A Mobile Robot Design for Home Security Systems

Thanh-Nam Pham
Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam
ptnam@ictu.edu.vn

Duc-Tho Mai
Academy of Cryptography Techniques, Hanoi, Vietnam
ductho-mai@actvn.edu.vn (corresponding author)

Received: 24 March 2024 | Revised: 19 April 2024 | Accepted: 6 May 2024

 DOI: https://doi.org/10.48084/etasr.7336

ABSTRACT

Home Security Systems (HSSs) have received much attention and have been widely adopted for practical deployment. However, detection and warning accuracy still need to be improved, along with the range of the realm, making it challenging to satisfy the user demands. This study proposes a security monitoring system based on mobile robots and the Internet of Things (IoT), allowing users to monitor and control devices remotely. The mobile robot integrated sensor systems and surveillance cameras are utilized for unauthorized early intrusion detection and to give users instant warnings. The data collected by the robot were stored on the Firebase Cloud server, and a mobile application and a Telegram interface were integrated to manage and control the system. In addition, adaptive motion control was adopted to correct errors in the robot's trajectory. The implementation results proved that this system operated effectively with a minimal response delay of 0.87–1.67 s and a high detection accuracy (96.25%) in two experimental cases, which makes it suitable for real-time applications.

Keywords-mobile robot; IoT; Telegram bot; Cloud Firebase; remote security systems

I. INTRODUCTION

The increasing demand for automated monitoring of indoor environments and the protection against unwanted intrusions, have promoted the development of intelligent systems based on mobile robots and the Internet of Things (IoT) [1]. Robots significantly expand the potential of surveillance systems from their traditionally passive roles to active surveillance to interact with the environment, with humans, or with other robots for more complex actions [2]. Another significant challenge is controlling the entire viewing angle of surveillance cameras for tracking a vast area, which requires increased budget and effort. The above systems must also show flexibility in their warning process for users with reliable, real-time handling methods. Several worldwide scholars have attempted to implement the monitoring system using mobile robots and passive infrared (PIR) sensors connected via applications or Telegram. Authors in [3] developed an IoT-based home security notification system that employed the Raspberry Pi as a microcontroller to prevent break-ins, theft, and potential home intruders. The microcontroller was equipped with a PIR sensor to detect the presence of intruders and a Raspberry Pi camera to capture the image of an intruder and send the notification message to the users deploying the Telegram bot installed on mobile phones. The author announced that the range of PIR detection and response time was 500 mm and from 1.1 to 5.5 s, respectively. Authors in [4] depicted an autonomous mobile robotic system for the surveillance of indoor environments related to environmental mapping, localization, and autonomous navigation. The system's architecture was illustrated putting into service a three-layer scheme that allows for modularity and flexibility and may supervise several primary navigation and specific surveillance tasks. Then, the built-in modules for detecting abandoned or removed objects and tracking people were developed. The pre-liminary experimental results are promising and demonstrate the effectiveness of the overall system. Authors in [5] introduced a home security system with the capacity for human detection. Their approach included a Raspberry Pi 3, Arduino, PIR sensors, a web camera, and a buzzer. The Support Vector Machine (SVM) and Histogram of Oriented Gradient (HOG) algorithms were employed for the human detection task. The outcome revealed that the system could detect intruders with an accuracy of 89%, and the average time for detection was 1.1–2.01 s. Authors in [6] introduced a system that consists of InfraRed (IR) sensors to sense any motion in the front door, which triggers a camera to capture an image for face recognition and send it to the owner through the Telegram interface. The proposed model was implemented in a Raspberry Pi, which could be interfaced with a mobile device or PC to operate efficiently and cheaply (with a cost of about $30). In addition, the authors added a temperature sensor, which could be deployed for fire detection, and a gas sensor for sensing gas leakage. The system was built for secure remote access and control of home appliances. In [7], the authors described the construction of a home monitoring system using a Raspberry Pi that worked as the control center, connected with several PIR sensors, as well as with the Telegram interface to monitor the safety of the house from intruders and being able to notify about the situation of the house. They experimented with early theft warning, gas leaks, or a fire with a high correction rate in real-time conditions. Several other studies have also focused on improving high intrusion detection accuracy utilizing AI [8], adding GSM technology to send warning SMS [9], and optimizing and combining distinct algorithms for a robot path [10-12]. However, the systems discussed have the weakness of slow response time. Also, camera-based surveillance systems cannot detect objects well when they move fast. Moreover, noise is a drawback of surveillance systems based on IR and PIR sensors. It causes false alarms due to animals moving into the monitoring area. In addition, scheduling moves according to the robot's trajectory has many errors, and the moving robot deviates from the desired trajectory. The cause of the trajectory deviation is floor roughness, uneven movement between the robot's wheel, or even obstacles. To overcome the above weaknesses, this paper proposes a security monitoring system based on a self-propelled robot around the monitoring area. The system can detect objects with high accuracy of 96.25% and send anomalous intrusions information with a low latency (0.87–1.67 s). Abnormal activity data, including images and live video streams, can be transmitted to a mobile application for proper handling. Moreover, this study also adopted an additional procedure for alerting, handling, and controlling the system through the Telegram application to enhance system flexibility and security.

