ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

A BLE BEACON GLOBAL TRACKING STRATEGY FOR SMALL ROBOT USING AN ESP32 BEACON SYSTEM AND GPS

Fredy H. Martínez S. and Angélica V. Rendón C.
Facultad Tecnológica, Universidad Distrital Francisco José de Caldas, Bogotá D. C., Colombia
E-Mail: fhmartinezs@udistrital.edu.co

ABSTRACT

This study aims to estimate the position of a small robot in both outdoor and indoor environments using a combination of RSSI-to-distance calculation, trilateration, and MQTT protocol for communication between ESP32 stations and the server. RSSI is utilized to approximate the distance between the ESP32 and the beacon, while trilateration is employed to determine the precise position of the beacon. The ratio of the imaginary circles is calculated by various signal measurement techniques, and the MQTT protocol is used to set up communication between ESP32 stations and the server. Once the beacon device is visible to all three ESP32 modules, trilateration takes place, and the position of the BLE beacon is displayed as a web page. The server collects the largest RSSI values from the nearest three ESP32 stations, and trilateration is performed from the distances obtained, with the coordinates being sent to the dashboard for visual representation. This strategy is supported by a GPS in outdoor environments, and indoors when structures allow it. Furthermore, the study includes the implementation of an experimental procedure and findings that involve utilizing ESP32 modules as scanning stations to track a BLE beacon.

Keywords: BLE beacon, ESP32, GPS, indoor/outdoor tracking, MQTT protocol, position estimation, RSSI, scanning stations, trilateration.

Manuscript Received 20 August 2023; Revised 27 February 2024; Published 30 March 2024

1. INTRODUCTION

In recent years, the demand for autonomous robots that can operate in diverse environments has increased significantly [1, 2]. Developing a reliable positioning system for such robots remains a significant challenge [3, 4]. While the Global Positioning System (GPS) has proven useful for outdoor localization due to its accuracy and availability, it is not suitable for indoor environments [5, 6]. Bluetooth Low Energy (BLE), on the other hand, has emerged as a promising solution for indoor localization due to its low power consumption and compatibility with smartphones [7, 8]. However, there are still limitations in terms of accuracy and reliability. In addition, there is a need for a low-cost and high-availability solution for small robots [9]. Therefore, this study proposes a global positioning system for small robots that work with GPS signals in outdoor environments and with BLE in indoor environments, incorporating Message Queuing Telemetry Transport (MQTT) through ESP32 controllers.

Various positioning systems have been proposed for robots, including GPS, BLE, and Wi-Fi [10]. Although GPS has been widely used for outdoor localization, its accuracy can be impacted by factors such as weather conditions, signal obstructions, and multipath errors [11]. Researchers have proposed different approaches to overcome these limitations, such as using a combination of GPS and other sensors or advanced algorithms for error correction [12]. Similarly, BLE has been increasingly used for indoor localization, but its signals can be affected by signal attenuation, multipath effects, and interference from other wireless devices [13]. To address these issues, researchers have proposed solutions such as using multiple BLE beacons, implementing advanced filtering algorithms, or integrating with other sensors [14].

Despite efforts to improve GPS and BLE-based positioning systems, there is still a need for a reliable and low-cost solution that can work in both outdoor and indoor environments [15]. Additionally, there is a need for a communication protocol that can handle data transmission between the localization system and the robot controller robustly and efficiently [16, 17]. This study aims to address these issues by proposing a global positioning system that works with GPS and BLE signals, incorporating MQTT through ESP32 controllers.

In this paper, three concepts are utilized to estimate the position of a robot in an indoor environment, namely the RSSI-to-distance calculation, trilateration using the distance calculated from RSSI, and MQTT protocol to establish communication between ESP32 stations and a server. The RSSI value, which indicates the strength of the beacon's signal, is used to approximate the distance between the ESP32 and the beacon. However, the RSSI value is susceptible to external factors such as absorption, interference, or diffraction, leading to unstable readings. Trilateration is employed to determine the relative location of the robot after the distances between the ESP32 modules and the beacon are calculated. The MQTT protocol is used for communication between the ESP32 stations and the server, with the data transmitted in the JSON packet format. The server uses the RSSI values to calculate the corresponding distances, which are then used in trilateration to obtain the coordinates of the beacon device. The final results are displayed on a web page. Additionally, environmental/external factors should be considered to rectify errors caused by signal propagation due to obstacles and other factors. These readings are contrasted by GPS readings when the building structure allows it; GPS is also prioritized in outdoor environments. Our proposed system has several advantages over existing solutions. Firstly, it provides accurate and reliable positioning in both outdoor and indoor environments. Secondly, it is a low-cost solution as it uses off-the-shelf components and open-source software. Thirdly, it ensures efficient and robust communication between the localization system and the robot controller through the MQTT protocol.

