International Journal of Autonomous Robotics and Intelligent Systems
Vol. 1, No. 2, December 2025 [189-203]

Design and Implementation of an Affordable GPS Based Rover for Path Planning with Obstacle Avoidance

Azuwir Mohd Nor[1*], Mohamad Ezral Baharudin[1], Mohd Zakimi Zakaria[1], Mohd Sazli Saad[1] and Ahmad Aslam Abdul Hafidz[1]

[1]Faculty of Mechanical & Technology Engineering, Universiti Malaysia Perlis (UniMAP), Pauh Putra Campus, 02600 Arau, Perlis, Malaysia

Received 30 October 2025, Revised 15 December 2025, Accepted 26 December 2025

ABSTRACT

In this paper, the design and development of an affordable GPS-guided autonomous rover that is real-time optimized for path and obstacle avoidance are presented. The system was built based on an ESP32 microcontroller, BN-880 GPS receiver and HC-SR04 ultrasonic ranging sensor with PID control law for directive adjustments. For safe travel, the rover used conditional statement and Vector Field Histogram (VFH) algorithm to determine real time path along a 180-degree field-of-scan falloff. In-field investigation at Dewan Ilmu Parking Lot, Universiti Malaysia Perlis tested the crucial behaviours of this system like GPS positional accuracies, reliability to avoid obstacle and stability of trajectory in PID control and non-PID configurations. The rover maintained a GPS precision radius to 2.5 meters and achieved a 100 percent success rate in obstacle avoidance. Quantitative comparisons revealed that PID-controlled navigation offered superior directional stability and turned more smoothly, whereas non-PID navigation yielded faster completion times but exhibited larger deviation errors. These results demonstrate the feasibility of developing an inexpensive and modular autonomous mobility platform for applications such as agricultural robotics, industrial surveillance and even field exploration.

Keywords: Autonomous Rover, GPS Navigation, Path Optimization, Obstacle Avoidance, PID Control

1. INTRODUCTION

Studies on autonomous rovers have shown that efficient outdoor navigation cannot rely solely on global positioning via the Global Positioning System (GPS) or other Global Navigation Satellite Systems (GNSS). Instead, robust sensor fusion and dynamic local path planning are essential to compensate for sensor limitations. For instance, single-frequency GPS modules typically exhibit horizontal errors ranging from 1 to 5 meters in open-sky conditions, with even greater inaccuracies in obstructed or multipath-rich environments [1][2]. In one experimental study involving a kinematic test of a single-frequency module, the median horizontal deviation was found to be between 1.2 and 1.3 meters, while 95% of the errors reached up to 2 meters without Real-Time Kinematic (RTK) correction [2]. Additionally, the study highlighted that motion-induced drift and multipath effects degraded positioning accuracy compared to static tests. These findings suggest that relying solely on low-cost GPS for predefined path-following, such as navigating plantation rows or corridors; may hinder precise, on-target applications.

Due to these challenges, sensor fusion has become one of the most important technologies for improving localization and control in autonomous rover. A generic multi-sensor fusion framework was proposed for intelligent vehicles and mobile robots [3]. The study showed that combining GPS / GNSS, Inertial Measurement Unit (IMU), wheel odometry, and other sensors can

*azuwir@unimap.edu.my

provide much better trajectory estimation accuracy as well as improved robustness in dynamic environments. A similar paper of sensor merger for localization of a mobile robot has appeared recently in which the fusion of GPS with Inertial Navigation System (INS), odometry and other proprioceptive or exteroceptive sensors to enhance navigation on outdoor situations were frequently adopted [4]. In some applications with field and agricultural robotics, which are known to be low-cost GNSS receiver users, real-time relative position methods (e.g. single- or double-differencing) revealed that more than 75% of the position estimations achieved an accuracy of less than 1 m during the open-sky experiments [5]. These results seem to indicate that for a cost-effective kind of implementation it would be interesting to use GPS in combination with IMU / magnetometer (for heading), as well as proximity / obstacle sensors (ultrasonic, infrared (IR) or LiDAR).

Path planning and obstacle avoidance is the counter part of localization. A survey of path-planning methods for mobile robots such as graph-based (e.g., A*, Dijkstra), sampling based (e.g., rapidly-exploring random tree (RRT), probabilistic roadmap method (PRM)) and gradient-based (e.g., potential fields) were reviewed in terms of application to dynamic ambient environments [6]. Another one focused on navigation and obstacle avoidance, showing that hybrid methods combining meta-heuristics (such as particle swarm optimization (PSO), genetic algorithm (GA)) with classical planners are becoming popular to face real time constraints and complex terrain [7]. Optimization of path planning based on crop growth status, field layout and avoidance obstacle for traveling in agricultural and outdoor environments has been brought by the researchers as well [8], showing that work optimization reduces nonproductive travel, improving coverage efficiency.

In case of GPS regulated rovers, one work presented an agricultural robot using GPS waypoints and ultrasonic sensors for obstacle detection with local path following to negotiate around rows [9]. The authors found it challenging to keep the paths straight in confined spaces, particularly when GPS error induced lateral drift. Rather than following a precise path, they routed around any detected obstacles on the fly. More recently, a similar trajectory tracking error of about 0.08 meters during outdoor tests on an unmanned ground vehicle was obtained using simplified second order Sliding Mode Control (SMC) together with GNSS and redundant inertial sensors and LiDAR-based obstacle detection [10]. This implies that cm-level error can be obtained for high-precision mobile platforms with expensive sensors and sophisticated control laws, which is cost-prohibitive and not necessarily compatible with low-cost educational prototypes. Integrating these strands, a structured architecture GPS-controlled rover can be proposed: Global localization through GPS / GNSS [11]; Heading refining from IMU / magnetometer and odometry [12]; Obstacles detection from ultrasonic / IR / LiDAR sensors [13]; Local path planning / optimization able to react on deviations and obstacle occurrences [14].

