

RANCANG BANGUN SISTEM PENCITRAAN LINGKUNGAN 3D DAN NAVIGASI PADA *AUTONOMOUS MOBILE ROBOT* MENGUNAKAN ALGORITMA D*

Yusuf Gutara¹⁾, Aris Budiarto²⁾, Susetyo Bagas Bhaskoro³⁾
Jurusan Teknik Mesin dan Manufaktur, Konsentrasi Teknik Elektromekanik
Politeknik Manufaktur Bandung
Jl. Kanayakan No. 21 – Dago, Bandung - 40135
Phone/Fax : 022. 250 0241 / 250 2649
Email: yusufgutara@gmail.com

ABSTRAK

Data spasial adalah data yang memiliki informasi ruang dan tempat. Data yang sudah terkomputerisasi berperan penting menemukan perubahan bagaimana menggunakan dan mengetahui informasi tentang lingkungan. Pengambilan data spasial dapat dilakukan secara manual, akan tetapi metode seperti itu memerlukan waktu yang lebih lama dan tidak dapat dilakukan pada tempat-tempat sempit dan berbahaya yang tidak dimungkinkan adanya manusia. Sensor jarak dapat digunakan untuk pengambilan data spasial, dengan ukuran sensor yang kecil, sebuah *unmanned vehicle* atau *autonomous mobile robot* dapat dengan mudah membawanya untuk menelusuri lingkungan, dengan begitu resiko dapat dikurangi.

Pada tugas akhir ini bertujuan untuk merancang suatu kendali robot pemetaan menggunakan sebuah sistem yang dapat menelusuri lingkungan dan dilengkapi dengan sensor jarak. Sensor akan digunakan untuk menerima data jarak dari sekeliling robot. Data tersebut akan digunakan untuk merepresentasikan lingkungan, juga sebagai input kendali untuk menelusuri lingkungan dan menghindari objek yang menghalangi. Sistem navigasi dapat diterapkan pada sebuah robot untuk melewati daerah atau tempat yang belum pernah dilalui sebelumnya. Algoritma D* dapat diterapkan sebagai navigasi, dengan algoritma ini, robot akan bergerak menuju titik koordinat yang diinginkan.

Berdasarkan percobaan yang telah dilakukan sebanyak dua kali pada empat skema, robot berhasil mencapai area yang ditentukan yaitu pada radius 30 cm dari titik yang diinputkan, dengan akurasi navigasi sebesar 68.39% pada sistem SLAM sementara dengan pengukuran secara manual akurasi navigasi sebesar 73.17% dan robot berhasil menghindari halangan yang ada. Pada proses sistem lokalisasi dibutuhkan objek disekitar robot kurang dari 3 meter untuk dapat berorientasi terhadap lingkungannya secara maksimal.

Kata kunci: *mapping*, data spasial, *mobile robot*, navigasi, lokalisasi

I. PENDAHULUAN

Dalam sistem pemetaan pada suatu tempat, biasanya pengambilan data secara manual akan dilakukan oleh operator [15], akan tetapi untuk tempat sempit atau berbahaya dapat menjadi resiko untuk operator itu sendiri. Sebuah robot dapat menggantikan peran dari operator, dalam robot tersebut dapat diterapkan sistem lokalisasi dan pemetaan. Lokalisasi sendiri adalah proses membangun hubungan antara sistem koordinat peta dan sistem koordinat lokal robot, sementara untuk pemetaan dapat digambarkan sebagai proses untuk menggambarkan lingkungan berdasarkan informasi di sekitarnya yang diwakili dalam gambar grafis [3]. Untuk dapat melaksanakan pekerjaannya robot harus memiliki sistem navigasi yang berjalan secara simultan dengan sistem lokalisasi dan pemetaan sehingga proses pengambilan data spasial dapat dilakukan secepat mungkin [2]. Dapat

disimpulkan tiga sistem utama yang ada pada alat ialah sistem navigasi, lokalisasi, dan pemetaan. Penelitian yang banyak dilakukan adalah pemetaan lingkungan secara 2D contohnya, penelitian yang dilakukan oleh Arun [11] dan Iksan dkk.[2], pada kedua penelitian itu menggunakan sebuah LIDAR 1D yang diputar untuk mendapatkan data 2D. Penelitian lainnya menggunakan LIDAR 2D seperti pada [15] dan [3]. Keempat penelitian tersebut telah mencapai akurasi lebih dari 90%. Tetapi, dengan menggunakan 2D sulit untuk mengetahui objek yang dipetakan. Sehingga diperlukan untuk mengubah data 2D ke dalam bentuk 3D pada lingkungan yang diinginkan.

