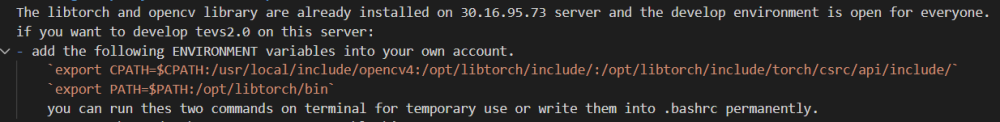
# VSCode工作空间环境配置

## 0. 准备

当前使用的代码版本：d549b9dda3bfa366fad87ddbf2a473e2f2fa8dc0

代码放到服务器后，按照README.md上，配置好libtorch和opencv的环境（如果您的机器已经成功运行过此项目，则表明已经配置成功）。



记得修改完后使用

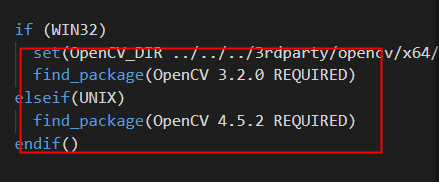
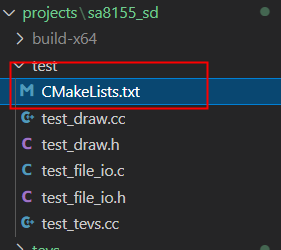
source ~/.bashrc

使配置生效。

**该版本下需要修改的地方如下**

## 1. 修改project/sa8155\_sd/test/CMakeLists.txt

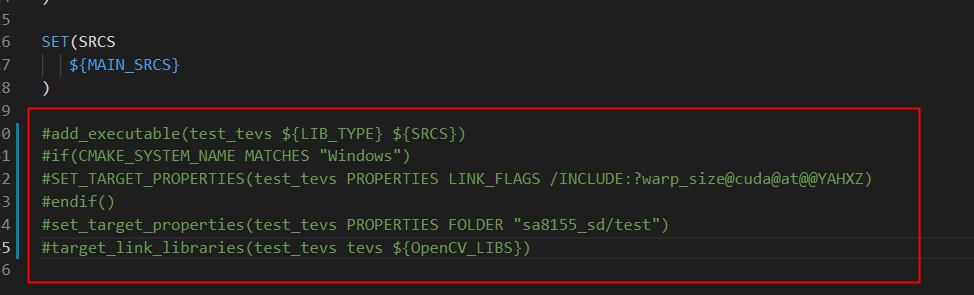
由于服务器的opencv版本有点问题，需要修改project/sa8155\_sd/test/CMakeLists.txt。将下图这个文件的CMakeLists.txt里进行修改。



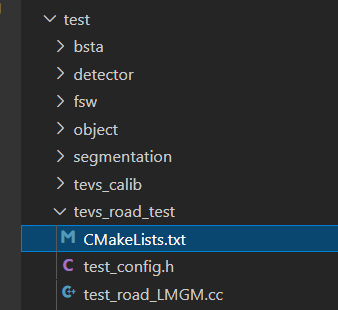
修改的内容是：把3.2.0 和4.5.2这个版本号去掉，修改后这一段代码如下：

if (WIN32)
set(OpenCV\_DIR ../../../3rdparty/opencv/x64/x64/vc14/lib)
find\_package(OpenCV REQUIRED)
elseif(UNIX)
find\_package(OpenCV REQUIRED)
endif()

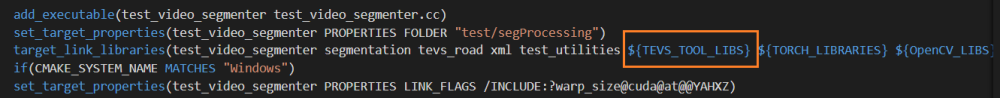
然后在此文件中注释掉下图CMakeLists.txt中的以下部分



## 2. 修改test/tevs\_road\_test/中CMakeLists.txt



a. 在下图中添加 ${TEVS\_TOOL\_LIBS} （约第15行）



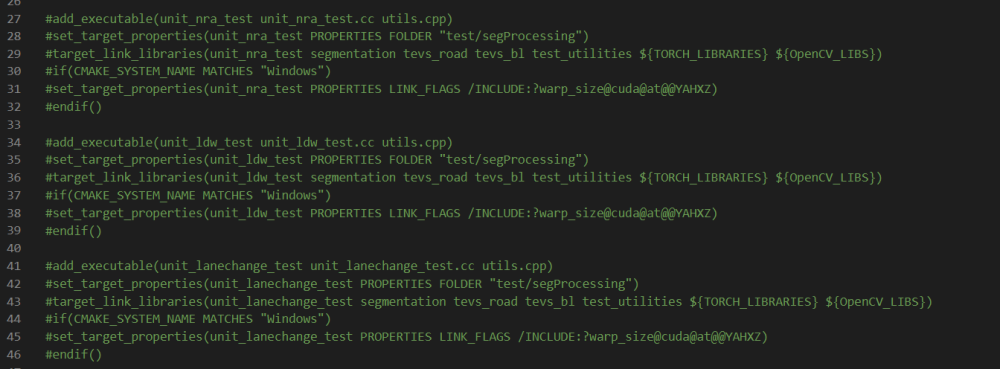
修改前：

target\_link\_libraries(test\_video\_segmenter segmentation tevs\_road test\_utilities ${TORCH\_LIBRARIES} ${OpenCV\_LIBS})

