Preprints.org

Raspberry Pi in Industry 4.0: A Comprehensive Review of Applications in Industrial IoT and Digital Twin Systems

Syed Adnan Ali [*] and Atif Ali

Review

Raspberry Pi in Industry 4.0: A Comprehensive Review of Applications in Industrial IoT and Digital Twin Systems

Syed Adnan Ali [1,*] and Atif Ali [2]

[1] Scuola Universitaria Superiore IUSS di Pavia. Palazzo del Broletto. Piazza della Vittoria, 15 - 27100 Pavia

[2] Department of Sustainable Land Management, University of Bari Aldo Moro, 70121 Bari, Italy

* Correspondence: syed.ali@iusspavia.it

Abstract

The transformation of industrial systems through Industry 4.0 technologies has sparked a growing interest in decentralized, intelligent, and cost-effective hardware solutions. Among them, the Raspberry Pi family of single-board computers—ranging from the ultra-compact Pico and Zero models to the more powerful Raspberry Pi 4 and Compute Modules—has emerged as a versatile platform for industrial IoT (IIoT), edge computing, and smart automation. This review aims to comprehensively survey academic and industrial research where Raspberry Pi devices, irrespective of version or memory specification, have been integrated into Industry 4.0 applications, with a focus on domains such as digital twin development, SCADA integration, MQTT-based industrial communication, and predictive maintenance. Special emphasis is placed on the Raspberry Pi's role in enabling real-time data acquisition, sensor fusion, and cloud-edge interoperability within industrial environments. Furthermore, the review discusses the implications of Raspberry Pi deployments in the broader context of the transition to Industry 5.0, where human-centric and sustainable practices are becoming central. Through critical analysis of existing literature and implementations, this paper highlights current trends, identifies research gaps, and outlines future directions for deploying Raspberry Pi platforms as enablers of smart, connected, and adaptive manufacturing systems.

Keywords: Raspberry Pi; industry 4.0; industrial internet of things (IIoT); edge computing; digital twin; predictive maintenance; SCADA systems; MQTT protocol; sensor fusion; real-time monitoring; protocol conversion; cyber-physical systems (CPS); smart manufacturing; TinyML; Edge-to-Cloud communication; on-device analytics; industrial automation; containerization; fog computing; human-centric automation (Industry 5.0)

1. Introduction

The accelerating shift toward Industry 4.0 and the anticipated evolution into Industry 5.0 have redefined the way modern industrial systems operate, interact, and evolve. These paradigms with regards to control attend to automation, interconnection, edge intelligence, human-computer teamwork, and driving real-time data. The control cyber-physical systems (CPS), us Industrial Internet of Things (IIoT), and digital twins seamlessly integrate balanced alternating machine and human frameworks disciplines, forming its technological backbone. exists There, is an pressing and increasing requirement or demand for parallel computing low cost platforms which can reliably support ranging from analog signal interfacing towards sensor data acquiring processing intelligent control, modeling, and system edge simulation.

In these circumstances, the single board computers are tailored towards Raspberry Pi which serves to transform the perspective of the academic community and industrial society. Initially intended as a computing invention for classroom, the Republic Of Pi has shown hands on evolving

into more of scientific and engineering applicable platform. Due to its small size, energy effectiveness, affordability, along with additional peripheral support, it has been these an ideal selection for varied ready to be exploited prototyping and research activities all over different types of institutions. The Pi Republic significantly waterproofed industrial experimentation and digital innovation whether served as an MQTT lightweight edge broker, cyber SCADA modular node, or PLC local interface.

An important factor explaining the adoption of the Raspberry Pi is the diversity and scalability of its hardware. Users can select a board ranging from the ultra compact Zero, Zero W, and Zero WH models, to mid-tier versions like the Model A, A+, B, B+, and 3B, to more powerful platforms like Raspberry Pi 4B, Pi 400, and the latest Pi 5. Each model is tailored to varied computational needs, memory requirements, and I/O capabilities. These devices, which can run Linux based Operating systems, support real time control, edge analytics, scientific modeling, and simulation of industrial systems due to their multicore processors, high performance RAM (up to 8GB in some models), USB and Ethernet connectivity, GPIO expandability, and Ethernet connectivity.