II. SYSTEM DESIGN

This section describes the structure and components of the proposed system. Mathematical models and algorithms to improve system performance are provided and analyzed.

A. System Architecture

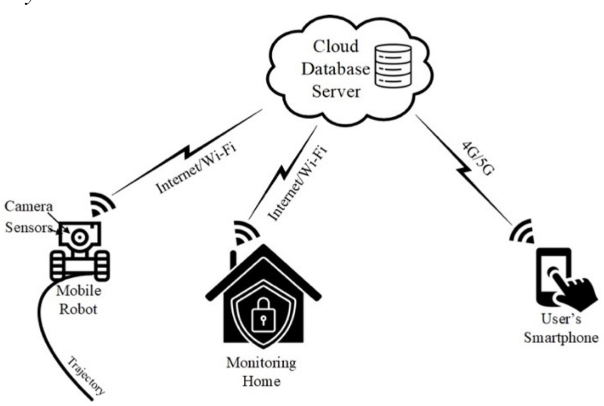

Fig. 1. System architecture.

The overall architecture of the proposed system is described in Figure 1. The monitored home and its electrical appliances are controlled through the user's smartphone. All systems communicate with each other by conducting and receiving data through a cloud server (Firebase real-time database). User's home consists of indoor electrical appliances (doors, lights) controlled by the ESP8266 Wi-Fi transceiver circuit, which gets and sends data from the Firebase database. The mobile robot system will use two Wi-Fi transceiver circuits, the NodeMCU Esp32-cam circuit, and the NodeMCU Esp8266 circuit, to send and receive data from the Firebase database. The smartphone is equipped with system management, monitoring, and control application (app) or a Telegram to get system data information. The app will connect via Wi-Fi or 4G/5G to send and receive data from the database during Firebase execution.

B. Robot Design

The diagram in Figure 2 demonstrates the connections between the robot's hardware components. To execute remote monitoring and security warning functions, firstly, the power supply block provides a power supply for all components in the robot circuit and generates suitable voltage and current for all components. The Arduino Uno will then process the data gathered from the sensors, which included a PIR sensor, an ultrasonic sensor, and a camera integrated into the ESP32. After the processing data step, this block sends the data to the system database and controls the motor block, including the L298N DC motor control circuit and the DC motor, to help the robot move in the correct orbit. In addition, the system sets a NodeMCU Esp32 to take data from the Firebase and controls the robot to play warnings, as well as servo operations to support optimizing the scanning angle of the motion body temperature sensor and camera vision.