The proposed system presents a novel and valuable solution for small robotics applications, enabling precise and autonomous navigation in both outdoor and indoor environments. The integration of GPS and BLE through ESP32 controllers and MQTT offers a low-cost and high-availability solution that can be easily adopted for various mobile robotics applications. The remainder of the article presents a detailed description of the system's design and implementation, as well as experimental results demonstrating the system's performance.

2. BACKGROUND

The field of robotics is constantly evolving, with new technologies being developed to improve the accuracy and efficiency of robot localization and tracking systems. One such technology is the use of beacons, as proposed by [18], [19] have explored the use of a single-beacon vehicle for range-only localization, which can support the navigation of autonomous underwater vehicles (AUVs). Another important aspect of robotics is obstacle avoidance, and [20] has provided an overview of localization techniques in Wireless Sensor Networks (WSNs), as well as obstacle avoidance path planning localization algorithms based on V Curve. In terms of indoor positioning systems (IPS), [21] have demonstrated the implementation of IPS using MQTT protocol, ESP32 modules, and a BLE device. Similarly, [22] have proposed a real-time indoor tracking and positioning system using BLE beacon and smartphone sensors.

The prevention of theft is a major concern for many businesses, and [23] has focused on the development of a low-cost smart antitheft system for small property detection. Additionally, [24] aims to design a system that uses Bluetooth for Personal Active Positioning and Tracking (PAPR) tracking. In the context of mechanical build imperfections such as misalignment, [25] have proposed an extended calibration approach to improve the motion response of Omnidirectional Mobile Robots (OMRs).

It is evident from the above works that the use of beacons and Bluetooth technology has emerged as a promising solution for localization and tracking in robotics. Further advancements in this area can improve the accuracy and efficiency of robotic systems, which will ultimately lead to their widespread adoption in various industries. Additionally, it is worth noting the influence of earlier works such as [26] and [27], which have laid the foundation for the current state of the field.

3. MATERIALS AND METHODS

In this study, the goal is to estimate the position of a small robot in both outdoor and indoor environments using a combination of RSSI-to-distance calculation, trilateration, and MQTT protocol for communication between ESP32 stations and the server. The hardware configuration needed to experiment required the following components:

- **ESP32 modules (Espressif Systems)**: ESP32 modules are used to scan for beacon devices in the area. Three ESP32 modules are required for the trilateration method to determine the position of the robot.
- **BLE beacon devices**: BLE beacon devices are used as the source of signal for RSSI measurements. We use as a BLE beacon an ESP32 module.
- **DragonBoard 410C development board (Arrow)**: The development board acts as a broker for the MQTT protocol to establish communication between ESP32 modules and the server. It is also used to display the position of the robot on a web page.
- **Node.js server**: The server is used for further calculations and receives data from the DragonBoard 410C.
- **Arduino Controlled Servo Robot (SERB)**: Robotic platform used in the experiments. The control unit has been replaced by an ESP32 module, which also operates as a BLE beacon. The robot was also equipped with a NEO-6M GPS (UBLOX) for localization in outdoor environments, and additional support in indoor environments that allow its use.

RSSI (Received Signal Strength Indicator) is used to approximate the distance between the ESP32 and the beacon. The value of RSSI depends on distance and broadcasting power value. At maximum broadcasting power, the RSSI value ranges from -26 (a few inches) to -100 (40-50 m distance). However, the values of RSSI are influenced by external factors such as absorption, interference, or diffraction, which cause instability in the RSSI value as the distance of the beacon from ESP32 varies.

