6. gmapping mapping algorithm

Gmapping: http://wiki.ros.org/gmapping/

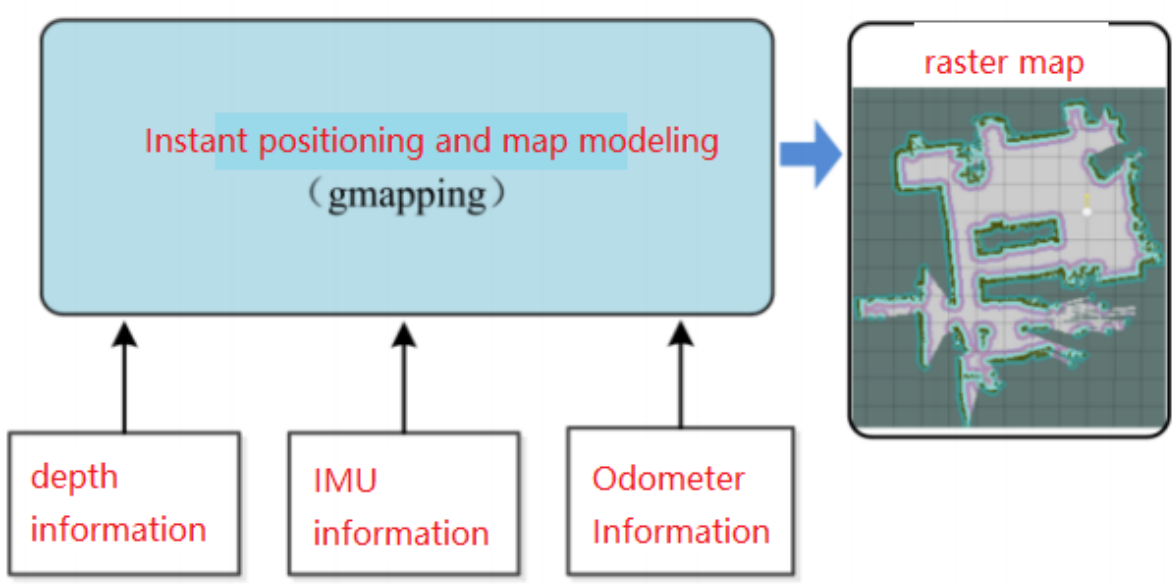
map_server: https://wiki.ros.org/map_server

6.1. Introduction

- gmapping is only applicable to points where the number of two-dimensional laser points in a single frame is less than 1440. If the number of laser points in a single frame is greater than 1440, then problems such as [[mapping-4] process has died] will occur.
- Gmapping is a commonly used open source SLAM algorithm based on the filtered SLAM framework.
- Gmapping is based on the RBpf particle filter algorithm, which separates the real-time positioning and mapping processes. Positioning is performed first and then mapping is performed.
- Gmapping has made two major improvements on the RBpf algorithm: improved proposal distribution and selective resampling.

Advantages: Gmapping can construct indoor maps in real time. The amount of calculation required to construct small scene maps is small and the accuracy is high.

Disadvantages: As the scene grows, the number of particles required increases because each particle carries a map, so the amount of memory and computation required when building a large map increases. Therefore it is not suitable for building large scene maps. And there is no loop detection, so the map may be misaligned when the loop is closed. Although increasing the number of particles can close the map, it comes at the expense of increased calculations and memory.



6.2. Use

Note: When building a map, the slower the speed, the better the effect (note that the rotation speed should be slower). If the speed is too fast, the effect will be poor.

According to different models, you only need to set the purchased model in [.bashrc], X1 (normal four-wheel drive) X3 (Mailun) Take X3 as an example

```
#Raspberry Pi 5 master needs to enter docker first, please perform this step
#If running the script into docker fails, please refer to ROS/07, Docker tutorial
~/run_docker.sh
```

Open the [.bashrc] file

```
sudo vim .bashrc
```

Find the [ROBOT_TYPE] parameters and modify the corresponding car model

```
export ROBOT_TYPE=X3 # ROBOT_TYPE: X1 X3 X3plus R2 X7
```

6.2.1. Start

Start the command (robot side). For the convenience of operation, this section takes [mono + laser + yahboomcar] as an example.

```
roslaunch yahboomcar_nav laser_bringup.launch # laser + yahboomcar
roslaunch yahboomcar_nav laser_usb_bringup.launch # mono + laser + yahboomcar
roslaunch yahboomcar_nav laser_astrapro_bringup.launch # Astra + laser +
yahboomcar
```

