IAES International Journal of Robotics and Automation (IJRA)
Vol. 15, No. 4, December 2026, pp. 846∼861
ISSN: 2722-2586, DOI: 10.11591/ijra.v15i4.pp846-861

An embedded vision-based autonomous line-following robot with PID trajectory control and barcode-assisted intersection navigation

Rafid Mohammed Khaleefah[1], Ali A. Abed[2], Mohammed Badr Algubili[1], Huda Mohammed Alsayednoor[3], Mahmood A. Al-Shareeda[4,5]
[1]Information and Communication Technology Centre, University of Basrah, Basrah, Iraq
[2]Mechatronics Engineering Department, University of Basrah, Basrah, Iraq
[3]Shatt Al-Arab University College, Basra, Iraq
[4]Department of Electronic Technologies, Basra Technical Institute, Southern Technical University, Basra, Iraq
[5]College of Engineering, Al-Ayen University, Thi-Qar, Iraq

Article Info

Article history:

Received Jun 9, 2026
Revised Aug 1, 2026
Accepted Nov 10, 2026

Keywords:

Autonomous mobile robot
Barcode-assisted navigation
Embedded navigation framework
Embedded vision
Line-following robot
PID trajectory control
Real-time autonomous navigation

ABSTRACT

Autonomous mobile robots have been widely applied in industrial automation, logistics, and smart indoor scenarios. These scenarios require navigation systems to adapt to resource-constrained platforms and achieve reliable real-time navigation with low computational demand. The embedded visual line-tracking framework proposed in this paper integrates the Pixy2 sensor, proportional–integral–derivative (PID) controller, and barcode-based intersection decision mechanism, and can realize precise navigation without relying on an external image processing platform. The navigation optimization framework proposed in this study integrates three categories of modules: on-board visual perception, mathematical trajectory modeling, and lightweight decision-making, to improve the cost-performance ratio of embedded navigation systems. A supporting PID controller reduces tracking errors and maintains motion stability, while barcode-aided navigation achieves reliable autonomous route selection at predefined intersections. The embedded indoor autonomous navigation framework proposed in this study was validated in structured indoor scenarios. Equipped with the Pixy2 smart vision sensor, the framework recorded an average tracking error of 2.02 cm, a navigation accuracy of 97.8%, and a barcode recognition accuracy of 97.5%. This lightweight solution does not require an external image processing platform, and can be adapted for autonomous navigation in warehouses, laboratories, and medical settings.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Mahmood A. Al-Shareeda
Department of Electronic Technologies, Basra Technical Institute, Southern Technical University
Basra 61001, Iraq
Email: mahmood.alshareedah@stu.edu.iq

1. INTRODUCTION

Autonomous mobile robots have become an essential component of modern intelligent systems owing to their ability to perform navigation, transportation, inspection, and monitoring tasks with minimal human intervention [1]–[3]. Their adoption has expanded across warehouses, hospitals, manufacturing plants, laboratories, and smart logistics environments, where reliable autonomous navigation directly affects operational

efficiency and safety [4]–[6]. In structured indoor environments, line-following navigation remains one of the most practical and cost-effective solutions because of its simple infrastructure requirements, high operational reliability, and ease of deployment [7], [8].

Conventional line-following robots primarily employ infrared sensor arrays for path detection [9]–[11]. Although such systems feature fast response speed and low deployment costs, they are vulnerable to interference from light conditions, surface reflection, and sensor noise. They also have limited environmental perception capabilities, cannot independently make autonomous path decisions at multi-branch intersections, and are only suitable for simple indoor scenarios.

To overcome the limitations of previous schemes, visual navigation has become a research hotspot: compared with traditional line sensors, it can provide rich environmental information and offers higher flexibility [12], [13]. However, most current mainstream solutions rely on Raspberry Pi devices equipped with OpenCV, which raises the system's complexity, cost, power consumption, and computing power requirements, restricting the practical deployment of lightweight robots.

In autonomous navigation systems, apart from the perception module, trajectory control is the core element that ensures accurate and stable navigation [14]–[16]. The widely used proportional–integral–derivative (PID) controller features simple implementation, low computing power consumption, and strong robustness. It can improve path accuracy, suppress oscillations, and maintain stable motion by compensating for tracking errors [17].

The field of embedded robots has achieved many notable advances in recent years, but it still faces four core unresolved challenges: building lightweight embedded vision systems, realizing independent operation that does not rely on external platforms, achieving stable trajectory tracking with low complexity, and completing reliable navigation. Furthermore, most existing studies only optimize a single module in isolation, and very few can integrate three core modules within a resource-constrained architecture.

Building on the unsolved navigation challenges introduced in the preceding text, this study presents its core achievement: an autonomous line-tracking framework that integrates the Pixy2 smart vision sensor, PID trajectory control, and a barcode-assisted intersection navigation module, all unified under a single embedded architecture. No additional new sensing hardware was developed for this work; instead, the framework focuses on integrating the three aforementioned core capabilities, is suited for structured indoor navigation, and features a low cost. The main contributions of this work are summarized as follows:

a. The lightweight embedded navigation framework developed by our research team integrates the Pixy2 smart vision sensor and the Arduino mobile robot. It can complete on-board line detection and barcode recognition, eliminating the need to rely on any external image processing platform.
b. In autonomous navigation scenarios, the PID trajectory control strategy can compensate for tracking deviations and maintain stable path following performance.
c. This paper independently proposed the barcode-assisted intersection decision mechanism, which can rely on reliable embedded visual information to successfully achieve autonomous route selection at predefined intersections.
d. This study constructed a mathematical model that describes the interactions among visual perception, tracking error estimation, PID control, and the motion of a differential-drive robot.
e. This study evaluates the self-developed indoor navigation system across four types of scenarios: straight-line paths, curved paths, barcode recognition, and intersection navigation, along three dimensions: tracking accuracy, navigation reliability, and system robustness.
f. We compared the navigation framework proposed in this paper with representative, mature navigation systems in the field using objective performance metrics, to verify its suitability for lightweight, low-cost embedded navigation scenarios.

The remainder of this paper proceeds sequentially through the following sections. Section 2 reviews related work in autonomous line-following and vision-based navigation systems. Section 3 presents the system model and problem formulation. Section 4 describes the proposed embedded vision-based navigation framework. Section 5 introduces the mathematical modeling and control design. Section 6 proposes the barcode-assisted intersection decision-making algorithm. Section 7 discusses research findings and comparative evaluation. Section 8 elaborates on future research directions, and section 9 concludes the entire paper.

2. RELATED WORK

In recent years, autonomous line-following robots have gained widespread attention, and have been implemented in the fields of industrial automation, warehousing and logistics, medical assistance, and intelligent transportation.

2.1. Infrared sensor-based line-following systems

Ramalakshmi *et al.* [18] proposes an intelligent waste collection and transportation system based on autonomous path-tracking robots. The system deploys infrared sensors in charge of path tracking, ultrasonic sensors responsible for obstacle avoidance, and a microcontroller to coordinate all operations. Experimental verification confirms that the system is practical and reliable, and can be implemented in scenarios including smart city operations and industrial waste management.

Boniol *et al.* [19] study presents a prototype of an indoor disinfection line-following robot built on Arduino, which is equipped with an automatic alcohol spraying mechanism. The robot integrates three core units: infrared path tracking, passive infrared human body detection, and servo-controlled spraying. Experiments confirm that this low-cost platform is compatible with structured indoor environments, and combines the functions of autonomous navigation and automatic spraying.

