

目录

1.检测工控机.....	2
2.检测人脸盒子.....	2
3.ROS 操作系统检测.....	2
4.检测音响设备.....	2
5.检测深度摄像头.....	4
6.检测激光雷达.....	4
7.检测超声传感器.....	5
8.检测红外传感器.....	6
9.检测路由器和 3G.....	6
10.云台控制检测.....	6
11.检测移动功能.....	7
11.1 移动速度.....	7
11.2 移动控制精度.....	8
11.3 检测伺服电机.....	8
12.人脸识别交互.....	8
12.1 人脸注册.....	8
12.2 人脸注销.....	10
12.2.1 注销指定用户.....	10
12.2.2 注销所有用户.....	10
12.3 人脸识别检测.....	10
13.语音交互.....	11
13.1 语音交互.....	11
13.2 语音控制.....	12
14.平板触控交互.....	12

14.1 APP 控制机器人.....	12
14.2 APP 与机器人交互.....	12

XBot 机器人主机（工控机，IP 地址 192.168.8.101）上执行。

1. 检测工控机

通过以下命令分别检测 CPU、内存、SSD、Ubuntu 系统信息：

`cat /proc/cpuinfo` #检测 cpu 信息

`cat /proc/meminfo` #检测内存信息

`ls SCSI` #查看 ssd 硬盘信息（请 `sudo apt install ls SCSI` 连接互联网后安装）

2. 检测人脸盒子

`ssh ubuntu@192.168.8.141`

输入密码 `youtu@)!^`（即 `youtu` 之后，按住 `shift` 后依次输入 2016）

`ls pci` #可查看到人脸盒子使用显卡型号为 Nvidia TK1

`cat /proc/cpuinfo` #可检测 cpu 信息

3. ROS 操作系统检测

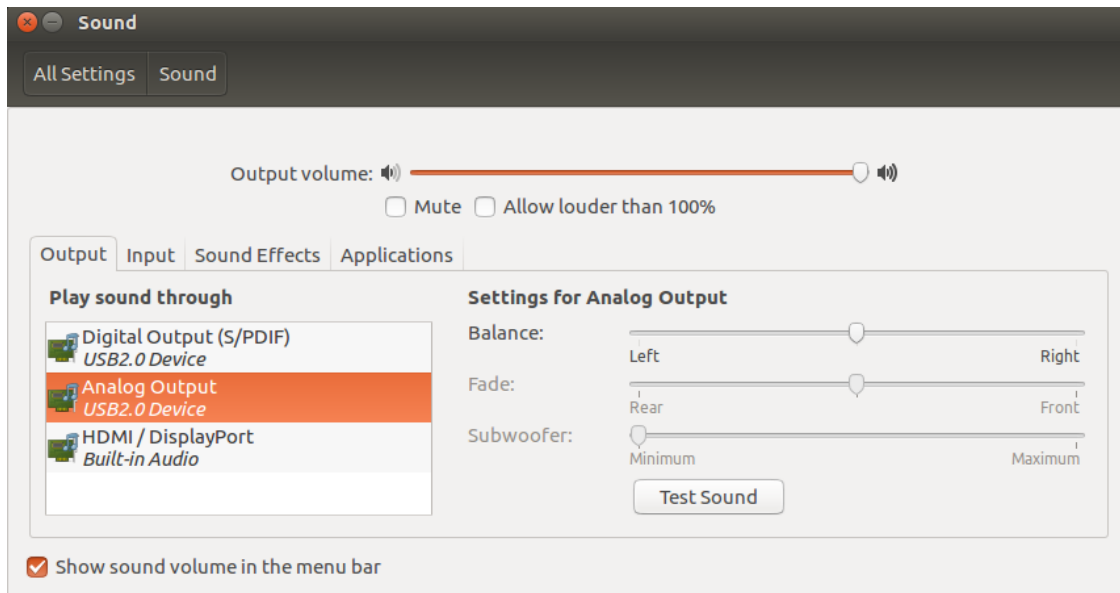
`roslaunch xbot_bringup xbot-u.launch` #使用 ROS 启动机器人