Terdapat dua penelitian yang telah dilakukan mengenai sistem pencitraan 3D menggunakan sensor 2D, yang pertama memutarnya dengan sebuah gimbal secara *tilt* untuk mendapatkan citra 3D dari lingkungan [5] dan yang kedua

dengan cara meletakkan sensor secara vertikal, proses *scanning* 3D akan terjadi saat robot bergerak pada sumbu X Y [4]. Kedua penelitian tersebut masih memiliki keterbatasan pada sisi navigasi yaitu tidak bisa mencapai titik yang diinginkan, karena tidak dirancang untuk itu. Algoritma yang biasanya digunakan untuk mencapai titik yang diinput oleh *user* adalah algoritma Dijkstra, A*, D*, dan *Rapidly-Exploring Random Trees* (RRT) [6]. Algoritma Dijkstra merupakan salah satu algoritma paling sederhana tetapi berdasarkan lingkungan algoritma A* memungkinkan untuk menyelesaikan pekerjaannya lebih cepat dibandingkan algoritma Dijkstra [6]. Sementara algoritma D* merupakan algoritma A* yang bersifat dinamik karena nilai *cost functions* dapat berubah-ubah berdasarkan *obstacle* yang muncul [6] [13]. Algoritma D* dapat digunakan pada lingkungan yang belum diketahui secara lengkap [13].

Karena peta tidak didefinisikan terlebih dahulu maka dipilih sistem navigasi menggunakan algoritma D* untuk mencapai titik koordinat tujuan, setelah itu akan diambil citra 3D pada lingkungan tersebut. Citra 3D akan diambil dengan cara memutar sensor menggunakan sebuah gimbal, penggunaan gimbal akan memerlukan lebih sedikit sensor, jika dibandingkan dengan sensor yang diletakan secara vertikal. Sehingga, pada tugas akhir ini dibuat *mobile robot* untuk pemetaan yang memiliki navigasi menggunakan algoritma D*

II. TINJAUAN PUSTAKA

Pada sistem terdapat dua metode utama yang diimplementasikan agar robot dapat melakukan navigasi kemudian mengambil citra 3D dari lingkungan tersebut, metode yang dipakai sebagai berikut.

2.1 Algoritma A* dan D*

Terdapat banyak algoritma yang dapat digunakan sebagai navigasi robot. Algoritma A* adalah salah satu algoritma yang direkomendasikan khususnya untuk lingkungan yang diketahui secara menyeluruh dan algoritma D* untuk lingkungan yang belum diketahui secara menyeluruh [13]. Algoritma D* merupakan algoritma A* dengan kemampuan mengubah nilai *cost function* tergantung dari *obstacle* yang muncul [6] [13]. Ide dari algoritma A* yaitu dengan menghitung jumlah dari biaya dari titik awal ke titik selanjutnya (G-cost), dengan titik sekarang dengan titik akhir (H-cost), jumlah antara H-cost dengan G-cost disebut F-

cost. Sehingga dapat ditulis dalam persamaan berikut.

$$F = G + H \quad (2.1)$$

Dimana :

G : Harga dari node awal ke node terdekat.

H : Harga dari node awal ke node akhir.

F : Total biaya G dan H.

Sebagai contoh penggunaan metode A* dapat dilihat pada Gambar 2.1.

| | | | | | | |
|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|--|
| | | | | | | |
| G = 14 H = 60 F = 74 | G = 10 H = 50 F = 60 | G = 14 H = 50 F = 54 | | | | |
| G = 10 H = 50 F = 60 | | G = 10 H = 30 F = 40 | | G = 52 H = 10 F = 62 | | |
| G = 14 H = 60 F = 74 | G = 10 H = 50 F = 60 | G = 14 H = 40 F = 54 | | G = 42 H = 20 F = 62 | G = 52 H = 10 F = 62 | |
| | G = 28 H = 60 F = 88 | G = 24 H = 50 F = 74 | G = 28 H = 40 F = 68 | G = 38 H = 30 F = 68 | | |

Gambar 2. 1 Lintasan yang dilalui [7]

Titik awal adalah kotak berwarna hijau dan titik akhir adalah kotak berwarna merah. Titik yang dipilih selanjutnya yaitu titik dengan F-cost paling kecil, perhitungan akan terus berjalan sampai robot berada di titik tujuan.

2.2 Kinematika

Kinematika adalah cabang ilmu yang menganalisis gerak tanpa memperhatikan apa yang menyebabkan gerak. Yang dimaksud gerak disini adalah semua jenis perpindahan, yang mencakup perubahan posisi dan orientasi. Secara umum, hampir semua masalah kinematika dapat ditafsirkan sebagai vektor tambahan yang ditransformasikan terhadap *frame* referensi [8].

Terdapat dua jenis analisis kinematika pada robot, kinematika langsung dan kinematika balik. Kinematika maju dapat digunakan saat sudut joint diketahui untuk menemukan data dalam koordinat kartesius. Sementara kinematika balik adalah sebaliknya. Berikut adalah formula transformasi untuk memindahkan informasi kinematik bolak-balik dari kerangka koordinat ke kerangka koordinat lainnya.