Neamah *et al.* [20] developed a high-speed line-following robot integrated with a dual-loop PID control system, which is equipped with a real-time sensing solution that fuses data from encoders, gyroscopes and accelerometers. This study also proposed a curvature-based speed optimization strategy. After testing under variable path working conditions, all three core performance indicators of the robot were effectively improved.

However, it is vulnerable to interference from lighting and ground reflections, suffers from sensor noise, cannot handle complex intersections and curved paths, and also lacks an intelligent decision-making mechanism for autonomous route selection.

2.2. Vision-based navigation systems

Traditional infrared sensors in the field of robot navigation have performance limitations [21]–[23]. For this reason, the academic community has increasingly adopted visual navigation technologies, which can provide richer environmental information. A study by Pati and Kala [24] on a vision system equipped with a PID controller confirmed that the performance of this type of technology is far superior to that of traditional solutions.

The study led by Huang *et al.* [25] addresses the problem of unmanned aerial vehicle (UAV) navigation failure in low-altitude GPS-denied environments. It carries out targeted analyses of visual sensing technology, navigation algorithms, and the challenges of real-world implementation. Supported by multimodal sensor fusion and algorithm optimization, the study verifies that visual navigation can greatly improve UAVs' positioning accuracy, environmental perception, and autonomous decision-making capabilities.

Dixit *et al.* [26] proposes a visual dynamic path planning framework for autonomous ground robots that is adapted for unstructured environments such as forest roads. Similarly, the vision-guided navigation framework proposed by Farkh and Aljaloud [27] adopts image line detection and PID control, achieving improved tracking accuracy and smoother navigation performance. However, this framework only focuses on trajectory tracking, and does not cover intelligent intersection processing scenarios.

Yu *et al.* [28] focus on the research field of indoor visual navigation that achieves camera localization via absolute pose regression (APR) in GPS-denied environments. Schmidt *et al.* [29] proposes a real-time credibility assessment framework for the visual perception system used in autonomous marine navigation. The framework can evaluate sensor performance in dynamic environments and identify declining reliability via degradation indicators.

Navigation systems based on OpenCV have been widely researched in academic circles. In Li [30] developed a trajectory-following vehicle leveraging this technology, which can realize target recognition and path tracking. While its environmental perception capability has been improved, its high demand for computing power has driven up both hardware costs and power consumption.

2.3. Intelligent control strategies

To improve navigation performance, the academic community has explored a variety of control algorithms [31], [32]. Among these, PID control is the most widely used, owing to its simplicity, robustness,

and suitability for real-time applications. It can adjust steering commands based on current, cumulative, and predicted tracking deviations to reduce tracking errors.

Wang *et al.* [33] propose a predictive load frequency control (LFC) framework that integrates high-precision wind power prediction and model predictive control (MPC). To support this framework, an enhanced GiReformer is developed to realize multi-step wind power prediction, while a Laguerre basis-based MPC is adopted to reduce computational complexity and handle operational constraints.

Izci *et al.* [34] focuses on the continuous stirred tank heater (CSTH) process, and proposes a temperature control strategy that integrates the two-degree-of-freedom PID acceleration controller (2DOF-PIDA) and the starfish optimization algorithm (SFOA).

Beyond traditional PID control, intelligent control technologies such as fuzzy logic have been proposed. The type-2 fuzzy logic controller developed by Castillo *et al.* [35] for vision-based line-tracking robots enhanced robustness under uncertain operating conditions, but its high computational complexity renders it difficult to deploy on resource-constrained embedded platforms.

2.4. Barcode and marker-based navigation approaches

Previous visual navigation research in the field of autonomous path guidance has mostly centered on navigation technologies that use visual markers, QR codes, and barcodes [36]–[38]. Visual markers can output clear navigation information and improve the reliability of intersection decision-making, but most existing solutions rely on external image processing platforms and only focus on the single task of localization.

Barcode-assisted navigation achieves accurate positioning through barcodes pre-deployed at work sites, is compatible with most indoor operation scenarios, and features core application advantages of low deployment costs and stable positioning.

2.5. Research gap analysis

To address the existing research gaps in the current field of autonomous navigation, this study proposes an embedded visual autonomous navigation framework that integrates Pixy2 on-board visual perception, PID trajectory control, and barcode-assisted intersection navigation. Developed for indoor scenarios, this framework can be practically deployed without requiring any additional sensing hardware. A comparison between the proposed framework and representative navigation systems is presented in Table 1.

Table 1. Comparison of representative vision-based navigation systems

Study	Vision Platform	Controller	External Computing	Intersection Navigation	Primary Focus
Pati and Kala [24]	Vision Camera	PID	Yes	No	Vision-based line tracking
Castillo *et al.* [35]	Vision Camera	Type-2 Fuzzy Logic	Yes	No	Robust tracking under uncertainty
Farkh and Aljaloud [27]	Vision Camera	PID	Yes	Limited	Vision-guided trajectory tracking
Li [30]	Camera + OpenCV	PID	Yes	Limited	OpenCV-based visual navigation
Proposed Framework	**Pixy2 Embedded Vision**	**PID**	**No**	**Yes (Barcode-Assisted)**	**Embedded perception and autonomous intersection navigation**

3. SYSTEM MODEL AND PROBLEM FORMULATION

This section presents the system architecture and mathematical formulation of the embedded vision-based navigation framework. The objective is to establish an integrated navigation model that combines on-board visual perception, real-time trajectory control, and barcode-assisted decision-making for autonomous operation in structured indoor environments.

3.1. System model

The proposed navigation framework consists of four interconnected functional modules that collectively enable real-time line following and autonomous intersection navigation. The overall system architecture is illustrated in Figure 1.

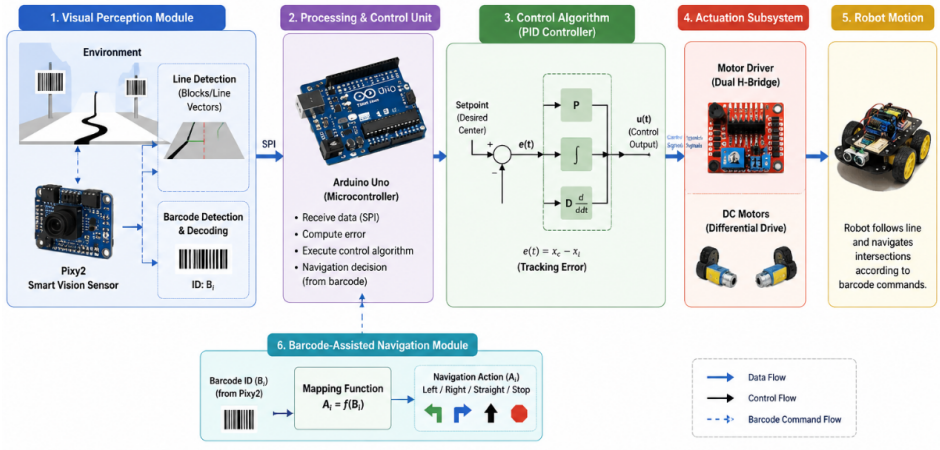

Figure 1. Overall architecture of the embedded vision-based autonomous navigation framework

