A Cutting-Edge IoT-Based Framework to Boost Collaboration within Peer-to-Peer Networked Multi-Robot Healthcare Systems

Hayet Tlijani
Electrical Engineering Department, LARA: Automatic Research Laboratory, National Engineers School of Tunis, Tunisia
tlijanihayet@gmail.com (corresponding author)

Hani Nasri
LSA: Laboratory of Advanced Systems, Polytechnic School of Tunisia, Tunis, Tunisia
haninasri14@gmail.com

Khaled Nouri
LSA: Laboratory of Advanced Systems, Polytechnic School of Tunisia, Tunis, Tunisia
khalednouri@gmail.com

Received: 14 July 2025 | Revised: 5 August 2025 and 26 August 2025 | Accepted: 30 August 2025

 DOI: https://doi.org/10.48084/etasr.13992

ABSTRACT

This study introduces an innovative Internet of Mobile Robots (IoMR) framework specifically designed for healthcare applications, focusing on improved collaboration and efficiency among a network of Connected Mobile Robots (CMRs). The CMRs play a pivotal role in critical tasks, such as patient monitoring, medication delivery, and environmental condition assessment, driving significant improvements in care delivery in healthcare settings. Equipped with cutting-edge embedded systems and intelligent functionalities, each CMR interacts dynamically with its surroundings and with a centralized IoT platform. This architecture facilitates real-time data sharing and collaborative decision-making among robots, fostering an ecosystem of responsive healthcare delivery. The design incorporates essential layers: a robust physical connectivity layer, a service layer dedicated to efficient data processing, and a comprehensive management dashboard that empowers operators with actionable insights. The framework employs advanced communication protocols, such as MQTT, to ensure reliable communication between various devices and mitigate interoperability, security, and scalability challenges. An IoMR prototype encompasses multiple mobile robots coordinated by a central processing unit, focusing on automation and efficiency in critical healthcare tasks. This implementation demonstrates not only resource-efficient processing but also minimal latency in task execution. The proposed approach optimizes operational efficiency in healthcare environments. By combining autonomous robotics and IoT technology, the proposed IoMR framework represents a significant advancement in healthcare delivery systems, ultimately improving patient outcomes and redefining the future of health services.

Keywords-Internet of Mobile Robots (IoMR); Connected Mobile Robots (CMRs); healthcare applications; autonomous data collection; real-time processing

I. INTRODUCTION

The rapid evolution of robotics, coupled with the transformative potential of the Internet of Things (IoT), has led to groundbreaking applications in various sectors, with healthcare at the forefront. As the demand for efficient and effective patient care increases, the integration of Connected Mobile Robots (CMR) into healthcare settings emerges as a promising solution to tackle critical challenges, including resource allocation, patient monitoring, and timely interventions. Extensive research underscores the profound impact of robotics in healthcare, demonstrating their ability to enhance operational efficiency and significantly improve patient outcomes. Previous studies have deployed robotic systems in medication management, telematics, and surgical assistance, highlighting their versatility in increasing the capabilities of healthcare professionals. Robotic technology has the potential to dramatically increase patient care standards and clinical productivity and improve patient and healthcare worker safety [1]. In addition, the integration of robotic-care scenarios is increasingly revolutionizing healthcare practices [2].

Despite these advances, significant challenges persist in coordinating systems that facilitate the seamless operation of multiple heterogeneous robots within dynamic healthcare environments. To bridge this gap, the establishment of a connected heterogeneous ecosystem, such as the IoT, becomes paramount. Within the IoT ecosystem, intelligent and autonomous objects deploy wireless communication technologies to collect, analyze, process, and distribute critical information. This synergy cultivates continuous connectivity and communication, delivering customized services and real-time data to users across geographical boundaries, thus addressing their evolving needs [3]. The advent of Industry 4.0 signals a profound change in this digital landscape, where IoT, Artificial Intelligence (AI), intelligent devices, and networked systems converge to redefine operational efficiency [4]. The evolution of industrial systems has propelled the emergence of networked multi-robot systems, enabling autonomous mobile robots to interact within heterogeneous environments while maintaining seamless connectivity with remote platforms and services [5]. As these systems gain momentum, the urgency for innovative architectural models and platforms that facilitate the integration of diverse applications across connected devices becomes increasingly critical, particularly in overcoming challenges related to coordination and collaboration in healthcare robotics.

To address these challenges, this study proposes the Internet of Mobile Robots (IoMR) framework, designed to foster collaboration among CMRs in the execution of essential healthcare tasks. Using a robust IoT infrastructure, the IoMR framework empowers these robots to communicate, share data, and collaborate to respond to a variety of healthcare needs. This investigation aims to contribute to the advancement of architectural innovations that significantly improve collaboration and operational capabilities within increasingly complex robotic networks [6]. This study offers a comprehensive overview of the IoMR architecture, its operational capabilities, and its potential to revolutionize healthcare services. It also explores how the coordination of robotic systems within this framework can transform healthcare delivery, resulting in a more responsive and efficient patient care paradigm. The IoMR framework introduces several innovative elements within the IoT domain and networked mobile robot systems. A key innovation is the development of a five-layer architectural model that assembles the system into physical, connectivity, service, platform, and presentation layers. This modular architecture ensures scalability, real-time data flow, and seamless interoperability between connected mobile robots and cloud-based applications.

Previous studies have explored the multifaceted aspects of mobile robotics, IoT ecosystems, and their applications within various sectors. Numerous studies have investigated networked embedded systems, specifically highlighting the roles of cloud computing, edge computing, and the IoT [7, 8]. In [9], the synergies between IoT and mobile robotics in healthcare were emphasized, examining the underlying technologies and their deployment in smart health services. In [10], an IoT-based search and find application was presented, where a heterogeneous team of robots collaboratively executed coordinated search operations over designated areas.