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \begin{bmatrix} \hat{T}.i & \hat{T}.j & \hat{T}.k \\ \hat{J}.i & \hat{J}.j & \hat{J}.k \\ \hat{R}.i & \hat{R}.j & \hat{R}.k \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} \quad (2.2)$$

Persamaan 2.2 merupakan *dot product* antara besarnya nilai sudut antara *frame* referensi dan

frame gerak dengan letak titik yang ada pada *frame* gerak, sehingga didapatkan titik yang telah digerakan terhadap *frame* referensi.

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 45 & -\sin 45 \\ 0 & \sin 45 & \cos 45 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 3 \\ 5 \\ 7 \end{bmatrix} \quad (2.3)$$

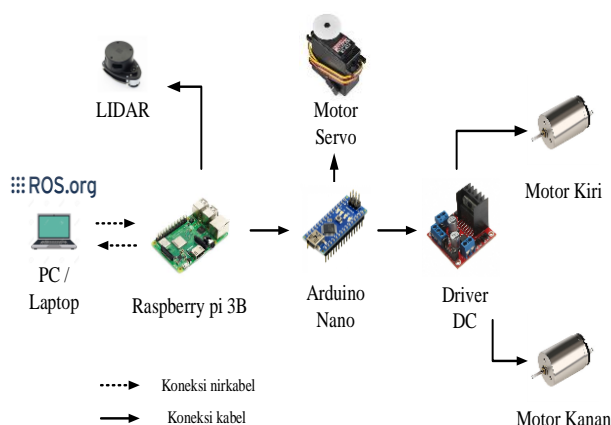
Sementara untuk gerakan translasi dapat diperoleh dengan cara melakukan penjumlahan pada masing-masing *axis*. Persamaan dapat dilihat pada persamaan 2.3. yaitu titik yang ada di *frame* referensi xyz diputar sejauh 45 derajat pada sumbu x kemudian dilakukan gerakan translasi sejauh (3, 5, 7).

III. METODOLOGI PENELITIAN

Pada tugas akhir ini robot dibuat dengan menerapkan VDI 2206 yaitu perancangan mekatronika. Perancangan meliputi tiga domain yaitu, domain mekanik, elektrik, dan program. Ketiga domain tersebut kemudian diintegrasikan.

3.1 Gambaran Umum Sistem

Sistem yang akan dibuat pada tugas akhir ini, terdiri dari sebuah sensor utama, sebuah penggerak gimbal dan dua buah penggerak robot. Sensor yang akan digunakan adalah sensor LIDAR. Pada sistem, LIDAR akan digunakan untuk sistem *obstacle avoidance* dan sistem lokalisasi. Sistem lokalisasi sendiri akan digunakan sebagai pendukung sistem navigasi dan pemetaan. Pada sisi aktuator, motor servo akan digunakan untuk menggerakkan gimbal. Kemudian sebagai penggerak utama sistem, digunakan dua buah motor DC.



Gambar 3. 1 Gambaran Umum Sistem

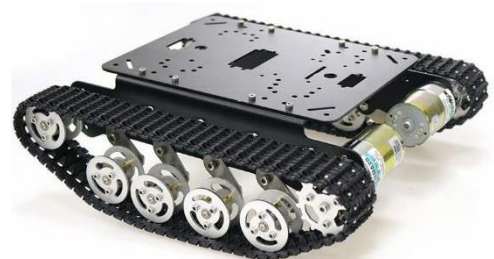
Pada sistem ini akan digunakan *Robot Operating System* (ROS) sebagai *software* pendukung sistem. Penggunaan ROS akan memudahkan dalam menjalankan banyak program pada sistem, memudahkan komunikasi

antar *machine*, memudahkan *parsing* data, selain itu terdapat *tools* seperti Rviz yang dapat digunakan sebagai *Human Machine Interface* (HMI) pada robot.

Robot akan menuju koordinat yang dipilih, setelah itu data 3D akan diambil dari sekeliling koordinat tersebut. Raspberry Pi sebagai *machine* akan menerima data jarak dari LIDAR kemudian dikirimkan ke *machine* utama yaitu PC/laptop menggunakan komunikasi *secure shell* (SSH). Setelah itu, pada *machine* utama data jarak digunakan untuk lokalisasi dan navigasi. Perintah pergerakan robot akan dikirimkan lewat Raspberry pi dan diproses pada Arduino Nano (*slave*).

