

Object Tracking Pergerakan Benda Berdasarkan Observasi Posisi dan letak *Obstacle* pada Lingkungan Statis

Gillang Al Azhar^{1*}, Adi Candra Kusuma², Leonardo Kamajaya³, Syarifatul Izza⁴

^{1,2,3}Politeknik Negeri Malang, Malang

⁴Politeknik Unisma Malang, Malang

*Penulis Korespondensi, email: gillang_al_azhar@polinema.ac.id

Received:25/09/2023

Revised:01/10/2023

Accepted:03/10/2023

Abstract. The displacement of an object can be tracked by using several sensors attached to the object's body. The sensor data obtained can be processed using several algorithms, including transformation of object movement, geometric transformation, and kinematics of object movement in a 2-dimensional plane. Based on these 3 algorithms, it is possible to estimate the actual position of a moving object by utilizing obstacles within the scope of the object's movement area. In the research carried out, a simulation program was created to simulate the implementation of these 3 algorithms in carrying out object tracking on a moving object. The simulation was created to test the performance of an algorithm created to track objects moving in a 2-dimensional plane. The results obtained from the simulation show that the average RMS (Root Mean Square) error between the actual coordinate point and the estimated coordinate point is 0.025m. Based on these results, it is shown that the implementation of the 3 algorithms used is able to provide accurate object position estimation data.

Keywords: Motion transformation, Geometric transformation, Kinematic movement, Object tracking

Abstrak. Perpindahan sebuah benda dapat di lacak dengan memanfaatkan beberapa sensor yang terpasang pada tubuh dari benda tersebut. Data sensor yang didapatkan dapat diolah dengan memanfaatkan beberapa algoritma, diantaranya adalah transformasi pergerakan benda, transformasi geometris, dan kinematik pergerakan benda pada bidang 2 dimensi. Berdasarkan 3 algoritma tersebut dapat diketahui estimasi posisi aktual dari benda yang bergerak dengan memanfaatkan *obstacle* yang ada di lingkup area pergerakan benda tersebut, dimana pada penelitian yang dilakukan dibuat simulasi program untuk mensimulasikan implementasi 3 algoritma tersebut dalam melakukan *object tracking* pada sebuah benda yang bergerak. Simulasi dibuat untuk menguji peforma algoritma yang dibuat untuk melakukan *tracking* pada benda yang bergerak pada bidang 2 dimensi. Hasil yang didapatkan dari simulasi menunjukkan bahwa rata – rata error RMS (Root Mean Square) antara titik koordinat aktual dan titik koordinat estimasi bernilai 0.025m, berdasarkan hasil tersebut ditunjukkan bahwa implementasi 3 algoritma yang digunakan mampu memberikan akurasi data estimasi posisi benda secara akurat.

Kata Kunci: Transformasi gerak, Transformasi geometris, Kinematik pergerakan, *Object tracking*

I. PENDAHULUAN

Melakukan *tracking* terhadap sebuah benda bergerak dapat dilakukan dengan berbagai macam cara. Metode yang paling mudah adalah dengan menggunakan berbagai macam sensor [1]. Sensor yang digunakan dapat terpasang pada objek benda yang ingin di *track* posisinya [2] [3], dapat juga terpasang diluar objek benda tersebut [4] [5]. Untuk pemasangan sensor diluar benda, dibutuhkan penempatan sensor yang mampu memberikan seluruh informasi didalam lingkup area yang

digunakan sebagai tempat pergerakan benda [6], jenis sensor yang digunakan dapat berupa *flat camera*, *omni-vision camera*, RFID, dll. Metode ini banyak diimplementasikan pada *tracking* yang dilakukan untuk mendeteksi posisi benda dilapangan, dimana pada tubuh objek yang dideteksi dipasangkan sebuah *marker* [7] atau benda lain yang dapat dideteksi oleh sensor eksternal, sehingga didapatkan data posisi benda tersebut. Namun kekurangan dari metode tersebut adalah sulitnya perakitan dan pengkondisian pemasangan sensor dilapangan. Untuk sensor yang terpasang

