随拍机器人项目开发经验分享

符国和

智能机器人运动与视觉实验室
华东师范大学
http://www.robotics.sei.ecnu.edu.cn

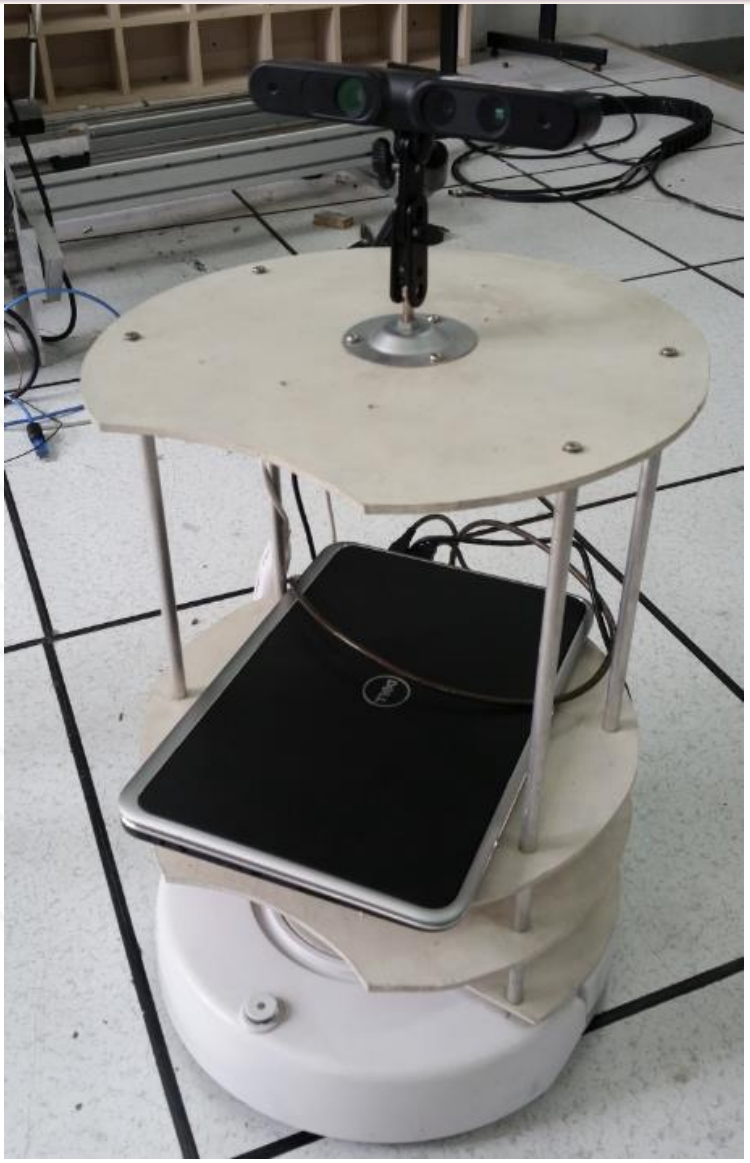
随拍机器简介

实现功能：

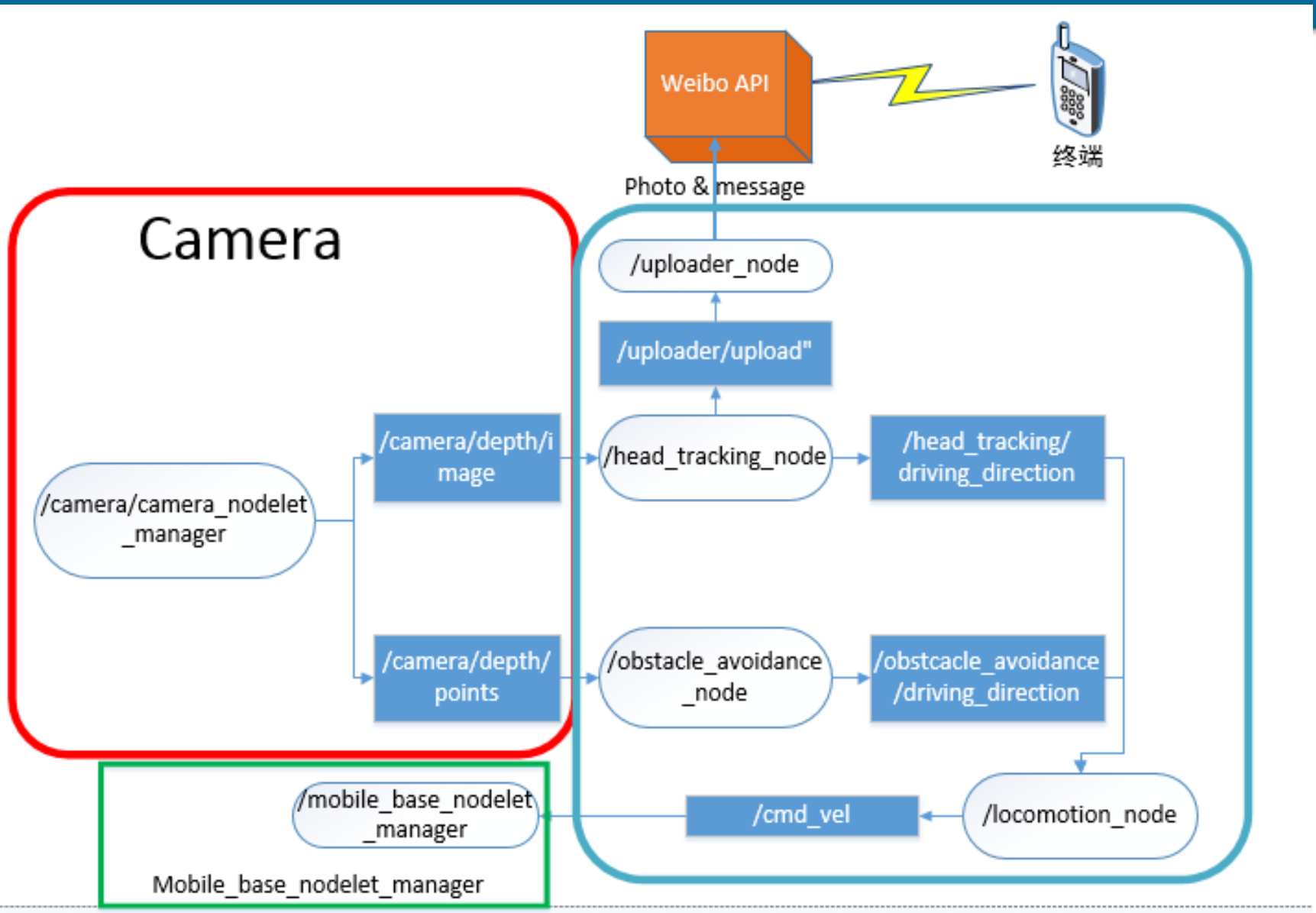
- 自主避障行走
- 头部检测
- 自动拍照
- 上传社交网络（微博）

扩展功能：

- 语音控制
- 手势控制



框架图


Weibo API
终端
Photo & message
Camera
/uploader_node
/uploader/upload"
/camera/depth/image
/head_tracking_node
/head_tracking/driving_direction
/camera/camera_nodelet_manager
/camera/depth/points
/obstacle_avoidance_node
/obstcacle_avoidance/driving_direction
/mobile_base_nodelet_manager
/cmd_vel
/locomotion_node
Mobile_base_nodelet_manager

Camera --Xtion Pro Live

最大功耗	低于2.5W
使用距离	0.8m ~ 3.5m之间
视野	58° H, 45° V, 70° D (水平, 垂直, 对角)
传感器	RGB& 深度& 麦克风*2