To determine the precise position of the beacon, the trilateration method is employed (Figure-1). This method is a geometrical model where the intersection formed by three imaginary circles of ESP32 access points determines the relative location of the user after the distances between ESP32 modules and the beacon are calculated. The radii of the imaginary circles are calculated by various signal measurement techniques such as Received Signal Strength (RSS), Time of Arrival (ToA), Time Difference of Arrival (TDoA), etc. Unlike the fingerprinting method, the trilateration method does not have an offline phase. However, the respective coordinates of the Access Points (AP) i.e., ESP32 modules and AP's MAC addresses transmitted, which are pre-stored either in the cloud or in a centralized database (DragonBoard 410C).

ARPN Journal of Engineering and Applied Sciences

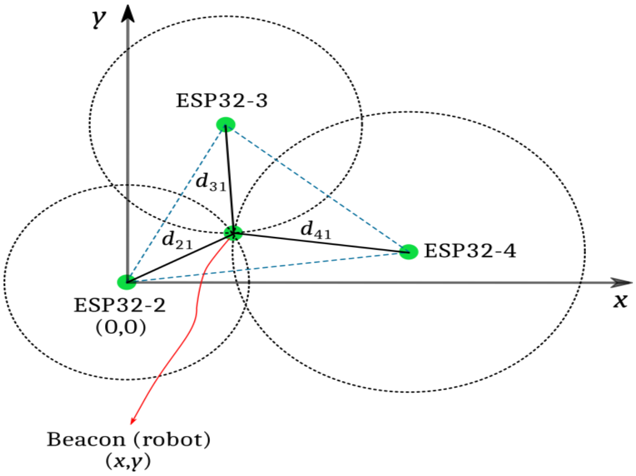

Figure-1. Use of trilateration with three ESP32 modules (modules 2, 3, and 4) to estimate the relative position of the robot, which has the BLE beacon (ESP32 1).

The MQTT protocol is used to set up communication between ESP32 stations and the server. MQTT is a publish-subscribe based messaging protocol that works upon TCP/IP protocol. It is designed for communication within a small area where a small code footprint is compiled, or when the network has limited bandwidth. An MQTT system consists of clients that communicate with a "broker," which is a server. A client can either be a publisher or a subscriber of information. Each client must be subscribed to the broker before publishing or subscribing to the information. Information is organized in a hierarchy of topics inside the server.

In this study, the ESP32 module scans the area for every small interval. Once the beacon device is visible to all three ESP32 modules, trilateration takes place, and the position of the BLE beacon is displayed on a web page on the DragonBoard display. The ESP32 modules are flashed with C code that scans for the beacon devices and filters to the required beacon. Filtering is done by scanning for the MAC address of the required beacon device, and its corresponding RSSI value is collected and formatted into the JSON packet with the MAC addresses of the ESP32 stations.

All three ESP32 stations are subscribed to the MQTT broker/server. The JSON data is sent to the MQTT broker, hosted by the DragonBoard. The Raspberry Pi listens to the data from the scanning stations (ESP32) always. This data is then sent to the node.js server, which is also subscribed to the MQTT broker. This data is used by the server for further calculations. The RSSI values received from the scanning stations are used as inputs to calculate the corresponding distances. The server collects the largest RSSI values from the nearest three ESP32 stations. These distances will be sent to the trilateration algorithm to obtain the coordinates of the beacon device. Trilateration is performed from the distances obtained, and the coordinates are sent to the dashboard for visual representation. The results are checked against the GPS position estimation if this signal exists, which is always the case when the robot is in an outdoor environment.

The experimental setup consisted of both indoor and outdoor environments, with different configurations of ESP32 modules and BLE beacon devices. The indoor environment was a 5x4 meter rectangular room, while the outdoor environment was an open field with an area of approximately 5000 square meters. Three ESP32 modules were strategically placed in the corners of the room, and a BLE beacon device was placed on the SERB robotic platform. For the outdoor environment, five ESP32 modules were placed at different locations to cover the entire area, and the SERB robotic platform was equipped with a NEO-6M GPS (UBLOX) for location estimation.
The SERB robotic platform used in the experiments had dimensions of 26 cm x 26 cm x 15 cm and was capable of moving in all directions (Figure-2). The robot was controlled by an ESP32 module, which also acted as a BLE beacon device. In addition to the BLE beacon device, the robot was equipped with a NEO-6M GPS (UBLOX) for location estimation in outdoor environments. The GPS provided additional support in indoor environments, allowing the robot to be used in both settings. The robot was capable of carrying small payloads and was powered by a rechargeable battery. The ESP32 module used to control the robot was programmed with custom code to scan for BLE beacon devices in the vicinity and relay the information to the server via MQTT protocol.

