Linear Differential Driven Wheel Mobile Robot Based on MPU9250 and Optical Encoder

Ilham Rustam [1], Nooritawati Md Tahir [2,3], Ahmad Ihsan Mohd Yassin [2], Nurbaiti Wahid [1], Abdul Hafiz Kassim [1]

[1]*School of Electrical Engineering Department, College of Engineering, Universiti Teknologi MARA, Terengganu, Malaysia*
[2]*School of Electrical Engineering Department, College of Engineering, Universiti Teknologi MARA, Selangor, Malaysia*
[3]*Institute for Big Data Analytics and Artificial Intelligence (IBDAAI), Universiti Teknologi MARA, Selangor, Malaysia*

***Abstract* – In this paper, the operability of a custom-built differential drive wheel mobile robot was evaluated using a self-driving algorithm in practical application. Several key parameters need to be realized by the WMR to enable succinct self-driving algorithm evaluations and parameter derivations. The main components of the WMR development were examined in this work, specifically parameter derivations, hardware implementation, steering algorithm and WMR operability. The result obtained indicate that with proper design and careful calibration of sensors, the required driving parameters could be realized. This enables the WMR to potentially serve as data collection platform for future evaluation of self-driving algorithms.**

***Keywords* – self-driving algorithm, autonomous vehicles, sensor fusion, inertial measurement unit, wheel encoders.**

DOI: 10.18421/TEM111-04
https://doi.org/10.18421/TEM111-04

Corresponding author: Ilham Rustam,
School of Electrical Engineering Department, College of Engineering, Universiti Teknologi MARA, Malaysia.
Email: ilhamr480@uitm.edu.my

Received: 02 October 2021.
Revised: 11 January 2022.
Accepted: 17 January 2022.
Published: 28 February 2022.

1. Introduction

Amongst the most popular platform used to study the practical applications of artificial intelligence (AI) algorithms are the robotics platforms. The robotics platforms can be divided into two classifications that is fixed or mobile [1]. As opposed to fixed robot, which mainly can be found in industrial settings, mobile robot offers much more dynamic and exhaustive computational problems, perfectly suited to challenge the AI algorithm computational capabilities. Consequently, wheeled mobile robot (WMR) was found to be one of the known being used due to its cost-effective nature that offers real world complexity in its problem. Within the research context, tracking a path trajectory, obstacle avoidance and point stabilization are amongst the most elaborated control problems to be solved in recent years. Moreover, path trajectory is the most popular due to the high nonlinearity nature and coupling property of the problem [2].

In addition, several notable works have successfully solved path tracking control problem of WMR using techniques such as sliding mode control [3], adaptive control [4], and backstepping control [5]. However, it should be noted that most of the work assumed that ideally there would be no wheel slippage or skidding occurrences during vehicle movements [6]. While this assumption is feasible during simulation, unfortunately this is not true in real world scenario or in practical applications.

Within the context of optimal path tracking and object avoidance of the WMR, wheel slips and skids could cause significant reading errors to WMR positioning and orientation [7], hence these issues need to be addressed appropriately. Some of the factors that caused wheel slips and skids include tire deformation, weak friction between road and the tires, unknown centrifugal force as well as external force acting on WMR [8], [9]. To the extent of our knowledge, limited researches were done related to this specific task among which some of it were reported in [10], [11], [12].

With this realization, this study will address these issues via a practical setting instead. Therefore, a differential WMR is constructed using currently accessible materials. However, before any attempt can be made in analysing the proposed algorithm performances, a mechanism that enables the detection of slip or skid is required. Conversely, although some works proposed the use of GPS sensor, the implementation of the GPS has physical limitation since it could not be used indoors or inside a tunnel [13]. As such, image processing or motor encoder protocol becomes a more viable solution that offers redundancy in detecting and manoeuvring the WMR during occurrences of such events.

Hence, to investigate and examine the feasibility of the said redundancy protocol, this study proceeds to design, construct and analyse an open source based differential drive WMR platform. This includes the Inertial Measurement Unit (IMU), optical motor encoders and other sensors that enable the real-time data monitoring and logging capabilities.