Researchers have also explored energy-efficient communication protocols tailored for IoT applications in healthcare settings. In [11], an energy-efficient IoT e-health framework incorporated AI to enhance the maintainability of diagnostic systems and facilitate reliable communication through a medical cloud infrastructure. In [12], a collaborative mobile service robot was designed to assist with various activities within hospital facilities. This study provided valuable insights into robot applications and their potential to improve healthcare delivery. In [13], the optimization of energy consumption in IoT-enabled mobile robots was investigated, discussing methods to improve operational efficiency and sustainability within healthcare contexts. In [14], coordination strategies for collaborative mobile robots operating in smart environments were outlined.

II. PROPOSED IOT-BASED SOLUTION

The proposed framework is characterized by a network of interconnected mobile robots and terminals that operate within a wireless environmental context.

A. Framework Architecture

The proposed architectural model allows the IoMR to execute tasks efficiently in heterogeneous environments. The IoT ecosystem fundamentally relies on wireless sensor networks that collect data from individual sensors and relay this information to the IoT server [7]. In IoMR, CMRs similarly process data obtained from their sensors and transmit pertinent information to the server, thereby facilitating data sharing among various devices within the network [8]. This capability not only enhances collaborative problem-solving but also enables the robots to adapt and respond more effectively to dynamic environments. Embedded applications within the IoMR ecosystem are designed to monitor critical environmental parameters, such as temperature and humidity levels, utilizing wireless sensor networks.

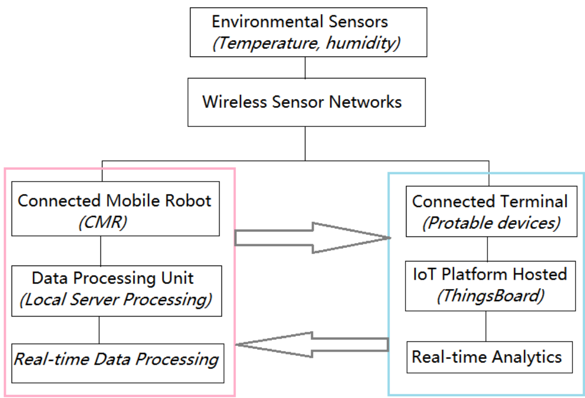

Fig. 1. Architecture of the IoMR framework.

Figure 1 illustrates the proposed IoMR framework. Each CMR operates as an embedded system, capable of autonomously interacting with the physical environment without the need for human intervention [7]. Within the IoT ecosystem, these CMRs serve as connected devices; however, they encounter several constraints, including limitations in memory, bandwidth, and energy consumption [14]. To ensure effective deployment, each CMR is tailored for specific tasks and equipped with intelligent capabilities that allow it to receive and transmit data using embedded smart sensors. Furthermore, each CMR is assigned distinct responsibilities and is linked to a centralized database on the IoT, which facilitates task sharing and information exchange between robots [8]. This connectivity significantly improves the performance of the multi-robot system, promoting improved collaboration and operational efficiency [15]. A CMR comprises three essential components:

- Data generation, storage, and transmission: This element encompasses the capabilities of the CMR to collect data from its environment and relay it to the network.
- Data processing algorithms: These algorithms may include machine learning and decision-making techniques that enable the CMR to analyze the data it collects.
- Ecosystem interaction: This component defines how the CMR engages with other connected devices within the network.

B. *Framework Embedded Applications and Computer Systems*

This study aimed to investigate comprehensive system settings and the development of various embedded applications and computer systems that collectively enhance the functionality of the IoMR ecosystem (Figure 2). The embedded applications developed include:

- Path Planning: This algorithm is designed to enable CMRs to compute optimal navigation routes while effectively accounting for environmental obstacles and dynamically changing conditions.
- Supervisory application: This provides real-time monitoring capabilities for each robotic unit, tracking critical parameters such as location, speed, and battery status, thus ensuring operational efficiency and facilitating timely interventions when necessary.
- Database management system: This is crucial for organizing and managing data generated within the IoMR framework, enabling efficient data retrieval and facilitating comprehensive analysis.

In addition to embedded applications, some computer systems were deployed within the IoMR ecosystem to enhance its operational capabilities:

- Terminal display: This serves as an information hub, providing real-time updates on the status of each CMR, which is essential for effective monitoring and management.
- Management dashboard: This dashboard provides an overarching view of the entire IoMR system, presenting critical metrics that allow administrators to manage operations effectively.

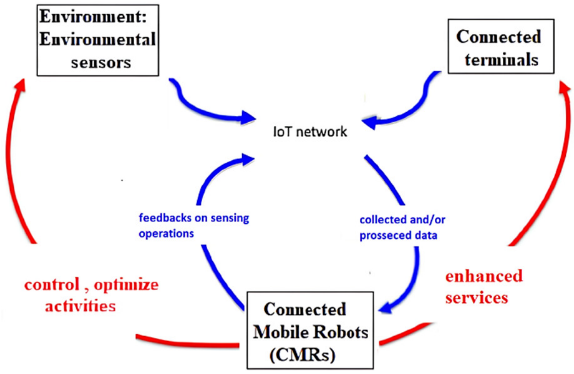

Fig. 2. IoMR framework settings.

III. IMPLEMENTATION OF THE PROPOSED IOMR FRAMEWORK

The implementation of the IoMR framework is essential for improving the coordination and efficiency of multi-robot systems.

A. *Five-Layered Architectural Model*

A five-layer architectural model is used to implement the proposed IoMR framework, as shown in Figure 3. This model consists of the following layers:

- Physical layer: This foundational layer consists of connected robots and wireless sensor networks. Components within this layer communicate via peer-to-peer connections, allowing data collected to be transmitted to the service layer for further analysis and processing.
- Connectivity layer and protocols: This layer features a broker that facilitates communication between connected devices within the IoMR ecosystem. It employs protocols that ensure the security, reliability, and scalability essential for the effective operation of multi-robot systems. This layer features a broker that facilitates communication between connected devices within the IoMR ecosystem and employs protocols that ensure the security, reliability, and scalability [16].
- Service layer: Functioning as the software backbone of the system, this layer is responsible for data processing and decision-making. It incorporates intelligent algorithms, including path planning and optimization techniques, to improve the operational efficiency of the multi-robot system. As highlighted in [10], the framework will also support services that allow a diverse group of robots to perform coordinated tasks on a unified platform.
- Platform layer: This layer manages various services distributed across the IoMR ecosystem. It also stores data generated by the framework, facilitating easy access and management.
- Presentation layer: This layer provides user-facing services derived from the platform layer. It is designed to enable the reuse of software components within the ecosystem for various stakeholders, including end-users such as web clients and decision-makers.

