

PEMETAAN KONTUR LINGKUNGAN PADA *MOBILE ROBOT* MENGGUNAKAN WIIMOTE

Randy Erfa Saputra

Fakultas Teknik Elektro, Universitas Telkom
resaputra@telkomuniversity.ac.id

Abstrak

Navigasi merupakan kemampuan dasar yang harus dimiliki oleh *mobile robot* sebelum dapat melaksanakan tugasnya dan pemetaan merupakan hal yang utama dalam permasalahan navigasi. Robot memerlukan peta sebagai dasar pengetahuan dari lingkungannya serta sensor-sensor yang berfungsi untuk mengenali keadaan lingkungannya. Pada penelitian ini dirancang sebuah sistem untuk membuat kontur lingkungan dengan memanfaatkan *Wiimote* pada *mobile robot*. Output yang dihasilkan berupa plot tiga dimensi dari titik-titik (x, y, z) yang diperoleh dari kontur permukaan yang dilewati robot, dengan mengimplementasikan perangkat lunak untuk komunikasi data antara laptop dengan *Wiimote* dan *mobile robot*, serta melakukan perhitungan untuk menentukan keadaan kontur lingkungan. Hasil yang diperoleh adalah sistem dapat membuat peta kontur lingkungan yang mendekati keadaan lingkungan yang sebenarnya dengan akurasi rata-rata sebesar 93%.

Kata kunci: *Wiimote, accelerometer, mobile robot, kontur*

Abstract

Navigation is basic skills that must be owned by the mobile robot before perform its tasks and mapping is a main issue in navigation problem. Robot needs a map as a basic knowledge of the environment and sensors to recognize the condition of its environment. In this paper, we designed a system to create environment contour by using *Wiimote* on the mobile robot. Outputs were produced in the form of three-dimensional plot of the points (x, y, z) that was obtained from surface contour passed by robot, by implementing software for data communication between laptop, *Wiimote*, and mobile robot, and performed calculation to determine the environment contour. The results show that system can create environment contour which nearly match the actual condition with average accuracy of 93%.

Keywords: *Wiimote, accelerometer, mobile robot, contour*

1. Pendahuluan

Mobile robot merupakan salah satu jenis robot yang memiliki kemampuan bergerak bebas di lingkungannya dan tidak terpaku pada satu tempat tertentu saja. Kata robot sendiri berasal dari bahasa Czech "robota" yang berarti pekerja dan diperkenalkan ke publik pada tahun 1920 melalui pertunjukan karya Karel Čapek yang berjudul *Rossum's Universal Robot* [7]. Pertunjukan tersebut bercerita tentang mesin yang menyerupai manusia tapi mampu bekerja terus-menerus tanpa lelah. Sedangkan jika merujuk pada *American Heritage Dictionary*, robot adalah sebuah alat mekanik yang terkadang menyerupai manusia dan mampu melakukan berbagai tugas rumit berdasarkan perintah atau telah terprogram sebelumnya.

Navigasi merupakan kemampuan dasar yang harus dimiliki oleh *mobile robot* sebelum dapat bergerak melaksanakan tugas dan fungsinya. Permasalahan navigasi dapat dimisalkan dengan menjawab pertanyaan dasar seperti "di manakah saya?", "kemana tujuan saya?", dan "bagaimanakah caranya agar saya dapat mencapai tujuan tersebut?".

Jawaban atas ketiga pertanyaan tersebut sangat penting untuk kepentingan navigasi pada robot. Oleh karena itu, robot memerlukan peta sebagai dasar pengetahuan dari lingkungannya serta sensor-sensor yang berfungsi untuk mengenali keadaan lingkungannya tersebut, sehingga masalah pemetaan ini menjadi hal yang utama dalam permasalahan navigasi pada robot [2,5].

Tujuan dari penelitian ini adalah untuk membuat peta dari lingkungan yang berkontur dengan memanfaatkan *Wiimote* pada *mobile robot*. Output yang akan dihasilkan berupa plot tiga dimensi dari titik-titik (x, y, z) yang diperoleh dari kontur permukaan yang dilewati robot.