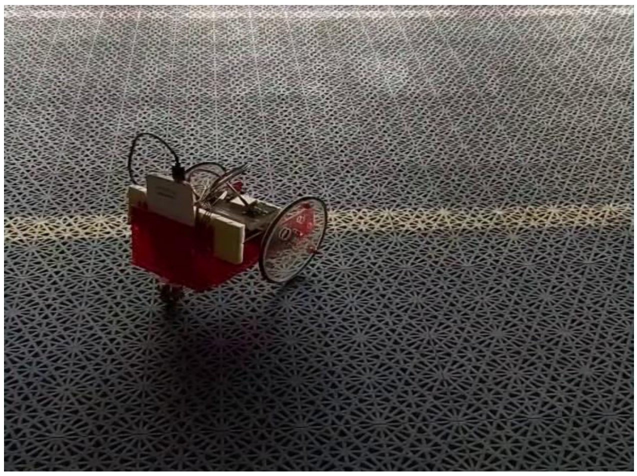

Figure-2. SERB Robot conditioned with beacon and GPS.

The data collection process begins with the ESP32 module scanning the area at frequent intervals to detect the BLE beacon devices. The scanning is done continuously to ensure real-time tracking of the robotic platform's position. Once the BLE beacon device is detected, the ESP32 modules collect the corresponding RSSI values, which are used to approximate the distance between the ESP32 and the beacon. It is worth noting that the values of RSSI are subject to external factors such as absorption, interference, or diffraction, which cause instability in the RSSI value as the distance of the beacon from ESP32 varies. To account for these external factors, the RSSI values are filtered to the required beacon device based on the MAC address of the device. The ESP32 modules then send the formatted RSSI values, along with the MAC addresses of the ESP32 stations, to the MQTT broker hosted by the DragonBoard 410C. The DragonBoard listens to the data from the scanning stations always, and the data is sent to the node.js server, which is also subscribed to the MQTT broker. The server uses the RSSI values to calculate the corresponding distances, and the largest RSSI values from the nearest three ESP32 stations are used to obtain the coordinates of the beacon device using the trilateration method.

The trilateration algorithm used in this study assumes that the BLE beacon signals propagate in a spherical shape and the RSSI values decay with distance. It also assumes that the positions of the BLE beacons are known and fixed and that the position of the robot can be accurately estimated. However, the algorithm has some limitations, including the presence of obstacles that can block or reflect the BLE signals, which can lead to errors in distance estimates. To mitigate these effects, a Kalman filter is used to estimate the position of the robot and reduce the impact of noise and errors in the measurements. The trilateration algorithm is applied by using the RSSI values collected from the BLE beacons and the known positions of the beacons to estimate the position of the robot. The results of the trilateration algorithm are validated by comparing them to the ground truth positions obtained using a motion capture system. The accuracy of the results is evaluated using the root mean square error (RMSE) and the mean error distance (MED) metrics.

The MQTT protocol is a lightweight messaging protocol designed for IoT devices with low bandwidth, high latency, or unreliable network connections. It is used for sending messages between devices and applications and is based on a publish/subscribe model. MQTT uses a broker to manage the message flow and allows clients to subscribe to topics of interest and publish messages on those topics. MQTT can be secured using Transport Layer Security (TLS) and supports authentication and authorization using username and password, X.509 client certificates, or other mechanisms. It also provides support for Quality of Service (QoS) levels to ensure reliable delivery of messages. The topic hierarchy used in MQTT is flexible and can be customized based on the application requirements. It is generally recommended to use a standardized message format such as JSON or XML to ensure interoperability between different systems. However, it is important to note that MQTT does not provide end-to-end encryption, so additional security measures may need to be implemented to protect sensitive data.