2. Research Background

Kinematics of wheel slips and skids can be derived using mathematical and physics equation that can be found as elaborated in literatures [14], [15], [16]. However, most of the studies limit their work without incorporating the kinematics due to its contribution towards the WMR dynamic complexity and nonlinearity that led to an extensive modelling computation cost [17]. The effect of incorporating wheel slip or skid can thus be observed through the Lagrange equation of the dynamic model, which is defined as equation of motion through consideration of the kinetic and potential energies of a given system. Without considering the wheel slip or skid, the resultant dynamic model can be written as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} = B(q)u + A^T\lambda \qquad (1),$$

where $M(q) \in \Re^{n\times n}$ is called the inertia matrix of the system, $C(q,\dot{q}) \in \Re^{n\times 1}$is the centrifugal and Coriolis matrix, $B(q) \in \Re^{n\times(n-m)}$ and $u \in \Re^{(n-m)\times 1}$ are the input transformation matrix and input vector respectively. $\lambda \in \Re^{m\times 1}$ is the Lagrange multipliers vector and $A(q) \in \Re^{m\times n}$ is the constraint matrix adjoined to the dynamic equation.

By considering the wheel slip or skid, the resultant dynamic model would then have to include a new state due to wheel slip such that:

$$S^T MS\dot{v} + S^T M\dot{S}v + S^T CSv = S^T Bu \qquad (2),$$

where $S(q) \in \Re^{n\times(n-m)}$ is a full rank matrix formed by smooth and linearly independent vector fields and $v(t) \in \Re^{(n-m)\times 1}$ is the vector of time functions for all t. Equation (2) is achieved simply by introducing ζ_i and η_i, which is the longitudinal slip displacement and the lateral slip displacement of the i-th wheel of WMR along with the dynamic equation and eliminating the Lagrange multipliers in equation (1).

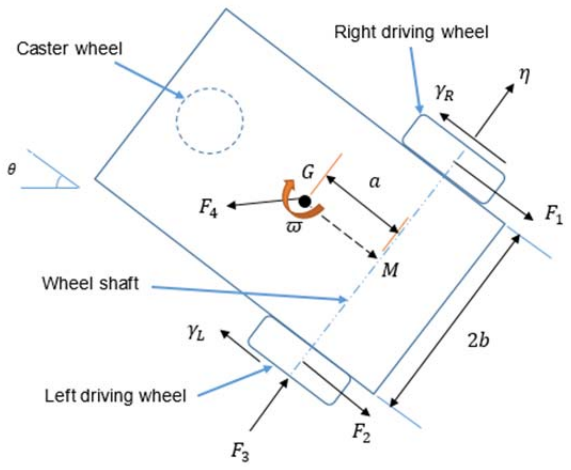

Figure 1. Differential drive wheel mobile robot with castor wheel

Equation (2) can then be elaborated by considering a nonholonomic two wheeled WMR with a castor wheel as shown in Figure 1. Centre mass of the platform is G with its travel coordinates (x_g, y_g), while M is the wheel shaft midpoint with coordinates (x_M, y_M). F_1 and F_2 are total longitudinal friction forces at the right and left wheel respectively. F_3 is the total lateral friction force along wheel shaft. F_4 and ϖ are external force and moment acting on G respectively. Consecutively, γ_L and γ_R is the longitudinal slip of the left and right wheels respectively while η denote the coordinate of the lateral slip along the wheel shaft. θ is the orientation of WMR.

Without consideration of wheel slip, linear and angular velocities of the WMR at point M are represented as [6], [18]:

$$\Theta = r\left(\dot{\phi}_R + \dot{\phi}_L\right)2$$
$$\mu = r(\dot{\phi}_R - \dot{\phi}_L)/(2b) \qquad (3),$$

where r is the radius of each driving wheel, b is the haft of the wheel, $\dot{\phi}_R$ and $\dot{\phi}_L$ are the angular velocities of the right and left wheel about the wheel shaft, respectively. Hence, kinematics of WMR can then be found as follows [6], [18]:

$$\dot{x}_M = \Theta\cos\theta$$
$$\dot{y}_M = \Theta\sin\theta \qquad (4),$$
$$\dot{\theta} = \mu$$