For lower end of cost products, where GPS error can be on the order of meters and obstacle sensing less robustly designed in, the path-planning module must recover by re-doing trajectories or executing corrective loops. For example, a rover can be kept within threshold margin of the reference path even through narrow corridors using compensated for by GPS errors integration (coming from sensor fusion) and dynamic obstacle avoidance. However, several research gaps remain. First, many studies consider high-grade sensors or augmented by RTK or others without exploring budget-level modules (e.g. single-frequency GPS module, low-cost IMU) inside small-scale agricultural rows, where they could all be hard to use due to tight-space operations. Second, in low-cost rover contexts just like ours, dynamic obstacle avoidance is somewhat ignored particularly outdoor while most path-planning research focuses on static or slowly changing obstacles. Third, the combination of real time waypoint navigation (GPS) and in immediate local obstacle avoidance is still not well represented, most especially for open-field activities where ground conditions and visibility to satellite signals change. Last, it is interesting if paths can be further optimized efficiently for coverage or obstacle avoidance on field conditions in specific to small-scale rover platforms.

This study proposed a system, which intends to fill in these gaps by using a low-cost microcontroller like ESP32, an inexpensive GPS module, magnetometer / IMU (accelerometer and gyroscope), HC-SR04 ultrasonic sensors and L298N motor driver board based on a GPS controlled rover capable of path optimisation and obstacle avoidance. By integration of sensor fusion (GPS + magnetometer + ultrasonic) and constrained outdoor corridors an optimal path was found which will be relevant for accessible research and educational autonomous mobile platforms.

2. MATERIAL AND METHODS

A rover was developed using an engineering approach that integrated mechanical design, embedded electronics, algorithm implementation, and real-world testing. As conceptually illustrated in Figure 1, the design process began with preliminary sketches and progressed through several iterations of prototyping, optimization, and testing: all conducted at Universiti Malaysia Perlis (UniMAP). Rather than following a rigid, linear process, the project embraced an iterative mindset: insights gained from early tests were used to refine both hardware and software components. The core objective was practical, to create a low-cost rover capable of navigating to GPS-defined coordinates, avoiding unplanned obstacles, and course-correcting mid-journey through PID-based heading control.

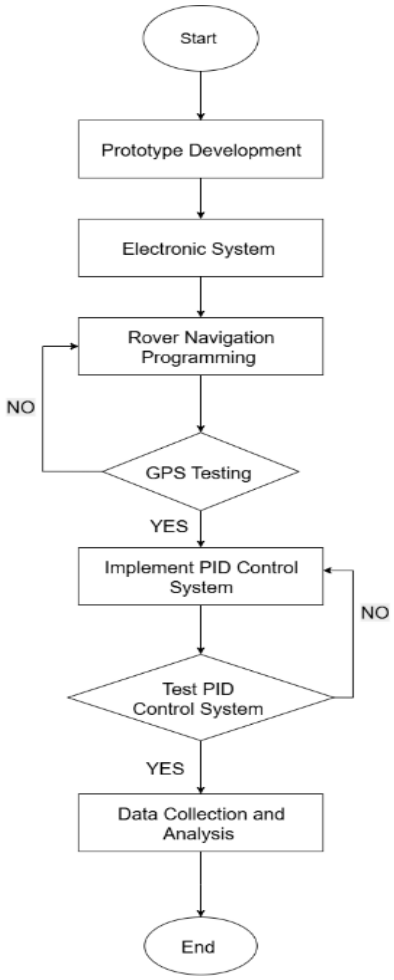


Figure 1. Flowchart of project development

This section has presented a structured and integrated methodology for the development of the proposed GPS-based autonomous rover. The overall system architecture was clearly decomposed into mechanical and electronic subsystems, followed by the formulation of the rover kinematic

model and the design of the control strategy. A closed-loop PID controller was employed to regulate heading stability during waypoint navigation, while the Vector Field Histogram (VFH) algorithm enabled real-time obstacle avoidance based on ultrasonic sensing. These components were organized into a coherent workflow, as illustrated in the proposed system flowchart, providing a step-by-step description from sensor data acquisition to motion execution. The experimental setup was then defined to ensure repeatable and objective performance evaluation under outdoor conditions. This systematic methodological framework establishes a clear foundation for the performance analysis and discussion presented in the subsequent section.

2.1 Rover Hardware Design and Architecture

The rover chassis serves as the structural foundation of the system, providing a base upon which all other modules are mounted. In this study, a four-wheel drive (4WD) configuration was selected for its mechanical robustness, ease of modification, and suitability for the intended tasks. A pre-assembled chassis significantly reduces the need for complex mechanical fabrication, allowing the development team to focus on navigation and obstacle avoidance using feedback systems. The chassis also houses key components such as the GPS module, ultrasonic sensors, battery pack, and motor driver. Mounting points were carefully repositioned to ensure that each component was installed in a way that preserves system balance and performance. For example, the BN880 GPS module and its integrated compass were mounted on an elevated bracket to optimize signal reception and reduce electromagnetic interference from nearby electronics. The final rover design is shown in Figure 2.

