

SIMULASI OPENSAM MENGGUNAKAN TURTLEBOT3 PADA ROS

OPENSAM SIMULATION ON ROS WITH TURTLEBOT3

Zaikiy Ni Starry¹, Agung Nugroho Jati², Faisal Candrasyah Hasibuan³^{1,2,3} Universitas Telkom, Bandungzaikiy@students.telkomuniversity.ac.id¹, agungnj@telkomuniversity.ac.id²,
faicanhasfcb@telkomuniversity.ac.id³**Abstrak**

Pemetaan pada ruang tertutup yang tidak diketahui apakah di dalamnya terdapat zat berbahaya atau tidak tanpa alat bantu dengan instrumen yang tepat sangat membahayakan. Terlebih jika manusia yang melakukan pengecekan tanpa alat bantu tambahan. Oleh karena itu, dibuat alat pemetaan ruang dan eksplorasi dengan menggunakan turtlebot3 burger untuk memudahkan manusia dalam melakukan pemetaan ruang.

Simulasi pendeteksian dan pemetaan pada ruang tertutup dilakukan dengan menggunakan turtlebot3 burger virtual pada ROS. Simulasi dilakukan dengan menentukan lokasi map yang belum diketahui sebelumnya. Kemudian setelah dilakukannya simulasi, langkah selanjutnya merealisasikannya juga dengan menggunakan turtlebot3 burger pada ruang tertutup untuk menghasilkan mapping.

Simulasi dan penelitian ini bertujuan untuk mengetahui lokasi yang belum diketahui dan melakukan pemetaan pada ruang tertutup dengan menggunakan teknik frontier dan openSLAM. Didapatkan hasil mapping lokasi dengan durasi 208.98 ROS Time, pada posisi awal robot (x,y)=(-1.5,-1.35) sampai dengan posisi akhir robot (x,y)=(5.7,-2.9).

Kata kunci : frontier exploration, gazebo, openSLAM.

Abstract

Without the use of the proper instrument, mapping in a closed environment where it is unknown whether there are hazardous compounds present is unsafe. It's better if the checking is done by humans without the use of any additional instruments. As a result, the turtlebot3 burger was used to construct a space mapping and exploration tool to make it easier for people to navigate space.

On ROS, a virtual turtlebot3 burger was used to perform detection and closure simulations. The simulation is carried out by constructing a map of an unknown site. After watching the simulation, the next step is to make it happen by generating a mapping with a turtlebot3 burger in a closed space.

The goal of this simulation and research is to use frontier and openSLAM approaches to locate an unknown place and map it in a closed environment. From the initial position of the robot (x,y)=(-1.5,-1.35) to the final position of the robot (x,y)=(5.7,-2.9), the results of the location mapping with a time of 208.98 ROS Time (5.7,-2.9).

Keywords: frontier exploration, gazebo, openSLAM.

1. Pendahuluan

Berdasar pada Undang-undang No. 1 tahun 1970 tentang Keselamatan dan Kesehatan Kerja, setiap aktivitas pekerjaan mewajibkan untuk melakukan perlindungan terhadap keselamatan kerja bagi pekerja, orang lain dan sumber-sumber produksi. Langkah-langkah penerapan, pembinaan dan evaluasi terhadap Keselamatan dan Kesehatan Kerja, harus dilakukan secara berkelanjutan untuk meningkatkan kualitas hasil kerja dan unsur Keselamatan dan Kesehatan Kerja terhadap pekerja, peralatan kerja dan lingkungan kerja. Aktivitas atau pekerjaan dalam ruang terbatas adalah salah satu aktivitas yang mengandung potensi bahaya sehingga sangat dibutuhkan penerapan program Keselamatan dan Kesehatan Kerja untuk menjamin keselamatan dan kesehatan bagi pekerja [1].

Oleh sebab itu, dibuat alat simulasi pemetaan lokasi yang belum diketahui dengan menggunakan turtlebot3 burger pada ROS. Simulasi ini dilakukan pada ruang tertutup untuk membuat map pada lokasi yang belum diketahui dengan menggunakan deteksi frontier dan *motion planning* pada *gmapping* di Gazebo dan Rviz. Berdasarkan keterangan yang sudah dijelaskan maka penelitian ini bertujuan untuk menghasilkan map pada lokasi yang belum diketahui yang selanjutnya hasil *mapping* bisa digunakan untuk menghasilkan peta sebaran gas.

2. Dasar Teori

2.1 Eksplorasi Frontier

Terdapat 4 arti kata 'frontier' di Kamus Bahasa Inggris Terjemahan Indonesia yang masuk ke dalam kelas kata nomina (kata benda). Menurut Kamus Bahasa Inggris Terjemahan Indonesia, arti kata frontier adalah padan. Arti lainnya dari frontier adalah garis perbatasan. Frontier masuk ke dalam bahasa Inggris yaitu bahasa Jermanik yang pertama kali dituturkan di Inggris pada Abad Pertengahan Awal dan saat ini merupakan bahasa yang paling umum digunakan di seluruh dunia [2].

Menurut A. E. Moodie frontier adalah zona (jalur) dengan lebar yang berbeda yang berfungsi sebagai pemisah dua wilayah yang berlainan negaranya. Menurut Weiger dan Moodie, Kristof frontier merupakan sebuah manifestasi dari kekuatan sentrifugal yang masih memungkinkan terjadinya saling interpenetrasi pengaruh antar dua negara yang berbatasan/bertetangga.

