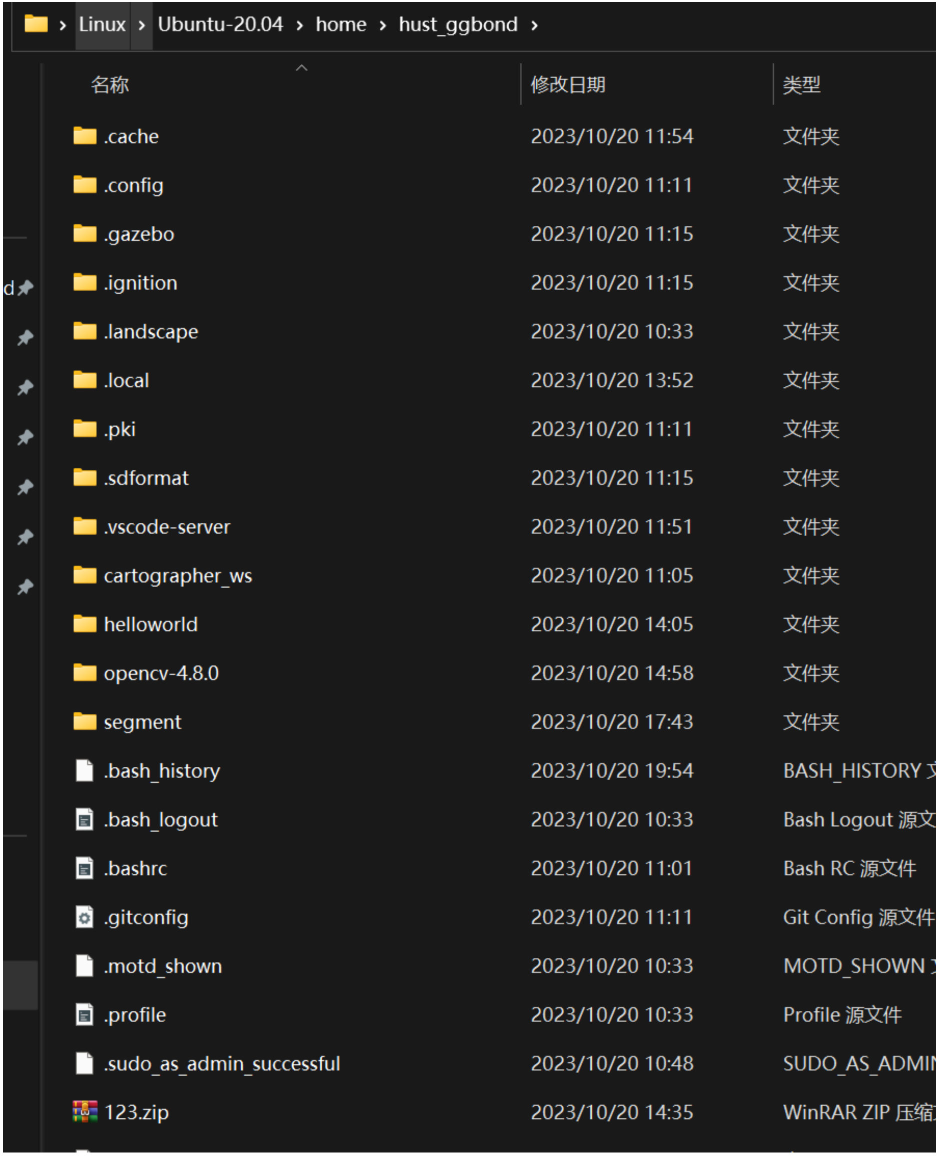
cmak学习提交文档

任务一：安装linux系统

由于双系统安装后一直出现键盘无法连接的问题，所以最后安装了WSL2 的Linux子系统，版本为Ubuntu 20.04.6LTS,同时安装了ros noetic版本和相应功能包



