_Mat()J，在下是新手不是很理解，包不是都导入了吗？希望您能解答 感激不尽

[← 回复](#)

🕒 2016年5月7日 00:18 ^



John Hany



没有完整的代码作参考，我猜想原因可能是

- 1.为android项目配置opencv环境本身存在问题
- 2.在java代码中在载入opencv库之前调用了opencv的接口（比如对opencv的操作应该在case LoaderCallbackInterface.SUCCESS之后，而不是在onCreate之中）
- 3.没有载入自定义的native库就调用了相应的函数

方便的话，贴上关键的代码，以便缩小错误的范围：)

回复

2016年5月7日 16:35



阿斯顿



大神，什么时候上AS的samples 呀？等着学习 ^-^

回复

2016年4月27日 12:38



阿斯顿



博主你好，我搭建好了环境配置，但是不知道怎么下手开发

回复

2016年4月24日 23:07 ^



John Hany



可以参考一下OpenCV Android SDK中的Samples，贴过来一些样例代码试试。过段时间我会发一些Android上OpenCV的代码，到时会通知你！

回复

2016年4月25日 15:02




changeway



请问博主，有没有研究过获取图片轮廓？没接触过OpenCV,目前无从下手，请赐教，哈哈哈



 回复 2016年4月19日 09:16 

John Hany



你指的是提取物体的轮廓么？可以参考一下这篇[官方教程](#)。

 回复 2016年4月19日 18:38

周



请问在不使用opencvmanager的情况下，怎么进行相关的开发呀？还有希望楼主出一个基于opencv的android人脸识别应用教程。博主很厉害，继续加油！！！！

 回复 2016年4月18日 18:00 

John Hany



多谢你的鼓励！可以把需要的opencv库文件单独打包在应用的目录下，但是这样又会大幅增加应用的大小。我目前正在整理一些教程，其中包含人脸识别的，预计过段时间可以发出来，到时候第一个通知你！

 回复 2016年4月18日 19:17 



sad完全

博主 Android Studio使用



OpenCV后，使APP不安装

OpenCV Manager即可运行 – Tail。 – 博客园 var  
currentBlogApp = ... [阅读更多 »](#)

回复

🕒 2016年4月23日 10:06



John Hany



仅仅更改这几行是不够的，我试验了一下，  
仍然需要安装opencvmanager。

回复

🕒 2016年4月25日 16:33



Polestar



把<OpenCV-android-sdk>/sdk/native/libs/<target\_arch>复制到<Project>/libs/<target\_arch>。

如果涉及Native Code的调用，修改Android.mk，加上两行代码：

```
OPENCV_CAMERA_MODULES :=  
OPENCV_INSTALL_MODULES :=
```



回复

🕒 2016年7月8日 00:15



wh0330



刚好在学习AS下的OpenCv，很有帮助，谢谢楼主！

回复

2016年4月13日 20:04 ^



John Hany



很高兴我的文章能够有所帮助，多谢支持！

回复

2016年4月13日 23:30



沙漠之风



写得真好

回复

2016年4月13日 00:53 ^



John Hany



多谢！多谢！

回复

2016年4月13日 08:33



Android上使用  
OpenCV处理图像 –  
JohnHany的博客



[...] 本文采用的Eclipse+ADT的开发流程目前已经过时，请参考《在Android Studio上进行OpenCV 3.1开发》，使用Android [...]



🕒 2016年1月28日 19:21



Proudly powered by WordPress | Theme: Hany Dark based on Underscores.