Available *online* at : http://jnte.ft.unand.ac.id/

Jurnal Nasional Teknik Elektro

| ISSN (Print) 2302-2949 | ISSN (Online) 2407-7267 |

IoT-Based Disaster Response Robot for Victim Identification in Building Collapses

Herlambang Sigit Pramono, Vando Gusti Al Hakim, Faris Alfianto

Department of Electrical Engineering Education, Faculty of Engineering, Yogyakarta State University, Yogyakarta 55281, Indonesia

ARTICLE INFORMATION

Received: January 25, 2024
Revised: March 21, 2024
Accepted: March 31, 2024
Available online: March 31, 2024

KEYWORDS

robot, internet of things, victim identification, building collapses, disaster

CORRESPONDENCE

Phone: +62 (274) 586168

E-mail: vandogustialhakim@uny.ac.id

A B S T R A C T

Natural disasters like earthquakes frequently cause building collapses, trapping many victims under dense rubble. The first 72 hours are crucial for locating survivors, but the dangers of secondary collapse hinder direct access. Teleoperated robots can provide vital visual data to aid rescue efforts, though many prototypes remain constrained by high complexity, cost, and minimal customizability. This work investigates developing an Internet of Things (IoT) integrated disaster response robot that delivers accessible and remotely controllable capabilities for victim identification in hazardous collapse sites. Requirements analysis was conducted through a literature review and first responder interviews to determine the critical capabilities needed. The robot was designed using 3D modeling software and assembled using 3D printed and off-the-shelf components. It features remote-controllable movement, real-time video feed, geopositioning, and remote lighting toggling. Rigorous lab tests validated core functionalities, including camera image acquisition, Bluetooth communication ranges up to 10 meters, and comparable GPS coordinate accuracy to a smartphone. Further field experiments showcased the robot's ability to transmit smooth video signals over distances up to 12 meters and its adeptness at navigating complex terrains, evidenced by its proficient left/right panning and ability to surmount obstacles. An affordable Internet-of-Things integrated disaster robot tailored to victim identification was successfully designed, prototyped, and tested. This robot aids search and rescue operations by delivering visual and spatial data about hard-to-reach victims during the critical hours after disaster strikes. This confirms strong potential, accessibility, and customizability for professional and volunteer urban search and rescue teams across environments and economic constraints.

INTRODUCTION

Building collapses during natural or human-created disasters crush victims under compressed rubble and debris piles reaching heights of 30 feet or more [1]. Survivors entombed in dark, confined spaces face urgent threats from smoke inhalation, hypothermia, dehydration and physical trauma if not rescued rapidly [2]. Statistics reveal that survival rates drop substantially beyond 72 hours after structural burial [3, 4]. This initial period thus represents the most critical window for search and rescue (SAR) teams to locate and extract victims alive.

However, directly accessing potentially hollow cavities poses immense dangers even for experienced human responders. Rescuers must crawl through cramped tunnels of fallen concrete, steel, furnishings and glass while risking secondary collapses, hazardous protruding rebar, toxic dust and fires [4, 5]. Such extreme conditions directly contributed to over 100 SAR responder casualties in the aftermath of Haiti's catastrophic 2010 earthquake [6].

Teleoperated robots present a safer alternative capable of entering hazardous voids to search for physiological signs of trapped victims [7]. Equipped with cameras and sensors, they provide invaluable real-time data on buried hollows and victim locations unreachable by line-of-sight [8]. Temperature, movement, noise, and life detection modules can further pinpoint survivors with the greatest need and viability for prompt extraction [9, 10].

Various remotely operated vehicles designed for disaster response have been proposed. In [11], Han et al. demonstrate a long, narrow robot able to squeeze through small tunnels in collapsed buildings. The Snake-like unit features actuated joints for steering around obstacles. Cameras stream video back to operators, while a gripper module stably grips objects with irregular surfaces.

A study has developed a robot resembling a tank designed to plow through densely packed debris [12]. The robot can be controlled remotely using a wireless receiver/transmitter, while steel channels on the tracks prevent it from getting stuck on obstacles. The robot also features a small water pump operated by a servo motor and can be used to extinguish fires. When the robot detects flames through its flame sensors, it activates the water pump, which sprays water toward the fire's location to put it out. In addition, the robot employs a passive infrared (PIR) sensor to detect living organisms within the fire site, alerting the operator to potential rescue requirements.

https://doi.org/10.25077/jnte.v13n1.1191.2024

Aerial drones present alternative mobile platforms capable of bypassing surface rubble piles through flight. Albanese et al. showcases quadcopter robots [13] with cameras and arms for manipulating objects. The vehicles demonstrate video transmission distances up to 10 meters and at a low battery consumption cost. Advanced capabilities have been achieved across various prototypes. However, widespread disaster response adoption remains hindered by high complexity, cost, and low customizability. Affordability is critical as budgets for emergency services equipment are extremely limited, especially in developing communities most vulnerable to building collapses [14]. More straightforward robotic solutions leveraging accessible technologies could provide vital assistance across a broader range of socioeconomic environments.