<PI5 needs to open another terminal to enter the same docker container

1. In the above steps, a docker container has been opened. You can open another terminal on the host (car) to view:

```
docker ps -a
```

```
jetson@ubuntu:~$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND       CREATED      STATUS       PORTS     NAMES
5b698ea10535   yahboomtechnology/ros-foxy:3.3.9   "/bin/bash"   3 days ago   Up 9 hours             ecstatic_lewin
jetson@ubuntu:~$
```

2. Now enter the docker container in the newly opened terminal:

```
docker exec -it 5b698ea10535 /bin/bash
```

```
jetson@ubuntu:~$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND       CREATED      STATUS       PORTS     NAMES
5b698ea10535   yahboomtechnology/ros-foxy:3.3.9   "/bin/bash"   3 days ago   Up 9 hours             ecstatic_lewin
jetson@ubuntu:~$ docker exec -it 5b698ea10535 /bin/bash
-------------------------------------------------------
my_robot_type: x3 | my_lidar: a1 | my_camera: astrapro
-------------------------------------------------------
root@ubuntu:/#
```

After successfully entering the container, you can open countless terminals to enter the container.

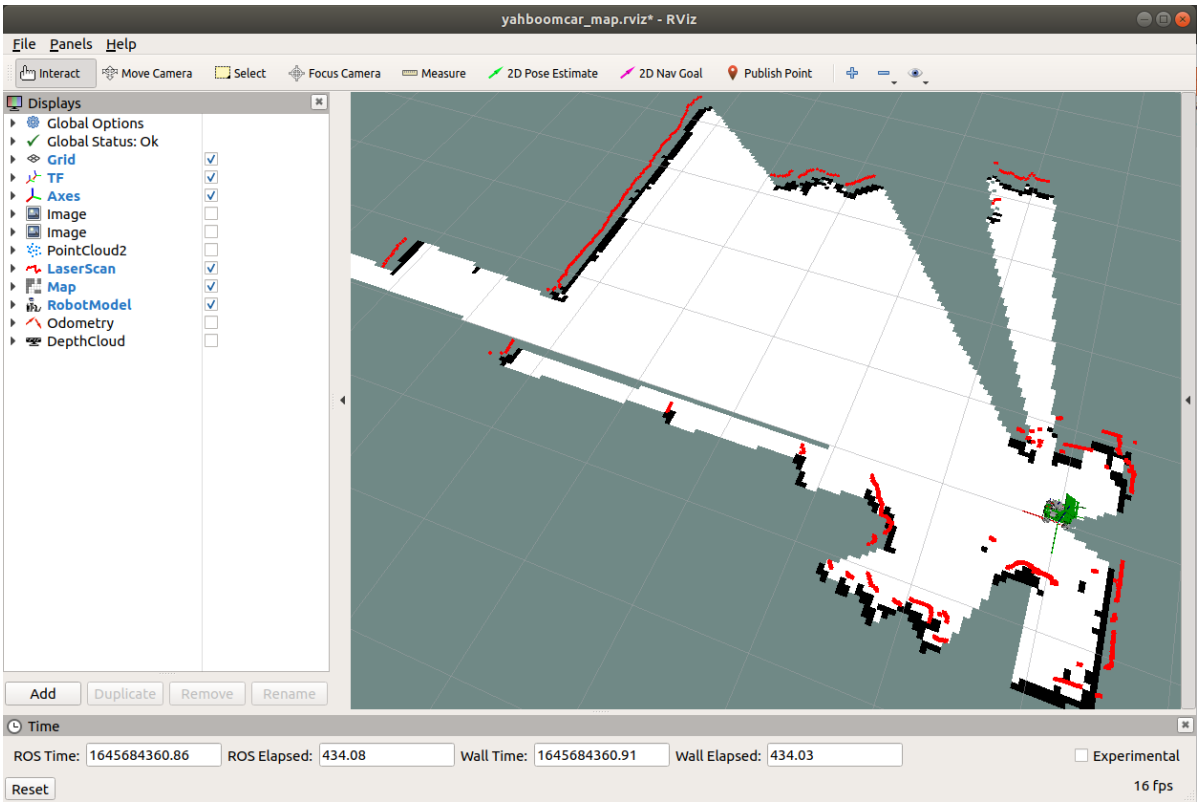
Mapping command (robot side)

```
roslaunch yahboomcar_nav yahboomcar_map.launch use_rviz:=false map_type:=gmapping
```

- [use_rviz] parameter: whether to enable rviz visualization.
- [map_type] parameter: Set the mapping algorithm [gmapping].