`rostopic list` #查询机器人启动后的 ROS 话题信息

4. 检测音响设备

检查音响可以在屏幕右上角音箱中检测。具体操作为“喇叭图标”→Sound Settings

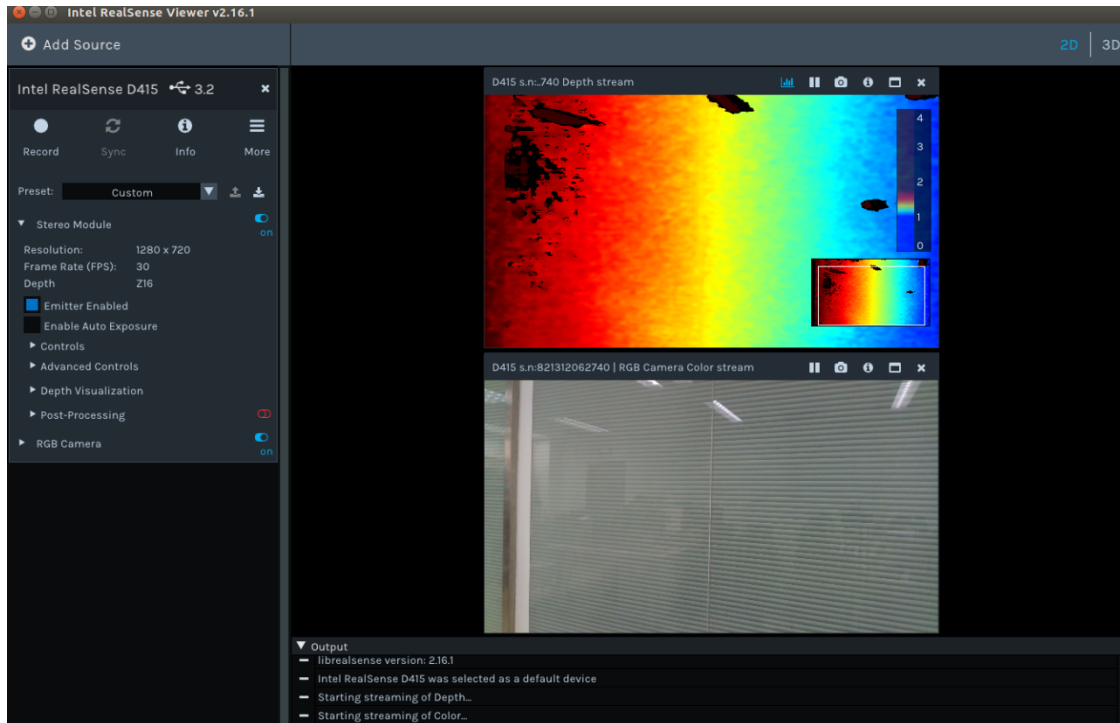
选择 **Output** 标签页，选择设备，调节 **Output volume** 调节输出音量到最大；点击 **Test Sound** 可以看到左喇叭和右喇叭分别测试的 **Test** 按钮，检测左右喇叭是否正常发音，方向是否准确，音量是否洪亮。