The node.js server used in this project serves as the central point for processing and displaying the location data collected from the ESP32 and BLE beacon devices. The server is implemented using several libraries, including the Express.js framework for handling HTTP requests and responses, the MQTT.js library for subscribing to and processing MQTT messages, and the mathjs library for performing trilateration calculations. The server also uses the Socket.io library to enable real-time communication between the server and the client-side web page. The location data is displayed on the web page using a simple user interface that includes a floor plan of the indoor environment and a marker indicating the estimated location of the SERB robotic platform. The web page is set to refresh every few seconds to ensure that the latest location data is displayed.

4. RESULTS AND DISCUSSIONS

The proposed system was implemented and tested in a real environment to evaluate its performance. The system consisted of three ESP32 modules, the SERB robot, and one DragonBoard. The ESP32 modules were configured to scan for the BLE beacon and send the received RSSI values to the DragonBoard via the MQTT broker.

The DragonBoard received the RSSI values from the ESP32 modules and calculated the corresponding distances using the RSSI-to-distance calculation method. The largest RSSI values from the nearest three ESP32 modules were used to calculate the coordinates of the beacon device using the trilateration method.

The ESP32 modules used in the experiment were placed at strategic locations in the indoor and outdoor environments. They were placed at a height of 1.5 meters above the ground to minimize the effects of ground reflections and other sources of interference. The BLE beacon device was placed at the SERB robot which moved through the environment each time its location was recorded. The experiment was conducted over 5 hours to collect a sufficient amount of data for analysis. Before experimenting, the system was calibrated using pre-calibration techniques to account for external factors that could affect RSSI values, such as signal attenuation due to walls and other obstacles, and electromagnetic interference from other devices. Additionally, filtering techniques were used to remove any outliers in the RSSI data that could affect the accuracy of the trilateration algorithm. These calibration and filtering techniques helped to improve the accuracy of the system's position estimates.

To implement the trilateration algorithm, the system selected the three closest ESP32 modules based on their received RSSI values. The coordinates of the beacon device were then calculated using the standard trilateration formula. This formula calculates the intersection of three spheres, each with a radius equal to the distance between the beacon device and the corresponding ESP32 module. The resulting point is the estimated position of the beacon device. The system also employed validation techniques to ensure the accuracy of the results. For example, the system compared the estimated position of the beacon device with the actual position of the robot and calculated the error margin. Additionally, the system tested the algorithm under different conditions, such as in various environments and with different types of obstacles, to ensure that the algorithm produced consistent results.

The network architecture used in the proposed system consists of an MQTT broker and the DragonBoard. The MQTT broker serves as a message broker that receives and distributes messages between the ESP32 modules and the DragonBoard. The DragonBoard subscribes to the specific topics on the broker, receives the

RSSI values from the ESP32 modules, and calculates the distances and coordinates of the BLE beacon device using trilateration. The MQTT broker used in the system was configured to support the MQTT protocol version 3.1.1 with a publish/subscribe communication model. The broker was deployed on a cloud server with a static IP address to ensure uninterrupted communication between the devices. The DragonBoard used a custom node.js server to receive the messages from the MQTT broker, process the data, and display the results on a web page. The server used the MQTT.js library to establish a connection with the broker and receive the RSSI values from the ESP32 modules. The messages were published to specific topics on the broker, which were subscribed to by the DragonBoard using the same MQTT.js library. This allowed for real-time communication and quick data processing.

The system was tested in a real environment with various obstacles and external factors that could affect the signal propagation. The results showed that the system was able to accurately estimate the position of the BLE beacon device with an error margin of 4.1% concerning the actual location of the robot, both in indoor and outdoor environments. It was observed that the magnitude of the errors depended to a large extent on the type of obstacles present in the environment.

The experiment requires a suitable environment to achieve reliable results. The following environmental conditions should be considered:

- **Absorption, interference, and diffraction**: RSSI values vary due to external factors, including absorption, interference, or diffraction. These factors affect the signal strength and result in an unstable RSSI value.
- **Signal propagation**: Signal propagation due to obstacles and other external factors adds another reason that causes errors in calculating RSSI-to-distance values. The experiment should be conducted in an open area with minimal obstructions to reduce signal propagation errors.
- **Limited bandwidth**: The MQTT protocol is designed for communication within a small area where a small code footprint is compiled, or when the network has limited bandwidth. Therefore, the experiment should be conducted in an area with limited network bandwidth to ensure smooth communication between the ESP32 modules and the DragonBoard.
- **Security**: The MQTT protocol sends connection credentials in plain text format and does not include any measures for security or authentication. Therefore, using TCP for data transmission to protect the integrity of transferred information from interception or duplication is recommended.