Cohesive disturbance parameters are required to enable extensive process dynamic profiling. Thus, wheel slippage and skidding kinematics are then elaborated. Under wheel slip and skid, Equations (2) and (3) would no longer stand true. Given, γ_L and γ_R, as the longitudinal slip and η as the lateral slip along the wheel shaft, the actual linear velocity of WMR along the longitudinal direction would be represented as:

$$\Omega = \frac{r(\dot{\phi}_R+\dot{\phi}_L)}{2} + \frac{\dot{\gamma}_R+\dot{\gamma}_L}{2} = \Theta + \dot{\chi}$$

$$\chi = (\gamma_L + \gamma_R)/2 \tag{5},$$

The actual angular velocity of the WMR is then computed as follows:

$$\omega = \frac{r(\dot{\phi}_R-\dot{\phi}_L)}{2b} + \frac{\dot{\gamma}_R-\dot{\gamma}_L}{2b} = \mu + \vartheta,$$

$$\vartheta = \frac{\dot{\gamma}_R-\dot{\gamma}_L}{2b} \tag{6},$$

3. Methodology

3.1. Hardware Description

In ensuring the success of any future steering or path planning algorithms, it is important for the developed WMR data plant to establish its ability to yield at minimum, the x-axis, $\dot{x}_M$, and y-axis, $\dot{y}_M$, coordinate as well as the angular velocity, ω parameter related to the travel distance made. This is to ensure proper real time WMR dynamics can quantitatively be measured followed by analyses upon successful data logging process.

Thus, several types of sensors have been identified to help obtain the required parameters. One of which is the IMU for attaining the WMR orientation, ψ or Yaw variable. An MPU9250 IMU unit was selected amongst various other IMU sensors due to the availability of gyroscope, accelerometer and magnetometer reading output with better WMR absolute orientation estimation accuracy [19]. However, calibration of its magnetometer sensor requires special attention due to its increase complexity. Improper calibration would cause drift accumulation, due to inappropriate handling of earth magnetic field and gravity direction that influences its output which overtime would result in significant error in data yield [20].

Several literatures have elaborated on the steps taken to handle the occurrences of drifts in its output. Amongst such are as reported in [21], [22], [23] that introduces the application of Attitude and Heading Reference System (AHRS) sensor fusion algorithm which utilizes the Quaternion estimation such as to filter out the aforementioned drift effect [24], [25]. This is done by rotating the body frame vector into inertial frame vector using the rotation matrix defined as:

$$\mathrm{R} = \begin{bmatrix} a^2+b^2-c^2-d^2 & 2bc-2ad & 2bd+2ac \\ 2bc+2ad & a^2-b^2+c^2-d^2 & 2cd-2ab \\ 2bd-2ac & 2cd+2ab & a^2-b^2-c^2+d^2 \end{bmatrix} \tag{7},$$

where a is the rotation scalar, b, c and d are the rotation around x-, y- and z-axis respectively. Once these parameters are defined, conversion from Quaternion into Euler Angles orientation could then be done via:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} arctan\frac{2(q_0q_1+q_2q_3)}{1-2(q_1^2+q_2^2)} \\ \arcsin(2(q_0q_2+q_3q_1)) \\ arctan\frac{2(q_0q_3+q_1q_2)}{1-2(q_2^2+q_3^2)} \end{bmatrix} \tag{8},$$

where ϕ, θ and ψ are the roll, pitch and yaw orientation. Obtaining this information, the estimation of angular velocity could then be computed by:

$$\omega = \mathrm{d}\psi/\mathrm{dt} \tag{9},$$

Next, several other techniques have been discussed on acquiring the x- and y-axis coordinates from specific WMR travel. As reported [26], this study has proposed the use of odometry data to plot distance travelled for each axis. This presents a challenge in itself due to the cumulative error in the sensor output which requires the use of filters and signal processing to remove and compute actual distance travelled.