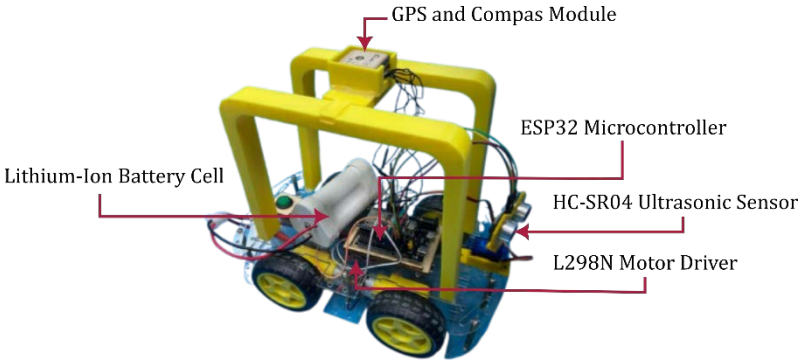


Figure 2. The GPS control rover

The electrical system of the GPS guided rover consists of five intergraded modules, all working together to provide strong data processing, accurate movement and fast motor response. The ESP32 microcontroller is the heart of this setup and acts as the central processor. The ESP32, which operates with a dual core Tensilica LX6 processor, performs both data acquisition from sensors and duties related to PWM signal generation for motor actuation simultaneously. This microcontroller communicates with the BN-880 Global Navigation Satellite System (GNSS) module through UART, which runs a GNSS receiver and digital compass to provide real-time latitude, longitude and heading information. In outdoor use, BN-880 provides horizontal precision within a range of 2 to 4 meters, while under the open sky and suitable for drone, aircraft, watercraft or robot navigation.

An HCSR04 ultrasonic sensor is utilized by the system for environmental awareness and to make sure that it operates safely. The sensor emits 40 kHz broadband bubble-burst pulses to detect the obstacles in front of it and up to 2 m far, which provides possibility for the controller of collision avoidance during steering. When switching on the case fans a motor driver board L298N dual H-bridge DC Motor drive is responsible for actuating the motors, transforming PWM signals from ESP32 in bidirectional current for two DC motors. The driver is capable of an input voltage up to

5-35 V and allows for a maximum peak current per channel of 2 A providing smooth and independent motion control.

The whole system is driven by a 11.1 V Lithium-ion battery, which can support about 1.5 to 2 hours' normal usage situations. All these elements together form compact, energy-efficient and fully-on-chip control. The full electrical and electronic setup is explained in Figure 3, illustrating its circuit schematic coupled with the between-module connections.

Summary of core system modules:

i. ESP32 MAIN CHIP: Dual-core Tensilica LX6 processor, sensor data processing and PWM signal / motor control.
ii. BN-880 GNSS / Compass Module: Offers lat. / long. and heading data sent to flight controller via UART communication (with an estimated horizontal accuracy of 2–4 meters).
iii. HC-SR04 Ultrasonic module: A sensor used for ultra-sonic sensing which is used for obstacle avoidance applications.
iv. L298N Motor Driver: Turns the pulse width modulation (PWM) of ESP32 into bidirectional drive signals that are suitable for use with DC motors; they support a supply voltage between 5 and 35 V and can deliver up to 2 A per channel peak current.
v. Power System: A 11.1 V lithium-ion battery pack provided energy for approximately 1.5-2 hours of use under normal workload conditions.

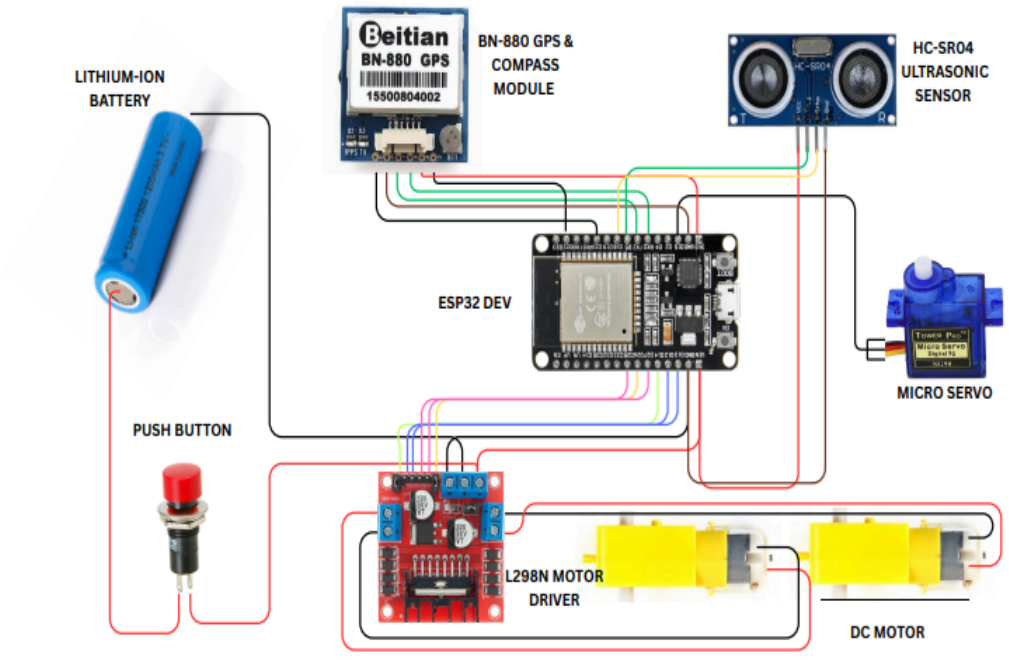


Figure 3. Schematic diagram of the electrical and electronic system

2.2 Kinematic Model

A kinematic model is a mathematical representation of the rover, including its functional dimensions and degrees of freedom. In rovers, the constituent elements that make up a kinematic model are wheels, actuators, and joints determining motion and steerage capabilities. Degrees of freedom define the mobility of the rover, while coordinate frames describe position and orientation Kinematic chains link the elements, facilitating efficient navigation and motion

planning in agricultural environments. The rovers are updated based on their velocity and orientation. The vehicle's state is defined as if in linear velocity, as shown in Equation (1):

$$\begin{aligned} \dot{x} &= rv\ cos\ (\theta) \\ \dot{y} &= rv\ sin\ (\theta) \\ \dot{\theta} &= \omega \end{aligned} \tag{1}$$

where $\dot{x}$ and $\dot{y}$ are the velocity along the x- and y- axis. The angular velocity is presented by $\dot{\theta}$ which affect the rovers heading, while r is radius of wheel and v linear velocity. For the differential drive rovers, the linear and angular velocity are functions of the wheel velocity, as in Equation (2):

$$\begin{aligned} v &= \frac{r}{2}(\omega r + \omega l) \\ \omega &= \frac{r}{l}(\omega r - \omega l) \end{aligned} \tag{2}$$

