

1 实践：G1 机器人仿真环境搭建与运动控制

1 作业目标

熟悉 G1 仿真环境的安装流程，掌握 G1 在仿真环境中的基本控制方法。

2 系统环境

推荐在 Ubuntu 20.04 系统下进行开发，暂不支持在 Mac、Windows 系统下进行开发。

3 安装流程

3.1 ROS1 环境安装，版本为 Noetic

使用鱼香ROS一键安装指令：

代码块

```
1  wget http://fishros.com/install -O fishros && . fishros
```

安装完毕后重启系统。

3.2 Gazebo 环境安装, 版本为 Gazebo Classic

参照以下安装流程：

1. 更新系统

代码块

```
1  sudo apt update && sudo apt upgrade -y
```

2. 安装依赖库

代码块

```
1  sudo apt install -y \  
2      cmake \  
3      curl \  
4      git \  

```

```
5     libfreeimage-dev \  
6     libprotoc-dev \  
7     protobuf-compiler \  
8     libignition-math6-dev \  
9     libsqlite3-dev \  
10    libtinysql2-dev \  
11    libgflags-dev \  
12    libavformat-dev \  
13    libavcodec-dev
```

3. 添加 Gazebo 官方软件源

代码块

```
1  sudo apt install -y wget  
2  wget https://packages.osrfoundation.org/gazebo.gpg -O /usr/share/keyrings/pkgs-  
   osrf-archive-keyring.gpg  
3  echo "deb [arch=$(dpkg --print-architecture) signed-  
   by=/usr/share/keyrings/pkgs-osrf-archive-keyring.gpg]  
   http://packages.osrfoundation.org/gazebo/ubuntu-stable $(lsb_release -cs) main"  
   | sudo tee /etc/apt/sources.list.d/gazebo-stable.list > /dev/null
```

4. 安装 Gazebo

代码块

```
1  sudo apt update  
2  sudo apt install -y gazebo11 libgazebo11-dev
```

5. 配置环境变量

代码块

```
1  echo "source /usr/share/gazebo/setup.sh" >> ~/.bashrc  
2  source ~/.bashrc
```

6. 验证安装

代码块

```
1  gazebo worlds/empty.world
```

安装完毕后重启系统。

3.4 Eigen安装，版本为3.4

编译并安装。

代码块

```
1 cd robocup_g1/eigen-3.4.0
2 mkdir build
3 cd build
4 cmake ..
5 sudo make install
```

3.5 RealSense SDK 安装

代码块

```
1 sudo apt-get install ros-noetic-realsense2-camera
```

3.6 Ignition-math 安装

代码块

```
1 sudo apt install libignition-math4-dev
```

如果有遇到以下依赖项缺少，可以参考安装，没有的话不需要安装：（可选）

代码块

```
1 sudo apt-get install ros-noetic-gazebo-ros-pkgs ros-noetic-gazebo-ros-control
2 sudo apt-get install ros-noetic-joint-state-controller ros-noetic-ros-control
  ros-noetic-ros-controllers
3 sudo apt-get install ros-noetic-robot-state-publisher
4 sudo apt-get install ros-noetic-xacro
```

4 编译流程

先下载项目代码。

代码块

```
1 cd ~/robocup_g1/
2 catkin_make
```

5 启动流程

5.1 启动 Gazebo:

这里对 CPU 内存需求比较大，建议16G+

代码块

```
1 source ~/robocup_g1/devel/setup.bash
2 roslaunch unitree_guide gazebo.launch
```

5.2 启动运控

代码块

```
1 source ~/robocup_g1/devel/setup.bash
2 ./devel/lib/unitree_guide/junior_ctrl
```

5.3 切换模式

在 `junior_ctrl` 主界面输入 `2`，控制机器人从 State_Passive 切换到 fixed stand。回到 Gazebo 主界面，按下暂停键，然后在主菜单中选择 Edit/Reset Model Poses 以重置机器人的位姿。在 `junior_ctrl` 主界面输入 `4`，控制机器人从 fixed stand 切换到 LOCOMOTION。回到 Gazebo 主界面，点击播放键，重启应用。

6 作业安排

6.1 使用按键控制机器人移动

- 前后运动：
 - `W` 键：向前运动
 - `S` 键：向后运动
- 左右平移：
 - `A` 键：向左平移
 - `D` 键：向右平移
- 左右旋转：

- J 键：向左旋转
- L 键：向右旋转

6.2 使用代码控制机器人（基于Python实现）

代码块

```
1 source ~/catkin_ws/devel/setup.bash
```

参照 `examples/move_g1_demo.py` 中的控制思路，控制机器人走出正方形或圆形等轨迹。