Eksplorasi adalah penjelajahan lapangan dengan tujuan memperoleh pengetahuan lebih banyak (tentang keadaan), terutama sumber-sumber alam yang terdapat di tempat itu (penyelidikan) penjajakan [3]. Frontier-based exploration digunakan untuk dapat menelusuri wilayah secara menyeluruh dimana dalam melakukan penelusuran pada wilayah tidak dikenali membutuhkan SLAM dan Navigation Stack agar robot dapat berjalan secara otonom, hasil peta yang telah dibuat menggunakan SLAM gmapping digunakan frontier-based exploration untuk membuat frontier point dimana digunakan robot untuk bergerak menuju titik yang ada pada perbatasan wilayah [4].

2.2 Pemetaan Ruang

Pemetaan ruang adalah kegiatan memetakan ruang baru atau belum diketahui sebelumnya yang dilakukan secara otomatis dengan menggunakan openSLAM pada ROS. Pemetaan ruang dilakukan dengan menggunakan gmapping dan move_base.

2.2.1. Gmapping

Pemetaan dengan gmapping menggunakan teknik occupancy grid map. Peta ini merupakan peta dua dimensi (2D) yang sering digunakan pada ROS. Occupancy grid map merupakan peta atau citra grayscale yang direpresentasikan dalam rentang nilai dari 0 sampai dengan 255. Nilai ini diperoleh dari probabilitas posterior menggunakan teorema Bayes. Metode gmapping membutuhkan informasi dari lidar dan data odometry robot. Lokalisasi pada gmapping menggunakan metode Monte Carlo Localization (MCL) atau sebut juga dengan filter partikel (particle filter). Sehingga metode ini cocok diterapkan pada turtlebot3 [5]. Gmapping disini lain menggunakan pendekatan lingkaran tertutup (closed loop approach) dimana jika robot kembali pada posisi awal lagi, peta yang dibentuk akan diperbarui [6].

2.2.2. Move_base

Move_base merupakan antarmuka ROS untuk mengkonfigurasi, menjalankan, dan berinteraksi dengan tumpukan navigasi pada robot [7]. Node move_base menyediakan antarmuka ROS untuk mengkonfigurasi, menjalankan, dan berinteraksi dengan navigasi pada robot. Blok diagram dari node move_base dan interaksinya dengan komponen lain.

2.3 OpenSLAM

OpenSLAM adalah penyedia platform bagi peneliti SLAM yang memberikan kemungkinan untuk mempublikasikan dan mempromosikan algoritma. Simultaneous Localization and Mapping (SLAM) telah dipelajari secara intensif di komunitas robotika di masa lalu. Berbagai teknik telah diusulkan tetapi hanya sedikit yang tersedia sebagai implementasi bagi masyarakat. OpenSLAM menyediakan bagi setiap peneliti SLAM yang tertarik sebuah repositori Subversion (svn) dan sebuah halaman web kecil untuk mempublikasikan dan mempromosikan project [8].

2.4 ROS

Robot Operating System (ROS) adalah framework pemrograman yang digunakan dalam bidang robotika. Filosofi dari ROS adalah untuk membuat perangkat lunak pada robot yang bisa bekerja di robot lain dengan melakukan perubahan kecil pada coding, dengan kata lain adalah untuk membuat fungsi pada robot yang dapat dibagi dan digunakan ulang pada robot lain tanpa harus membuat ulang fungsi tersebut dari awal agar sesuai dengan robot yang digunakan. ROS menyediakan fasilitas sistem operasi standar seperti abstraksi hardware, kontrol perangkat tingkat rendah, pelaksanaan fungsi yang umum digunakan, message passing diantara proses, dan manajemen paket. Hal ini didasarkan pada arsitektur ROS yang didasarkan pada topologi terpusat di mana pemrosesan terjadi di node yang dapat menerima input atau mengeluarkan output, seperti sensor multipleks, kontrol, aktuator, dan sebagainya [9]. Robot Operating System (ROS) bukanlah sistem operasi yang sebenarnya, tetapi kerangka kerja dan seperangkat alat yang menyediakan fungsionalitas sistem operasi pada cluster komputer yang heterogen. Kegunaannya tidak terbatas pada robot, tetapi sebagian besar alat yang disediakan difokuskan untuk bekerja dengan perangkat keras peripheral [10].

2.5 Gazebo

Gazebo merupakan sebuah aplikasi simulasi open source yang dapat mengola perkembangan mesin dinamis yang dikembangkan oleh Open Source Robotics Foundation [11]. Gazebo dapat mensimulasikan perancangan robot serta melatih sistem artificial intelligent dengan kemampuan simulasi yang akurat, perancangan simulasi dapat di desain

baik didalam maupun diluar ruangan yang kompleks [12]. Pada tugas akhir ini Gazebo digunakan sebagai simulasi perancangan lokasi yang belum diketahui dan di desain sedemikian rupa dengan frontier dan obstacle di dalamnya.