Camera --Xtion Pro Live

启动Xtion Pro Live：

: $ roslaunch openni2_launch openni2.launch

启动 Kinect：

$ roslaunch openni_launch openni.launch

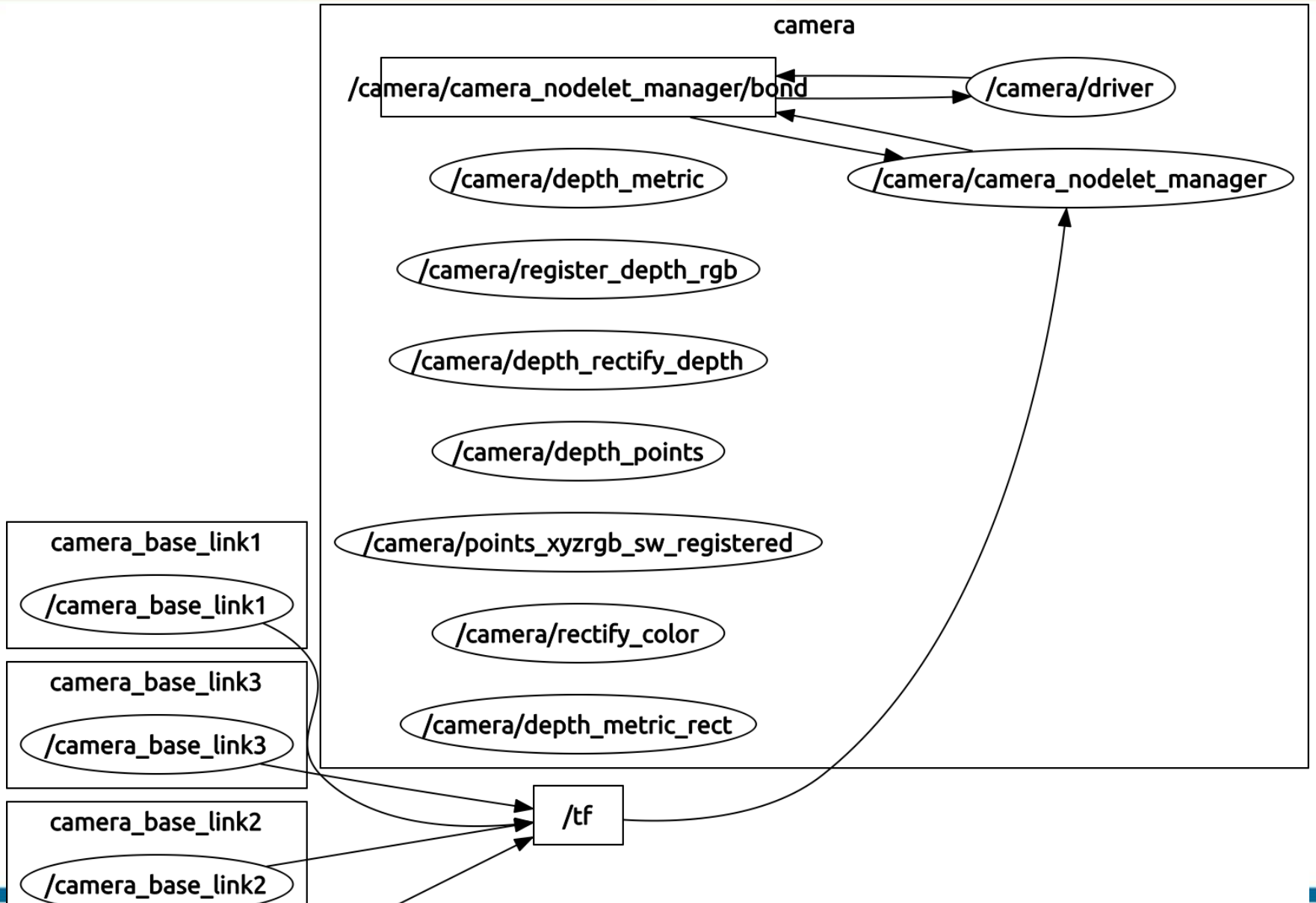
查看graph ：

$ rqt_graph

或者

$ rosrun rqt_graph rqt_graph

Camera Rosgraph



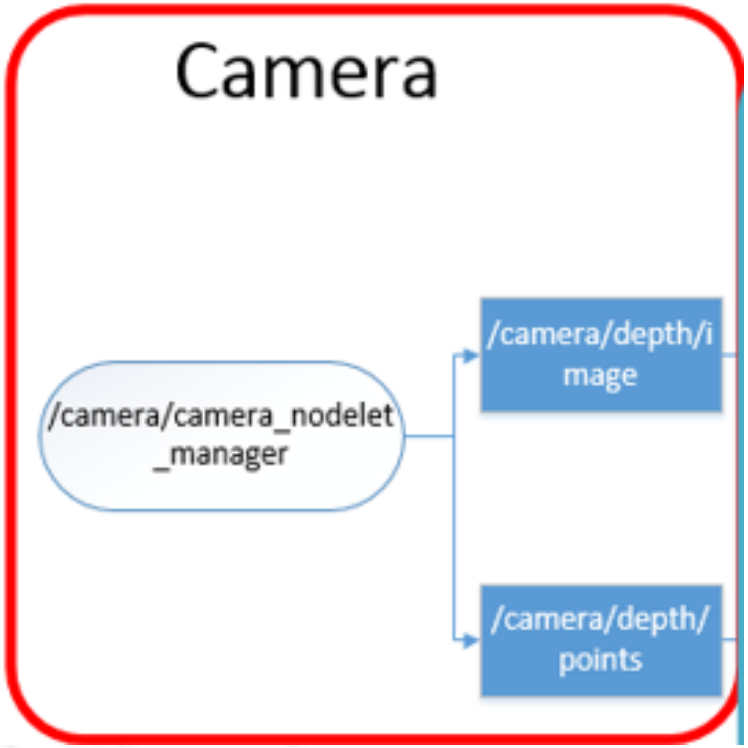
Camera rqt_image_view

查看Camera 输出流：

$ rqt_image_view

在下拉框中，选择/camera/depth/image

没有/camera/depth/points ?



Camera rviz

查看Camera 输出流：

没有/camera/depth/points ?

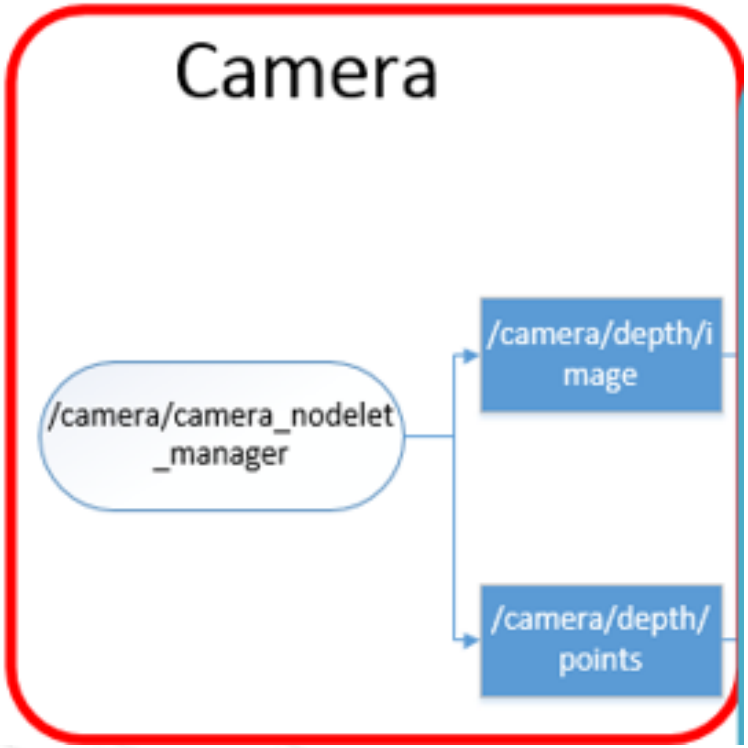
启动rviz：

$ rviz

修改 Fixed Frame 为camera_link

