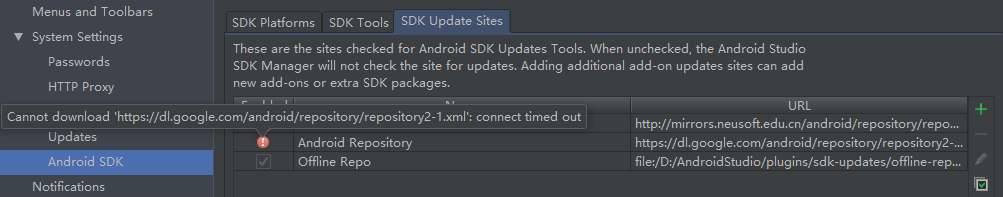
Tengine\_Android

（20181205）

1. open the existing project
2. 运行可能会显示：NDK not configured.

这是由于SDK更新的问题，因为在SDK manager当中显示的就是在dl.google.com上下载的SDK更新，实际上我们用的都是国内网站，至于怎么上google进行下载也不清楚。

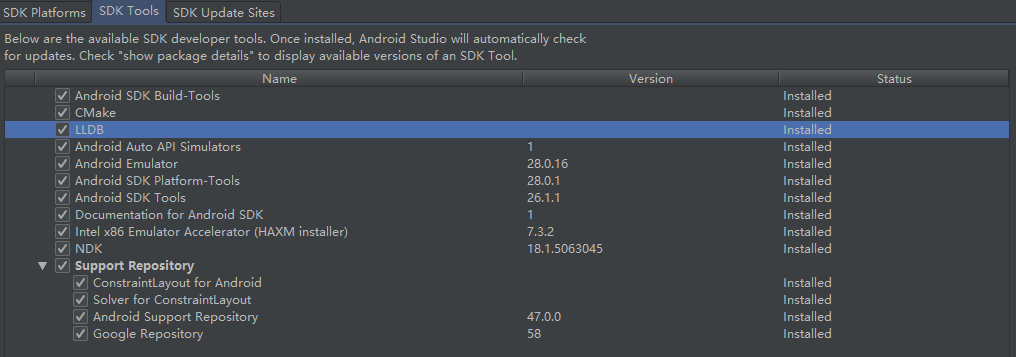


根据网上的一些讨论和博客发现遇到这个问题可以尝试修改一下hosts，这与翻墙可能有关。

<https://blog.csdn.net/xwq911/article/details/41823947>

1. 运行时可能还会显示错误：Failed to find Cmake.

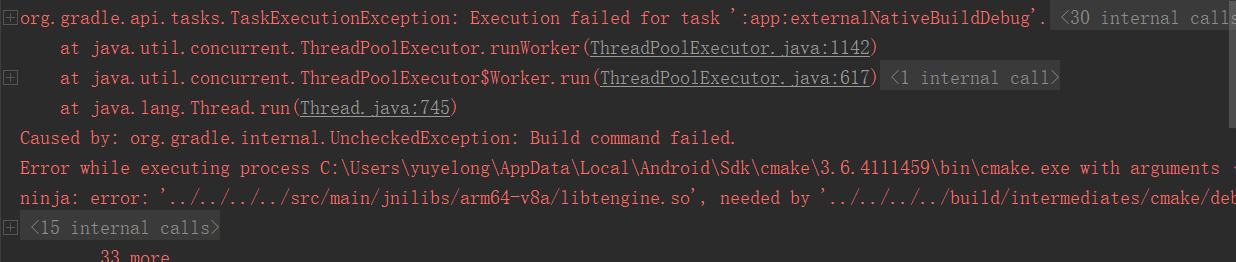
这也是更新问题，SDK manager 当中需要安装这几个更新：NDK,LLDB,CMake

