Development and Hardware Implementation of IoT-Based Patrol Robot for Remote Gas Leak Inspection

Original Scientific Paper

Tasneem Yousif
Computer Engineering Department Laboratories
Department of Computer Engineering, University of Bahrain
Skhair, Kingdom of Bahrain
tasneemyousif@hotmail.com

Wael El-Medany
Computer Engineering Department Laboratories
Department of Computer Engineering, University of Bahrain
Skhair, Kingdom of Bahrain
welmedany@uob.edu.bh

Abstract – *The Internet of Things Robot (IoTR) is an emerging paradigm that brings together robotic systems with the Internet of Things (IoT) that connect sensors and smart objects pervasively embedded in everyday environments. With the recent developments in robotic system applications, it becomes apparent that the mobile robot has great importance in real-world applications such as navigation and surveillance. One of the most important applications of a mobile robot is patrolling and gas leak detection. This paper proposes a real-time IoT Robot (IoTR) that can be used indoors or outdoors for gas leak detection purposes. The proposed mobile robot is equipped with microphones, speakers, the hub of smart sensors that are necessary for patrolling and gas leak detection, a high-resolution IP video camera for live video streaming, Bluetooth for indoor applications and tracking, and GPS/GPRS for outdoor applications and tracking. The experimental testing of the preliminary prototype confirms the design objectives. The robot has been tested for indoor and outdoor modes; the robot can detect gas leakage and provides a live video streaming of the surrounding area, which can be tracked on Google maps. At the same time, the robot can be controlled remotely through a mobile app or website, the robot can move autonomously and avoid obstacles. The proposed work provides a low-cost IoT robot through the use of the available and cheap components and sensors, which featured a high quality at the same time. Our proposed system exhibits promising gas sensing performance in harsh environments, using intelligent gas sensors that have a fast response (>10s), low cost, high sensitivity, long life, robustness, and physical size.*

Keywords*: Gas Leak Detection, IoT Robot, Remote Monitoring, Patrol Robot, Tracking System, Real-Time*

1. INTRODUCTION

With the development of society, economy and rapid developments of autonomous mobile robots that have been developed to tackle the challenges of the petroleum industry, increasing the protection measurements, enhancing the quality of critical patrol and gas areas and decreasing costs are now possible. The use of security patrol robots has become very important recently because it helps to prevent human lives from danger and reduce human errors. The patrol robot can be used indoors or outdoors depending on the applications, with automatic obstacle avoidance as a common feature [1, 2]. Oil and petroleum companies need a safe environment for their critical work. The industry of oil and gas is now positively looking for advanced robotic solutions in conjunction with the growth of global demand and depleting resources for fossil fuels. These smart robotic systems are used to increase their productivity and safety [3]. Health and safety awareness is so essential for all workers and technicians who are working in those companies [4, 5].

Explosive and toxic gases are surrounding us everywhere such as gas stations, power plants, landfill sites, hotels, kitchens, wells, oil, and gas companies [3]. They have different sources such as welding, swamp wells, volcanos, grinding, mining, petroleum areas, etc. Explosive materials, toxic wastes, and hazardous gases endanger our lives. They cause chronic and dangerous diseases for humans such as pneumonia and angina pectoris. Also, it may cause instant death for humans. Moreover, accidents that happen due to these hazardous gases cause property damage, substantial money, injuries, grieving families, and many other fatalities [3, 6, 7]. These gases are very hazardous for companies in the oil and gas industry due to their critical job. Therefore, oil and gas companies need to comply with strict gas safety regulations to ensure no gas leakage occurs. The employees and the people working in these companies deserve to work in a safe and healthy environment. Our solution is a creative cost-effective and scalable solution for health and safety environments to employees and employers on-site, potentially from remote sites.

The presented work in this paper is extended research of the proposed remote monitoring system that has been introduced by [8, 9]. The research aims to develop real-time [10] remote monitoring and control IoT robot that is used for security and detecting explosive gases at the same time. The robot is equipped with a video camera and smart sensors for the smart detection tasks carried out by the robot. The proposed system consists of different subsystems, the controlling unit where it is either indoors using Bluetooth connection or outdoor controlling using the internet connection, the monitoring unit with a live streaming video camera, and the detection unit that consists of smart sensors and is used for detecting explosive gases. The IoT robot will work as security surveillance in real-time and can be tracked and controlled remotely through the Global Positioning System (GPS) and General Packet Radio Service (GPRS) network [11]. The robot will be able to change its route once the stationary alarm sensors are triggered. This feature will help the pipeline inspectors and the patrolling employees to maintain a safe distance in the remote areas of the site in case of any hazardous gas leakage. Additionally, the robotic system gives instant data analysis of the surrounding gases for the users using data visualization dashboards.

The paper is organized as follows: Section two addresses the state-of-the-art, where related work is described. Section three outlines the existing problem and the corresponding solution provided by the IoT robot, by describing the full system architecture and the main subsystems. Section four outlines the software user interface. In section five, the testing and validation results are discussed. Finally, section six provides conclusions and future research.

2. RELATED WORK

Gas leakage detection in industrial facilities is critical for ensuring the safety of human life, stationary gas sensors can be used to discover the gas leaks. In the following section, a review of the state-of-the-art mobile robot applications in security and gas leakage will be covered. A mobile robotic system for remote leak sensing and localization in large industrial environments has been presented by [12] to develop the RoboGas Inspector shown in Fig. 1. The gas leak detection technology for LNG facilities in Australia is shown in Fig. 2.

In [13], the authors introduced a novel smart security robot that used to work in dark non-vision environments, since most of the methods available are not that capable to work in a dark environment. The proposed work uses fuzzy logic and neural network methods to determine the situations that are not normal in the environment area, and they used a different level of danger alarm. The research shows a general view of the proposed security system as shown in Fig. 3. The system starts with the initialization step (all devices will be turned on), then the path planning operation, and then a motion control operation using four different fuzzy logic modules to monitor the area. During this stage, if any abnormal thing happens, the system will be switched to the alarm system to alarm the user.

Fig. 1. The RoboGas Inspector [12]

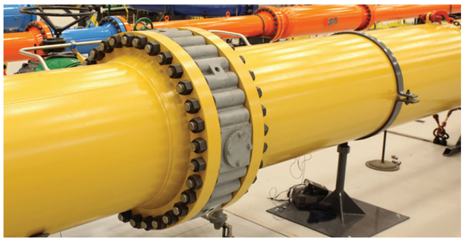


Fig. 2. Gas-Leak Detection Technology for LNG Facilities in Australia [12]

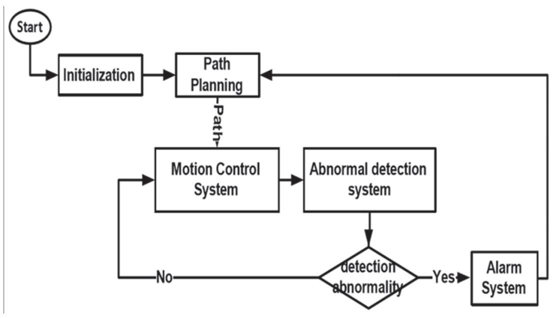


Fig. 3. General view Proposed Security System [13]

The authors in [14] proposed an android controlled surveillance spy robot with a wireless night vision camera. They aim to save human lives in the military sector where this robot is used in wars to monitor the area where it is dangerous for the army. Their robot consists of a night vision camera fixed to it to monitor the area while saving human lives. This robot is an Arduino-based controlled robot that used a developed android application for controlling it. However, their system does not have a gas detection system.

A fire robot based on a quad-copter has been developed in [15]. This robot is required to detect fires. They used an algorithm for processing the video signal to detect the fire and for the robot realization. Also, they used a software program using the SDK software developer kit to control the robot's motion.

The research introduced by [16] aims to develop a surveillance system that uses an unmanned aerial vehicle (UAV). The UAV robot has mobility capability. The authors proposed a surveillance robot for indoor monitoring. They combined six components of the system, such as unmanned aerial vehicles, vision-based pose estimation, vision-based state estimation, patrol path planning, UAV controller, joystick controller, and priority assignment for different inputs of the robot[16]. These components are used for autonomous patrol and surveillance systems. They developed and tested their system in the indoor environment. Fig. 4 shows the system architecture of the system.