添加pointcloud2

选择 /camera/depth/points topic 主题



Camera 多机分享（实践）

（设置虚拟机为桥接模式，启动ubuntu）
首先，关闭你所有的终端（Terminal）

关闭 所有 roscore 和roslaunch 启动的窗口 （ Ctrl +C ）

打开 .bashrc 文件：
$ sudo gedit ~/.bashrc

在.bashrc 文件后面加入ROS_MASTER_URI 和ROS_IP：

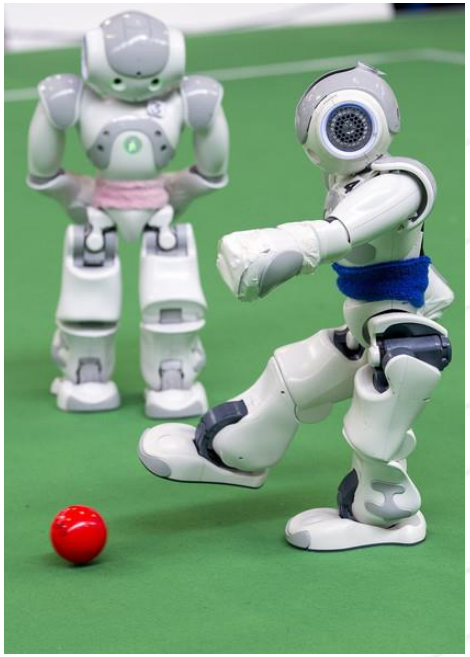
export ROS_MASTER_URI=http://**192.168.1.118**:11311
export ROS_IP=**192.168.1.119 （你的Ubuntu IP，通过ifconfig 查看）**

保存 .bashrc 文件

然后，打开新的Terminal ， 输入：

$ rqt_image_view　　(千万不要启动roscore)

多机通信的应用



Robocup

手机控制

Fetch Robotics

避障算法设计

node：/obstacle_avoidance_node

Subscribed Topic:

