

FUNDAÇÃO EDSON QUEIROZ

UNIVERSIDADE DE FORTALEZA

Centro de Ciências Tecnológicas – CCT

Disciplina – Sistemas Autônomos

Lidar

G. A. Davi.

Resumo

O intuito desse projeto é comparar a eficácia de algoritmos de localização e mapeamento aplicados ao sensor laser Hokuyo 30LX.

Introdução

Objetivos:

Comparar algoritmos de localização e mapeamento

Materiais e Métodos

A máquina usada foi:

OS: MacOS High Sierra 10.13.6

PROCESSOR: 2,2 GHz intel core i7

RAM: 16 GB 1600 MHz DDR3

GRAPHICS: Intel Iris Pro 1536 MB

ROM: 200 GB

Para tal, foi adicionado uma partição ao disco de 50 GB com o sistema operacional Ubuntu 18.04, no qual foi instalado os softwares necessários.

Sensor lidar – Hokuyo 30lx

Matlab

Baixar o Linux Ubuntu 18.04 https://www.ubuntu.com/download/desktop /thank-you?country=BR&version=18.04.1&architecture=amd64

Preparar o Dual Boot e instalar o Ubuntu https://www.maketecheasier.com/ install-dual-boot-ubuntu-mac/ (Caso encontre problemas com o particionamento do disco, leia resolução de problemas no anexo)

Instalar ROS Melodic Desktop-Full Install http://wiki.ros.org/melodic/Installation /Ubuntu

Instalar Gazebo http://gazebosim.org/tutorials?tut=ros\_installing

Baixar Laser Hokuyo - Driver Hokuyo Node https://github.com/ros-drivers/hokuyo node

Configurar o WorkSpace http://wiki.ros.org/catkin/Tutorials/create\_a \_workspace

Para configurar o Laser Hokuyo 30LX, siga o tutorial abaixo (retirado de https://github.com/afonsohfontes/Jaguar/blob/master/Relatorio.pdf em 28 de agosto de 2018):

Obs: é possível que seja necessário a instalação de outras bibliotecas. Caso isso seja verdadeiro, procure a solução em correção de erros, no anexo.

# Após o workspace ter sido criado, copie a pasta do Hokuyo Node e cole dentro da pasta “src” do seu novo workspace. Abra o terminal e entre na do diretório raiz do seu workspace e configura o driver executando os comandos:

# cd ∼/home/user/WORKSPACE (exemplo)

# Configurar driver e instalar dependências do mesmo

# Catkin make

# Se a maquina não tive processamento e memória suficientes, essa instalação pode travar. Então execute:

# Catkin make –j3

Retirado de: https://github.com/afonsohfontes/Jaguar/blob/master/Relatorio.pdf, página 9.

Antes de executar qualquer linha de comando, lembre-se sempre de executar o comando “source setup.bash” para cada novo terminal (como executado nos tutoriais e discutido aqui: <https://answers.ros.org/question/206876/how-often-do-i-need-to-source-setupbash/>)

Execute o ROS e o Lidar (retirado de https://github.com/afonsohfontes /Jaguar/blob/master/Relatorio.pdf em 28 de agosto de 2018)

# EXECUÇÃO DO ROS E LASER

# Após ser realizado esses passos o driver do laser HOKUYO estará instalado corretamente no ROS. Para executar o ROS execute os seguintes comandos no TERMINAL:

# Rodar o ROS MASTER

# **Roscore**

# Executar driver LASER - Abra outro terminal e execute o comando:

# **Rosrun hokuyo node hokuyo node**

# Lembre que o laser deve estar conectado via usb com o computador. Para conferir se o driver executou corretamente é necessario abrir outro terminal e executar o comando do ROS para listar todos os tópicos que estão sendo publicados, que correspondem a todos dados de sensores que estão sendo executados em tempo real. Comando:

# **Rostopic list**

# No terminal deverá aparecer os dados semelhantes ao da imagem abaixo:

# Procure um tópico com /scan ou um nome semelhante a “scan”, que corresponderá

# aos dados do laser. Em seguida abra outro terminal, e execute o seguinte comando:

# **Rostopic echo “.../scan”**

# Os dados do laser deverão aparecer na tela de acordo como mostrado na imagem

# abaixo.

# Estes dados estão dispostos em vetores e correspondem as distancias calculadas por cada ponto de infravermelho emitidos no ambiente pelo laser HOKUYO. Com o laser funcionando iremos visualizar os dados recebidos utilizando um software que já é instalado nativamente com o ROS. É um software de visualização, compatível com vários tipos de dados referentes a aplicações de robótica e muito utilizado no desenvolvimento dessas aplicações por possibilitar a visualização de forma abstraída de toda aplicação robótica em funcionamento, como também um mapeamento sendo realizado e a rota do robô pelo ambiente.

# Para executar o Rviz execute o seguinte comando:

# **Rosrun rviz rviz**

# Uma interface gráfica irá ser executada. Para estudar essa interface e aprender de forma completa o manuseio da mesma. Acesse o link: http://wiki.ros.org/rviz/UserGuide

# Na side bar do lado esquerdo, no canto inferior clique no botão “ADD” e procure o tópico /scan na lista, na janela que ira se abrir. Após adicionar o tópico /scan marque a check box na side bar. E você deverá visualizar os dados do laser.

Retirado de https://github.com/afonsohfontes /Jaguar/blob/master/Relatorio.pdf página 8 e 9.