Substitute v and ω (angular velocity) into the kinematic equation so that it will be in Equation (3):

$$\begin{aligned} \dot{\mathrm{x}} &= \frac{r}{2}(\omega r + \omega l)\cos\ (\theta) \\ \dot{y} &= \frac{r}{2}(\omega r + \omega l)\sin\ (\theta) \\ \dot{\theta} &= \frac{r}{l}(\omega r + \omega l) \end{aligned} \tag{3}$$

The two-dimensional kinematic model finds wide applications in robotics and vehicle navigation. It is highly important for path planning because it allows rovers to calculate accurate trajectories to reach a target location efficiently. In the field of control systems, this model assists in devising algorithms that will control wheel velocities to perform the motion accurately. From Figure 4, the two-dimensional kinematic modelling has been shown. This model further contributes to the conduction of mock-up studies by which engineers test and improve their navigation strategies in virtual environments before real-world implementation. The simplicity and effectiveness of the control make it crucial for rover systems in agriculture, exploration, and industrial automation.

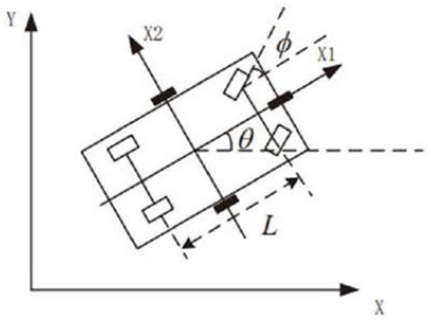


Figure 4. Two-dimensional kinematic modelling

2.3 Control System

In practical applications, the smooth and accurate stable movement of a differential-drive rover is realized through the coordination of proportional (P), integral (I), and derivative (D) terms using PID controller. The proportional term deduces an immediate corrective action in proportion to the present tracking error; the integral term removes the steady-state error by

integrating past errors over time and derivative term improves system stability by anticipating the future trends of errors and reducing overshoot.

The adequate seasoning of these parameters is crucial for good controller performance and must be bear in mind the environmental variations and operation. Despite established tuning procedures like Ziegler–Nichols providing a guide for an initial setting, it was found that a significant amount of trial and error is needed to find the best values. In practice, the presence of real-world conditions such as uneven terrain, wheel slip and payload variations - require iterative context specific tuning that may require some combination of algorithm tuning as well as practical on-hand trial-and-errors validation for robust control performance.

2.3.1 PID Controller

The PID controller corrected heading errors derived from the compass by adjusting left / right wheel speeds. The control law is expressed as in Equation (4):

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \tag{4}$$

where $e(t)$is the instantaneous heading error.

After fine tuning, the optimal parameters were determined experimentally at Kp = 3 and Kd = 40 (Ki ≈ 0). These gains ensured smooth convergence without oscillations under 12 V supply and flat-terrain conditions.

2.3.2 Heading Error Computation

The error represents the degree of angular difference between the reference and the current heading of rover in control systems, and robotics more generally, it is used to determine how much a system deviates from some target, which is then that used as input for corrective action. From -1 to 1 so without normalizing this error can be outside of the range [-180, 180], hence the rover may take an unnecessary longer turn to get aligned with it. The general rule of thumb for raw error is as in Equation (5):

$$\text{Raw Error} = \text{Set point heading} - \text{Current heading} \tag{5}$$

To ensure the rover takes the shortest path, normalization adjusts the error to lie within the range [-180°, 180°]. If the error is greater than 180°, it means the shorter path is in the opposite direction which is counterclockwise, so to calculate the formula, use Equation (6):

$$\text{Normalized Error} = ((\text{raw error} + 180° \mod 360°) - 180° \tag{6}$$

To calculate the error, ensure the rover turns in the direction of the smallest angle difference. Calculating the modulus difference and then modifying it as needed can help one to pursue the shortest path. Based on the shortest angular distance to the objective, the rover has a clear directive to turn either clockwise, which is a positive mistake, or anticlockwise, which is a negative error, after this procedure is finished.

2.3.3 Control Problem Formulation

Figure 5 illustrates how the rover uses a closed-loop feedback control system to precisely follow the intended trajectory while preserving exact control of its position and orientation. By using real-time feedback of its present state (x, y, θ), this method promises that the rover constantly modulates its mobility, therefore reducing mistakes in both position and orientation. The

procedure starts in the trajectory planning module, which offers the rover's intended location (x_D, y_D). and orientation (θ_D). These principles outline the target rover must reach. Practically, these values might be dynamically produced in response to real-time navigation needs including obstacle avoidance or as part of a pre-defined course. The system references the required location and orientation as inputs. The rover aims to minimise the difference throughout time between its present state (x, y, θ) and the reference inputs.