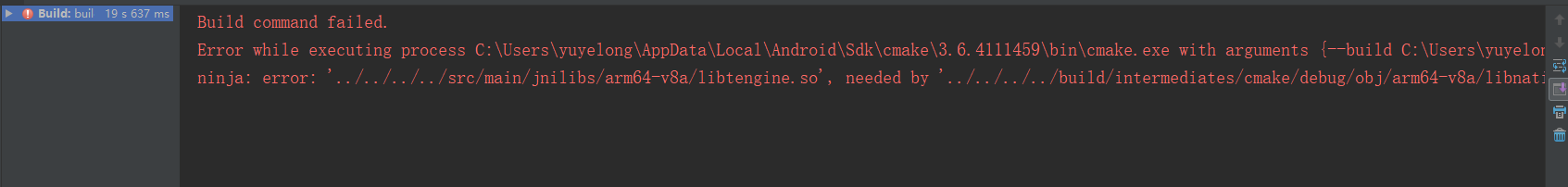


<https://blog.csdn.net/weixin_41189380/article/details/79049045>

（据说gradle failed是经常的事情，还有上述更新的问题与网络有关，多试几次就过去了）

1. 以上是在configure build上遇到的问题，上述解决办法可以排除这几个问题configure build成功，但是在Run tasks的时候又会遇到不同的问题：





阅读错误提示以后发现这是大概由于缺少了libtengine.so文件，于是乎我们就要去阅读一下tengine的东西：

在郑楷发的网页上有这么一句话：

* Android版参考 [build\_andorid](https://github.com/OAID/Tengine/blob/master/doc/build_android.md) 文档，在PC端的Linux系统上按文档编译Tengine，再通过adb移动到手机端，后续如果用Android Studio开发app，可以将编译得到的libtengine.so添加到工程里（目前没有参考的文档，如果遇到什么问题请及时跟我联系，我帮你找工程师问问，最近也尽量催他们出份Android Studio的开发文档）

这里链接一下郑楷的博客吧：

<http://hey-yahei.cn/timeline/>

build\_android文档链接如下：

<https://github.com/OAID/Tengine/blob/master/doc/build_android.md>

根据我的理解，按照上述说明结合例程大致意思是这样：

1. 首先需要编译Tengine（这部分需要在linux系统下面操作，我的ubuntu系统尚且搞 不懂，所以下面说的so文件是从公司那里拿的开源文件），生成一个libtengine.so 文件，将其放在以下目录：

project目录/app/src/main/jnilibs/libtengine.so

这样以后可以将整个工程文件编译成功，包括Run tasks部分

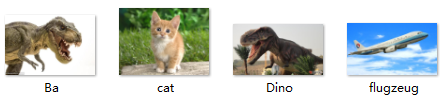
（......待补充so文件的作用）

1. 其次，有一些文件需要准备：

【1】生成的模型文件：frozen\_mobilenet\_v1\_224.pb（.pb文件是指tensorflow框架 下训练的模型，与此类似的有caffemodel文件的caffe模型,还有其他各种框架）

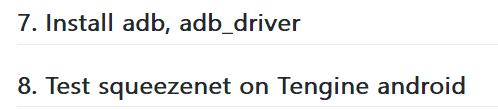
（......待补充pb文件的生成过程与frozen操作，里面应该包括一些训练模型）

【2】测试数据集：

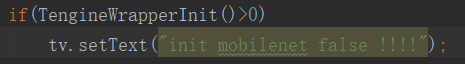


【3】label标签文件：synset\_words.txt

1. 根据build\_android文档说明的第7、8步需要将上述所说的数据通过adb传输到手机 上，不过文档当中说的要install adb与adb\_driver都是不需要的，因为Android Studio软件已经帮我们安装好了这些，具体地址在SDK安装目录下面（这个具体后面 部分会说）：C:\Users\yuyelong\AppData\Local\Android\Sdk\platform-tools