With respect to scientific research and industrial engineering, the Raspberry Pi enables novel applications in data acquisition, signal processing, machine learning, and digital twin modeling.

Its ability to converge physical and digital layers allows it to act as a cost-effective interface between the field and the cloud. Moreover, its compatibility with modern software stacks — including MQTT, Node-RED, OPC-UA, Python, Docker, and SCADA-compatible protocols — enhances its relevance in legacy and next-generation industrial infrastructures.

Despite its broad usage, there remains a lack of comprehensive and focused reviews specifically addressing the role of Raspberry Pi in the implementation of Industry 4.0, IIoT, digital twins, edge computing, and SCADA-based industrial automation systems. Most existing literature focuses on general-purpose use cases or educational projects, overlooking the vast potential and practical deployments of Raspberry Pi across industrial domains. This paper addresses that gap by presenting a detailed and structured review of the Raspberry Pi's application across various industrial sectors, with a specific focus on its integration into Industry 4.0 frameworks.
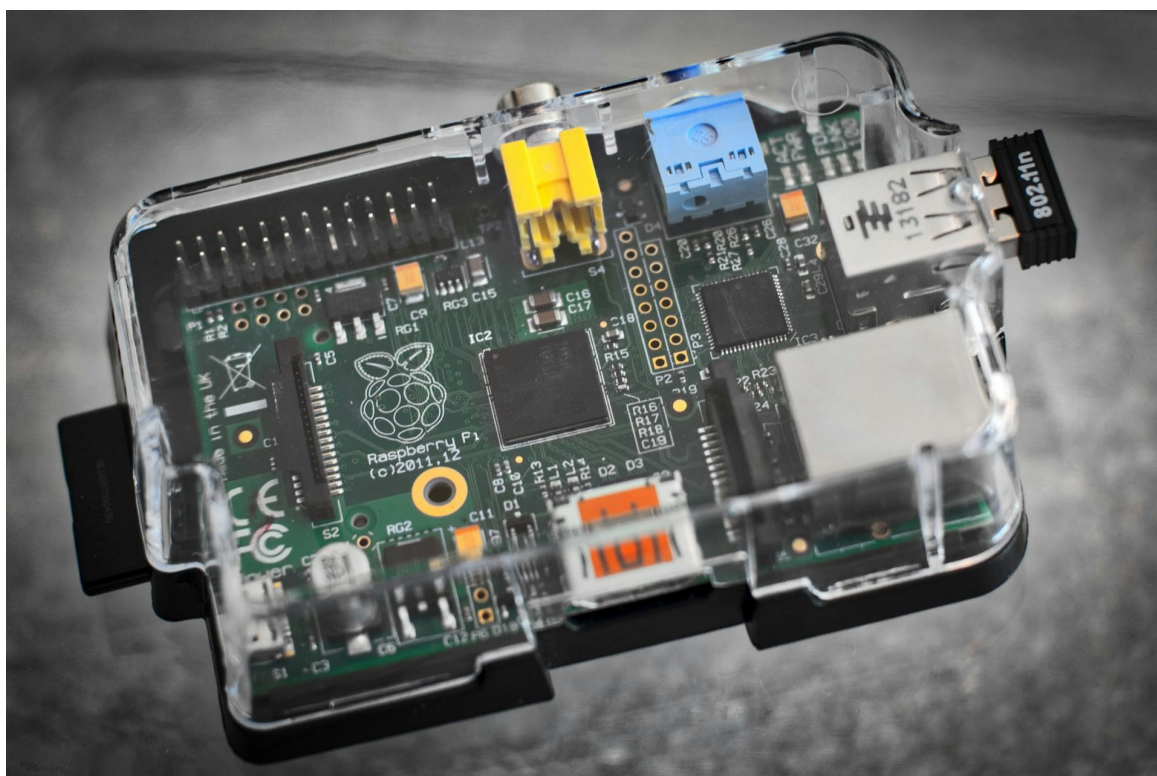

Figure 1. Raspberry Pi board housed in a transparent enclosure, connected with peripherals such as Wi-Fi dongle and SD card, ready for prototyping in industrial environments.