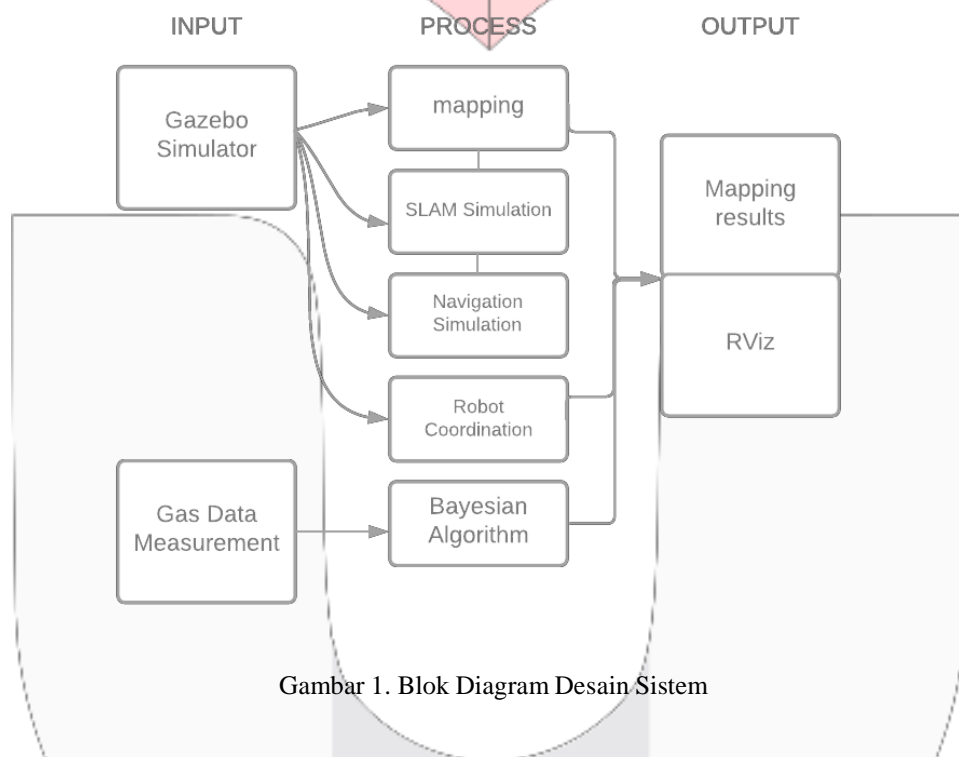
2.6 Rviz

Rviz adalah alat perangkat lunak visualisasi 3D untuk robot, sensor, dan algoritma. Digunakan untuk melihat persepsi robot (nyata atau simulasi). Tujuan dari rviz adalah untuk memvisualisasikan keadaan robot. Dengan menggunakan data sensor untuk mencoba membuat gambaran akurat tentang apa yang terjadi di lingkungan robot [13].

3. Perancangan Sistem

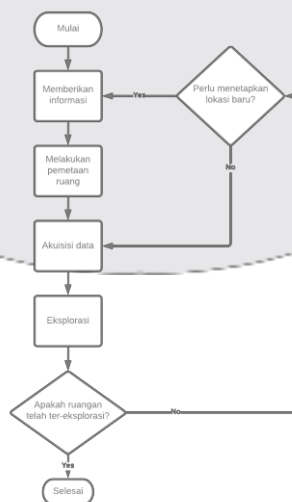
3.1 Desain Sistem

Eksplorasi dilakukan dengan frontier exploration ROS, menggunakan alat ini memerlukan konfigurasi robot nyata atau simulasi yang menyediakan fungsionalitas pemindai laser atau sensor yang dapat meng-clear ruangan dan menandai rintangan. Untuk penggambaran peta dengan menggunakan openSLAM dengan memperkirakan lokasi saat diruang sembarang yang belum dikenal. SLAM adalah fitur turtlebot.



Gambar 1. Blok Diagram Desain Sistem

3.2 Flowchart Perancangan Sistem



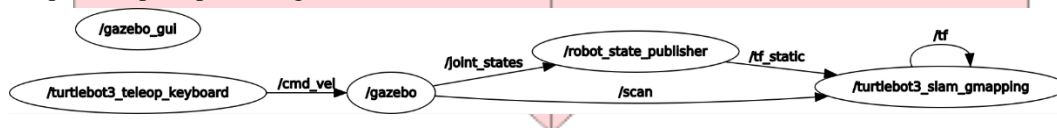
Gambar 2. Flowchart Perancangan Sistem

3.3 Fungsi dan Fitur

Fungsi dari Tugas Akhir ini adalah melakukan eksplorasi dan pemetaan ruang untuk mendapatkan peta sebaran material gas pada ruang tertutup. Fitur yang digunakan yaitu Turtlebot3 Burger sebagai robot yang akan meng-explore dan melakukan pemetaan ruang tertutup, di desain dengan menggunakan frontier dan openSLAM sebagai pemetaan ruang dan eksplorasi pada ruang tertutup. Ketika melakukan mapping robot akan berjalan sesuai perintah keyboard yang kita klik dengan fitur teleop_key pada ROS, posisi awal penempatan robot sesuai dengan (x,y) yang sudah di tentukan. Setelah melakukan mapping lokasi, dilakukan penyimpanan hasil mapping dengan perintah save_map pada ROS dengan hasil format file map adalah (.yaml).

3.4 Perancangan Simulasi Gazebo

Simulasi ini menggunakan Gazebo Simulator sebagai media perjalanan turtlebot3 yang kemudian nantinya akan di visualisasikan pada Rviz. Pada Gazebo Simulator di rancang bangunan tembok berbentuk persegi dan terdapat obstacle di dalamnya untuk mengetahui apakah robot dapat menghindari tembok saat melakukan motion planning. Adapun Graph ROS pada perancangan sistem.