pada objek benda bergerak, dapat berupa sensor yang memberikan informasi terkait jarak *obstacle* [8], dapat juga memberikan informasi terkait jarak pergerakan benda tersebut melalui actuator yang terpasang pada objek tersebut seperti roda [9] [10], motor [11] [12], propeller, dll. Jenis sensor yang digunakan dapat berupa sensor laser, lidar, rotary encoder [13] [14], IMU [15], dll. Untuk metode ini memiliki kemudahan untuk dilakukan, karena sensor – sensor yang digunakan langsung dipasangkan di tubuh benda yang dideteksi. Pada penelitian yang dilakukan, berfokus pada *tracking* posisi yang memanfaatkan sensor yang terpasang pada objek benda yang dideteksi [16] [17]. Untuk dapat menentukan estimasi posisi dari benda melalui sensor yang terpasang pada tubuh benda tersebut, dibutuhkan beberapa algoritma yang mampu mendukung pengolahan data jarak menjadi posisi, seperti algoritma transformasi pergerakan, transformasi geometris [18], dan kinematik gerak benda. Untuk menguji coba implementasi beberapa algoritma tersebut yang digunakan untuk melakukan *tracking* posisi benda, dibuat sebuah simulasi program yang dikondisikan seperti kondisi sebuah benda yang berjalan di sebuah lingkup area. Diharapkan dengan menggunakan data hasil simulasi dapat didapatkan gambaran akurasi peforma implementasi beberapa algoritma untuk melakukan *tracking* pergerakan benda. Pada simulasi yang dibuat, dikondisikan pergerakan benda dan pembacaan sensor yang digunakan masih bersifat ideal.

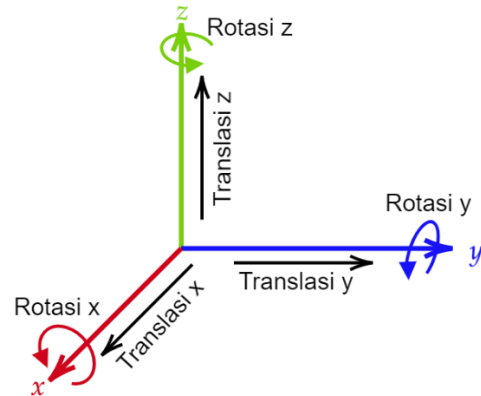
II. METODOLOGI

Beberapa pembahasan dan dasaran teori yang mendukung dan mendasari pelaksanaan penelitian yang dilakukan dituliskan sebagai berikut

A. Transformasi Pergerakan Bidang 2 Dimensi

Sebuah benda jika diberikan sebuah gaya, maka akan memiliki kemampuan untuk bergerak, baik secara translasi maupun rotasi. Pergerakan yang dilakukan oleh benda tersebut dapat diproyeksikan kedalam sumbu kartesius, dimana pada sumbu tersebut meliputi posisi dalam 3 dimensi, yaitu sumbu x , sumbu y , dan sumbu z . Dimana masing – masing sumbu dapat memproyeksikan 2 gerak,

yaitu translasi dan rotasi. Gerak translasi merupakan pergerakan yang secara visual berpindah secara linier terhadap masing – masing sumbu kartesius, sedangkan gerak rotasi merupakan pergerakan yang membuat benda bergerak secara berputar terhadap masing – masing sumbu kartesius. Hal ini dapat dilihat pada gambar 1 berikut



Gambar 1. Pergerakan pada sumbu kartesius berdasarkan referensi sumbunya

Sehingga berdasarkan pergerakan tersebut dapat dituliskan persamaan yang mewakili posisi dan rotasi dari sebuah benda yang bergerak, atau dapat disebut dengan pose ($\mathbf{x}$), yaitu

$$\mathbf{x} = \begin{bmatrix} \xi \\ \mathbf{M} \end{bmatrix} \quad (1)$$

Dimana ξ merupakan data koordinat dari posisi benda yaitu $[x, y, z]^T$, dan $\mathbf{M}$ adalah sudut rotasi benda terhadap masing – masing sumbu kartesius yaitu $[\theta_x, \theta_y, \theta_z]^T$. Namun Dalam kasus tertentu, sebuah benda hanya dapat bergerak didalam sumbu 2 dimensi saja, sehingga untuk benda yang hanya bergerak pada bidang 2 dimensi, (1) dapat disederhanakan, dimana untuk benda yang hanya bergerak pada bidang 2 dimensi nilai translasi pada bidang z bernilai 0, dan rotasi terhadap sumbu x dan y juga dianggap bernilai 0, sehingga dapat dituliskan

$$\mathbf{x} = \begin{bmatrix} [x, y, 0]^T \\ [0, 0, \theta_z]^T \end{bmatrix} \quad (2)$$