Turn on the visual interface (virtual machine side)

```
roslaunch yahboomcar_nav view_map.launch
```



The gap at the back of the robot is due to the obstruction caused by the installation position of the display screen, so a certain range of radar data is blocked. The shielding range can be adjusted, or it can not be blocked according to the actual situation. For specific operations, see [01. Radar Basic Course].

6.2.2. Controlling the robot

- Keyboard controls robot movement

```
rosrun teleop_twist_keyboard teleop_twist_keyboard.py # System integration
roslaunch yahboomcar_ctrl yahboom_keyboard.launch # Custom
```

- Control the robot movement with the handle

Make the robot cover the area to be mapped and the map should be as closed as possible.

There may be some scattered points during the mapping process. If the mapping environment is well closed, relatively regular, and the movement is slow, the scattering phenomenon will be much smaller.

6.2.3. Map saving

```
rosrun map_server map_saver -f ~/yahboomcar_ws/src/yahboomcar_nav/maps/my_map # The first way
bash ~/yahboomcar_ws/src/yahboomcar_nav/maps/map.sh # The second way
```

The map will be saved to the ~/yahboomcar_ws/src/yahboomcar_nav/maps/ folder, a pgm image and a yaml file.

map.yaml

```
image: map.pgm
resolution: 0.05
origin: [-15.4,-12.2,0.0]
Negate: 0
occupied_thresh: 0.65
free_thresh: 0.196
```

Parameter analysis:

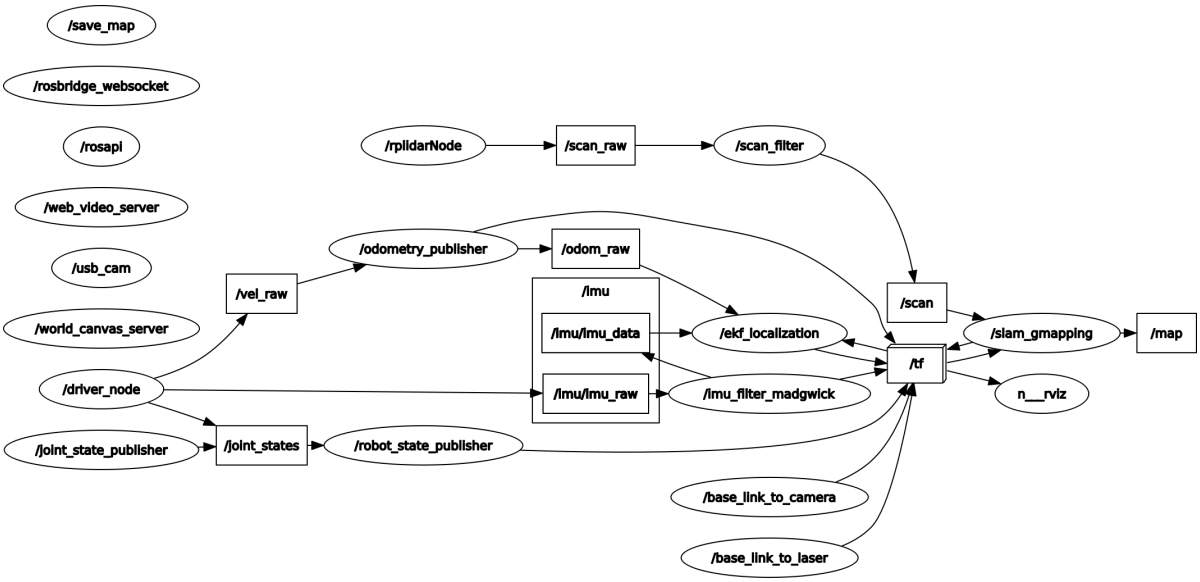
- image: The path of the map file, which can be an absolute path or a relative path.
- resolution: resolution of the map, meters/pixel
- Origin: 2D pose (x, y, yaw) in the lower left corner of the map. The yaw here is rotated counterclockwise (yaw=0 means no rotation). Many parts of the current system ignore the yaw value.
- negate: whether to reverse the meaning of white/black and free/occupied (the interpretation of the threshold is not affected)
- occupied_thresh: Pixels with an occupation probability greater than this threshold will be considered fully occupied.
- free_thresh: Pixels with occupancy probability less than this threshold will be considered completely free.