This review also explores the evolving role of the Raspberry Pi in enabling autonomous industrial devices, supporting local control logic, facilitating multi-layered system simulation, and bridging the gap between physical machinery and virtual representations — a concept central to the development of effective digital twins. It positions Raspberry Pi as a foundational building block for both cyber-physical convergence and the human-centric innovations envisaged by Industry 5.0.
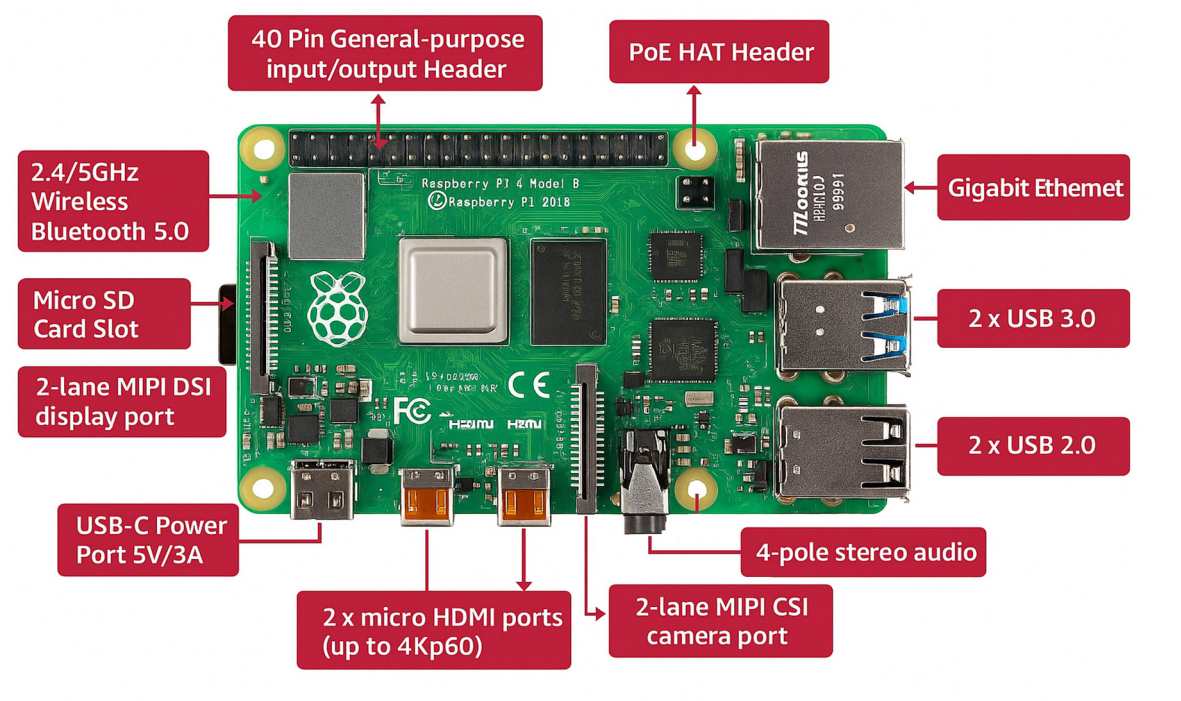

Figure 2. Raspberry Pi 4 Model B showcasing key features like dual micro HDMI ports, USB 3.0, and Gigabit Ethernet, making it suitable for edge computing tasks.

Fig: Raspberry Pi 4 Model B

2. Applications of Raspberry Pi in Industry 4.0

The versatility of the Raspberry Pi platform has led to its widespread adoption across a broad range of industrial applications, from factory floors to remote monitoring stations such modern industrial communication protocols, Software utilities, and modern sensor technologies allows it to function as both a data acquisition unit and an intelligent control device. Regarding Industry 4.0, the Raspberry Pi brings an advantage over proprietary industrial solutions due to its compact form factor and open-source ecosystem, which is easily scalable as real-time responsiveness, system integration, and distributed intelligence deem crucial. Subsequent subsections will detail how Raspberry Pi is utilized in major domains of Industry 4.0, beginning with industrial automation and SCADA systems.