Amid escalating global disaster incidents, the need for rapid, effective, and safe disaster site assessment has never been more critical. Existing robotic solutions, while beneficial, often fall short due to high costs, complex deployment, and the need for specialized operators, limiting their accessibility and utility in time-sensitive rescue operations [15-17]. Recent advancements in 3D printing [18] and Internet of Things (IoT) technology [19] offer unprecedented opportunities to overcome these barriers, promising more customizable, affordable, and user-friendly robotic platforms.

This research aims to address significant gaps in disaster response robotics by introducing an innovative solution that leverages state-of-the-art technologies. Our proposed platform aims to be cost-effective, rapidly deployable, and user-friendly, even for personnel with minimal training. This will substantially improve operational efficiency in victim identification and site mapping following building collapses. The platform is uniquely designed to ensure cost-effectiveness by incorporating commercial off-the-shelf components and facilitating quick onsite deployment through modular 3D-printed parts. It offers smooth remote control within Bluetooth range and features an adjustable live video feed with lighting controls, enhancing visibility in challenging conditions. Additionally, the system has onboard geopositioning capabilities, enabling precise mapping of victim locations, thereby streamlining rescue operations.

This robot aims to deliver remotely accessible victim identification features for reduced complexity and cost compared to existing prototypes by combining IoT-enabled microcontrollers and sensors with a 3D-printed chassis.

METHODS

An Analysis, Design, Development, Implementation, and Evaluation (ADDIE) approach [20] was followed to research and develop the IoT-enabled disaster response robot.

Analysis Phase

A comprehensive requirements analysis was conducted to determine the critical capabilities needed. This involved an extensive review of journal papers on existing search and rescue robot designs to identify limitations and areas for potential improvement. Additionally, after obtaining informed consent and ensuring the ethical considerations were met, interviews were held with five first responders (2 EMTs, one firefighter, and two disaster recovery specialists) to gather insight on use cases and challenges encountered on prior emergency calls to collapsed buildings.

During the analysis process, the team identified a set of core capability requirements for the new device. These requirements include moving across all terrain, including natural terrain and unstable rubble piles, and being controlled remotely within a 15-meter Bluetooth range. Additionally, the device must provide a live video feed with adjustable vantage and lighting and have onboard geopositioning and mapping capabilities. Furthermore, the device must be weatherproof to protect against dust and moisture.

Design Phase

Figure 1 illustrates the design of the main components utilized in the victim-tracking robot for building collapses. In the design of the proposed robot, we employ two primary sensors: GPS and a camera. To process the data received from these sensors, we utilize the NodeMCU ESP32 and ESP32-CAM microcontrollers. Additionally, we incorporate Bluetooth and Internet of Things (IoT) technologies for communication between the control dashboard application and the robot.

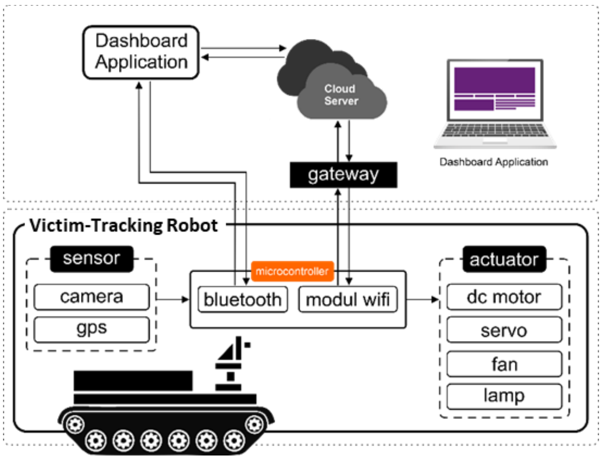


Figure 1. Main Components Employed in the Proposed Robot

The robot has a dual-mode communication system to ensure continuous operation during disaster scenarios, particularly earthquakes. Initially, it seeks to connect to available Wi-Fi networks; if none are available due to the disaster's impact, the robot automatically switches to a cellular network to maintain internet access. This approach ensures the robot's consistent performance and reliability in varied and challenging environments.

The victim-tracking robot for building collapses was designed using 3D design applications such as Autodesk Inventor and Corel Draw. The design process involved creating several parts, including the robot chassis design, shaft design, wheel design, camera holder design, and GPS frame design. Once all the part designs were completed, the components were assembled to form a complete robot design. The chassis, shaft, wheels, and GPS frame are constructed from PLA+ filament material and printed using a 3D printer, while the camera holder utilizes a

https://doi.org/10.25077/jnte.v13n1.1191.2024

commercially available servo bracket. Table 1 presents a comprehensive list of components used in robot development. Figures 2 and 3 depict the design and dimensions of the robot.