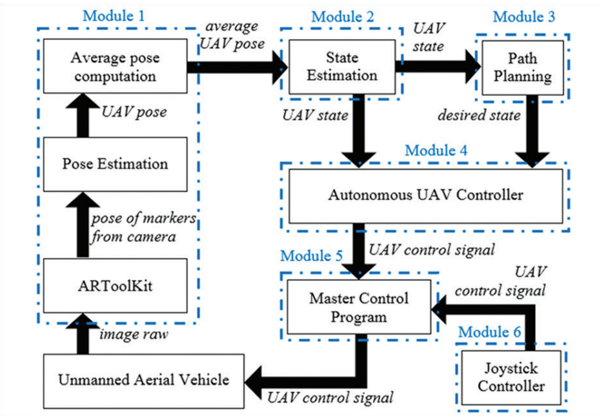


Fig. 4. System Architecture of the autonomous patrol and surveillance system using UAV [16]

In [17], the system is integrated with a mobile robot that can detect carbon monoxide (CO) with the temperature. The authors stated that their system tackled the challenges of the current CO gas sensors which are the limited detection ranges and sensitivity due to environmental factors such as temperature and humidity. This system lacks flexibility due to the usage of one gas sensor compared to our system. Besides, it does not have outdoor applications which are fundamental to industrial and landfill applications.

The authors in [18] introduced a mobile robot with a gas leakage detector for safety purposes. The system has an array of 16-metal oxide sensors (e-nose) that can detect the concertation of the gases by obtaining 16 voltage measurements. The system focused on detecting gas leakage in closed rooms due to the few amounts of airflow surrounding the rooms.

The authors proposed natural gas detection by using an intelligent gas searching method achieved by a swarm optimization algorithm [19]. Their system has a detection strategy for gas detection, gas tracking, and gas source localization with search time consumed between 27s-92s. The drawback of the discussed research is that the detection system has not yet been established with a real experimental robot based on relevant hardware and software equipment. Furthermore, their system needs more optimization for the user searching algorithm and a real simulation platform for the gas sensors analysis.

The system proposed in [20], provided a fireNose that contains three Metal-Oxide (MOX) sensors (SnO_2, WO_3, and NiO). They used online unsupervised gas discrimination algorithms that increased the sensitivity for the gas sensors.

2.1 COMPARISON TO STATE-OF-THE-ART RELATED WORK

The proposed design has been compared to state-of-the-art related work; the comparison has been carried out with other systems that are similar to the proposed system. The gas inspector system that has been presented by [21] and shown in Fig. 1 uses multiple measurement principles for remote sensing. This solution is used for the detection of hazardous gases by using novel leak-detection technologies. However, this system is portable which will make human life endanger. The explosive detector introduced by [22], is using Fido® XT. The Fido solution uses polymer-based technology to achieve faster detection results. However, this solution is a handheld detector. The IoTR has more advantages than stationary sensors network (SN) and portable sensors. The IoTR has the flexibility and feasibility to combine a variety of gas sensors depending on the application. Besides, the IoTR has the feature to explore the environment and detect gas leakages remotely. In contrast, handheld sensors have to be carried by the users to the required areas to detect the hazard gases, thus this jeopardizes human lives. The IoTR will be equipped with intelligent sensors to provide accurate computational gas results, gas sources localization, live tracking system, real-time visual monitoring, map-based approaches, alarming system, and remote-controlling. On the other hand, the portable sensors and stationary sensors have a limitation of inability to sense accurately when there is a variance in optimal sensor positions across long distances. Our system is a real-time system with a tracking system, live video streaming, and a web-based or mobile application to access the sensors reading remotely with intelligent dashboard visualizations. Furthermore, our mobile robot is operated remotely from anywhere on the planet, therefore, our system emphasizes the outdoor controlling. On the other hand, other solutions use limited distance controlling. Additionally, the gas sensors that have been used in the proposed system have advantages of low cost, fast time response, which is less than 10s, stability, physical size, and long life. Moreover, our robotic system has a reasonable cost compared to other solutions in the market. Our IoTR cost is ≈1000$.

3. IoTR ARCHITECTURE

The proposed IoT robot shown in Fig. 5 handles different kinds of tasks depending on the usage and the application. The proposed mobile robot has the flexibility to be equipped with a specific sensor based on the application. The robot can be remotely controlled with the mobile application or through the website to go through narrow or tough locations to discover the gas leakages in its surrounding area. It is equipped with a tracking device and a live video streaming camera for real-time monitoring. Additionally, the robotic system provides instant notifications through text messages on cell phones or text emails. Also, the system provides a voice alarming once the sensors trigger high concertation of the hazard gases. Besides, the robotic system provides gas sensing analysis reports on a daily, monthly, and yearly basis using visualization dashboards [23, 24]. The robotic system architecture for relaying data between the robot and the monitoring server is illustrated in Fig. 6. The patrol robot system depends on the IoT and cloud computing features. The system has database storage on the cloud server which will store all the data required for the system. These data contain the gas sensing results, the temperature measurements, the humidity measurements, the GPS coordinates data, the response time, the latest controlling request that is required for remote controlling, and the live streaming video. This full safety system sends and receives requests through GPRS/GSM communications. Fig. 7 shows the main hardware components required for real-time monitoring, tracking, and controlling, including Arduino Microcontroller (Fig. 7 (a)), Bluetooth shield (Fig. 7 (b)), GPRS shield (Fig. 7 (c)) and GPS shield (Fig. 7 (d)), all interfaced with the IoT robot. Fig. 8, shows the State Machine Chart (SMC) for the microcontroller Interfaced with the complete System. SMC illustrates the two different ways of accessing the system which are accessing the mobile application and websites. The mobile application has the function of indoor controlling using Bluetooth and autonomous controlling using a ready path follower. The website has three main functions: outdoor controlling, live tracking, and gas sensing results. All of these functions were successfully achieved through the communication between the GPRS and the web-server. Besides, there is a function of live video streaming using an IP camera to be accessed through the website or mobile application.

The patrol robot consists of four main subsystems:

1. Live video streaming using IP camera
2. Real-time remote controlling system whether indoor or outdoor.
3. Real-time tracking using GPS/GPRS and Google Map technologies.
4. Gas leakage inspection system using intelligent gas sensors.

The main four subsystems are discussed in the following subsections.

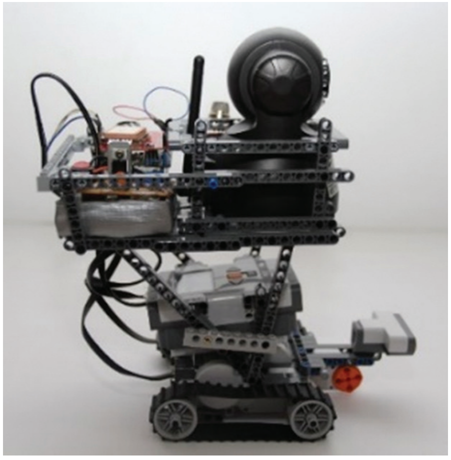


Fig. 5. IoT Robot Prototype

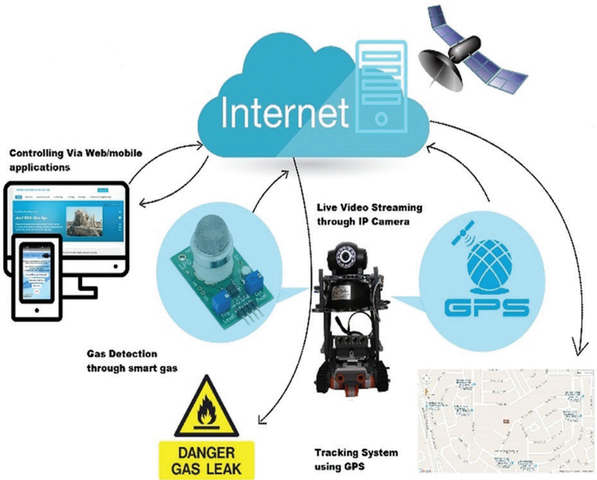
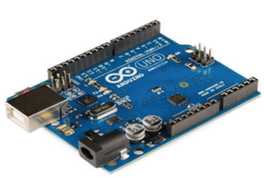