› Linux › Ubuntu-20.04 › home › hust_ggbond ›

名称	修改日期	类型
.cache	2023/10/20 11:54	文件夹
.config	2023/10/20 11:11	文件夹
.gazebo	2023/10/20 11:15	文件夹
.ignition	2023/10/20 11:15	文件夹
.landscape	2023/10/20 10:33	文件夹
.local	2023/10/20 13:52	文件夹
.pki	2023/10/20 11:11	文件夹
.sdformat	2023/10/20 11:15	文件夹
.vscode-server	2023/10/20 11:51	文件夹
cartographer_ws	2023/10/20 11:05	文件夹
helloworld	2023/10/20 14:05	文件夹
opencv-4.8.0	2023/10/20 14:58	文件夹
segment	2023/10/20 17:43	文件夹
.bash_history	2023/10/20 19:54	BASH_HISTORY 文
.bash_logout	2023/10/20 10:33	Bash Logout 源文
.bashrc	2023/10/20 11:01	Bash RC 源文件
.gitconfig	2023/10/20 11:11	Git Config 源文件
.motd_shown	2023/10/20 10:33	MOTD_SHOWN 文
.profile	2023/10/20 10:33	Profile 源文件
.sudo_as_admin_successful	2023/10/20 10:48	SUDO_AS_ADMI
123.zip	2023/10/20 14:35	WinRAR ZIP 压缩

任务三，用GCC编译hello world代码

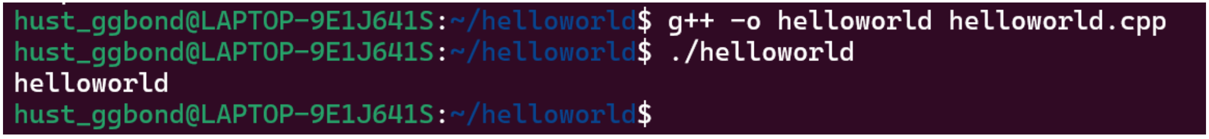
以下是我用来测试编译的helloworld文件代码

```
#include<stdio.h>
#include<iostream>
using namespace std;
int main(){

cout<<"helloworld"<<endl;

return 0;

}
```