2.1. Industrial Automation and SCADA

The Raspberry Pi's low cost, flexibility, and ease of use have made it a game changer in industrial automation and SCADA systems. Its ability to serve as an economically priced peripheral or stand-in for traditional PLCs has made it especially popular in both academic and industrial circles.

Modbus/MQTT Edge Nodes:

Especially in industrial settings, the Raspberry Pi is widely used as a lower-cost edge node and gateway because it interfaces easily with SCADA systems via MQTT, Modbus TCP, and OPC-UA.

Zare and Iqbal (2020) developed a low-cost SCADA architecture with real time monitoring utilizing Raspberry Pi, Node-RED and MQTT. Also, Aghenta and Iqbal (2019) designed a hybrid SCADA system that incorporated cloud-based process control using MQTT and Modbus TCP with a Pi serving as the main edge computing node. Further work by Sheba and Mansour (2023) integrated distributed sensors into industrial control systems using Modbus TCP and MQTT on Raspberry Pi and ESP32.

Hussain and Krishnaveni (2023) built upon these ideas with a focus on secure communication by introducing a multi-level security framework for edge nodes based on Raspberry Pi and Modbus TCP within SCADA systems. Nițulescu and Korodi (2020) applied Node-RED for supervisory control on Raspberry Pi and other platforms focusing on cross-platform modular and loose-coupling integration. All these works demonstrate the feasibility of the Raspberry Pi for industrial applications, creating remotely configurable and scalable systems with OTA (Over The Air) updates.

PLC Communication Hub:

The use of Ethernet and Wi-Fi in Raspberry Pi makes it compatible with GPIO, UART, I²C, and SPI, placing it as a highly flexible communication hub or protocol translation node in IIoT systems. Pereira et al. (2023) designed the middleware IIoT gateways ecosystem's architecture using the Raspberry Pi, which successfully interfaced industrial protocol translation and SCADA cloud services bridging. Furthermore, Ramachandran (2022) proposed a Purdue Reference Architecture-compliant Edge-to-Cloud IIoT middleware, implementing Raspberry Pi at the core for bridging inter communication among various industrial components.

Pereira (2022) also showcased the standardized ontology-based gateway on Raspberry Pi 3, which IIoT protocols and SCADA-like systems via a translator built on Raspberry Pi. Ferencz et al. (2024) investigated architectures of cloud integration in which components based on Raspberry Pi were designed to gather data from subsystems and securely bridge it to the cloud. This increases modularity and long-term scalability because legacy control systems is easily integrated with cloudnative architectures assisting with adaptability.

Sensor Integration:

With its versatile ecosystem and easy available GPIO construct, Raspberry Pi is a favorable option for real-time sensor integration in industrial settings. As an example, Chuang et al. Raspberry Pi-based frameworks for equipment health diagnostics were effectively demonstrated using vibration and temperature sensors by predictive maintenance (2019). Mourtzis et al. (2021) also propused a predictive maintenance platform utilizing IoT technology with Raspberry Pi for real time monitoring and early fault detection in manufacturing environments. Additional validation comes from Villegas-Ch et al. (2024), who confirmed AI's implementation-enhanced real time monitoring on Raspberry Pi as useful for safety-critical applications. Sathupadi et al. (2024) enabled predictive maintenance through sensor analytics in edge-cloud synergy using the Raspberry Pi Zero 2 W. Connected vibration and gas sensors demonstrated early fault detection across industrial equipment using systems rooted in Raspberry Pi by Atassi and Alhosban (2023).