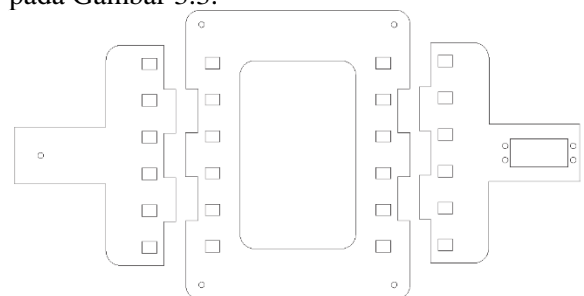
3.2 Domain Rancangan Mekanik

Chassis merupakan komponen mekanik utama pada sistem ini, karena merupakan bagian badan dari robot. Di pasaran *chassis* yang dapat digunakan untuk robot banyak dijual, maka pada sistem ini digunakan *chassis* yang sudah ada, dengan mempertimbangkan dimensi yang pas untuk meletakkan komponen. Selain itu *chassis* harus stabil jika robot dibuat lantai ke-2. Pada Gambar 3.2 yaitu *chassis* yang digunakan pada sistem.



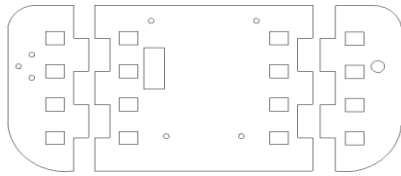
Gambar 3. 2 Chassis tank (sumber: bukalapak.com)

Setelah *chassis* tersedia, selanjutnya merancang bagian gimbal. Gimbal terdiri dari dua bagian utama, pertama yaitu dasar gimbal yang akan berfungsi sebagai penahan gimbal gerak dan tempat diletakkannya motor penggerak gimbal. Rancangan dasar gimbal dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Gimbal Tetap

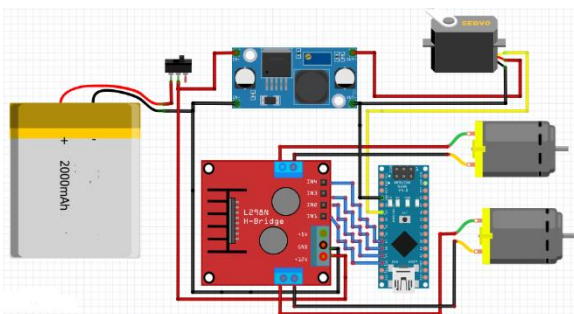
Bagian selanjutnya dari gimbal, adalah gimbal gerak. Gimbal gerak merupakan dudukan dari sensor yang akan diputar untuk mendapatkan data secara 3D. Kedua bagian ini akan dibuat dengan bahan akrilik dan dibuat dengan *laser cutting*.



Gambar 3. 4 Gimbal Gerak

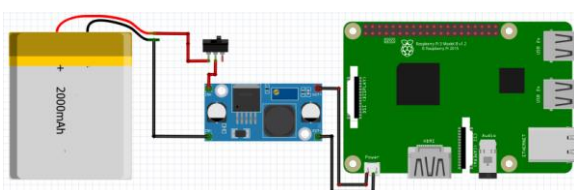
3.3 Domain Rancangan Elektrik

Rangkaian elektrik terkumpul dalam satu rangkaian *motherboard*. Selain menjadi papan PCB utama *motherboard* digunakan untuk mengkoneksikan sumber tegangan dengan beban dan sumber data dengan pengolah data. Komponen terhubung dengan *motherboard* yang pertama adalah baterai, pada Gambar 3.5 baterai digunakan khusus untuk sumber tegangan motor servo dan DC, terpisah dengan sumber tegangan yang digunakan pada Raspberry PI dan Arduino. Diperlukan satu tambahan komponen untuk menyesuaikan tegangan sumber dengan tegangan kerja motor servo, dari tegangan baterai dua sel yaitu 7.4V diubah menjadi 5V. Sementara, motor DC tidak memerlukannya.



Gambar 3. 5 Rangkaian Elektrik 1

Data yang diperlukan yang digunakan pada masing-masing aktuator yaitu PWM. Data PWM akan diubah menjadi kecepatan motor pada motor DC dan menjadi besaran sudut pada motor servo. Selain pada Gambar 3.5, komponen lain yang terhubung ke *motherboard* yaitu pada Gambar 3.6.

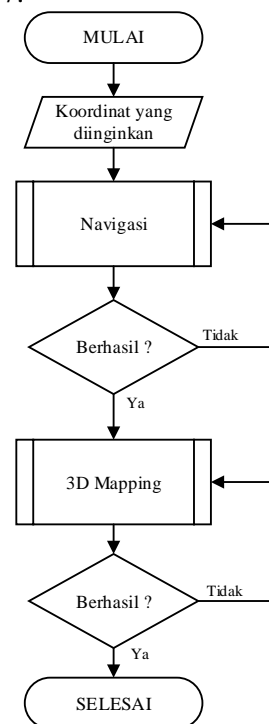


Gambar 3. 6 Rangkaian Elektrik 2

Untuk bisa menghidupkan Raspberry Pi lebih lama maka baterai antara mini komputer dengan motor dipisah. Tegangan yang sumbernya didapat dari baterai 2 sel diturunkan menjadi 5V. Koneksi antara rangkaian elektrik 1 dan 2 digunakan komunikasi serial yang dihubungkan dengan USB, antara Raspberry PI dengan Arduino Nano. Pada konektor USB yang lain juga digunakan untuk sensor RPLidar.