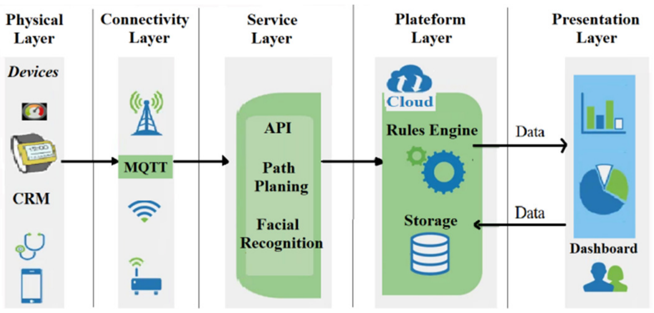

Fig. 3. Architecture of the proposed IoMR framework.

B. Challenges

The development of the IoMR presents several challenges that must be addressed to ensure effective operation and integration.

- Reliability: The methods employed for data collection and transfer significantly influence the quality of information. The MQTT protocol is used to ensure reliable data transmission between CMRs in the IoMR ecosystem. This approach ensures error correction through TCP/IP connections to the broker, facilitating the transmission and reception of packets (Tables V and VI). This solution supports a point-to-point messaging model, allowing the CMR sender to specify the intended receiver directly in a topic when sending a message. Thus, the CMR receiver obtains the message without the need to subscribe to the topic in advance, thereby reducing the subscription registration overhead.
- Interoperability: Implementing an IoT solution highlights the heterogeneity of connected devices. To establish a cohesive multi-robot system, specific technologies were selected to accommodate the diverse CMRs within the framework. In addition, these CMRs must communicate with other systems, including computers, software applications, and a centralized broker for data sharing. Despite continuous communication among CMRs, the varying types of devices require a robust interoperability framework.
- Security: At the transport level, communication is encrypted and identities are authenticated. The MQTT protocol provides client identifiers and user/password combinations to authenticate devices within the service layer of the IoMR ecosystem. These security features are inherent to the protocol for each device connected to the IoMR framework. Furthermore, encryption is employed to secure transmitted data within the service layer's operational codes.
- Scalability: The IoT solution for the multi-robot system integrates mobile robots, computers, and web software systems into a cohesive ecosystem. To facilitate scalability in scenarios with multiple dispersed objects, the MQTT protocol is employed alongside ThingsBoard as a middleware dashboard. Both technologies allow the IoMR ecosystem to expand the number of connected devices and enhance overall capabilities. Specifically, MQTT allows Machine-to-Machine (M2M) communication at any time and place, utilizing a messaging broker based on the Point-to-Point messaging model. The ThingsBoard platform offers management capabilities for newly connected devices.

IV. ENVIRONMENT AND HARDWARE OVERVIEW OF THE IOMR FRAMEWORK

The proposed IoMR framework was developed and physically deployed using a combination of Raspberry Pi-based robots, sensors, and embedded systems. Each robot performs a specific role, but is part of a unified collaborative system that communicates via a centralized Broker robot. To ensure efficient navigation within the hospital-like environment, a precalculated path optimization module based on a Genetic Algorithm (GA) was integrated into the Broker robot. This allows the system to compute optimal delivery and monitoring paths before execution, reducing collisions, travel time, and overall system latency. This system was not only simulated, but also physically implemented using the hardware components listed in Table I and tested in a controlled environment designed to mimic a hospital setting.

A. Hardware Components

The IoMR prototype was realized using a combination of carefully selected hardware components. The computing backbone consists of three Raspberry Pi units: a Raspberry Pi 4, deployed on the Broker robot to perform both coordination and GA-based path planning, a Raspberry Pi 3, integrated into the Technical robot, and a Raspberry Pi 2 coupled with a HealthyPi v4 shield, serving as the Nurse robot. For health monitoring, the system leverages the HealthyPi v4 platform, supported by an ESP32 microcontroller, allowing real-time acquisition of key physiological signals such as peripheral oxygen saturation (SpO_2), heart rate, and body temperature.

Environmental and contextual awareness are provided by specialized sensing devices. A DHT11 sensor continuously monitors room humidity and ambient temperature, while an MLX90614 infrared sensor offers high-precision thermal measurements particularly suited for surgical-room conditions. Image processing capabilities are incorporated through the Raspberry Pi Camera Module V2 (RPI-CAM-V2), mounted on the Technical robot, to support vision-based tasks such as recognition and navigation. Locomotion and control are achieved through G M20-180SH motors, which are driven by L298N dual H-bridge modules and L293D motor ICs. These are complemented by all-terrain tires that improve mobility and adaptability in indoor environments.

Stable and efficient power delivery is ensured using adjustable LM2596 DC–DC converter modules, capable of operating across a wide voltage range (3–40 V) to meet the requirements of diverse hardware subsystems. Finally, the system integrates peripheral devices, including relay modules, Wi-Fi communication modules, wireless keyboards, and display units, which together facilitate seamless interaction, connectivity, and real-time user feedback.

B. Task Distribution in the IoMR Framework

The proposed IoMR framework comprises a prototype system integrating three mobile robots (Table II) and a central PC. The framework is specifically designed to advance peer-to-peer collaboration in networked multi-robot systems, with a particular focus on applications in smart healthcare services. Each robot establishes wireless connectivity via a 2.4 GHz Wi-Fi module, while power requirements are met using 5V/12V DC converters tailored to the specifications of individual hardware modules.