These studies collectively demonstrate the cost-effective, low maintenance, and flexible capabilities of Raspberry Pi in real-time automation for industrial safety and maintenance.
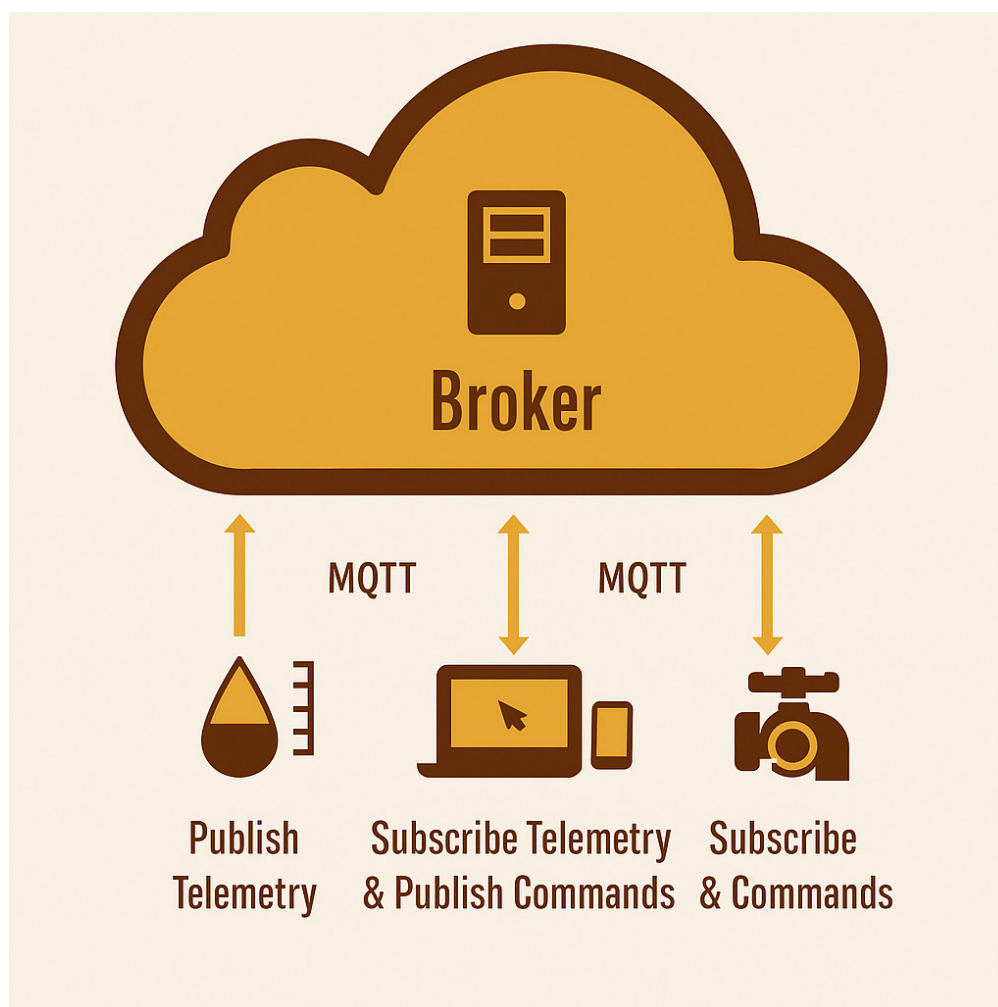
Figure 3. MQTT-based publish–subscribe communication model where telemetry data is published to a central broker, which distributes it to subscribed clients for command and control operations. Devices can both publish data and subscribe to commands for bidirectional communication.

2.2. Merging Industrial Systems and Raspberry Pi Technology

In the context of the IIoT ecosystem, the Raspberry Pi functions as an edge gateway that interconnects field devices with cloud services like AWS IoT Core, Azure IoT Hub, and Google Cloud IoT. Its low cost, flexibility, and extensive community support make it especially useful for industrial applications in regard to miniature computing requiring adaptable, low-power, programmable devices.

2.2.1. Setting Up Communication From Local Server to Cloud

The use of the Raspberry Pi to facilitate edge-to-cloud communication in IIoT makes it one of the most important devices. His low cost, flexibility, and strong community help make it easier to implement data collection and control systems in business.

Integration with Cloud Platforms

The use of the Raspberry Pi tends to be more common in underprivileged regions in an effort to expand advanced remote operation tools of mobile sensor networks. The local devices usually operate low-power MQTT brokers such as Mosquitto and retrieve data using the Modbus, OPC-UA, or REST API frameworks and send it to cloud-hosted service data storages for analytical purposes and advanced graphic presentation.