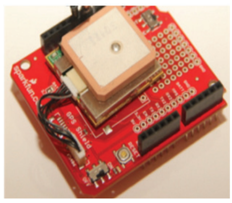
Fig. 6. IoTR System Architecture



(a) Ardunio Uno R3

(b) Bluetooth Sheild

(c) GPRS Sheild



(d) GPS Sheild

Fig. 7. Hardware components interfaced with the IoTR

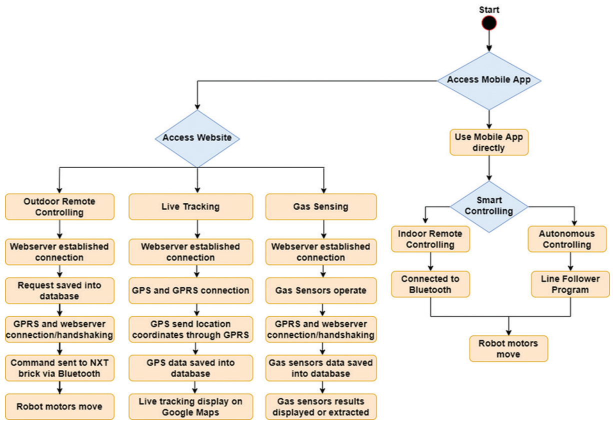


Fig. 8. State Machine Chart for the microcontroller Interfaced with the complete System

3.1 REAL-TIME VISUAL MONITORING

The patrol robot has one eye to provide a real-time visual monitoring of the robot's location and the surrounding area. The IP camera shown in Fig. 9 has been used for this purpose. It has several features such as, it can be accessed through the internet, has night visibility up to 15 or 20 meters, a low-cost, support smartphone, it is equipped with a motion sensor for automatic triggering of the camera, and can take a snapshot or record live video streaming. This robotic system can be used remotely from homes or work areas. Therefore, the camera has been accessed remotely through a static IP address from an Internet Service Provider (ISP) which is required for the stability and security of the network. As the main purpose of this robotic system is security, therefore, one user can access the camera per account. A DNS (Domain Name System) has been assigned to the IP address. The user account is synchronized with the IP address by registering that account in the router with the specific port address to allow any incoming connection to access the IP address through that port.

Fig. 9. IP Camera

3.2 REAL-TIME REMOTE CONTROLLING

To control the movement of the robot, some commands must be sent from the mobile apps or web-server to move it forward, backward, right, left, or stop its movement [25, 26]. These commands will be saved in the web-server's database. The robot moves in a specific direction through a series of communications and handshaking that have been done between the web-server, GPRS, and Bluetooth modules.

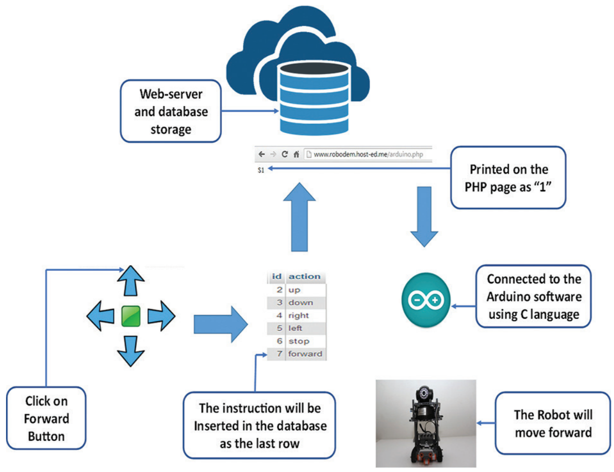


Fig. 10. Outdoor Controlling Process Steps

As the main purpose of this robotic system is to sense dangerous areas, outdoor controlling has been accomplished to save human lives. The robot has been controlled remotely by a developed web application by giving instructions for spatial movement or stopping. These instructions are represented by specific commands that are understandable by the NXT intelligent brick of the robot. As illustrated in Fig. 10, the user will click on one of the buttons to control the robot by moving it forward, backward, left, right, or stop it. Each request is associated with that clicked button will be received by the web-server. Then, the web-server network will send the request to be saved in the web-server database storage. The instruction will be saved in the last row at the instructions table. After that, the handshaking will be accrued between the web server and the GPRS. Then, the web-server will send the command through a generated PHP code that contains the movement command with a unique code. For instance, forward instruction is represented by '1' code. Each code has its response to be sent through the HTTP protocol between the web-server, the connected GPRS, Bluetooth, and Arduino. After that, the GPRS will send the response to the Bluetooth module. Finally, the motors of the robot will be able to move or stop based on each command. Fig. 11 shows the flowchart of the controlling process.

The goal of outdoor controlling cannot be achieved through the GPRS shield only because LEGO Mindstorm NXT robotics cannot be connected to the GPRS directly. LEGO Mindstorm NXT has only the ability to communicate with other devices and shields via Bluetooth. For this reason, a Bluetooth shield is used to integrate a Bluetooth module that can receive the data from the GPRS and send it to the robot by connecting the Bluetooth shield with the GPRS shield. Fig. 12 shows the hardware components required for real-time monitoring and controlling. The general architecture of the Bluetooth protocol telegram is shown in Fig. 13. Byte 0 contains the telegram type. The three types of telegrams (byte 1, 2, 3) represent the direct command telegram, reply command telegram, and the system command telegram respectively.

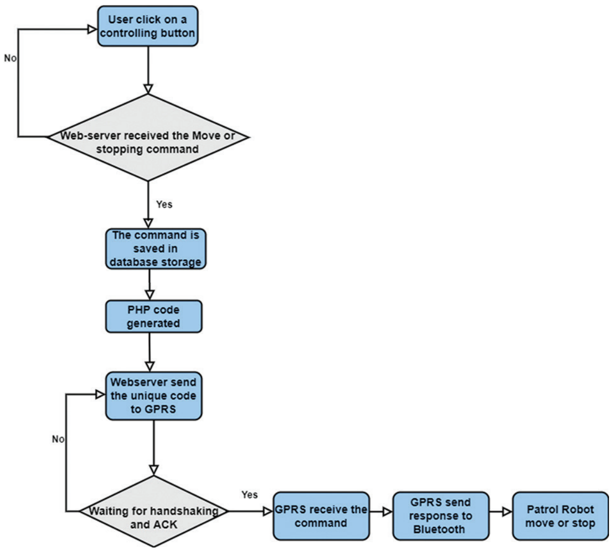


Fig. 11. Flowchart of the outdoor controlling process

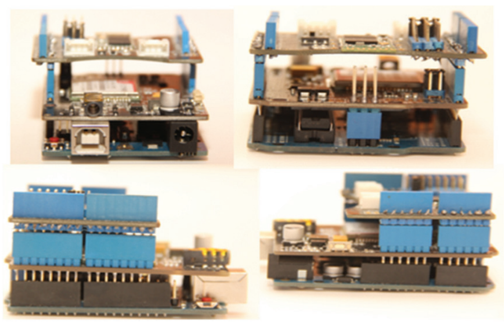


Fig. 12. Connected GPRS shield, Bluetooth shield, and Arduino UNO

For this system design, the direct command and the reply command have been used. Table 1 shows some important commands with their types that have been used for the robotic system. The other bytes consist of the command itself and the reply command based on each telegram type. The direct commands have a limited size which is 64 bytes including the byte that represents the telegram type. Two additional bytes are not included in the size limit, and they should be in front of the Bluetooth message as shown in Fig. 14. Based on testing and experiments, the motor's medium speed was set to 360 degrees/seconds. Fig. 15 shows that the patrol robot has reached a distance of 100m which is represented by the objective (green mark) within 10.2s. Therefore, the speed is accelerated by 9.79 m/s. While the patrol robot reached the same objective (red mark) within 5.1s with a fast speed of 19.58 m/s. In this system design, two motors have been connected to ports B and C of the NXT Brick. There are lists of commands used for the spatial movement of the patrol robot are listed below [27]:

Fig. 13. The General Architecture of the Bluetooth Protocol of LEGO Mindstorm NXT

Fig. 14. Bluetooth Message Architecture

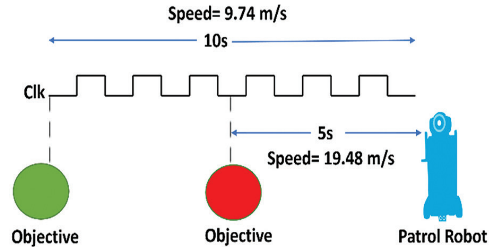


Fig. 15. Speed test bench for the IoTR movement

I. Moving the robot forward with fast speed

The command shown in Fig. 16-a is used for moving the robot forward with fast speed. The command consists of 14 bytes. The first two bytes contain the length of the message without counting these two bytes, the command will contain 12 bytes in decimal. The number should be converted to hexadecimal to be 0x0C in the first byte. The second byte contains 0x00 because one byte is enough for representing 12 in hexadecimal. The third byte has the value of 0x80 which is the command type that is used to send direct commands without waiting for a response. The fourth byte contains 0xff which means moving all motors. The fifth byte contains 0x64 in hexadecimal which is the turn ratio, and it has a range (-100 to 100) in decimal. This value should be positive to move the robot forward. The maximum speed here is 100 in decimal (64 in hexadecimal). The sixth byte contains 0x07, which are used for the mode byte which means turn on all motors. The remaining bytes will be the same for all commands. The meaning for 0x00 in the seventh byte is disabling the regulation. In the eighth byte, it is for clearing the turn ratio to move straight. The ninth byte has 0x20 enumerations which are used for setting the output to be run. Finally, the last four bytes which contain (0x00 0x00 0x00 0x00) are used for the taco limit to continue running indefinitely.