5. 检测深度摄像头

`realsense-viewer` #启动深度摄像头观看软件

将 Stereo Module 和 RGB Camera 对应的开关按钮设置为 on 状态，可以看到深度图像和 RGB 图像信息。



6. 检测激光雷达

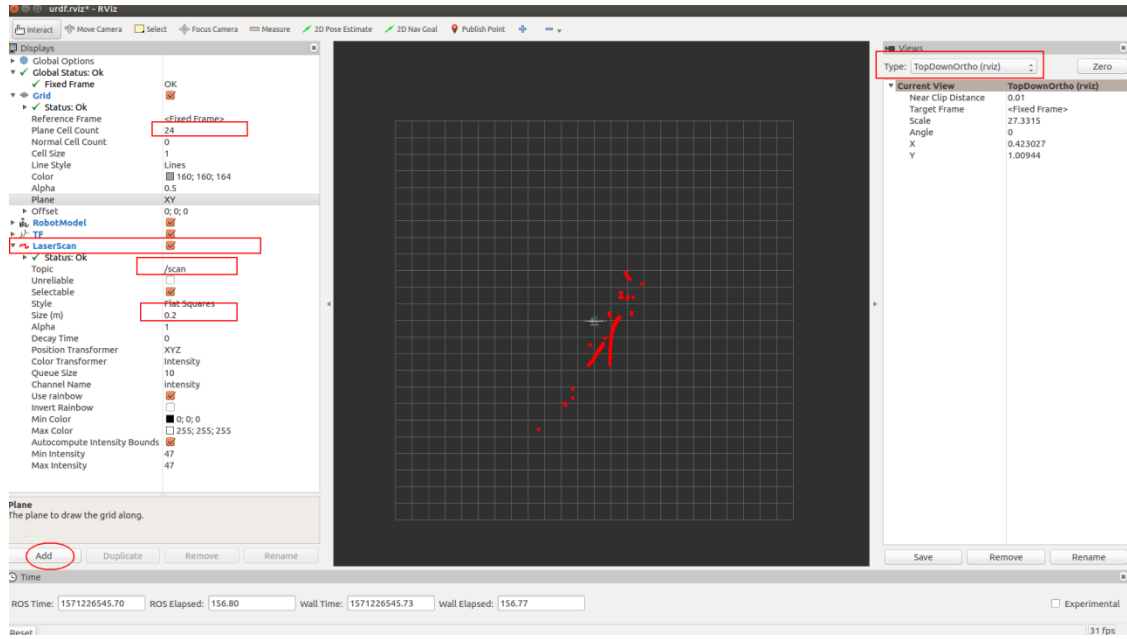
目测激光雷达正常旋转

`rostopic echo /scan` #获取到激光雷达的数据，检测激光雷达正常工作

使用 `rviz` 工具测试激光雷达

`roslaunch xbot_description display.launch` #启动 xbot rviz 图形化显示，操作步骤如下

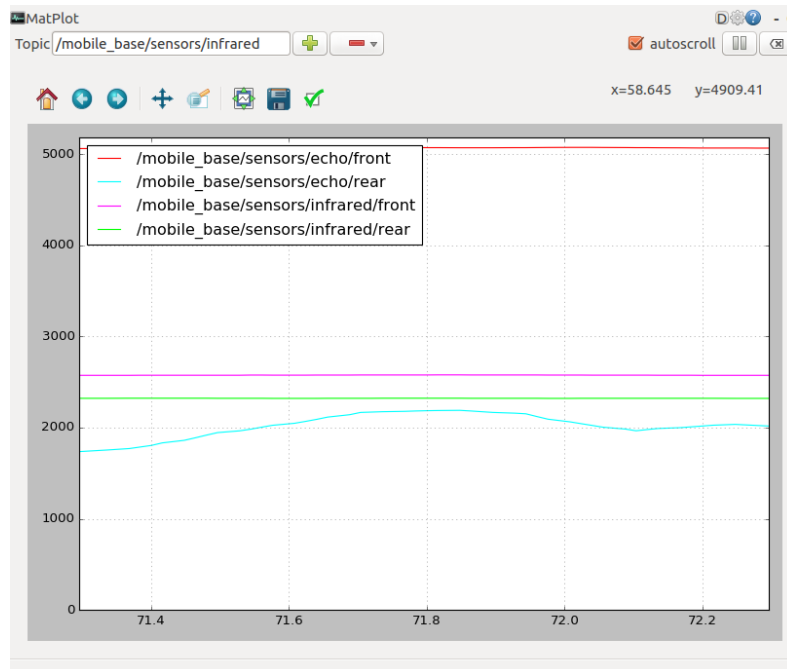
- 1) 点击 Add 添加 LaserScan，并将其 topic 设置为 /scan。为了方便观察，将 Size（雷达信号显示宽度）设置为 0.2；
- 2) 将右侧 Views 下的 Type 设置为 TopDownOrtho(rviz)，调整显示视角，方便查看；
- 3) 如果需要测量激光雷达所达距离，打开 Grid，将其 Plane Cell Count 设置为合适的值，一小格代表 1 米。



7. 检测超声传感器

终端窗口键入 `rqt_plot`，在该界面中，Topic 框中输入 `/mobile` 则下面自动出现多种传感器备选项，末尾以 `echo` 结尾的，是超声传感器。将其添加到显示区，在机器人前后方设置障碍物并同时观测其波形。

