

autoware 安装教程(source 版本安装)

1. 系统: ubuntu18.04/Melodic 的系统依赖项

```
$ sudo apt-get update
```

```
$ sudo apt-get install -y python-catkin-pkg python-rosdep ros-$ROS_DISTRO-catkin gksu
```

```
$ sudo apt-get install -y python3-pip python3-colcon-common-extensions python3-setuptools  
python3-vcstool
```

```
$ pip install -U setuptools
```

2. CUDA 支持的附加系统依赖项

有关 CUDA10.0 的安装说明, 请参阅:

<https://docs.nvidia.com/cuda/archive/10.0/cuda-installation-guide-linux/index.html>

有关 CUDA9.0 的安装说明, 请参阅:

<https://docs.nvidia.com/cuda/archive/9.0/cuda-installation-guide-linux/index.html>

有关 CUDA8.0 的安装说明, 请参阅:

<https://docs.nvidia.com/cuda/archive/8.0/cuda-installation-guide-linux/index.html>

注意: 要在 Melodic 上启用 CUDA, 需要更新 Eigen

eigen3.37 版本的下载地址: <https://gitlab.com/libeigen/eigen/-/releases/3.3.7>

```
$ cd && wget https://gitlab.com/libeigen/eigen/-/archive/3.3.7/eigen-3.3.7.zip #Download Eigen
```

```
$ mkdir eigen && tar --strip-components=1 -xzf 3.3.7.tar.gz -C eigen #Decompress
```

```
$ cd eigen && mkdir build && cd build && cmake .. && make && make install #Build and install
```

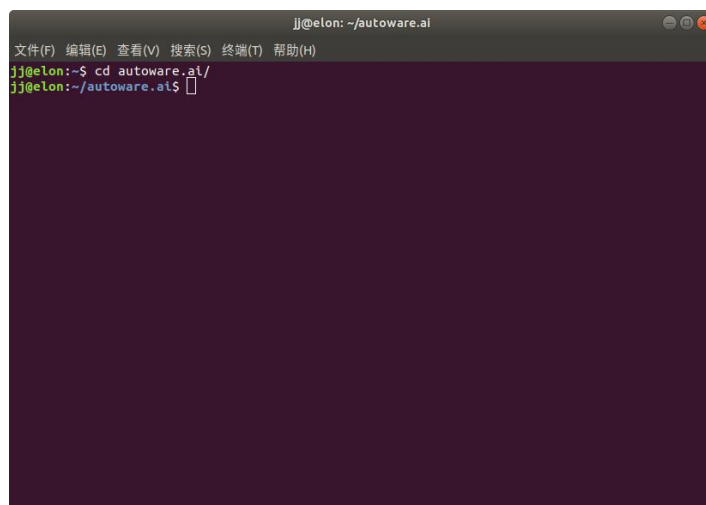
```
$ cd && rm -rf 3.3.7.tar.gz && rm -rf eigen #Remove downloaded and temporary files
```

如何构建工作区

1. 创建工作区

```
$ mkdir -p autoware.ai/src
```

```
$ cd autoware.ai
```



2. 下载 autoware.ai 的工作区配置

对于 1.12.0 版本

```
$ wget -O autoware.ai.repos
```

```
"https://raw.githubusercontent.com/autowarefoundation/autoware_ai/1.12.0/autoware.ai.repos"
```

对于 1.13 版本

```
$ wget -O autoware.ai.repos
```

```
"https://raw.githubusercontent.com/autowarefoundation/autoware_ai/1.13.0/autoware.ai.repos"
```

Autoware1.13 压缩包

链接: <https://pan.baidu.com/s/1HdUMOIInPdyBb9qpsaHekAA?pwd=vhk1>

提取码: vkh1

对于 1.14 版本

```
$ wget -O autoware.ai.repos
```

```
"https://raw.githubusercontent.com/autowarefoundation/autoware_ai/1.14.0/autoware.ai.repos"
```

