

从 VMware 安装 Ubuntu 22.04 到 RM 视觉项目环境配置培训手册

版本：V1.0

编写日期：2025-08-08

编写人：张俊杰

适用对象：26 赛季视觉组新人

目录

1. 培训目标.....	3
2. 系统与软件要求.....	3
3. VMware 虚拟机安装与配置	3
4. Ubuntu 系统安装.....	12
5. 系统基础配置	13
6. 视觉项目开发环境搭建	13
7. 项目代码拉取与运行测试	27

1. 培训目标

- 熟悉 VMware 虚拟机安装与 Ubuntu 系统配置流程
- 掌握视觉项目所需环境的安装步骤
- 能够独立完成代码拉取与运行测试

2. 系统与软件要求

项目	版本/要求
VMware	17.x 及以上
Ubuntu	22.04 LTS
Python	3.8+
CUDA（可选）	11.8 / 12.x

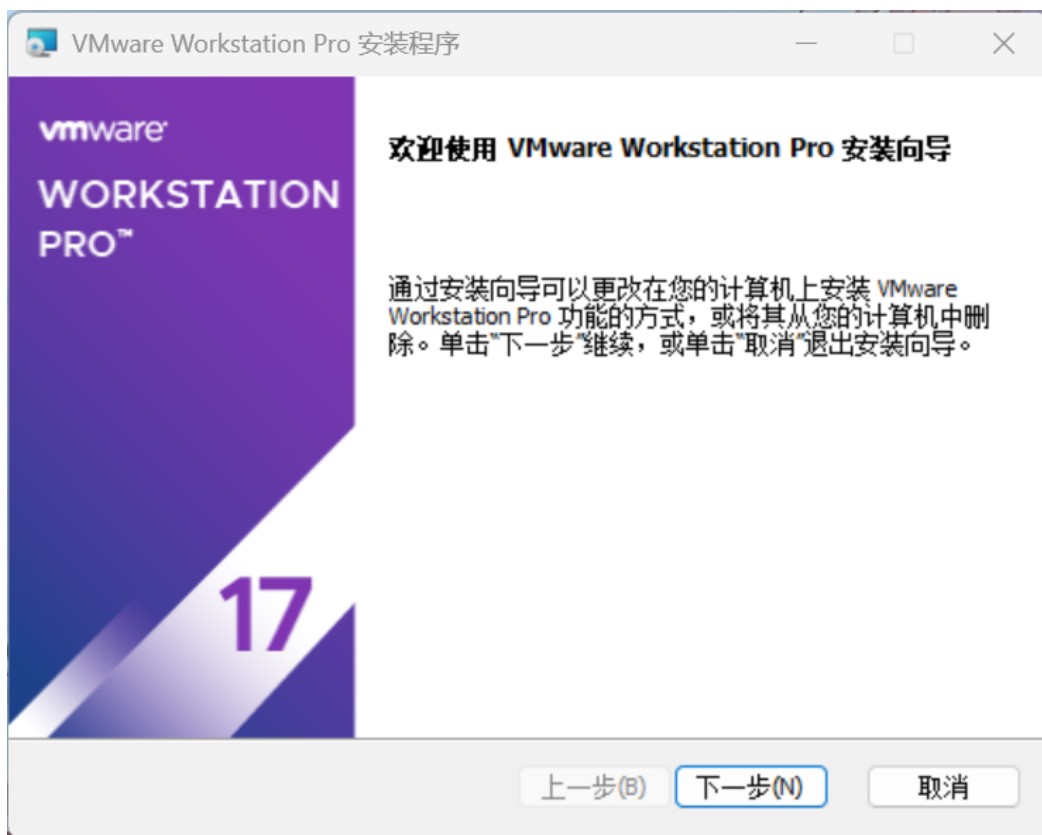
3. VMware 虚拟机安装与配置

1. 下载 VMware Workstation 并安装（Windows 平台）
- 官方地址：<https://www.vmware.com>（下面所有的安装包均会提供）

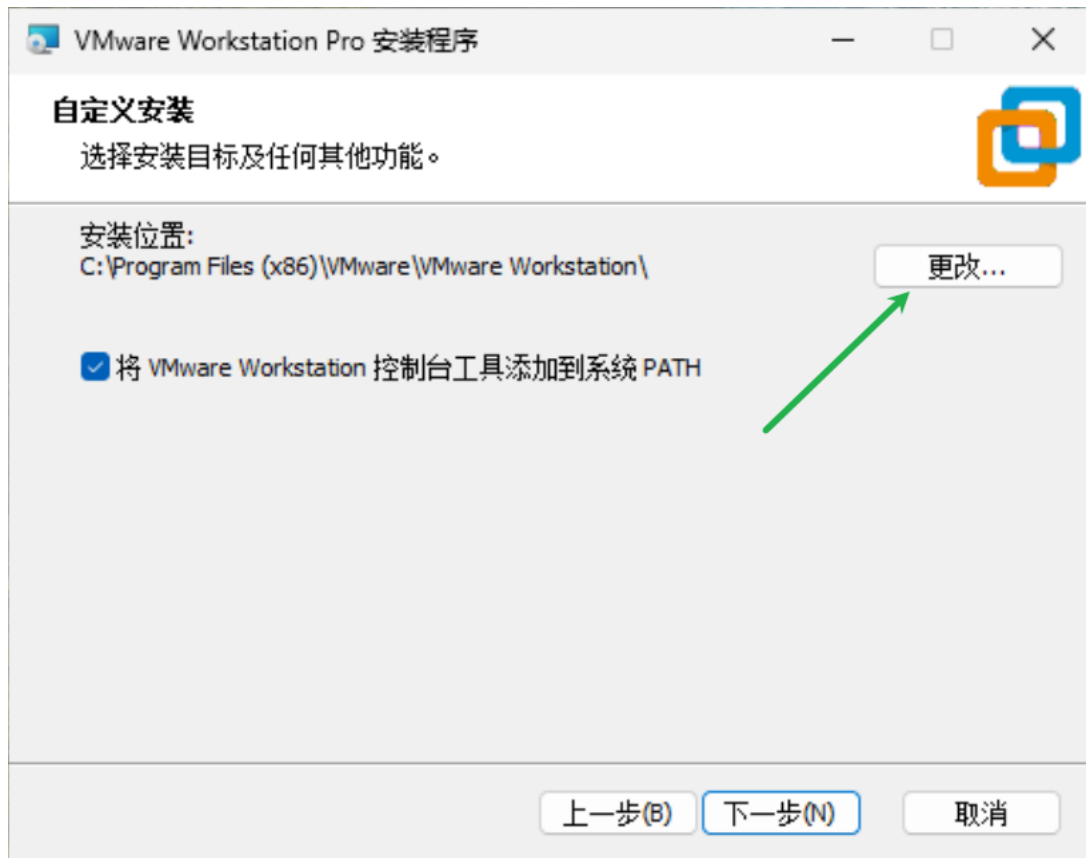
 VMware17.6.exe	2024/9/3 17:39	应用程序	251,368 KB
--	----------------	------	------------