Gambar 3. ROS Graph

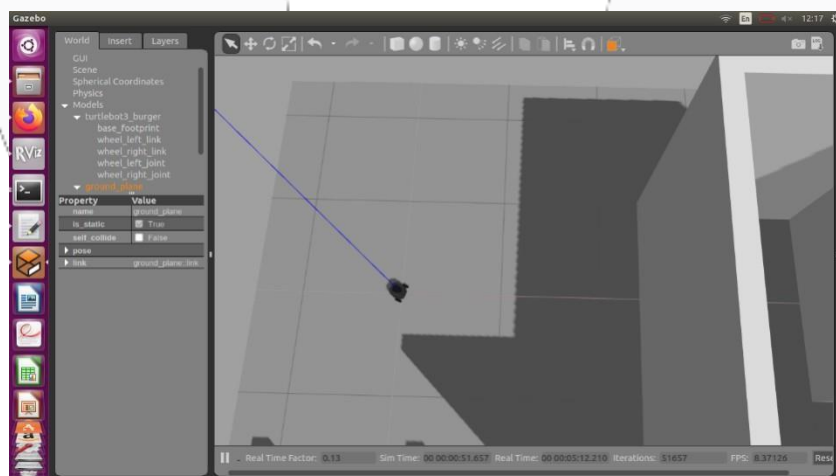
3.5 Perancangan Simulasi Rviz

Proses visualisasi mapping dan navigasi ditampilkan pada Rviz. Proses mapping dilakukan untuk mendapatkan hasil map pada lokasi yang belum diketahui dengan OpenSLAM, sementara proses navigasi dilakukan sebagai pilihan motion planning yang efektif dengan menggunakan navigation stack. Proses ini dipilih karena user hanya perlu menentukan titik awal robot dan goal reached pada Rviz. Perancangan dilakukan pada posisi awal robot (x,y)=(-1.5,-1.35) sampai dengan posisi akhir robot (x,y)=(5.7,-2.9).

4. Hasil dan Pengujian Sistem

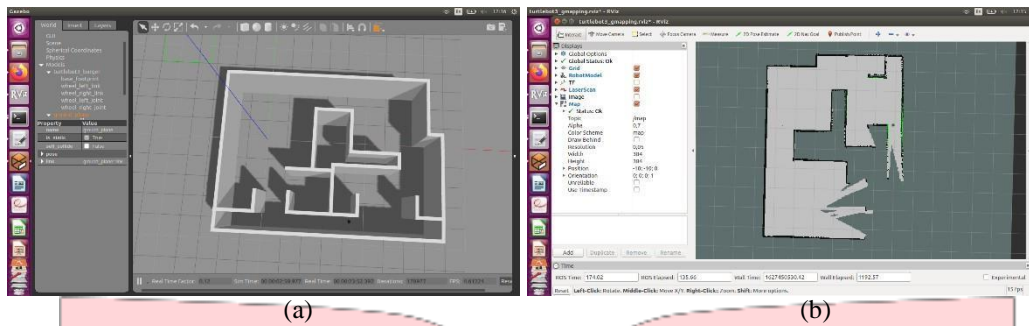
4.1 Pengujian SLAM Simulation tanpa Obstacle

Tujuan utama pengujian SLAM Simulation ini untuk mendapatkan hasil map di suatu lokasi yang belum diketahui sebelumnya, apakah turtlebot3 dapat bergerak melakukan pemetaan dan eksplorasi ruangan sampai ke titik terakhir (frontier), dan agar dapat mengetahui apakah robot turtlebot3 dapat menghindari benda di sekitar dan melakukan pemetaan ke daerah yang belum tereksplorasi. Simulai lokasi yang belum diketahui dirancang menggunakan Gazebo Simulator.



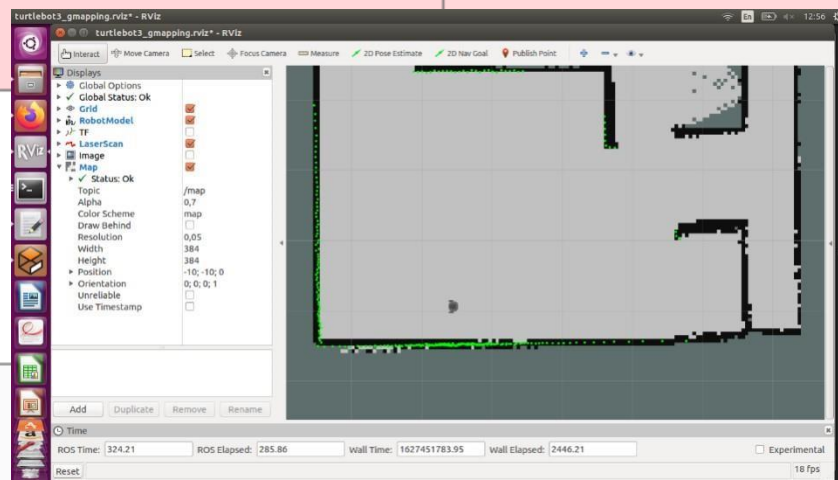
Gambar 4. Simulasi Lokasi yang Belum Diketahui

Titik awal turtlebot3 di posisikan pada (x,y)=(0,0), kemudian dilakukan proses SLAM dengan menggunakan teleoperation node sampai seluruh lokasi tereksplorasi.