The functionality of each module is summarized as follows.

a. Embedded visual perception module: According to research by Bondarko, the Pixy2 smart vision sensor can complete on-board image acquisition, line vector extraction and barcode recognition, and transmit data to the controller via the SPI communication interface, a function that reduces the computational overhead of embedded processors [39].
b. Processing and control unit: This autonomous navigation system uses the Arduino Uno microcontroller as its central controller. It receives line position and barcode information from visual sensors, calculates the lateral tracking error, runs PID to generate steering commands, and the relevant logic [40].
c. Motion control subsystem: The robot employs a differential-drive mechanism consisting of a dual H-bridge motor driver and four DC geared motors [41]. The controller adjusts the pulse-width modulation (PWM) duty cycle supplied to each motor, enabling accurate speed regulation and trajectory correction.
d. Barcode-assisted navigation module: Barcodes deployed at pre-defined intersections carry navigation instructions. After being detected by visual sensors, these instructions can be converted into driving actions: turning left, turning right, proceeding straight, and stopping. This approach requires no additional positioning infrastructure, and its rationality has already been supported by existing academic literature [42], [43].

The embedded vision sensor of this navigation system continuously collects visual information, extracts line feature and barcode identifier data, and uploads these datasets. The controller calculates tracking errors to generate navigation commands, drives the differential drive subsystem to adjust wheel speeds, and achieves continuous trajectory tracking and autonomous intersection navigation.

3.2. Navigation environment

The indoor navigation robot system defined in this paper is deployed in structured indoor passageways marked with high-contrast navigation lines. Encoded barcodes are pre-set at all intersections. When carrying out its work, the robot tracks along the path, and synchronously parses the barcodes to determine the correct operation to execute at the intersection.

3.3. Problem formulation

The visual navigation of the autonomous mobile robot in this study must achieve a dual set of objectives: first, minimize the lateral deviation between the desired trajectory and the measured path, and second, maintain motion stability. To derive the lateral tracking error, this study defines xc as the coordinate of the camera image center, and xl as the position of the detected path line. $e(t) = x_c - x_l$. Accurate navigation is achieved when the tracking error approaches zero throughout the navigation process, $e(t) \to 0, \quad \forall t \in [0, T]$. Accordingly, the trajectory tracking problem is formulated as the minimization of the accumulated tracking error, $\min J = \int_0^T e^2(t)\, dt$, where J denotes the cumulative tracking cost, $e(t)$ is the lateral tracking error, and T represents the navigation duration. Minimizing this objective improves trajectory accuracy while maintaining stable robot motion.

3.4. Intersection decision problem

When the robot reaches an intersection, multiple navigation paths become available. Conventional line-following systems frequently encounter ambiguity in these situations because line information alone is insufficient to determine the intended route. To resolve this issue, barcode markers are employed to explicitly encode navigation instructions.

4. PROPOSED EMBEDDED VISION-BASED NAVIGATION FRAMEWORK

This study proposes an embedded visual navigation framework that is compatible with structured indoor environments, and can realize autonomous line-following and intersection-aware navigation. The framework integrates four core modules including a lightweight visual sensor, and boasts the dual advantages of accurate and reliable navigation and low computational complexity.

The overall architecture of the proposed system is illustrated in Figure 2. The framework consists of four main layers: i) visual perception layer, ii) control and processing layer, iii) actuation layer, and iv) navigation decision layer.

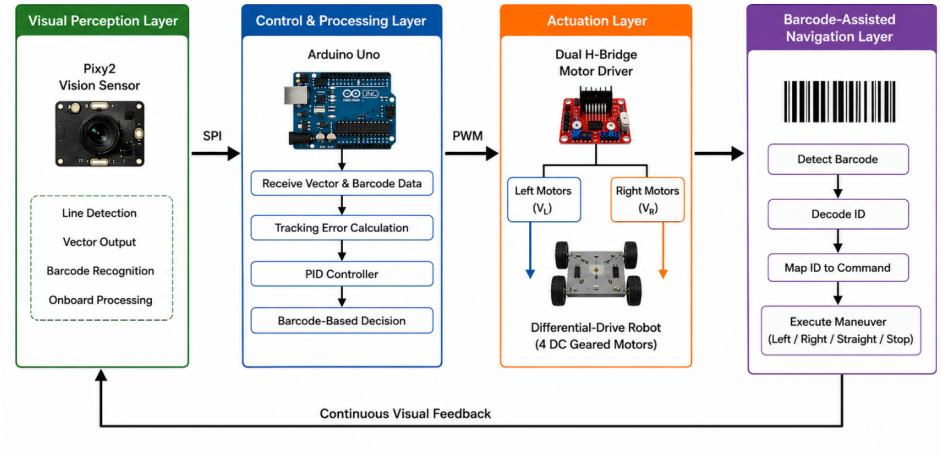

Figure 2. Overall architecture of the proposed embedded vision-based autonomous navigation system integrating Pixy2 vision sensing, PID-based trajectory control, barcode-assisted decision-making, and differential-drive robot actuation

4.1. Visual perception layer

When autonomous robots travel to an intersection, they face multiple available paths. Traditional line-following systems cannot determine a pre-set route relying only on line information, so this paper uses barcode markers to encode navigation commands to eliminate path ambiguity.

The core responsibility of the visual perception layer is to acquire and process the environmental information required for autonomous navigation. This layer is equipped with the Pixy2 smart vision sensor, which can detect navigation lines and recognize barcodes at pre-set decision points. Compared with traditional camera systems that rely on external image processing platforms, this sensor has built-in on-board processing capabilities that can reduce the computing load on the embedded controller.

This study constructs a navigation system for resource-constrained embedded robot platforms. The system is equipped with an Arduino Uno controller and path awareness sensors, and transmits two types of data via the SPI protocol: path geometric features that support the estimation of lateral tracking errors for PID trajectory control, and barcode identifiers that support intersection scenario processing. Its on-device visual processing architecture strikes a balance between reliable real-time detection and low computational complexity, adapting to the platform's resource constraints.

4.2. Control and processing layer

The control and processing layer of the embedded navigation framework constructed in this study takes the Arduino Uno microcontroller as its core, interfaces with the Pixy2 visual sensor via the SPI protocol,

and receives the real-time line vectors and barcode information generated by the sensor.

After the trajectory tracking control system of the wheeled mobile platform described in this paper receives visual data, the main controller first calculates the lateral tracking error by comparing the detected line position with the reference value of the camera frame center, inputs this error into a PID controller to generate a steering correction value, then sends a control signal to the motor driver to complete the adjustment of differential wheel speeds.

The embedded system used in this paper employs the SPI communication interface, which can reliably and efficiently transmit processed visual information between the vision sensor and the embedded controller. This design decouples perception and control, allowing the Arduino Uno to focus exclusively on control computation and navigation decision-making, without needing to undertake any image processing work. The system achieves real-time navigation while keeping communication overhead low.

4.3. Actuation layer

The drive execution layer receives steering commands from the PID controller, converts them into differential wheel motion, and realizes accurate trajectory tracking and autonomous maneuvering. The differential-drive chassis robot platform designed in this paper is equipped with four DC gear motors. The left and right wheel groups have independent speed control; the platform realizes forward movement and steering by adjusting the speed difference between the wheels, and can stably maintain its motion trajectory throughout the entire operation process.