Another alternative approach is to incorporate the use of a motor encoder sensor. This study incorporated the use of optical encoders from Pololu due to its flexibility and adaptability to the chosen micro-metal gear motors chosen in the WMR design. Thus, to set the distance to travel, the followings need to be observed [27]:

$$\tau_{count} = \frac{\mathrm{d}}{\pi D} \times c_{revo} \tag{10},$$

where τ_{count} is the target count to set, d is the distance to travel, D, is the wheel diameter and c_{revo} is the count per revolution. However, several previous works as discussed in [28], [29] have highlighted the limitations of this approach towards the occurrences of wheel slip or skid, resulting in significant error yield at its output. This is because during such event, the force applied at the tires would overcome the surface traction causing the wheel to spin unhindered by any traction forces [6]. This resulted in erroneous wheel revolution readings as these revolutions are not the actual linear WMR displacements. However, this study has chosen to incorporate optical encoder as odometry reading to establish the WMR ability in translating linear wheel displacement via the encoder count without considering wheel slip or skid. Further, researchers in [30] have elaborated that this could potentially be used as a wheel slip or skid detection measures in future experiment and analysis.

3.2. Steering Algorithm

A steering algorithm known as straight-line driving algorithm was designed and developed to test the drivability of the constructed WMR. The algorithm would fully utilize all of the variables yielded by the sensors involved in the WMR to drive the WMR via differential control of the left and right DC motor. This is done by controlling the differences in the encoder count output of the left and right wheel encoders that were installed. Pseudocode of the straight-line driving algorithm is elaborated below:

```
while (left count value && right count value less than targetCount);
    {
get LeftCountVal & RightCountVal;
if (left count value less than right count value);
{       increase power Left by power Offset;
                decrease power Right by power Offset;
}
if (left count value more than right count value)
{       decrease power Left by power Offset;
                increase power Right by power Offset;
}
    }
brake ();
return.
```

In order to drive in a straight line, either forward or backward, both of the encoder counts need to have the same values but in opposite polarity. Given any discrepancies between these variables would indicate that the WMR is discretely turning, either left or right, due to the fact that the side with the lower count is inherently rotating slower than the other side. This would then cause the WMR to turn into that particular side for as long as the duration continues.

To compensate for this, corrective differential steering algorithm developed would increase the power supplied to the DC Motor, in increments of 5% from the existing supplied power, for that specific DC motor. The increments would be assigned to the side that yield the lowest encoder counts as compared to the other.

Additionally, to ensure faster corrective steering being implemented, the side that has the higher counts would have its power decremented in steps of 5% from its existing supplied power. This would continue as long as discrepancies were detected between the two variables or until the set target counts achieved successfully. Upon reaching the target, the algorithm would execute the braking procedure which mean cutting off the power to both DC motors installed.

3.3. WMR operability

It is deemed important to ensure the overall cohesive operation would function accordingly. Thus, the WMR operability is tested in real time. As such, a comprehensive exercise has been designed, taking into account both basic and practical aspects of the WMR functionality. The designed exercise is as depicted in Figure 2. which showed the baseline objectives for the WMR.

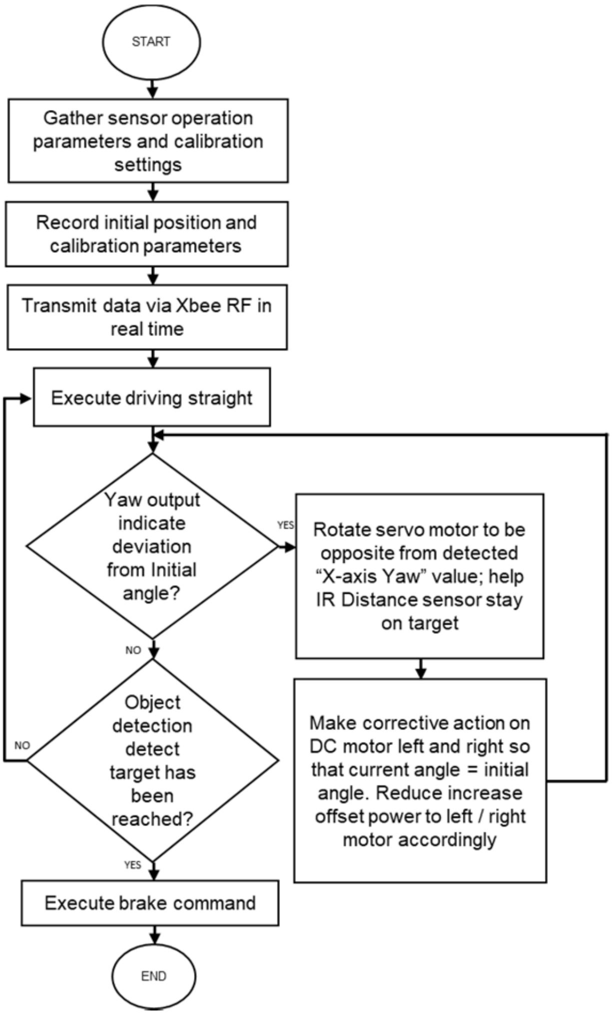
START
Gather sensor operation parameters and calibration settings
Record initial position and calibration parameters
Transmit data via Xbee RF in real time
Execute driving straight
Yaw output indicate deviation from Initial angle?
YES
Rotate servo motor to be opposite from detected "X-axis Yaw" value; help IR Distance sensor stay on target
Make corrective action on DC motor left and right so that current angle = initial angle. Reduce increase offset power to left / right motor accordingly
NO
Object detection detect target has been reached?
NO
YES
Execute brake command
END