超声测距公式：y 轴的值/58 = 实际距离



8. 检测红外传感器

终端窗口键入 `rqt_plot`，Topic 输入 `/mobile`，以 `infrared` 结尾的是红外测距传感器，我们加上后，将机器人抬起，改变离地距离，同时观察波形变化。波形变化规律为，离地距离越远，y 轴越低。

9. 检测路由器和 3G

机器人自带的路由器的默认 IP 地址为：192.168.8.1

在任意连接了机器人 `wifi`（名称 `xbotxxxx`）的 PC 电脑上的浏览器中，输入进入登录界面，输入用户名 `admin`、密码 `admin` 登录。

如果在路由器中插入了 `SIM` 卡，还可以支持 `4G` 网络。可以在登录路由器后，选择“网络设置”，接着选择“`4G`”进行设置。



10. 云台控制检测

键入以下命令查看舵机俯仰角运转是否正常，最后面的数字代表其角度，范围是-60——30 之间，超过范围机器人不响应。

```
rostopic pub -1 /mobile_base/commands/pitch_platform std_msgs/Int8 "data: 0"
```

键入以下命令查看舵机水平偏转角是否运转正常，范围是-90——90 之间，超过范围机器人不响应。

```
rostopic pub -1 /mobile_base/commands/yaw_platform std_msgs/Int8 "data: 0"
```

11. 检测移动功能

```
roslaunch xbot_tools keyboardteleop.py #启动 xbot 键盘运动控制工具
```

进入机器人操作界面，界面中已有提示控制命令如下图，可试验机器人行动是否正常。

```
xbot@xbot:~$ roslaunch xbot_tools keyboard_teleop.py

Reading from the keyboard and Publishing to Twist!
-----
Moving around:
   u       i       o
   j       k       l
   m       ,       .
For Holonomic mode (strafing), hold down the shift key:
-----
   U       I       O
   J       K       L
   M       <       >
anything else : stop
q/z : increase/decrease current speeds by 10%
w/x : increase/decrease only linear speed by 10%
e/c : increase/decrease only angular speed by 10%
CTRL-C to quit

currently:      speed 0.5      turn 1.0
```

11.1 移动速度

机器人移动的速度详见上图中的 **currently** 行所示的 **speed**（线速度）和 **turn**（角速度）。

按 **z** 键可以调低速度，将速度降低 10%。按 **q** 键可以将速度增加 10%。目前机器人建议设置的速度不超过 2m/s。最精确的速度控制可达 0.01m/s。

可以按照命令窗口提示的按键进行速度调试，支持连续多次设置速度。

11.2 移动控制精度

连续多次按 **z** 降低速度到 **speed** 为 0.01 左右。

新开一个终端，输入命令 **rostopic echo /odom**

然后将鼠标再次聚焦到 **keyboard_teleop.py** 窗口，并持续按 **i** 键驱动机器人前进，同时观察 **odom** 中 **linear** 的 **x** 值在 0.01 左右。证明机器人的速度进度可达 0.01m/s。

11.3 检测伺服电机

在 **keyboard_teleop.py** 窗口，按下 **L** 键（右转）或者 **J** 键（左转），此时两轮的速度是不同的。这是因为我们每个轮子都有一个伺服电机，控制轮子的速度。

12. 人脸识别交互

12.1 人脸注册

人脸识别摄像头图像检测正常之后,接着就可以开始人脸注册了.

XBot-U 机器人上搭载了一台人脸识别盒子，固定 IP 地址为 **192.168.8.141**。

正常情况下在未注册任何人脸时通过地址 **http://192.168.8.141:8000/management/userids** 在已连接机器人无线网的设备上浏览器查看已注册用户 **id** 返回的为空数组。

人脸识别功能主要分为注册人脸，管理人脸和识别人脸三个部分。接下来将依次为您介绍它们的使用方法。

注册人脸分为人脸图片注册和实时拍照注册两种方式。

首先，将您的机器人连接显示器与键盘鼠标。因为机器人的开机自启动程序已经占用了人脸识别摄像头，所以要先在机器人的终端上输入如下指令终止全部服务：

```
sudo service xbot stop
```