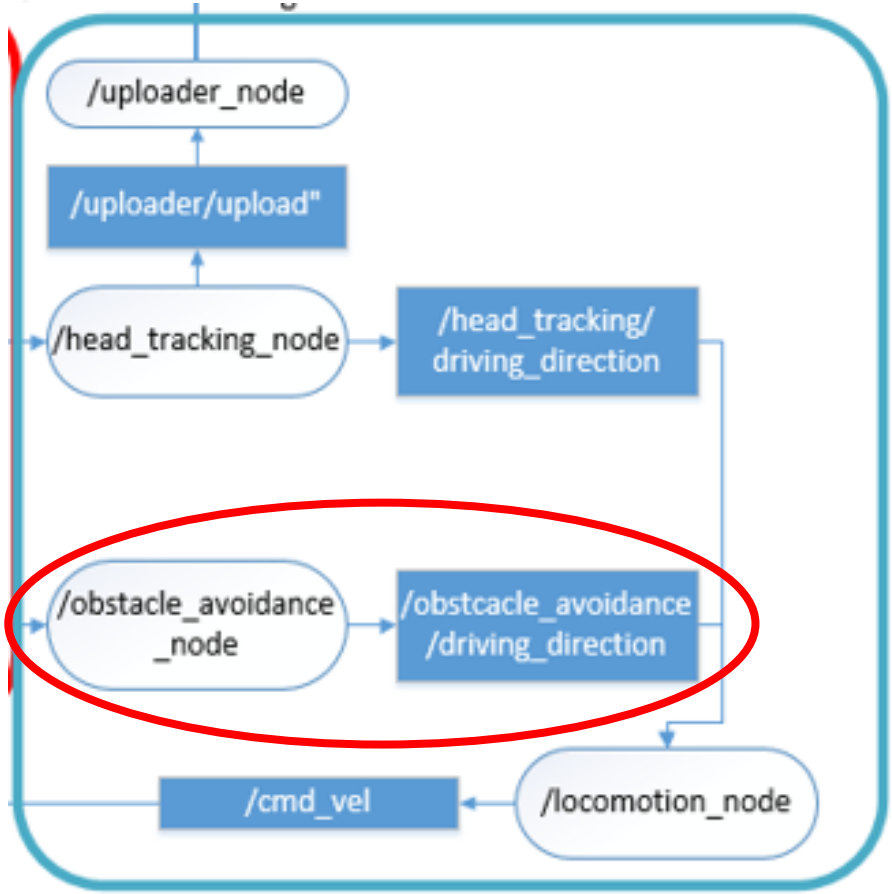
/camera/depth/points

Type: sensor_msgs/PointCloud2

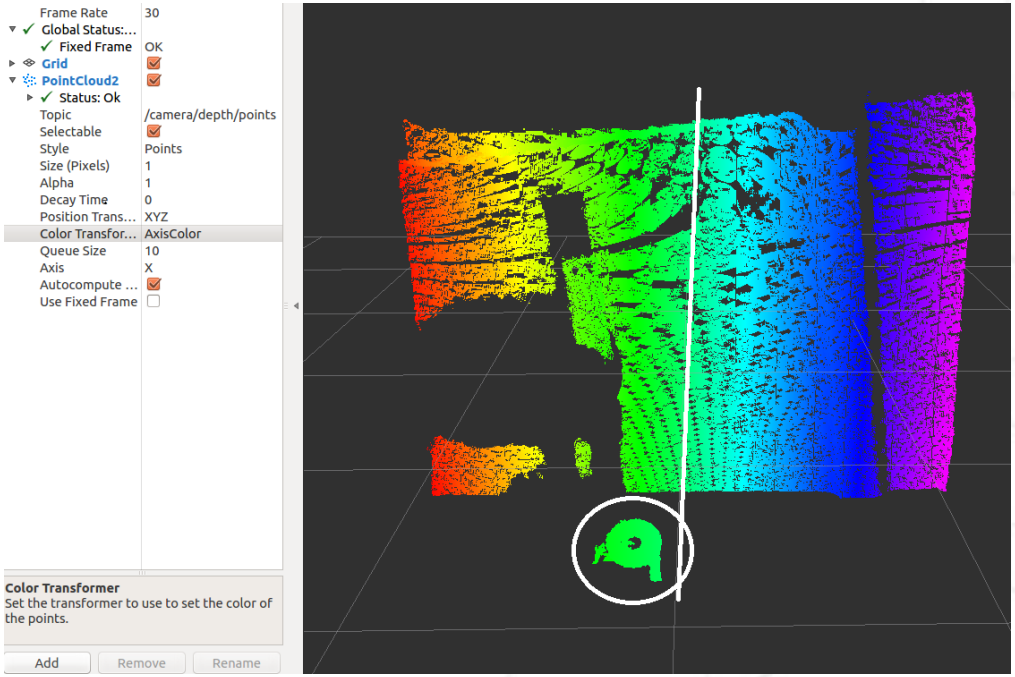
Published Topic:

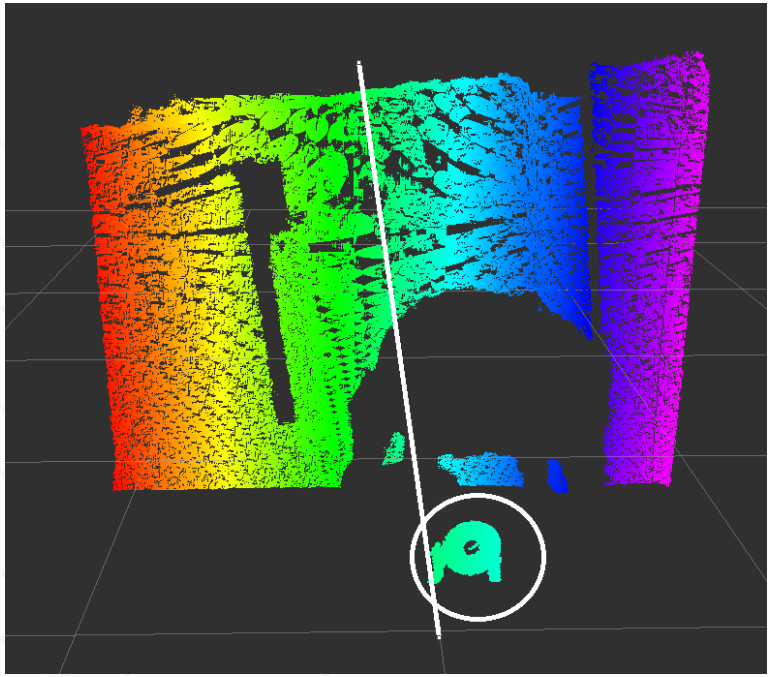
/obstcacle_avoidance/driving_direction

Type: std_msgs/String



避障算法设计


Frame Rate 30
Global Status:...
Fixed Frame OK
Grid
PointCloud2
Status: Ok
Topic /camera/depth/points
Selectable
Style Points
Size (Pixels) 1
Alpha 1
Decay Time 0
Position Trans... XYZ
Color Transfor... AxisColor
Queue Size 10
Axis X
Autocompute ...
Use Fixed Frame
Color Transformer
Set the transformer to use to set the color of the points.
Add
Remove
Rename



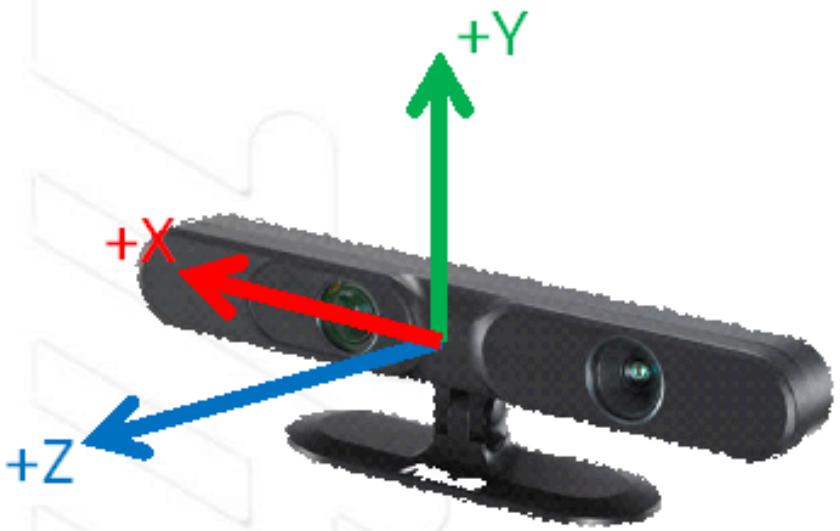
避障算法设计

算法思想：

1. 设定一个ROI区域：
   focus_field_width=0.45;
   focus_field_heigh=1.4;
   focus_field_depth=0.8;

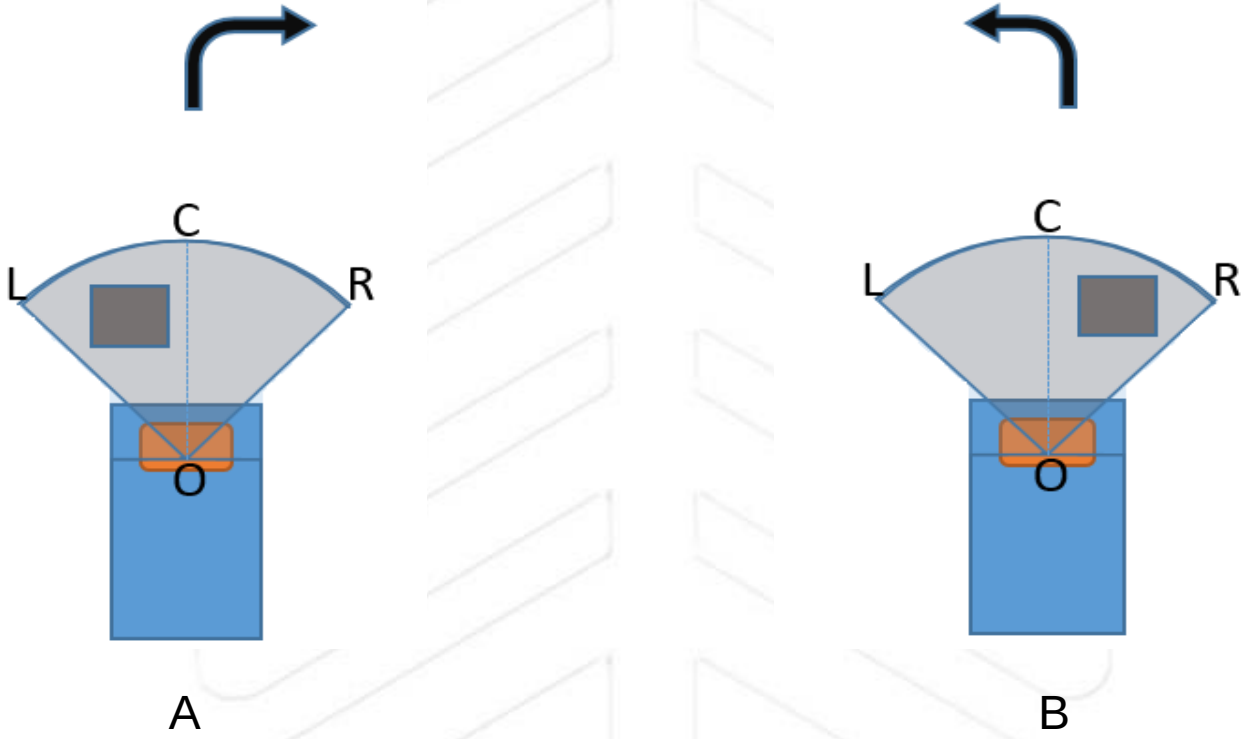
2. 求ROI区域内的质心值；
   centroid_x /= cloud_filtered->size()