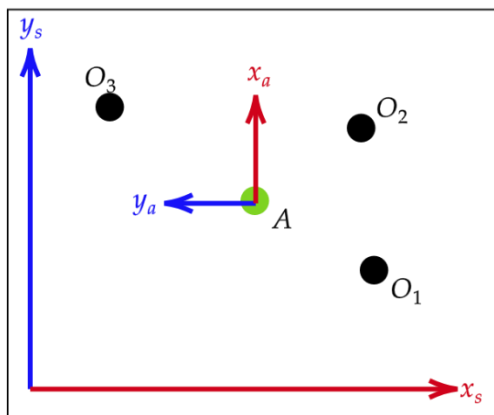
$$= \begin{bmatrix} [x, y]^T \\ \theta_z \end{bmatrix} \quad (3)$$

$$= \begin{bmatrix} x \\ y \\ \theta_z \end{bmatrix} \quad (4)$$

Sehingga untuk pergerakan benda didalam bidang 2 dimensi dapat dituliskan posisi dan putarannya dalam bentuk matriks seperti pada (4).

B. Transformasi Geometris Posisi Benda

Posisi sebuah benda didalam sebuah lingkungan dapat diketahui melalui data – data benda lain yang juga berada pada lingkungan yang sama. Hal ini dapat dilakukan bila lingkungan yang terkait digambarkan menjadi sebuah area yang dapat dipetakan kedalam sebuah sumbu kartesius. Dimana titik acuan (sumbu $x = 0, y = 0$) harus bersifat statis dan tidak boleh berpindah. Hal ini dapat digambarkan sebagai berikut



Gambar 2. Ilustrasi penggambaran lingkungan benda kedalam sumbu kartesius

Pada gambar 2, dapat dilihat bahwa lingkungan yang terkait digambarkan dengan persegi dengan garis hitam, yang mana didalamnya terdapat sebuah benda yang bergerak dan memiliki arah pergerakan terhadap sumbu utama yang dinotasikan dengan A_S , serta terdapat 3 buah benda lainnya (*obstacle*) yang digunakan sebagai acuan posisi yang dinotasikan sebagai O_1 untuk *obstacle* 1, O_2 untuk *obstacle* 2, dan O_3 untuk *obstacle* 3. Dimana untuk masing – masing objek, memiliki 2 buah koordinat posisi, yaitu koordinat *Obstacle* terhadap sumbu utama ($\{S\}$) atau dapat dinotasikan sebagai O_S , dan koordinat Objek terhadap benda A atau dapat dinotasikan sebagai O_A . Sehingga berdasarkan gambar tersebut dapat dibuat sebuah persamaan yaitu

$$O_S = R_{A/S} \cdot O_A + A_S \quad (5)$$

Dimana $R_{A/S}$ merupakan matrix rotasi dari benda A terhadap sumbu utama ($\{S\}$), yang jika dijabarkan menjadi berikut

$$R_{A/S} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \quad (6)$$

dengan θ merupakan sudut yang dibentuk antara arah hadap benda A terhadap sumbu x dari $\{S\}$. Berdasarkan implementasinya, nilai variabel yang dibutuhkan untuk dicari nilainya adalah A_S , dimana posisi benda dicari nilai posisinya berdasarkan *obstacle* yang ada di lingkungan, dimana untuk posisi *obstacle* – *obstacle* tersebut telah diketahui nilainya (Studi Kasus pada lingkungan statis), sehingga dalam kondisi seperti ini, dapat dimanfaatkan (5) untuk disubstitusikan agar dapat dicari nilai A_S , sehingga didapatkan

$$A_S = O_S - R_{A/S} \cdot O_A \quad (7)$$

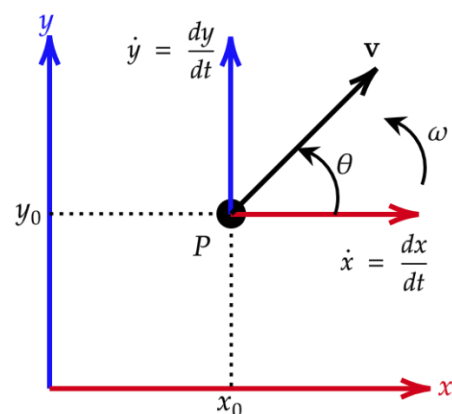
C. Kinematik Pergerakan Benda pada Bidang 2 dimensi