Autoware1.14 压缩包

链接: <https://pan.baidu.com/s/1f-cT1oDjuGOQxS7mizRAbA?pwd=bk9i>

提取码: bk9i

3. 将 Autoware.ai 下载到工作区

```
$ vcs import src < autoware.ai.repos
```

有时候 2, 3 两步会经常报错, 网络链接不稳定, 出现下载不了, 安装不了的情况。基于此我们采用小鱼大佬的方法。做完这一步直接跳过第 4 步, 进入第 5 步。

```
$ sudo apt install python-pip
```

```
$ sudo pip install rosdep
```

```
$ sudo rosdep init
```

```
$ rosdep update
```

```
$ rosdep install -y --from-paths src --ignore-src --rosdistro $ROS_DISTRO
```

4. 使用 rosdep 安装

```
$ rosdep update
```

```
$ rosdep install -y --from-paths src --ignore-src --rosdistro $ROS_DISTRO
```

我这里的第二个采用的是: rosdep install --from-paths src --ignore-src -r -。

5. 编译工作区

支持 CUDA

```
$ AUTOWARE_COMPILE_WITH_CUDA=1 colcon build--cmake-args
```

```
-DCMAKE_BUILD_TYPE=Release --continue-on-error
```

不支持 CUDA

\$ colcon build --cmake-args -DCMAKE_BUILD_TYPE=Release --continue-on-error

在经历过第 5 步之后，会出现一些编译的错误，例如：ndt_gpu 中 CUDA 报错

修改 CUDA 相关的版本检查参数

修改文件为：

autoware.ai/src/autoware/common/autoware_build_flags/cmake/autoware_build_flags-extras.cmake

修改后的结果为：

```
打开(O)  autoware_build_flags-extras.cmake  保存(S)
~/autoware.ai/src/autoware/common/autoware_build_flags/cmake

if(CMAKE_CXX_COMPILER_ID STREQUAL "GNU")
  set(CMAKE_SHARED_LINKER_FLAGS "${CMAKE_SHARED_LINKER_FLAGS} -Wl,-no-undefined")
elseif(CMAKE_CXX_COMPILER_ID MATCHES "Clang")
  set(CMAKE_SHARED_LINKER_FLAGS "${CMAKE_SHARED_LINKER_FLAGS} -Wl,-undefined,error")
endif()

# Enable support for C++14
if(${CMAKE_VERSION} VERSION_LESS "3.1.0")
  set(CMAKE_CXX_FLAGS "${CMAKE_CXX_FLAGS} -std=c++14")
else()
  set(CMAKE_CXX_STANDARD 14)
endif()

message(STATUS "CUDA compilation status: ${ENV{AUTOWARE_COMPILE_WITH_CUDA}}.")

macro(AW_CHECK_CUDA)
  if ($ENV{AUTOWARE_COMPILE_WITH_CUDA})
    find_package(CUDA REQUIRED)
    find_package(Eigen3 REQUIRED)

    if(NOT ${CUDA_VERSION} VERSION_LESS "11.1"
      AND NOT ${CUDA_VERSION} VERSION_EQUAL "11.1" )
      message(FATAL_ERROR "GPU support on Melodic requires CUDA<=11.1")
    endif()
    if(${CUDA_VERSION} VERSION_GREATER "9.1"
      AND ${CMAKE_VERSION} VERSION_LESS "3.12.3")
      unset(CUDA_cublas_device_LIBRARY CACHE)
      set(CUDA_cublas_device_LIBRARY ${CUDA_cublas_LIBRARY})
      set(CUDA_CUBLAS_LIBRARIES ${CUDA_cublas_LIBRARY})
    endif()
    if ("${ENV{ROS_DISTRO}}" STREQUAL "melodic" AND ${EIGEN3_VERSION_STRING} VERSION_LESS "3.3.7")
      message(FATAL_ERROR "GPU support on Melodic requires Eigen version>= 3.3.7")
    endif()
    set(USE_CUDA ON)
  else()
    message(WARNING "CUDA support is disabled. Set the AUTOWARE_COMPILE_WITH_CUDA environment variable and recompile to enable it")
    set(USE_CUDA OFF)
  endif()
endmacro()
```

将 10.0 改成 11.1，因为我安装的是 11.1 版本的 CUDA，因人而异，也可以是其他的版本，根据个人的电脑配置进行更改。更改之后在运行一遍：\$ AUTOWARE_COMPILE_WITH_CUDA=1 colcon build--cmake-args -DCMAKE_BUILD_TYPE=Release --continue-on-error

6. 启动 autoware，界面如下：