II. Move the robot forward with medium speed

This command moves the robot forward with medium speed, Fig. 16-b shows the command. This command is similar to the command of (moving the robot forward with fast speed) with one difference in the fifth byte. The range is (-100 to 100) in decimal, the half of this range has been taken which is "50" in decimal then we converted it to hexadecimal which is "32". Therefore, the byte will be 0x32.

I. Moving the robot backward with fast speed

The command is shown in Fig. 16-c, used for moving the robot backward with fast speed. For this command, the fifth byte has been adjusted to be 0x9C. A negative value is chosen to move the robot backward. Therefore, the fifth byte represents the least two significant bits for -100 in decimal. The maximum speed has been used in a negative hexadecimal representation.

II. Moving the robot backward with medium speed

For moving the robot backward with medium speed, the command is shown in Fig. 16-d. The Fifth-byte value is decreased to be 0xCE in hexadecimal, which represents the least two bits -50 in decimal.

III. Move the robot to the left

This command is used to move the robot to the left. Fig. 16-e shows the structure of moving the robot to the left command. Moving all the motors doesn't require moving the robot to the left. For moving the robot to the left, motor B should move to the left and motor C should stop. Therefore, the value of the fourth byte will be modified to be 0x01 to select the output of motor B. Fifth byte will be 0x32 to move the robot with a medium speed.

IV. Moving the robot to the right

The command is shown in Fig. 16-f, used for moving the robot to the right, motor C should move to the right and motor B should stop. This command is similar to the command of moving the robot to the left, the difference is changing the fourth byte to be 0x02. So, motor C will be selected to be turned to the right.

V. Stop moving the robot

The last command is shown in Fig. 16-g is used to stop the robot's movement. All motors will be in an idle mood in this command. Thus, the fifth byte will be set to be 0x00.

Table 1. The important commands with their types

The command	The type
0x00	Direct command telegram, response required
0x01	System command telegram, response required
0x02	Reply telegram
0x80	Direct command telegram, no response required

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x0C	0x00	0x80	0x04	0xff	0x64	0x07	0x00	0x00	0x20	0x00	0x00	0x00	0x00

(a) Forward Command telegram with Maximum Speed

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x0C	0x00	0x80	0x04	0xff	0x32	0x07	0x00	0x00	0x20	0x00	0x00	0x00	0x00

(b) Forward Command Telegram with Medium Speed

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x0C	0x00	0x80	0x04	0xff	0x9C	0x07	0x00	0x00	0x20	0x00	0x00	0x00	0x00

(c) Backward Command Telegram with Maximum Speed

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0x0C	0x00	0x80	0x04	0xff	0xCE	0x07	0x00	0x00	0x20	0x00	0x00	0x00	0x00

(d) Backward Command Telegram with Medium Speed

0	1	2	3	4	5	6	7	8	9	10	11	12
0x0C	0x00	0x80	0x04	0x01	0x32	0x07	0x00	0x00	0x20	0x00	0x00	0x00

(e) Moving Left Command Telegram

0	1	2	3	4	5	6	7	8	9	10	11	12
0x0C	0x00	0x80	0x04	0x02	0x32	0x07	0x00	0x00	0x20	0x00	0x00	0x00

(f) Moving Right Command Telegram

0	1	2	3	4	5	6	7	8	9	10	11	12
0x0C	0x00	0x80	0x04	0xff	0x00	0x07	0x00	0x00	0x20	0x00	0x00	0x00

(g) Stop Moving Command Telegram

Fig. 16. Different Command Telegram used for IoTR movement

3.3 REAL-TIME TRACKING SYSTEM

The tracing system is a fundamental sub-system in this robotic system as it is used for monitoring the movements of the patrol robot and live navigation for outdoor applications. Besides, the tracking system provides some other data such as current time, date, and the coordinates of the location during the movements of the robot. GPS module has been added to the hardware components for achieving a live tracking system. The GPS data will be sent to the web-server database through the GPRS connection. Then the user will be able to display the live location on Google Maps alongside the timing log data. The flowchart of the tracking system is shown in Fig. 8.

The GPS module that has been used in this system is EM-406. The GPS module is connected to the SparkFun GPS shield as shown in Fig. 17. The GPS Shield is connected at the top of the hardware implementation as shown in Fig. 18. The GPS shield has been connected with the GPRS shield using the pin assignment as given in Table 2. Pin 2 and 3 of the Arduino UNO are used only for the serial mode of the GPRS. To tackle this challenge, three steps have been accomplished. Firstly, the GPS shield should be connected at the top of the hardware system as these two pins can not be used for the serial mode of the GPS. Secondly, these two pins have been flexed to be out of connection with the Arduino. Thirdly, two other pins (9,10) will be plugged using two jumper wire connectors for the serial mode of the GPS, instead of the tucked GPS pins (1,3). Pin (2) will be connected with pin (9) of the GPS shield and pin (3) will be connected with pin (10).

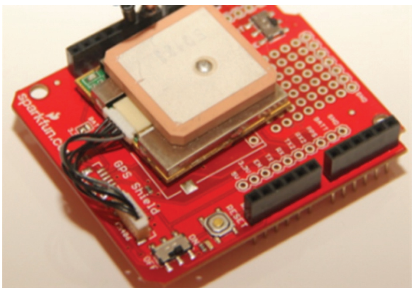


Fig. 17. GPS Module connected to the GPS Shield

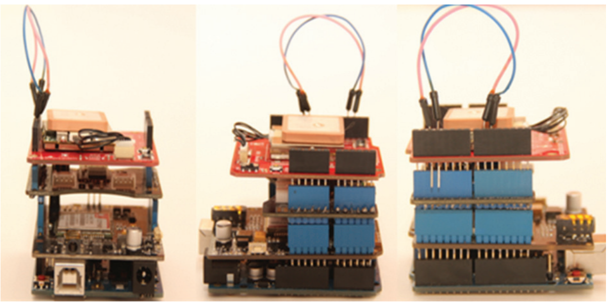


Fig. 18. GPS shield connected on the top of the hardware implementation

Table 2. EM- 406 GPS module pin assignment

Pin no.	Pin name	Usage
1	Enable/disable	For ON and OFF
2	GND	Provide the ground for the board
3	GPS-Rx	Receiving software commands
4	GPS-Tx	Outputting the measurement and navigation data to the user software
5	VIN	For DC supply

3.4 GAS LEAKAGE INSPECTION SYSTEM

The gas explosion occurs from a gas leakage with the excitement of an inflammation source [28]. The gas explosion happens with three conditions which are an explosive gas, an ignition source, and an oxidizer such as air or oxygen [29]. The most common explosive gases that are used for cooking and heating purposes are methane, butane, propane, and natural gas. In this proposed robotic system, LPG (Liquefied petroleum gas) has been the main explosive gas for testing and validation purposes. The reason for choosing LPG is this gas is available in the market and can be tested for personal and educational usage. While other gases such as natural gas and propane need a special safe environment for testing. LPG is propane or butane or a mixture of 60% propane and 40% butane that is used for commercial use. LPG gas is highly used for heating in homes or hotels or vehicles, however, LPG gas leakage is explosive and flammable [30].

For the LPG, the flame point is considered almost at the same point as the butane at 1970 °C in the air condition. In this research, we are not able to measure the flame point or the level of danger in terms of temperature, but we used a specific sensor to detect the LPG gas under certain conditions which is the MQ-6 gas sensor. There are different types of gas sensors in the market. These gas sensors have a small heater with an electrochemical sensor [31]. Besides, they are used at room temperature and outdoors, below are four different gas and chemical sensors that have been tested and validated.

A. MQ-5 Gas Sensor

MQ-5 sensor can be used in several types of applications such as "Domestic gas leakage detector, industrial combustible gas detector, and portable gas detector" [32]. This gas sensor is used to detect the leakages of natural gases.