这里只需要管理员运行，然后跟着引导安装即可

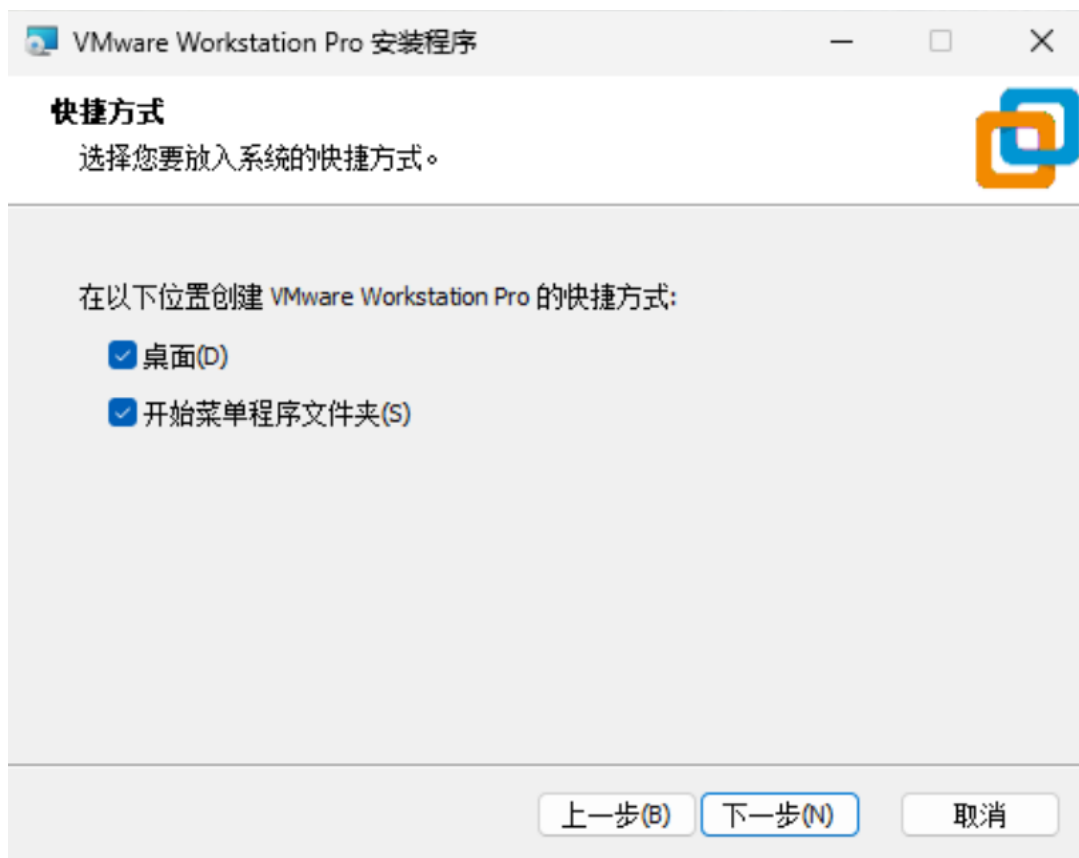
这里只需按下一步



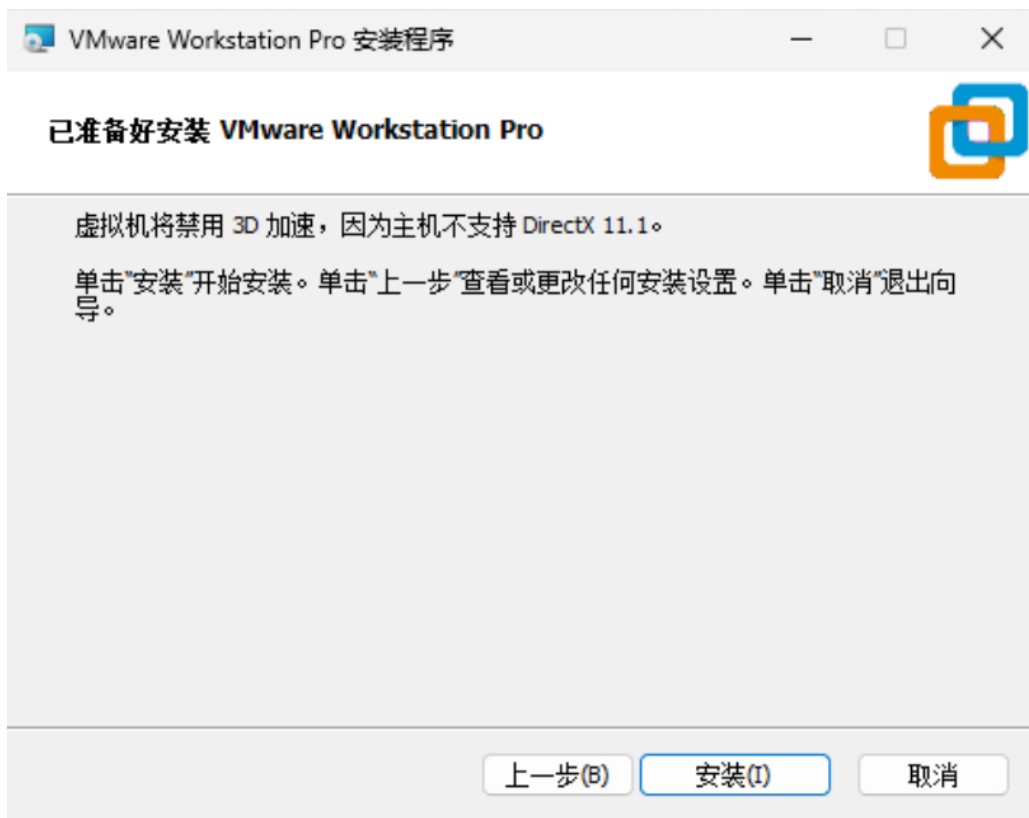
这里请自行更改你安装的位置，一般放在 D 等非 C 盘，然后点击下一步



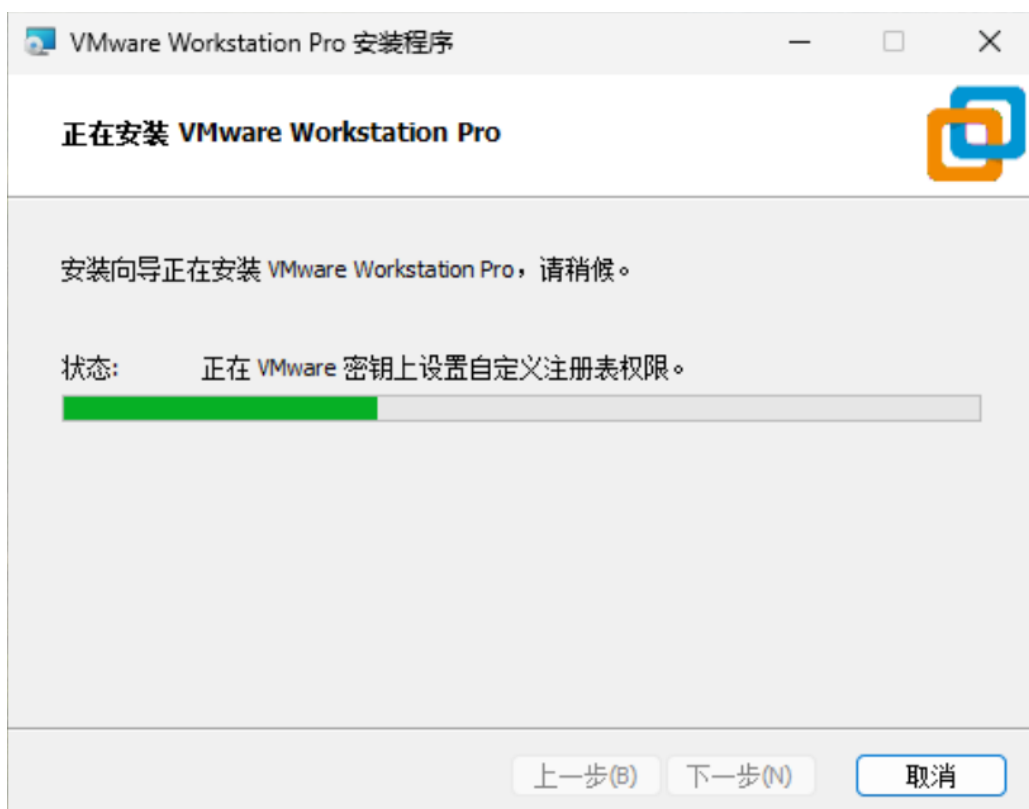


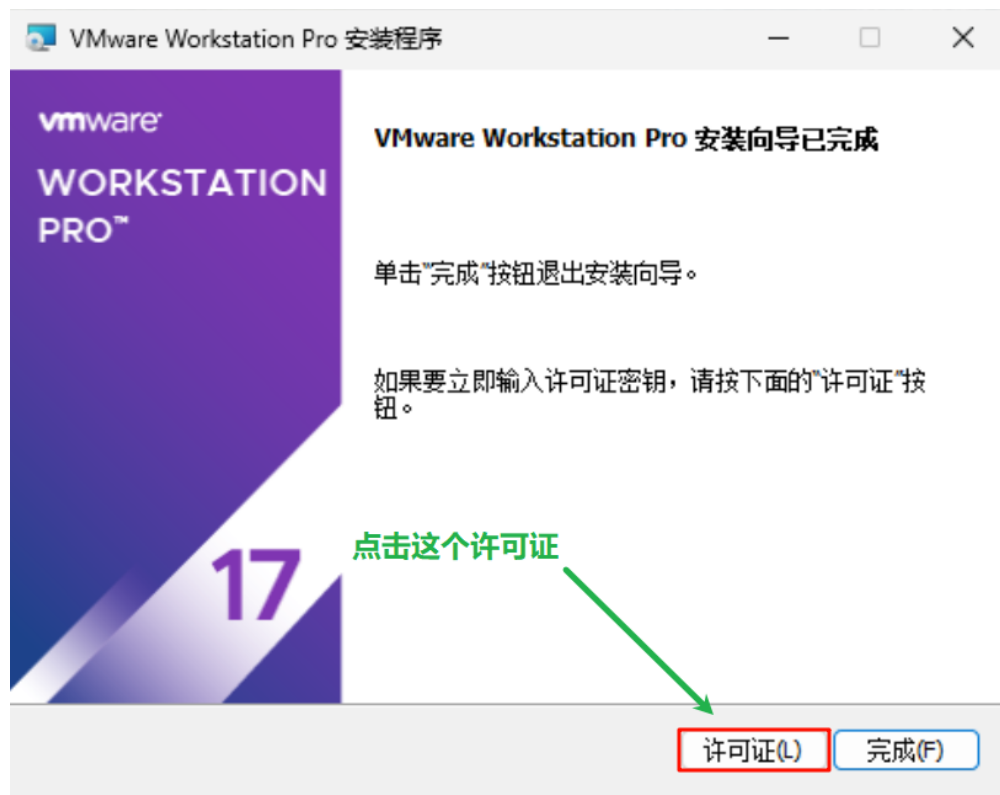


这里默认设置即可，直接点击下一步

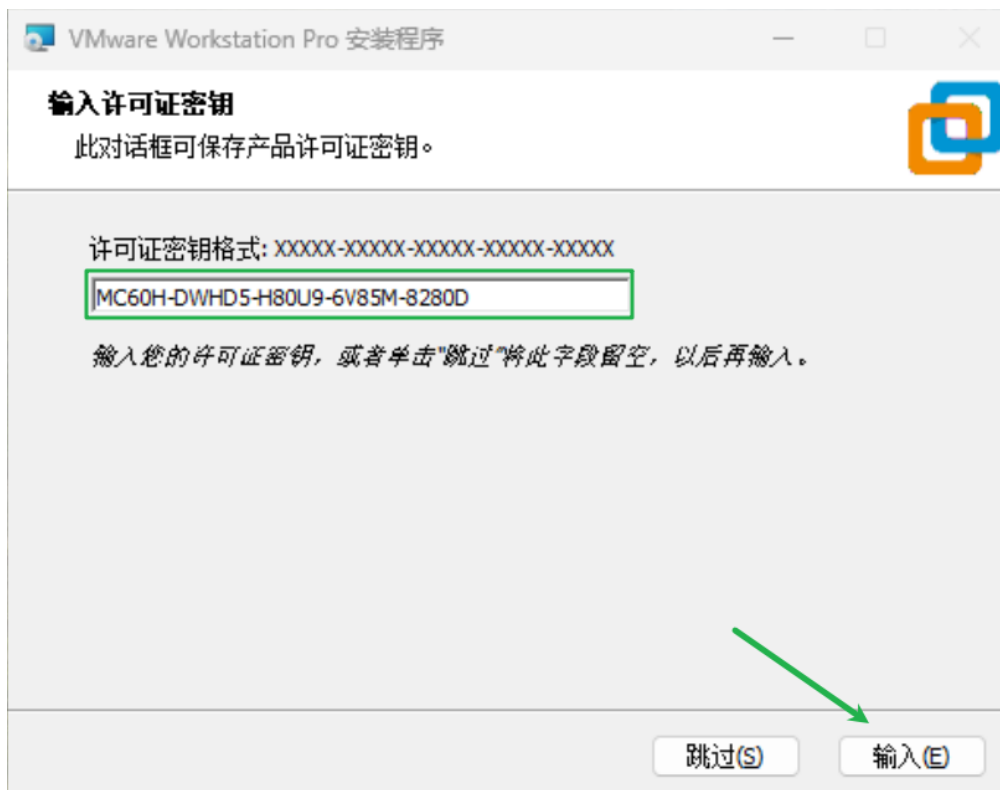


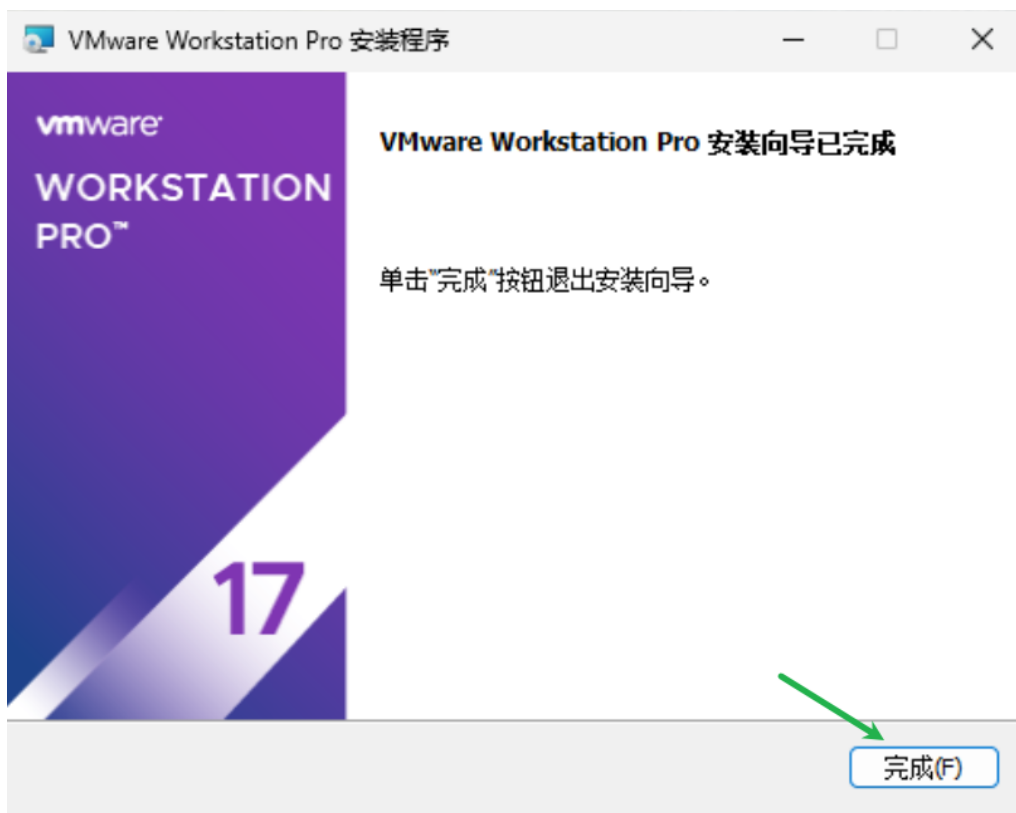
静待片刻