The size of the indoor and outdoor environments may affect the accuracy of the system, as the distance between the BLE beacon and the ESP32 modules could vary greatly depending on the size of the area. In larger environments, the coverage area of the ESP32 modules may be limited, which could result in a lower number of RSSI readings and a less accurate trilateration estimate. Similarly, the placement of the ESP32 modules could also affect the accuracy of the system, as modules placed in areas with more obstacles or interference could produce less reliable RSSI values. To improve the system's accuracy in larger environments, more ESP32 modules could be added to increase the coverage area, or a different localization algorithm could be used. It is important to note that the proposed system may have limitations in very large environments, and additional research may be necessary to determine its suitability for such applications.

These results demonstrate the feasibility of using the proposed system for tracking objects or individuals equipped with BLE beacon devices in indoor environments. The system has potential applications in various fields, such as asset tracking, indoor navigation, and security monitoring. Further development of the system could include improving the accuracy of the position estimates and expanding the coverage area by adding more ESP32 modules or optimizing the deployment of the modules.

The MQTT protocol used in the proposed system poses potential security and privacy risks as it sends connection credentials in plain text format, which can be intercepted or duplicated by unauthorized entities. To ensure the security of the system, the use of TCP for data transmission is recommended to protect the integrity of transferred information. Moreover, implementing authentication and encryption mechanisms such as SSL/TLS to secure the communication channel between the ESP32 modules and the DragonBoard can provide an extra layer of security.

While the proposed system demonstrated accurate BLE beacon tracking in a controlled environment, several limitations need to be addressed for its implementation in real-world scenarios. One of the significant limitations is the limited coverage area of the ESP32 modules, which can result in inaccurate distance estimation and position calculation. This limitation can be addressed by optimizing the placement of the ESP32 modules or using additional modules to increase the coverage area. Furthermore, the proposed system is designed to track a single BLE beacon device, and scaling up to track multiple devices may require modifications to the system architecture and algorithms. Another limitation is the sensitivity of the system to external factors such as interference, diffraction, and signal propagation due to obstacles, which can result in inaccurate RSSI-to-distance values. Incorporating pre-calibration or filtering techniques to account for external factors can help address this limitation. Overall, addressing these limitations can lead to further development of the proposed system for various applications such as indoor navigation, asset tracking, and security monitoring.

CONCLUSIONS

In conclusion, the study proposed a system that estimates the position of a small robot in both outdoor and indoor environments using a combination of RSSI-to-distance calculation, trilateration, and MQTT protocol for communication between ESP32 stations and the server. The hardware configuration consisted of three ESP32 modules, the SERB robot, and one DragonBoard. The system was implemented and tested in a real environment to evaluate its performance. The system was able to accurately estimate the position of the BLE beacon device with an error margin of 4.1% concerning the actual location of the robot.

The system's proposed solution addressed the limitations of using RSSI in localization, such as external factors influencing RSSI values as the distance of the beacon from ESP32 varies. The trilateration method was employed to determine the precise position of the beacon. The use of the MQTT protocol enabled communication between ESP32 stations and the server, allowing for a small code footprint and limited bandwidth network usage.

The results showed that the system's accuracy was maintained in different environments with various obstacles and external factors that could affect signal propagation. The proposed system can be used in various applications such as indoor navigation and asset tracking. Further improvements to the proposed system could include incorporating other signal measurement techniques such as Time of Arrival (ToA) or Time Difference of Arrival (TDoA) to increase the accuracy of the trilateration method. Also, including more ESP32 modules in the hardware configuration could enhance the system's accuracy by providing more data points for the trilateration algorithm.

ACKNOWLEDGEMENT

This work was supported by the Universidad Distrital Francisco José de Caldas, in part through CIDC, and partly by the Facultad Tecnológica. The views expressed in this paper are not necessarily endorsed by Universidad Distrital. The authors thank the research group ARMOS for the evaluation carried out on prototypes of ideas and strategies.