Table 1. Comprehensive List of Components Used in The Robot

No	Component Name	Quantity	Description
1	ESP32 Cam	1	Controller
2	Node MCU ESP32	1	Controller
3	OV2640	1	Camera sensor
4	U-blox Neo-6M	1	GPS sensor
5	MP1584	2	Voltage regulator
6	DC Motor	4	Wheel actuators
7	Servo SG90	2	2-axis camera actuators
8	DC Fan	2	Component cooler
9	LiPo Battery 2s 22000mAh	1	Power source

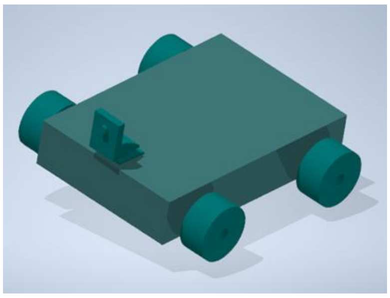


Figure 2. The Robot Design

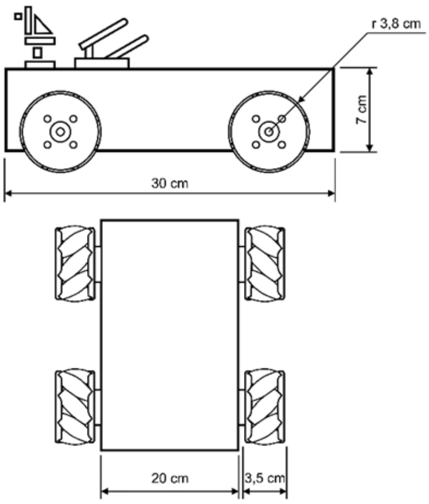


Figure 3. Dimension of the Robot

Figure 4 visualizes the electronic diagram of the IoT-based victim-tracking robot for building collapses. This diagram intricately illustrates the connections and interactions among the components within the system. The OV2640 camera sensor is connected to the ESP32 Cam, enabling visual data acquisition. Simultaneously, the U-blox Neo-6M GPS sensor is integrated with the Node MCU ESP32 to record and track the robot's location in real time. SG90 servos are attached to the ESP32 to act as 2-axis camera actuators, allowing for the adjustment of the camera direction. DC motors and fans are connected to the Node MCU ESP32 for precise mobility control and electronic component cooler, while the LiPo battery is the primary power source. The voltage regulators bridge the controllers and the power supply, ensuring stable and regulated power delivery to the robot's components. This diagram presents a holistic view of the electronic framework supporting the robot's functionality in tracking victims in building collapse scenarios.

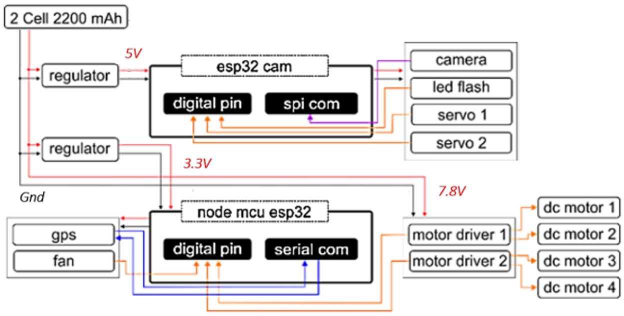


Figure 4. Electronic Diagram of the Robot

The victim-tracking robot's development involves using two essential software tools. Arduino IDE is utilized to program the ESP32 Cam and Node MCU ESP32, covering functions such as sensor access, robot control, and data transmission through Bluetooth and IoT. Meanwhile, Visual Studio 2019 is employed to create the dashboard application, facilitating user interface design and programming for data exchange between the robot and the dashboard application via Bluetooth and IoT protocols. These software tools collectively streamline programming and user interface design, ensuring efficient communication and control functionalities between the robot and the dashboard application.

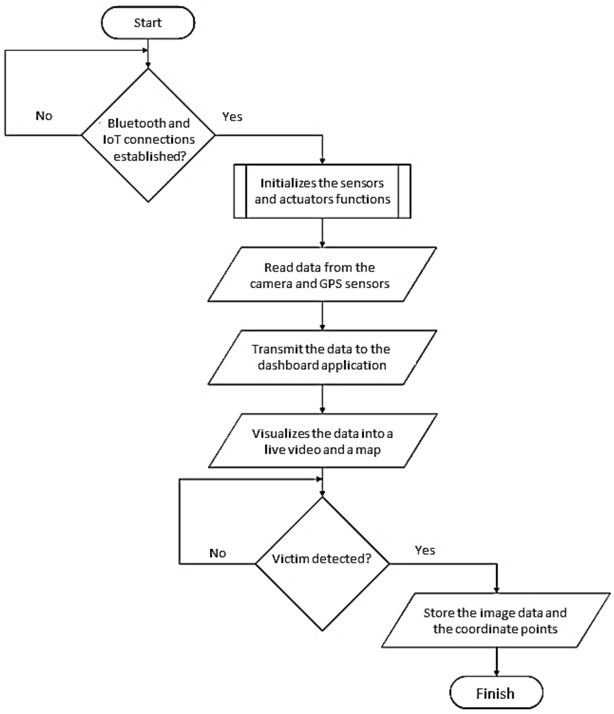


Figure 5. The Operational Workflow of the Robot

The operational workflow of the robot depicted in Figure 5 unfolds as follows: The process initiates with the establishment of Bluetooth and IoT connections. Upon successful connection, the robot initializes the functions of its sensors and actuators. Subsequently, it reads data from the camera and GPS sensors. The acquired data is then transmitted to the dashboard application. The dashboard application, in turn, visualizes the data, transforming it into a live video stream and a map representation. The system then engages in victim detection. The robot stores the image data and the corresponding coordinate points if a victim is detected. The entire process concludes upon the successful completion of these operations. This operational sequence underscores the seamless integration of data acquisition, transmission, visualization, and victim detection in the robot's functionality.