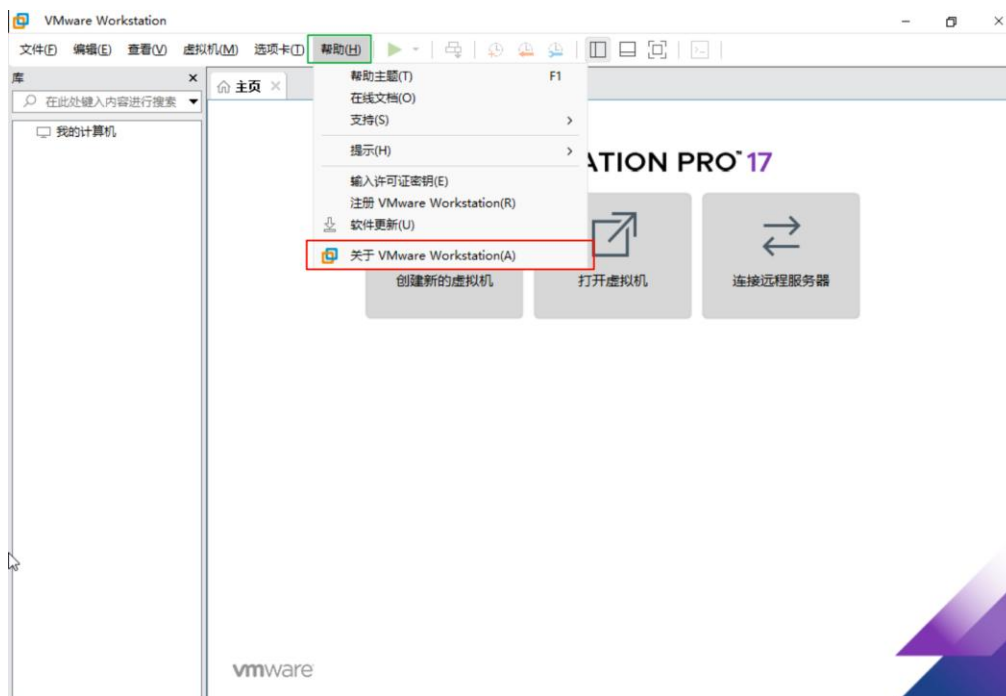


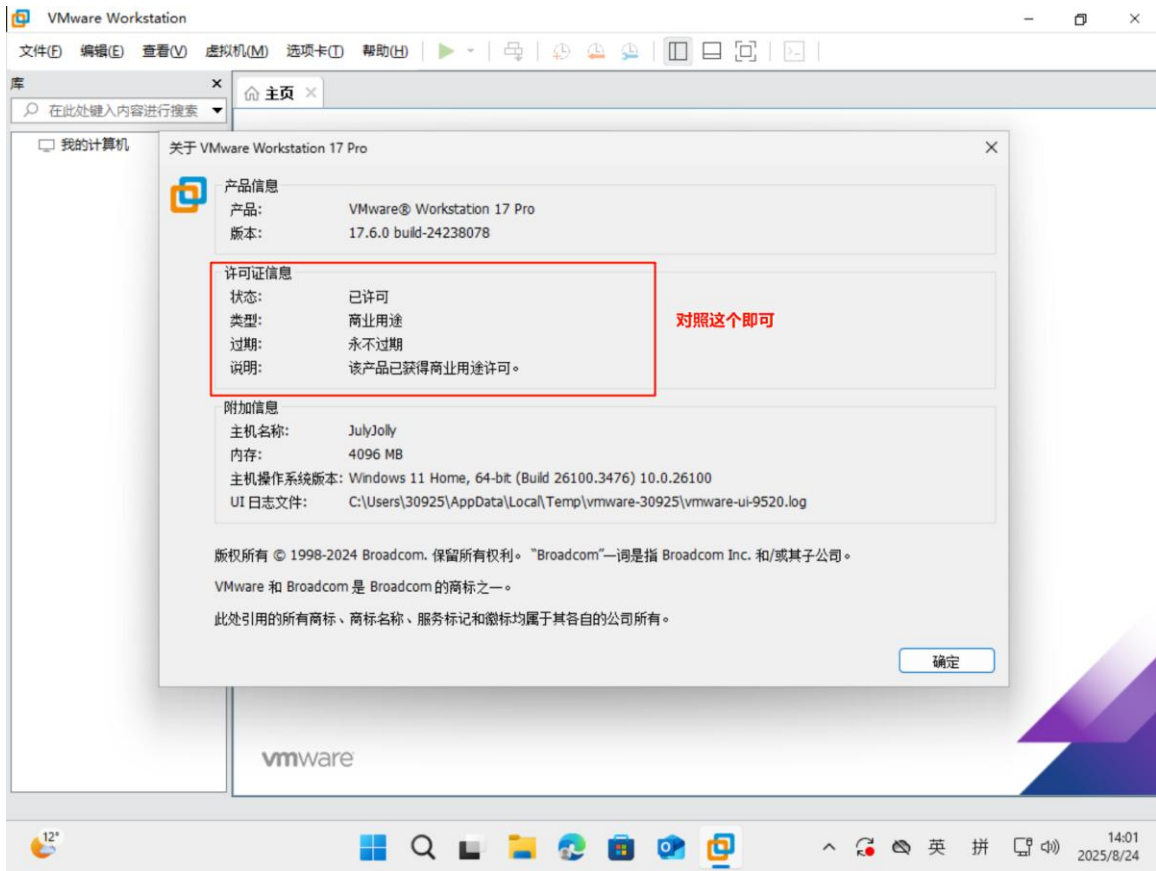
输入 MC60H-DWHD5-H80U9-6V85M-8280D





安装完成右键打开 VMware，点击帮助，然后点击关于 VMware Workstation





出现这个表示 vmware workstation 安装完成了！

2. 创建 Ubuntu 虚拟机

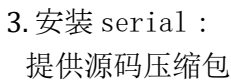
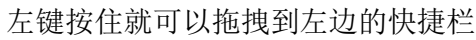
- 分配 CPU/内存（推荐 2 核/8GB）
- 网络建议 NAT 或桥接