Gambar 5. (a) Tampilan pada Gazebo, (b) Tampilan pada Rviz

Hasil mapping pada RViz menghasilkan titik-titik frontier berwarna hijau



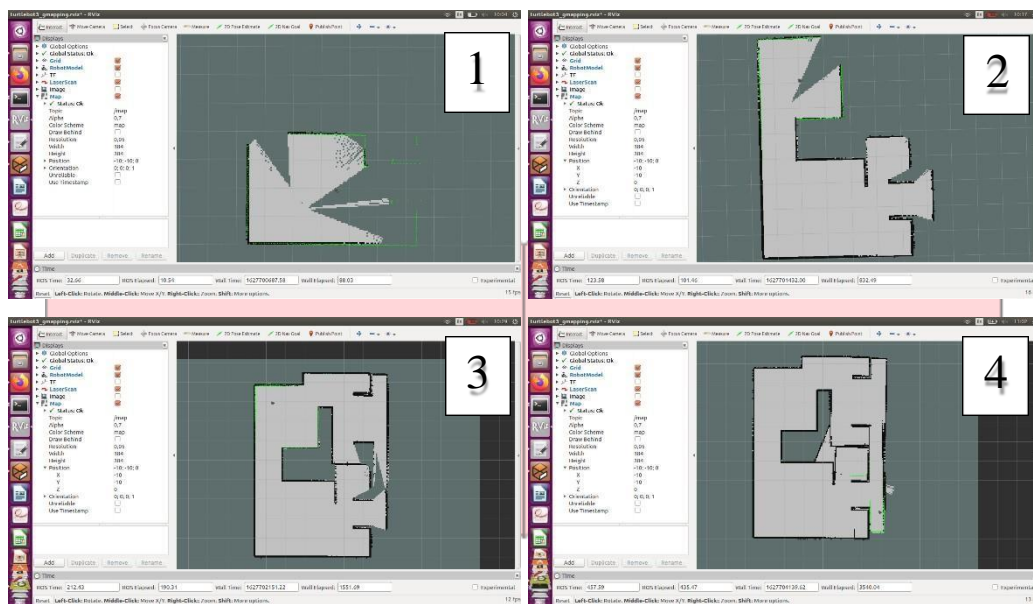
Gambar 6. Tampilan Frontier pada Rviz

Proses mapping menghasilkan data frame pada setiap titik eksplorasinya yang dapat dilihat pada Tabel 4.1. Parameter ini menetapkan nilai skor minimum yang menentukan keberhasilan atau kegagalan tes pencocokan data pindaian sensor. Hal ini diharapkan dapat mengurangi kesalahan posisi robot pada area yang luas.

Tabel 1. Data Frame Paramater Minimum Score

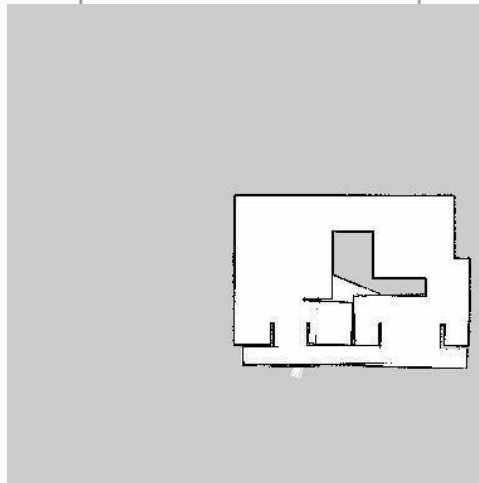
No	Frame	linearUpdate	angularUpdate	Laser Pose	Average Scan Matching Score	Registering Scans
1	0	0	0	-0.029084 -0.0031371 -3.11929	-	First Scan
2	3	1.8838E-07	5.45413E-06	-0.029084 -0.00313692 -3.11928	291.615	Done
3	6	1.8838e-07	5.45413e-06	-0.0290841 -0.00313674 -3.11928	291.632	Done
4	7	1.8838e-07	5.45414e-06	-0.0290841 -0.00313657 -3.11927	291.742	Done
5	8	3.13967e-07	9.09024e-06	-0.0290843 -0.00313627 -3.11926	291.787	Done
6	9	2.51174e-07	7.2722e-06	-0.0290843 -0.00313603 -3.11925	291.767	Done
7	10	3.76762e-07	1.09083e-05	-0.0290845 -0.00313568 -3.11924	291.803	Done
8	11	1.88381e-07	5.45416e-06	-0.0290845 -0.0031355 -3.11924	291.855	Done
9	12	3.13969e-07	9.09028e-06	-0.0290846 -0.00313521 -3.11923	291.763	Done
10	13	1.88887e-07	5.46882e-06	-0.0290847 -0.00313503 -3.11922	291.828	Done

Pada simulasi ini digunakan $\text{map_update_interval} = 2.0$, parameter ini menentukan waktu dalam memperbarui peta. Semakin kecil nilainya, semakin sering peta diperbarui.



Gambar 7. Proses Mapping

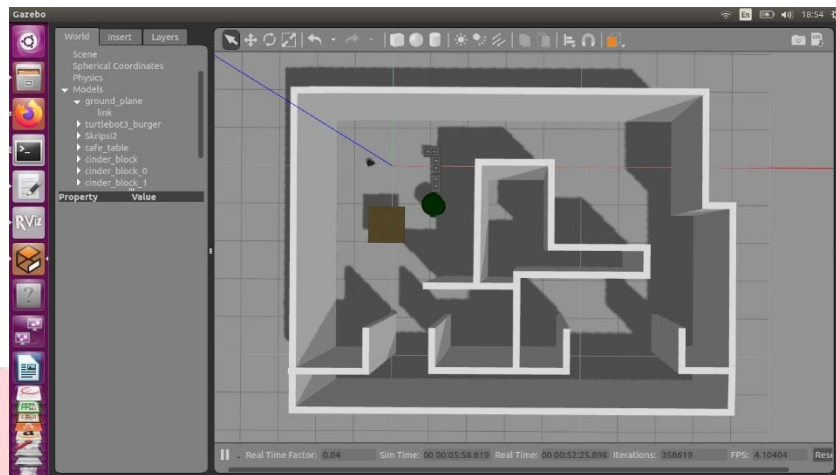
Hasil mapping yang telah disimpan dengan hasil format (.pgm) dapat dilihat pada Gambar 4.4. Hasil peta dengan Occupancy Grid Map (OGM) dua dimensi, yang biasa digunakan di ROS. Peta yang disimpan akan terlihat seperti gambar di bawah ini, di mana area putih adalah area bebas tabrakan sedangkan area hitam adalah area yang ditempati dan tidak dapat diakses, dan area abu-abu mewakili area yang tidak diketahui. Peta ini selanjutnya digunakan untuk Navigasi.