3. If cloud_filtered->size() <3000
   直行
   else
   If centroid_x>0 then
   左转
   else
   右转

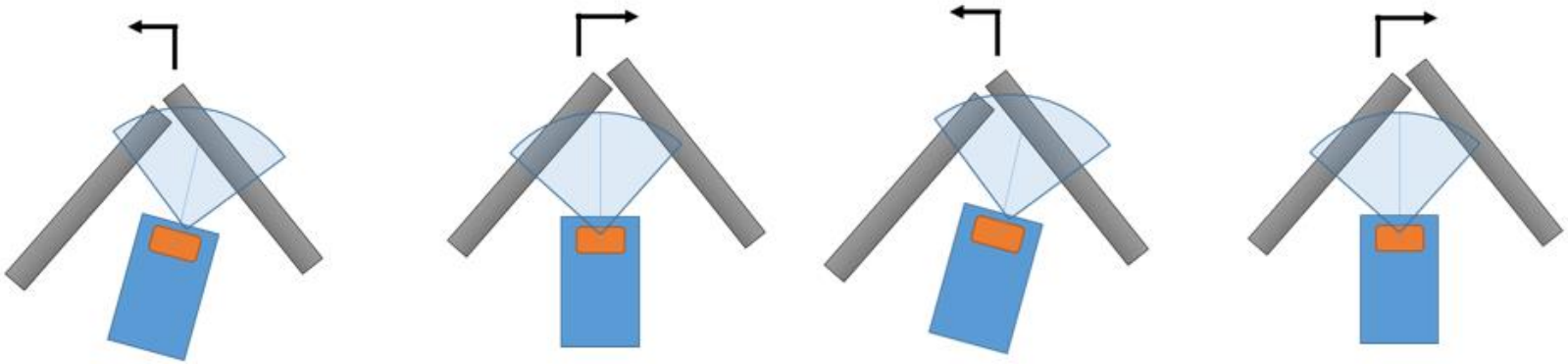


右手定则

避障算法设计

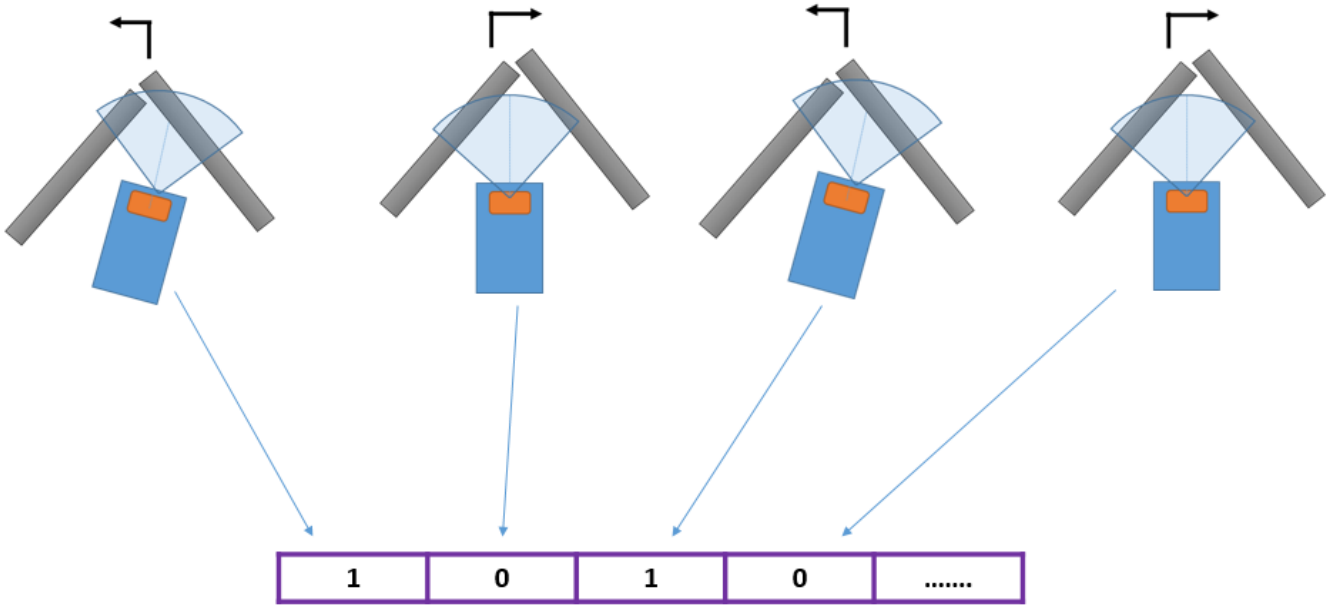


墙角问题



Deadlock

Binary Exponential Turning



Deadlock

Binary Exponential Turning

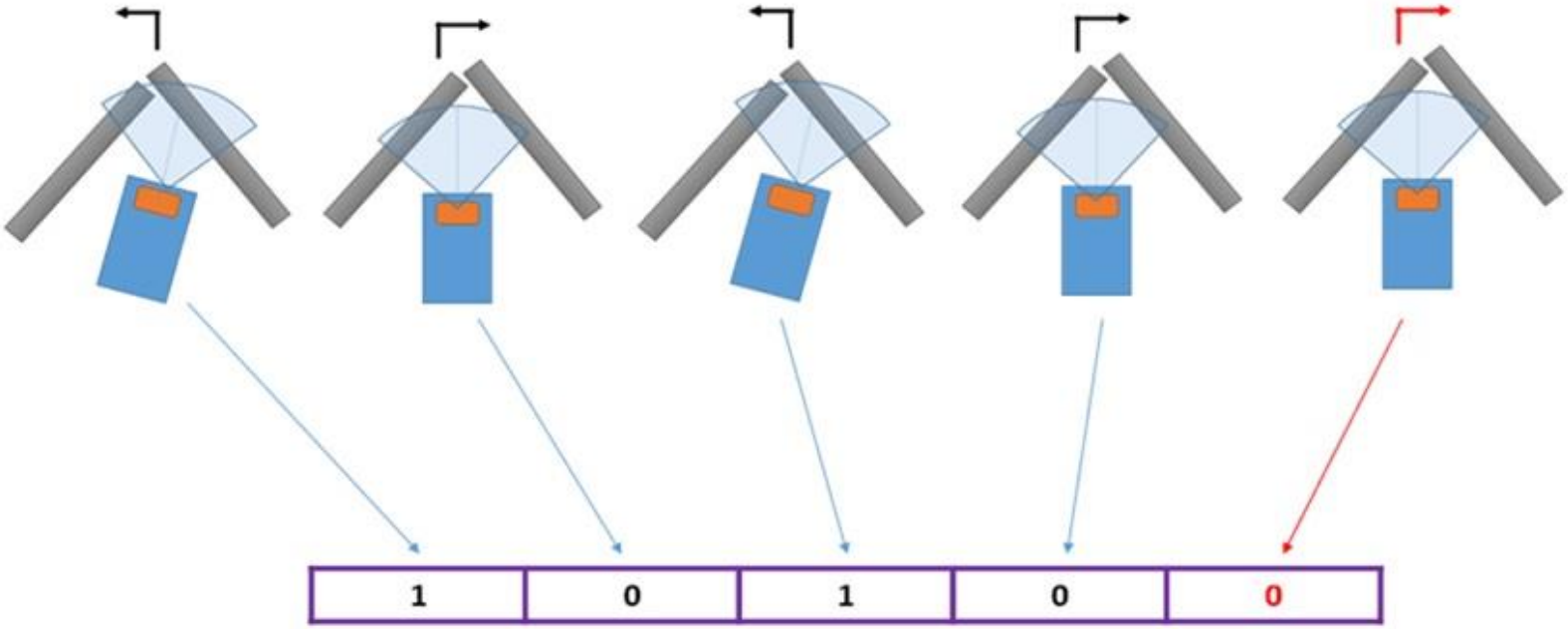
If $\underbrace{1010\ldots}_{n = 1,2,3,\ldots,n}$ then

make 2^n-1 right turns

End if

0101 followed by 1010

Binary Exponential Turning



Deadlock Prevention

Binary Exponential Turning

人脸检测设计

node：/head_tracking_node

Subscribed Topic:

/camera/depth/image