B. MQ-6 Gas Sensor

MQ6 has been used due to two reasons which are low conductivity in clean air and low sensitivity to cigarette smoke and alcohol. Therefore, it is easy for testing and validation. Moreover, the MQ-6 gas sensor has a fast response time which is less than 10s [33].

C. ChemSee's Nitro-Pen Sensor

Trinitrotoluene (TNT) is a popular explosive material that has been used in many military and industry applications [34]. ChemSee's Nitro-Pen Sensor is a very effective chemical detector that can be built into the patrol robot to detect TNT. Fig. 19 shows the TNT testing using that detector.

D. Nanotechnology Chemical Sensor

A new chemical sensor chip joint with carbon Nanotubes has been made[35]. It enables the rapid detection of TNT in rivers and reservoirs. Fig. 20 shows the chemical sensor chip using Nanotechnology and the detection of the TNT process. This sensor is used by semiconducting carbon nanotube network transistors to make extremely sensitive sensors that are capable of operating stably underwater. Therefore, it is sensitive in the range of a few parts per billion for the detection of explosive compounds such as TNT [35]. Fig. 21 shows a diagram of a Nano-tube transistor on a flexible chip for detecting toxins or explosives in a water sample.

Based on experiments and testing, this research focused on using two sensors which are MQ5 and MQ6. The usage of MQ5 and MQ6 has the advantages of its low cost, availability in the market, and is lightweight compared to other gas sensors. On the other hand, these types of sensors have a lack of selectivity and cross-sensitivity to environmental factors such as temperature and humidity. In order to tackle this challenge, two sensors have been used, one for operating different temperature measurements and the other one for humidity measurements. These gas sensors obtain fast and calibrated measurements for concertation type of gases. The main hardware components that are used for hazard gases detection are explained in the following subsections.

3.4.1 MQ6 GAS SENSOR

MQ-6 gas sensor has a high sensitivity to LPG, Iso-butane, and propane gas. MQ6 gas sensor has a shield that is compatible with the Arduino shield. This shield of MQ6 has 3 pins which are Vcc, GND, and Vout. These 3 pins have connected to the Arduino shield; The Vcc pin of the MQ6 sensor is connected to +5v of the Arduino shield. The GND pin of the MQ6 sensor is connected to the GND of the Arduino shield. The Vout pin of the MQ6 sensor is connected to the A0 pin, which represents analog input to the Arduino. Fig. 22 shows the schematic circuit for the MQ6 sensor connected to the Arduino shield.

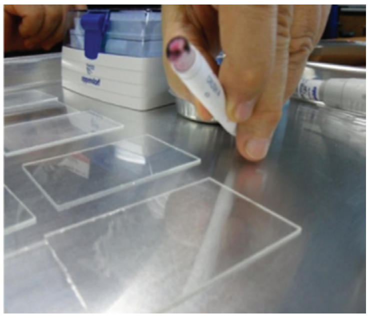


Fig. 19. TNT Testing using ChemSee's Nitro-pen

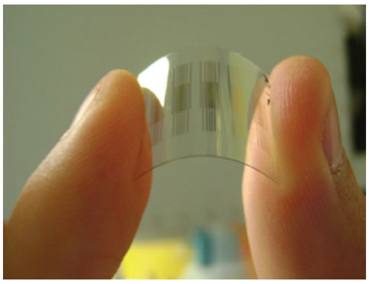


Fig. 20. The Chemical Sensor Chip using Nanotechnology and detection of TNT process

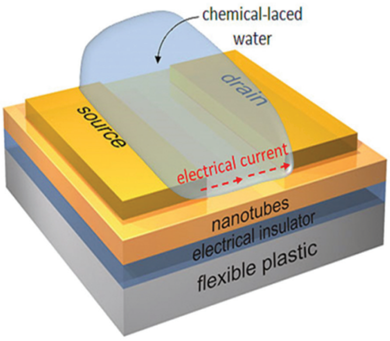


Fig. 21. The diagram of a Nano-tube transistor on a flexible chip for detecting toxins or explosives in the water

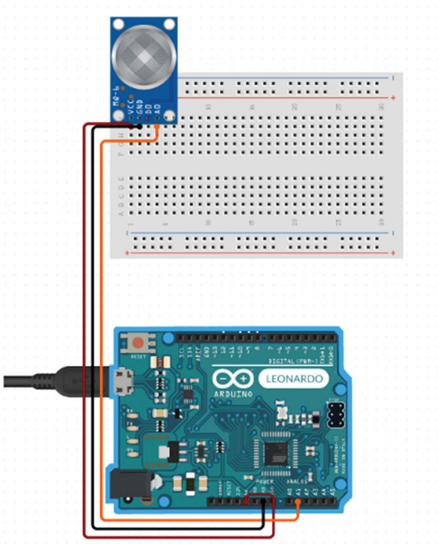


Fig. 22. Schematic circuit for MQ6 Gas sensor connected to Arduino UNO

The sensor is connected at the top of the GPS shield as shown in Fig. 23. According to [36], the sensor provides a reading of 100mv in the clean air, then the output voltage starts to increase according to the increase of the gas concertation. The resistance ratio of the sensor is calculated using Rs/Ro. Rs means the resistance of the sensor in 1000ppm methane under different temperatures and humidity. Ro means the resistance of the sensor in the environment of 1000ppm propone. Rs calculated as in Equation (1).

$$Rs = \left(\frac{v_c}{v_{RL} - 1}\right) * RL \tag{1}$$

V_{RL} is the voltage across the load resistance which is the voltage output of the MQ6. V_c is the reference voltage which is equal to 5v. RL is the load resistance which is 10 kΩ. PPM value of the LPG concertation is calculated using Equation (2).

$$PPM = \sqrt[-0.421]{\frac{Rs}{Ro * 18.446}} \tag{2}$$

The calibration of the sensor has been done with the collaboration of the HPC company team. Based on experiments and testing, the threshold level of the gas sensor result is 750mv to be considered a dangerous level.

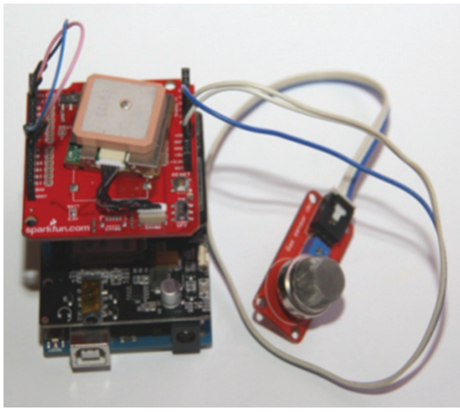


Fig. 23. Gas detection using MQ-6 Gas Sensor and Interfacing to System Board

3.4.2 TEMPERATURE SENSOR

The proposed gas sensor that has been used, worked within a certain condition of temperature which is in a range of -10C to 70C. This was the reason to use a temperature sensor to measure the temperature in the tested environment if it meets this condition. The temperature sensor used for that purpose is LM35. It works in a temperature range between -55C to 150C [37]. This range will contain the range of the gas sensor condition. LM35 has three pins to be connected. The first pin from the left is connected to the VCC of the Arduino. The second pin is the Vout. Vout is connected to resistance that equals 4.7 KΩ and it is connected to the analog input pin A2 of the Arduino. The third pin is connected to the common GND in the Arduino chip. Fig. 24 shows the schematic design of the LM35 sensor with the Arduino shield.

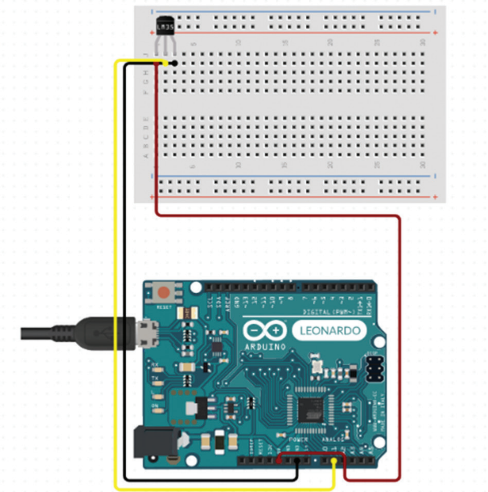


Fig. 24. Schematic design of the LM35 connected to Arduino shield

Connecting this sensor in the proposed system will be on the top of the GPS shield as shown in Fig. 25. The 4.7 KΩ resistance is used for the safety of the circuit and to avoid burning of the temperature sensor. The output value obtained from the analog pin will be converted to mv and then it will be analyzed and converted to Celsius degrees in the software development code using C language.