2. *Wiimote*

Wii remote, atau yang biasa disingkat dengan "*Wiimote*", merupakan pengendali utama untuk *console* Nintendo Wii. Salah satu fitur utama dari *Wiimote* ini adalah kemampuannya untuk mendeteksi pergerakan (*motion sensing*), yang memungkinkan penggunaanya berinteraksi dan memanipulasi item-item di layar melalui pengenalan gerak (*gesture*

recognition) dan penunjukan yang akurat dengan menggunakan *accelerometer* dan teknologi sensor optik. Fitur lainnya adalah kemampuan *Wiimote* untuk menggabungkan perangkat ekstensi lainnya, seperti *Nunchuk* dan *Classic Controller* untuk menambah sensasi dan pengalaman bermain *game*. Namun, dalam penelitian ini cukup hanya menggunakan sebuah *Wiimote* tanpa perangkat ekstensi.

Wiimote memiliki kemampuan untuk mendeteksi percepatan dari ketiga sumbu X, Y, dan Z melalui penggunaan *accelerometer* ADXL330. *Accelerometer* ini memiliki jangkauan pengukuran sebesar $\pm 3g$ (*gravity*) dengan tingkat sensitivitas sebesar 1% [1]. *Accelerometer* ini memiliki enam derajat kebebasan dimana sumbu positif X ke arah kiri, sumbu positif Y ke arah belakang, dan sumbu Z positif ke arah atas. Gambar 1 menunjukkan sistem koordinat tersebut beserta dengan arah positif-negatifnya.

Pada keadaan diam, *Wiimote* tidak memberikan data percepatannya melainkan data hasil pembacaan sensor pada keadaan gravitasi yang dialami oleh *Wiimote*. Misalnya, saat diletakkan di atas meja yang datar dengan keadaan *Wiimote* menghadap ke atas, maka data pembacaan sensor akan memberikan nilai $+1g$ pada arah sumbu Z dan nilai 0 pada sumbu X dan Y. Hal ini disebabkan pada saat *Wiimote* diletakkan di atas meja, gaya gravitasi hanya bekerja pada arah sumbu Z. Fitur *motion sensing* inilah yang akan digunakan di dalam penelitian ini.

3. Kinematika Mobile Robot

Mobile robot merupakan salah satu jenis robot yang memiliki kemampuan bergerak bebas di lingkungannya dan tidak terpaku pada satu tempat tertentu saja. Dewasa ini *mobile robot* banyak digunakan baik untuk penelitian ataupun untuk membantu manusia mengerjakan tugas spesifik, misalnya sebagai penyedot debu atau pemotong rumput. Selain itu, *mobile robot* juga dapat ditemukan dalam industri, militer, dan keamanan lingkungan.

Kinematika adalah ilmu yang mempelajari tentang gerak dari suatu benda tanpa memperhitungkan penyebab gerak benda tersebut. Jika dikaitkan dengan *mobile robot*, ilmu ini mempelajari pergerakan robot terhadap sistem kerangka koordinat tanpa memperhatikan gaya yang menyebabkan pergerakan tersebut. Pada penelitian ini digunakan *mobile robot* jenis *differential drive*.

Gambar 2 menunjukkan beberapa parameter yang digunakan pada kinematika *mobile robot* jenis *differential drive*. Beberapa parameter tersebut antara lain:

- (x, y) merupakan koordinat posisi dari robot.
- v merupakan kecepatan linier dari robot.
- V_L dan V_R merupakan kecepatan linier dari masing-masing roda (kanan dan kiri).
- θ merupakan orientasi dari robot.

- R merupakan jari-jari rotasi dari robot.
- L merupakan jarak dari kedua roda.

Nilai V_L dan V_R diperoleh dari pembacaan *encoder* pada setiap sumbu roda robot. *Encoder* yang digunakan pada penelitian ini memiliki resolusi 20 pulsa/rotasi dan dilakukan *sampling* setiap detik untuk mendapatkan jumlah pulsa n . Sehingga apabila roda memiliki diameter d , maka nilai V_L dan V_R dapat dihitung dengan menggunakan persamaan:

$$V_{R/L} = \frac{n}{20} \times \pi d \quad (1)$$

Hubungan antara masing-masing parameter pada Gambar 2 di atas adalah sebagai berikut [3]:

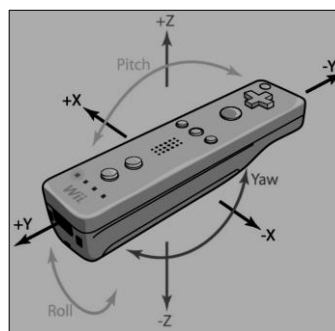
$$v = \frac{V_R + V_L}{2} \quad (2)$$

$$\omega = \frac{V_R - V_L}{L} \quad (3)$$

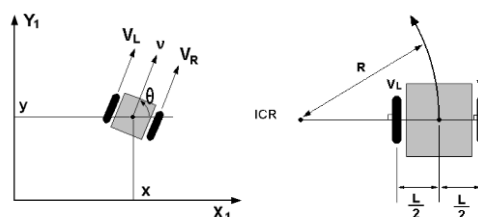
Nilai ω merupakan kecepatan angular dari robot dan bernilai positif untuk putaran yang berlawanan dengan arah jarum jam.

$$R = \frac{L}{2} \times \frac{V_R + V_L}{V_R - V_L} \quad (4)$$

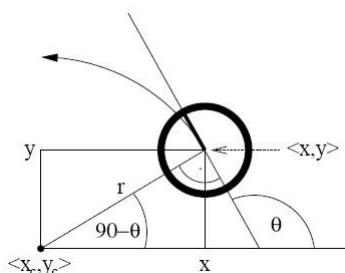
Dari persamaan (2) dapat terlihat bahwa ω akan bernilai 0 apabila nilai V_R sama dengan nilai V_L . Keadaan tersebut akan terpenuhi ketika robot bergerak lurus, yaitu apabila kecepatan roda kanan dan kiri sama besar. Namun, hal yang sebaliknya terjadi pada persamaan (3). Nilai R akan menjadi tak hingga apabila robot bergerak lurus dikarenakan nilai V_R dan V_L sama besar.



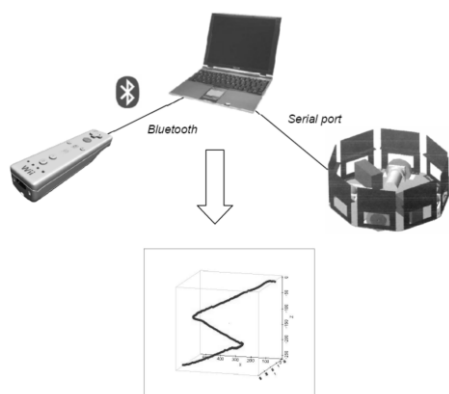
Gambar 1. Sistem Koordinat *Wiimote*



Gambar 2. Model *Differential Drive*



Gambar 3. Model Pergerakan Robot [4]



Gambar 4. Deskripsi Umum Sistem



Gambar 5. Mobile Robot dan Laptop

Ketika robot bergerak lurus selama selang waktu Δt dengan v konstan, maka posisi robot selanjutnya (x', y') dapat dicari dengan menggunakan persamaan berikut:

$$x' = x + v\Delta t \cos \theta \quad (5)$$

$$y' = y + v\Delta t \sin \theta \quad (6)$$

Namun apabila robot sedang dalam keadaan berbelok ke kiri atau ke kanan, model pergerakan dari Gambar 3 dapat digunakan untuk menentukan posisi dari robot dengan syarat nilai v dan ω dijaga konstan. Hal ini dapat dilakukan dengan mencari nilai x_c dan y_c terlebih dahulu.

$$x_c = x - \frac{v}{\omega} \sin \theta \quad (7)$$

$$y_c = y + \frac{v}{\omega} \cos \theta \quad (8)$$

Nilai x_c dan y_c ini merupakan titik pusat rotasi robot ketika sedang berbelok ke kiri atau ke kanan. Dari titik pusat ini, dapat dicari posisi dan orientasi robot selanjutnya setelah robot bergerak selama selang waktu Δt dengan v dan ω konstan, dengan menggunakan persamaan berikut:

$$x' = x_c + \frac{v}{\omega} \sin(\theta + \omega\Delta t) \quad (9)$$

$$y' = y_c - \frac{v}{\omega} \cos(\theta + \omega\Delta t) \quad (10)$$

$$\theta' = \theta + \omega\Delta t \quad (11)$$

Sehingga posisi robot pada peta dapat digambarkan berdasarkan perhitungan-perhitungan tersebut.