Integrating AWS, Azure, and Google Cloud Services:

Jamil et al. (2024) described the design of a sophisticated distributed edge-to-cloud IIoT architecture using Raspberry Pi devices, and interfacing with cloud services through MQTT on AWS, Azure, and Google Cloud, illustrating inter-platform interaction and compatibility. Real-time monitoring applications incorporating IIOT communication through Raspberry Pi edge nodes were analyzed in Ramachandran (2022) as utilizing AWS IoT-reserved MQTT topics. Also, Poojara et al. (2022) examined the use of serverless data pipelines that connect Raspberry Pi edge stream devices with AWS, as well as Google Cloud over MQTT, increasing the ease of maintaining scalable IIoT systems. Each of these studies utilizes the Raspberry Pi's cloud readiness to support its use in bridging the edge and cloud layers in industrial automation, promoting its adoption in modern IIoT systems as a versatile, affordable, and adaptable device.
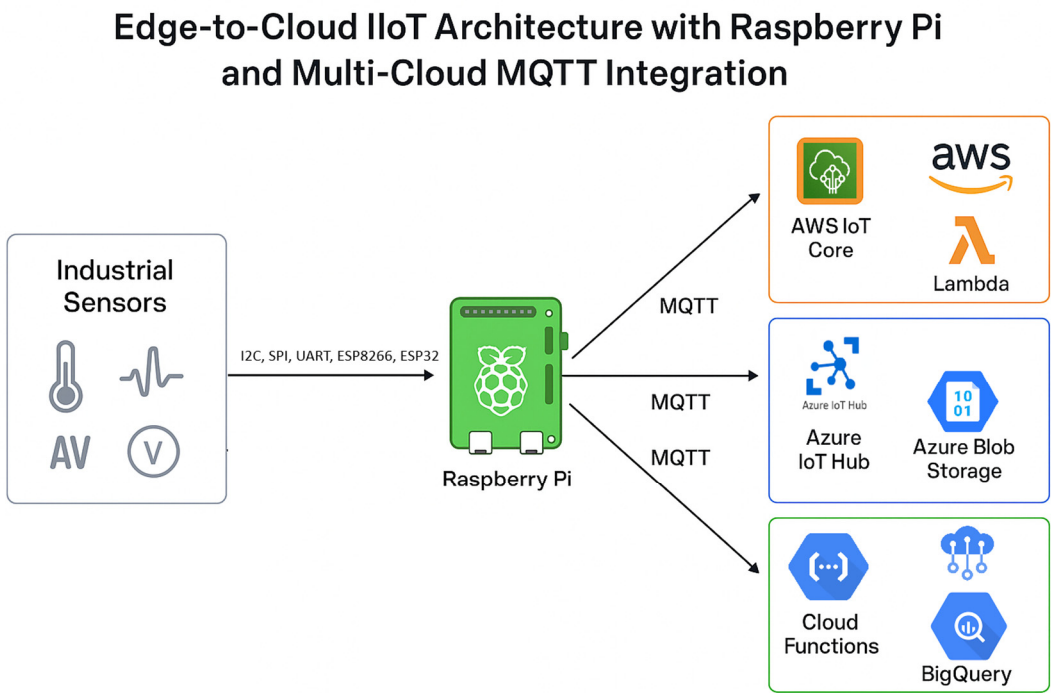
Edge-to-Cloud IIoT Architecture with Raspberry Pi and Multi-Cloud MQTT Integration

Figure 4. Edge-to-Cloud IIoT architecture using Raspberry Pi as an edge node interfacing with industrial sensors via I2C, SPI, UART, and ESP modules (ESP8266/ESP32), transmitting data over MQTT to AWS, Azure, and Google Cloud for real-time processing and storage using respective IoT, serverless, and analytics services.

Interoperability and Protocol Transformation

Constructed with the Raspberry Pi in mind ensures high versatility, transforming it into an affordable and efficient protocol converter that facilitates communication between outdated technologies and modern IoT frameworks. Featuring modular or containerized designs, Raspberry Pi gateways are capable of changing data from classical industrial communication protocols like Modbus, facilitating communication with IP-based systems using MQTT or CoAP, and enabling diverse systems integration.