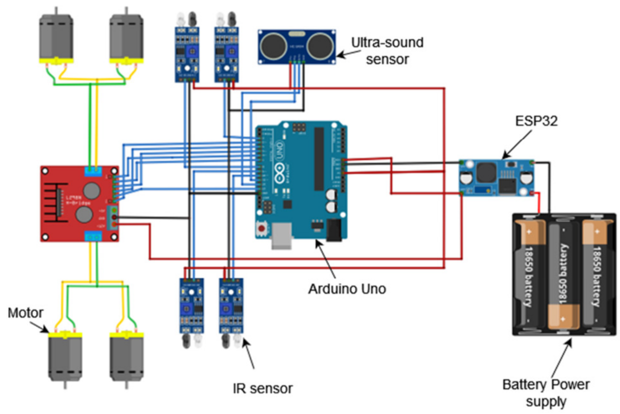

Fig. 2. System principle diagram of the proposed robot.

C. System Process

The algorithm flowchart of the system is depicted in Figure 3. The system works in the following order: Initially, the security monitoring robot is programmed to move around the area of the house in automatic mode. However, the administrator can install and control the robot directly through a smartphone device. During the movement, if the surveillance robot detects people or moving objects through the sensors and cameras assigned to it, the indoor control system, such as lights and alarms, will turn on immediately. At the same time, an alert will be sent to the user through the application software interface running on the smartphone. Also, during intrusion detection, the robot will capture live photos and send them to the system database; these images will be displayed to users when checking intrusion alerts on their mobile devices. After about 1 min, if the robot does not detect any stranger objects moving around that area, the system will return to a safe state, which means controlling the devices in the house to return to their previous regular operation. The robot will take a signal from a pre-programmed PIR sensor to follow a predetermined path. Image data from the camera assigned to the robot are always transmitted to the app running on the user's smartphone. Users can observe the monitoring space through this real-time data stream.

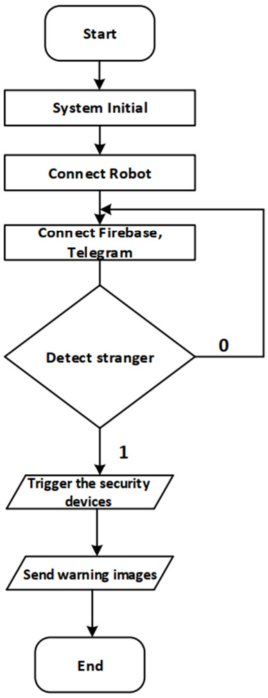

Fig. 3. The algorithm flowchart of the proposed system.

D. Trajectory Error Model

During the movement, the robot's trajectory may appear errors and not follow the pre-established path. This faulty travel trajectory is depicted in Figure 4. Floor roughness, obstacles, or uneven movement between the robot wheels can cause motion errors, leading to failure of correctly completing the security monitoring task of the robot around the monitoring area. In Figure 4, e_θ is the robot's movement direction error, defined by the difference between the robot's current and desired trajectory movement direction. e_d is the difference in the distance between the robot's current trajectory and the desired trajectory position, e_t is the difference in distance between the robot's current position and its desired position in its trajectory. e_θ is the most crucial factor affecting the system's accuracy, leading to changes of e_d and e_t. A control model for modeling the factors mentioned above follows.

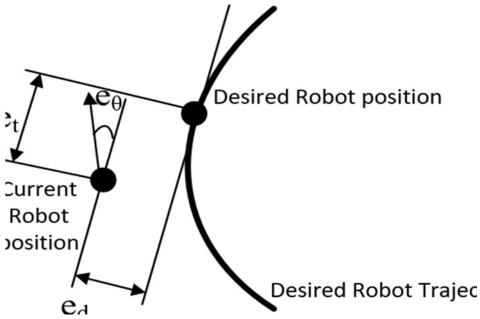

Fig. 4. Error illustration in the moving trajectory of the robot.