4. Perancangan dan Implementasi

Gambar 4 menunjukkan deskripsi umum sistem yang dibangun. Sistem tersebut terdiri dari *Wiimote* yang tersambung melalui antarmuka *Bluetooth* ke laptop, sedangkan laptop tersambung ke *mobile robot* menggunakan antarmuka kabel konversi *USB to serial*. Kemudian sistem ini akan diujikan pada lingkungan berkotur dan menghasilkan output berupa plot tiga dimensi dari kontur lintasan yang dilewati robot.

4.1. Spesifikasi Mobile Robot

Spesifikasi perangkat keras yang dimiliki *mobile robot* adalah sebagai berikut [6]:

- a. Lingkungan operasi *mobile robot*
 - 1) Berlantai dasar yang rata dan tidak bergelombang.
 - 2) Suhu ruangan diasumsikan $\pm 25^\circ$ dimana komponen elektronik bekerja normal.
- b. *Plant* mekanik
 - 1) Robot berbentuk segi delapan yang memiliki diameter 55 cm, tinggi 22 cm, dan terdiri atas tiga tingkat.
 - 2) Bahan yang digunakan untuk *chassis*, *body*, dan pembatas tingkat dua adalah plat aluminium, sedangkan untuk pembatas tingkat tiga adalah *acrylic*. Gambar 5 menunjukkan bentuk fisik robot dengan laptop diletakkan di atasnya.
- c. Aktuator
 - 1) Robot bergerak menggunakan dua buah roda yang memiliki diameter 12 cm dan metode pengendaliannya menggunakan sistem *differential steering*.
 - 2) Mekanisme roda gigi antara motor dan roda penggerak menggunakan tipe *spurs gear* dengan perbandingan gigi 21:75.
 - 3) Robot juga menggunakan dua buah *omni-directional wheel* untuk menjaga

keseimbangan dan sekaligus berfungsi sebagai *roller*.

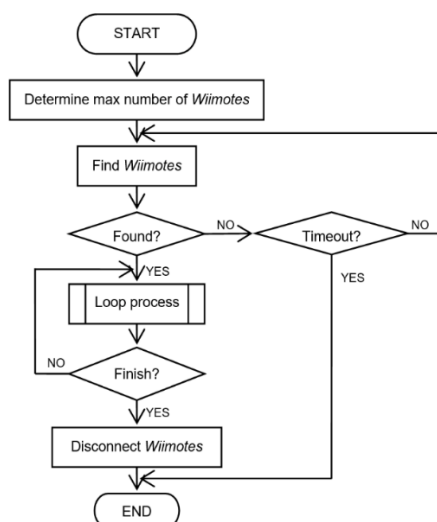
- 4) Penguat dan pengendali arah putaran motor menggunakan *driver* motor L298.
- d. Sistem pengendali dan komunikasi
- 1) Robot menggunakan dua buah pengendali, yaitu mikrokontroler AVR ATMEGA32 dan sebuah laptop sebagai *human-machine interface*.
 - 2) Komunikasi antara mikrokontroler dan laptop menggunakan komunikasi serial RS232 dengan antarmuka kabel konversi *USB to serial*.

4.2. Implementasi Perangkat Lunak

Perangkat lunak yang diimplementasikan terdiri dari dua bagian, yaitu:

- a. Bagian pertama merupakan program utama yang dijalankan dari laptop. Program ini akan menangani rutin komunikasi dengan *Wiimote* melalui konektivitas *Bluetooth*, komunikasi dengan mikrokontroler melalui *port serial*, melakukan perhitungan-perhitungan posisi dan kinematika robot, serta menghasilkan keluaran berupa ilustrasi grafik.
- b. Bagian kedua merupakan program yang ditanamkan ke dalam mikrokontroler ATMEGA32. Program ini bertugas untuk menerima perintah dari laptop melalui komunikasi serial, yang selanjutnya akan mengendalikan pergerakan *mobile robot* dengan membangkitkan sinyal PWM.

Dalam implementasinya, program utama yang dijalankan dari laptop ditulis dalam bahasa pemrograman C++ menggunakan perangkat lunak *Code::Blocks v8.02* yang bersifat *open source* dan *cross platform*. Sedangkan program yang ditanamkan ke dalam mikrokontroler ditulis dalam bahasa pemrograman C menggunakan perangkat lunak *CodeVisionAVR v1.25*.