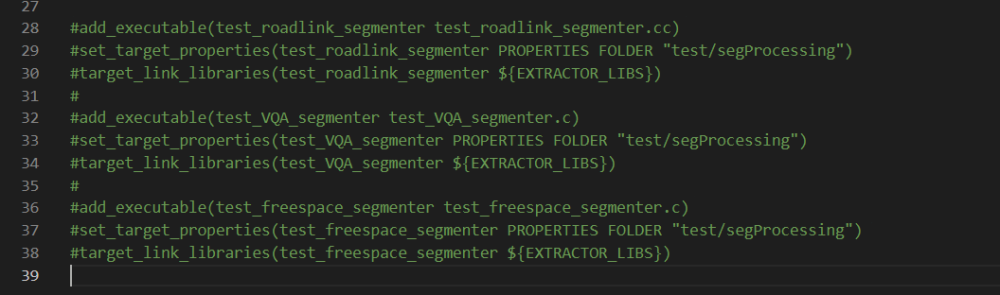
修改后：

target\_link\_libraries(test\_video\_segmenter segmentation tevs\_road test\_utilities ${TEVS\_TOOL\_LIBS} ${TORCH\_LIBRARIES} ${OpenCV\_LIBS})

b．在此文件中屏蔽下图内容

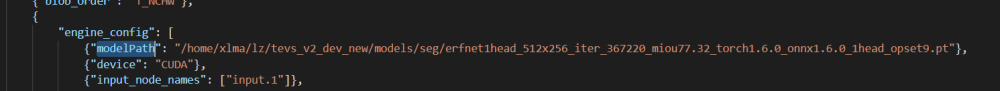


并屏蔽掉下面的内容：



## 3. 修改segmentaiotn\_libtorch.json

将工程路径下 cfgs/segmentaiotn\_libtorch.json 中第30行的“modelPath”值更新为此工程中 models/seg/路径下的 erfnet... opset9.pt:



然后将此文件第6行"{"model\_input\_size": [**712**, 256]}, " 修改为 "{"model\_input\_size": [**512**, 256]}, "。

## 4. 文件准备

a. 在data路径下，上传 vs\_calib\_results.txt，请从组内同学出获取此文件。 b. 在data路径下上传待处理的avi视频文件，例如 ch1\_20180101\_001603\_0011.avi，此文件请从组内同学处获取。 c. 在合适的路径（如 test/tevs\_road\_test/ 路径下）下新建 data 文件夹，然后在data文件夹中新建 1.txt，其中内容如下：

（test/tevs\_road\_test/data/1.txt **content**）

/home/path/to/your\_tevs\_project/cfgs/segmentation\_libtorch.json
/home/path/to/your\_tevs\_project/test/tevs\_road\_test/data/vs\_calib\_results.txt
/home/path/to/your\_tevs\_project/test/tevs\_road\_test/data/ch1\_20180101\_001603\_0011.avi

文件中这3行中的“/home/path/to/your\_tevs\_project/”均需要改成您的工程文件路径，后两行的文件路径就是前面a,b 两个步骤涉及的文件。

## 5. 配置vscode工作空间：

在项目根路径下创建 .vscode文件夹（注意，此文件夹名以”.”开头），然后在.vscode中新建 launch.json文件，并修改内容如下：

{
// Use IntelliSense to learn about possible attributes.
// Hover to view descriptions of existing attributes.
// For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
"version": "0.2.0",
"configurations": [
{
"name": "(gdb) Launch",
"type": "cppdbg",
"request": "launch",
"program": "/home/path/to/your\_tevs\_project/projects/sa8155\_sd/tevs/bin/x64\_linux/Debug/test\_video\_segmenter",
"args": ["/home/path/to/your\_tevs\_project/test/tevs\_road\_test/data/1.txt"],
// "args": [],
"stopAtEntry": false,
"cwd": "${fileDirname}",
"environment": [],
"externalConsole": false,
"MIMode": "gdb",
"setupCommands": [
{
"description": "Enable pretty-printing for gdb",
"text": "-enable-pretty-printing",
"ignoreFailures": true
}
],
"preLaunchTask": "make",
"miDebuggerPath": "/usr/bin/gdb"
}
]
}

launch.json文件修改的内容是"program"和"args"这两个键的值，注意需要将这两个键的值中的"/home/path/to/your\_tevs\_project/"修改为您的tevs工程路径，建议使用绝对路径；其中 program 表示可执行文件的位置，args 表示函数中的传参，此路径中的1.txt就是上一步骤（5. 文件准备-c）中新建的文件。

## 6. 创建build目录

在项目的根目录下，创建 build文件夹，并在终端切换至此目录下：

mkdir build
cd build

在终端执行：

cmake -DCMAKE\_BUILD\_TYPE=Debug ../projects/sa8155\_sd

等待cmake完成，然后继续在build路径下执行：

make

或者执行：make –j8。

## 7. 代码更正

**a**. 将 test/tevs\_road\_test/tevs\_video\_segmenter.cc中VSStatus RunTevsInVideo(const char\* input\_data) 函数（约254行处）中 uint8\_t b\_show\_seg = VS\_FALSE 更改为 uint8\_t b\_show\_seg = VS\_TRUE

**b**. 屏蔽（注释）掉test/tevs\_road\_test/tevs\_video\_segmenter.cc 中所有cv操作中关于 namedWindow、imshow、waitKey、destroyWindow等的语句。

**c**. 在大约470 行处 if (b\_show\_freespace) 之后，插入如下语句：cv::imwrite("/home/path/to/your\_tevs\_project/test/tevs\_road\_test/data/test\_seg\_result.jpg",cv\_image\_show);

**d**. 将原文件中的 char sz\_frm\_info[100]; 这一句删除，并将其插入至410行： int frameId = startFrameId + f; 前面或者后面。

**e**. 在test/bsta/unit\_test.cc 中约第44行处 插入此语句： int gdw\_absta3[1000];

## 8. 后续操作

后续修改代码后，只需要在build路径下执行

make

再在VSCode中按 F5（或Ctrl + F5） 调试即可。