Figure 2. WMR operation flow

Figure 2. indicates that an allowance for initialization, calibration and initial data transmission phase via XBee RF protocol with Tera Term software which encapsulate the data logging procedure, needs to be addressed. The data would then be stored in real time mode. Once these steps are achieved, the WMR would execute the straight-line driving algorithm until both wheel encoder counts has registered that set the target count.

During this process, if any discrepancies were detected between left and right wheel encoder counts, the corrective differential steering algorithm would be executed to ensure the WMR travelled in the straight-line mode successfully. It is also noted that for specific angle of Yaw deviations from initial angle, a calculated inverse response in distance sensor target tracking would be mitigated.

4. Result and Discussion

In this section, the performance of the constructed WMR is discussed to confirm its reliability in yielding the identified parameters as mentioned earlier. For this exercise, a coarse surface of an office carpet was used as the experiment track due to the travel length available that is accommodative to the developed steering algorithm for it to run unhindered. Furthermore, having a coarse surface for the WMR to travel provide the means for this study to minimize the possibility of wheel slip and skid from occurring. Next, Table 1. tabulated the output collected from the XCTU software generated in real time as the WMR travelled across the length as programmed.

Table 1. shows that the resultant sampling time between each row of data, is around 13 to 14 milliseconds. Several variables can be observed being logged in Table 1. that are the left and right motor encoder count yield which are LEnc Count and REnc Count respectively, the left and right motor power, which are LM Power and RM Power respectively, difference between current and last count of left and right motor encoder count which are diff_l and diff_r respectively, the Yaw output measured in degree angle which is Yaw, the 3-axis accelerometer output measured in milli-Gravity which are X-acc, Y-acc and Z-acc respectively, and the time duration each output is being logged which is Timer measured in milliseconds as well as other parameters included in the output command that are non-pertinent for this study discussions.

Table 1. indicated that all sensor outputs and data transmissions had successfully been captured during the experiment. Additionally, it can be seen that the WMR has stopped its movement at +/- 10,000 counts for both left and right motor encoder as shown by the last row of LEnc Count and REnc Count specifically at 10,026 and 9,864 counts respectively. This finding follows the calculations done in (10) which indicated that to reach a travel distance of 100 cm, the objective counts required is at 10, 0020.79 counts for each DC motor. This information was then embedded into the developed algorithm as shown in the first row of Table 1. along with the initial motor power. Slight variations over the count numbers are contributed by the mechanical losses that is unique to each motor.

Next, the Yaw output from IMU sensor was plotted against the duration in milliseconds and is used to validate the performance of the straight-line driving algorithm in practical application. As indicated in Figure 3., at 11.65 seconds onward, the built WMR managed to drive at relatively straight line despite several slight perturbances observed. This proves that the algorithm has successfully executed the corrective steering measure as deviations between both the left and right wheel encoder counts, were suppressed accordingly until the end of the experiment runtime.