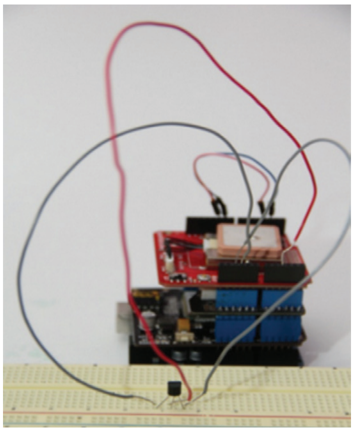


Fig. 25. LM35 Temperature Sensor connected to the system

4. SOFTWARE USER INTERFACE

Patrol robot communicates with mobile apps or web-server through GPRS network to send or receive commands. The default setting of the baud rate for the GPRS is 19200. Table 3 displays the AT commands that have been used for the connection of GPRS [9]. The web-based system software with smart applications has been designed and tested, and it is available online, a screenshot of the website system is shown in Fig. 26. Additionally, the mobile application has been developed and tested to achieve the main four sub-systems of the patrol robot. Some of the main pages of the mobile application are shown in Fig. 27. The users have the flexibility to use both of them based on their preferences. Fig. 28 shows the flow chart for the mobile application of IoTR.

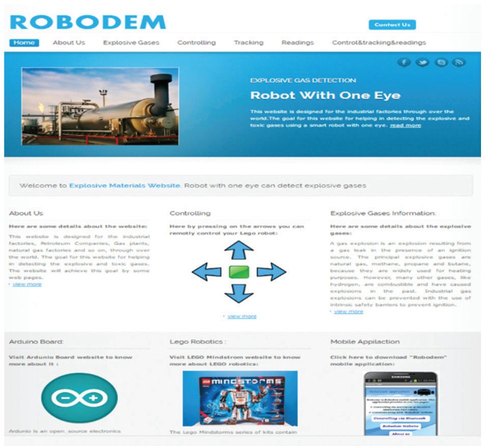


Fig. 26. The interface for the IoTR website

Table 3. AT commands used for the GPRS

The AT command	The meaning
AT+CSQ	Check Signal quality
AT+CGATT	Check the status of Packet service attach
AT+SAPBR	Bearer settings for applications based on IP
AT+HTTPINIT	Initiate the HTTP request

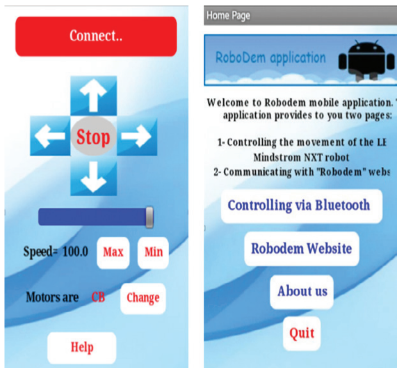


Fig. 27. Some pages of IoTR Mobile Application

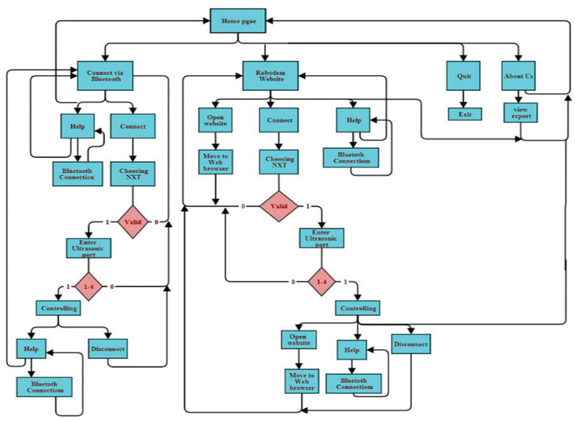


Fig. 28. Flow chart for mobile application of IoTR

5. TESTING AND VALIDATION RESULTS

This section presents the experimental results in terms of testing and validation for the whole system and subsystems. Different experiments have been conducted mostly at the University of Bahrain campus whether indoor or outdoor. Testing web-server and mobile apps verified successfully by sending and receiving the data from and into the database. The full hardware implementation of the IoTR prototype is shown in Fig. 29, which contains all the hardware units and sensors. The testing and validation results of each subsystem have been discussed in the following subsections.

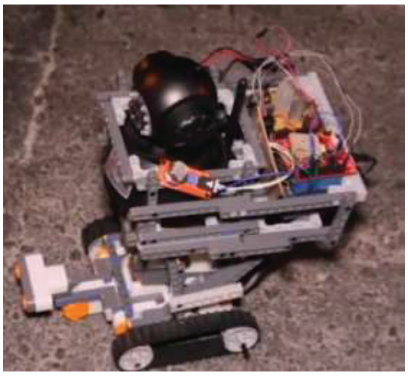


Fig. 29. IoT Robot

5.1 OUTDOOR CONTROLLING RESULTS

The IoT robot has been tested for outdoor controlling subsystem as shown in Fig. 30. The purpose of testing the outdoor controlling is to verify the output of the software development code and the hardware implementation by getting a fast response from IoTR. For the hardware implementation of the controlling system, the hardware units which are the Arduino UNO board, GPRS shield, and the Bluetooth shield have been connected. Then, the software code for controlling is uploaded and run. The results have been shown on the serial mode that obtains the data log of the controlling. Verifying the setup of the GPRS connection is an important step, as the GPRS is responsible for receiving the data from the database of the web-server and sending it back to the mobile robot to make a successful controlling through Bluetooth. After making a connection between the Arduino shield and the GPRS shield, the serial mode has been shown the connection data of the GPRS as in Fig. 31. The serial mode showed the SAPBR connection type of the GPRS is established. Also, it verified the local APN server which is the 'VIVA' network provider that has been used. After that, the AT command and the HTTP request have been successfully sent.

Fig. 30. Testing the Outdoor Controlling System

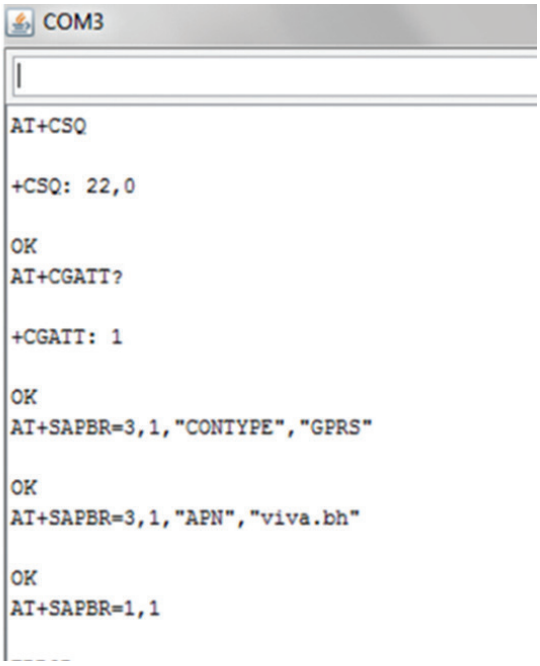


Fig. 31. Setting up the GPRS for outdoor controlling

The GPRS can transmit data outdoor at unlimited distances. Therefore, the users can remotely control the robot from anywhere to avoid accidents and save their lives in harsh environments. The data transmission speed (baud rate) used is 19200 bits per second (bps).

5.2 TRACKING SYSTEM RESULTS

The main purpose of testing the GPS shield is to verify the tracking and the localization of the IoT robot. Firstly, the GPS shield has been connected to the hardware units. After a few seconds, accurate GPS data has been received which are the date, time, longitude, and latitude of the location. Fig. 32 shows the GPS data and results on the serial mode. The tracking system has been tested to trigger the last location reporting and real-time tracking. Fig. 33 shows the results of the last location and reading history through the website. The live tracking has been tested at different locations when the robot moves. Fig. 34 and Fig. 35 show two different live locations for the movement of the robot.