It is assumed that at time t, the position of the mobile robot is (x_t, y_t) and its orientation is θ_t, given by:

$$x_t = x_{t-1} + \Delta u_t \cos\big((\theta_t + \theta_{t-1})/2\big) + \beta_x \Delta u_t \quad (1)$$

$$y_t = y_{t-1} + \Delta u_t \sin\big((\theta_t + \theta_{t-1})/2\big) + \beta_y \Delta u_t \quad (2)$$

$$\theta_t = \theta_{t-1} + \Delta\theta_t + \beta_\theta \Delta u_t \quad (3)$$

$$\Delta u_t = (\Delta n_t^L + \Delta n_t^R)/2 \quad (4)$$

$$\Delta\theta_t = (\Delta n_t^R - \Delta n_t^L)/b_w \quad (5)$$

where Δn_t^L and Δn_t^R are the measured motions of the left and right robot wheels after each sampling interval t. Δu_t is the distance traveled by the robot after each sampling interval t. $\Delta\theta_t$ is the deviation of the robot's path after each sampling interval t. β_x , β_y and β_θ are coefficients. This model can represent important error sources and will be utilized in the adaptive motion control. From (3), the orientation error is a linear function of the distance traveled by the mobile robot.

E. Position-based Path Planning

In the proposed system, the robot knows its position after each interval. It moves considering coordinate markers deploying RFID technology. Position-based path planning is adopted to let the robot move precisely along a predefined sequence of waypoints. Trajectories with different curves can be planned, optimizing criteria such as overall time or energy usage. In the planning step, assume the current position of the robot at time t is (x_t, y_t, θ_t) and the next waypoint to move is at (W_x, W_y). It must first rotate to point toward the reference point. The vector direction it should point to is:

$$\begin{pmatrix} d_{x_t} \\ d_{y_t} \end{pmatrix} = \begin{pmatrix} W_{x_t} - x_t \\ W_{y_t} - y_t \end{pmatrix} \quad (6)$$

The orientation angle α_t that the robot must drive in is given by:

$$\alpha_t = tan^{-1} \frac{d_{y_t}}{d_{x_t}} \quad (7)$$

where α_t has a value in the range $0 < \alpha_t \leq \pi$. Therefore, the angle that the robot must rotate is $\beta = \alpha - \theta$. If the robot moves as efficiently as possible, care should be taken to shift this angle by adding or subtracting 2π to ensure that $-\pi < \beta \leq \pi$. Then the robot should drive forward in a straight line through distance $d = \sqrt{d_{x_t}^2 + d_{y_t}^2}$.

One of the important issues for mobile robot systems is avoiding obstacles on a predetermined moving path. Normally, this travel route is in a defined area. If nothing is unusual, the robot will perform the task smoothly and accurately. However, in some situations, obstacles suddenly appear on the robot's travel path, which affect the task and performance of the system. Several solutions have been suggested to address this problem, such as free segment and turning point [13], collision avoidance based on passive agents [14], dynamic window approach [13, 14], and AI algorithms [15, 16]. This study adopted a dynamic obstacle avoidance solution based on dynamic moving speed adjustment and optimal turning point control in case of unexpected obstacles appearing on the path or the in case of trajectory errors, as discussed above.

1) Adaptive Obstacle Avoidance Algorithm based on the Awareness of the Surrounding Environmental Context

Suppose that at time t_i, the robot moves to position s_i on the predetermined trajectory and detects an obstacle in its path based on the distance measured from the ultrasonic sensor that is equipped with. The robot's next destination position on the desired moving trajectory is point D_j, at time t_j. During monitoring, the robot will always move along a predetermined path. However, when the robot detects an obstacle in the direction of movement at a distance:

$$d_t \geq d_r + \delta \tag{8}$$

where d_r is the distance obtained from the ultrasonic sensor mounted on the robot and δ is an error margin of distance, then the robot needs a method to avoid this obstacle and continue its task according to the desired trajectory. Three ultrasonic sensors have been installed to detect obstacles in front of the robot. Each sensor covers a 60^0 angle, so half of the plane in front of the robot could be covered in the direction of movement. The effective object detection range of these sensors is 4–160 cm. These three sensors will yield the smallest collision warning threshold distance value. The value of δ parameter depends on the sensor error, and usually it is about 3-5% of the measured distance.