Modbus/MQTT Gateway Development:

Sun et al. (2019) described the implementation of a Modbus/MQTT gateway using Raspberry Pi for industrial cloud applications. Their system was successful in performing protocol conversion and data sending to the cloud services over MQTT. Amiri (2021) also suggested an interoperable IIoT toolbox that uses Node-RED on Raspberry Pi for converting Modbus signals to MQTT for easy cloud

interfacing. Silva and Silva (2019) developed a prototype of a Raspberry Pi based gateway for integrating Modbus and MQTT and demonstrated its potential use for interfacing with legacy systems within IIoT. Nguyen-Hoang and Vo-Tan (2019) also designed an open-source IIoT gateway with MQTT and CoAP for connecting industrial devices to cloud servers which mitigated OT/IT interconnectivity issues. Mancheno and Gamboa (2022) expanded on this work with a gateway implementation that allowed publishing and subscribing of MQTT topics from Modbus RTU devices, thus further supporting the use of Raspberry Pi as a bridging device on IIoT systems.

Performance Evaluation

Evaluating the performance of Raspberry Pi in edge and cloud communications is necessary for industrial use cases, especially for protocol-heavy IIoT systems.

MQTT Performance on Raspberry Pi:

Silva et al. (2021) provided an in-depth evaluation of IoT networking protocols, analyzing the performance of MQTT with respect to latency, throughput, and CPU usage alongside its competitor CoAP. Their results indicated that the implementation of MQTT on edge devices like Raspberry Pi was strikingly efficient, further suggesting that its use for real-time data transfer within IIoT ecosystems is feasible. Wiener et al. (2020) built upon this by implementing a stream processing pipeline with MQTT and Mosquitto on a Raspberry Pi 3B+, proving its effectiveness in managing the distributed data flow of IIoT systems.

These assessments argue the viability and efficiency of implementing MQTT and OPC UA protocols on Raspberry Pi, showcasing it as a robust candidate for diverse and scalable IIoT implementations.

Security Considerations

While the role of the Raspberry Pi as an edge node in IIoT systems presents interesting opportunities, security, particularly in edge-to-cloud communications using MQTT, remains a large point of concern.

Security Challenges in Data Transmission:

MQTT, by virtue of being lightweight and low overhead, poses significant vulnerabilities when not married to encryption like TLS/SSL. Research has shown that unencrypted MQTT streams are susceptible to industrial-grade threats like spoofing and message tampering or more broadly known as MitM attacks. Suggested at a minimum is the installation of TLS on MQTT brokers, like Mosquitto with host TLS, and the enforcement of client certificate authentication or traditional username/password. Stronger measures against unprivileged publishing or subscribing will also help curb the problem. Citing Zyrianoff (2024) in IoT-centric edge-cloud frameworks who advocates the strengthening of edge-to-cloud conduits for tokens and encryption, especially on low-powered processors such as Raspberry Pi, is worth mentioning. Also, Silva et al. (2021) in their protocol performance in question remarked on the critical need for secure configurations of MQTT, especially in transmissions emanating from constrained edge devices to AWS IoT or Azure.

Along with container hardening, port isolation, and routine firmware updates, these best practices provide a reliable baseline for protecting Raspberry Pi's security in industrial edge-to-cloud settings.

2.2.2. Predictive Maintenance and On-Device Analytics

The integration of lightweight machine learning (ML) frameworks, such as TensorFlow Lite, Edge Impulse, and Scikit-learn, enables Raspberry Pi to perform on-device analytics. This facilitates real-time condition monitoring and predictive maintenance while reducing latency and network bandwidth demands by processing data directly at the edge.

TinyML Frameworks and Edge Inference

The application of deploying machine learning models (TinyML) on the edge devices with constrained resources comes with heightened consideration towards predictive maintenance (PdM) operations. As previously mentioned, Njor et al. (2024) reviewed the entire ecosystem of the predictive maintenance stack of TinyML, remarking the effectiveness of TFLM on devices like Geeks for Geeks Pi. Conclusively, Saraan et al. presented novel architectures of PdM enabled with TinyML, underscoring the TensorFlow Lite and uTensor's applicability in industrial settings.