Pada dasarnya sebuah benda yang bergerak dapat dibuat pemodelan gerakanya berdasarkan vector kecepatan dari benda tersebut. Untuk pergerakan translasi dan rotasi sebuah benda dapat dituliskan persamaan gerakanya sebagai berikut

$$s = s_0 + v t \quad (8)$$

$$\theta = \theta_0 + \omega t \quad (9)$$

dimana (8) merupakan persamaan gerak translasi dan (9) merupakan persamaan gerak rotasi. Gambaran untuk persamaan kinematik pergerakan benda pada bidang 2 dimensi dapat dilihat pada gambar 3 berikut



Gambar 3. Gerak translasi dan rotasi 2D

berdasarkan gambar tersebut, vektor $\mathbf{v}$ menotasikan kecepatan translasi dari benda P . Vektor $\mathbf{v}$ terdiri dari

$$v_x = \dot{x} = \frac{dx}{dt} \quad (10)$$

Persamaan (10) merupakan kecepatan translasi pada sumbu x , serta

$$v_y = \dot{y} = \frac{dy}{dt} \quad (11)$$

Persamaan (11) merupakan kecepatan translasi pada sumbu y . Dengan benda P yang memiliki arah Gerakan, maka (10) dan (11) dapat dituliskan sebagai berikut

$$v_x = v \cos \theta \quad (12)$$

$$v_y = v \sin \theta \quad (13)$$

Mengacu pada (4), Kecepatan translasi dan rotasi dapat dijadikan satu vektor dan dapat dituliskan sebagai berikut

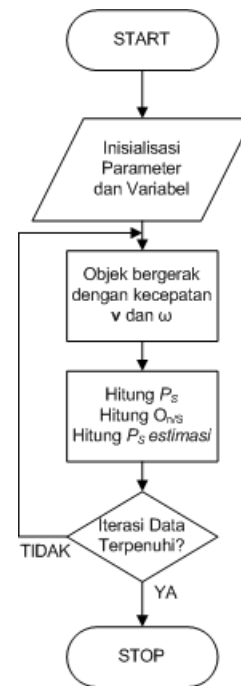
$$\dot{\xi} = [\mathbf{v}, \dot{\theta}]^T \quad (14)$$

$$= [v_x, v_y, \omega]^T \quad (15)$$

D. Perancangan Simulasi

Simulasi yang dirancang, adalah untuk mengetahui performa dari implementasi algoritma transformasi pergerakan, transformasi geometris posisi benda serta kinematic pergerakan benda pada bidang 2 dimensi. Simulasi dibuat untuk mengetahui estimasi posisi dari sebuah benda yang bergerak melalui posisi *obstacle* yang ada pada lingkungan yang telah dipetakan kedalam sumbu kartesius 2 dimensi. Beberapa variabel yang diujicobakan didalam simulasi adalah sebagai berikut

- P_S : Posisi koordinat benda P terhadap $\{S\}$
- $\bar{P}_S$: Estimasi Posisi Koordinat benda P berdasarkan posisi *obstacle*
- $O_{n/S}$: Posisi koordinat *obstacle* terhadap $\{S\}$
- $O_{n/P}$: Posisi koordinat *obstacle* terhadap benda P
- θ : Sudut antara arah hadap benda P dengan sumbu x_S



Gambar 4. Flowchart jalannya simulasi

Untuk alur teknis simulasi yang dijalankan tergambar pada flowchart yang ada pada gambar 4, untuk penjelasannya adalah sebagai berikut