释放人脸识别摄像头之后，继续输入下面指令启动 ROS：

```
roscore
```

接下来就是任意从下面的两种人脸注册方式中选择一种进行人脸注册。目前支持使用摄像头交互注册和使用已有的照片注册两种方式，命令分别如下：

#使用摄像头交互进行注册：

```
roslaunch xbot_face 01face_register.py camera [camera_index]
```

#使用照片进行注册：

```
roslaunch xbot_face 01face_register.py image [user_name]
```


使用摄像头交互注册，需要根据设备情况填写[camera_index]参数，该参数表示设备的id。我们可以通过以下命令查询：

ls /dev

执行后会得到以下结果：

```
xbot@xbot:~$ ls /dev
autofs          loop0           sensor          tty31           tty8            ttyUSB2
block           loop1           serial          tty32           tty9            uhid
bsg             loop2           sg0             tty33           ttyprintk       uinput
btrfs-control   loop3           shm             tty34           ttyS0           urandom
bus             loop4           snapshot        tty35           ttyS1           usb
char           loop5           snd             tty36           ttyS10          userio
console         loop6           stderr          tty37           ttyS11          v4l
core            loop7           stdin           tty38           ttyS12          vcs
cpu             loop-control    stdout          tty39           ttyS13          vcs1
cpu_dna_latency mapper          tpm0            tty4            ttyS14          vcs2
cuse            mcelog          tpmrm0          tty40           ttyS15          vcs3
disk            media0          tty             tty41           ttyS16          vcs4
dri             media1          tty0            tty42           ttyS17          vcs5
drm_dp_aux0     media2          tty1            tty43           ttyS18          vcs6
drm_dp_aux1     mei0            tty10           tty44           ttyS19          vcsa
ecryptfs        mem             tty11           tty45           ttyS2           vcsa1
fb0             memory_bandwidth tty12           tty46           ttyS20          vcsa2
fd              queue           tty13           tty47           ttyS21          vcsa3
full            net             tty14           tty48           ttyS22          vcsa4
fuse            network_latency tty15           tty49           ttyS23          vcsa5
hidraw0         network_throughput tty16           tty5            ttyS24          vcsa6
hidraw1         null            tty17           tty50           ttyS25          vfio
hpet            port            tty18           tty51           ttyS26          vga_arbiter
hugepages       ppp             tty19           tty52           ttyS27          vhci
hwrng           psaux           tty2            tty53           ttyS28          vhost-net
i2c-0           ptmx            tty20           tty54           ttyS29          vhost-vsock
i2c-1           ptp0            tty21           tty55           ttyS3           video0
i2c-2           pts             tty22           tty56           ttyS30          video1
i2c-3           random          tty23           tty57           ttyS31          video2
i2c-4           rfkill          tty24           tty58           ttyS4           video3
initctl         rplidar         tty25           tty59           ttyS5           xbot
input           rtc             tty26           tty6            ttyS6           zero
kmsg            rtc0            tty27           tty60           ttyS7
kvm             sda             tty28           tty61           ttyS8
lightnvm        sda1            tty29           tty62           ttyS9
lirc0           sda2            tty3            tty63           ttyUSB0
log             sda3            tty30           tty7            ttyUSB1
```

其中，video*这些设备就是摄像头设备，在机器人上一般有4个摄像头。其中video3是人脸注册摄像头，我们一般用该摄像头进行人脸注册，传入摄像头id号3，则命令如下：

```
roslaunch xbot_face 01face_register.py camera 3
```

执行以上命令就会打开一个摄像头窗口，在窗口中能够清楚的看到图像信息。此时您需要用鼠标点击摄像头窗口（此时窗口就处于激活状态），然后将人脸置于窗口中间，调整人脸在窗口中的位置和清晰度，合适时，按下R键，此时摄像头窗口的画面将会定格。

然后返回到终端，命令窗口会提示您输入姓名拼音，如"xjpcnew03"(注意要加双引号)并回车，如果下一行返回的是0，表示注册成功，其后只需要鼠标点击摄像头窗口并按Q或者直接终端键入ctrl+c即可退出人脸识别程序，此时就完成了一次人脸注册。

```
xj@xj-ThinkPad-T470p:~$ rosrunc xbot_face 01face_register.py camera 0
使用摄像头交互进行注册:
请用拼音输入你的名字:
"xjpcnew03"
返回状态: 0 成功
```