3. 网络配置注意事项：公司内网桥接模式可能需要申请 IP


```
Activities Terminal 8月8 15:39 julijuly@julijuly: ~  
[~][0.88s] CMD Result:success 1091|80... connected.orce.py  
欢迎使用一键安装ros和ros2,支持树莓派jetson,本工具由作者小鱼提供  
欢迎使用ros2开发工具,其工具由(角青ros)小鱼提供  
小鱼:如需安装ros2,请运行:catkin_ws --name ros2 --rosdistro foxy --source ros2 --source ros2 --source ros2 --source ros2 --source ros2  
=====按下 任意 键 安装并清理三方源, 如果不知道选什么,请选择=====  
Run CMD Task[sudo apt install curl -y]  
[~][4.92s] CMD Result:success  
Run CMD Task[sudo apt search group2 ]  
[~][0.47s] CMD Result:success  
Run CMD Task[sudo apt install group2 -y]  
[~][4.21s] CMD Result:success  
正在查找最佳的密钥服务: [https://github.com/rosdistro/ros/master/ros.asc', 'https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc']  
[https://github.com/rosdistro/ros/master/ros.asc 耗时:0.41s  
[https://raw.githubusercontent.com/rosdistro/ros/master/ros.asc 耗时:0.56s  
已自动选择最佳密钥服务:https://github.com/rosdistro/ros/master/ros.asc  
Run CMD Task[curl -s https://github.com/rosdistro/ros/master/ros.asc | sudo apt-key add -]  
[~][0.47s] CMD Result:success  
Run CMD Task[sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys F42ED0FB817C54]  
[~][11.58s] CMD Result:success  
Run CMD Task[curl -s https://github.com/rosdistro/ros/master/ros.asc | sudo gpg --no-default-keyring --keyring gnupg-ring:/etc/apt/trusted.gpg.d/ros.gpg --import]  
[~][0.42s] CMD Result:success  
Run CMD Task[sudo chmod o+r /etc/apt/trusted.gpg.d/ros.gpg]  
[~][0.00s] CMD Result:success  
Run CMD Task[dpkg --get-architecture]  
[~][0.00s] CMD Result:success  
根据您的系统,为您推荐安装源为: [http://mirrors.tuna.tsinghua.edu.cn/ros2/ubuntu/]  
创建文件:/etc/apt/sources.list.d/ros-fish.list  
Run CMD Task[sudo apt update]  
[~][8.76s] CMD Result:success  
Run CMD Task[sudo apt search ros-base ]  
[~][0.58s] CMD Result:success  
恭喜,成功安装ros2,接下来可以使用apt安装ros或者使用[1]-键安装ros安装!  
Run CMD Task[sudo apt search ros-base ]  
[~][0.59s] CMD Result:success  
Run Choose Task: [请输入括号内的数字]  
请选择您要安装的ros版本名称(请注意ros1和ros2区别):  
[1]:shumble(ros2)  
[2]:tram(ros2)  
[3]:rolling(ros2)  
[4]:none  
请输入[1]内的数字以选择:  
[~][0.54s] CMD Result:success  
Run CMD Task[sudo apt install aptitude -y]  
[~][11.11s] Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu [arm64] amd64 libaptget amd64 1.1.0-1ubuntu1 [arm 68 kB]
```

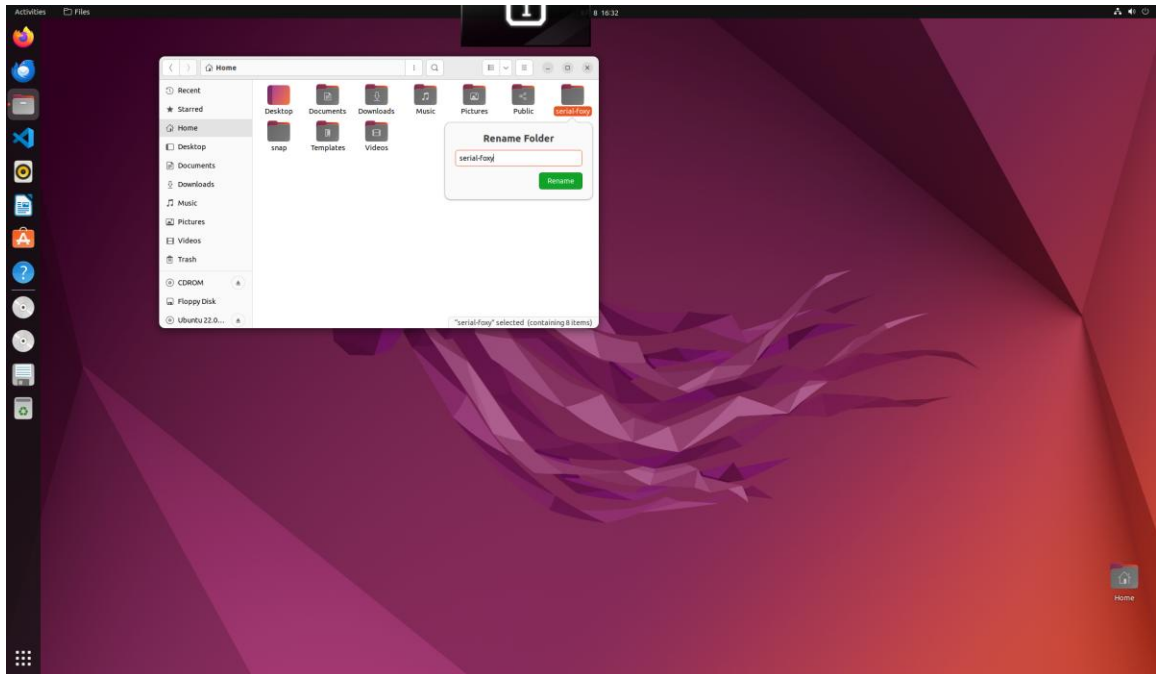
选择桌面版即可, 这里大概等个 5~6 分钟 (与你给 Ubuntu 的虚拟内存有关)

```
Activities Terminal 8月8 15:40 julijuly@julijuly: ~  
[~][0.88s] CMD Result:success 1091|80... connected.orce.py  
欢迎使用一键安装ros和ros2,支持树莓派jetson,本工具由作者小鱼提供  
欢迎使用ros2开发工具,其工具由(角青ros)小鱼提供  
小鱼:如需安装ros2,请运行:catkin_ws --name ros2 --rosdistro foxy --source ros2 --source ros2 --source ros2 --source ros2 --source ros2  
=====按下 任意 键 安装并清理三方源, 如果不知道选什么,请选择=====  
Run CMD Task[sudo apt install curl -y]  
[~][4.92s] CMD Result:success  
Run CMD Task[sudo apt search group2 ]  
[~][0.47s] CMD Result:success  
Run CMD Task[sudo apt install group2 -y]  
[~][4.21s] CMD Result:success  
正在查找最佳的密钥服务: [https://github.com/rosdistro/ros/master/ros.asc', 'https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc']  
[https://github.com/rosdistro/ros/master/ros.asc 耗时:0.41s  
[https://raw.githubusercontent.com/rosdistro/ros/master/ros.asc 耗时:0.56s  
已自动选择最佳密钥服务:https://github.com/rosdistro/ros/master/ros.asc  
Run CMD Task[curl -s https://github.com/rosdistro/ros/master/ros.asc | sudo apt-key add -]  
[~][0.47s] CMD Result:success  
Run CMD Task[sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys F42ED0FB817C54]  
[~][11.58s] CMD Result:success  
Run CMD Task[curl -s https://github.com/rosdistro/ros/master/ros.asc | sudo gpg --no-default-keyring --keyring gnupg-ring:/etc/apt/trusted.gpg.d/ros.gpg --import]  
[~][0.42s] CMD Result:success  
Run CMD Task[sudo chmod o+r /etc/apt/trusted.gpg.d/ros.gpg]  
[~][0.00s] CMD Result:success  
Run CMD Task[dpkg --get-architecture]  
[~][0.00s] CMD Result:success  
根据您的系统,为您推荐安装源为: [http://mirrors.tuna.tsinghua.edu.cn/ros2/ubuntu/]  
创建文件:/etc/apt/sources.list.d/ros-fish.list  
Run CMD Task[sudo apt update]  
[~][8.76s] CMD Result:success  
Run CMD Task[sudo apt search ros-base ]  
[~][0.58s] CMD Result:success  
恭喜,成功安装ros2,接下来可以使用apt安装ros或者使用[1]-键安装ros安装!  
Run CMD Task[sudo apt search ros-base ]  
[~][0.59s] CMD Result:success  
Run Choose Task: [请输入括号内的数字]  
请选择您要安装的ros版本名称(请注意ros1和ros2区别):  
[1]:shumble(ros2)  
[2]:tram(ros2)  
[3]:rolling(ros2)  
[4]:none  
请输入[1]内的数字以选择:  
[~][0.54s] CMD Result:success  
Run CMD Task[sudo apt install aptitude -y]  
[~][11.11s] Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu [arm64] amd64 libaptget amd64 1.1.0-1ubuntu1 [arm 68 kB]
```