Gambar 6. Diagram Alir Komunikasi *Wiimote*

4.2.1. Komunikasi *Wiimote*

Wiimote berkomunikasi dengan laptop menggunakan antarmuka *Bluetooth*. Oleh karena itu, laptop yang akan digunakan harus memiliki perangkat *Bluetooth* terlebih dahulu, baik itu yang telah terintegrasi di dalam laptop atau dapat juga ditambahkan perangkat *USB Bluetooth* bagi laptop yang belum memilikinya. Adapun laptop yang digunakan dalam penelitian ini telah terintegrasi dengan *Bluetooth* dan menggunakan *Toshiba Bluetooth Stack* sebagai *driver*-nya. Gambar 6 menunjukkan diagram alir komunikasi *Wiimote* dengan laptop.

Sebelum dilakukan proses perhitungan posisi dan penggambaran grafik, dilakukan pengambilan data dari sensor yang terdapat pada *Wiimote*. Data yang diambil adalah data *pitch* berupa besaran sudut kemiringan dalam derajat. Nilai sudut ini bernilai positif ketika robot sedang berjalan menanjak dan bernilai negatif ketika robot sedang berjalan menurun. Secara *default*, fitur *motion sensing* pada *Wiimote* berada dalam keadaan tidak aktif, sehingga untuk mengecek apakah *Wiimote* mengalami perubahan kemiringan atau tidak, fitur tersebut perlu diaktifkan terlebih dahulu. Aktivasi fitur *motion sensing* ini dapat dilakukan dengan memanggil fungsi `wiimote_motion_sensing()`.

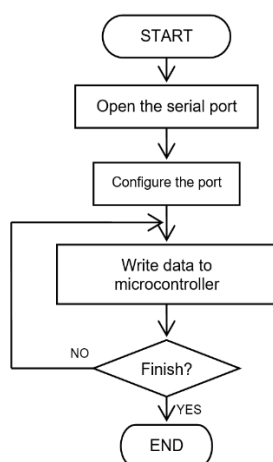
```
wiimote_motion_sensing(wiimotes[0],1);
```

Setelah fitur *motion sensing* diaktifkan, pengecekan terhadap perubahan kemiringan yang dialami oleh *Wiimote* dapat dilakukan dengan memeriksa *event* yang terjadi pada *Wiimote* dengan menggunakan baris perintah sebagai berikut:

```
if (wiimotes[0]->event == WIIUSE_EVENT)
{
    if (WIIUSE_USING_ACC(wiimotes[0]))
    {
        pitch = wiimotes[0]->orient.pitch;
    }
}
```

Dari baris program di atas dapat dilihat bahwa ekspresi `if` yang pertama akan bernilai TRUE apabila telah terjadi *event* pada `wiimotes[0]`. Jika benar, ekspresi `if` yang kedua akan memeriksa apakah *event* tersebut terjadi karena perubahan kemiringan pada *Wiimote*. Jika benar terjadi karena perubahan kemiringan, maka program akan menyimpan nilai sudut kemiringan *Wiimote* ke dalam variabel `pitch`. Nilai sudut `pitch` ini akan digunakan dalam menentukan koordinat z dalam penggambaran plot tiga dimensi lintasan yang dilewati robot. Jika nilai z terlebih dahulu diinisiasi dengan 0, maka koordinat z selanjutnya (z') setelah robot bergerak selama selang waktu Δt dengan v konstan dan sudut `pitch` θ dapat dicari dengan menggunakan persamaan berikut:

$$z' = z + v\Delta t \sin \theta \quad (12)$$



Gambar 7. Diagram Alir Komunikasi Mikrokontroler



Gambar 8. Posisi Wiimote di Atas Mobile Robot



Gambar 9. Pengujian di Dalam Laboratorium

Untuk menentukan setiap koordinat (x, y, z) pada penggambaran plot tiga dimensi, nilai (x, y, z) awal terlebih dahulu diinisiasi dengan $(0, 0, 0)$. Kemudian dilakukan proses iterasi dari persamaan (1) sampai dengan (12) setiap satu detik, sehingga ketika diimplementasikan ke dalam program, nilai Δt pada persamaan (5), (6), serta (9) sampai dengan (12) adalah 1 dan diperoleh koordinat (x, y, z) baru pada setiap detiknya. Nilai-nilai koordinat ini disimpan di dalam *array* untuk kemudian digambarkan sebagai plot tiga dimensi dengan menggunakan *library* OpenCV.