REFERENCES

[1] Lars Kunze, Nick Hawes, Tom Duckett, Marc Hanheide and T. Krajnak. 2018. Artificial intelligence for long-term robot autonomy: A survey. arXiv: 1-8.

[2] Justinas Miseikis, Pietro Caroni, Patricia Duchamp, Alina Gasser, Rastislav Marko, Nelija Miseikiene, Frederik Zwilling, Charles de Castelbajac, Lucas Eicher, Michael Fruh and Hansruedi Fruh. 2020. Lio-a personal robot assistant for human-robot interaction and care applications. IEEE Robotics and Automation Letters, 5(4): 5339-5346. ISSN 2377-3766. doi:10.1109/lra.2020.3007462.

[3] Muhammad Fadhil Ginting, Kyohei Otsu, Jeffrey A. Edlund, Jay Gao and Ali-Akbar Agha-Mohammadi. 2021. CHORD: Distributed data-sharing via hybrid ROS 1 and 2 for multi-robot exploration of large-scale complex environments. IEEE Robotics and Automation Letters, 6(3): 5064-5071. ISSN 2377-3766. doi:10.1109/lra.2021.3061393.

[4] F. Martínez, E. Jacinto and F. Martínez. 2020. Obstacle detection for autonomous systems using stereoscopic images and bacterial behaviour. International Journal of Electrical and Computer Engineering, 10(2): 2164–2172. ISSN 2088-8708. doi:http://doi.org/10.11591/ijece.v10i2.pp2164-2172.

[5] Paula Louro, Manuela Vieira, Pedro Vieira, João Rodrigues and Mirtes de Lima. 2021. Geo-localization using indoor visible light communication. In: Optical Sensors 2021, edited by Robert A. Lieberman, Francesco Baldini and Jiri Homola. SPIE. doi:10.1117/12.2589477.

[6] Korhan Cengiz. 2021. Comprehensive analysis on least-squares lateration for indoor positioning systems. IEEE Internet of Things Journal, 8(4): 2842-2856. ISSN 2327-4662. doi:10.1109/jiot.2020.3020888.

[7] Danshi Sun, Erhu Wei, Zhuoxi Ma, Chenxi Wu and Shiyi Xu. 2021. Optimized CNNs to indoor localization through BLE sensors using improved PSO. Sensors, 21(6): 1995. ISSN 1424-8220. doi:10.3390/s21061995.

[8] Petros Spachos and Konstantinos Plataniotis. 2021. Ble beacons in the smart city: Applications, challenges, and research opportunities. arXiv: 1-6. doi:10.1109/IOTM.0001.1900073.

[9] F. Martinez, C. Hernandez and A. Rendon. 2020. Identifier of human emotions based on convolutional neural network for assistant robot. TELKOMNIKA (Telecommunication Computing Electronics and Control), 18(3): 1499-1504. ISSN 1693-6930. doi:10.12928/telkomnika.v18i3.14777.

[10] Noopur Tyagi, Jaiteg Singh and Saravjeet Singh. 2022. Review of indoor positioning system: Technologies and applications. In: 2022 International Conference on Data Analytics for Business and Industry (ICDABI). IEEE. doi:10.1109/icdabi56818.2022.10041504.

[11] Ali Raza, Lazar Lolic, Shahmir Akhter and Michael Liut. 2021. Comparing and evaluating indoor positioning techniques. In: 2021 International Conference on Indoor Positioning and Indoor

Navigation (IPIN). IEEE. doi:10.1109/ipin51156.2021.9662632.

[12] Manjarini Mallik, Ayan Kumar Panja and Chandreyee Chowdhury. 2023. Paving the way with machine learning for seamless indoor-outdoor positioning: A survey. Information Fusion, 94(1): 126-151. ISSN 1566-2535. doi:10.1016/j.inffus.2023.01.023.

[13] Pothuri Surendra Varma and Veena Anand. 2022. Intelligent scanning period dilation based wi-fi fingerprinting for energy efficient indoor positioning in IoT applications. The Journal of Supercomputing, 2022. ISSN 0920-8542. doi:10.1007/s11227-022-04980-9.