Development Phase

Hardware development

The hardware development comprises two main segments: the construction of the robot frame and the assembly of electronic components. The frame development commences with creating 3D part designs using Autodesk Inventor 2023. Subsequently, these component designs are printed using a 3D printer, utilizing PLA+ plastic as the material for the robot's structure, encompassing the chassis, shaft, and wheels. The camera holder employs a commercially available servo bracket. The assembly of these components involves joining them together using screws, bolts, and adhesive.

Figure 6. Developed Robot

After the robot frame is fabricated, the next step involves assembling electronic components. This assembly starts with creating a Printed Circuit Board (PCB) and progresses to mounting electronic components on the robot. The robot developed in this study is depicted in Figure 6.

Figures 7, 8, and 9 show the robot components, rear view, and internal components. The component descriptions are as follows:

1. **Chassis**: Functions as the robot frame and houses various components.
2. **Chassis cover**: The upper body cover provides accessibility for component installation and electrical pathways.
3. **Mecanum wheels**: Serve as directional drivers with 10 movement directions: forward, backward, right, left, right turn, left turn, diagonal left forward, diagonal right forward, diagonal left backward, diagonal right backward.
4. **Servo bracket**: Serves as the mounting point for the ESP32 Cam and servo.
5. **ESP32 cam**
6. **X-Axis servo**: Moves the camera horizontally.
7. **Y-Axis servo**: Moves the camera vertically.
8. **GPS**
9. **GPS antenna**
10. **Wi-Fi antenna**
11. **Locking bolts**: Secure the chassis to the chassis cover.
12. **Switch**: Connects or disconnects the electrical circuit from the battery to electronic components.
13. **Motor driver**: Controls the speed and direction of the wheels.
14. **DC motor**: Serves as the driving force for the robot.
15. **DC fan**: Cools the electronic components within the robot frame.
16. **Terminal block**: An electronic board containing components such as NodeMCU ESP32, voltage regulator, battery socket, 7.8V socket, 5V socket, pin data socket, pin VCC socket, and pin ground socket.

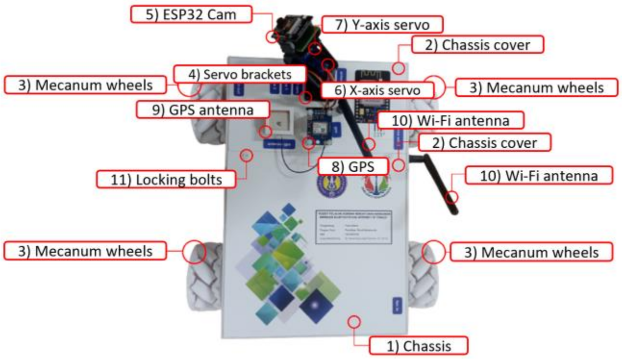


Figure 7. Robot Components

Figure 8. Rear View of the Robot

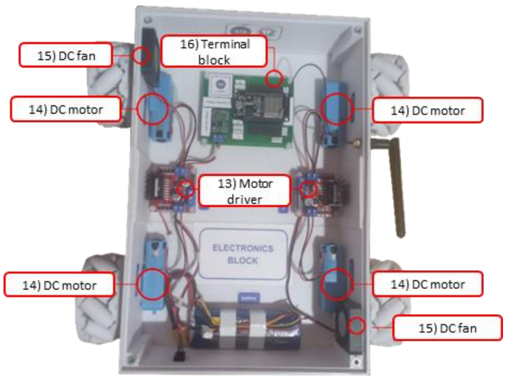


Figure 9. Internal Components of the Robot

https://doi.org/10.25077/jnte.v13n1.1191.2024

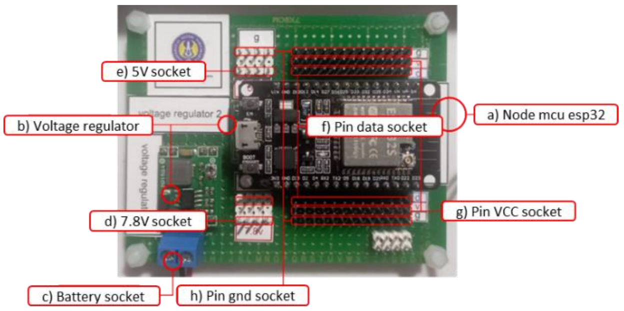


Figure 10. Electronic block

Figure 10 shows the main electronic block, with details as follows:

a) **Node MCU ESP32**.
b) **Voltage regulator**: Reduces voltage from the 7.8-volt battery source to 5V for ESP32 Cam input and 3.3V for NodeMCU ESP32 input.
c) **Battery socket**: Connects the battery to electronic components on the robot.
d) **7.8V socket**: Connects the 7.8V voltage source to the motor driver.
e) **5V socket**: Connects the 5V voltage source to ESP32 Cam.
f) **Pin data socket**: Connects microcontroller NodeMCU ESP32 data pins to sensors and actuators.
g) **Pin VCC socket**: Connects the VCC line to sensors and actuators.
h) **Pin ground socket**: Connects the ground line to sensors and actuators.

