

从今天开始，会陆续发表HEVC学习过程中的一些总结，主要目的是与大家一起交流学习，由于水平有限，恳请大家能够不吝赐教。

由于这是第一次写博客，而且学习也是个循序渐进的过程，第一篇文章就简单介绍下如何使用HEVC的官方标准测试模型HM吧。其实如果之前学习过H.264的话，肯定也是用过JM的，HM的使用方法与JM基本上是相同的，只是配置文件的内容有所不同罢了。好了，不多说，马上进入正题。

首先自然是先把这个测试模型下载下来，链接地址如下：https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/，这个网址下有各个版本的HM，到目前为止，最新版本是HM 9.0，可以根据自己的实际需求进行选择。这里需要注意的是，下载时需要使用SVN，如windows下的TortoiseSVN。

其次，最好附带下载HM的reference software manual即参考软件手册，这个手册的主要内容包括HM的下载地址和使用方法，更重要的是，里面包含了对配置文件各个参数的详细解释，可以说是使用HM的必备手册。

最后，就是真正地编译HM工程了。从运行环境来讲，既可以在linux系统下使用makefile对HM进行编译，也可以在windows系统下使用IDE如VS对HM进行编译。我个人是用VS2008作为开发环境，因此，这里也只介绍使用VS编译HM。关于linux下如何进行，请参看前面提到的参看软件手册吧 😊。

打开下载下来的HM文件夹，工程存放在build目录下，工程名为HM_VC9(对VS2008而言)。右击“Solution 'HM_VC9'(7 projects)”->“build solution”。正常的话，等待片刻所有工程都能得到正确的编译，并最终在HM文件夹下生成bin目录，在目录xxx\bin\vc9\Win32\Debug下，最终会生成两个可执行文件，TAppEncoder.exe和TAppDecoder.exe，分别问编码和解码的可执行文件，到了这一步，表明你之前的步骤都是正确的。接下来就能进入到编解码器的使用当中来了。

先说编码器的使用：右击“TAppEncoder”->“Set as Startup Project”，接着，再次右击“TAppEncoder”->“Properties”->“Configuration Properties”->“Debugging”，在弹出的右窗口中有这么两行需要关注：“Command Arguments”和“Working Directory”。前者用于输入运行时的命令参数，后者用于指定工程的工作目录。先说工作目录，将其设置为xxx\bin\vc9\Win32\Debug，即上一步中编译生成的可执行文件的绝对路径。接下来，为了方便起见，我们可以把需要使用到的配置文件、yuv测试序列复制到该目录下。与JM有所不同的是，我们需要至少使用两个配置文件，即在HM文件夹的cfg目录中的某一个配置文件如encoder_intra_main.cfg，以及per-sequence目录下的某一个配置文件如BasketballDrill.cfg。前者主要用于配置编码器的各种编码参数，后者主要用于指定输入yuv测试文件，分辨率，待编码帧数等等。在完成了上述步骤之后，我们就可以在“Command Arguments”一栏填入-c encoder_intra_main.cfg -c BasketballDrill.cfg，表明这个编码器使用这两个配置文件所指定的参数进行编码。

此外，值得一提的是，使用配置文件不是必须的，我们也可以使用纯命令行的输入方式对编码器的各个参数进行指定，具体使用方法，请参考软件手册。在一些特定情况下如只需修改其中的少数参数值，这种纯命令行方式还是比较方便的，但是，这种做法容易漏掉一些参数值的设置（尽管它们有默认值），而且对于经常修改多个参数来说，远不如直接修改配置文件来得方便和准确。因此，个人推荐使用上一种方法进行设置。

在进行如上配置后，正常情况下编码器就能将你指定的yuv测试文件编码为二进制码流文件了。

再说解码器的使用：在到填入“Command Arguments”之前，跟编码器是完全一样的，这里就不再重复了，输入的命令如下-b xxx.bin -o deco.yuv，-b指定输入的二进制码流，即编码输出码流文件，-o指定输出的yuv文件。设置完毕之后，运行解码器，应该能够解码出deco.yuv文件来了。

至此，简单的HM使用介绍完毕，欢迎大家对该文考虑不周和错误的地方进行补充和指定。

原文地址：http://blog.csdn.net/hevc_cjl/article/details/8169182

文章标签：[hevc](#) [hm](#) [cfg](#) [yuv](#)

个人分类：[HM](#) [视频编码](#)