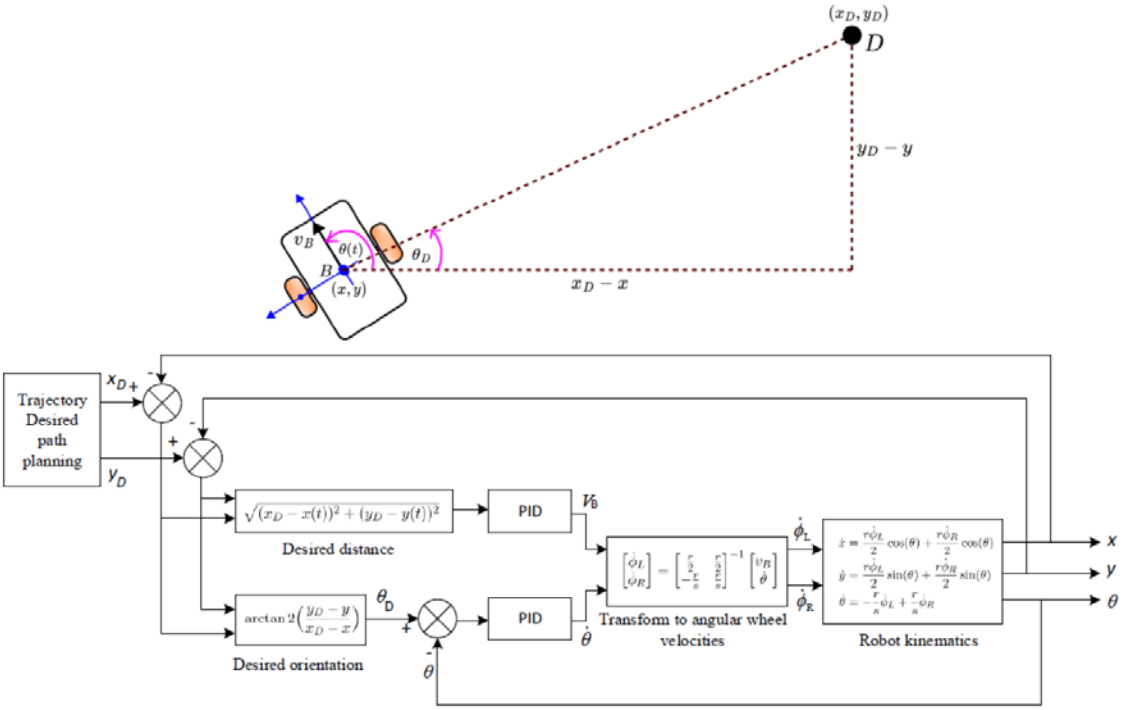


Figure 5. Kinematic model and PID controller structure for autonomous rover navigation

The Euclidean distance between the rover's present (x, y) location and the intended (x_D, y_D). position is the position error, often known as distance error. It can be estimated as in Equation (7):

$$e_d = \sqrt{(x_D - x)^2 + (y_D - y)^2} \tag{7}$$

It measures the distance of the rover to its goal position. A larger position error indicates that the rover needs to move more aggressively to reduce gap, while smaller error signals that the rover is nearing the target.

The orientation error measures the angular difference between the rover's current heading (θ) and the desired heading θ_D. The desired orientation θ_D is too determined using the arctan2 function. This error helps the rover align itself correctly with the desired direction of travel. The orientation is then calculated as in Equations (8) and (9):

$$\theta_D = arctan2((y_D - y), (x_D - x) \tag{8}$$

$$e_\theta = (\theta_D - \theta) \tag{9}$$

2.4.4 Vector Field Histogram (VFH)

One of the methods has been used to avoid obstacle is Vector Field Histogram (VFH) algorithm. The integration of VFH is crucial to ensure the rover can travel through partially known or dynamic obstacle. The core function of VFH in this project is interpret sensor form HC-SR04 ultrasonic sensor which mounted at front of the rover. This sensor is responsible to continuously measuring the distance at front rover. For this project, the threshold is 70 cm where if the sensor detects obstacle or object below than 70 cm range the rover will stop and decide by the conditional logic. It begins by create a local two-dimensional grid based on the real time sensor readings. The grid will capture the presence and position of obstacle around the rover in short range.

Figure 6 shows how the rover avoids the obstacle using the VFH algorithm for obstacle avoidance. The yellow circle with radiating lines represents the rovers' different positions and the field of view of the rover were divided into angular sector. The grey pentagonal is represent as the obstacle in the rover's environment. The black arrow shows the heading of the rovers. the figure depicts the rover's motion through a sequence of position, adapting path based on sensor data to avoid collisions while progressing toward the goal.

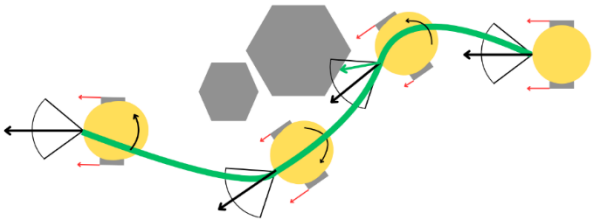


Figure 6. Vector Field Histogram (VFH) obstacle avoidance path planning

2.4.5 Algorithm Development

The control logic combined the GPS navigation module, VFH obstacle-avoidance routine and PID heading regulator.

The pseudocode are as follows:

i. Acquire GPS (latitude, longitude) and compass (heading).
ii. Compute target bearing and distance to waypoint.
iii. If distance < 2.5 meters → stop (success).
iv. Else check ultrasonic range:
    a. If obstacle < 30 cm → execute VFH avoidance.
    b. Else → apply PID correction to heading.
v. Repeat until goal reached.

This logic was implemented in C++ using the Arduino IDE for the ESP32 platform.

2.4.6 Experimental Setup

Field experiments were conducted at Dewan Ilmu Parking Lot, UniMAP, on a flat concrete surface with open-sky visibility to minimize GPS multipath errors. Testing sessions occurred between 09:00 and 11:00 a.m. to ensure stable satellite geometry. Table 1 shows the overview of experimental tests.