（：这里先说一下，一开始我因为不知道流程所以将上述文件都放在电脑端的project 工程文件夹下面，结果运行的时候出现错误：

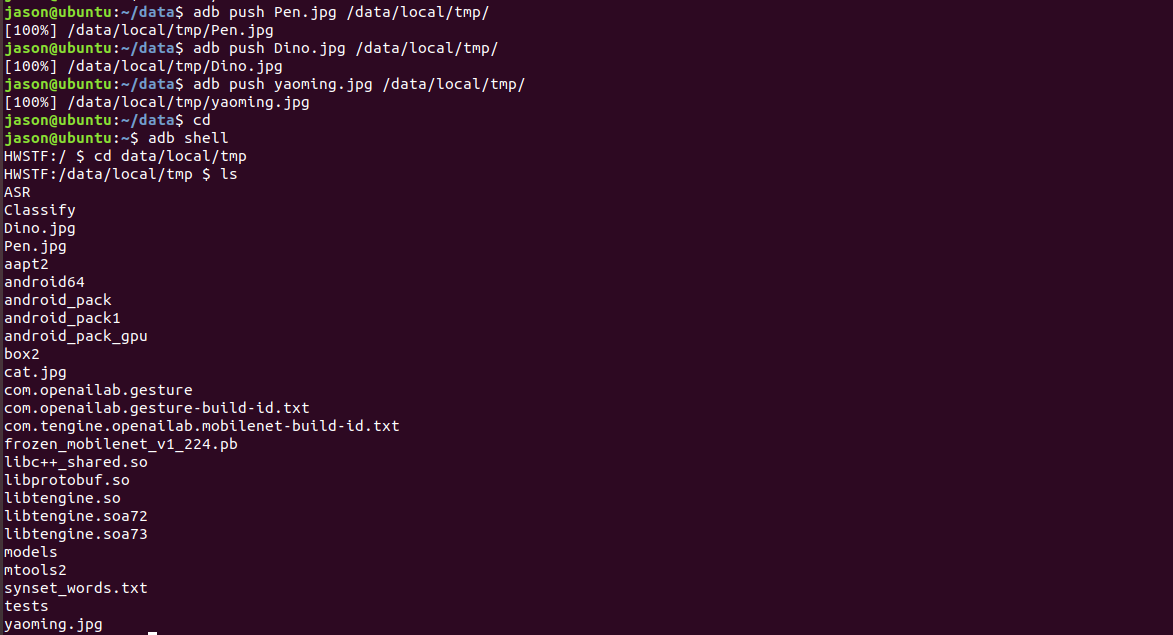


1. 那就说说怎么使用adb传输file到手机上吧，实际上可以进行双向传输：教程可以参 考：<http://adbshell.com/commands/adb-devices>

主要就是使用两个指令：adb push 与adb shell

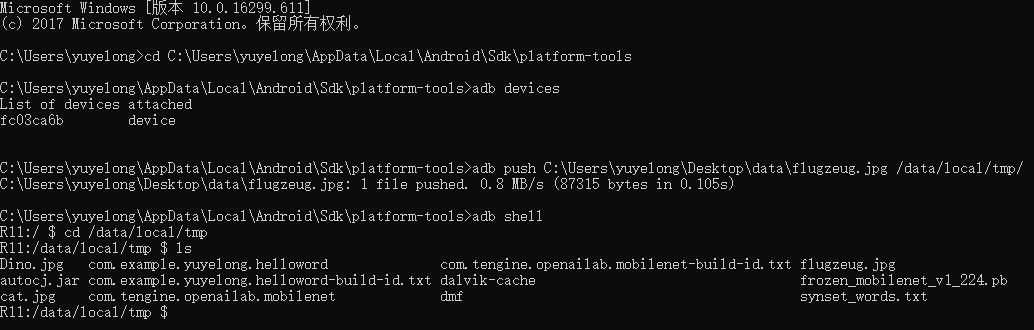
具体操作如下：这里我也想说说那个build\_android文档写的代码可读性不强的原因是 它省略了很多东西，我这里写的清楚点。

