

- 1. 校准里程计
- 2. 手动校准IMU
- 2. 自动校准IMU
- 3. 验证

## 1. 校准里程计

---

首先按照11. PIBOT的控制及校准校准好里程计

## 2. 手动校准IMU

---

- 1. 启动imu bringup

```
roslaunch pibot_bringup bringup_with_imu.launch
```

- 2. 查看imu数据 `rostopic echo /imu/data_raw`

```
angular velocity:  
  x: -0.00568883214164  
  y: -0.000536621494293  
  z: 0.00121228414775  
angular_velocity_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,  
 0.0, 0.0, 0.0]  
linear_acceleration:  
  x: 0.0336265  
  y: -0.014684  
  z: 9.8876055  
linear_acceleration_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0,  
0.0, 0.0, 0.0]
```

校准使得红色方框中接近于0（除了z轴加速度），图中实际为校准完成的结果

如果相差较大，则需要运行第三步

- 3. 校准

```
rosservice call /imu/calibrate_imu
```

保持小车静止状态，新的窗口运行上面命令，待bringup输出日志即可

```
[ INFO] [1533733592.644998817]: Imu sensor not active any more  
[ INFO] [1533733597.252551972]: Calibrating accelerometer and gyroscope complete.  
[ INFO] [1533733597.252667140]: Bias values can be saved for reuse.  
[ INFO] [1533733597.252748398]: Accelerometer: x: 0.027488, y: 0.022307, z: -0.361297  
[ INFO] [1533733597.252823979]: Gyroscope: x: -0.039662, y: 0.080867, z: 0.001414  
[ INFO] [1533733597.265483686]: Initializing Imu sensor  
[ INFO] [1533733597.298700038]: Imu sensor activated
```

- 4. 更新校准数据

```
roscd pibot_bringup/launch
vi bring_with_imu.launch
```

把bringup校准后的输出（3中图片红色部分）的数据填入的bring\_with\_imu.launch文件中即可

```
<launch>
  <node name="pibot_driver" pkg="pibot_bringup" type="pibot_driver" output="screen">
    <roscparam file="$(find pibot_bringup)/params/base_params_with_imu.yaml" command="load" />
  </node>

  <node pkg="pibot_imu" type="pibot_imu" name="pibot_imu" output="screen" respawn="false">
    <roscparam>
      imu/accelerometer_bias: {x: 0.027488, y: 0.022307, z: -0.361297}
      imu/gyroscope_bias: {x: -0.039622, y: 0.080867, z: 0.001414}
    </roscparam>
    <param name="imu/perform_calibration" value="false" />
  </node>

  <node pkg="tf" type="static_transform_publisher" name="base_imu_to_base_link"
    args="0 0.0 0 0 0.0 0.0 /base_link /imu_link 40" />
```

## 2. 自动校准IMU

支持自动校准IMU，只需要设置imu/perform\_calibration为ture即可，执行roslaunch pibot\_bringup bringup\_with\_imu.launch后会有如下输出

```
[ INFO] [1455208231.671303289]: Odom sensor activated
[ INFO] [1455208231.679188115]: Kalman filter initialized with odom measurement
[ INFO] [1455208231.737567]: Publishing combined odometry on
[ WARN] [1455208231.765133995]: Calibrating accelerometer and gyroscope, make sure robot is s
tationary and level.
```

自动校准需要一些数据作为输入，这时候需要保持imu静止平放，直到输出如下信息校准完成

```
[ INFO] [1455208231.757567]: Publishing combined odometry on
[ WARN] [1455208231.765133995]: Calibrating accelerometer and gyroscope, make sure robot is s
tationary and level.
[ INFO] [1455208236.824720661]: Calibrating accelerometer and gyroscope complete.
[ INFO] [1455208236.824843452]: Bias values can be saved for reuse.
[ INFO] [1455208236.824889243]: Accelerometer: x: 0.623448, y: -0.171533, z: -1.674385
[ INFO] [1455208236.824927159]: Gyroscope: x: -0.099189, y: -0.061359, z: 0.052344
[ INFO] [1455208236.849414334]: First IMU message received.
[ INFO] [1455208236.852046012]: Initializing Imu sensor
[ INFO] [1455208236.877941629]: Imu sensor activated
```

## 3. 验证

重启bringup和输出imu数据（前面1和2）

```
angular_velocity:
  x: -0.000444689252853
  y: -0.000733621494293
  z: -0.000199857926123
angular_velocity_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
linear_acceleration:
  x: 0.0115745
  y: 0.0167555
  z: 9.814422
linear_acceleration_covariance: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
```

可以看到得到一个更

加接近于0的一组数据