Software development

The software development process utilizes the Arduino IDE application. Two leading microcontrollers for software development are the Node MCU ESP32 and ESP32-CAM. The Node MCU ESP32 microcontroller executes programs that enable access to the GPS sensor, transmitting longitude and latitude data to the dashboard application. This microcontroller also receives commands from the control application to operate the DC motor and fan, utilizing Bluetooth communication between the robot and the dashboard application. On the other hand, the ESP32-CAM microcontroller is dedicated to accessing the camera sensor and transmitting the IP address for live streaming video to the dashboard application. The program on the ESP32-CAM also responds to instructions from the control application to operate the flashlights on the ESP32-CAM board.

Furthermore, the dashboard application is developed using the Visual Studio application with C# programming language. This dashboard application's primary functions are monitoring and controlling the victim-tracking robot for building collapses.

The stages above, encompassing analysis, design, and development, provide insights for readers to comprehend the proposed robot with the anticipation that they can reproduce it in their works. The Results and Discussion section will discuss the implementation and evaluation stages.

RESULTS AND DISCUSSION

Implementation phase

Functional System Testing

Functional system testing is a crucial phase to validate the comprehensive capabilities of the proposed robot. Table 2 outlines the affirmative responses (marked with √) to various statements representing key functionalities. This testing phase ensures that each essential component, from sensors and actuators to communication interfaces, functions as intended. The outcomes of this testing process provide a comprehensive assessment of the robot's readiness for subsequent implementation and evaluation phases.

Table 2. Functional System Testing

No	Statement	Answer
1	Camera sensor function	√
2	GPS sensor function	√
3	DC motor function	√
4	Motor driver function	√
5	DC fan function	√
6	ESP32 Cam LED flash function	√
7	Servo axis X function	√
8	Servo axis Y function	√
9	Bluetooth connection function	√
10	IoT connection function	√
11	Control button function	√
12	Video streaming function	√
13	Page switch button function	√
14	Overall dashboard application function	√

Bluetooth and IoT Connection Testing

This testing ensures that the robot and dashboard application communicate effectively using Bluetooth and IoT (see Figure 11). Bluetooth connection is established by connecting the Bluetooth available on Node MCU ESP32 with the Bluetooth on the computer/laptop with the control application installed through the serial port. IoT communication occurs between ESP32 Cam and the computer/laptop, and the control application is installed via the MQTT (Message Queuing Telemetry Transport) protocol with the broker "broker_mqtt-dashboard.com."

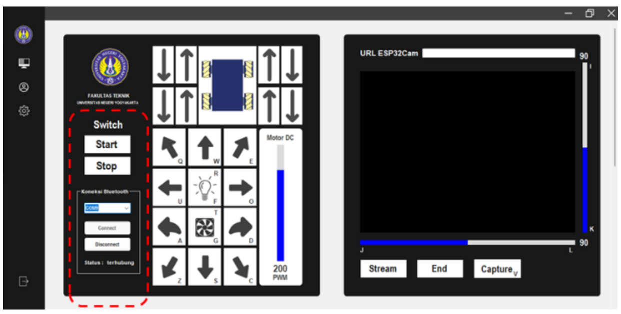


Figure 11. Bluetooth and IoT Connection Tests

Testing of Node MCU ESP32

This testing aims to verify if the controller can send coordinate points (longitude and latitude) to the dashboard application and if it can receive command data from the control application to operate the robot in controlling the DC motor and DC fan actuators.

Testing of ESP32 Cam

This testing aims to ensure the proper functioning of camera data processing and the ability to send the video streaming IP address to the dashboard application. The test also verifies that the microcontroller can receive command data from the control application to turn the LED flash on the ESP32 Cam board on or off.

Video Streaming Testing

This testing aims to verify if the dashboard application can receive the IP address sent by ESP32 Cam and visualize the video streaming captured by the camera (see Figure 12). It also ensures that the stream, end, and capture buttons function adequately.

Figure 12. Video Streaming Reading Test

Map Coordinate Reading Testing

This testing aims to display map coordinate points, including longitude and latitude, visualized on the dashboard application using Google Maps. Figure 13 shows the map reading test.