It can also be observed that there exist significant deviations prior to 11.65 seconds. Figure 4., which plots the angle output of the IMU unit, is referred to help elucidate the root cause of this deviation. Note that given a proper calibrated MPU9250 unit, as discussed in [20], [25], an accurate angle output reading relative to earth magnetic field could be achieved. The range is from 0^0 to -180^0 in the clockwise rotations (CW) and 0^0 to 180^0 at the counter-clockwise rotation (CCW).

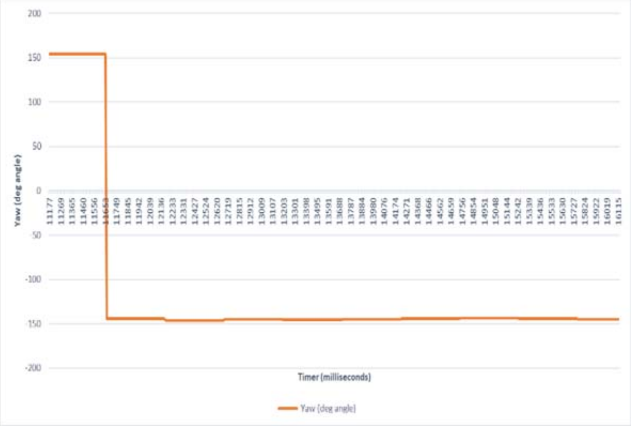

Figure 3. Yaw (degree angle) output of MPU9250 reading for straight line driving algorithm

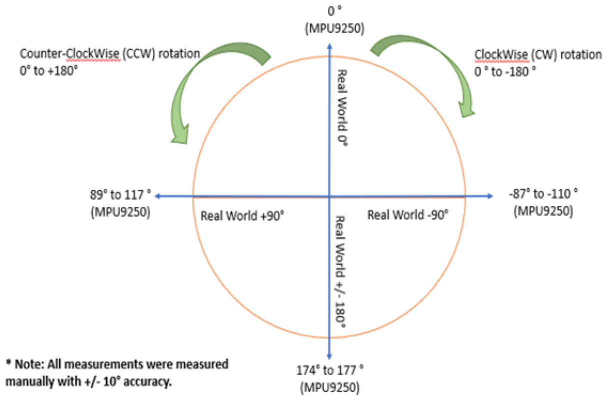

Figure 4. Actual angle output of MPU9250 in relative to earth magnetic field

Table 1. Sample of wirelessly logged data for straight line driving algorithm

Driving for 100.00 cm (10020.79 count) at 128 motor power										
LEnc Count	REnc Count	LM Power	RM Power	diff_l	diff_r	Yaw (deg angle)	X-acc (mg)	Y-acc (mg)	Z-acc (mg)	Timer:
3	0	128	128	0	0	154.4	-189.33	78.8	928.77	11177
13	8	128	128	4	4	154.4	-177.25	82.76	922.79	11191
27	24	128	128	18	12	154.4	-187.38	88.87	915.65	11203
51	45	113	143	17	19	154.4	-190	129.88	919.98	11215
72	72	128	128	26	25	154.4	-177.98	57.56	919.13	11229
93	98	113	143	18	26	154.4	-202.21	52.49	900.27	11242
114	127	128	128	23	28	154.4	-205.81	94.24	927.92	11255
↓	↓									↓
9897	9706	128	128	35	24	-144.7	-184.81	58.96	942.93	16060
9927	9730	113	143	34	24	-144.7	-182.13	59.14	934.69	16074
9957	9759	98	158	27	26	-144.7	-176.94	69.34	949.65	16088
9982	9792	83	173	29	32	-144.7	-178.65	69.64	940.73	16102
10005	9830	98	158	21	35	-144.7	-190.31	62.44	952.45	16115
10026	9864	113	143	24	34	-144.7	-170.53	83.56	943.3	16130

In this realization and based on cross examination of Table 1., the WMR starting position for the trial run could be found at +154.4^{0}angle. Referring to Figure 4., the starting position was therefore at the CCW region of the IMU rotation. As the WMR move forward, a slight heading deviation to the left side causes the WMR to cross the 180^0 angle threshold of the CCW region, resulting in the Yaw output to yield a negative value of -144.2^0 angle indicating that it is now operating in the CW region. This is the main cause of the observed deviation shown in Figure 3., prior to 11.65 seconds mentioned earlier.