Table 1 Overview of experimental tests

Test	Environment	Main Metric (s)	Notes
GPS accuracy	Open sky / near building	Positional error (m)	10 samples per site
Obstacle avoidance	Controlled straight line	Success rate (%)	Detection < 30 cm
Path optimization	Open field	Time (s), distance (m)	PID vs Non-PID

Three distinct tests were performed:

i. GPS accuracy test — ten repeated latitude / longitude readings at fixed positions under open-sky and near-building conditions.
ii. Obstacle avoidance test — obstacles (cardboard boxes, 30 × 30 cm) placed at 1 m intervals; success defined as collision-free traversal.
iii. Path optimization test — comparative runs using PID-controlled and non-PID navigation; metrics included completion time, deviation angle, and final distance from waypoint.

Position and heading data were logged via Bluetooth serial output to a laptop at 1 Hz sampling rate. Trajectory plots were generated by overlaying GPS coordinates on satellite imagery using Google Earth Pro. Statistical analysis (mean, standard deviation) was applied to compare PID and non-PID performance metrics.

3. RESULTS AND DISCUSSION

A set of realistic field experiments was conducted to evaluate the performance of the GPS-guided rover under various operational conditions. The evaluation focused on three key aspects: the accuracy of GPS-based position control, the reliability of obstacle avoidance, and the effectiveness of path optimization using both PID and non-PID control strategies. All experiments were monitored visually as the rover navigated the test field, with its location recorded every second via Bluetooth serial communication to a laptop. In each test scenario, the rover was assigned to travel toward a sequence of predefined GPS waypoints, typically spaced 5 to 8 meters apart.

3.1 GPS Accuracy Evaluation

The BN-880 GPS + Compass module was objectively tested to identify its positional stability and repeatability in controlled conditions. Studies were conducted in two environments: open-sky and semi-obstructed environment, close to a building wall being on the 3 meters distance. There were 10 consecutive readings taken at each identified test location. The differences between these measurements and their computed mean location were used to calculate positional error. Table 2 shows the results of the accuracy test.

Table 2 Result of the GPS accuracy test

Environment	Average Error (m)	Std. Deviation (m)	Observation
Open sky	2.47	0.68	Stable signal, minimal drift
Near building	3.82	1.05	Multipath reflections, signal delay

Experiments showed that open-sky operation provides a mean error radius of 2.47 meters, enough for practical short-range autonomous outdoor navigation.

3.2 Obstacle Avoidance Performance

The obstacle avoidance system was evaluated by placing 30 cm × 30 cm cardboard boxes at randomly but evenly spaced positions along the rover's straight-line path. A detection threshold of 30 cm was set for the experiments. The Vector Field Histogram (VFH) algorithm was employed

to dynamically plan avoidance paths in real time. The rover avoided all obstacles in ten consecutive trials, achieving a 100% success rate without any collisions. Additionally, the ultrasonic sensor demonstrated stable detection performance over a distance of up to 2 meters, with consistent response times averaging approximately 60 ms. A summary of the obstacle avoidance test results is presented in Table 3, while Figure 7 illustrates the sequential trajectory of the rover as it navigated around the introduced obstacles. These results prove that ultrasonic sensing can be successfully incorporated into the VFH algorithm for active obstacle avoidance in real-time.

Table 3 Summary of obstacle avoidance test

Parameter	Measured Value
Detection range	2.0 m (max)
Avoidance activation	30 cm threshold
Success rate	100 %
Missed detections	0
Average reaction delay	0.06 s

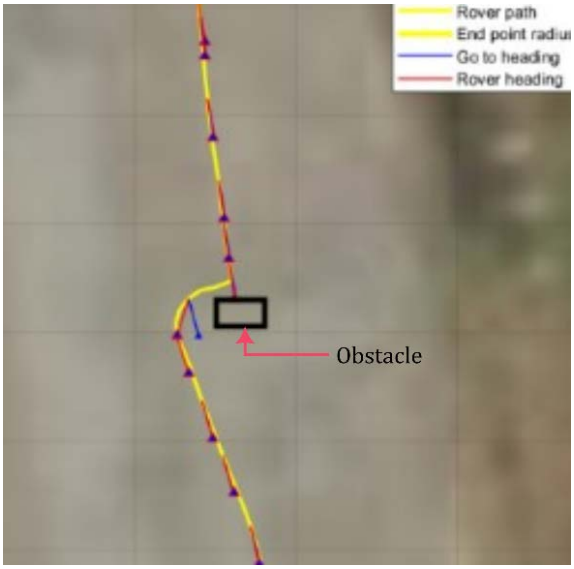
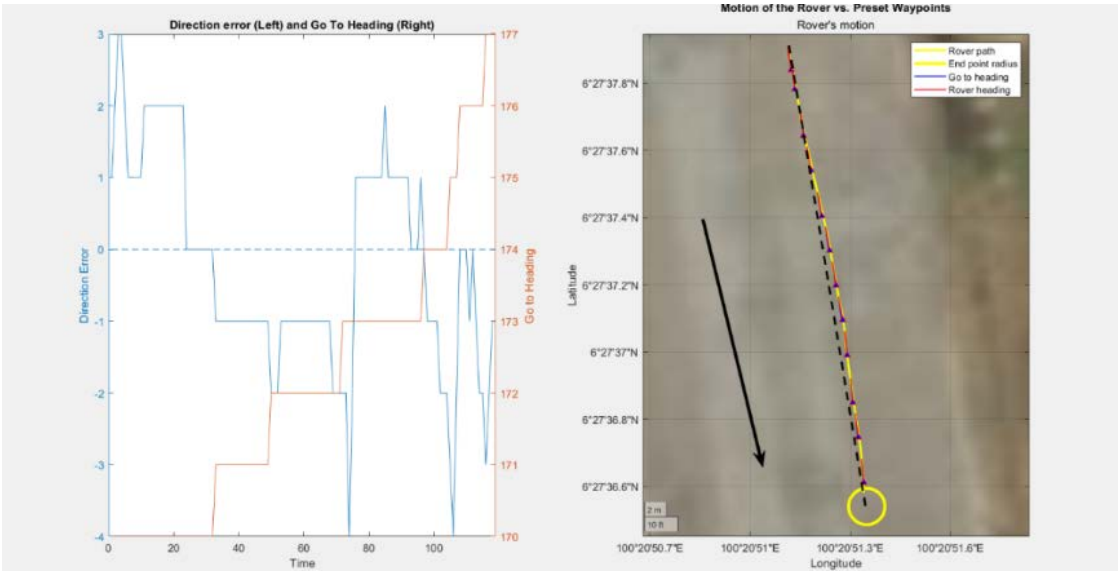