6.2. Topics and services

Subscription topic	Type	Description
tf	tf/tfMessage	Used for conversion between lidar coordinate system, base coordinate system, and odometer coordinate system
scan	sensor_msgs/LaserScan	Lidar scan data
Post Topic	**Type**	**Description**
map_metadata	nav_msgs/MapMetaData	Publish map Metadata
map	nav_msgs/OccupancyGrid	Publish map raster data
~entropy	std_msgs/Float64	Publish an estimate of the entropy of the robot pose distribution
Service	**Type**	**Description**
dynamic_map	nav_msgs/GetMap	Get map data

Node view

```
rqt_graph
```



6.3. Configuration parameters

Parameters	Type	Default value	Description
~throttle_scans	int	1	Each time this number of frames of laser data is received, only one frame of data is processed. By default, each frame of data is processed

Parameters	Type	Default value	Description
~base_frame	string	"base_link"	Robot base coordinate system
~map_frame	string	"map"	Map coordinate system
~odom_frame	string	"odom"	Odometer coordinate system
~map_update_interval	float	5.0	Map update frequency, the lower the value, the greater the computational load
~maxUrange	float	80.0	The maximum range that the laser can detect
~sigma	float	0.05	standard deviation of endpoint matches
~kernelSize	int	1	Search in the corresponding kernel
~lstep	float	0.05	Optimization step size during translation
~astep	float	0.05	Optimization step size during rotation
~iterations	int	5	Number of iterations to scan for matches
~lsigma	float	0.075	Laser standard deviation of likelihood calculation
~ogain	float	3.0	Used to smooth the resampling effect during likelihood calculation
~lskip	int	0	Number of beams skipped per scan
~minimumScore	float	0	The minimum value of scan matching results. When using a laser scanner with a limited range (e.g. 5m), jumping in large open spaces can be avoided
~srr	float	0.1	Translation function (rho/rho), odometry error during translation
~srt	float	0.2	rotation function (rho/theta), odometry error in translation
~str	float	0.1	Translation function (theta/rho), odometry error when rotating
~stt	float	0.2	Rotation function (theta/theta), odometry error when rotating

Parameters	Type	Default value	Description
~linearUpdate	float	1.0	Process the laser scanning data every time the robot translates this distance
~angularUpdate	float	0.5	The robot processes laser scanning data every time it rotates this distance
~temporalUpdate	float	-1.0	Process a scan if the latest scan is processed slower than the update. Turn off time-based updates when this value is negative
~resampleThreshold	float	0.5	Neff-based resampling threshold
~particles	int	30	Number of particles in the filter
~xmin	float	-100.0	Initial minimum size of map in x direction
~ymin	float	-100.0	The initial minimum size of the map in the y direction
~xmax	float	100.0	The initial maximum size of the map in the x direction
~ymax	float	100.0	The initial maximum size of the map in the y direction
~delta	float	0.05	map resolution
~llsamplerange	float	0.01	Translation sampling distance for likelihood calculation
~llsamplestep	float	0.01	translation sampling step for likelihood calculation
~lasamplerange	float	0.005	Rotated sampling distance for likelihood calculation
~lasamplestep	float	0.005	Rotation sampling step size for likelihood calculation
~transform_publish_period	float	0.05	TF transform publishing time interval
~occ_threh	float	0.25	Threshold for raster map occupancy
~maxRange(float)	float	-	The maximum range of the sensor

6.4, TF transformation

Required TF transformation	**Description**
laser-->base_link	The transformation between the laser radar coordinate system and the base coordinate system is generally published by robot_state_publisher or static_transform_publisher
base_link-->odom	Transformation between the map coordinate system and the robot odometer coordinate system, estimating the robot's pose in the map
Released TF Transform	**Description**
map-->odom	Transformation between the map coordinate system and the robot's odometry coordinate system, estimating the robot's pose in the map

View tf tree

```
rosrun rqt_tf_tree rqt_tf_tree
```