[14] Baoding Zhou, Zhiqian Wu, Zhipeng Chen, Xu Liu and Qingquan Li. 2023. Wi-fi RTT/encoder/INS-based robot indoor localization using smartphones. IEEE Transactions on Vehicular Technology, 2023(1): 1-13. ISSN 1939-9359. doi:10.1109/tvt.2023.3234283.

[15] Gunwoo Lee, Byeong-Cheol Moon, Sangjae Lee and Dongsoo Han. 2020. Fusion of the SLAM with wi-fi-based positioning methods for mobile robot-based learning data collection, localization, and tracking in indoor spaces. Sensors, 20(18): 5182. ISSN 1424-8220. doi:10.3390/s20185182.

[16] Ketan Ramaneti, Nikita Mohanty and Vinoth Babu Kumaravelu. 2021. IoT based 2d indoor navigation system using BLE beacons and dijkstra's algorithm. In: 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE. doi:10.1109/icccnt51525.2021.9580047.

[17] Mahdi Naimi Jamal, Tian Jin and Farnaz Sadeghpour. 2022. A web-based platform for real-time visualization of tof UWB tracking in construction sites. In: Lecture Notes in Civil Engineering, Springer Nature Singapore. 335-343. doi:10.1007/978-981-19-0968-9_27.

[18] Nicolas Lhomme, Benjamin E. Barrowes and David C. George. 2011. EMI sensor positioning using a beacon approach. In: SPIE Proceedings, edited by Russell S. Harmon, Jr. John H. Holloway and J. Thomas Broach. SPIE. doi:10.1117/12.886642.

[19] Yew Teck Tan, Rui Gao and Mandar Chitre. 2014. Cooperative path planning for range-only localization using a single moving beacon. IEEE Journal of Oceanic Engineering, 39(2): 371-385. ISSN 0364-9059. doi:10.1109/joe.2013.2296361.

[20] N. Magadevi and V. Jawahar Senthil Kumar. 2017. Energy efficient, obstacle avoidance path planning trajectory for localization in wireless sensor network. Cluster Computing, 22(S5): 10529-10535. ISSN 1386-7857. doi:10.1007/s10586-017-1098-7.

[21] Sourabh R. Misal, S. R. Prajwal, H. M. Niveditha, H. M. Vinayaka and S. Veena. 2020. Indoor positioning system (IPS) using ESP32, MQTT and bluetooth. In: 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC). IEEE. doi:10.1109/iccmc48092.2020.iccmc-00015.

[22] Thai-Mai Thi Dinh, Ngoc-Son Duong and Kumbesan Sandrasegaran. 2020. Smartphone-based indoor positioning using BLE iBeacon and reliable lightweight fingerprint map. IEEE Sensors Journal, 20(17): 10283-10294. ISSN 1530-437X. doi:10.1109/jsen.2020.2989411.

[23] Nikos Papadakis, Nikos Koukoulas, Ioannis Christakis, Ilias Stavrakas and Dionisis Kandris. 2021. An IoT-based participatory antitheft system for public safety enhancement in smart cities. Smart Cities, 4(2): 919-937. ISSN 2624-6511. doi:10.3390/smartcities4020047.

[24] Mohamed Zied Chaari, Rashid Al-Rahimi and Abdulaziz Aljaberi. 2021. Real-time monitoring of indoor healthcare tracking using the internet of things based IBeacon. In: Lecture Notes in Networks and Systems, Springer International Publishing. 332-342. doi:10.1007/978-3-030-82529-4_32.

[25] Alexandru-Tudor Popovici, Constantin-Catalin Dosoftei and Cristina Budaciu. 2022. Kinematics calibration and validation approach using indoor positioning system for an omnidirectional mobile robot. Sensors, 22(22): 8590. ISSN 1424-8220. doi:10.3390/s22228590.

[26] Thomas Yunck, William Melbourne and C. Thoenton. 1985. GPS-based satellite tracking system for precise positioning. IEEE Transactions on Geoscience and Remote Sensing, GE-23(4): 450-457. ISSN 0196-2892. doi:10.1109/tgrs.1985.289434.

[27] Jonathan W. Campbell. 2002. STARSAT: a joint NASA/AF project for laser calibration of small objects in space. In: SPIE Proceedings, edited by Claude R. Phipps. SPIE. doi:10.1117/12.482037.