3.4 Domain Rancangan Informatik

Untuk melakukan tugasnya robot harus memiliki sistem pengendali agar sesuai dengan algoritma yang diinginkan. Algoritma pada sistem ini dapat dilihat pada *flowchart* pada Gambar 3.7.

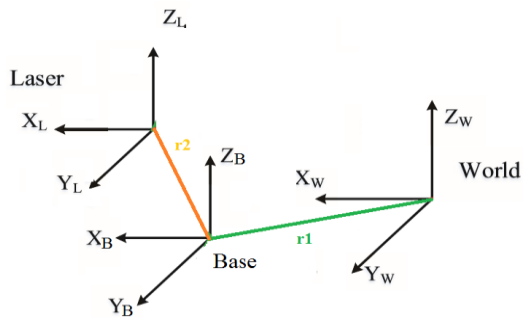


Gambar 3. 7 Flowchart algoritma utama robot

Dimulai dengan memasukkan koordinat yang diinginkan pada sistem. Pada blok navigasi koordinat tujuan dengan koordinat sekarang dari robot akan dilakukan perhitungan untuk mengetahui koordinat mana disekeliling robot yang memiliki jarak terpendek. Setelah itu robot akan maju/berputar ketitik sekeliling sampai robot sampai pada koordinat yang diinginkan.

Lalu jika robot sudah diam di titik yang diinginkan robot akan melakukan pengambilan data secara 3D. Untuk mengubah data menjadi citra lingkungan secara 3D maka digunakan perhitungan kinematika untuk mengetahui hubungan antara koordinat satu dengan yang lain. Kemudian robot akan menunggu koordinat tujuan berikutnya.

Navigasi dilakukan dengan menggunakan algoritma D* sementara untuk citra 3D didapatkan dengan cara mencari hubungan antara *frame* yang ada pada sistem, kemudian mengubahnya menjadi terhadap *frame* referensi.



Gambar 3. 8 Frame koordinat yang ada pada sistem

Frame base robot dapat diperoleh dengan menggunakan lokalisasi dari *hector slam*. Sementara data rotasi dari *frame* laser dapat diperoleh dari data pergerakan motor servo pada gimbal.

IV. HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan hasil implementasi dari navigasi dan *forward kinematic*. Robot yang dibuat dapat dilihat pada Gambar 4.1.

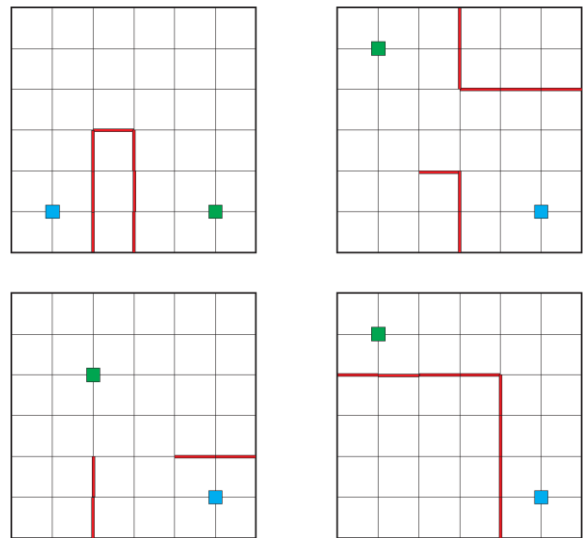


Gambar 4. 1 Sistem yang telah diintegrasikan

4.1 Pengujian Navigasi

Pengujian navigasi dilakukan di sebuah *field* dengan dimensi 180x180 cm dengan bantuan ubin 30x30 cm sebagai *grid* dan dilengkapi pembatas sepanjang 30 cm sebanyak delapan buah yang dapat dipindahkan. Pada pengujian ini dibagi menjadi empat skema yang masing-masing diuji sebanyak dua kali. Skema dapat dilihat pada Gambar 4.2. Data yang diambil dari

pengukuran sebenarnya dan menggunakan sistem SLAM



Gambar 4. 2 Skema pada pengujian navigasi

Titik awal robot ditandai dengan kotak biru dan titik tujuan dari robot ditandai dengan kotak hijau. Pada pengujian ini akan navigasi dikatakan berhasil ketika robot berada pada radius 30 cm dari titik yang diinputkan. Kemudian perhitungan error dilakukan dengan cara mencari persentase dari titik koordinat sebenarnya terhadap titik koordinat yang diinputkan.