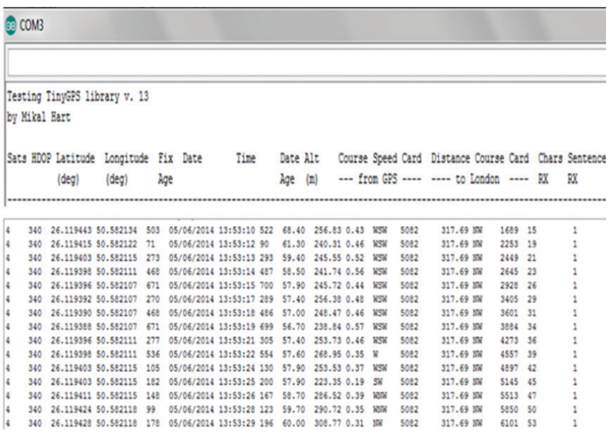


Fig. 32. The GPS data results on serial mode

id	latitude	longitude
1	26.23024	25.1000000
2	26.230236000000000	50.581074000000000
3	26.230495000000000	50.581062000000000
4	26.230097000000000	50.581093000000000
5	26.229809000000000	50.581017000000000
6	26.229956000000000	50.580967000000000
7	26.230247000000000	50.580685000000000
8	26.230190000000000	50.580750000000000
9	26.230288000000000	50.580788000000000

Fig. 33. The last location of the robot on Google map and the reading results of the GPS

Fig. 34. Live Tracking - Test location -1

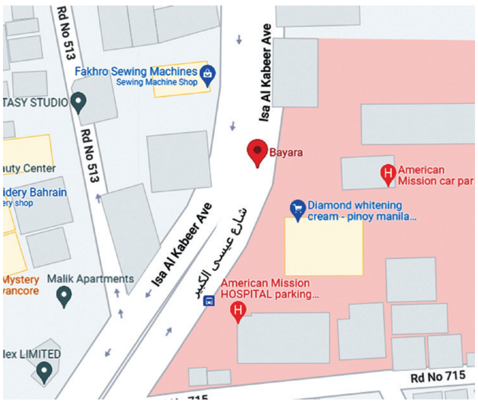


Fig. 35. Live Tracking - Test location -2

5.3 GAS SENSORS RESULTS

Firstly, the gas sensor (MQ6) is tested using a voltammeter. The results obtained from the voltammeter vary from 100 to 1200 mv. Secondly, the MQ-6 sensor has a high sensitivity to Propane, Butane, and LPG. Therefore, the testing has been done using a cigarette lighter or smoker because it contains a percentage of butane. The temperature sensor has been tested by using a candle. After that, the voltage readings are increased when the smoker cigarette gets closer to the sensor. On the other hand, the voltage readings decreased slightly when the cigarette lighter got away from the sensor. All of these testing scenarios mean that the changing of the voltage readings depended on the concentration of the ppm of the gases. The concentration of the gases should be between 200-10000ppm. Fig. 36 shows the results of the MQ6 sensor. The change of LPG concertation and the temperature values have been validated under the same measurements. Fig. 37 shows the LPG monitoring chart. Fig. 37 Illustrates that the threshold level is 750 mV. If the LPG ppm concertation is less than the threshold which is 750 mv, it means the surrounding area is safe and contains fresh air. On the other hand, if the LPG ppm concertation is equal to or more than 750 mv, it means there is a leakage of hazardous gases that will cause accidents and explosions. The reading of smart sensors has been verified through the database storage. Besides, the system generates user-friendly dashboard visualization reports for the gases and temperature sensors readings based on filtration by the date and time. Fig. 38 shows the sensor's results with showing the red highlighted row that represents a gas leakage. Finally, the system sends a warning message and a sign of gas leaks through the website and mobile application once there is a trigger of a gas leakage as shown in Fig. 39. Moreover, if the concentration of the explosive gas is getting higher, the Piezo buzzer will make an alarming sound.

```
COM3
GAS_SENSOR = 199.00mv
GAS_SENSOR = 197.00mv
GAS_SENSOR = 197.00mv
GAS_SENSOR = 700.00mv
GAS_SENSOR = 747.00mv
GAS_SENSOR = 752.00mv
GAS_SENSOR = 1012.00mv
GAS_SENSOR = 1015.00mv
GAS_SENSOR = 1017.00mv
GAS_SENSOR = 1016.00mv
GAS_SENSOR = 888.00mv
GAS_SENSOR = 719.00mv
GAS_SENSOR = 584.00mv
GAS_SENSOR = 522.00mv
GAS_SENSOR = 486.00mv
```

Fig. 36. Gas Sensor Results

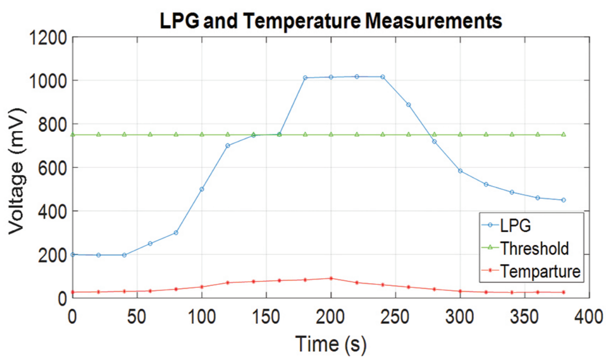


Fig. 37. LPG and Temperature Measurements

ID	Time	temperture	Gas
64	27-09-2014 17:40:55	27.83203321.00000*C	321 mv
65	27-09-2014 17:40:58	26.85547321.00000*C	321 mv
66	27-09-2014 17:41:03	28.32031320.00000*C	320 mv
67	27-09-2014 17:41:06	28.32031320.00000*C	320 mv
68	27-09-2014 17:41:10	27.34375319.00000*C	319 mv
69	27-09-2014 17:41:14	28.32031318.00000*C	318 mv
70	27-09-2014 17:41:17	27.83203316.00000*C	316 mv
71	27-09-2014 17:41:22	28.80859316.00000*C	316 mv
72	27-09-2014 17:41:24	26.36719312.00000*C	312 mv
73	27-09-2014 17:41:29	27.83203311.00000*C	311 mv
74	27-09-2014 17:41:32	27.83203309.00000*C	309 mv
75	27-09-2014 17:41:37	27.83203309.00000*C	309 mv
76	27-09-2014 17:41:39	27.34375308.00000*C	308 mv
77	27-09-2014 17:41:44	27.34375309.00000*C	309 mv
78	27-09-2014 17:41:47	28.32031309.00000*C	309 mv
79	27-09-2014 17:41:52	29.29688310.00000*C	310 mv
80	27-09-2014 17:41:55	28.32031310.00000*C	310 mv
81	27-09-2014 17:41:59	26.85547310.00000*C	310 mv
82	27-09-2014 17:42:02	29.29688310.00000*C	310 mv
83	27-09-2014 17:42:06	27.83203310.00000*C	310 mv
84	27-09-2014 17:42:10	27.34375310.00000*C	310 mv
85	27-09-2014 17:42:14	29.78516309.00000*C	309 mv
86	27-09-2014 17:42:17	27.34375309.00000*C	309 mv
87	27-09-2014 17:42:22	28.80859308.00000*C	308 mv
88	27-09-2014 17:42:24	42.48047576.00000*C	576 mv
89	27-09-2014 17:42:29	23.43750772.00000*C	772 mv
90	27-09-2014 17:42:32	38.08594637.00000*C	637 mv
91	27-09-2014 17:42:37	24.90234504.00000*C	504 mv
92	27-09-2014 17:42:40	24.41406469.00000*C	469 mv
93	27-09-2014 17:42:44	28.80859452.00000*C	452 mv

Fig. 38. The dashboard report of the gas sensor and temperature results

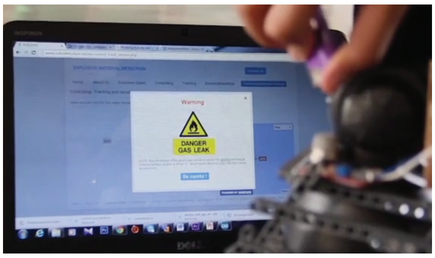


Fig. 39. A warning message appeared at the user-interface

6. CONCLUSIONS AND FUTURE WORK

This work describes the development and hardware implementation of a low-cost real-time IoT robot that can be used either indoors or outdoors for surveillance and gas leakage inspection purposes. The system has many features that are used by the security and gas detection system to locate the robot using GPS/GPRS for outdoor applications or Bluetooth for indoor applications. The design approach is economically better and has real-time features when compared to other approaches in the state-of-the-art related work. The proposed design achieves a low-cost and high quality, it has been tested indoors and outdoors for real-time monitoring, gas leakage inspection, remote controlling, live tracking, and video streaming. The developed autonomous mobile robot has the obstacle avoidance feature, which is a common feature of robotic patrolling. However, an optimized algorithm for obstacle detection is planned for future work. The hardware and software design of the gas detection system has been validated and tested using intelligent gas sensors. LPG gas has been sensed using MQ6 gas sensor. More different gases will be tested in future work. The network protocols between the microcontroller of the robot and the web-server have been established and tested for three main features which are controlling using the website and the mobile application, live video streaming using IP cameras, and live tracking on google maps.