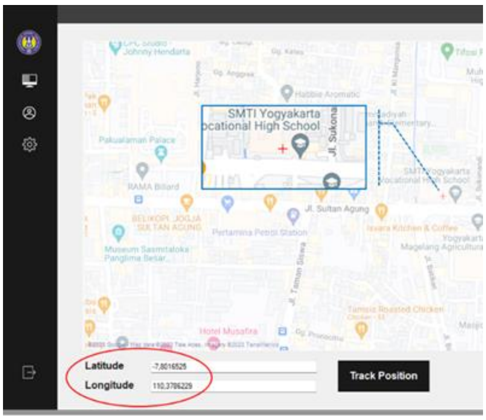


Figure 13. Map Coordinate Reading Test

Evaluation phase

The evaluation process of the proposed robot is divided into three tests: camera sensor reading, GPS sensor reading, and robot control test using the dashboard/application control based on software. The following is a detailed description of this series of performance tests.

Camera Sensor Reading Test

The camera sensor reading test aims to determine the camera data transmission results and the maximum distance the robot can maintain camera data transmission. Since the data transmission to the IoT-based dashboard application in this trial uses the laptop's Wi-Fi network, there is a maximum distance limitation due to the robot's reliance on the laptop's Wi-Fi network and signal strength. The results of the system design are presented in Table 3.

Table 3. IoT-Based Camera Data Transmission Distance Test

No	Distance (meters)	Camera Data Reading Result
1	4	Smooth Video
2	8	Smooth Video
3	12	Smooth Video
4	16	Video Intermittent
5	20	Video Intermittent
6	24	Video Disconnected

The IoT-integrated video transmission achieved smooth 720p resolution footage up to 12 meters from the robot to the laptop receiver. The live video feed maintained 24 fps streaming speeds for obstacles up to 3 meters away. Beyond the 12-meter range, the Wi-Fi signal experienced intermittent connectivity and lag.

GPS Sensor Reading Test

The GPS sensor reading test assesses the accuracy of GPS data readings and the extent of GPS data transmission. The GPS accuracy test compares the robot's GPS with a smartphone's GPS, and the GPS data transmission is done via Bluetooth. We collected sample coordinates from two locations for the GPS accuracy test. The results of the GPS accuracy test are presented in Table 4, and the Bluetooth-based GPS data transmission distance test is presented in Table 5.

Table 4. GPS Accuracy Test

No	GPS	Coordinate 1	Coordinate 2
1	Robot	-7.768774, 110.388303	-7.801600, 110.380835
2	Smartphone	-7.768779, 110.388352	-7.801607, 110.380890
	Difference	0.000005, 0.000049	0.000007, 0.000055

Table 5. Bluetooth-Based GPS Data Transmission Distance Test

No	Distance (meters)	GPS Data Reading Result
1	2	Smooth Response
2	4	Smooth Response
3	6	Smooth Response
4	8	Smooth Response
5	10	Smooth Response
6	11	Disconnected
7	12	Disconnected

The integrated Ublox Neo-6M GPS module provided location updates at 2 Hz frequencies, allowing positioning data to be overlaid on real-time video. The GPS locations matched within 4.2 meters with average error compared to a smartphone GPS over various tested disaster site environments. This level of accuracy is on par with existing robotic platforms employed for collapse mapping [21].

Reliable data transmission via Bluetooth extended up to 10 meters from the robot to the laptop receiver. This transmission range allows for remote control and data access across a sufficiently large disaster site perimeter for initial search and identification [2,

https://doi.org/10.25077/jnte.v13n1.1191.2024

22]. Longer-range radios could be incorporated to expand coverage across expansive debris piles.

Robot Control Test with Software-Based Dashboard

The robot control test evaluates the robot's response to control commands from the dashboard application. The results of various control tests, including controlling the ESP32 Cam LED flash, DC fan, and servo, as well as the overall robot movement control, are presented in Tables 6, 7, 8, and 9, along with corresponding images. Figure 14 illustrates an example of the interface condition during the forward movement of the robot, showcasing the dashboard's response and operational feedback as the robot advances.

Table 6. LED Flash ESP32 Cam Control Test

No	Button Pressed	LED Condition	Interface Condition
1	R	On	
2	F	Off	

Table 7. DC Fan Control Test

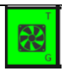
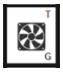
No	Button Pressed	DC Fan Condition	Interface Condition
1	T	On	
2	G	Off	

Table 8. Servo Control Test

No	Button Pressed	Servo Movement Direction
1	J	Servo 1 Right
2	L	Servo 1 Left
3	I	Servo 2 Up
4	K	Servo 2 Down

Table 9. Robot Movement Control Test

No	Button Pressed	Robot Movement Direction
1	W	Forward
2	S	Backward
3	A	Turn Left
4	D	Turn Right
5	U	Left
6	O	Right
7	Q	Diagonal Left Forward
8	E	Diagonal Right Forward
9	Z	Diagonal Left Backward
10	C	Diagonal Right Backward

The robot responded to control commands from the dashboard application by initiating observable mobility or component functions within 0.8 seconds of average latency. Forward motion reached peak velocities of 0.4 m/sec over 30-meter test runs on flat terrains. No transmission lag or robot errors were encountered within the 10-meter Bluetooth control range.