Gambar 4. 3 Lintasan pada skema pertama

Pada Gambar 4.3 merupakan lintasan gerak yang dibuat oleh robot menggunakan sistem SLAM. Dari empat skema yang telah dilakukan dapat dituliskan kedalam sebuah tabel untuk mengetahui perbandingan antara koordinat yang didapat dari sistem SLAM dengan koordinat yang didapat dengan cara pengukuran manual. Koordinat dari sistem SLAM dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Koordinat akhir didapat dari sistem SLAM

| Skema | Koordinat input X (cm) | Koordinat input Y (cm) | Koordinat SLAM X (cm) | Koordinat SLAM Y (cm) | Error X (cm) | Error Y (cm) |
|-----------|------------------------|------------------------|-----------------------|-----------------------|--------------|--------------|
| 1 | -1 | -120 | 4 | -121 | 5 | 1 |
| 2 | 90 | 90 | 81 | 87 | 9 | 3 |
| 3 | 120 | 120 | 115 | 108 | 5 | 12 |
| 4 | 120 | 120 | 116 | 120 | 4 | 0 |
| Rata-rata | | | | | 5.75 | 4 |

| Skema | Koordinat input X (cm) | Koordinat input Y (cm) | Koordinat SLAM X (cm) | Koordinat SLAM Y (cm) | Error X (cm) | Error Y (cm) |
|-----------|------------------------|------------------------|-----------------------|-----------------------|--------------|--------------|
| 1 | -1 | -120 | 2 | -121 | 3 | 1 |
| 2 | 90 | 90 | 85 | 86 | 5 | 4 |
| 3 | 120 | 120 | 114 | 106 | 6 | 14 |
| 4 | 120 | 120 | 113 | 124 | 7 | 4 |
| Rata-rata | | | | | 5.25 | 5.75 |

Kemudian koordinat akhir yang didapat dari pengukuran manual dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Koordinat akhir dari pengukuran manual

| Skema | Koordinat input X (cm) | Koordinat input Y (cm) | Koordinat X manual (cm) | Koordinat Y manual (cm) | Error X (cm) | Error Y (cm) |
|-----------|------------------------|------------------------|-------------------------|-------------------------|--------------|--------------|
| 1 | -1 | -120 | -5 | -119 | 4 | 1 |
| 2 | 90 | 90 | 88 | 83 | 2 | 7 |
| 3 | 120 | 120 | 116 | 98 | 4 | 22 |
| 4 | 120 | 120 | 114 | 110.5 | 6.5 | 9.5 |
| Rata-rata | | | | | 4.12 | 9.87 |

| Skema | Koordinat input X (cm) | Koordinat input Y (cm) | Koordinat X manual (cm) | Koordinat Y manual (cm) | Error X (cm) | Error Y (cm) |
|-----------|------------------------|------------------------|-------------------------|-------------------------|--------------|--------------|
| 1 | -1 | -120 | -7 | -118 | 6 | 1 |
| 2 | 90 | 90 | 88.5 | 86.5 | 1.5 | 1 |
| 3 | 120 | 120 | 108 | 112 | 12 | 8 |
| 4 | 120 | 120 | 115.5 | 112 | 4.5 | 8 |
| Rata-rata | | | | | 8 | 6 |

Untuk menghitung persentase ketepatan navigasi dibutuhkan selisih antara titik yang diinputkan dengan titik sebenarnya. Pada pengujian ini navigasi dikatakan gagal saat titik sebenarnya robot berada diluar radius 30 cm dari titik yang diinputkan sehingga dapat dihitung dengan selisih antar titik dibagi 30 kemudian dikali 100%.

Tabel 4. 3 Error pada sistem SLAM

| Skema | Error X (cm) | Error Y (cm) | Jarak titik (cm) | Error (%) |
|-----------|--------------|--------------|------------------|-----------|
| 1 | 5 | 1 | 5.1 | 17 |
| 2 | 9 | 3 | 9.48 | 31.6 |
| 3 | 5 | 12 | 13 | 43.33 |
| 4 | 4 | 0 | 4 | 13.33 |
| Rata-rata | | | | 26.31 |

| Skema | Error X (cm) | Error Y (cm) | Jarak titik (cm) | Error (%) |
|-----------|--------------|--------------|------------------|-----------|
| 1 | 4 | 1 | 4.1 | 13.66 |
| 2 | 5 | 4 | 6.4 | 21.33 |
| 3 | 4 | 22 | 22.3 | 74.33 |
| 4 | 6.5 | 9.5 | 11.5 | 38.33 |
| Rata-rata | | | | 36.91 |