4.2.2. Komunikasi Mikrokontroler

Mikrokontroler berkomunikasi dengan laptop melalui port serial dengan menggunakan antarmuka

kabel *USB to serial*. Gambar 7 menunjukkan diagram alir komunikasi mikrokontroler.

Komunikasi serial ini pada aplikasinya digunakan untuk mengirimkan suatu perintah kepada mikrokontroler untuk menggerakkan *mobile robot*. Data yang dikirimkan terdiri atas tiga komponen, yaitu kecepatan motor kiri, kecepatan motor kanan, dan orientasi dari kedua motor. Sementara itu, mikrokontroler bertugas untuk selalu siap menerima data yang dikirimkan oleh laptop dari port serial.

5. Pengujian

Sebelum melakukan pengujian, ada beberapa hal yang perlu dipersiapkan. Persiapan meliputi hal-hal seperti menyiapkan koneksi *Bluetooth* antara *Wiimote* dengan laptop serta pengaturan tata letak *Wiimote* dan laptop di atas *mobile robot*.

Mula-mula aktifkan mode pencarian pada *Wiimote* terlebih dahulu dengan cara menekan tombol 1 dan 2 secara bersamaan. Kemudian pada menu *Bluetooth* yang terdapat pada laptop, aktifkan koneksi dengan "Nintendo RVL-CNT-01" yang merupakan representasi dari *Wiimote*.

Kemudian letakkan laptop dan *Wiimote* di atas *mobile robot* dengan posisi *Wiimote* menghadap ke depan seperti ditunjukkan oleh Gambar 8. Setelah laptop dan *Wiimote* diposisikan dengan benar, maka langkah selanjutnya adalah menyambungkan kabel *USB to serial* ke port USB laptop dan mulai menjalankan program yang telah diimplementasikan.

Ketika program pertama kali dijalankan, program akan membuka port serial terlebih dahulu, diikuti dengan membuat koneksi dengan *Wiimote*. Apabila berhasil, maka akan tampil pesan "Connected to *Wiimote*" pada jendela konsol. Selanjutnya dilakukan pengujian pada beberapa tempat sebagai berikut:

a. Laboratorium AVR

Lokasi ini merupakan lokasi pengujian pertama yang dilakukan. Di dalam laboratorium, lingkungan dapat dibuat sedemikian rupa sehingga mudah dilakukan pengukuran untuk perbandingan antara hasil yang diperoleh dengan keadaan sebenarnya. Pengujian ini hanya terdiri dari satu jalur lurus yang menanjak seperti yang ditunjukkan oleh Gambar 9.

Pada hasil pemetaan yang ditunjukkan oleh Gambar 10 terlihat sedikit *noise* berupa permukaan yang tidak sepenuhnya halus dan rata. Hal ini disebabkan karena *mobile robot* mengalami sedikit getaran pada saat menanjak, sehingga hasil pembacaan *Wiimote* menjadi sedikit terganggu. Kemudian pada hasil akhir sumbu X menunjukkan nilai 314, yang berarti robot telah maju sejauh 314 cm pada bidang alasnya (bukan pada bidang miring), dan kondisi sebenarnya adalah sejauh 330 cm atau 11 grid lantai. Sedangkan pada sumbu Z menunjukkan nilai 31, yang berarti robot lebih tinggi 31 cm dari posisi

awal, dan kondisi sebenarnya adalah 29 cm. Perbedaan yang tidak terlalu signifikan ini menunjukkan bahwa sistem telah berfungsi cukup baik dan akurat.

b. *Pelataran parkir*

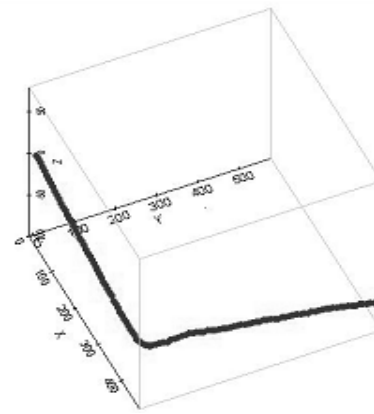
Lokasi kedua bertempat di pelataran parkir depan gedung Ahmad Bakrie (dulu disebut Labtek VIII). Pengujian ini melibatkan satu belokan ke kiri dan permukaannya menurun seperti yang ditunjukkan oleh Gambar 11, sedangkan hasil pemetaan di lokasi ini dapat dilihat pada Gambar 12.