公司工程师做的实例：



结合自己的操作，详细如下，我觉得下面的文档谁看了都会做了哈哈，傻瓜式教学：

首先打开cmd（我都是在windows10下面完成的，这里不需要linux系统）：



第一指令行：cd C:\Users\yuyelong\AppData\Local\Android\Sdk\platform-tools进入adb 目录

第二指令行：C:\Users\yuyelong\AppData\Local\Android\Sdk\platform-tools>adb devices

查看adb连接设备，这里想说的是一定要把手机上的USB调试打开才能显示设备。

第三指令行：adb push C:\Users\yuyelong\Desktop\data\flugzeug.jpg /data/local/tmp/

adb push C:\Users\yuyelong\Desktop\data\cat.jpg /data/local/tmp/

adb push C:\Users\yuyelong\Desktop\data\model3model.pb /data/local/tmp/

adb push C:\Users\yuyelong\Desktop\data\yuyelong.txt /data/local/tmp/

adb push C:\Users\yuyelong\Desktop\data\myface.png /data/local/tmp/

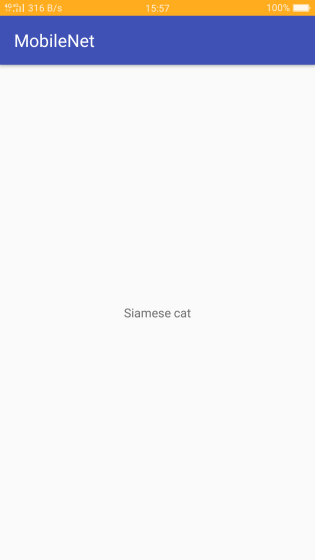
这里说说adb push的格式，这在之前的网页上有adb push和adb pull方面的说明

### adb push <local> <remote>

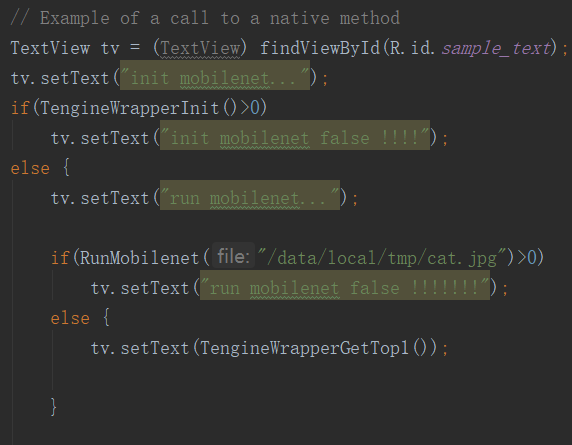
如果出现adb error:No such file or directory这样的错误一定一定要检查一下文件的命名

第四指令行：adb shell并进入cd /data/local/tmp可以查看手机文件的存储文件信息。

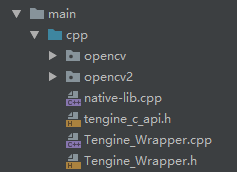
1. 在将一些文件通过adb传送至手机上以后，就可以将工程进行下载到手机上了，结果总是好的：



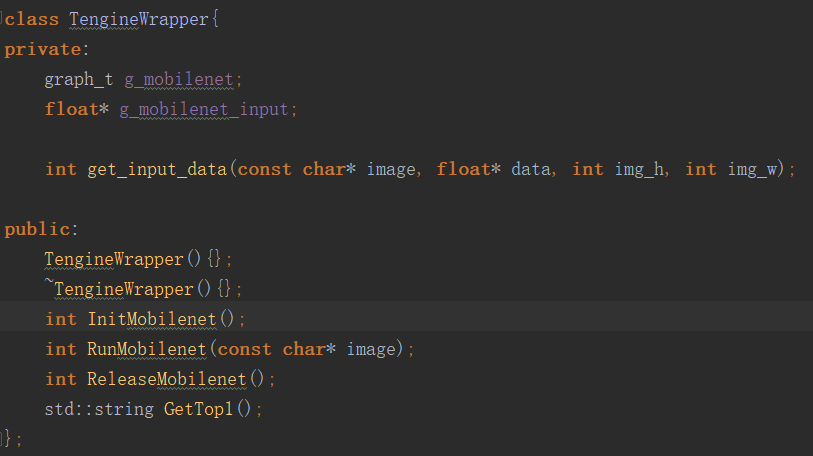
成功以后来说说在程序上是怎么实现的吧。首先是MainActivity.java当中有这么几行：



其实这就是为什么我们的手机app界面只有那一行小小的字啦，代码里面就是只描述了这行字的变化。还需要看两个文件，分别是Tengine\_Wrapper.cpp和Tengine\_Wrapper.h文件：