注册的信息可以通过在浏览器中直接输入接口

<http://192.168.8.141:8000/management/userids> 返回，查看返回结果是否包含刚刚注册的用户 id: xjpcnew03。

12.2 人脸注销

12.2.1 注销指定用户

注销指定用户，建议在注销之前，通过 <http://192.168.8.141:8000/management/userids> 接口查询已经注册的用户 id。

假设删除 xjnew03 用户，那么执行命令，返回为 0 则表示注销成功：

```
roslaunch xbot_face 02face_logout.py xjpcnew03
```

```
xj@xj-ThinkPad-T470p:~$ roslaunch xbot_face 02face_logout.py xjpcnew03
返回状态: 0 成功
```

再次访问 <http://192.168.8.141:8000/management/userids> 接口查询已经注册的用户，已经没有 xjnew03，该用户已经成功被注销。

12.2.2 注销所有用户

注销所有用户则只需要输入参数 all：

```
roslaunch xbot_face 02face_logout.py all
```

```
xj@xj-ThinkPad-T470p:~$ roslaunch xbot_face 02face_logout.py all
{'u'Ret': 0}
```

12.3 人脸识别检测

人脸注册后，再次启动 XBot

```
roslaunch xbot_bringup xbot-u.launch
```

或重启机器，人脸识别程序会随机启动，直接查看人脸识别发出的识别结果 topic 即可

```
rostopic echo /xbot/face_result
```

执行以上命令之后，会一直不停的输出以下格式的信息：

```
face_exist: False
name: "UNKNOWN"
confidence: 0.0
```

当无人脸置于摄像头视阈范围，那么则会输出 *faceexist: False name: "UNKNOWN"*；当摄像头捕获到人脸信息时，会进行识别，并输出 *faceexist: True*，且 *name* 会输出匹配的注册用户名以及 *confidence* 把握度信息。

```
face_exist: True
name: "xijing"
confidence: 0.632823
```

13. 语音交互

13.1 语音交互

调用播放文字的方法为

```
rosservice call /xbot/play "loop: false
mode: 2
audio_path: "
tts_text: '请和我打招呼'"
success: True
```

mode:1 表示播放音频文件；此时 *audio_path* 的值需要设置为音频文件的路径；

mode:2 表示文字转换为语音；此时 *audiopath* 为空，*ttstext* 非空；

调用对话的方法为：

```
rosservice call /xbot/chat "start_chat: true"
```

对话即打开。

可以与机器进行简单对话,对话请注意在机器的嘟声之后对他说话:

- 你好
- 你多大了?
- 你能做什么?
- 介绍一下重德智能

13.2 语音控制

`rosservice call /xbot/chat "start_chat: true"`打开语音对话之后，可以通过语音对机器人下达控制命令：

- 向前走
- 向左转
- 向右转
- 向后退
- 关闭：对机器人说“关闭”机器人将关闭对话服务。下次对话之前，需要使用命令 `rosservice call /chat "start_chat: true"`再次启动对话服务。

14. 平板触控交互

14.1 APP 控制机器人

拿出机器人配套的平板，首先连接机器人 **wifi**，然后再打开“**UXBot 助手 APP**”，当界面显示“连接 **ROS** 服务器成功”则表示 **app** 已经连上机器人，可以正常操控机器人了。

1) app 遥控机器人移动

通过“显控”面板下的“方向控制轮盘”控制机器人运动。主要测试向前、向后、左前、右前、左后、右后、左转、右转几个方向的运动控制。

2) app 控制云台水平旋转和竖直俯仰

通过调整“显控”面板下的“水平云台角度”控制机器人水平旋转的角度。通过调整“竖直云台角度”控制机器人水平旋转的角度。

14.2 APP 与机器人交互

APP 除了控制机器人，还能支持状态查询，如“显示电池剩余电量”、“摄像头图像回传”、“设置机器人的速度”等等。具体可以参考平板上安装的 **APP** 功能。