7. REFERENCES

[1] D.-M. Tsai, T.-H. Tseng, "A template reconstruction scheme for moving object detection from a mobile robot", Industrial Robot: An International Journal, 2013, pp. 559-573.

[2] D. Xu, X. Peng, "Multi-level offloading decision on efficient object tracking for humanoid robot", Advanced Robotics, Vol. 30, No. 15, 2016, pp. 979-991.

[3] A. Shukla, H. Karki, "Application of robotics in onshore oil and gas industry—A review Part I", Robotics and Autonomous Systems, Vol. 75, 2016, pp. 490-507.

[4] D. A. Anisi, J. Gunnar, T. Lillehagen, C. Skourup, "Robot automation in oil and gas facilities: Indoor and onsite demonstrations", Proceedings of the IEEE RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18-22 October 2010, pp. 4729-4734.

[5] M. Choyekh et al. "Development and Operation of Underwater Robot for Autonomous Tracking and Monitoring of Subsea Plumes After Oil Spill and Gas Leak from Seabed and Analyses of Measured Data", Applications to Marine Disaster Prevention, 2017, pp. 17-93.

[6] Y. Li, Q. Liu, W. Li, "Development of a novel oil and gas in-pipe robot", International Journal of Mechatronics and Manufacturing Systems, Vol. 8, No. 3-4, 2015, pp. 102-115.

[7] Y. Tipsuwan, P. Hoonsuwan, "Design and implementation of an AUV for petroleum pipeline inspection", Proceedings of the 7th International Conference on Information Technology and Electrical Engineering, Chiang Mai, Thailand, 29-30 October 2015, pp. 382-387.

[8] T. M. Yousif, A. K. Alharam, W. Elmedany, A. A. Al Khalaf, Z. Fardan, "ROBODEM Remote Monitoring System Using Web/Mobile Applications", Proceedings of the Fifth International Conference on e-Learning, Manama, Bahrain, 18-20 October 2015, pp. 285-289.

[9] T. M. Yousif, A. K. Alharam, W. Elmedany, A. A. AlKhalaf, Z. Fardan, "GPRS-based robotic tracking system with real time video streaming", Proceedings of the IEEE 4th International Conference on Future Internet of Things and Cloud Workshops, Vienna, Austria, 22-24 August 2016, pp. 299-303.

[10] M. Sharma, H. Elmiligi, F. Gebali, "Performance evaluation of real-time systems", International Journal of Computing and Digital Systems, Vol. 4, No. 1, 2015.

[11] D. M. Alghasra, H. Y. Saeed, "Guiding Visually Impaired People with NXT Robot through an Android Mobile Application", International Journal of Computing and Digital Systems, Vol. 2, No. 3, 2013.

[12] S. Soldan, J. Welle, T. Barz, A. Kroll, D. Schulz, "Towards autonomous robotic systems for remote gas leak detection

and localization in industrial environments", Springer Field and service robotics, 2014, pp. 233-247.

[13] M. Faisal, H. Mathkour, M. Alsulaiman, "Smart mobile robot for security of low visibility environment", Proceedings of the 5th National Symposium on Information Technology: Towards New Smart World, Riyadh, Saudi Arabia, 17-19 February 2015, pp. 1-6.

[14] J. Patoliya, H. Mehta, H. Patel, "Arduino controlled war field spy robot using night vision wireless camera and Android application", Proceedings of the 5th Nirma University International Conference on Engineering, Ahmedabad, India, 26-28 November 2015, pp. 1-5.

[15] V. A. Shadrin, S. A. Lisakov, A. N. Pavlov, E. V. Sypin, "Development of fire robot based on quadcopter", Proceedings of the 17th International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices, Erlagol, Russia, 30 June - 4 July 2016, pp. 364-369.

[16] K. S. Lee, M. Ovinis, T. Nagarajan, R. Seulin, O. Morel, "Autonomous patrol and surveillance system using unmanned aerial vehicles", Proceedings of the IEEE 15th International Conference on Environment and Electrical Engineering, Rome, Italy, 10-13 June 2015, pp. 1291-1297.

[17] W. Rahmaniar, A. Wicaksono, "Design and implementation of a mobile robot for carbon monoxide monitoring", Journal of Robotics and Control, Vol. 2, No. 1, 2021, pp. 1-6.

[18] J. Palacín et al., "Application of an array of metal-oxide semiconductor gas sensors in an assistant personal robot for early gas leak detection", Sensors, Vol. 19, No. 9, 2019, p. 1957.

[19] T. Ma, S. Liu, H. Xiao, "Location of natural gas leakage sources on offshore platform by a multi-robot system using particle swarm optimization algorithm", Journal of Natural Gas Science and Engineering, Vol. 84, 2020, p. 103636.

[20] Y. Xing et al., "FireNose on mobile robot in harsh environments", IEEE Sensors Journal, Vol. 19, No. 24, 2019, pp. 12418-12431.

[21] S. Soldan, J. Welle, T. Barz, A. Kroll, D. Schulz, "Towards autonomous robotic systems for remote gas leak detection and localization in industrial environments", Springer Field and Service Robotics, 2014, pp. 233-247.

[22] D. S. Moore, R. J. Scharff, "Portable Raman explosives detection", Analytical and bioanalytical chemistry, Vol. 393, No. 6, 2009, pp. 1571-1578.

[23] M. Ghazal, R. Hamouda, S. Ali, "A smart mobile system for the real-time tracking and management of service queues", International Journal of Computing and Digital Systems, Vol. 5, No. 04, 2016.

[24] Y. Ismail, "A fast diamond motion estimation search algorithm for real time video applications", International Journal of Computing and Digital Systems, Vol. 3, No. 2, 2014, pp. 101-110.

[25] C. Bouras, A. Papazois, N. Stasinos, "Cross-platform Mobile Applications with Web Technologies", International Journal of Computing and Digital Systems, Vol. 4, No. 03, 2015.

[26] M. Eltoweissy, W. El-Medany, "Introduction to Special Issue on Mobile Applications", International Journal of Computing and Digital Systems, Vol. 5, No. 01, 2016.

[27] LEGO MINDSTORM, LEGO MINDSTORM NXT Direct Commands, http://remy-manu.no-ip.biz/Java/Tutoriels/Robot/ressources/LEGO_MINDSTORMS_NXT_Direct_commands.pdf (accessed: 2022)

[28] P. X. Thivel, Y. Bultel, F. Delpech, "Risk analysis of a biomass combustion process using MOSAR and FMEA methods", Journal of hazardous materials, Vol. 151, No. 1, 2008, pp. 221-231.

[29] V. Schröder, K. Holtappels, "Explosion characteristics of hydrogen-air and hydrogen-oxygen mixtures at elevated pressures", Hydrogen Knowledge Centre, 2005.

[30] H. Fayaz, R. Saidur, N. Razali, F. S. Anuar, A. R. Saleman, M. R. Islam, "An overview of hydrogen as a vehicle fuel", Renewable and Sustainable Energy Reviews, Vol. 16, No. 8, 2012, pp. 5511-5528.

[31] H. Maurer, F. Rieger, E. Linder, "Electro-chemical sensor construction," Modern Environmental Analysis Techniques for Pollutants, USA, Technical Report TR-0572595, 1979.

[32] J. Sun, Y. Li, X. Yan, "The design of automatic detection processing device of gas leakage based on the MB95204K", IEEE International Conference on Electrical and Control Engineering, 2011, pp. 1807-1809.

[33] A. Shrivastava, R. Prabhaker, R. Kumar, R. Verma, "GSM based gas leakage detection system", International Journal of Technical Research and Applications, Vol. 1, No. 2, 2013, pp. 42-45.

[34] S. J. Toal, J. C. Sanchez, R. E. Dugan, W. C. Trogler, "Visual detection of trace nitroaromatic explosive residue using photoluminescent metallole-containing polymers", Journal of forensic sciences, Vol. 52, No. 1, 2007, pp. 79-83.

[35] P. Shabecoff, A. Shabecoff, "Poisoned profits: The toxic assault on our children", 2nd Edition, Random House, 2008.

[36] HANWEI Sensors, Technical Data MQ6 Gas Sensor, https://datasheetspdf.com/pdf-file/699271/HANWEI/MQ6/1 (accessed: 2021)

[37] J. Akhavan, "The chemistry of explosives", 4th Edition, Royal Society of Chemistry, 2011.