Baixar MRPT – Navegationque pode ser encontrado aqui: <https://github.com/mrpt-ros-pkg/mrpt_navigation>, colar a pasta no src do seu workspace e, por fim, executar & catkin\_make, da mesma forma feita para instalar o package Hokuyo\_node.

<http://wiki.ros.org/mrpt_localization>

Para o erro

para adicioanr o map

Usage: static\_transform\_publisher x y z qx qy qz qw frame\_id child\_frame\_id period(milliseconds)

rosrun tf static\_transform\_publisher 0.0 0.0 0.0 0.0 0.0 0.0 map laser 100

Desligou o pc e não sabe como reabrir tudo?? Siga o passo a passo abaixo:

1. Novo terminal
2. $ roscore (ativar o roscore)
3. Novo terminal
4. $ cd <seu\_workspace> (exemplo: $ cd catkin\_ws)
5. $ source devel/setup.bash (Por garantia, faça isso em todo novo terminal)
6. $ rosrun hokuyo\_node hokuyo\_node (iniciar o sensor)
7. Novo terminal
8. $ source devel/setup.bash (Por garantia, faça isso em todo novo terminal)
9. rostopic echo /scan (Verifica se você realmente está recebendo os dados do sensor)
10. $ rosrun tf static\_transform\_publisher 0.0 0.0 0.0 0.0 0.0 0.0 map laser 100 (se n fizer isso, ao abrir o rviz, aparecera um erro ao mapa. entao devemos adicionar nosso mapa laser, algo no fixed frame. Usage: static\_transform\_publisher x y z qx qy qz qw frame\_id child\_frame\_id period(milliseconds) ) Para mais informações: http://wiki.ros.org/tf
11. $ roslaunch mrpt\_localization demo.launch (iniciar a demo)

1 de setembro de 2018: Encontrei um erro no “map” da demo que ainda não consegui resolver, fazendo com que a mesma não opere como o esperado. Espero resolve-lo em breve.

A demo pode ser encontrada em: https://github.com/mrpt-ros-pkg/mrpt\_navigation/blob/master/mrpt\_localization/launch/demo.launch  
  
https://answers.ros.org/question/209276/rviz-slam-rplidar-problem/

<https://answers.ros.org/question/288945/rviz-errortransform-senderunknown_publisher-for-frame-x-frame-x-does-not-exist/>

<http://wiki.ros.org/mrpt_localization>

http://wiki.ros.org/mrpt\_navigation/Tutorials/Installing

Instalar o ros (core, para visualizar os dados, e transmitir via red)

Ler o Sensor

Monitorar e aplicar o slam a partir de um outro computador (com matlab e visualizar, aplicar o slam, no sensor dando)

Implementar via ros algoritmo de comparação entre dois algoritmos

Bibliografia

Evangelista, D.M. e Silva, R.F. e Pinto, I.R.S e AZEVEDO, E.A. Recuperação de funcionamento do Dr Robot Jaguar, Pages 5-9 Agust 2018, Pages 1342-1352. Disponível em: https://github.com/afonsohfontes/Jaguar/blob/master/Relatorio.pdf

Dual boot on mac: <https://www.maketecheasier.com/install-dual-boot-ubuntu-mac/>

Resolução de problemas:

Particionamento do disco:

Ao tentar fazer o dual boot, não conseguia adicionar uma partição ao disco. Mesmo tendo bastante espaço para o mesmo, o sistema acusade de não haver espaço suficiente. O problema foi gerado pelo Time Machine Snapshot, responsável pelo backup da máquina. Esse problema foi solucionado com a remoção dos Snapshots.

A solução encontra-se em: https://www.macworld.com/article/3260635/macs/how-to-delete-time-machine-snapshots-on-your-mac.html

Não consegui instalar no próprio mac.

Instalação do Hokuyo Node:

Caso seja lançado uma mensagem de erro, durante a execução do catkin\_make, dizendo que faltou arquivos ou packages, você deve adicionar outros packages na pasta src do seu workspace (a mesma pasta do Hokuyo\_node)

Para solucionar, vc deve fazer:

1. Retirar todos os packages da pasta src
2. Adicionar o package driver\_common na pasta src, este você pode encontrar em: https://github.com/ros-drivers/driver\_common.git
3. Executar $ catkin\_make no terminal no diretório do seu workspace
4. Adicionar o package Hokuyo\_node na pasta src
5. Executar catkin\_make no terminal no diretório do seu workspace

https://answers.ros.org/question/240235/how-to-install-packages-from-github/

A idéia, como pode-se notar, é executar o catkin\_make toda vez que um novo pacote for adicionado ao workspace. Além disso, devemos também adicionar somente um pacote por vez, para evitar problemas. Outros pacotes que podem ser requistados:

<https://github.com/ros-drivers/urg_c>

<https://github.com/ros-drivers/urg_node>

Erro ao executar Hokuyo\_node (failed to open port – No authorized):

Caso haja problema ao executar

$ rosrun hokuyo\_node hokuyo\_node

Por não ter autorização para acessar a porta usb, você deve ir para a raiz do sistema:

$ cd

e executar:

$ sudo chmod a+rw /dev/ttyACM0

add o driver\_common

catkin\_mak

dps add hokuyo

catkin\_make

se algo n funcionar mais, refazer o workspace

Comados interessantes ao usar o terminal:

para mostrar todas a funcoes associadas ao nome escrito no terminal, nome + tab (x3

) + y

Cancelar uma execução: control + c (x2)