305 KB

将其解压到 home 目录下，重命名为 serial



双击进去里面有个 **readme.pdf** 像这样，安装他的引导，打开终端输入下列命令即可

Get the code:

```
# open a new terminal
cd serial
mkdir build
```

Build:

```
cd build
cmake ..
make
```

Install:

```
sudo make install
```

这里就是编译这里的源码，然后把库载入系统

```
Activities Terminal 8 16:35 ohyr -serial/build

julyjelly@julyjelly:~/serial$ mkdir build && cd build
julyjelly@julyjelly:~/serial/build$ cmake ..
-- The C compiler identification is GNU 11.4.0
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working C compiler: /usr/bin/cc - skipped
-- Detecting C compile features
-- Detecting C compile features - done
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++ - skipped
-- Detecting CXX compile features
-- Detecting CXX compile features - done
-- Found amem_cmake: 1.3.12 (/opt/ros/humble/share/amem_cmake/cmake)
-- Found python3: /usr/bin/python3 (found version "3.10.12") Found components: Interpreter
-- Configuring done
-- Generating done
-- Build files have been written to: /home/julyjelly/serial/build
julyjelly@julyjelly:~/serial/build$ make
[ 25%] Building CXX object CMakeFiles/serial.dir/src/serial.cc.o
[ 50%] Building CXX object CMakeFiles/serial.dir/src/impl/uart.cc.o
[ 75%] Building CXX object CMakeFiles/serial.dir/src/impl/uart_ports/uart_ports_linux.cc.o
[100%] Linking CXX shared library libserial.so
[100%] Built target serial
julyjelly@julyjelly:~/serial/build$ sudo make install
[sudo] password for julyjelly:
Install the project:
-- Install configuration: --
-- Installing: /usr/local/share/serial/environment/library_path.sh
-- Installing: /usr/local/share/serial/environment/library_path.dsv
-- Installing: /usr/local/lib/libserial.so
-- Installing: /usr/local/include/serial/serial.h
-- Installing: /usr/local/share/amem_index/resource_index/package_run_dependencies/serial
-- Installing: /usr/local/share/serial/environment/amem_prefix_path.sh
-- Installing: /usr/local/share/serial/environment/amem_prefix_path/serial
-- Installing: /usr/local/share/serial/environment/amem_prefix_path.dsv
-- Installing: /usr/local/share/serial/environment/path.sh
-- Installing: /usr/local/share/serial/environment/path.dsv
-- Installing: /usr/local/share/serial/local_setup.bash
-- Installing: /usr/local/share/serial/local_setup.sh
-- Installing: /usr/local/share/serial/local_setup.dsv
-- Installing: /usr/local/share/serial/package.dsv
-- Installing: /usr/local/share/amem_index/resource_index/packages/serial
-- Installing: /usr/local/share/serial/cmake/amem_cmake_export_include_directories-extras.cmake
-- Installing: /usr/local/share/serial/cmake/amem_cmake_export_libraries-extras.cmake
-- Installing: /usr/local/share/serial/cmake/serialConfig.cmake
-- Installing: /usr/local/share/serial/cmake/serialConfig-version.cmake
-- Installing: /usr/local/share/serial/package.xml
julyjelly@julyjelly:~/serial/build$
```

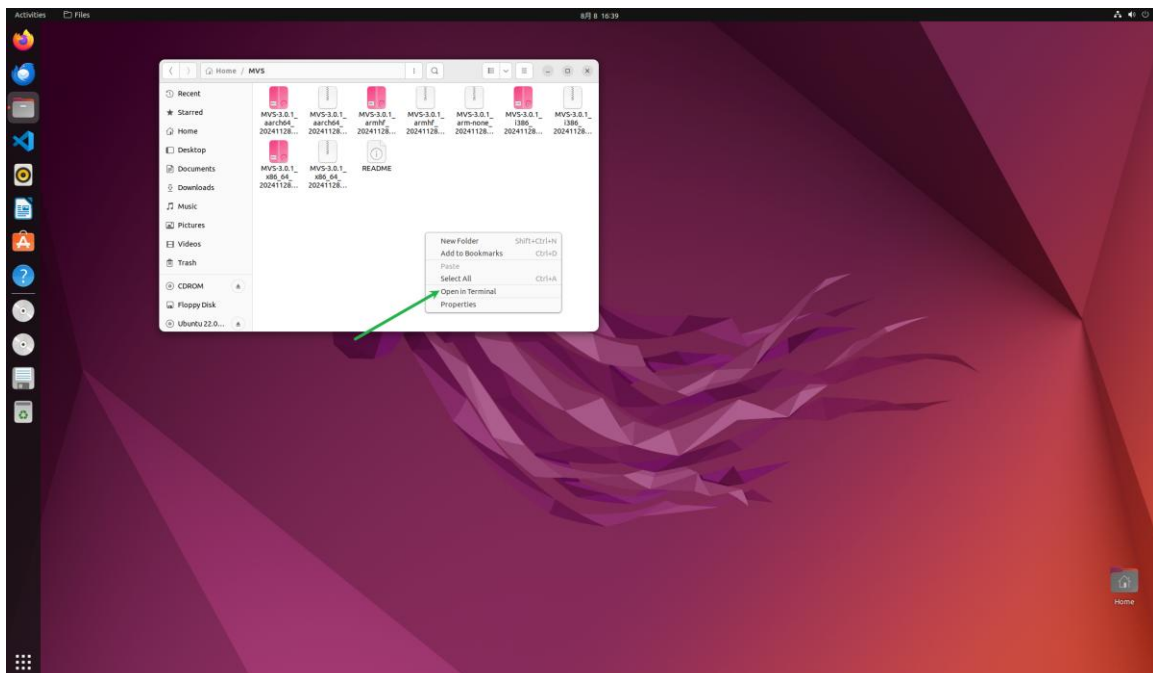
4. 安装 海康工业相机驱动 MVS :

提供驱动压缩包, 同样解压到 home 目录

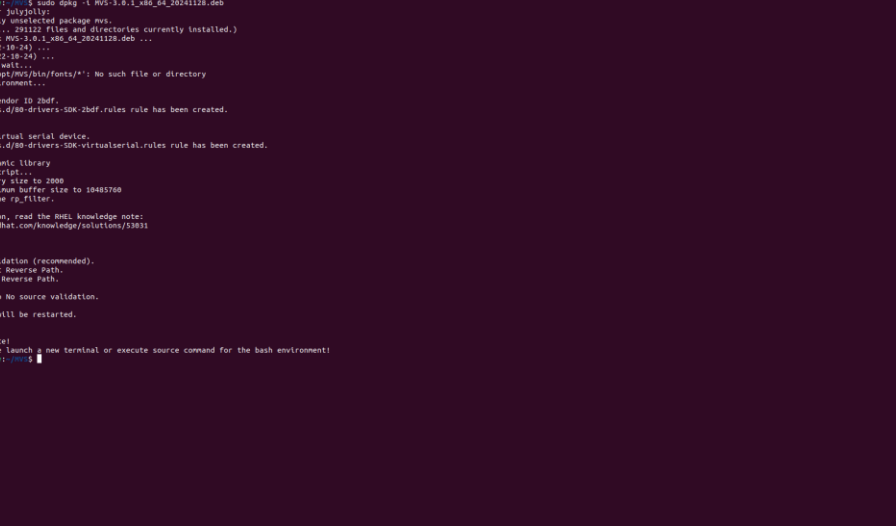
 MVS.zip 2025/8/8 16:24 压缩(zipped)文件夹 545,160 KB

解压后双击打开, 在 MVS 目录下右键在终端打开输入如下命令:

`sudo dpkg -i MVS-3.0.1_x86_64_20241128.deb`



如下图，重启即可生效。



```
julyjolly@julyjolly: ~$ sudo dpkg -i MVS-3.0.1
MVS-3.0.1_sarch64_20241128.deb  MVS-3.0.1_i386_20241128.deb
MVS-3.0.1_armhf_20241128.deb  MVS-3.0.1_arm_20241128.deb
julyjolly@julyjolly: ~$ sudo dpkg -i MVS-3.0.1_i386_20241128.deb
[sudo] password for julyjolly:
Selecting previously unselected package mvs.
(Reading database ... 291122 files and directories currently installed.)
Preparing to unpack MVS-3.0.1_i386_20241128.deb ...
Unpacking mvs (2022-10-24) ...
Setting up mvs (2022-10-24) ...
Install MVS, please wait...
cp: cannot stat '/opt/mvs/bin/fonts/*': No such file or directory
Set up the SDK environment...
Adding rules for vendor ID 2bdf.
The /etc/udev/rules.d/80-drivers-SDK-2bdf.rules rule has been created.
Adding rules for virtual serial device.
The /etc/udev/rules.d/80-drivers-SDK-virtualserial.rules rule has been created.
Create link to dynamic library
Starting execute script...
Setting usb's memory size to 2000
Setting socket maximum buffer size to 10485760
Configuration of the rz-filter.
For more information, read the README knowledge note:
https://access.redhat.com/knowledge/solutions/53031

Supported modes:
0 - No source validation (recommended).
1 - RFC3704 Strict Reverse Path.
2 - RFC3704 Loose Reverse Path.

Setting the mode to No source validation.
The network stack will be restarted.

Install MVS complete!
Tips: You should be launch a new terminal or execute source command for the bash environment!
julyjolly@julyjolly: ~$
```

5.安装自动求导库 Ceres :