Tabel 4. 4 Error pada pengukuran manual

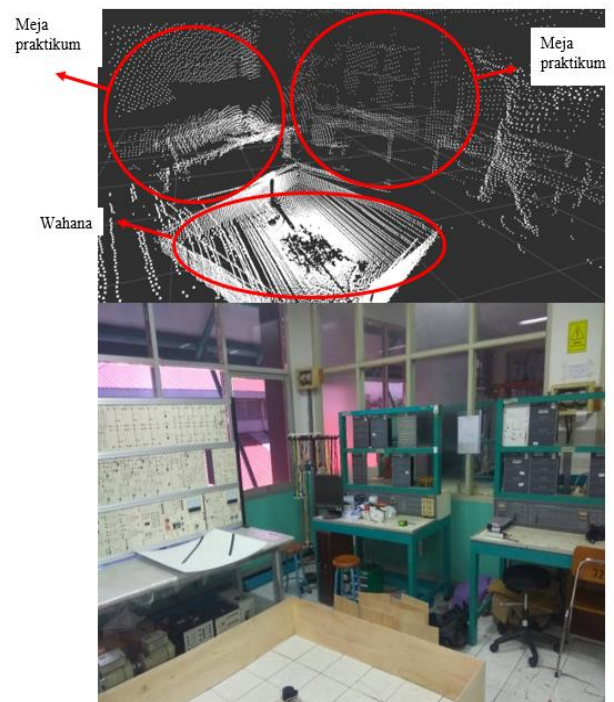
| Skema | Error X (cm) | Error Y (cm) | Jarak titik (cm) | Error (%) |
|-----------|--------------|--------------|------------------|-----------|
| 1 | 3 | 1 | 3.2 | 10.66 |
| 2 | 5 | 4 | 6.4 | 21.33 |
| 3 | 6 | 14 | 15.2 | 50.66 |
| 4 | 7 | 4 | 8.1 | 27 |
| Rata-rata | | | | 27.41 |

| Skema | Error X (cm) | Error Y (cm) | Jarak titik (cm) | Error (%) |
|-----------|--------------|--------------|------------------|-----------|
| 1 | 6 | 1 | 6.1 | 20.33 |
| 2 | 1.5 | 1 | 1.8 | 6 |
| 3 | 12 | 8 | 14.4 | 48 |
| 4 | 4.5 | 8 | 9.2 | 30.66 |
| Rata-rata | | | | 26.25 |

Pada data yang diperoleh dari lokalisasi dapat disimpulkan bahwa akurasi dari sistem navigasi yang dibuat 68.39%, sementara dengan pengukuran secara manual akurasi navigasi sebesar 73.17%. Pada delapan kali percobaan robot berhasil 100% menghindari halangan yang ada, dapat dilihat pada *trajectory* yang dibuat robot pada sistem SLAM.

4.1 Pengujian Citra 3D

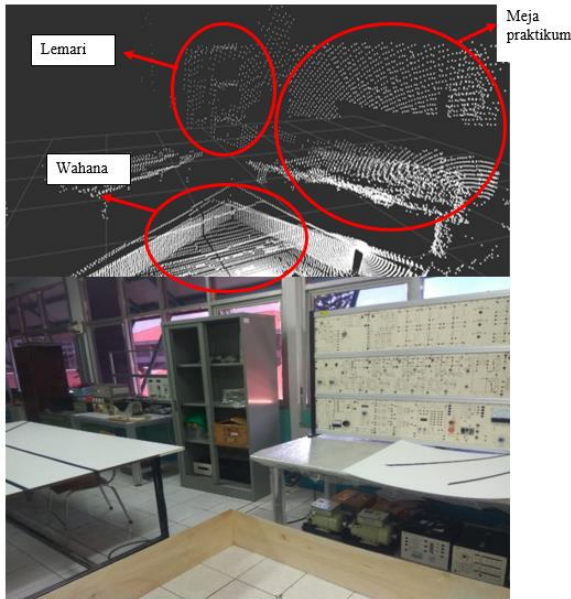
Fungsi utama lain dari sistem yaitu mengambil citra 3D dari lingkungan. Pada pengujian ini dilakukan di Lab. Teknik Penggerak, Politeknik Manufaktur Bandung. Hasil dapat dilihat pada Gambar 4.4 dan Gambar 4.5.



Gambar 4. 4 Perbandingan citra 3D dengan foto 1

Jika dibandingkan pada Gambar 4.4, bentuk lemari, meja praktikum, dan dinding *field* pada citra 3D telah menyerupai benda aslinya. Error yang ada pada citra 3D, dapat disebabkan oleh

latency data pergerakan motor servo, yaitu perbedaan waktu antara proses perhitungan dengan posisi motor servo yang dibutuhkan.



Gambar 4. 5 Perbandingan citra 3D dengan foto 2

V. KESIMPULAN

Berdasarkan pengujian yang dilakukan terhadap sistem yang dibangun dan selama proses penyusunan karya tulis ini, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Koordinat titik ukur pada lingkungan dapat diperoleh dengan melakukan perhitungan kinematika antara *frame* koordinat referensi, robot, dan laser.
2. Menggunakan sistem lokalisasi *scan matching* akan sangat tergantung dari lingkungan, lingkungan dinamis akan mengganggu sistem lokalisasi.
3. Sensor RPLidar A1M8 hanya dapat mendeteksi objek yang terletak kurang dari 3 Meter.
4. Sensor yang ada pada sistem memiliki akurasi sebesar 98.35%.
5. Akurasi dari navigasi sebesar 68.39% pada sistem SLAM sementara dengan pengukuran secara manual akurasi navigasi sebesar 73.17%.
6. Dari delapan kali percobaan yang dilakukan, robot berhasil menghindari halangan yang ada.