5. Conclusion

As a conclusion, the built WMR has fulfilled the aim of this study which is to provide a real time quantitative value of Yaw and distance travel in an ideal practical application that has no slip or skid occurrences. These variables could then be further processed to yield the required of the x-axis, $\dot{x}_M$and y-axis, $\dot{y}_M$ coordinate as well as the angular velocity, ω parameters that further could be used to help train a path planning and object avoidance algorithm. Taken into account that all data logging activity have successfully been done wirelessly, this would enable unencumbered movement of the WMR, thus, is well-suited to perform its function as a data platform in future endeavour.

Acknowledgements

This study is funded by the Fundamental Research Grant Scheme for Research Acculturation of Early Career Researchers (FRGS-RACER): 600-IRMI/FRGS-RACER 5/3/ (001/2019) Research Management Centre, Universiti Teknologi MARA, Malaysia. The authors thanked Mr George Hercuse of Neuramatix Sdn Bhd for their support and assistance as the technical experts in this research.

References

[1]. M. Ben-Ari and F. Mondada. (2017). *Elements of Robotics NN ML in robotics*. Fundamentals.

[2]. Zheng, T., Yang, A., Wang, M., Huang, M., & Rao, B. (2015, July). Neural network-based adaptive tracking control of mobile robots in the presence of modelling error and disturbances. In *2015 34th Chinese Control Conference (CCC)* (pp. 3173-3178). IEEE.

[3]. Ciubucciu, G., Filipescu, A., Filipescu, A., Filipescu, S., & Dumitrascu, B. (2016, June). Control and obstacle avoidance of a WMR, based on sliding-mode, ultrasounds and laser. In *2016 12th IEEE International Conference on Control and Automation (ICCA)* (pp. 779-784). IEEE.

[4]. Dumlu, A., & Ayten, K. K. (2017, October). Sensorless model reference adaptive system control for mobile robot. In *2017 5th International Symposium on Electrical and Electronics Engineering (ISEEE)* (pp. 1-6). IEEE.

[5]. Brahmi, A., Saad, M., Gauthier, G., Zhu, W. H., & Ghommam, J. (2017). Tracking control of mobile manipulator robot based on adaptive backstepping approach. *International Journal of Digital Signals and Smart Systems*, *1*(3), 224-238.

[6]. Nguyen, T., & Le, L. (2018). Neural network-based adaptive tracking control for a nonholonomic wheeled mobile robot with unknown wheel slips, model uncertainties, and unknown bounded disturbances. *Turkish journal of Electrical engineering & Computer sciences*, *26*(1), 378-392.

[7]. Hoang, N. B., & Kang, H. J. (2016). Neural network-based adaptive tracking control of mobile robots in the presence of wheel slip and external disturbance force. *Neurocomputing*, *188*, 12-22.

[8]. Vidhyaprakash, D., & Elango, A. (2017). Studies on Affecting Factors of Wheel Slip and Odometry Error on Real-Time of Wheeled Mobile Robots: A Review. *International Journal of Mechanical and Mechatronics Engineering*, *10*(7), 1361-1368.

[9]. Yoo, S. J. (2013). Adaptive neural tracking and obstacle avoidance of uncertain mobile robots with unknown skidding and slipping. *Information Sciences*, *238*, 176-189.

[10]. Li, S., Ding, L., Gao, H., Chen, C., Liu, Z., & Deng, Z. (2018). Adaptive neural network tracking control-based reinforcement learning for wheeled mobile robots with skidding and slipping. *Neurocomputing*, *283*, 20-30.

[11]. Li, W., Ding, L., Liu, Z., Wang, W., Gao, H., & Tavakoli, M. (2016). Kinematic bilateral teledriving of wheeled mobile robots coupled with slippage. *IEEE Transactions on Industrial Electronics*, *64*(3), 2147-2157.

[12]. Naveed, K. (2013). *Slip and Skid Control of a Wheeled Mobile Robot*. COMSATS Institute of Information Technology Islamabad – Pakistan.

[13]. Alatise, M. B., & Hancke, G. P. (2017). Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter. *Sensors*, *17*(10), 2164.