Motor commands are generated using pulse-width modulation (PWM) signals produced by the Arduino Uno controller and applied through a dual-channel H-bridge motor driver. The driver regulates the voltage supplied to each motor, enabling continuous speed adjustment and smooth steering during navigation.

Let V_L and V_R denote the linear velocities of the left and right wheel sets, respectively. The steering correction generated by the PID controller, denoted by $u(t)$, modifies the nominal forward velocity V_{base} according to $V_L = V_{\text{base}} + u(t)$, $V_R = V_{\text{base}} - u(t)$. When a robot deviates from its predefined trajectory, the controller will adjust the rotational speed of its wheels in a differentiated way to reduce tracking errors and guide the robot back to the preset path. During intersection navigation, the barcode-aided decision module updates steering instructions based on decoded navigation commands, enabling the robot to complete actions of turning left, turning right, moving straight, or stopping while maintaining motion stability.

Integrating pulse-width modulation-based motor control with differential drive kinematics allows for the creation of a lightweight and efficient drive mechanism that meets the real-time requirements of autonomous navigation in structured indoor environments.

4.4. Barcode-assisted navigation layer

This paper proposes a barcode-aided navigation layer, which binds pre-defined barcode identifiers to navigation instructions and can achieve deterministic routing at intersections. Compared with traditional visual line-following systems, this layer eliminates path ambiguity and supports autonomous maneuver operations.

Let B_i denote the detected barcode identifier and A_i represent the corresponding navigation action. The mapping between barcode identifiers and navigation commands is defined as $A_i = f(B_i)$, where $f(\cdot)$ denotes the predefined barcode-to-command mapping function. For example, $f(B_1) =$ Left, $f(B_2) =$ Right, $f(B_3) =$ Straight, $f(B_4) =$ Stop. After the automatic navigation system identifies the trigger barcode, it first completes the pre-set path control step, then initiates the system reset procedure, and prepares to enter the next round of operating status.

4.5. Implementation details

The proposed navigation framework was implemented using an Arduino Uno microcontroller integrated with a Pixy2 smart vision sensor through the serial peripheral interface (SPI) communication protocol. During operation, the Pixy2 sensor performs onboard image processing and transmits line-vector coordinates together with barcode identifiers to the controller, thereby eliminating the need for external image-processing hardware.

The Arduino Uno executes the navigation algorithm in a cyclic manner. For each control cycle, the received line-vector information is used to compute the lateral tracking error, which is subsequently processed by the PID controller to generate the steering correction signal. The resulting control signal is converted into pulse-width modulation (PWM) outputs that regulate the left and right wheel velocities through a dual-channel H-bridge motor driver. When a barcode is detected, the corresponding navigation command is retrieved from

a predefined lookup table and temporarily overrides the trajectory controller to execute the required maneuver. After completing the maneuver, the controller automatically resumes the normal line-following mode.

The navigation framework was implemented using lightweight integer arithmetic and simple decision logic to ensure efficient execution on the resource-constrained embedded platform. Since the computationally intensive image-processing tasks are performed directly by the Pixy2 sensor, the Arduino Uno is dedicated to control computation, barcode interpretation, and motor actuation, thereby supporting reliable real-time autonomous navigation.

5. MATHEMATICAL MODELING OF LINE TRACKING

This section presents the mathematical formulation of the proposed vision-based line-tracking framework. The model describes the robot kinematics, line detection process, tracking error computation, and PID-based steering control used to achieve stable trajectory tracking. The overall mathematical workflow is illustrated in Figure 3.

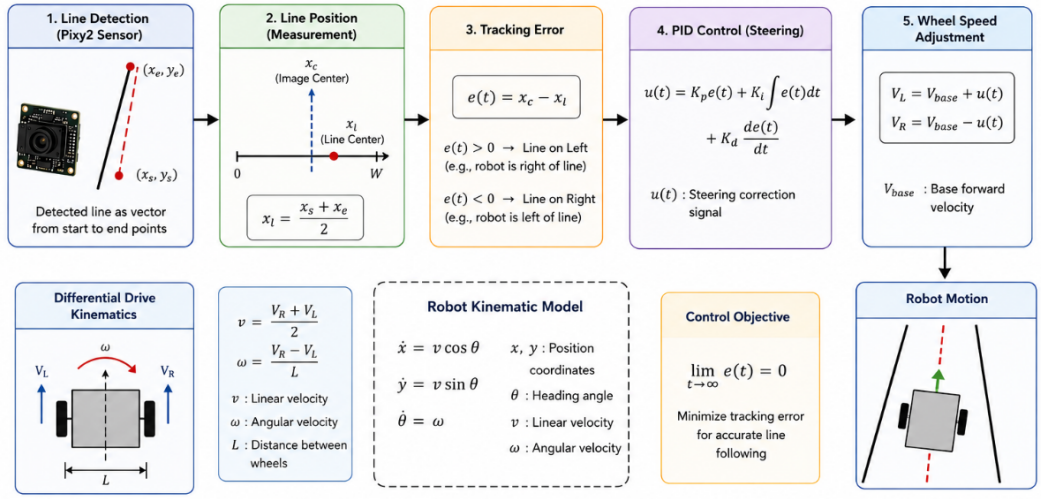

Figure 3. Mathematical model of the proposed vision-based line-tracking framework

a. Robot kinematic model: The robotic platform adopts a differential-drive configuration with independently controlled left and right wheel sets. The robot motion is described by $\dot{x} = v\cos\theta$, $\dot{y} = v\sin\theta$, $\dot{\theta} = \omega$, where (x, y) denotes the robot position, θ is the heading angle, v is the linear velocity, and ω is the angular velocity. The linear and angular velocities are computed as $v = \frac{V_R+V_L}{2}$, $\omega = \frac{V_R-V_L}{L}$, where V_L and V_R denote the left and right wheel velocities, respectively, and L is the distance between the wheel sets.

b. Line detection model: The Pixy2 vision sensor represents the detected navigation line as a vector with endpoints $P_s = (x_s, y_s)$ and $P_e = (x_e, y_e)$, where (x_s, y_s) and (x_e, y_e) denote the start and end coordinates of the detected line segment. The estimated horizontal position of the navigation line is $x_l = \frac{x_s+x_e}{2}$.

c. Tracking error formulation: The lateral tracking error is defined as the difference between the image center and the detected line position, $e(t) = x_c - x_l$, where x_c is the center of the camera frame and x_l is the detected line position. The objective of the controller is to minimize $e(t)$ throughout the navigation process.

d. Control objective: The trajectory tracking objective is to minimize the lateral tracking error while maintaining stable robot motion, $\lim_{t\to\infty} e(t) = 0$. This condition indicates convergence of the robot toward the desired navigation path.

e. Steering correction model: The steering correction is generated using a proportional–integral–derivative (PID) controller, $u(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de(t)}{dt}$, where K_p, K_i, and K_d denote the proportional, integral, and derivative gains, respectively. The controller output adjusts the wheel velocities according to $V_L = V_{\text{base}} + u(t)$, $V_R = V_{\text{base}} - u(t)$, where V_{base} is the nominal forward velocity. This control law continuously compensates for trajectory deviations and enables stable line-following during straight segments, curved paths, and intersection maneuvers.