7. Keakuratan Citra 3D tergantung dengan keakuratan sensor, navigasi, dan motor servo.

Saran untuk penelitian selanjutnya yang berkaitan dengan sistem yang dituliskan pada karya tulis ini, adalah sebagai berikut:

1. Sensor LIDAR merupakan komponen utama pada sistem, maka direkomendasikan untuk mengganti RPLidar A1M8 dengan Hokuyo utm-30lx, karena memiliki spesifikasi lebih tinggi, yaitu *angular resolution* sebesar 0.25 derajat, dengan jarak ukur 30 meter.
2. Menambahkan sensor *ping* untuk *obstacle avoidance*, karena keterbatasan dari LIDAR dalam mendeteksi objek transparan.
3. Menggabungkan sistem lokalisasi, SLAM dengan *odometry* menggunakan metode *complementary filter*.
4. Menambah sensor IMU untuk menangani *offset* karena permukaan tanah yang tidak rata dan mengetahui kemiringan dari sensor saat diputar oleh gimbal.
5. Menambah fitur IOT agar dapat dikendalikan dari jarak yang jauh.

DAFTAR PUSTAKA

- [1] F. Ardilla, "Path Tracking Pada Mobile Robot Dengan Umpan Balik," in The 13th Industrial Electronics Seminar, Surabaya, 2011.
- [2] I. Maulana, "Aplikasi Lidar Untuk Pemetaan dan Navigasi pada Lingkungan Tertutup," in e-Proceeding of Engineering, Bandung, 2018.
- [3] M. A. Markom, "A Mapping Mobile Robot using RP Lidar Scanner," in 2015 IEEE International Symposium on Robotics and Intelligent Sensors, Langkawi, 2015.
- [4] M. Chen, S. Yang, X. Yi and D. Wu, "Real-time 3D Mapping using a 2D Laser Scanner and IMU-aided Visual SLAM," in 2017 IEEE International Conference on Real-time Computing and Robotics (RCAR 2017), Okinawa, 2017.

- [5] M. González, S. Alvarado, N. Certad and A. Terrones, "Autonomous 2D SLAM and 3D Mapping of an Environment Using a Single 2D LIDAR and ROS," in 2017 Latin American Robotics Symposium (LARS 2017) and 2017 Brazilian Symposium on Robotics (SBR 2017) , Curitiba, 2017.
- [6] M. Karova, D. Zhelyazkov and . M. Todorova, "Path Planning Algorithm for Mobile Robot," in Proceedings of the 15th International Conference on Applied Computer Science, Konya, 2015.
- [7] NN, "[Algoritma] Belajar Algoritma A* Untuk Pencarian Jalur / Rute Terdekat," Duniadigit, 22 August 2013. [Online]. Available: <http://duniadigit.blogspot.com/2013/08/belajar-algoritma-untuk-pencarian-jalur.html>. [Accessed 22 July 2019].
- [8] R. N. Jazar, Theory of Applied Robotics: Kinematics, Dynamics, and Control, Second Edition, Melbourne: Springer, 2010.
- [9] R. Siegwart and I. R. Nourbakhsh, Introduction to Autonomous Mobile Robots, Cambridge: The MIT Press , 2004.
- [10] Robotics Online Marketing, "The Latest Technological Innovations in Autonomous Mobile Robots," Robotics, 15 May 2018. [Online]. Available: <https://www.robotics.org/blog-article.cfm/The-Latest-Technological-Innovations-in-Autonomous-Mobile-Robots/97>. [Accessed 25 July 2019].
- [11] S. Arun and S. Krishna, "Autonomous 2D Mapping of an Unknown Environment Using Single 1D LIDAR and ROS," International Journal of Engineering Research & Technology, vol. 7, no. 03, pp. 10-13, 2018.
- [12] S. Banker, "The Autonomous Mobile Robot Market Is Taking Off Like A Rocket Ship," Forbes, 11 March 2019. [Online]. Available: <https://www.forbes.com/sites/stevebanker/2019/03/11/the-autonomous-mobile-robot-market-is-taking-off-like-a-rocket-ship/#24b0bc981603>. [Accessed 25 July 2019].
- [13] S. Gopikeishnan, B. Shravana and H. Golea, "Path Planning Algorithms: A Comparative Study," in Indian Institute of Space Science & Technology, Thiruvananthapuram, 2011.
- [14] S. Kohlbrecher, J. Meyer, O. v. Stry and U. Klingauf, "A Flexible and Scalable SLAM System with Full 3D Motion Estimation," in 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, 2011.
- [15] S. Prayoga, "Sistem Pemetaan Ruang 2D Menggunakan Lidar," Jurnal Integrasi, vol. IX, no. 1, pp. 73-79, 2017.