C. Execution Phases

All components of the IoMR framework communicate through the MQTT protocol, with the Broker robot serving as the central hub. At system initialization, a GA-based path optimization module is executed on the Broker robot to compute the shortest collision-free trajectories for the Technical and Nurse robots. All devices run on Linux-based operating systems, including Raspbian and Ubuntu Core (for HealthyPi). The system evaluation was structured into two operational phases:

- Execution t1: System initialization encompassing boot-up, robot recognition, environment sensing, and GA-based path computation.
- Execution t2: Complete task execution, including vital-signs collection, facial recognition, item delivery, path following, and reporting.

TABLE I. IOMR ROBOTS AND CORE HARDWARE

Robot/component	Platform/device	Functionality/notes
Broker robot	Raspberry Pi 4	Connected via Wi-Fi. Powered via 5/12V DC converters.
Technical robot	Raspberry Pi 3	Uses DHT11 and MLX90614 sensors, Wi-Fi, 5V/12V DC power.
Nurse robot	Raspberry Pi 2 + HealthyPi v4	HealthyPi v4 with ESP32, Wi-Fi, 5V/12V DC power.
HealthyPi v4	ESP32-based module	Integrated with Raspberry Pi 2 (Nurse Robot) Captures vitals: SpO_2, heart rate, temperature.
DHT11 sensor	Digital temp and humidity sensor	Used on Technical robot. Monitors room environmental conditions
MLX90614 sensor	Infrared temperature sensor	Used on Nurse robot. Monitors patient's forehead temperature
Camera + recognition software	RPI-CAM-V2 + software	Used on Technical robot. Real-time face recognition.
Motors+Drivers	G M20-180SH, L298N, L293D	All-terrain tires. Integrated on mobile robots. Movement control.
Power Management	LM2596 adjustable DC-DC modules	3–40 V output range. Regulated power supply for all devices
Peripherals	Relay modules, Wi-Fi modules, wireless keyboards, displays.	Integrated across all robots. Support system interaction, mobility control, and communication.

TABLE II. IOMR ROBOTS AND TASK ASSIGNMENTS

Robot	Platform	Tasks Performed
Broker robot	Raspberry Pi 4	Handles all communication. Publishes/subscribes to MQTT topics. Runs GA-based path optimization to assign optimal routes to robots
Technical robot	Raspberry Pi 3	Face recognition (RPI-CAM-V2), scanner distribution, room humidity and temperature detection (DHT11, MLX90614)
Nurse robot	Raspberry Pi 2 + HealthyPi v4	Measures patient vitals (SpO_2, HR, temperature). Sends data to broker. Medication delivery logic.

V. EVALUATION AND DISCUSSION

The prototype of the innovative IoMR integrates three mobile robots and a PC, aimed specifically at enhancing collaboration in peer-to-peer networked multi-robot systems within the realm of smart health services. The proposed framework effectively addresses the inherent challenges associated with embedded systems, demonstrating efficient energy consumption and minimal processing time and memory usage, thus adhering to essential constraints related to time and memory complexity. The IoMR framework is designed to ensure continuous real-time interaction between mobile robots and their environment. A comprehensive wireless sensor network underpins perception and localization capabilities, while robust connectivity facilitates effective communication of computational resources and integration of CMRs. Furthermore, the software platform enhances accessibility to shared data and services between all connected devices, eliminating the reliance on computational resources through VMware, which is commonly required in cloud-based systems.

A. Energy Consumption

After completing 500 runs of the IoMR ecosystem, key metrics related to energy consumption were recorded (1). The curve in Figure 4 illustrates the energy usage over time, corresponding to these data, indicating an energy consumption of 90.5 watt-seconds over 16 minutes and 60 milliseconds. From this analysis, it is concluded that this ecosystem, powered by a 12V, 7Ah battery, can support data transmission for up to 23 hours, demonstrating the efficiency of the IoMR framework in enabling continuous operation in smart health applications. Moreover, applying the mean function (2) to the density data further substantiates these findings, highlighting the system's robustness in real-world scenarios.

$$E(t) = \sum_{i=0}^{n} P_i \times \Delta t_i \quad (1)$$

where $E(t)$ is the accumulated energy at time t, $P_i = V_i \times I_i$ is the instantaneous power at timestamp t_i, and Δt_i is the time interval between consecutive readings.

$$E_{avg} = \frac{\sum E_i}{N} \quad (2)$$

where E_i is an individual energy consumption measurement, and N is the total number of measurements over run times.

TABLE III. DATA SAMPLE FROM 500 RUNS

Time (s)	Energy (J)	Voltage (V)	Current (A)	Power (W)
0	0	3.587	0.3	1.0761
1	0.001104	3.587	0.3	1.0761
2	0.004385	3.587	0.3	1.0761
3	0.009844	3.587	0.3	1.0761
....	....	....	....	....
497	269.13949	3.587	0.3	1.0761
498	270.223391	3.587	0.3	1.0761
499	271.309353	3.587	0.3	1.0761

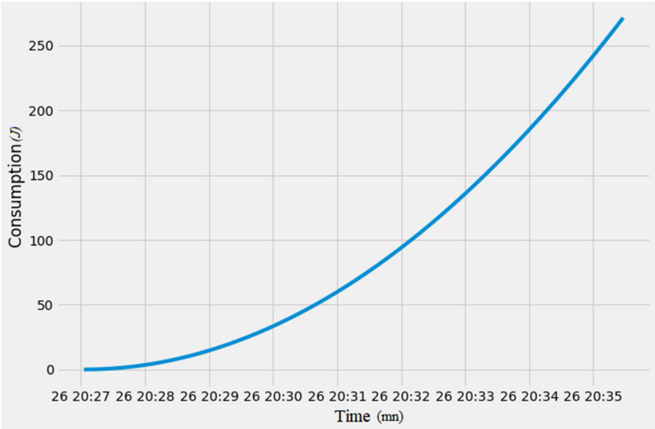

Fig. 4. Energy consumption over 500 runs.