Type: sensor_msgs/Image

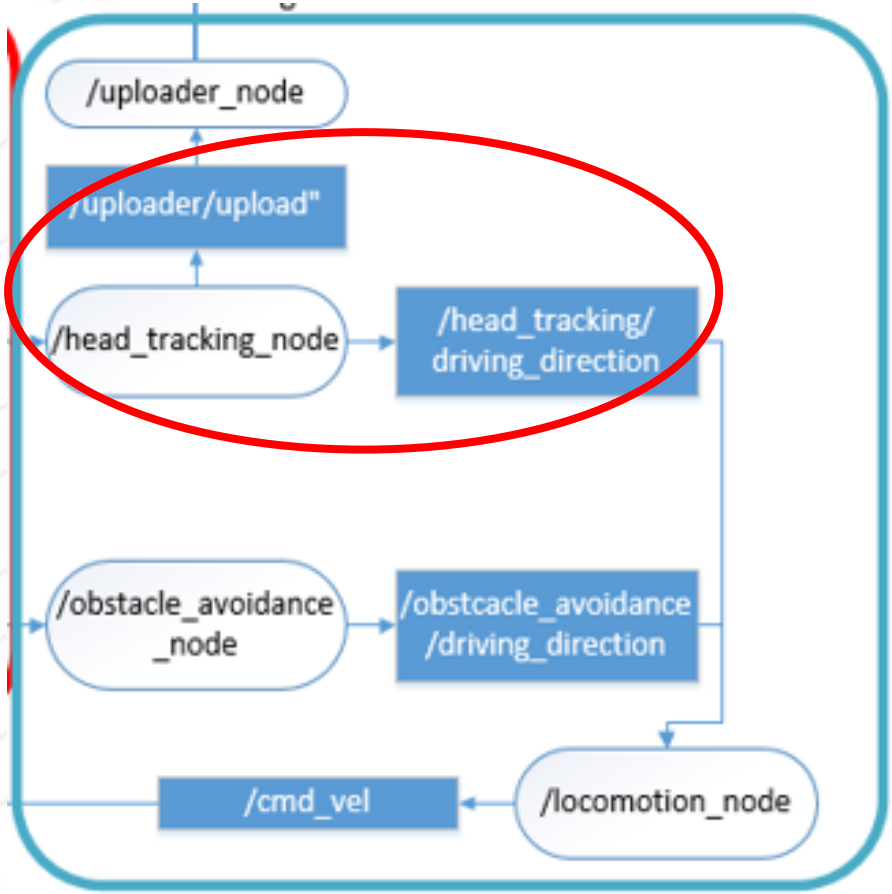
Published Topics:

/head_tracking/driving_direction

Type: std_msgs/String

/uploader/upload

Type: sensor_msgs/Image

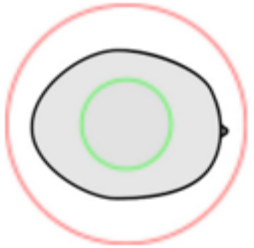


头部检测设计

算法思想：

1. 采用深度图像

2. 设定一个Blob ：
   大小约20 cm ×15 cm ×25 cm（长x宽x高）



头部俯视图

3. 计算局部最小值 和 周围的梯度确定 Blob

4. Blob大小确定是否是头部
   计算Blob大小，必须先把原始深度图像转换到欧几里得空间。

参考 paper： Jens Garstka and Gabriele Peters. **View-dependent 3D Projection using Depth-Image-based Head Tracking.** In Proceedings of the 8th IEEE International Workshop on Projector-Camera Systems, pages 52 - 58, 2011.