6. BARCODE-ASSISTED INTERSECTION DECISION

This paper proposes a barcode-assisted intersection decision-making mechanism that establishes a mathematical model for autonomous path selection at pre-set navigation intersections. Compared with traditional line-following systems that rely on trajectory tracking, this mechanism can output deterministic commands via barcode markers, and completely eliminates path ambiguity.

6.1. Barcode detection and decoding

Barcode markers are deployed at predefined intersections to encode navigation instructions. During navigation, the Pixy2 vision sensor continuously scans the environment and performs onboard barcode recognition. The decoded identifier is transmitted to the Arduino Uno controller through the SPI communication interface for decision making.

6.2. Navigation command mapping

The detected barcode identifier is mapped to a predefined navigation command using a lookup table, $A_i = f(B_i)$, where A_i denotes the navigation action associated with barcode identifier B_i. Typical command mappings include $f(B_1) = \text{Left}, \quad f(B_2) = \text{Right}, \quad f(B_3) = \text{Straight}, \quad f(B_4) = \text{Stop}$. This deterministic mapping enables reliable route selection at navigation intersections.

6.3. Decision procedure

The navigation decision process is activated only when a valid barcode is detected. After decoding the barcode identifier, the controller retrieves the corresponding navigation command from the lookup table and executes the required maneuver. Once the maneuver is completed, the controller automatically resumes normal PID-based line-following operation. The complete workflow of the proposed navigation framework is summarized in Algorithm 1.

Algorithm 1: Embedded vision-based navigation

```
Input: Line vector L, barcode identifier B_i
Output: Motor control command
Initialize system;
while navigation is active do
    Receive visual data from Pixy2 via SPI;
    Compute tracking error and execute PID control;
    if barcode detected then
        Decode B_i and obtain navigation command A_i = f(B_i);
        Execute maneuver (Left, Right, Straight, or Stop);
    Generate PWM signals and drive the motors;
```

6.4. Computational complexity analysis

The computational complexity of the proposed intersection decision mechanism is low because barcode recognition is performed by the onboard Pixy2 vision sensor. Consequently, the Arduino Uno controller only performs barcode lookup and command execution. Since each barcode identifier is directly associated with a predefined navigation command, the lookup operation requires constant-time complexity, $T(n) = O(1)$. Therefore, the proposed decision mechanism is computationally efficient and suitable for real-time autonomous navigation on resource-constrained embedded platforms.

7. RESULTS AND PERFORMANCE EVALUATION

This paper presents the experimental deployment results of the proposed embedded vision autonomous navigation system. The system is evaluated across four core dimensions: robot line-following accuracy, PID controller performance, barcode intersection processing capability, and navigation reliability. To verify its effectiveness and replicability, multiple sets of repeated experiments were conducted under uniform working conditions.

7.1. Experimental setup

The experimental track for the autonomous mobile robots in this study includes straight sections, curved sections, T-junctions, and crossroads. Barcodes are installed in front of each intersection. The robots are equipped with Pixy2 sensors and Arduino Uno main controllers, and all experiments are conducted under

stable indoor lighting conditions. The hardware configuration and experimental parameters are summarized in Table 2.

Table 2. Experimental setup and system parameters

Parameter	Value
Microcontroller	Arduino Uno
Vision Sensor	Pixy2 Smart Camera
Motor Driver	Dual H-Bridge Driver
Actuators	Four DC Geared Motors
Communication Interface	SPI
Robot Configuration	Differential Drive
Track Length	2 m
Track Width	1.2 m
Line Width	3 cm
Intersection Types	T and Four-Way
PID Parameters	$K_p = 0.45$, $K_i = 0.02$, $K_d = 0.12$
Base Speed	0.30 m/s
Lighting Condition	Indoor Environment
Number of Trials	5

7.2. Tracking accuracy analysis

The tracking performance of the proposed navigation framework was evaluated on straight segments, curved paths, and intersections. Five independent trials were conducted under identical experimental conditions. Table 3 summarizes the measured average and maximum lateral tracking errors.

Table 3. Statistical summary of tracking accuracy

Statistic	Value
Mean tracking error	2.02 cm
Standard deviation	0.19 cm
Minimum error	1.80 cm
Maximum error	2.30 cm
Coefficient of variation	9.4%

Across all test trials of this experiment, the system achieved an average lateral tracking error of 2.02 cm, with a maximum error of 4.38 cm. This maximum error only occurred in scenarios that require rapid steering, such as sharp turns and intersections. These results confirm that the system can achieve stable trajectory tracking in the tested indoor navigation scenarios.

7.3. PID response analysis

This study focuses on robot operation scenarios. It evaluates the effectiveness of PID controllers by analyzing the transient and steady-state behaviors of tracking errors. Figure 4 presents the trajectory tracking error response of the experimental robot. The controller successfully reduced tracking deviations and stabilized the robot around the desired trajectory with minimal oscillations. Table 4 summarizes the main control-performance metrics. The results confirm that the selected PID parameters provide a suitable balance between responsiveness and stability while minimizing tracking errors.

7.4. Barcode recognition performance

To evaluate the effectiveness of the barcode-assisted navigation mechanism, the robot was tested at multiple intersections containing predefined navigation commands. The barcode recognition accuracy is summarized in Figure 5. The recognition accuracy of the barcode-assisted navigation layer reaches 97.5%, which can support reliable instruction decoding and decision-making.

7.5. Intersection handling performance

This study evaluates the autonomous navigation capability of robots when traversing two types of intersections: T-intersections and four-way intersections. All results are presented in the intersection performance Table 5. Experimental results confirm that this navigation framework can reliably execute directed maneuvers and maintain stable trajectory tracking.

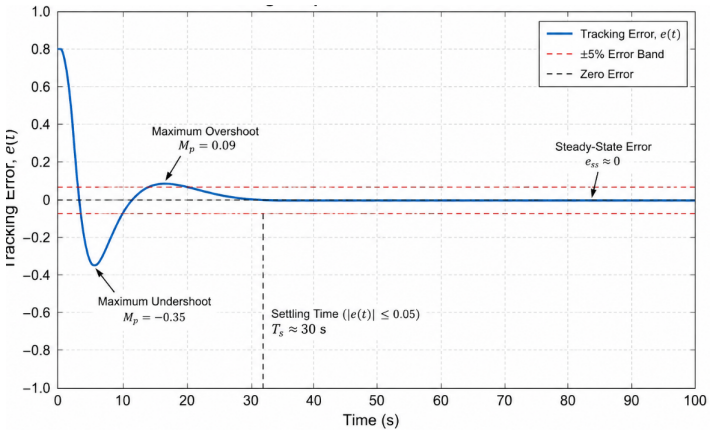

Figure 4. Tracking error dynamics demonstrating the stability and convergence of the PID controller

Table 4. PID controller performance metrics

Metric	Value
Average tracking error	0.6 cm
Maximum overshoot	3 cm
Settling time	8 s
Steady-state error	0 cm
Navigation accuracy	97.8%