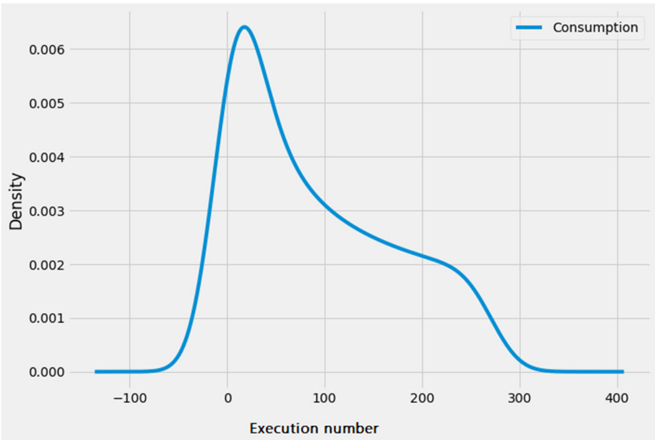

Fig. 5. Distribution of energy consumption over 500 runs.

B. Distribution of Energy Consumption for a Mobile Robot in the IoMR Framework

The Kernel Density Estimation (KDE) plot estimates the probability density function $\hat{f}(E)$ (3) of energy consumption.

$$\hat{f}(E) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{E-E_i}{h}\right) \quad (3)$$

where K(u) is a kernel function (usually Gaussian)

$$k(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2} \quad (4)$$

with $h \approx 1$ J and $n = 500$ data points.

The estimated density of the distribution of energy consumption offers significant insights into the operational efficiency of a mobile robot within the IoMR. The curve of Figure 5 is divided into three distinct parts, each representing different energy utilization patterns.

- Central peak: The high density around the zero-energy mark indicates that the robot predominantly operates at baseline energy levels, suggesting a high efficiency during standard tasks. This peak underscores the robot's optimization for minimal energy consumption in routine operations.
- Positive consumption values: As the analysis moves into the positive energy consumption range, there is a gradual decline in density. This trend suggests that elevated energy usage is infrequent, implying that the robot typically requires higher power only for specialized tasks (such as navigating complex environments or managing heavier loads).
- Negative consumption values: The presence of negative values reflects periods during which the robot enters low-power modes, such as idling or energy recovery through regenerative processes. This aspect points to sophisticated energy management strategies, showcasing the robot's ability to conserve power during less intensive operations.

C. Performance of the IoMR Framework

Table IV provides a structured overview for analyzing data related to computing devices, specifically various Raspberry Pi models, focusing on their CPU speeds, memory usage, and execution times. Figure 6 visually represents these metrics. The data encompasses the following metrics for each device: Device Identifier, CPU Speed (MHz), Available Space (MB), Memory Usage (a percentage of total memory), and Execution Time (ms).

TABLE IV. PERFORMANCE METRICS OF RASPBERRY PI MODELS

Devices	CPU speed (MHz)	Available space (MB)	Memory usage (%)	Execution time (ms)
Pi4	1400	40	3.2%	2
Pi3	1400	36	1.8%	2
Pi2	700	40	3.2%	1.6
HealthyPi	700	-	-	1

1) CPU Speed

An evaluation of the CPU performance of four computing devices, namely Pi4, Pi3, Pi2, and HealthyPi, reveals distinct operational capabilities based on their respective frequencies, as noticed in Figure 6(a). Pi4 and Pi3 operate at a CPU speed of 1400 MHz, which signifies their high-performance processing capabilities, making them well-suited for demanding applications. In contrast, Pi2 and HealthyPi function at a lower frequency of 700 MHz. Although this results in reduced processing power, it enhances energy efficiency for less resource-intensive tasks. This differentiation underscores the importance of aligning the selection of devices with the specific computational requirements of the intended applications. Thus, the proposed IoMR framework optimizes both performance and energy consumption.

2) *Free RAM*

The free RAM differs among the devices. In fact, Pi4 and Pi2 each offer 40 MB of free RAM, surpassing the 36 MB available in Pi3. The greater memory capacity of Pi2 and Pi4 is likely to enhance performance when managing larger datasets. This capability highlights how the IoMR framework architecture effectively supports scalability. In contrast, the slightly lower free RAM of Pi3 may impose limitations on its throughput capabilities for handling large-scale applications.

3) *Memory Usage*

Memory utilization among devices reflects the combined influence of computational workload and hardware capability. The Raspberry Pi 4 reported the highest load (3.2%), which is expected given its role in executing concurrent tasks, including the MQTT broker, GA-based path planning, and multi-threaded processes. In contrast, the Raspberry Pi 3 demonstrates considerably lower usage (1.8%) because its image recognition pipeline streams frames sequentially with minimal memory caching, thereby reducing RAM demand. Interestingly, the Raspberry Pi 2, although assigned relatively simple sensing tasks, exhibits the same utilization level as the Pi 4 (3.2%). This anomaly arises from its legacy hardware constraints combined with the inefficiencies of an older Raspbian build, which resulted in disproportionate memory consumption relative to its workload.

While this mixed-hardware configuration introduces variability in resource utilization, it also underscores the robustness of the IoMR framework. By distributing tasks according to device capabilities, the system demonstrates adaptability across heterogeneous computing environments. This characteristic mirrors realistic deployment scenarios in smart healthcare, where legacy and modern devices often coexist, and highlights the framework's potential for scalable and resource-aware integration.

4) *Execution Time*

Execution times across the IoMR robots are primarily determined by the complexity of their assigned tasks rather than by the raw CPU speed. The HealthyPi v4, part of the Nurse robot, achieves the fastest execution (1.0 ms) due to the bare-metal ESP32 implementation and minimal overhead when measuring patient vitals. The Raspberry Pi 2 in the Nurse robot completes its lightweight tasks—collecting SpO2, heart rate, and forehead temperature using the MLX90614 sensor, as well as handling medication delivery—in 1.6 ms despite its slower 700 MHz CPU. The Raspberry Pi 3 in the Technical robot requires slightly longer (2.0 ms) to process real-time face recognition via the RPI-CAM-V2 and to monitor environmental conditions with the DHT11 sensor. The Raspberry Pi 4 in the Broker robot also records 2.0 ms, reflecting its dual responsibilities of managing all MQTT communications and executing GA-based path planning for route optimization. These results indicate that execution time is largely dictated by task demands and system roles, rather than processor speed alone.