A solution was proposed for the robot to temporarily move along another path to avoid obstacles with the shortest travel distance (minS_{ij}) or minimum travel time to point D_j (min$\Delta t = t_j - t_i$). In Figure 5, the distance is S_{ij}=$S_{ij} = \sum S_{iA} + S_{Aj}$ in the case of left rotation or S_{ij}=$S_{ij} = \sum S_{iB} + S_{Bj}$ in the case of right rotation. During the movement, the robot will continuously monitor obstacles based on the distance d_t between the robot and the obstacle, as in (8). The robot will turn left or right to ensure the smallest distance S_{ij}, based on the robot knowing in advance its moving trajectory at the next point. The system will calculate and make a turning control decision for the robot. The recommended algorithm assumes that the robot will return to a predetermined desired trajectory when it moves around obstacles. The algorithm is optimal when the distance and time the robot needs to use to return to the predetermined path are minimal. This method achieves high accuracy in cases where there are many obstacles in the monitoring environment, as the robot will move to follow the first obstacle encountered on the predetermined path.

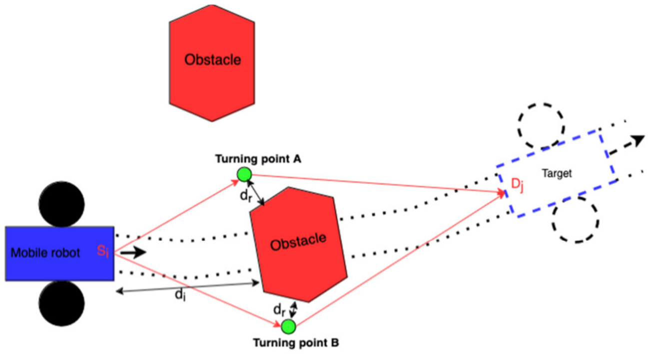

Fig. 5. The proposed obstacle avoidance algorithm.

2) Sensor Data Fusion to Enhance the Intruder Detection Capacity

A sensor data fusion method is adopted to increase the effectiveness of detecting intruders in the monitored area. PIR sensors were deployed to spot moving objects, like humans or animals, using the infrared radiation they emit. The effective operating range of the PIR sensor is less than 6 m. The monitoring range is divided into two areas: (1: common detection area and 2: emergency detection area). In area 1, the PIR sensor works effectively, detecting intrusive objects at about 6 m and combining with the surveillance camera mounted on the robot to send images to the user's management application. In area 2, the PIR sensor is combined with the ultrasonic sensor to detect intruding objects at a close range (about 1.4 m). These objects often appear suddenly or move at speed and approaching the warning area. This improves the system's accuracy, enabling it to provide alerts covering incursion instances. The subsequent section provides a detailed account of the experimental assessment of the system's accuracy.

III. EXPERIMENTAL RESULTS AND DISCUSSION

A. Hardware and Software Implementation

A security surveillance robot prototype is portrayed in Figure 6(a). PIR sensors and cameras are assigned on the front of the robot. The main interface of the system management software is installed on the user's smartphone and is illustrated in Figure 6(b). The software interface on the smartphone has functions, such as displaying the remaining battery life of the robot, capturing photos, viewing live surveillance cameras, playing alarms, controlling the robot's speed, and activating the robot. In addition, it enables monitoring the operating status of appliances, turning off the intruder detection and warning states, shutting down the camera scanning angle, activating automatic patrolling by utilizing the robot without following the predefined path, and also stopping the robot. A function screenshot is displayed in Figure 6(c). To improve security, the management and control of the system were enforced based on the Telegram bot. Figure 7(a) describes the execution of the management and control software using Telegram. Users control the monitoring features through messages sent into the chat group. Figure 7(b) shows the database system execution based on the Cloud Firebase Server.