在终端输入：



```
sudo apt update
```

```
sudo apt install libceres-dev
```

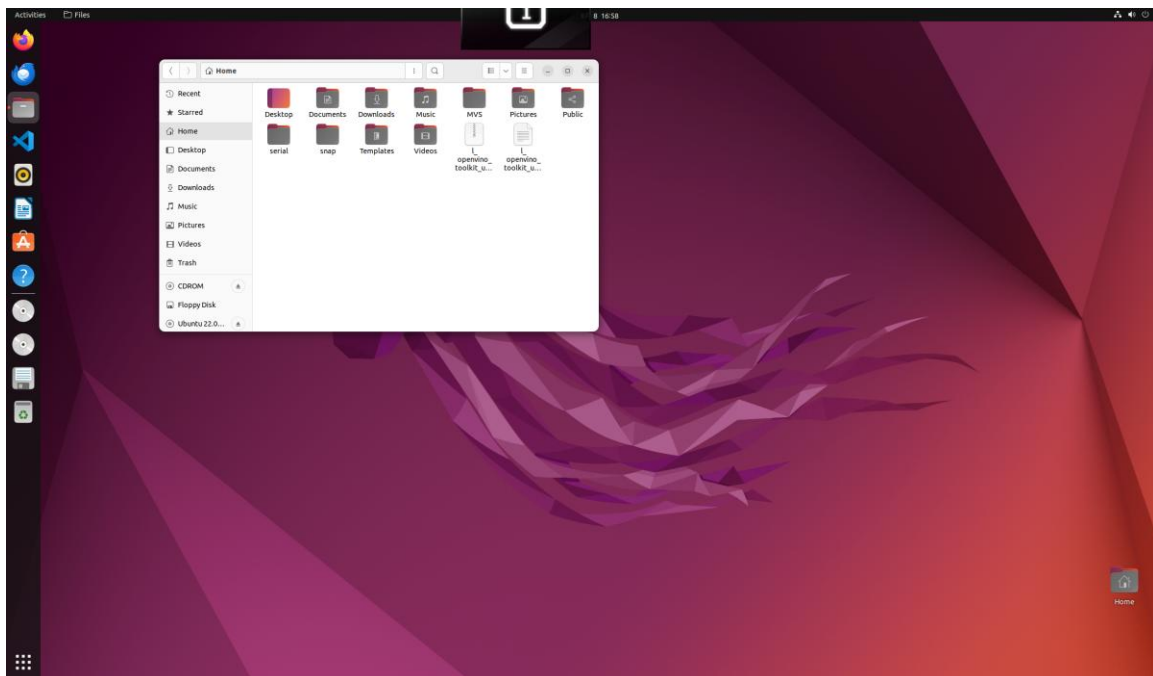
[illegible]

6. 安装 cpu 加速 openvino2023 :

提供关键 openvino2023 库压缩包及校验包

 I_openvino_toolkit_ubuntu22_2023.3.0.137...	2025/4/6 11:52	压缩存档文件夹	52,420 KB
 I_openvino_toolkit_ubuntu22_2023.3.0.137...	2025/4/6 11:51	SHA256 文件	1 KB

拷贝到 home 目录下，打开终端

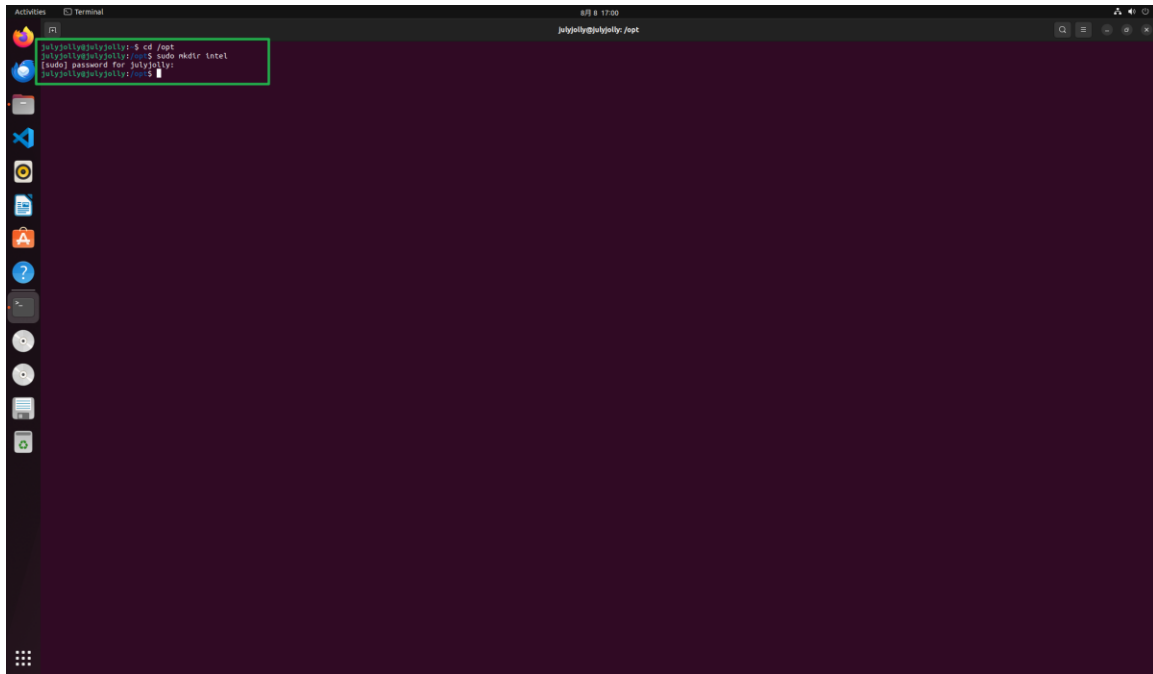


输入

```
cd /opt
```

```
sudo mkdir intel
```

创建存放 openvino 库的文件夹



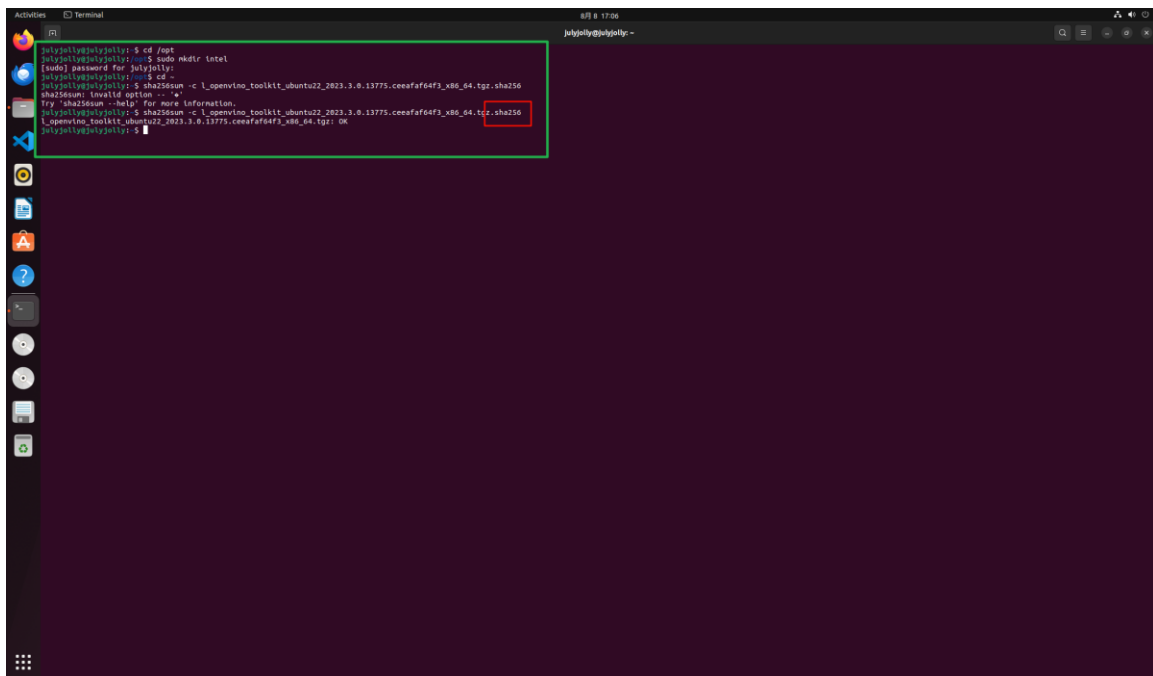
```
julyjuly@julyjuly:~$ cd /opt
julyjuly@julyjuly:~/opt$ sudo mkdir intel
[sudo] password for julyjuly:
julyjuly@julyjuly:~/opt$
```

输入: `cd ~` 返回 home 目录

输入: `sha256sum -c`

`l_openvino_toolkit_ubuntu22_2023.3.0.13775.cceafaf64f3_x86_64.tgz.sha256`

这个命令用来校验库，第一次复制进去可能会打印报错日志，第二遍手敲 `sha256sum -c l_op` 在 `tab` 补齐即可，补齐别忘了 `.sha256`



```
julyjuly@julyjuly:~$ cd /opt
julyjuly@julyjuly:~/opt$ sudo mkdir intel
[sudo] password for julyjuly:
julyjuly@julyjuly:~/opt$ cd ~
julyjuly@julyjuly:~$ sha256sum -c l_openvino_toolkit_ubuntu22_2023.3.0.13775.cceafaf64f3_x86_64.tgz.sha256
sha256sum: l_openvino_toolkit_ubuntu22_2023.3.0.13775.cceafaf64f3_x86_64.tgz: OK
julyjuly@julyjuly:~$
```

输入

```
sudo tar xf l_openvino_toolkit_ubuntu22_2023.3.0.13775.cceafaf64f3_x86_64.tgz -C /opt/intel
```