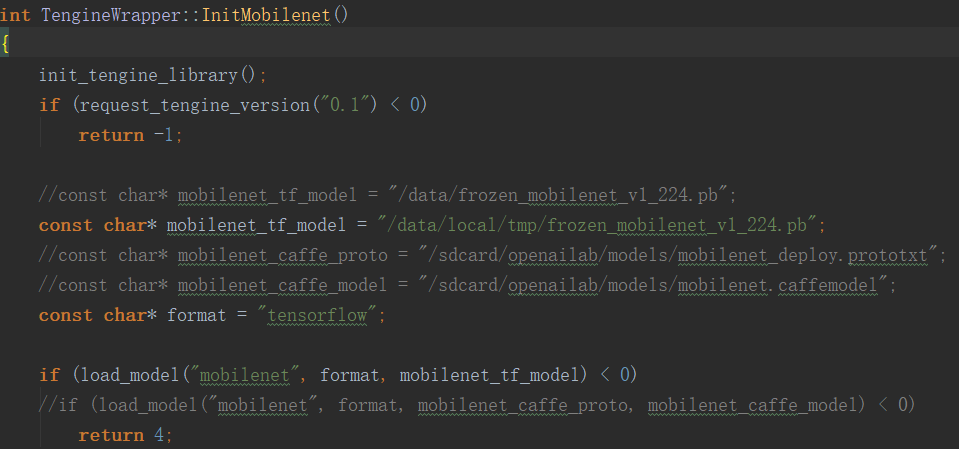


h文件里面其实就是TengineWrapper这个class的描述

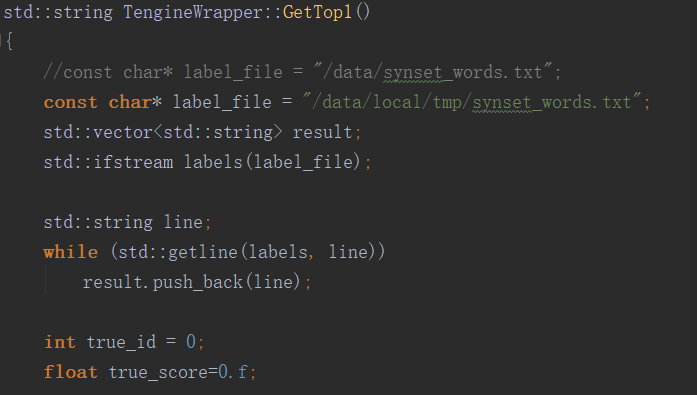


cpp文件就是各个函数的描述，这里不做一一介绍，但是比较重要的几行是：

这里标明了model加载的地址，就是之前我们用adb传输到手机上的文件



这里有加载lable文件的地址：



1. 其他问题
2. 运行成灰色的问题，就一开始运行程序的时候发现运行的标志成灰色，而且那个倒三角的标志也是空的，但是这个问题在之后就突然消失了，又变成了正常可以运行的情形：



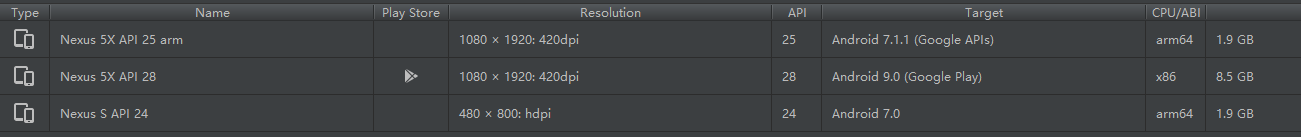
1. cpu架构是arm64-v8a

这个问题我其实也说不清楚，但是意思就是对于不同型号的手机他们cpu不同，所以会有这种问题：

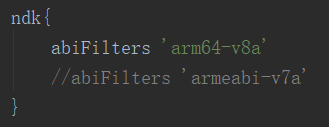


这个时候可以有几种办法：

【1】重新选择新的API模拟器，不过不知为什么这些模拟器往往都是启动不了？？？可能是因为性能太低



【2】在工程文件当中哪里会有这个cpu/api架构的设置呢？是在app文件夹下面的build.gradle文件里面有这么一个东西：



这里也顺便说说NDK与SDK两者的区别：

SDK: Software Development Kit 软件开发工具包

通常使用java语言开发，他包含了SDK Manager 和 AVD Manage对于android系统的一些开发版本的管理以及模拟器管理

NDK: Native Development Kit

通常使用C/C++开发，当我们要操作底层或者内存的时候，NDK是必须的。

<https://blog.csdn.net/cc20032706/article/details/53739600>