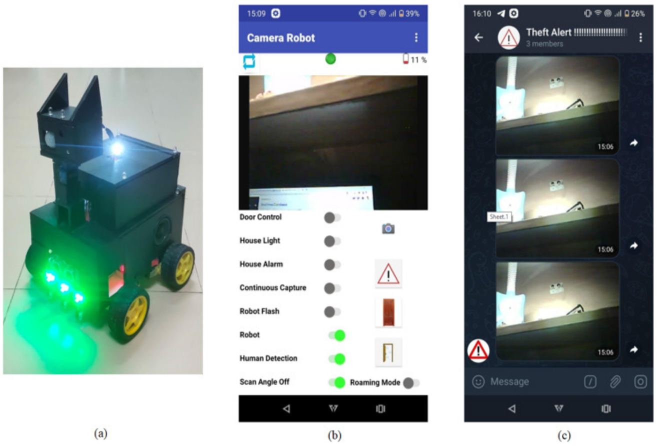

Fig. 6. Hardware and software implementation: (a) Security surveillance robot prototype, (b) mobile app interface, (c) theft alert warning.

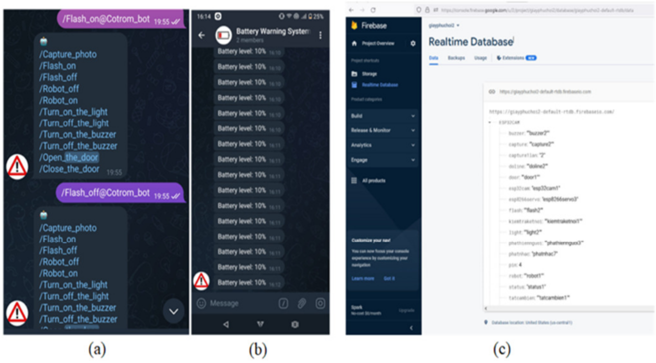

Fig. 7. Implementation of system management application via (a) Telegram bot, (b) Cloud Firebase Server.

B. Result Analysis

1) Evaluation of the Sensor Errors in the Mobile Robot Security System

Sensor errors are an important factor affecting the accuracy and performance of mobile security surveillance robot systems. As described above, the introduced system deployed a data fusion method combining ultrasonic and infrared sensors to detect obstacles and intruding objects. Each collected data point was measured 20 times in the experiment. The sensor error was also limited to 3 - 5%, which is acceptable for ultrasonic and passive infrared sensors with distances from 4 cm to 160 cm. Table I presents the experimental results of sensor data fusion.

TABLE I. DISTANCE ERROR OF THE SENSOR DATA FUSION METHOD

Detection distance (cm)	4	10	20	30	50	70	100	140
Mesured dis. (avg. 20 times)	4.18	10.41	20.71	29.09	51.1	68.94	103	133.02
Distance error (cm)	0.11	0.41	0.71	-0.91	1.09	-1.06	3.01	-6.98
Distance error (%)	4.32	3.96	3.41	3.13	2.14	1.53	2.92	5.24

After testing the system, the response time in detecting people or objects with the body temperature sensor ranges from 0.87 to 1.67 s. The response time between the robot and the home electrical system with two control methods (mobile app and Telegram) is summarized in Table II. Based on the test results' analysis, it can be seen that the system achieves higher responsiveness and accuracy by being controlled with a smartphone app when pushing and receiving data in a real-time database. In addition to having a different response delay between the two control methods, each method has particular strengths.

- (a) Mobile Application: an easy-to-see interface that combines all the robot features and home electrical equipment systems. The response time is reliable at 0.87 s.
- (b) Telegram does not excel in monitoring and controlling robots and electrical equipment systems, and does not have an eye-catching interface for users and a stable response. Telegram is an option due to its popularity and security.