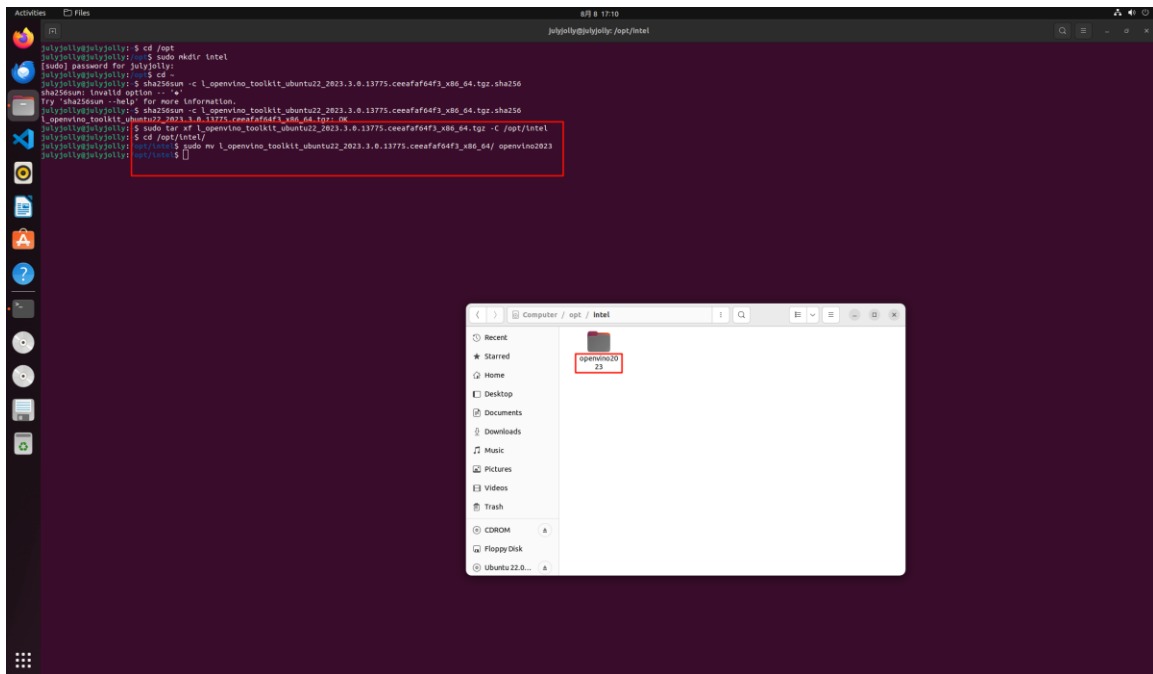
将 openvino 库解压到 intel 文件夹

然后，输入

```
cd /opt/intel/
```

```
sudo mv l_openvino_toolkit_ubuntu22_2023.3.0.13775.cceafaf64f3_x86_64/ openvino2023
```

将 intel 的 openvino 改名为 openvino2023



输入 `cd ~` 回到 home 目录，接着依次输入下面命令行安装 openvino APT

```
wget https://apt.repos.intel.com/intel-gpg-keys/GPG-PUB-KEY-INTEL-SW-PRODUCTS.PUB
```

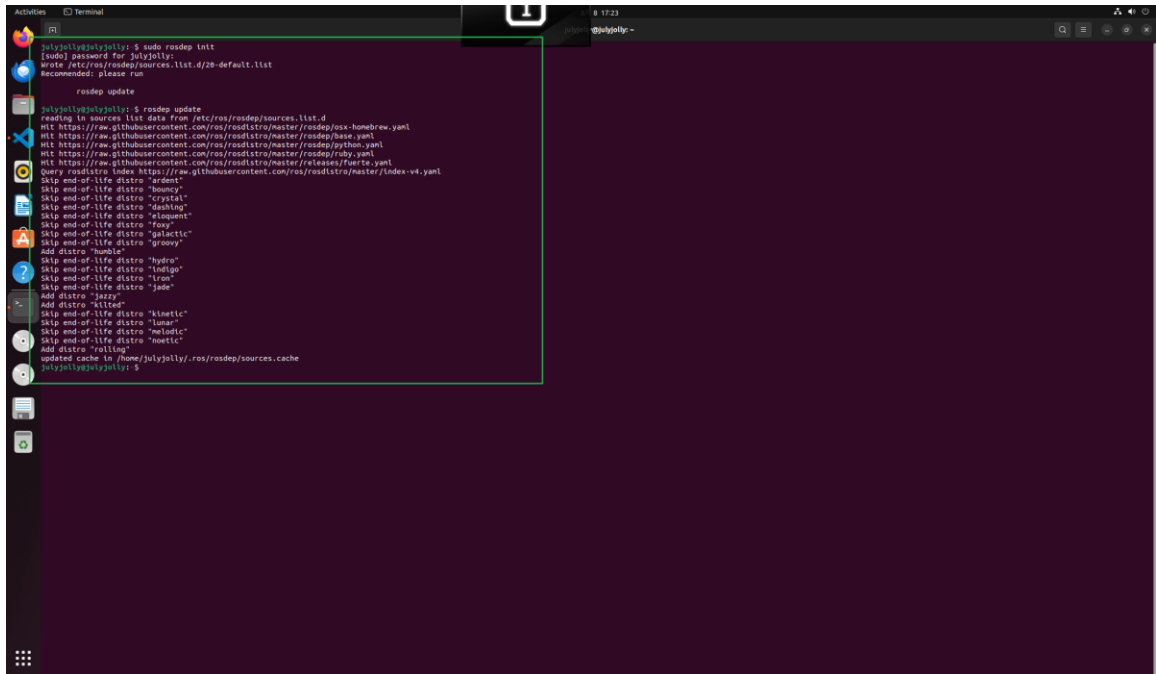
```
sudo apt-key add GPG-PUB-KEY-INTEL-SW-PRODUCTS.PUB
```

```
echo "deb https://apt.repos.intel.com/opencvino/2023 ubuntu22 main" | sudo tee /etc/apt/sources.list.d/intel-openvino-2023.list
```

```
sudo apt update
```

```
apt-cache search openvino
```

```
sudo apt install openvino-2023.1.0
```

```
julyjolly@julyjolly:~$ sudo rosdep init
[sudo] password for julyjolly:
wrote /etc/ros/rosdep/sources.list.d/20-default.list
Recommended: please run

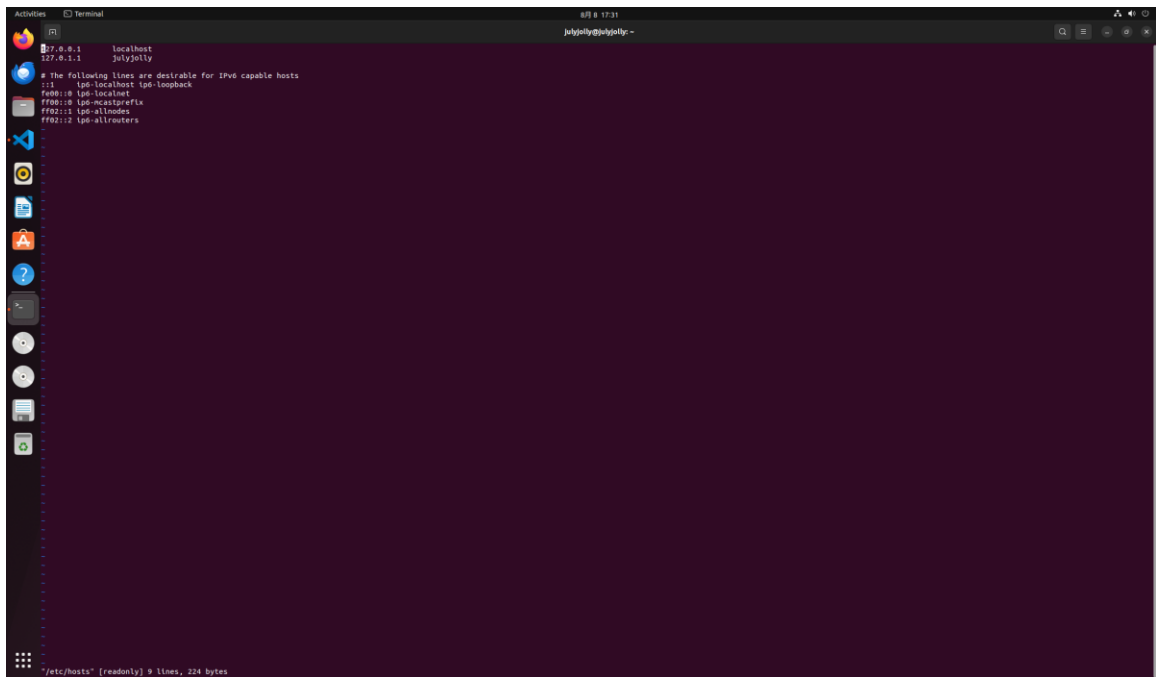
  rosdep update

julyjolly@julyjolly:~$ rosdep update
reading in sources list data from /etc/ros/rosdep/sources.list.d
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/osx-homebrew.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/base.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/patches.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/rosdep/ruby.yaml
Hit https://raw.githubusercontent.com/ros/rosdistro/master/releases/fuente.yaml
Query rosdistro index https://raw.githubusercontent.com/ros/rosdistro/master/index-v4.yaml
Skip end-of-life distro "ardent"
Skip end-of-life distro "bouncy"
Skip end-of-life distro "crystal"
Skip end-of-life distro "dashing"
Skip end-of-life distro "eloquent"
Skip end-of-life distro "foxy"
Skip end-of-life distro "galactic"
Skip end-of-life distro "groovy"
Add distro "humble"
Skip end-of-life distro "hydro"
Skip end-of-life distro "indigo"
Skip end-of-life distro "iron"
Skip end-of-life distro "jade"
Add distro "jazzy"
Add distro "kinetic"
Skip end-of-life distro "kinetic"
Skip end-of-life distro "lunar"
Skip end-of-life distro "melodic"
Skip end-of-life distro "noetic"
Add distro "rolling"
update cache in /home/julyjolly/.ros/rosdep/sources.cache
julyjolly@julyjolly:~$
```