头部检测设计

Blob定义： http://wiki.ros.org/cmvision

计算机视觉中的Blob是指图像中的具有相似颜色、纹理等特征所组成的一块连通区域。

简单来说，blob分析就是在一块“光滑”区域内，将出现“灰度突变”的小区域寻找出来。

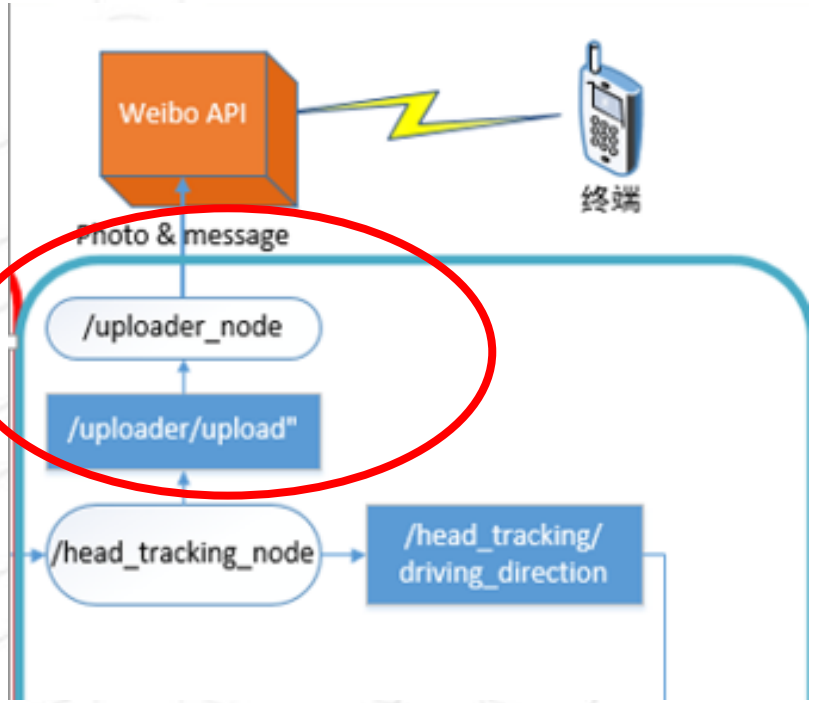
Uploader_node设计

node：/uploader_node

Subscribed Topic:

/uploader/upload

Type: sensor_msgs/Image



Uploader_node设计

设计步骤：

1. 注册新浪微博

2. 申请一个开发者应用
   App Key
   App Secret

3. 安装微博SDK（Ubuntu +python）
   $ sudo pip install sinaweibopy
   $ python
   $import weibo

   如果没有报错，则说明安装成功

   具体教程，请参考我的博客：
   http://blog.sina.com.cn/s/blog_c3d349f60101kobi.html

4. 编写发微博的python代码：

```
import weibo

f = open('/home/exbot/take_pictures/xx.png','rb')

r=client.statuses.upload.post(status=content,pic=f)

f.close()
```

Uploader_node设计

rp_uploader_node.py × uploader.launch ×

```
<launch>
  <node ns="rp" name="uploader_node" pkg="rp_uploader"
type="rp_uploader_node.py" output="screen">
    <param name="sina_api_key" value="[redacted]" type="string"/>
    <param name="sina_api_secret" value="[redacted]"
type="string"/>
    <param name="call_back" value="[redacted]" type="string"/
>
    <param name="access_token" value="[redacted]"
type="string"/>
    <param name="token_secret" value="[redacted]" type="string"/>
    <param name="photo_title" value="Robot Photographer's Picture"
type="string"/>
    <param name="photo_description" value="You have been captured"
type="string"/>
    <param name="photo_tags" value="robot photographer" type="string"/>
  </node>
</launch>
```

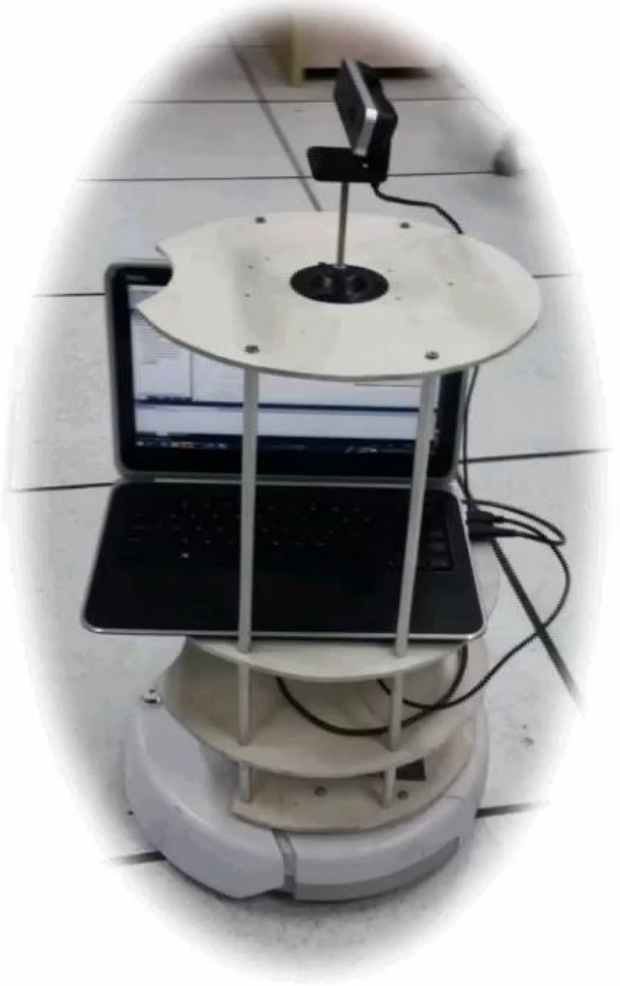
具体教程，请参考我的博客：
http://blog.sina.com.cn/s/blog_c3d349f60101kobi.html

Uploader_node设计

具体教程，请参考我的博客：

http://blog.sina.com.cn/s/blog_c3d349f60101kobi.html

Demo



CONTAIN AUDIO

ROSBOT:
A LOW-COST
AUTONOMOUS
FRIEND FINDER

Thank You

http://www.robotics.sei.ecnu.edu.cn