[14]. Ali, M. A., & Mailah, M. (2019). Path planning and control of mobile robot in road environments using sensor fusion and active force control. *IEEE Transactions on Vehicular Technology*, *68*(3), 2176-2195.

[15]. Chen, W. J., Jhong, B. G., & Chen, M. Y. (2016). Design of path planning and obstacle avoidance for a wheeled mobile robot. *International Journal of Fuzzy Systems*, *18*(6), 1080-1091.

[16]. Gupta, V., Bendapudi, N., Kar, I. N., & Saha, S. K. (2018, March). Three-stage computed-torque controller for trajectory tracking in non-holonomic wheeled mobile robot. In *2018 IEEE 15th International Workshop on Advanced Motion Control (AMC)* (pp. 144-149). IEEE.

[17]. Boukens, M., Boukabou, A., & Chadli, M. (2017). Robust adaptive neural network-based trajectory tracking control approach for nonholonomic electrically driven mobile robots. *Robotics and Autonomous Systems*, *92*, 30-40.

[18]. Yun, D. U. (2016). *Kinematics and Dynamic Modeling and Simulation Analysis of Three-wheeled Mobile Robot*. 2016 MDM International Conference on Mechanics Design, Manufacturing and Automation.

[19]. Treffers, C., & van Wietmarschen, L. (2016). *Position and orientation determination of a probe with use of the IMU MPU9250 and a ATmega328 microcontroller*. TUDelft.

[20]. Tomasch, G., & Winer, K. (2019). Limits of Absolute Heading Accuracy Using Inexpensive MEMS Sensors, Limits Absol. Head. Accuracy Using Inexpensive MEMS Sensors, pp. 1-14.

[21]. Euston, M., Coote, P., Mahony, R., Kim, J., & Hamel, T. (2008, September). A complementary filter for attitude estimation of a fixed-wing UAV. In *2008 IEEE/RSJ international conference on intelligent robots and systems* (pp. 340-345). IEEE.

[22]. Lu, J., Lei, C., Li, B., & Wen, T. (2016). Improved calibration of IMU biases in analytic coarse alignment for AHRS. *Measurement Science and Technology*, *27*(7), 075105.

[23]. Roh, M. S., & Kang, B. S. (2018). Dynamic accuracy improvement of a MEMS AHRS for small UAVs. *International Journal of Precision Engineering and Manufacturing*, *19*(10), 1457-1466.

[24]. de Vries, M. (2019). *Multi-Rate Unscented Kalman Filtering for Pose Estimation: Using a car-like vehicle-platform*. Delft University of Technology. [master thesis].

[25]. Madgwick, S. (2010). An efficient orientation filter for inertial and inertial/magnetic sensor arrays. *Report x-io and University of Bristol (UK)*, *25*, 113-118.

[26]. Fahmizal, F., Priyatmoko, A., Apriaskar, E., & Mayub, A. (2019, October). Heading Control on Differential Drive Wheeled Mobile Robot with Odometry for Tracking Problem. In *2019 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)* (pp. 47-52). IEEE.

[27]. Megalingam, R. K., Varghese, J. M., & Anil, S. A. (2016, January). Distance estimation and direction finding using I2C protocol for an auto-navigation platform. In *2016 International Conference on VLSI Systems, Architectures, Technology and Applications (VLSI-SATA)* (pp. 1-4). IEEE.

[28]. Calero, D., Fernandez, E., & Parés, M. E. (2017). Autonomous Wheeled Robot Platform Testbed for Navigation and Mapping Using Low-Cost Sensors. *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *42*, 67-74.

[29]. Onyekpe, U., Palade, V., Herath, A., Kanarachos, S., Fitzpatrick, M. E., & Christopoulos, S. R. G. (2021). WhONet: Wheel Odometry Neural Network for Vehicular Localisation in GNSS-Deprived Environments. *arXiv preprint arXiv:2104.02581*.

[30]. Reina, G., Ojeda, L., Milella, A., & Borenstein, J. (2006). Wheel slippage and sinkage detection for planetary rovers. *IEEE/Asme Transactions on Mechatronics*, *11*(2), 185-195.