In summary, the efficiency of IoMR computing devices is influenced by task complexity, system role, memory usage, and execution times, rather than CPU speed alone. The Raspberry Pi 4 and Pi 3, with higher processing capabilities, perform well for demanding tasks such as GA-based path planning and real-time image recognition while efficiently managing memory. The Raspberry Pi 2, assigned lightweight Nurse robot tasks including vital-sign collection and medication delivery, completes its operations efficiently despite slower hardware. Meanwhile, the HealthyPi v4 achieves the fastest execution due to its optimized bare-metal design, making it particularly well-suited for lightweight embedded applications where rapid response is critical.

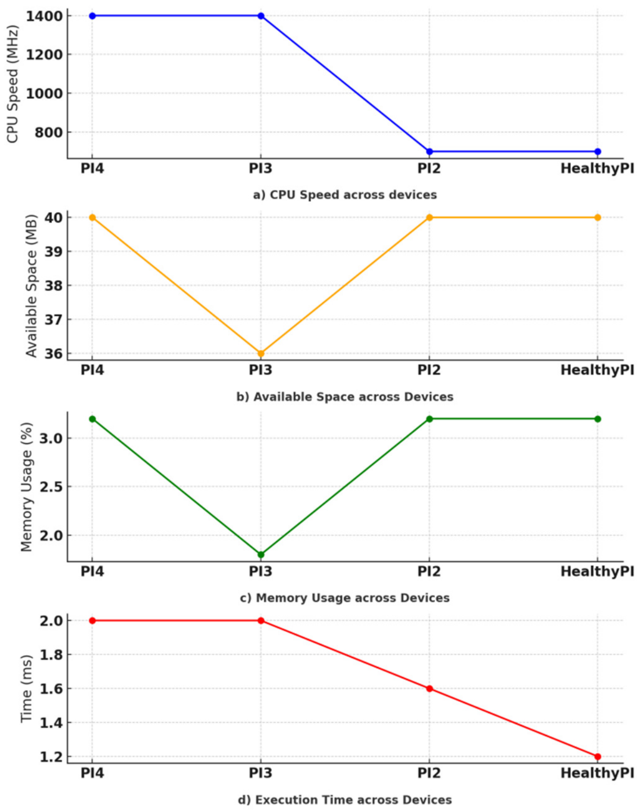

Fig. 6. Execution process of the IoMR framework

D. *Data Transfer*

Tables V and VI offer a structured data comparison of various robot operations, featuring key metrics such as CPU temperature, speed, voltage, packet reception, and transmission across two execution phases (t1 and t2). Differences in performance metrics between the two executions indicate varying workloads over time. This analysis underscores the distinct performance and operational requirements of robots within the hospital environment. Using the metrics in Figure 7, each robot's performance can be optimized, and the overall efficiency of robotic operations in healthcare can be improved. The execution metrics for Execution t1 in Table II and Execution t2 in Table III demonstrate the efficiency of data transfer between CMRs within the IoMR framework.

TABLE V. PERFORMANCE METRICS OF ROBOTS' EXECUTION T1

Robot	CPU temp. (°C)	CPU speed (MHz)	Packets received (MB)	Packets transferred (MB)
Broker	40	1400	1	10
Technical Department	55	1470	1.61	6
Nurse	40	700	1.56	5.11

TABLE VI. PERFORMANCE METRICS OF ROBOTS' EXECUTION T2

Robot	CPU temp (°C)	CPU speed (MHz)	Packets received (MB)	Packets transferred (MB)
Broker	50.5	1400	1.44	16.16
Robot Technical Department	56	1470	1.5	5.44
Robot Nurse	40.6	700	1.56	4.19

1) CPU Temperature

The CPU temperature monitoring provides insights into the operational efficiency of the robots under load conditions. The Broker robot demonstrated an increase in temperature from 40°C during Execution t1 to 50.5°C in Execution t2. This indicates a manageable thermal response to the increased data processing demands. Meanwhile, the Technical robot maintained a consistently higher temperature, ranging from 55°C to 56°C, while efficiently managing challenging tasks with negligible temperature fluctuations. This reflects a well-balanced workload. Additionally, the stable temperatures observed in the Nurse Robot underscore its reliability in resource use.

2) CPU Speed

The CPU speed remained remarkably consistent for all robots throughout both executions (Execution t1 and Execution t2), with no fluctuations in MHz. This stability suggests that the robots operate optimally within their designed parameters. In addition, they effectively process tasks without the need to overclock or reallocate additional resources. This efficiency indicates an optimized operational framework that allows each robot to perform at peak performance while minimizing excessive energy expenditure.

3) Packets Transferred

The notable variability in packets transferred highlights the relative efficiency of each robot in managing data outputs. The Broker robot significantly increased its output from 10 MB to 16.16 MB in Execution t2, indicating a highly effective data transfer capability and enhanced responsiveness to data demands. This improvement reinforces its role as a central hub within the IoMR framework. Conversely, the Technical robot experienced a slight decline in transferred packets, decreasing from 6 MB to 5.44 MB, while the Nurse robot saw a reduction from 5.11 MB to 4.19 MB. Although these decreases are minor, they underline issues that could potentially affect performance.