Figure 7 A sample of avoidance paths generated when obstacles were introduced

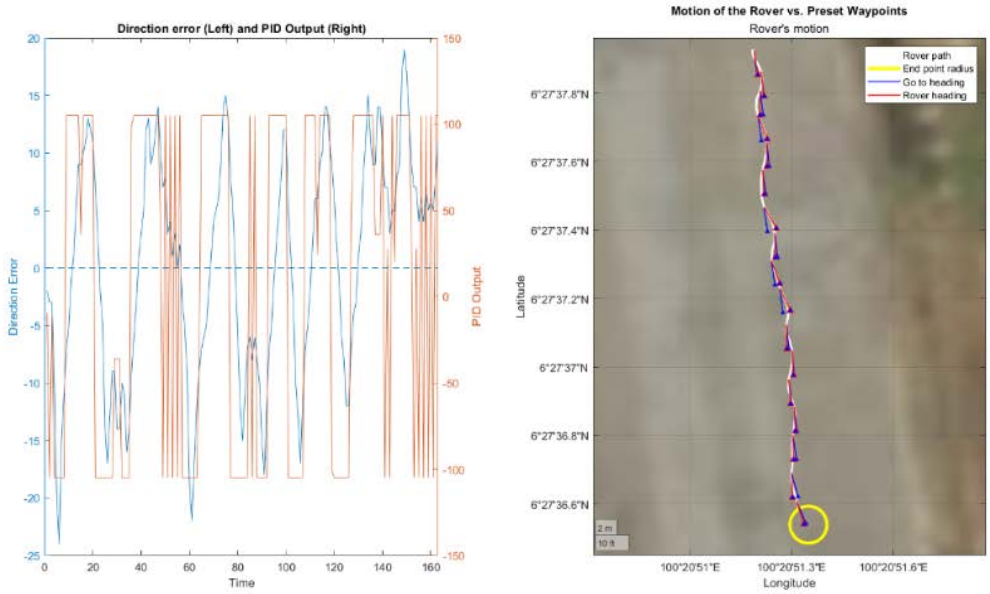
3.3 Path Optimization Using PID and Non-PID Control

This experiment compared the rover's path-tracking performance under two control configurations: (1) PID control (Kp = 3, Kd = 40), and (2) non-PID control (open loop, no feedback correction). Metrics included completion time, trajectory deviation, and final distance from the target waypoint. Figure 8 illustrates the performance of the non-PID system through two parts: a graph of direction error and Go to Heading over time, and a trajectory plot on a satellite image, while Figure 9 shows the performance when PID controller is activated.



(a) (b)

Figure 8. (a) Rover's direction error (blue line) and go to heading (orange line) over the times, and (b) the trajectory plot on satellite images for non-PID system



(a) (b)

Figure 9. (a) Rover's direction error (blue line) and PID output (orange line), and (b) trajectory plot on satellite images for PID value K_P=5 and K_D=40

While a non-PID navigation resulted in marginally faster converging times, it suffered from increased overshoot and directional drift, particularly around sharp turns. Rover with PID controller has smoother velocity transitions and better waypoint convergence. This is consistent with previous control research [15][16] which has demonstrated that P and D terms are capable of effectively reducing heading errors in differential-drive rovers. The trade-off between speed and accuracy of performance accords with what theory lowing PID control values

stability/precision over time efficiency. Table 4 lists the path optimization test result with PID and without PID controller.

Table 4 Summary of path optimization test

Parameter	PID Control	Non-PID Control
Mean completion time (s)	60.4	52.7
Mean distance error (m)	1.2	2.4
Mean heading deviation (°)	3.8	7.2
Stability rating	High	Moderate
Overshoot	Negligible	Noticeable

3.4 Overall System Integration and Observations

In both PID control and non-PID control scenarios, the rover has a 100% obstacle avoidance rate and testified to the efficiency of the VFH based avoidance system. The PID-based rover showed better path control on the navigation tests, since the controller constantly adjusted motor output to minimize heading error and require that direction components followed desired waypoint directions. By contrast, the non-PID rover did not have any correction applied to it; consequently, adjustments to its desired path were more common.

In terms of completion time, the non-PID configuration achieved a faster average of 58.78 seconds, compared to 75.14 seconds for the PID-controlled system. The higher speed in the non-PID case was due to looser path correction, which resulted in reduced control precision and, consequently, lower navigation accuracy. Both configurations successfully stopped within the GPS error margin of 2.5 meters from the waypoint. The non-PID system had an average final distance of 0.72 meters from the target, while the PID-controlled system averaged 1.04 meters. Despite the slight difference, both results were comparable in accuracy. Although the PID approach was slower, it provided more reliable navigation by actively correcting heading deviations, an advantage that becomes more important in applications requiring high accuracy or precision.

Once the GPS, ultrasonic sensors, compass, and PID control were fully integrated, the system demonstrated stable and reliable performance. Communication between modules via UART and digital I/O remained robust, with no data loss observed during a continuous 60-minute experimental run. Under moderate workload conditions, the average battery life was approximately 95 minutes, indicating the system's potential for extended field applications such as crop surveying or campus monitoring.