进行gcc编译时输入的指令

```
g++ -o helloworld helloworld.cpp
```

输出结果如图



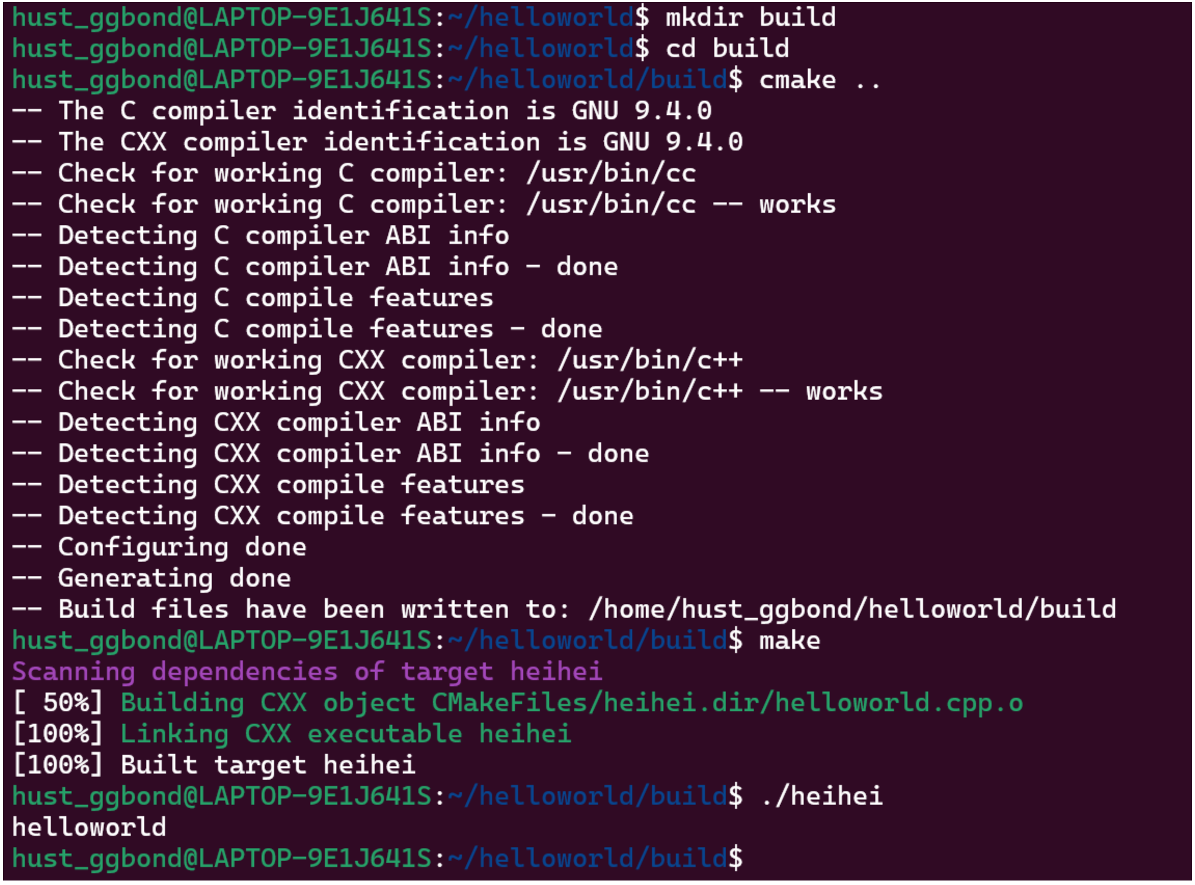
```
hust_ggbond@LAPTOP-9E1J641S:~/helloworld$ g++ -o helloworld helloworld.cpp
hust_ggbond@LAPTOP-9E1J641S:~/helloworld$ ./helloworld
helloworld
hust_ggbond@LAPTOP-9E1J641S:~/helloworld$
```

任务四 cmake的简单使用

以下是我编写的CMakeLists.txt源码

```
cmake_minimum_required(VERSION 3.16.3)
project(haha)
add_executable(heihei helloworld.cpp)
set(CMAKE_CXX_STANDARD 11)
```

以下是实现结果



```
hust_ggbond@LAPTOP-9E1J641S:~/helloworld$ mkdir build
hust_ggbond@LAPTOP-9E1J641S:~/helloworld$ cd build
hust_ggbond@LAPTOP-9E1J641S:~/helloworld/build$ cmake ..
-- The C compiler identification is GNU 9.4.0
-- The CXX compiler identification is GNU 9.4.0
-- Check for working C compiler: /usr/bin/cc
-- Check for working C compiler: /usr/bin/cc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Configuring done
-- Generating done
-- Build files have been written to: /home/hust_ggbond/helloworld/build
hust_ggbond@LAPTOP-9E1J641S:~/helloworld/build$ make
Scanning dependencies of target heihei
[ 50%] Building CXX object CMakeFiles/heihei.dir/helloworld.cpp.o
[100%] Linking CXX executable heihei
[100%] Built target heihei
hust_ggbond@LAPTOP-9E1J641S:~/helloworld/build$ ./heihei
helloworld
hust_ggbond@LAPTOP-9E1J641S:~/helloworld/build$
```

任务六 cmake实践

以下是我补充编写的CMakeLists.txt文件源码

```
cmake_minimum_required(VERSION 2.8)

project(segmentation)

set(CMAKE_CXX_STANDARD 11)


find_package(OpenCV 4 REQUIRED)

include_directories(${CMAKE_CURRENT_SOURCE_DIR}/include)

include_directories(${OPENCV_INCLUDE_DIRS})

set(SOURCE_FILES src/main.cpp src/segment.cpp )


add_executable(segementation ${SOURCE_FILES})

target_link_libraries(segementation ${OpenCV_LIBS})
```

此外，还更改了main文件中关于图片的地址。