Gambar 8. Hasil Mapping Lokasi Menggunakan Gazebo Simulator dan Rviz

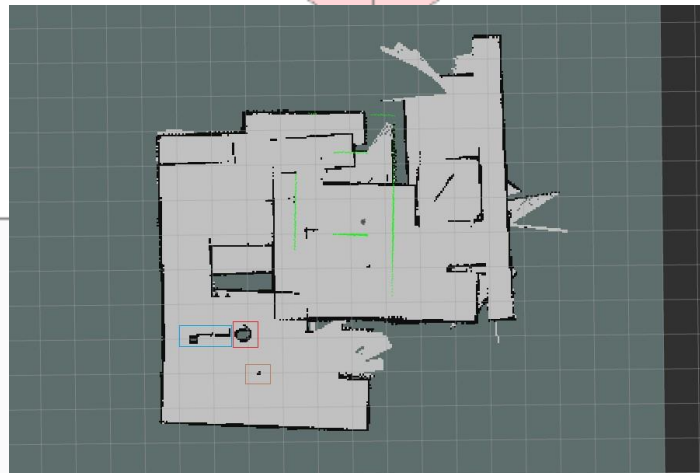
4.2 Pengujian SLAM Simulation tanpa Obstacle

Tujuan utama pengujian SLAM Simulation dengan obstacle ini untuk mendapatkan hasil map di suatu lokasi yang belum diketahui sebelumnya, apakah turtlebot3 dapat bergerak melakukan pemetaan dan eksplorasi ruangan sampai ke titik terakhir (frontier), dan agar dapat mengetahui apakah robot turtlebot3 dapat menghindari benda di sekitar dan melakukan pemetaan ke daerah yang belum tereksplorasi. Simulasi lokasi yang belum diketahui dirancang menggunakan Gazebo Simulator.



Gambar 9. Simulasi SLAM dengan *Obstacle* Menggunakan Gazebo Simulator

Dari gambar di atas dapat kita lihat bahwa di sekitar garis biru terdapat 1 meja, 1 tempat sampah, dan 3 batu bata yang disusun menyamping. Ke lima benda ini digunakan sebagai alat obstacle sebagai bahan pengujian apakah robot bisa menghindari benda yang terapat pada ruangan atau tidak. Dari hasil pengujian di dapatkan hasil *mapping*.

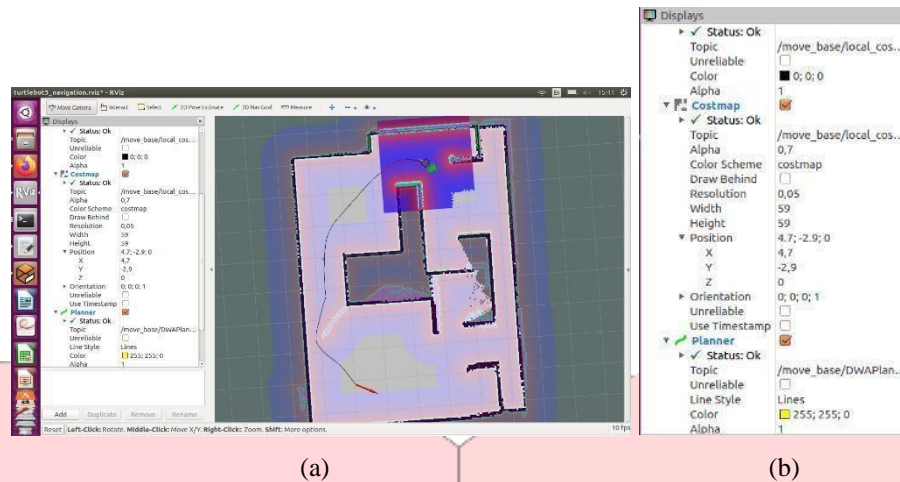


Gambar 10. Hasil *Mapping* dengan *Obstacle*

Dari gambar 3.7 dapat dilihat terdapat 3 kotak dengan warna berbeda yaitu diantaranya kotak biru merepresentasikan hasil mapping dari tiga buah kotak batu bata, kotak warna merah merepresentasikan hasil mapping satu buah tong sampah, dan kotak warna coklat merepresentasikan hasil mapping satu buah meja. Perbedaan garis hitam pada hasil mapping menggambarkan penglihatan pada kamera robot dimana garis pada tong sampah lebih jelas dibandingkan dengan meja, hal itu dikarenakan tong sampah lebih jelas terlihat oleh kamera dibandingkan dengan meja yang hanya di tangkap pada bagian bawah meja, yaitu penyanggah atau besi meja.

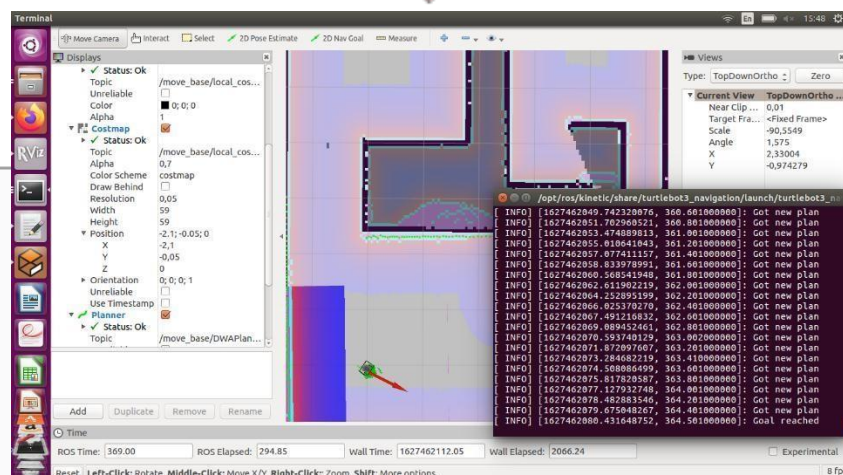
4.3 Pengujian Navigation Stack

Tujuan utama pengujian ini untuk mengetahui posisi robot dengan koordinat (x,y) dengan titik koordinat awal robot (x,y)=(0,0). Proses navigation stack sebelumnya harus sudah memiliki map lokasi yang sudah disimpan terlebih dahulu dengan format (.pgm) kemudian dilanjutkan untuk Navigasi.