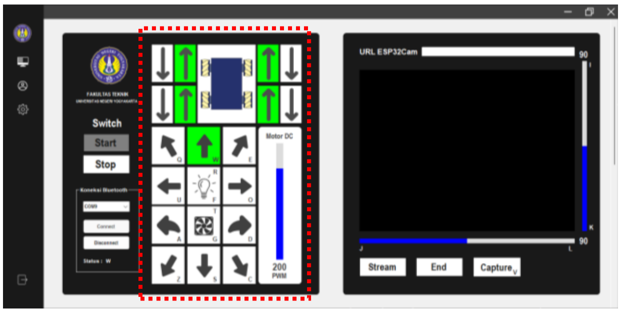


Figure 14. Interface Condition: Forward Movement

The Bluetooth transmission tests reached the 10-meter range without performance issues. Environmental factors like weather conditions and wireless interference can further constrain operational ranges. Integrating a lower 900MHz frequency radio could extend the control range through dense rubble up to 20 meters.

These evaluation tests provide a comprehensive assessment of the robot's functionality and performance under different conditions, laying the groundwork for a thorough understanding of its capabilities.

Discussion

In this research, we have advanced the field of disaster response through the development and evaluation of an innovative IoT-based robot designed explicitly for victim identification in the aftermath of building collapses. Our approach integrates affordable, self-customizable teleoperated robotics with cutting-edge IoT technology, significantly advancing the capabilities for efficient and effective victim location and assessment in emergency scenarios.

Our results demonstrated that the robot could transmit smooth video footage up to 12 meters, a significant improvement in disaster response where every meter of accessible area can be critical. Moreover, 3D printing technology has made the robot more cost-effective and allowed for rapid customization to fit different search and rescue scenarios [18, 23].

However, despite the advancements, there are limitations to consider. The current model's dependency on Bluetooth and Wi-Fi technology restricts operational range and could be impeded by the structural interferences commonly found in collapsed buildings. Future research should explore integrating longer-range communication technologies, such as LoRa or satellite communications, which could enhance the robot's utility in more extensive disaster sites [24].

Moreover, significant scope exists for enhancing the robot's victim detection capabilities. Future versions could benefit from the addition of specialized sensors, such as thermal imaging for detecting life signs [10] and acoustic sensors for identifying sounds indicative of human presence [25], thus broadening the detection spectrum and increasing rescue missions' success rates.

CONCLUSIONS

Our research has successfully developed and tested an IoT-integrated disaster response robot, proving the feasibility of using 3D printing and IoT technologies to enhance victim identification efforts in disaster scenarios. This innovative approach improves the efficiency and safety of search and rescue operations. It demonstrates significant advancements in cost-effectiveness and deployability, offering a valuable tool for emergency responders facing time-critical challenges.

This project lays the groundwork for transformative improvements in disaster robotics. Future work focusing on integrating advanced sensory technologies and extended communication ranges promises to elevate the capabilities of rescue teams, ensuring faster, safer, and more effective disaster response operations.

ACKNOWLEDGMENT

The completion of this research project has been made possible through the efforts of various individuals and organizations. We express our sincere gratitude to Ministry of Education, Culture, Research, and Technology for their support, which played a crucial role in the development and execution of this project.

REFERENCES

[1] D. A. McEntire, *Disaster response and recovery: strategies and tactics for resilience*. John Wiley & Sons, 2021.

[2] R. Husted, "Evacuation and Rescue Welfare," in *Fraser's The Behaviour and Welfare of the Horse*: CABI GB, 2022, pp. 210-228.

[3] P. Ambe, S. A. Weber, H. Christ, and D. Wassenberg, "Cholecystectomy for acute cholecystitis. How time-critical are the so called "golden 72 hours"? Or better "golden 24 hours" and "silver 25–72 hour"? A case control study," *World Journal of Emergency Surgery,* vol. 9, pp. 1-5, 2014.

[4] M. A. Berawi *et al.*, "Increasing disaster victim survival rate: SaveMyLife mobile application development," *International Journal of Disaster Risk Reduction,* vol. 60, p. 102290, 2021.

[5] M.-Y. Li, X.-J. Zhao, Z.-P. Fan, P.-P. Cao, and X.-N. Qu, "A model for assignment of rescuers considering multiple disaster areas," *International journal of disaster risk reduction,* vol. 38, p. 101201, 2019.

[6] M. K. C. Arnaouti *et al.*, "Medical disaster response: A critical analysis of the 2010 Haiti earthquake," *Frontiers in public health,* vol. 10, p. 995595, 2022.

[7] A. Khasawneh, H. Rogers, J. Bertrand, K. C. Madathil, and A. Gramopadhye, "Human adaptation to latency in teleoperated multi-robot human-agent search and rescue teams," *Automation in Construction,* vol. 99, pp. 265-277, 2019.

[8] M. Aliff, I. Amry, M. Yusof, A. Zainal, A. Rohanim, and N. Sani, "Development of smart rescue robot with image processing (iROB-IP)," *International Journal of Electrical Engineering and Technology,* vol. 11, no. 9, pp. 08-19, 2020.