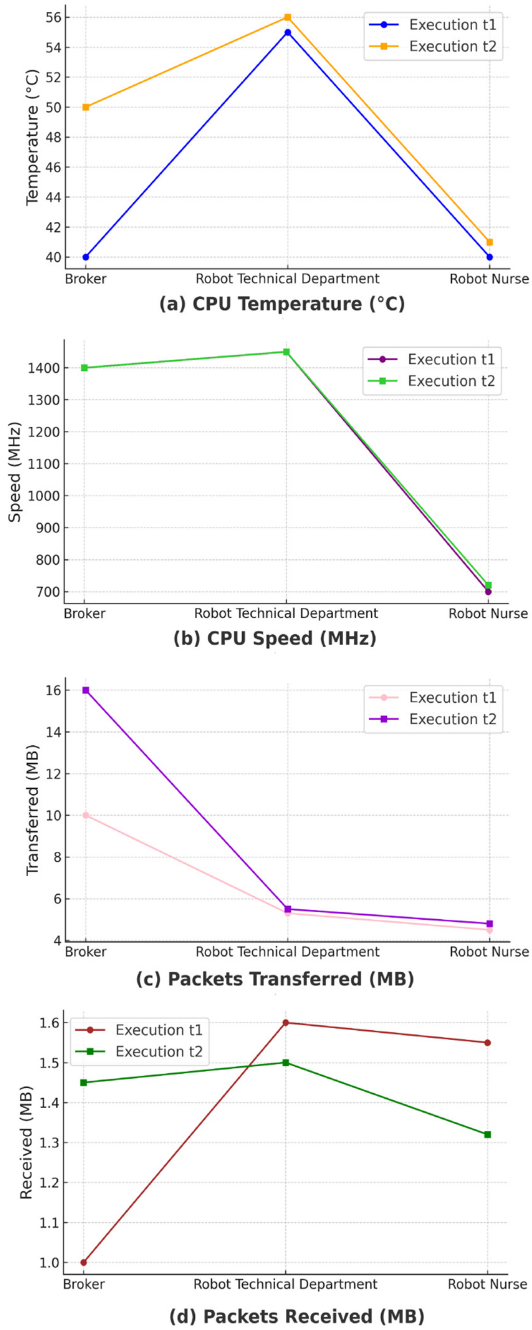

Fig. 7. Data transfer of the IoMR framework.

4) Packets Received

The received packets show notable efficiency in data handling, particularly with the Broker robot, which increased its reception from 1 MB to 1.44 MB in execution t2. This increase indicates improved communication protocols that enable the CMR to interact more efficiently with the network. The Technical robot proves its critical role as a data analyzer, as it handles higher packet volumes and processes substantial information without compromising performance. This confirms the effectiveness of the CMRs' design in the IoMR framework.

In practical terms, these observations highlight that the robotic system functions within an optimized architecture, effectively balancing performance and resource management. This balance is crucial for real-world applications in dynamic environments such as hospitals and technical departments. The focus on efficient processing capabilities not only enhances operational performance but also contributes to the long-term sustainability and reliability of the proposed IoMR framework.

VI. CONCLUSION

The proposed IoMR framework represents a significant advancement in IoT by unifying mobile robotics within a scalable, intelligent system. This research contributes a novel five-layer architectural model—comprising physical, connectivity, service, platform, and presentation layers—to enable seamless interoperability, real-time data exchange, and secure cloud-robot integration. Embedded applications, including adaptive path planning and supervisory monitoring, enhance robotic autonomy and operational efficiency, while computer-based systems, such as terminal displays and centralized dashboards, provide operators with real-time oversight. By leveraging broker-based connectivity and scalable protocols, the IoMR framework supports large-scale deployments, bridging embedded systems, intelligent control, and IoT networking to revolutionize robotic ecosystems. The IoMR framework, designed for healthcare applications, demonstrates superior performance in execution time, energy consumption, and memory usage compared to related works, leveraging its tailored architecture and optimized computing devices (Pi4, Pi3, Pi2, and HealthyPi).

The HealthyPi device excels with the fastest execution time, optimized for lightweight tasks critical to embedded healthcare applications such as real-time patient monitoring and medication delivery. The framework's service layer ensures efficient data processing, minimizing latency in collaborative decision-making among CMRs. The empirical data illustrated in section IV indicate robust performance, with HealthyPi outperforming other devices for time-sensitive tasks. In contrast, studies in [9, 11] focus on IoT-based healthcare monitoring, but do not provide specific execution time metrics, relying on general sensor data processing without optimized robotic integration. In [13], path optimization for mobile robots was addressed, but lacked execution time data for healthcare tasks. In [12], collaborative robots were discussed, but this work did not quantify processing speeds or latency, limiting direct comparison. The use of high-speed processors (Pi4, Pi3) for demanding tasks and HealthyPi for lightweight operations ensures faster execution than the less specialized systems in previous works.

The IoMR framework achieves remarkable energy efficiency. As mentioned previously, over 500 runs, the framework consumed 90.5 watt-seconds over 16 minutes and 60 milliseconds, allowing a 12V-7Ah battery to support continuous data transmission for up to 23 hours. This efficiency is critical for sustained healthcare operations, such as patient monitoring in smart environments. Pi2 balances speed and power for less intensive tasks, while HealthyPi minimizes energy use for lightweight applications. Comparatively, an energy-efficient IoT model for e-health used AI heuristics and homomorphic secret sharing in [11], but specific energy metrics (e.g., watt-seconds or operational duration) were absent, and robotic collaboration was out of focus. In [14], energy-efficient edge optimization was performed using graph theory, reporting improved throughput but no precise battery life or consumption data. In [8], the sustainability of cloud robotics was assessed, but it lacked healthcare-specific energy metrics. Other works, such as [10, 13, 17], do not provide detailed energy consumption data for healthcare settings. The quantified 23-hour continuous operation demonstrates the superior energy efficiency of IoMR compared to existing frameworks.

IoMR's computing devices demonstrate effective memory management, with Pi4 and Pi3 handling demanding applications through high memory capacity, while HealthyPi optimizes resource allocation for lightweight tasks, ensuring minimal memory overhead. The framework's service layer facilitates efficient data handling, reducing memory bottlenecks in real-time CMR coordination. Previous works provide limited insight into memory usage. In [9, 11], IoT platforms were utilized for healthcare data, but did not specify memory optimization strategies or metrics. In [12], the focus was on collaborative robots, but no details were provided on memory management. In [14], lightweight authentication for IoT could potentially reduce memory demands, but it lacked explicit metrics. In [16], multi-robot communication was discussed without addressing memory usage. IoMR's tailored device configurations provide a clear advantage in managing memory effectively for healthcare applications.