我这里运气好一次就通过了，如果没通过也别着急在 `/etc/hosts` 文件中添加相应的 ip 即可，

输入 `vi /etc/hosts`

进去之后就是这样的



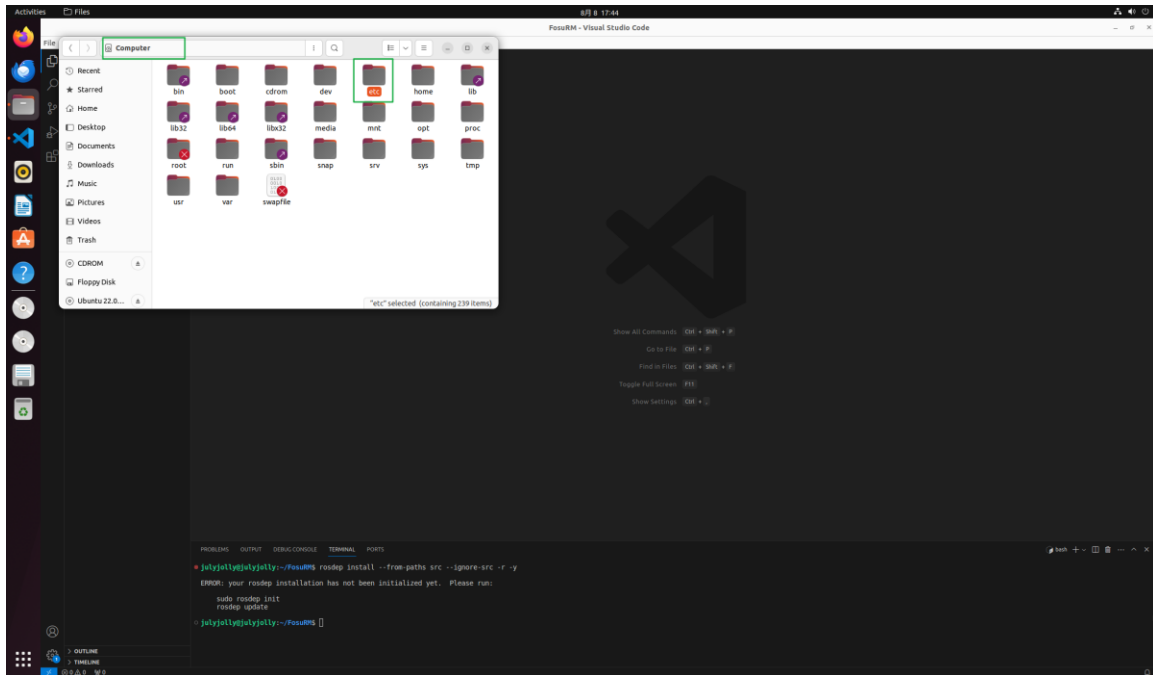
```
127.0.0.1    localhost
127.0.0.1    julyjolly

# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe80::     ip6-localhost
ff02::     ip6-mcastprefix
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters

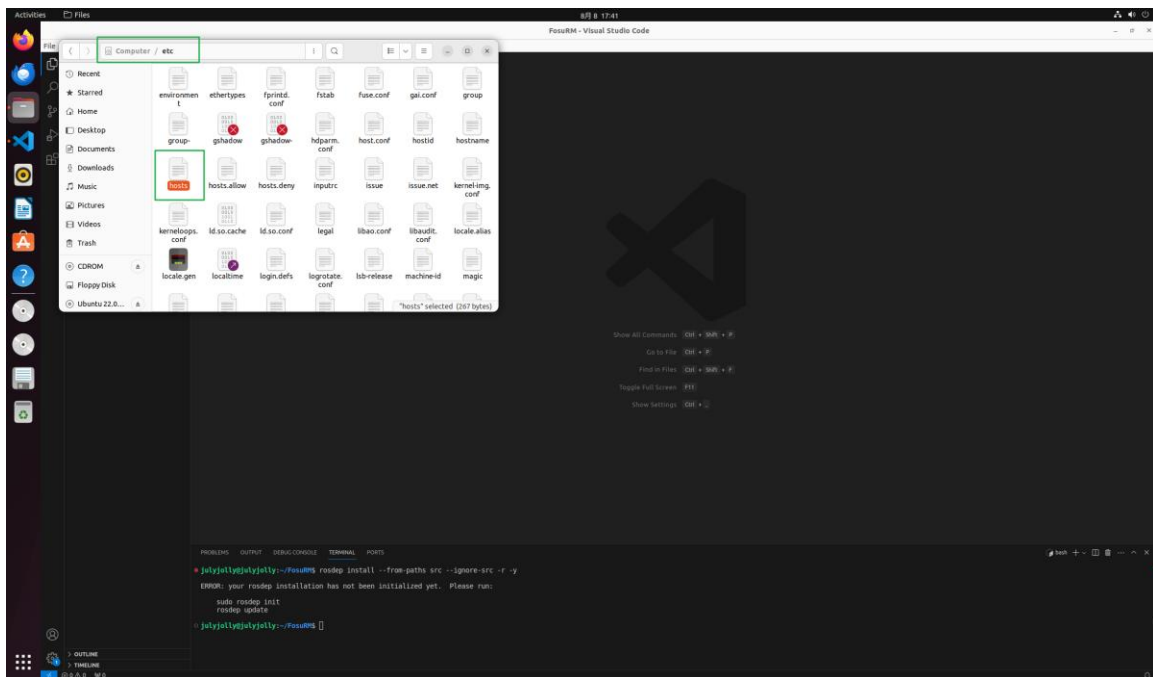
/etc/hosts [readonly] 9 lines, 224 bytes
```

然后在最后一行输入 [185.199.111.133 raw.githubusercontent.com](https://raw.githubusercontent.com) 就可以了

vi 会很难使用，也可以在文件夹中找到/etc 这个目录，如何找到/etc，只需在 home 目录下，按住你删除文字的那个按键，按两下就会跳转到



双击打开 hosts 再在最后一行添加 185.199.111.133 raw.githubusercontent.com 然后保存就好，保存的时候应该要你输入用户的密码。



8.一些疏漏的 ros-humble 包：

只要在终端输入

```
sudo apt install ros-humble-joint-state-publisher
```

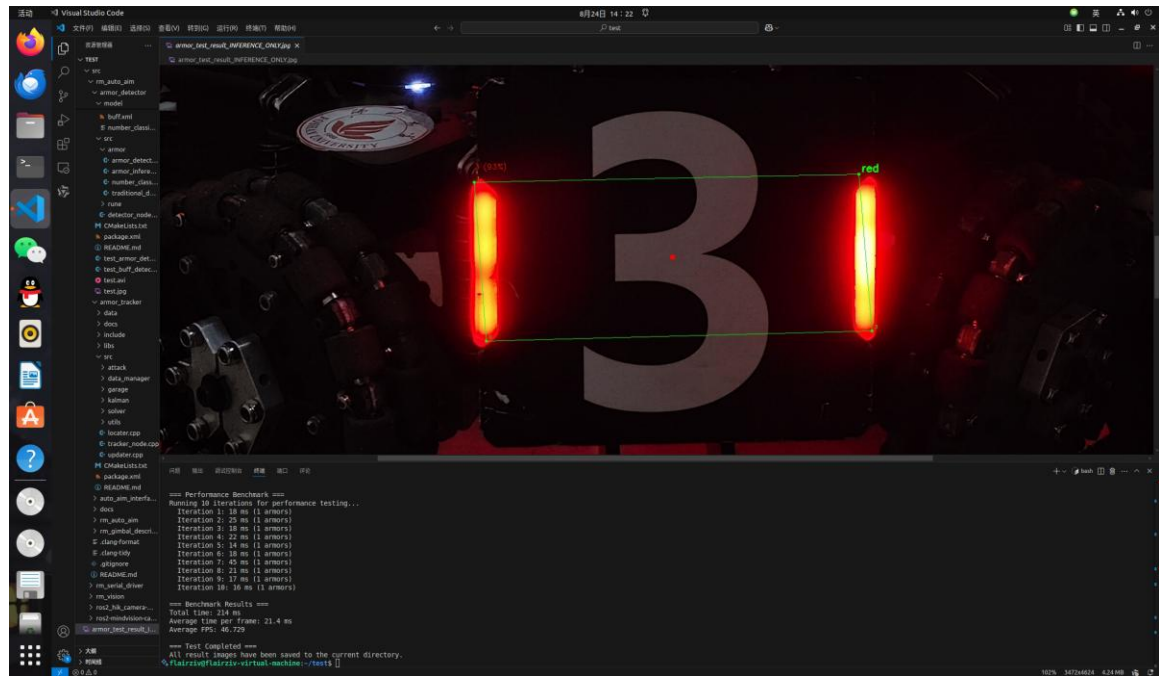
```
sudo apt install ros-humble-asio-cmake-module
```

7. 项目代码拉取与运行测试

按照战队代码 readme 编译运行，测试环境配置是否完全

装甲板识别效果

```
./build/armor_detector/test_armor_detector
```



能量机关识别效果

```
./build/armor_detector/test_buff_detector
```