1. Posisi awal benda ditentukan terlebih dahulu
2. Benda diberikan kecepatan $\mathbf{v}$ dan ω . Pergerakan benda dihitung dengan kinematik gerak benda, dimana data ini dijadikan nilai sebenarnya dari posisi benda (P_S)
3. Posisi koordinat *obstacle* terhadap benda P ($O_{n/S}$) dihitung sebagai jarak dengan menggunakan *euclidean distance* dengan berdasarkan pada P_S dan $O_{n/S}$
4. Estimasi Posisi benda ($\bar{P}_S$) dihitung dengan menggunakan (7) berdasarkan masing – masing *obstacle* dan P_S dihitung dengan menggunakan (8) dan (9).
5. Nilai akhir $\bar{P}_S$ dihitung berdasarkan nilai rata – rata yang didapatkan dari masing – masing posisi *obstacle*.

III. HASIL DAN PEMBAHASAN

A. Simulasi tracking posisi benda dengan kinematik gerak benda

Pada bagian ini, disimulasikan sebuah benda yang diberikan kecepatan pergerakan translasi ($\mathbf{v}$), serta

kecepatan sudut arah hadap (ω) untuk bergerak pada sumbu x dan y dari sebuah lingkungan. Berdasarkan nilai kecepatan tersebut akan dilakukan *tracking* perpindahan posisi dengan memanfaatkan (8) dan (9). Kecepatan translasi yang diinputkan akan diolah menjadi kecepatan pada masing – masing sumbu menggunakan (12) dan (13). Berikut listing program yang digunakan untuk simulasi kinematika pergerakan benda

```
import numpy as np

pos_x = 0.0
pos_y = 0.0
th = np.radians(0.0)

v = 2.0 # m/s
w = 1.0 # rad/s
ts = 0.1 # s

for i in range(0, 7):
    vx = v * np.cos(th)
    vy = v * np.sin(th)

    pos_x = pos_x + vx*ts
    pos_y = pos_y + vy*ts
    th = th + w*ts

    print(pos_x)
    print(pos_y)
    print(np.degrees(th))
    print("=====\n")
```

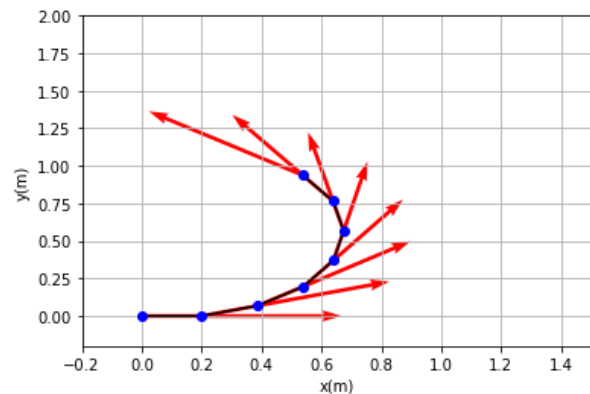
Gambar 5. Contoh Listing Program untuk pengujian kinematik gerak benda

Berdasarkan contoh listing program yang ditunjukkan pada gambar 5, dilakukan untuk melakukan *tracking* sebanyak 7 kali dengan kecepatan konstan yaitu $v = 2.0$ dan $\omega = 3.5$, berikut hasil update posisi yang didapatkan,

Tabel 1. Data *tracking* posisi dan arah hadap benda

Sampling	Posisi x (m)	Posisi y (m)	Arah hadap (°)
0	0.0	0.0	0
1	0.2	0.0	20.05
2	0.38	0.06	40.11
3	0.54	0.20	60.16
4	0.64	0.37	80.21
5	0.67	0.56	100.26
6	0.63	0.76	120.32
7	0.54	0.93	140.4

Berdasarkan hasil yang ditunjukkan pada tabel 2, dimana didapatkan data koordinat x , koordinat y , dan arah hadap objek/benda yang bergerak dengan kecepatan linear 2.0 m/s dan kecepatan rotasi 3.5 rad/s selama 7 kali iterasi atau selama 0.7 detik. Berdasarkan data tersebut dapat divisualisasikan seperti pada gambar 6 berikut



Gambar 6. Representasi posisi benda dan arah hadapnya terhadap sumbu kartesius lingkungan

Terlihat bahwa program simulasi mampu menunjukkan pergerakan yang rasional dengan kondisi aktual yang diinginkan, yaitu dengan nilai v yang positif dan ω yang positif, akan menghasilkan pergerakan benda yang membentuk pola berbelok kearah kiri.