TABLE II. RESPONSE TIME EVALUATION OF THE TWO CONTROL METHODS

Number of tests		Control methods	
		Mobile app	Telegram
1	Response time (s)	0.5	1.06
2		1.81	3.38
3		0.76	1.67
4		0.5	1.51
5		1.12	1.27
6		1.17	1.12
7		1.09	1.23
8		0.39	1.14
9		0.61	1.98
10		0.82	2.34
	Average (s)	0.87	1.67

Table III compares the experimental results run on PIR sensors and the mobile robot regarding the methodology, expected outputs, cost, detection accuracy, and average response time. As it can be observed from the Table, it was concluded that, in the present study, the design had the highest detection rate of 96.25% and lower response time than that of the other studies. Authors in [6] reported the most insufficient detection accuracy (75%) because they utilized IR sensors, which are significantly impacted by the noise. The increased detection rate efficiency values from [4, 5, 7] were 82%, 87%, and 89%, respectively, proving that these authors adopted additional motion sensors, image processing cameras, mobile robots, and especially applied machine learning algorithms (SVM, HOG). The current work employed multiple sensors (ultrasonic and PIR sensors) combined with a mobile robot to obtain better performance. Some recent technologies such as SIM900 GSM module and advanced algorithms will be integrated into the next study.

The primary goal is to ensure continuity in security monitoring, so the high speed of robot movement dramatically affects overall performance because it cannot detect intruders in time. Figure 6 exhibits the abnormal detection accuracy when the robot moves at different velocities. As speed increases, the accuracy of detecting intruding objects decreases. With the robot's moving speed of 0.278 m/s (~ 10 km/h), the accuracy is very high at 96.25%, whereas at a speed of 0.833 m/s (~30 km/h), the accuracy in object detection is still high at 80.3%.

TABLE III. COMPARISON BETWEEN SURVEILLANCE MOBILE ROBOT SYSTEMS

Ref	[4]	[5]	[6]	[7]	This work
Methodology	RFID Device, Mobile Robot, Camera	Raspberry Pi3, PIR Sensor, ML Algorithm	Telegram, IR Sensor	PIR sensor, Raspberry Pi, Telegram	Ultrasonic sensors, Telegram, PIR sensor, Mobile Robot Camera
Objective	Object detection	Object detection	Object recognition	Temperature, object movement	Object detection
Cost ($)	~ 50	NA	30	NA	~ 50
Detection accuracy (%)	82%	89%	75%	87%	96.25%
Average response time (s)	2	1.1 - 2.01	1.8	1.5 - 3	0.87 - 1.67

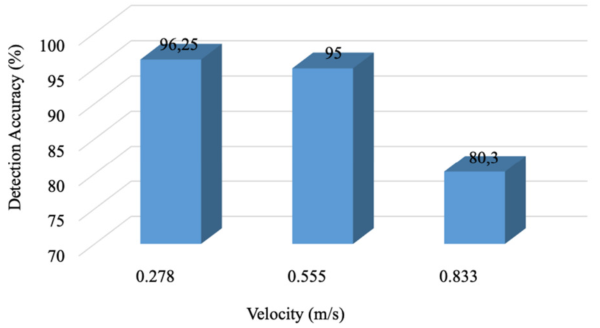

Fig. 8. Accuracy comparison of the system whith changing robot speed.

IV. CONCLUSION

This paper presents an automatic robot system for security monitoring and remote warning tasks with the highest detection accuracy (96.25%) and a shorter average response time (0.87 to 1.67 seconds) than other related studies. These parameters are adequate for real-time surveillance applications like smart homes and intrusion detection systems. This study's design is currently being performed to collect data from the sensors, including PIR and ultrasonic, then process and send it to the user's smartphone through mobile apps or Telegram, along with warnings and statistics. The system should be soon improved by integrating the SIM900A Module to send an SMS warning when the power is off, selecting and optimizing the mobile robot's path-finding algorithms, and depicting more intrusion scenarios to meet user security demands.

REFERENCES

[1] A. Hanah, R. Farook, S. J. Elias, M. R. A. Rejab, M. F. M. Fadzil, and Z. Husin, "IoT Room Control And Monitoring System Using Rasberry Pi," in *2019 4th International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)*, Nov. 2019, pp. 1–4, https://doi.org/10.1109/ICRAIE47735.2019.9037759.

[2] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, San Francisco, CA, USA, 2000, pp. 476–481, https://doi.org/10.1109/ROBOT.2000.844100.

