

速通安装环境

1 前言

- 本教程适用于安装时出现软件仓库报错（即在add-apt-repository命令时报错）以及编译时耗时过长甚至死机的情况
- 本教程除了提供安装的命令，会在适当位置提供解释，以帮助大家在出错时理解错误原因
- 注：本教程不提供原本是apt install安装的包，因为那些包几乎不可能装不上，真的装不上了很有可能是软件仓库环境出问题了

2 安装cmake

任何一个包在linux几乎都有两种安装方式：

1. 使用包管理器安装（在Ubuntu中就是apt指令）
2. 获取该包的源码并编译安装（耗时长，但能得到最新的版本和对本机的一定优化）

cmake容易的安装方式就是使用包管理器安装：

```
1 | sudo apt install cmake # 一句命令就可以了，不需要额外命令，包管理器会自动寻找这个包所依赖的库，然后自动安装这些库
```

3 安装OpenCV

两种方法同上

如果这里有同学已经参考环境配置文档中内容配置，请先执行下面这条语句：

```
1 | sudo add-apt-repository -r "deb http://security.ubuntu.com/ubuntu xenial-security main"
```

这句命令是在删除之前加入到仓库配置的地址，因为这个地址是16.04的仓库，仓库的公钥与当前安装的20.04系统无法匹配，所以一定会报错

```
错误:3 http://security.ubuntu.com/ubuntu xenial-security InRelease
  由于没有公钥，无法验证下列签名： NO_PUBKEY 40976EAF437D05B5 NO_PUBKEY 3B4FE6ACC0B21F32
获取:5 http://cn.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
正在读取软件包列表... 完成
W: GPG 错误: http://security.ubuntu.com/ubuntu xenial-security InRelease: 由于没有公钥，无法验证下列签名： NO_PUBKEY 40976EAF437D05B5 NO_PUBKEY 3B4FE6ACC0B21F32
E: 仓库 “http://security.ubuntu.com/ubuntu xenial-security InRelease” 没有数字签名。
N: 无法安全地用该源进行更新，所以默认禁用该源。
N: 参见 apt-secure(8) 手册以了解仓库创建和用户配置方面的细节。
```

我们可以参考[opencv官网提供的apt安装方法](#)使用编译安装的方法

也可以直接安装apt提供的包：

```
1 | sudo apt install libopencv-dev # 这个命令会把contrib包括在内
```

由于包比较大，如果使用虚拟机或者电脑的内存不大，建议直接用apt install的方式安装，不容易爆内存且安装速度异常缓慢（编译可能花1个小时及以上）

如果爆内存，整个电脑就会当场宕机

4 安装Eigen

同理，这个包依然可以用apt安装

```
1 | sudo apt install libeigen3-dev
```

5 一点小tip

当你不知道需要安装的包叫什么，不确定有没有这个包应该怎么查找呢

5.1 最简单的方法——Tab键的提示

一般的开源库都是以lib开头的，所以我们可以先打一句（以pcl为例）：

```
1 | sudo apt install libpcl # 先别按回车，按一下Tab，会像vscode代码提示一样给出一些提示的包选项
```

```
ethan@ubuntu2204-11:~/文档$ sudo apt install libpcl
libpcl1                                libpcl-recognition1.12
libpcl1-dev                            libpcl-registration1.12
libpcl-apps1.12                        libpcl-ros-dev
libpcl-common1.12                      libpcl-ros-features0d
libpcl-conversions-dev                 libpcl-ros-filter1d
libpcl-dev                             libpcl-ros-filters0d
libpcl-doc                             libpcl-ros-io0d
libpcl-features1.12                   libpcl-ros-segmentation0d
libpcl-filters1.12                     libpcl-ros-surface0d
libpcl-io1.12                          libpcl-ros-tf1d
libpcl-kdtree1.12                      libpcl-sample-consensus1.12
libpcl-keypoints1.12                   libpcl-search1.12
libpcl-ml1.12                          libpcl-segmentation1.12
libpcl-msgs-dev                        libpcl-stereo1.12
libpcl-octree1.12                      libpcl-surface1.12
libpcl-outofcore1.12                   libpcl-tracking1.12
libpcl-people1.12                      libpcl-visualization1.12
```

5.2 如果还是找不到，那就用apt list/search命令

我们来看看list命令的描述信息（使用apt --help，实际上，大多数命令都支持使用--help）：

```
ethan@ubuntu2204-11:~/文档$ apt --help
apt 2.4.5 (amd64)
用法： apt [选项] 命令

命令行软件包管理器 apt 提供软件包搜索，管理和信息查询等功能。
它提供的功能与其他 APT 工具相同（像 apt-get 和 apt-cache），
但是默认情况下被设置得更适合交互。

常用命令：
list - 根据名称列出软件包
search - 搜索软件包描述
```

我们看到有一个search命令和一个list命令，它们的区别在于：

- search命令除了会用关键字对包名进行检索，也会对包的描述信息(description)进行检索，且检索的内容只需要包含该关键字即可
- list命令只检索包名，且需要使用正则表达式进行对字符串进行完整匹配

请各位分别尝试按照下面描述的方法尝试两者的使用：

5.2.1 apt list

```
1 apt list eigen
```

如果你打了这句话之后可能啥都没看到，这很正常，刚刚我们提到：list需要正则表达式对包名进行完整匹配，所以这里的命令应该改为：

```
1 apt list *eigen*
```

相信这回就有东西了

5.2.2 apt search

```
1 apt search eigen
```

打完发现：完蛋，怎么这么多，怎么看得了呢？

面对这种方法有两种解决办法：

1. 采用管道+less命令将结果截胡，像浏览网页文字一样浏览

```
1 apt search eigen | less # 退出请按q键
```

2. 采用管道+grep命令查找其中符合关键字信息的行

```
1 apt search eigen | grep eigen
```

这种方法得到的结果相比而言就少很多了，但是相应的，软件包描述就看不到了

5.3 依然找不到？那就试试在这个网站找

[ubuntu包官方搜索网站](#)