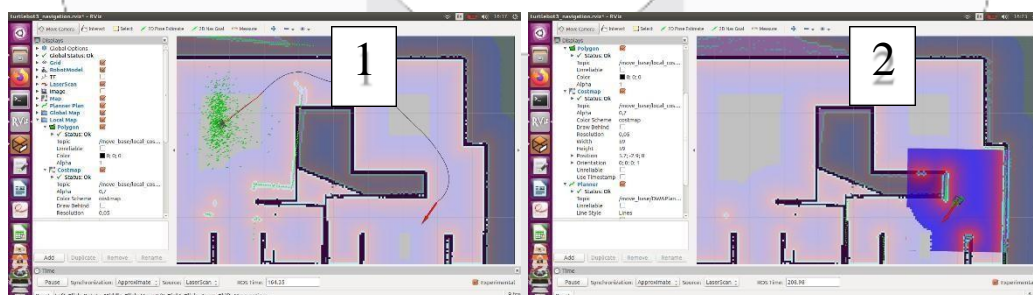
Gambar 11. (a) Proses Navigasi Stack pada Rvis, (b) Info Posisi Turtlebot3

Gambar (a) menunjukkan setiap perjalanannya menghasilkan titik koordinat (x,y) yang berubah-ubah sesuai dengan titik lokasi turtlebot3 dengan posisi X-Y yang dapat dilihat pada Gambar (b). Perjalanan turtlebot3 sesuai dengan arah 2D Pose Estimate sampai dengan 2D Nav Goal yang sudah di tentukan. Jika proses navigation stack selesai turtlebot3 berhenti dan mengeluarkan info *goal reached*.



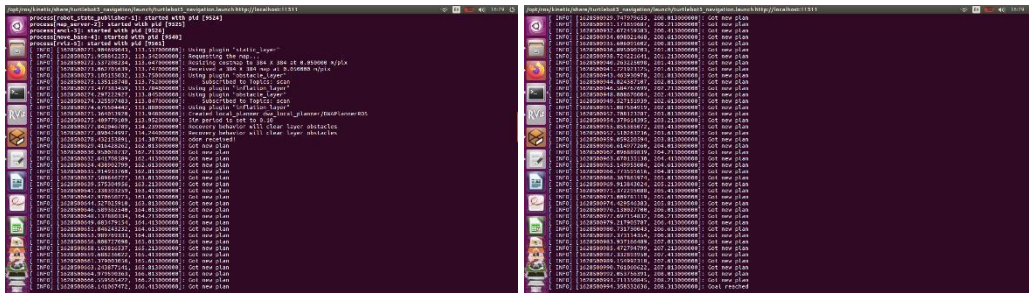
Gambar 12. Proses *Navigation Stack* Selesai

Perancangan dilakukan pada posisi awal robot $(x,y)=(-1.5,-1.35)$ sampai dengan posisi akhir robot $(x,y)=(5.7,-2.9)$ dengan durasi penyelesaian dari titik awal ke titik akhir adalah kurang lebih 208.98 ROS Time.



Gambar 13. Proses Navigasi pada Rviz

Proses yang dilakukan navigasi dari titik awal sampai dengan titik akhir



Gambar 14. Proses yang Dilakukan Navigasi pada Rviz

Dimana dalam perjalanannya robot melakukan proses new plan pada setiap node perjalanan hingga akhirnya berhenti pada titik terakhir dengan keluaran goal reached yang menandakan proses navigasi selesai.

4.3 Uji Akurasi

Uji akurasi dilakukan untuk mengetahui persentase akurasi ketika dilakukan perbandingan antara ground truth dengan hasil mapping. Perhitungan uji akurasi dengan menggunakan metode Euclidean Distance, menghitung persentase error, dan menghitung persentase tingkat akurasi.

4.3.1 Euclidean Distance

Euclidean distance adalah perhitungan untuk mengukur jarak dua titik dalam euclidean space yang mempelajari hubungan antara sudut dan jarak. Dalam matematika euclidean distance digunakan untuk mengukur dua titik dalam satu dimensi yang memberikan hasil seperti perhitungan Pythagoras [14]. Persamaan euclidean distance dapat dilihat pada persamaan (1).

Berikut ini adalah persamaan euclidean distance :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

(1) Dimana :

d = Jarak

x1 = Koordinat titik awal x

x2 = Koordinat titik akhir x

y1 = Koordinat titik awal y

y2 = Koordinat titik akhir y

4.3.2 Menghitung Persentase Error

Untuk menghitung persentase error dari hasil perhitungan, digunakan perhitungan galat relatif dan dikonversi kedalam persentase yang ditunjukkan pada persamaan (2)

$$\alpha = \left| \frac{\alpha^* - \alpha}{\alpha^*} \right| \times 100\% \quad (2)$$

Dimana α adalah nilai ground truth dan α^* adalah nilai hasil mapping.

4.3.3 Menghitung Persentase Tingkat Akurasi

Untuk menghitung persentase keakuratan dari hasil perhitungan, digunakan perhitungan yang ditunjukkan pada Persamaan (7)

$$100\% - \alpha = \text{Akurasi}$$

(3) Dimana α adalah nilai yang didapatkan dari perhitungan persentase error.