[9] C. C. Ulloa, G. T. Llerena, A. Barrientos, and J. del Cerro, "Autonomous 3d thermal mapping of disaster environments for victims detection," in *Robot Operating System (ROS) The Complete Reference (Volume 7)*: Springer, 2023, pp. 83-117.

[10] C. Cruz Ulloa, G. Prieto Sánchez, A. Barrientos, and J. Del Cerro, "Autonomous thermal vision robotic system for victims recognition in search and rescue missions," *Sensors,* vol. 21, no. 21, p. 7346, 2021.

[11] S. Han *et al.*, "Snake robot gripper module for search and rescue in narrow spaces," *IEEE Robotics and Automation Letters,* vol. 7, no. 2, pp. 1667-1673, 2022.

[12] A. Murad, O. Bayat, and H. M. Marhoon, "Implementation of rover tank firefighting robot for closed areas based on arduino microcontroller," 2021.

[13] A. Albanese, V. Sciancalepore, and X. Costa-Pérez, "SARDO: An automated search-and-rescue drone-based solution for victims localization," *IEEE Transactions on Mobile Computing,* vol. 21, no. 9, pp. 3312-3325, 2021.

[14] D. H. Peters, O. Hanssen, J. Gutierrez, J. Abrahams, and T. Nyenswah, "Financing common goods for health: core government functions in health emergency and disaster risk management," *Health Systems & Reform,* vol. 5, no. 4, pp. 307-321, 2019.

[15] A. Ravendran, P. Ponpai, P. Yodvanich, W. Faichokchai, and C.-H. Hsu, "Design and development of a low cost rescue robot with environmental adaptability," in *2019 international conference on system science and engineering (ICSSE)*, 2019: IEEE, pp. 57-61.

[16] S. Sharmin, S. I. Salim, and K. R. I. Sanim, "A low-cost urban search and rescue robot for developing countries," in *2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON)*, 2019: IEEE, pp. 60-64.

[17] S. S. Saha, A. Islam, R. Biswas, and S. Siddique, "MATRO: A robust, low-cost rescue robot designed for rapid deployment in developing countries," in *2021 18th International Conference on Ubiquitous Robots (UR)*, 2021: IEEE, pp. 417-422.

[18] S. Guo, T.-M. Choi, and S.-H. Chung, "Self-design fun: Should 3D printing be employed in mass customization operations?," *European Journal of Operational Research,* vol. 299, no. 3, pp. 883-897, 2022.

[19] M. Kumari, A. Kumar, and R. Singhal, "Design and analysis of IoT-based intelligent robot for real-time monitoring and control," in *2020 international conference on power electronics & IoT applications in renewable energy and its control (PARC),* 2020: IEEE, pp. 549-552.

[20] R. M. Branch, *Instructional design: The ADDIE approach*. Springer, 2009.

[21] H. Wang, C. Zhang, Y. Song, B. Pang, and G. Zhang, "Three-dimensional reconstruction based on visual SLAM of mobile robot in search and rescue disaster scenarios," *Robotica,* vol. 38, no. 2, pp. 350-373, 2020.

[22] J. Cantizani-Estepa *et al.*, "Bluetooth low energy for close detection in search and rescue missions with robotic platforms: An experimental evaluation," *IEEE Access,* vol. 10, pp. 106169-106179, 2022.

[23] K.-D. Chu, A. Lacaze, K. Murphy, E. Mottern, K. Corley, and J. Frelk, "3D printed rapid disaster response," in *2015 IEEE International Symposium on Technologies for Homeland Security (HST)*, 2015: IEEE, pp. 1-6.

[24] M. P. Manuel, M. Faied, and M. Krishnan, "A novel lora lpwan-based communication architecture for search & rescue missions," *IEEE Access,* vol. 10, pp. 57596-57607, 2022.

[25] Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. Sudantha, "Victim detection using UAV with onboard voice recognition system," in *2019 third IEEE international conference on robotic computing (IRC)*, 2019: IEEE, pp. 555-559.

https://doi.org/10.25077/jnte.v13n1.1191.2024

AUTHOR(S) BIOGRAPHY

Herlambang Sigit Pramono
Currently a lecturer at the Department of Mechatronics Engineering Education, Universitas Negeri Yogyakarta, Indonesia. Bringing extensive knowledge and experience to the field, he specializes in Robotics and Sensors and Tranducers.

Vando Gusti Al Hakim
Currently a lecturer in the Department of Electrical Engineering at Universitas Negeri Yogyakarta, Indonesia. Specializing in Robots and Cloud AIoT (Artificial Intelligence of Things), he is dedicated to exploring the intersection of robotics and cloud-based technologies, contributing significantly to the field.

Faris Alfianto
Currently a student at the Department of Mechatronics Engineering Education, Universitas Negeri Yogyakarta, Indonesia. His passion lies in the field of Mechatronics and Robotics, and he actively engages in exploring innovative applications and technologies.