B. Simulasi pembacaan jarak obstacle ke benda berdasarkan titik koordinat terhadap sumbu utama

Pada simulasi ini, digunakan untuk menguji algoritma penghitungan jarak dari posisi aktual benda dengan posisi *obstacle* yang terletak dilapangan. Pengukuran jarak disimulasikan dengan cara menghitung nilai *euclidean distance* dari 2 titik koordinat yang dimiliki, yaitu koordinat aktual objek dan koordinat actual dari *obstacle*. Untuk persamaan yang digunakan adalah

$$d = \sqrt{(O_x - P_x)^2 + (O_y - P_y)^2} \quad (16)$$

serta untuk mengetahui posisi *obstacle* terhadap benda dapat dihitung dengan

$$\theta_{obs/s} = \tan^{-1} \left(\frac{O_y - P_y}{O_x - P_x} \right) \quad (17)$$

$$\theta_{obs/A} = \theta_{obs/S} - \theta_{A/S} \quad (18)$$

Untuk listing program simulasi yang digunakan untuk mendapatkan data pada tabel 2, dapat dilihat pada gambar 7.

```
import numpy as np

x_robot = 0.5
y_robot = 0.75
h_robot = np.radians(45.)

x_obs = 2.25
y_obs = 3.15

ed = (x_robot - x_obs)**2 + (y_robot - y_obs)**2
d = np.sqrt(ed)
th = np.arctan2(y_obs-y_robot, x_obs - x_robot)
thr = th - h_robot

print(d)
print(np.degrees(thr))
```

Gambar 7. Listing program untuk pengujian euclidean distance

Berdasarkan listing program tersebut ,didapatkan hasil pengujian (17) dan (18) kedalam implementasi simulasi program yang dibuat

Tabel 2. Hasil pengujian implementasi *Euclidean distance* antara posisi objek dengan *obstacle*

	Koordinat Objek	Koordinat Obstacle	Jarak (m)	Posisi terhadap benda (°)
1	x: 1.5 y: 2.5 θ: 45°	x: 3.25 y: 3.0	1.82	-29.05
2	x: 2.5 y: 1.75 θ: 90°	x: 1.25 y: 5.4	3.85	18.9
3	x: 0.5 y: 0.75 θ: 120°	x: 2.25 y: 3.15	2.97	-66.09
4	x: 0.5 y: 0.75 θ: 45°	x: 2.25 y: 3.15	2.97	8.90
5	x: 0.5 y: 0.75 θ: 0°	x: 4.25 y: 0.75	3.75	0.0
6	x: 0.5 y: 0.75 θ: 0°	x: 0.5 y: 4.75	4.00	90.0
7	x: 2.5 y: 2.5 θ: 0°	x: 4.5 y: 4.5	2.83	45.0

Berdasarkan hasil yang ditunjukkan Tabel 2, dapat dilihat pengujian dilakukan untuk membuktikan

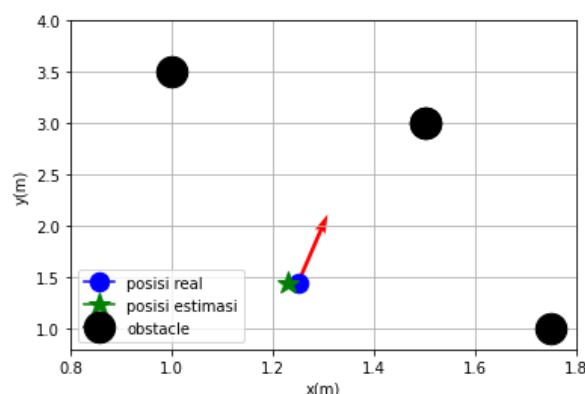
akurasi perhitungan *Euclidean distance*, dimana data hasil perhitungan jarak dan identifikasi posisi *obstacle* menunjukkan hasil yang sesuai dan rasional dengan kondisi yang sebenarnya.