4.3 Hasil Uji Akurasi

Berikut adalah tabel klasifikasi uji simulasi pada Gazebo Simulator dan Rviz.

Tabel 2. Tabel Klasifikasi Uji Simulasi

Klasifikasi Uji Simulasi	Ground Truth						Hasil Mapping					
	Terminal		Turtle		GUI		Rviz		Turtle		GUI	
	x	y	x	y	x	y	x	y	x	y	x	y
Titik awal	-0.030	1.418	-0.000	0.000	0.332	-1.356	-1.450	-1.450	-0.000	0.000	0.048	-1.246
Titik akhir	0.988	0.001	1.019	0.001	0.494	-1.588	-0.45	-1.45	1.013	0.002	0.727	-0.945
Titik akhir ke-2	0.965	-1.006	0.963	-1.063	0.74	-1.192	-0.45	-2.45	0.997	-1.008	0.58	-1.192

Setelah dilakukan perhitungan di dapatkan hasil uji akurasi dari titik awal robot menuju titik akhir robot.

Tabel 3 Tabel Perhitungan *Euclidean Distance*, *Error*, dan Persentase Akurasi

Jarak pada Terminal	Jarak pada Rviz	Error	Akurasi (%)
1.743	1.000	42.62	57.38

Jarak Turtle GT	Jarak Tutle Maps	Error	Akurasi (%)
1.019	1.041	2.2	97.8

Jarak GUI GT	Jarak GUI Maps	Error	Akurasi (%)
0.282	0.741	61.94	38.06

Dari perhitungan tabel di atas di dapatkan hasil presentase akurasi pada perbandingan ground truth (terminal) dan hasil mapping (Rviz) dari nilai pada terminal, turtle, dan GUI. Dengan hasil masing-masing tingkat akurasi 57,38% pada terminal dan RViz, 97,8% pada turtle ground truth dan turtle hasil mapping, dan 38,06% pada GUI ground truth dan GUI hasil mapping.

5. Kesimpulan

Dari serangkaian pengujian simulasi yang dilakukan sistem pada pengujian SLAM dan Navigation Stack, maka dapat disimpulkan bahwa:

1. Sistem dapat melakukan *mapping* lokasi yang belum diketahui dan menghindari *obstacle* dengan menggunakan Gazebo Simulator sebagai alat simulasi dan turtlebot3 sebagai alat yang digunakan untuk *mapping*.
2. Pengujian simulasi dengan OpenSLAM dan Frontier menghasilkan *mapping* dengan durasi penyelesaian kurang lebih 208.98 ROS Time dari titik $(\blacklozenge, \blacklozenge) = (-1.5, -1.35)$ menuju titik $(\blacklozenge, \blacklozenge) = (5.7, -2.9)$.
3. Didapatkan hasil uji akurasi dengan hasil masing-masing tingkat akurasi 57,38% pada terminal dan RViz, 97,8% pada turtle *ground truth* dan turtle hasil *mapping*, dan 38,06% pada GUI *ground truth* dan GUI hasil *mapping*.

REFERENSI

- [1] D. Ruang and T. Con, "Pengetahuan Umum Tentang Ruang Terbatas (Confined Space)." p. 2018, 2018.
- [2] Lektur.ID, "4 Arti Kata Similarity di Kamus Bahasa Inggris Terjemahan Indonesia." 2020.
- [3] Kamus besar bahasa Indonesia, "Arti kata eksplorasi - Kamus Besar Bahasa Indonesia (KBBI) Online," *Kemendikbud*. 2016.
- [4] T. Tao, Y. Huang, F. Sun, and T. Wang, "Motion planning for SLAM based on frontier exploration," *Proc. 2007 IEEE Int. Conf. Mechatronics Autom. ICMA 2007*, no. 60605021, pp. 2120–2125, 2007, doi: 10.1109/ICMA.2007.4303879.
- [5] A. Rahman, "Penerapan SLAM Gmapping dengan Robot Operating System Menggunakan Laser Scanner pada Turtlebot," *J. Rekayasa Elektr.*, vol. 16, no. 2, 2020, doi: 10.17529/jre.v16i2.16491.
- [6] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 34–46, 2007, doi: 10.1109/TRO.2006.889486.
- [7] E. Marder-Eppstein, "move_base - ROS Wiki," *ROS.org*. 2016, [Online]. Available: http://wiki.ros.org/move_base.
- [8] "OpenSLAM - The resource for open source SLAM implementations."
- [9] L. B. Slam, S. Localization, S. Localization, E. K. Filter, and A. Fastslam, "BAB I," pp. 1–5.
- [10] R. Singgih Pradipta, "Prototipe Troli Pengikut Otomatis Menggunakan Pengolahan Citra Kamera Pixy Cmcum 5 Berbasis Arduino Pengolahan Citra Kamera Pixyc Mucam 5 Berbasis Arduino," p. 20, 2016, [Online]. Available: <http://eprints.ums.ac.id/46600/>.
- [11] E. M. Hoffman *et al.*, "Yarp based plugins for gazebo simulator," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8906, pp. 333–346, 2014, doi: 10.1007/978-3-319-13823-7_29.
- [12] "Gazebo."
- [13] Addison Sears-Collins, "What is the Difference Between rviz and Gazebo? – Automatic Addison." 2020, [Online]. Available: <https://automaticaddison.com/what-is-the-difference-between-rviz-and-gazebo/>.
- [14] Y. Miftahuddin, S. Umaroh, and F. R. Karim, "Perbandingan Metode Perhitungan Jarak Euclidean, Haversine, Dan Manhattan Dalam Penentuan Posisi Karyawan," *J. Tekno Insentif*, vol. 14, no. 2, pp. 69–77, 2020, doi: 10.36787/jti.v14i2.270.