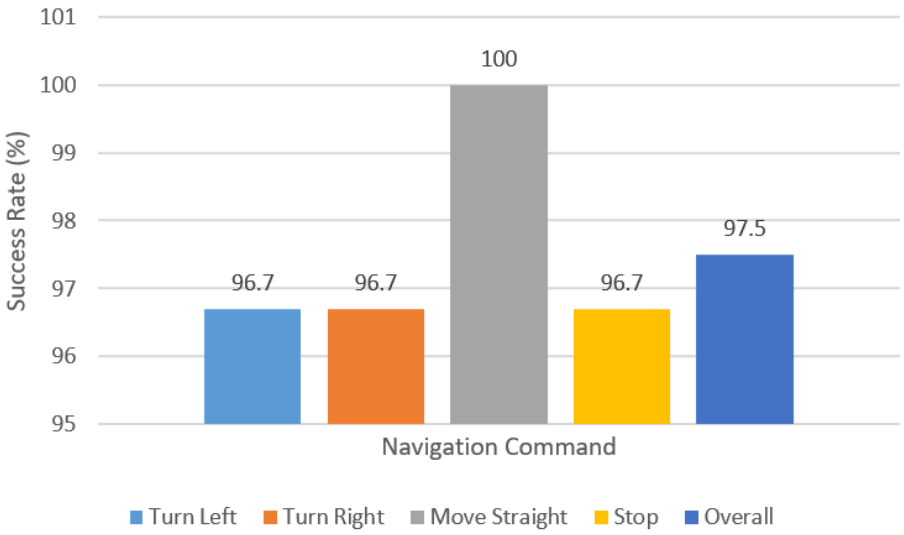

Figure 5. Barcode recognition performance

Table 5. Intersection navigation performance

Intersection type	Number of tests	Success rate
T-Intersection	30	96.7%
Four-Way Intersection	30	96.7%
Overall	**60**	**96.7%**

7.6. Comparative evaluation

To verify the effectiveness of the self-developed embedded visual navigation framework proposed in this study, we selected two representative categories of navigation systems from existing literature, and carried

out a comprehensive comparative evaluation across six core dimensions including sensing technology and control strategy.

7.6.1. Comparison with existing navigation systems

Table 6 compares the proposed framework with representative vision-based navigation systems reported in the literature. The comparison focuses on objective architectural characteristics, including the vision platform, control strategy, dependence on external image-processing hardware, intersection navigation capability, and reported navigation accuracy.

Table 6. Comparison with representative vision-based navigation systems

Study	Vision platform	Controller	External processing	Intersection navigation	Navigation accuracy
Pati and Kala [24]	Vision Camera	PID	Required	No	94.0%
Castillo *et al.* [35]	Vision Camera	Type-2 Fuzzy	Required	No	95.6%
Farkh and Aljaloud [27]	Vision Camera	PID	Required	Limited	96.1%
Li [30]	Camera + OpenCV	PID	Required	Limited	96.5%
Proposed Framework	**Pixy2 Embedded Vision**	**PID**	**Not Required**	**Barcode-Assisted**	**97.8%**

As shown in Table 6, the proposed framework combines onboard embedded vision, PID-based trajectory control, and barcode-assisted intersection navigation within a lightweight architecture that does not require external image-processing hardware. Compared with representative vision-based navigation systems, the proposed framework achieves the highest reported navigation accuracy while maintaining autonomous intersection navigation using embedded visual processing.

7.6.2. Comparison of hardware complexity

The visual navigation framework proposed in this paper adopts a lightweight hardware architecture, and features a lower hardware barrier than existing navigation systems that rely on OpenCV. The framework's visual sensor has built-in line detection and barcode recognition functions, so the microcontroller only undertakes control and decision-making tasks. Table 7 summarizes the hardware-level comparison.

Table 7. Hardware-level comparison

System	Processing unit	Power	Complexity
IR-Based Robot	Arduino	Low	Low
OpenCV-Based Robot	Raspberry Pi	High	High
Vision PID Robot	Embedded PC	Medium	Medium
Proposed Framework	**Arduino + Pixy2**	**Low**	**Low**

7.7. Discussion

The experimental results demonstrate that the proposed embedded vision-based navigation framework provides reliable line-following and intersection navigation using a lightweight hardware architecture. The average tracking error of 2.02 cm and the high navigation success rates indicate that the integration of embedded visual perception with PID-based trajectory control enables stable operation in structured indoor environments.

The barcode-aided navigation mechanism is an extension of traditional line-following solutions, with its core improvement being the ability to make deterministic decisions at intersections. Unlike traditional solutions that only rely on path geometry, the framework proposed in this study uses pre-defined barcode markers to transmit clear navigation instructions. This design not only reduces path ambiguity, but also maintains continuous trajectory tracking.

The embedded visual navigation architecture proposed in this study has been verified through experiments, and its core advantages are prominent. In terms of hardware, the Pixy2 visual sensor undertakes image processing and barcode decoding tasks, while the Arduino Uno is only responsible for control and decision-making; this architecture can operate in real time without relying on an external computing platform. Compared with mainstream solutions, it has excellent navigation accuracy, simple deployment, and lower costs, making it suitable for various scenarios such as warehouse automation and laboratory logistics.

7.8. Experimental limitations

The navigation framework proposed in this paper can operate stably under preset evaluation conditions, but it still has three limitations. All experiments were only completed in a structured indoor environment

with preset paths and barcode markers. Only a small number of tests were conducted under constant light conditions, and no assessments were carried out in dynamic environments with moving obstacles or changing light. These limitations may all affect the framework's deployment performance in real-world scenarios.

8. FUTURE RESEARCH DIRECTIONS

Several research directions can further enhance the proposed embedded vision-based navigation framework.

a. Adaptive control: Future work will investigate adaptive and learning-based control strategies, such as adaptive PID, fuzzy control, and model predictive control (MPC), to improve navigation performance under varying operating conditions.
b. Dynamic environment navigation: The current framework will be extended by incorporating obstacle detection and real-time path planning to support autonomous navigation in dynamic and unstructured environments.
c. Advanced visual perception: Future studies will explore the integration of lightweight deep-learning models and advanced visual markers, such as AprilTags and QR codes, to improve perception robustness and localization accuracy.
d. Multi-robot and IoT integration: The proposed framework can be expanded to cooperative multi-robot systems and integrated with edge computing and IoT technologies for coordinated navigation and remote monitoring in smart industrial environments.

9. CONCLUSION

This paper proposes an autonomous track-following robot equipped with a visual module. Its core configurations include the Pixy2 intelligent visual sensor, the PID trajectory control algorithm, and a barcode-assisted intersection navigation module. Without relying on external image processing hardware, the robot can achieve autonomous path tracking and deterministic path selection. Tests have verified its stable performance. This robot is applicable to four major fields: warehouse automation, laboratory logistics, educational robots, and indoor service robots. Future research will expand into three areas: adaptive control, obstacle avoidance, and enhanced visual perception.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT (MANDATORY)

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Rafid Mohammed Khaleefah	✓	✓	✓		✓	✓		✓	✓		✓			
Ali A. Abed		✓		✓	✓					✓		✓		
Mohammed Badr Algubili				✓		✓	✓	✓		✓				
Huda Mohammed Alsayednoor		✓	✓		✓		✓	✓		✓				
Mahmood A. Al-Shareeda	✓	✓			✓					✓		✓	✓	

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal Analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject Administration
Fu : **Fu**nding Acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

DATA AVAILABILITY

Data availability is not applicable to this paper as no new data were created or analyzed in this study.

REFERENCES

[1] M. B. Alatise and G. P. Hancke, "A review on challenges of autonomous mobile robot and sensor fusion methods," *IEEE access*, vol. 8, pp. 39 830–39 846, 2020, doi: 10.1109/ACCESS.2020.2975643.