[3] M. N. Osman, M. H. F. Ismail, K. A. Sedek, N. A. Othman, and M. Maghribi, "A Low-Cost Home Security Notification System Using IoT and Telegram Bot: A Design and Implementation," *Journal of Computing Research and Innovation*, vol. 7, no. 2, pp. 327–337, Sep. 2022, https://doi.org/10.24191/jcrinn.v7i2.325.

[4] D. Di Paola, A. Milella, G. Cicirelli, and A. Distante, "An Autonomous Mobile Robotic System for Surveillance of Indoor Environments," *International Journal of Advanced Robotic Systems*, vol. 7, no. 1, Mar. 2010, Art. no. 8, https://doi.org/10.5772/7254.

[5] N. Surantha and W. R. Wicaksono, "Design of Smart Home Security System using Object Recognition and PIR Sensor," *Procedia Computer Science*, vol. 135, pp. 465–472, Jan. 2018, https://doi.org/10.1016/j.procs.2018.08.198.

[6] N. Hema and J. Yadav, "Secure Home Entry Using Raspberry Pi with Notification via Telegram," in *2020 6th International Conference on Signal Processing and Communication (ICSC)*, Noida, India, Mar. 2020, pp. 211–215, https://doi.org/10.1109/ICSC48311.2020.9182778.

[7] I. G. M. N. Desnanjaya and I. N. A. Arsana, "Home security monitoring system with IoT-based Raspberry Pi," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 3, pp. 1295–1302, Jun. 2021, https://doi.org/10.11591/ijeecs.v22.i3.pp1295-1302.

[8] N. Made, N. Piarsa, and A. Sasmita, "Telegram Bot Integration with Face Recognition as Smart Home Features," *International Journal of Computer Applications*, vol. 182, no. 13, pp. 42–47, Sep. 2018, https://doi.org/10.5120/ijca2018917778.

[9] R. Hasan, M. M. Khan, A. Ashek, and I. J. Rumpa, "Microcontroller Based Home Security System with GSM Technology," *Open Journal of Safety Science and Technology*, vol. 5, no. 2, pp. 55–62, May 2015, https://doi.org/10.4236/ojsst.2015.52007.

[10] L. Zhang, Y. Zhang, and Y. Li, "Mobile Robot Path Planning Based on Improved Localized Particle Swarm Optimization," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6962–6972, Mar. 2021, https://doi.org/10.1109/JSEN.2020.3039275.

[11] F. H. Ajeil, I. K. Ibraheem, M. A. Sahib, and A. J. Humaidi, "Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm," *Applied Soft Computing*, vol. 89, Apr. 2020, Art. no. 106076, https://doi.org/10.1016/j.asoc.2020.106076.

[12] X. Deng, R. Li, L. Zhao, K. Wang, and X. Gui, "Multi-obstacle path planning and optimization for mobile robot," *Expert Systems with Applications*, vol. 183, Nov. 2021, Art. no. 115445, https://doi.org/10.1016/j.eswa.2021.115445.

[13] J. Hong and K. Park, "A new mobile robot navigation using a turning point searching algorithm with the consideration of obstacle avoidance," *The International Journal of Advanced Manufacturing Technology*, vol. 52, no. 5, pp. 763–775, Feb. 2011, https://doi.org/10.1007/s00170-010-2749-5.

[14] K. M. Zuhaib, J. Iqbal, A. M. Bughio, S. A. A. S. Bukhari, and K. Kanwar, "Collision Avoidance of a Kinodynamically Constrained System from Passive Agents," *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6760–6765, Feb. 2021, https://doi.org/10.48084/etasr.4022.

[15] F.-F. Li, Y. Du, and K.-J. Jia, "Path planning and smoothing of mobile robot based on improved artificial fish swarm algorithm," *Scientific Reports*, vol. 12, no. 1, Jan. 2022, Art. no. 659, https://doi.org/10.1038/s41598-021-04506-y.

[16] L. Yang *et al.*, "Path Planning Technique for Mobile Robots: A Review," *Machines*, vol. 11, no. 10, Oct. 2023, Art. no. 980, https://doi.org/10.3390/machines11100980.