c. *Selasar koridor*

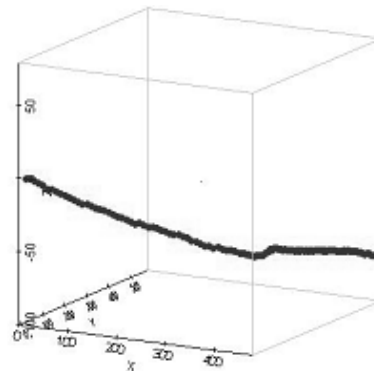
Lokasi ketiga bertempat di selasar koridor menuju ke pelataran parkir depan gedung Ahmad Bakrie (dulu disebut Labtek VIII). Pengujian ini terdiri dari satu belokan ke kiri ditambah satu belokan ke kanan dan permukaannya menurun seperti yang ditunjukkan oleh Gambar 13, sedangkan hasil pemetaan di lokasi ini dapat dilihat pada Gambar 14.

d. *Ramp Campus Center*

Lokasi keempat bertempat di *ramp Campus Center*. Gambar 15 menunjukkan hasil pemetaan di lokasi ini, sedangkan Gambar 16 menunjukkan keadaan lokasi yang terdiri dari beberapa belokan dan permukaannya menurun.

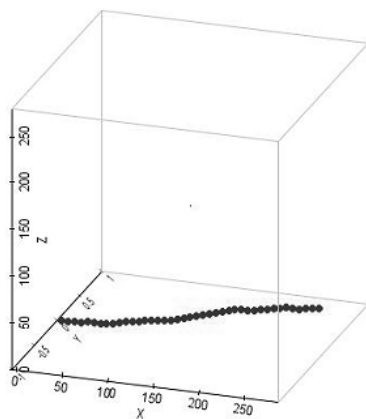


(a)

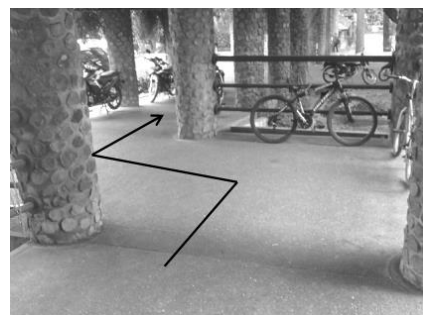


(b)

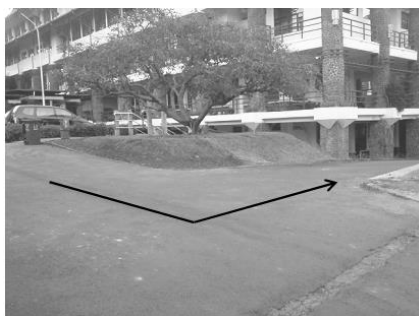
Gambar 12. Hasil Pemetaan di Pelataran Parkir



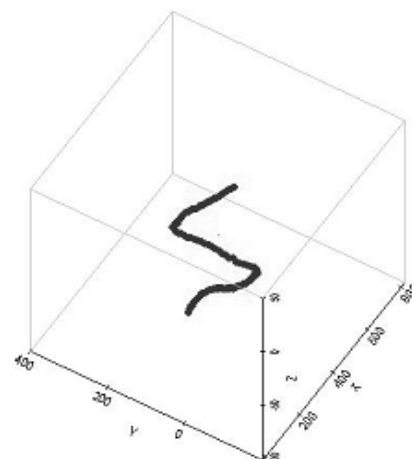
Gambar 10. Hasil Pemetaan di Dalam Laboratorium



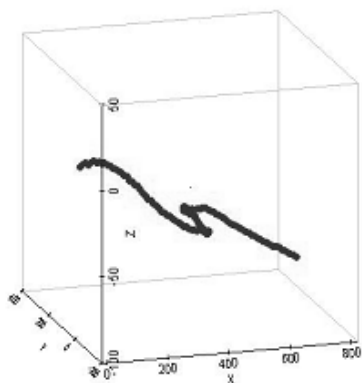
Gambar 13. Selasar Koridor



Gambar 11. Pelataran Parkir

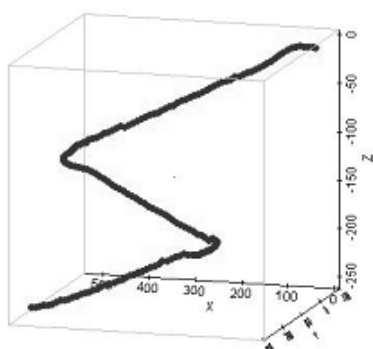


(a)