[2] Y. Liu, S. Wang, Y. Xie, T. Xiong, and M. Wu, "A review of sensing technologies for indoor autonomous mobile robots," *Sensors*, vol. 24, no. 4, p. 1222, 2024, doi: 10.3390/s24041222.

[3] T. Dai *et al.*, "Autonomous mobile robots for exploratory synthetic chemistry," *Nature*, vol. 635, no. 8040, pp. 890–897, 2024, doi: 10.1038/s41586-024-08173-7.

[4] V. Ušinskis, M. Nowicki, A. Dzedzickis, and V. Bučinskas, "Sensor-fusion based navigation for autonomous mobile robot," *Sensors*, vol. 25, no. 4, p. 1248, 2025, doi: 10.3390/s25041248.

[5] M. N. Jawad *et al.*, "Energy-aware lifecycle cryptographic trust framework for secure communication in sustainable power systems," *International Journal of Robotics and Control Systems*, vol. 6, no. 3, pp. 2548–2572, 2026, doi: 10.31763/ijrcs.v6i3.2536.

[6] Y. Chen, C. Cheng, Y. Zhang, X. Li, and L. Sun, "A neural network-based navigation approach for autonomous mobile robot systems," *Applied Sciences*, vol. 12, no. 15, p. 7796, 2022, doi: 10.3390/app12157796.

[7] Y. Tang, M. A. Zakaria, and M. Younas, "Path planning trends for autonomous mobile robot navigation: A review," *Sensors*, vol. 25, no. 4, p. 1206, 2025, doi: 10.3390/s25041206.

[8] S. Ganesan, B. Ramalingam, and R. E. Mohan, "A hybrid sampling-based RRT* path planning algorithm for autonomous mobile robot navigation," *Expert Systems with Applications*, vol. 258, p. 125206, 2024, doi: 10.1016/j.eswa.2024.125206.

[9] G. Dong, X. Lou, and H. Wang, "Towards robust UAV navigation in agriculture: Key technologies, application, and future directions," *Plants*, vol. 15, no. 9, p. 1303, 2026, doi: 10.3390/plants15091303.

[10] M. Al-Maliki *et al.*, "Automation under constraint evidence of automated software engineering practices in communication system," *International Journal of Robotics and Control Systems*, vol. 6, no. 4, pp. 3389–3411, 2026, doi: 10.31763/ijrcs.v6i4.2500.

[11] L. Yang *et al.*, "A review of the development of instrument and meter inspection robot," *Review of Scientific Instruments*, vol. 96, no. 10, p. 101501, 2025, doi: 10.1063/5.0280132.

[12] M. Y. Arafat, M. M. Alam, and S. Moh, "Vision-based navigation techniques for unmanned aerial vehicles: Review and challenges," *Drones*, vol. 7, no. 2, p. 89, 2023, doi: 10.3390/drones7020089.

[13] D. Li, M. Zhu, S. Wang, Y. Hu, F. Yuan, and J. Yu, "A vision-based navigation system with markerless image registration and position-sensing localization for oral and maxillofacial surgery," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–11, 2023, doi: 10.1109/TIM.2023.3241059.

[14] Q. Mumuni, A. Olayiwola-Mumuni, and A. Yussouff, "The advent of the proportional integral derivative controller: A review," *Journal of Advances in Engineering Technology*, vol. 2, no. 10, pp. 5–22, 2023, doi: 10.24412/2181-1431-2023-2-5-22.

[15] M. M. Qasim *et al.*, "Privacy-preserving software engineering protocol for V2V charging using lattice-based linkable ring signature," *Journal of Communications*, vol. 21, no. 2, pp. 261–273, 2026, doi: 10.12720/jcm.21.2.261-273.

[16] M. Filo, S. Kumar, and M. Khammash, "A hierarchy of biomolecular proportional-integral-derivative feedback controllers for robust perfect adaptation and dynamic performance," *Nature communications*, vol. 13, no. 1, p. 2119, 2022, doi: 10.1038/s41467-022-29640-7.

[17] A. A. Jamil, W. F. Tu, S. W. Ali, Y. Terriche, and J. M. Guerrero, "Fractional-order PID controllers for temperature control: A review," *Energies*, vol. 15, no. 10, p. 3800, 2022, doi: 10.3390/en15103800.

[18] K. Ramalakshmi, B. Mohan, P. Vigneshwaran, M. Suresh, K. Subramani, and R. Venkatesh, "Smart waste management system using autonomous line following robot," in *2025 2nd International Conference on Integration of Computational Intelligent System (ICICIS)*. IEEE, 2025, pp. 1–5.

[19] M. J. C. Boniol, S. X. O. Aggalut, T. Danilo Jr, K. D. Sulayman, J. K. C. Tapic, and G. J. Santos, "Arduino-based line-following robot with automated alcohol spraying mechanism,"*UKR Journal of Multidisciplinary Studies (UKRJMS)*, 2026, doi: 10.5281/zenodo.20357040.

[20] H. A. Neamah and L. Xiaodong, "Development of an advanced high-speed line-following robot for enhancing autonomous robotics," in *2025 IEEE 12th International Conference on E-Learning in Industrial Electronics (ICELIE)*. IEEE, 2025, pp. 1–6, doi: 10.1109/ICELIE64733.2025.11244757.

[21] E. A. Rodríguez-Martínez, W. Flores-Fuentes, F. Achakir, O. Sergiyenko, and F. N. Murrieta-Rico, "Vision-based navigation and perception for autonomous robots: Sensors, SLAM, control strategies, and cross-domain applications—a review," *Eng*, vol. 6, no. 7, p. 153, 2025, doi: 10.3390/eng6070153.

[22] Y. Alkendi, L. Seneviratne, and Y. Zweiri, "State of the art in vision-based localization techniques for autonomous navigation systems," *IEEE Access*, vol. 9, pp. 76 847–76 874, 2021, doi: 10.1109/ACCESS.2021.3082778.

[23] M. H. Abidi, A. N. Siddiquee, H. Alkhalefah, and V. Srivastava, "A comprehensive review of navigation systems for visually impaired individuals," *Heliyon*, vol. 10, no. 11, 2024, doi: 10.1016/j.heliyon.2024.e31825.

[24] C. S. Pati and R. Kala, "Vision-based robot following using PID control," *Technologies*, vol. 5, no. 2, p. 34, 2017, doi: 10.3390/technologies5020034.