However, several limitations were identified:

i. Magnetic interference: Close to the metal will interfere in bearing reading.
ii. Occasional lag (~1 s) led to minor overshooting of waypoints due to GPS delay.
iii. Loud ultrasonic returns: Also, we had to filter out false echoes caused by reflective surfaces in the software.

Despite these difficulties, the system showed excellent reliability and utility as a research and educational robotics tool.

4. CONCLUSION

This work presented the development of a functional GPS-servo controlled automation rover for reliable mobility, optimal path alignment and strong obstacle avoidance at low-cost. The integration of an ESP32 microcontroller and a BN-880 GPS + compass module with HC-SR04

ultrasonic sensor provided the real-time control and waypoint tracking with positional accuracy radius = 2.5 meters. Using PID controller with gains and, the heading error was very effectively reduced that made the motion more stabilized and performed better smooth than without using PID (open loop). Furthermore, by using the VFH algorithm for path planning provided efficient collision avoidance; tests on field at Dewan Ilmu, UniMAP showed that 100% success rate without collisions in terrain environment were successfully accomplished by the rover.

In terms of performance, PID-based navigation provided better tracking and reduced overshooting but with a little increase in travelling time. Some practical importance for feedback control approaches is underlined by this result, in the case of robotic and autonomous systems, where accuracy and reliability can be considered more important than velocity.

In conclusion, the presented prototype system shows that a low-cost, modular and easily replicable rover can be designed to have reliable autonomy capabilities that are useful in research and light-duty field work.

For future work consideration:

i. Multi-waypoint navigation enables the rover to travel to multiple coordinates autonomously.
ii. Dynamic PID control based on adaptive and fuzzy logic for tuning of parameters across different terrains.
iii. RTK-GNSS for sub-meter positioning accuracy.
iv. Fully autonomous SLAM (Simultaneous Localization and Mapping) exploration with no waypoints required.
v. Such improvements will expand the use of this platform in agriculture, intelligent transportation system and industrial automation applications.

ACKNOWLEDGEMENTS

The authors express sincere gratitude to the Faculty of Mechanical & Technology Engineering, Universiti Malaysia Perlis (UniMAP), for providing laboratory facilities and technical support during the research. Special thanks are extended to the DyCos Research Group for assistance in testing and data collection.

REFERENCES

[1] S. Rudolph et al., 2019. Assessment of the position accuracy of a single-frequency GPS, Precision Agriculture, vol. 20, no. 1, pp. 19–39. doi:10.1007/s11119-018-9578-1.
[2] G. S. Carvalho et al., 2023. Performance Analysis of Relative GPS Positioning for Low-Cost Agricultural Robot, Sensors, vol. 23, no. 21. doi:10.3390/s23218835.
[3] M. Osman et al., 2021. A generic multi-sensor fusion scheme for localization of intelligent vehicles and mobile robots, Measurement and Control, vol. 54, no. 10, pp. 2714–2730. doi:10.1177/01423312211011454.
[4] A. Waga, S. Benhlima, A. Bekri, J. Abdouni, 2025. A survey on autonomous navigation for mobile robots: From traditional techniques to deep learning and large language models, Journal of King Saud University – Computer and Information Sciences, vol. 37, art. 198. doi:10.1007/s44443-025-00216-x.
[5] D. Radočaj et al., 2022. A Low-Cost Global Navigation Satellite System Positioning Method for Agricultural Machinery, Applied Sciences, vol. 12, no. 2. doi:10.3390/app12020457.

[6] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, P. Wang, 2023. Path planning techniques for mobile robots: Review and prospect, Expert Systems with Applications, vol. 227. doi: 10.1016/j.eswa.2023.120254.
[7] N. AbuJabal, M. Baziyad, R. Fareh, B. Brahmi, T. Rabie, M. Bettayeb, 2024. A Comprehensive Study of Recent Path-Planning Techniques for Mobile Robots, Sensors, vol. 24, no. 24, art. 8089. doi:10.3390/s24248089.
[8] Z. Yao, 2024. Agricultural machinery automatic navigation technology, Robotics and Autonomous Systems, 2024. doi: 10.1016/j.robot.2024.xxxxxx.
[9] R. Moeller, 2020. GPS-Guided Autonomous Robot with Obstacle Avoidance and Path Optimization, MSc. Thesis.
[10] I. González-Hernández, J. Flores, S. Salazar, R. Lozano, 2025. Robust and Precise Navigation and Obstacle Avoidance for Unmanned Ground Vehicle, Sensors, vol. 25, no. 14, art. 4334. doi:10.3390/s25144334.
[11] A. Elfes, 2022. Autonomous mobile robots: Perception, mapping, and navigation, IEEE Control Systems Magazine, vol. 2, no. 6, pp. 14–23.
[12] M. Moeller, 2020. GPS-guided rovers for agricultural applications, Journal of Field Robotics, vol. 37, no. 4, pp. 789–804. doi:10.1002/rob.21903.
[13] H. Ali et al., 2021. Low-cost GNSS for outdoor robot navigation, Sensors, vol. 21, no. 2, pp. 112–122. doi:10.3390/s21020112.
[14] S. Borenstein, Y. Koren, 1991. The vector field histogram: Fast obstacle avoidance for mobile robots," IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 278–288. doi:10.1109/70.88137.
[15] M. Deshmukh et al., 2024. PID tuning using particle swarm optimization, Control Engineering Practice, vol. 125, p. 105426. doi: 10.1016/j.conengprac.2024.105426.
[16] S. Wu et al., 2023. Network centrality-guided multi-objective path optimization, IEEE Transactions on Cybernetics, vol. 53, pp. 3456–3469.