C. Perbandingan hasil simulasi estimasi posisi dengan hasil simulasi posisi sebenarnya

Pada bagian ini, disimulasikan algoritma *object tracking* yang telah dirancang sesuai dengan II.D, dimana dibandingkan hasil *object tracking* aktual dengan hasil *object tracking* yang didapatkan dari data jarak antara benda dengan *obstacle* dilingkungan. Berikut data hasil simulasi yang didapatkan

Tabel 3. Data pengujian object tracking

No	Koordinat Aktual	Koordinat Estimasi	Error RMS
1	x: 2.3 y: 1.4	x: 2.33 y: 1.38	0.037
2	x: 1.5 y: 2.8	x: 1.51 y: 2.81	0.014
3	x: 1.25 y: 1.45	x: 1.23 y: 1.44	0.022
4	x: 3.75 y: 2.95	x: 3.72 y: 2.97	0.036
5	x: 4.15 y: 2.20	x: 4.13 y: 2.21	0.022
6	x: 1.85 y: 2.35	x: 1.88 y: 2.33	0.036
7	x: 1.35 y: 2.15	x: 1.35 y: 2.16	0.010



Gambar 8. Visualisasi posisi estimasi dan posisi aktual

Berdasarkan hasil yang ditampilkan pada tabel 3 dan visualisasi dari gambar 8, dapat dilihat bahwa nilai error dari data aktual dan estimasi rata – rata memiliki nilai yang kecil, yaitu berkisar antara 0.014 hingga 0.037, dimana rata – rata nilai RMSnya adalah sebesar 0.025 meter. Hal ini menunjukkan bahwa algoritma *object tracking* yang

mengimplementasikan 3 algoritma transformasi dan kinematika gerak, mampu memberikan data yang cukup akurat, hal ini terbukti dari 7 kali percobaan dengan nilai koordinat tujuan gerak yang berbeda, rata – rata memiliki nilai error RMS yang kurang dari 0.1 m.

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil yang didapatkan dari uji coba simulasi yang dilakukan, dapat disimpulkan bahwa posisi sebuah benda dapat di *tracking* dengan memanfaatkan sensor yang dipasangkan pada benda tersebut, sebagai contoh sensor jarak yang digunakan untuk membaca jarak *obstacle* di area sekitar benda objek tersebut berada, dimana data sensor tersebut dapat diolah untuk menjadi sebuah data posisi estimasi melalui implementasi algoritma transformasi dan kinematika gerak benda. Hal ini dibuktikan dari hasil simulasi yang didapatkan, dimana nilai error RMS yang didapatkan dari nilai koordinat aktual dan koordinat estimasi memiliki akurasi yang cukup baik ditunjukkan dengan nilai rata-rata error RMS sebesar 0.025 m. yang mana masih dibawah batas toleransi error sebesar 0.15 m.

Saran untuk pengembangan simulasi program adalah dengan menambahkan nilai noise pada data pembacaan jarak *obstacle*, serta dapat diujicobakan dengan menambahkan simulasi pembacaan pergerakan dengan sensor pergerakan yang juga diberikan besaran noise, sehingga hasil yang didapatkan dari simulasi bersifat tidak ideal dan mendekati kondisi yang sebenarnya di lapangan.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan oleh penulis untuk institusi Politeknik Negeri Malang yang telah menyediakan tempat untuk melakukan penelitian dan prasarana untuk menyelesaikan artikel yang dibuat.