[25] T. Huang, "Investigation into vision-based navigation systems for unmanned aerial vehicles," in *2025 2nd International Conference*

on Mechanics, Electronics Engineering and Automation (ICMEEA 2025). Atlantis Press, 2025, pp. 181–190, doi: 10.2991/978-94-6463-821-9_21.
[26] R. Dixit and K. Subramaniam, "Vision-based navigation system for autonomous ground robots in dynamic environments," *Association Journal of Interdisciplinary Technics in Engineering Mechanics*, vol. 3, no. 2, pp. 10–13, 2025.
[27] R. Farkh and K. Aljaloud, "Vision navigation based PID control for line tracking robot," *Intelligent Automation & Soft Computing*, vol. 35, no. 1, pp. 901–911, 2023, doi: 10.32604/iasc.2023.027614.
[28] J. Yu, Y. Yang, and V. Namboodiri, "Large-scale benchmarking of vision-based indoor localization using absolute pose regression," in *2025 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2025, pp. 1–6, doi: 10.1109/IPIN66788.2025.11213320.
[29] J. E. Schmidt, B. Rokseth, and R. Galeazzi, "Online performance evaluation of vision-based perception system for autonomous ship navigation," in *Journal of Physics: Conference Series*, vol. 3123, no. 1. IOP Publishing, 2025, p. 012025, doi: 10.1088/1742-6596/3123/1/012025.
[30] J. Li, "OpenCV-based PID control line following vehicle with object recognition and reaction," in *AIP Conference Proceedings*, vol. 3144, no. 1. AIP Publishing LLC, 2024, p. 050009, doi: 10.1063/5.0214211.
[31] A. Francis *et al.*, "Principles and guidelines for evaluating social robot navigation algorithms," *ACM Transactions on Human-Robot Interaction*, vol. 14, no. 2, pp. 1–65, 2025, doi: 10.1145/3700599.
[32] S. Silvestrini and M. Lavagna, "Deep learning and artificial neural networks for spacecraft dynamics, navigation and control," *Drones*, vol. 6, no. 10, p. 270, 2022, doi: 10.3390/drones6100270.
[33] P. Wang, J. Guo, F. Cheng, Y. Gu, F. Yuan, and F. Zhang, "A MPC-based load frequency control considering wind power intelligent forecasting," *Renewable Energy*, vol. 244, p. 122636, 2025, doi: 10.1016/j.renene.2025.122636.
[34] D. Izci *et al.*, "A new intelligent control strategy for CSTH temperature regulation based on the starfish optimization algorithm," *Scientific Reports*, vol. 15, no. 1, p. 12327, 2025, doi: 10.1038/s41598-025-96621-3.
[35] O. Castillo, P. Cortés-Antonio, P. Melin, and F. Valdez, "Type-2 fuzzy control for line following using line detection images," *Journal of Intelligent & Fuzzy Systems*, vol. 39, no. 5, pp. 6089–6097, 2020, doi: 10.3233/JIFS-189081.
[36] X. Yu *et al.*, "Positioning, navigation, and book accessing/returning in an autonomous library robot using integrated binocular vision and QR code identification systems," *Sensors*, vol. 19, no. 4, p. 783, 2019, doi: 10.3390/s19040783.
[37] B. Debnath, A. Ghosh, and S. Deb, "Designing a rapid optical response trigger (RORT) for self-navigating and path explorer robot," *Procedia Computer Science*, vol. 167, pp. 1543–1552, 2020, doi: 10.1016/j.procs.2020.03.365.
[38] B. Zhang, S. Li, J. Qiu, G. You, and L. Qu, "Application and research on improved adaptive Monte Carlo localization algorithm for automatic guided vehicle fusion with QR code navigation," *Applied Sciences*, vol. 13, no. 21, p. 11913, 2023, doi: 10.3390/app132111913.
[39] V. M. Bondarko, "Vadim Glezer's modules model as a possible basis for vision perception description," *Journal of Optical Technology*, vol. 91, no. 8, pp. 570–576, 2025, doi: 10.1364/jot.91.000570.
[40] B. Bai *et al.*, "Microcomb-based integrated photonic processing unit," *Nature Communications*, vol. 14, no. 1, p. 66, 2023, doi: 10.1038/s41467-022-35506-9.
[41] G.-R. Duan, "Substability and substabilization: Control of subfully actuated systems," *IEEE Transactions on Cybernetics*, vol. 53, no. 11, pp. 7309–7322, 2023, doi: 10.1109/TCYB.2023.3242277.
[42] N. Deyhim, M. Adeola, and S. B. Bhakta, "Evaluation of barcode-assisted medication preparation technology for liquid medication doses," *American Journal of Health-System Pharmacy*, vol. 82, no. 9, pp. e447–e456, 2025, doi: 10.1093/ajhp/zxae339.
[43] R. Lopez and A. Rose, "Review and concepts on how construction and facilities management can utilize barcode and QR code technology to access documents onsite," *Procedia Computer Science*, vol. 253, pp. 1313–1322, 2025, doi: 10.1016/j.procs.2025.01.193.

BIOGRAPHIES OF AUTHORS



Rafid Mohammed Khaleefah is a lecturer and researcher at the University of Basrah, Iraq. She received the B.Sc. degree in computer engineering from Iraq University, Iraq, in 2014, and the M.Sc. degree in computer engineering from the University of Budapest, Hungary, in 2022, with specialization in internet architecture and services and smart cities. Her research interests include artificial intelligence, internet of things (IoT), smart cities, robotics, embedded systems, and cybersecurity. She is currently working at the Information and Communication Technology Center (ITCC), University of Basrah, and teaching at the Department of Mechatronics Engineering. She has published and presented research papers in AI, IoT, and intelligent systems. She can be contacted at email: rafeed.rmk@uobasrah.edu.iq.



Ali A. Abed is a professor at the University of Basrah, Department of Mechatronics Engineering in Iraq. He received the B.Sc. & M.Sc. degrees in electrical engineering in 1996 & 1998, respectively, and a Ph.D. in computer & control engineering in 2012. His field of interest is robotics, computer vision, and IoT. He is an IEEE senior member, IEEE member in Robotics & Automation Society, IEEE member in IoT Community, and member of ACM. He is currently supervising a group of researchers working on developing deep learning models for computer vision robotic applications. He can be contacted at email: ali.abed@uobasrah.edu.iq.



Mohammed Badr Algubili is a researcher and academic at the University of Basrah, Iraq. His research interests include artificial intelligence, internet of things (IoT), smart systems, networking, and information technology applications. He has contributed to several scientific publications and research activities in the fields of intelligent systems and modern computing technologies. He is currently affiliated with the Information and Communication Technology Center (ITCC), University of Basrah. He can be contacted through his academic profiles on Google Scholar, Scopus, ORCID, and ResearchGate, or at email: mohammed.badr@uobasrah.edu.iq.



Huda Mohammed Alsayednoor is affiliated with the Department of Information Technology at Al-Arab University College, Basra, Iraq. She is an emerging researcher in the field of intelligent systems and cybersecurity, with a particular focus on secure and efficient computing in internet of things (IoT) and fog computing environments. She is actively engaged in collaborative research with international teams, aiming to develop practical and scalable solutions for next-generation computing environments. She is committed to advancing her expertise in intelligent network systems and contributing to high-impact research in the areas of fog computing and secure distributed architectures. She can be contacted at email: huda1994noor@gmail.com.



Mahmood A. Al-Shareeda is a distinguished academic and researcher, currently serving as a lecturer at Southern Technical University in Basra, Iraq. He previously held a postdoctoral fellowship at the National Advanced IPv6 Centre (NAv6), Universiti Sains Malaysia (USM), in Penang, Malaysia. He completed his B.Sc., M.Sc., and Ph.D. degrees in Iraq, Lebanon, and Malaysia, respectively, reflecting a rich and diverse educational background across three nations. His outstanding contributions to research and academia have earned him global recognition, including being listed among the top 2% of scientists worldwide by Stanford University/Elsevier for the significant impact of his scholarly work. His expertise encompasses a wide range of research areas, including network monitoring, internet of things (IoT), vehicular ad-hoc network (VANET) security, and IPv6 security. He can be contacted at email: Mahmood.alshareedah@stu.edu.iq.