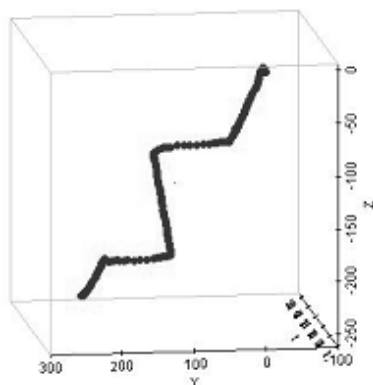


(b)

Gambar 14. Hasil Pemetaan di Selasar Koridor



(a)



(b)

Gambar 15. Hasil Pemetaan di Ramp Campus Center



Gambar 16. Ramp Campus Center

Tabel 1. Pemetaan Jarak pada Sumbu X, Y, dan Z

Lokasi	Hasil Pemetaan (cm)			Jarak sebenarnya (cm)		
	ΔX	ΔY	ΔZ	ΔX	ΔY	ΔZ
1	314	-	31	330	-	29
2	476	570	65	500	600	60
3	837	210	80	900	200	75
4	514	215	260	550	200	250

Tabel 2. Perhitungan Selisih dan Error

Lokasi	Selisih (cm)			Error		
	ΔX	ΔY	ΔZ	ΔX	ΔY	ΔZ
1	16	-	2	4,8%	-	6,9%
2	24	42	4	4,8%	7%	6,7%
3	63	10	5	7%	5%	6,7%
4	36	13	18	6,5%	6,5%	7,2%

Perbandingan hasil pemetaan dan keadaan sebenarnya dari keempat lokasi tersebut ditunjukkan pada Tabel 1, sedangkan perhitungan selisih dan error ditunjukkan pada Tabel 2. Dari tabel tersebut dapat disimpulkan bahwa error berkisar antara 4,8% sampai 7,2% dengan rata-rata error adalah 6,3%. Sehingga rata-rata akurasi yang diperoleh adalah $100\% - 6,3\% = 93,7\%$.

6. Kesimpulan

Sistem yang dibangun dapat melakukan tugas dan fungsinya dengan baik, yaitu membuat peta kontur lingkungan yang mendekati keadaan yang sebenarnya dengan akurasi rata-rata sebesar 93%. Beberapa noise yang terdapat pada peta disebabkan oleh permukaan tanah yang sedikit bergelombang. Hal itu dapat mengganggu pembacaan informasi dari Wiimote karena accelerometer yang digunakan memiliki tingkat sensitivitas yang cukup tinggi.

Sistem yang dibangun dapat digunakan dan dikombinasikan dengan metoda pemetaan lain untuk membentuk interpretasi terhadap lingkungan yang lebih lengkap pada penelitian yang lebih lanjut.

Daftar Pustaka

- [1] Analog Devices, "ADXL330 Accelerometer Datasheet", Diakses pada Februari 2015 http://www.analog.com/static/imported-files/data_sheets/ADXL330.pdf
- [2] Borenstein, J., B. Everett, dan L. Feng, "Navigating Mobile Robots: Systems and Techniques", Massachusetts, A. K. Peters, Ltd., 1996.
- [3] Dudek, Gregory, dan Michael Jenkin, "Computational Principles of Mobile Robotics", United Kingdom, Cambridge University Press, 2010.
- [4] Thrun, Sebastian, Wolfram Burgard, dan Dieter Fox, "Probabilistic Robotics", Seattle, University of Washington, 2000.
- [5] Thrun, Sebastian, "Robotic Mapping: A Survey", Pittsburgh, Carnegie Mellon University, 2002.

- [6] Wardhana, Aryo, "*Perancangan dan Realisasi Platform serta Sistem Penggerak Autonomous Mobile Robot*", Bandung, Institut Teknologi Nasional, 2007.
- [7] Zunt, Dominik, "*Who Did Actually Invent The Word “Robot” and What Does It Mean?*", Karel Čapek website, diakses pada 13 Januari 2015.