REFERENSI

- [1] A. Elmquist and D. Negrut, "Methods and Models for Simulating Autonomous Vehicle Sensors," *IEEE Transactions on Intelligent Vehicles*, vol. 5, pp. 684-692, 2020.
- [2] G. A. Azhar, T. Winarno and S. Izza, "Implementasi g-h Filter Pada Sensor Kompas Sebagai Peningkatan Akurasi Trajectory Tracking Robot Differential Drive," *Journal of Mechanical and Electrical Technology*, vol. 1, 2022.
- [3] I. Siradjuddin, I. M. Fitriani, R. A. Asmara, M. Junus, T. S. Patma, G. A. Azhar and H. Setiawan, "A study of a discrete Bayes and a Kalman filter computational Complexity and performance in the case of 1D robot localization," *Journal of Physics: Conference Series*, vol. 1402, p. 044026, 12 2019.
- [4] G. Fabregat, J. A. Belloch, J. M. Badía and M. Cobos, "Design and Implementation of Acoustic Source Localization on a Low-Cost IoT Edge Platform," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, pp. 3547-3551, 2020.
- [5] T. Shu, S. Gharaaty, W. Xie, A. Joubair and I. A. Bonev, "Dynamic Path Tracking of Industrial Robots With High Accuracy Using Photogrammetry Sensor," *IEEE/ASME Transactions on Mechatronics*, vol. 23, pp. 1159-1170, 2018.
- [6] J. Li, M. Ran and L. Xie, "Efficient Trajectory Planning for Multiple Non-Holonomic Mobile Robots via Prioritized Trajectory Optimization," *IEEE Robotics and Automation Letters*, vol. 6, pp. 405-412, 2021.
- [7] R. Kumar, S. Singh and V. K. Chaurasiya, "A Low-Cost and Efficient Spatial-Temporal Model for Indoor Localization "H-LSTMF"," *IEEE Sensors Journal*, vol. 23, pp. 6117-6128, 2023.
- [8] G. A. Azhar, T. Winarno and A. Komarudin, "Kontrol Sudut Elevasi Robot Pelontar Softsaucer dengan Metode PID," *Jurnal Elektronika Otomasi Industri*, vol. 4, pp. 9-14, 2020.
- [9] G. A. Azhar, Sungkono, M. N. Achmadiyah and S. Izza, "Peningkatan Kestabilan Sistem Kontrol UGV melalui Optimalisasi Manajemen Core dan Free-RTOS pada ESP32," *Jurnal Elektronika dan Otomasi Industri*, vol. 10, 2023.
- [10] G. A. Azhar, T. Winarno and S. Izza, "Sistem Distribusi Data Kontrol Pada Differential Drive Mobile Robot Menggunakan Robot Operating System," *Journal of Mechanical and Electrical Technology*, vol. 1, 2022.

- [11] I. Siradjuddin, G. Azhar, S. Wibowo, F. Ronilaya, C. Rahmad and R. O. H. A. D. I. Erfan, "A General Inverse Kinematic Formulation and Control Schemes for Omnidirectional Robots," *Engineering Letters*, vol. 29, 1 2022.
- [12] I. Siradjuddin, G. A. Azhar, A. Murdani and M. L. M. Faizin, "Desain dan pemodelan kontrol kinematik pergerakan robot beroda dengan menggunakan 6 roda omni-wheels," *Jurnal Eltek*, vol. 18, pp. 116-127, 2020.
- [13] I. Siradjuddin, L. Kamajaya, S. Wibowo, A. A. Rofiq, G. A. Azhar and M. Khairuddin, "Near Real Time Simulation of an Independent Steering Independent Driving Mobile Robot," in *2021 3rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA)*, 2021.
- [14] I. Siradjuddin, T. Winarno, M. Khairuddin, M. N. Achmadiah, R. P. Wicaksono and G. Al Azhar, "An Exponential Decreased Kinematic and PID Low Level Control Schemes for an Omni-Wheeled Mobile Robot," in *2021 International Conference on Electrical and Information Technology (IEIT)*, 2021.
- [15] S. Sungkono, G. A. Azhar, A. C. Kusuma and S. Izza, "Differential Drive Mobile Robot Motion Accuracy Improvement with Odometry-Compass Sensor Fusion Implementation," *ELKHA: Jurnal Teknik Elektro*, vol. 15, 2023.
- [16] Y. Maddahi and K. Zareinia, "Nonparametric Bootstrap Technique to Improve Positional Accuracy in Mobile Robots With Differential Drive Mechanism," *IEEE Access*, vol. 8, pp. 158502-158511, 2020.
- [17] B. Kim and K. Yi, "Probabilistic and Holistic Prediction of Vehicle States Using Sensor Fusion for Application to Integrated Vehicle Safety Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 2178-2190, 2014.
- [18] I. Siradjuddin, G. A. Azhar, C. Rahmad, F. Rohman, I. M. Fitriani, E. R. K. Pradani, E. Rohadi, F. Ronilaya and R. A. Asmara, "A Non-Iterative Solution for Rigid Body Transformation Estimation," in *2018 International Conference on Applied Science and Technology (iCAST)*, 2018.