The proposed IoMR framework outperforms related works in execution time, energy consumption, and memory usage due to its specialized architecture and optimized devices. HealthyPi's fast execution, 23-hour battery life from a 12V-7Ah battery, and efficient memory allocation across Pi4, Pi3, Pi2, and HealthyPi enable robust, low-latency, and resource-efficient healthcare delivery. Prior works, such as [8, 9, 11, 12, 13, 14, 15, 17], either lack healthcare-specific robotic integration, detailed performance metrics, or both, making IoMR a pioneering solution for smart health applications.

REFERENCES

[1] D. Silvera-Tawil, "Robotics in Healthcare: A Survey," *SN Computer Science*, vol. 5, no. 1, Jan. 2024, Art. no. 189, https://doi.org/10.1007/s42979-023-02551-0.

[2] A. Maibaum, A. Bischof, J. Hergesell, and B. Lipp, "A critique of robotics in health care," *AI & Society*, vol. 37, no. 2, pp. 467–477, Jun. 2022, https://doi.org/10.1007/s00146-021-01206-z.

[3] V. Bhardwaj, A. Anooja, L. S. Vermani, Sunita, and B. K. Dhaliwal, "Smart cities and the IoT: an in-depth analysis of global research trends and future directions," *Discover Internet of Things*, vol. 4, no. 1, Oct. 2024, Art. no. 19, https://doi.org/10.1007/s43926-024-00076-3.

[4] R. Chataut, A. Phoummalayvane, and R. Akl, "Unleashing the Power of IoT: A Comprehensive Review of IoT Applications and Future Prospects in Healthcare, Agriculture, Smart Homes, Smart Cities, and Industry 4.0," *Sensors*, vol. 23, no. 16, Aug. 2023, Art. no. 7194, https://doi.org/10.3390/s23167194.

[5] H. Kabir, M. L. Tham, and Y. C. Chang, "Internet of robotic things for mobile robots: Concepts, technologies, challenges, applications, and future directions," *Digital Communications and Networks*, vol. 9, no. 6, pp. 1265–1290, Dec. 2023, https://doi.org/10.1016/j.dcan.2023.05.006.

[6] R. R. Shamshiri, E. Navas, V. Dworak, T. Schütte, C. Weltzien, and F. A. A. Cheein, "Internet of robotic things with a local LoRa network for teleoperation of an agricultural mobile robot using a digital shadow," *Discover Applied Sciences*, vol. 6, no. 8, Jul. 2024, Art. no. 414, https://doi.org/10.1007/s42452-024-06106-7.

[7] A. M. Al Shahrani, M. A. Alomar, K. N. Alqahtani, M. S. Basingab, B. Sharma, and A. Rizwan, "Machine Learning-Enabled Smart Industrial Automation Systems Using Internet of Things," *Sensors*, vol. 23, no. 1, Dec. 2022, Art. no. 324, https://doi.org/10.3390/s23010324.

[8] D. R. K. Sharma, R. K. Saxena, S. Kushwaha, B. Partap, and S. Sharma, "An Internet of Data Center Robotic Things for Cloud Robotics: A Concept of Data Center for a Green Environment," *Journal of Science and Technology Paper*, vol. 9, no. 10, 2024.

[9] A. Kumar, D. Dixit, A. Kumar, and A. Yadav, "Importance and use of IOT in Healthcare," *International Journal of Computer Applications*, vol. 175, no. 24, pp. 9–15, Oct. 2020, https://doi.org/10.5120/ijca2020920768.

[10] A. Barnawi, "An Advanced Search and Find System (ASAFS) on IoT-Based Mobile Autonomous Unmanned Vehicle Testbed (MAUVET)," *Arabian Journal for Science and Engineering*, vol. 45, no. 4, pp. 3273–3287, Apr. 2020, https://doi.org/10.1007/s13369-020-04364-2.

[11] A. Rehman, T. Saba, K. Haseeb, S. Larabi Marie-Sainte, and J. Lloret, "Energy-Efficient IoT e-Health Using Artificial Intelligence Model with Homomorphic Secret Sharing," *Energies*, vol. 14, no. 19, Jan. 2021, Art. no. 6414, https://doi.org/10.3390/en14196414.

[12] N. Valchkova, R. Zahariev, C. Totomirova, G. Angelov, Y. Paunski, and V. Cvetkov, "Approach to Designing a Mobile Service Collaborative Robot for Healthcare and Disadvantaged People," in *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Tenerife, Spain, Jul. 2023, pp. 1–6, https://doi.org/10.1109/ICECCME57830.2023.10252774.

[13] Y. Cao, K. Ni, T. Kawaguchi, and S. Hashimoto, "Path Following for Autonomous Mobile Robots with Deep Reinforcement Learning," *Sensors*, vol. 24, no. 2, Jan. 2024, Art. no. 561, https://doi.org/10.3390/s24020561.

[14] T. Saba, A. Rehman, K. Haseeb, S. A. Bahaj, and G. Jeon, "Energy-Efficient Edge Optimization Embedded System Using Graph Theory with 2-Tiered Security," *Electronics*, vol. 11, no. 18, Sep. 2022, Art. no. 2942, https://doi.org/10.3390/electronics11182942.

[15] J. Liu, "Multi agent systems: Studying coordination and cooperation mechanisms in multi-agent systems to achieve collective goals efficiently," *Journal of Artificial Intelligence Research*, vol. 4, no. 1, pp. 30–43, 2024.

[16] J. Gielis, A. Shankar, and A. Prorok, "A Critical Review of Communications in Multi-robot Systems," *Current Robotics Reports*, vol. 3, no. 4, pp. 213–225, Aug. 2022, https://doi.org/10.1007/s43154-022-00090-9.

[17] T. N. Pham and D. T. Mai, "A Mobile Robot Design for Home Security Systems," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 14882–14887, Aug. 2024, https://doi.org/10.48084/etasr.7336.