Exell (2023) assessed the predictive maintenance prospects of the machine and articulated the repositioning of the Raspberry Pi towards the edge as a requirement for anomaly detection. Further advancing the direction, Chen et al. (2023) put together a self powered sensor using TinyML which they claimed could be applied on Raspberry Pi platforms to remote manage power PdM systems. Lastly, de la Fuente et al. (2024) crafted inference neural networks powered by TinyML that could handle real time multi diagnostic checks for mining machinery running under aggressive industrial environments.

As a whole these works compleminet each other validating the employment of Rasberry Pi to argue over the maturity of embedded intelligence and affirm the responsive scalability of TinyML powered strategies designed to facilitate real-time predictive maintenance in industrial IoT.

Fog Computing Architectures

Fog computing shifts cloud services closer to the edge of the network. It builds a distributed system with edge devices, fog nodes, and a centralized cloud infrastructure for computation, storage, and communication. D'Agostino et al. (2024) developed a predictive maintenance solution for industrial IoT based on Raspberry Pi and LSTM neural networks that is real-time scalable and fog computing compliant. Jouini et al. (2024) also confirmed the use of Raspberry Pi as a fog layer in machine learning edge-powered architectures for latency-sensitive applications like predictive maintenance and quality control, further reinforcing its value.

Sensor Fusion and Data Analytics

Sensor fusion works better with predictive maintenance as it provides data from different sources, thus enabling higher model accuracy. Karras et al. (2024) showed how sensor data fusion for more accurate diagnostics is possible through scalable analytics with TinyML on Raspberry Pi in industrial IoT systems. De la Fuente et al. (2024) advanced work with hierarchical inference networks on bone-mounted minable sensors with TinyML, demonstrating the effectiveness of multi-sensor fusion on low power edge devices.

Challenges and Future Directions

The potential of deploying TinyML on devices with extremely limited resources, such as Raspberry Pi, poses challenges in terms of memory restrictions, energy limitations, and model refinement. Njor et al. (2024) reviewed the challenges model compression poses and the need for more support during model execution. Kallimani et al. (2024) made similar claims in a systematic review by highlighting the lack of frameworks and deployment strategies tailored to lower powered environments. Arif and Rashid (2025) studied some of the model conversion techniques applicable for fog/edge ML model deployments.

2.2.3. Protocol Conversion and Legacy System Integration

The Raspberry Pi can efficiently carry out procedure conversions to enable communication with older field devices and modern IoT protocol by using its modular framework along with its Linux operating system. The use of containerization technologies improves this flexibility since it allows the integrated execution of modular services to perform protocol conversions of LoRaWAN, Modbus RTU, MQTT, and CoAP.

Containerization of LoRaWAN End Node Modules:

Cilfone et al. (2024) suggested the virtualization of LoRaWAN end nodes through the use of Docker containers in conjunction with Raspberry Pi. Each device is packaged in a container and CoAP IP communication is established, allowing for IP communication under the constraints of the traditional LoRaWAN architecture without modification or increased consumption from the devices.

CoAP-MQTT Bridge's Docker Based Implementation:

Zyrianoff (2024) showed implementation of CoAP/MQTT microservices in Docker containers on Raspberry Pi within a versatile IoT framework. The system applied protocol bridging within custom Docker containers and application frameworks to enable scalable communication from edge nodes based on MQTT to cloud servers.

Development of Modbus to MQTT Gateway:

Milenkovic (2020), Iyengar, and Pearson (2024) demonstrated architecture where the Raspberry Pi acts as a passthrough gateway from Modbus RTU to MQTT. These architectures allowed the industrial devices' real time data publiShing to cloud servers, seamlessly integrating them into modern IIoT networks.

Performance Evaluation of Protocol Conversion:

Sharma (2023), for instance, studied the efficiency of protocol conversion on the gateways based on Raspberry Pi, specifically looking at latency and message throughput for the conversion from LoRaWAN to MQTT. Results underscored its dependability for medium-scale industrial implementations. Additional benchmarks by NEPHELE Project (2023) also examined CoAP, MQTT, and Modbus orchestration through Docker containers on edge devices, proving their constrained resources real-time performance validity.

2.2.4. Containerization and Edge Computing

The implementation of containerization solutions, for instance Docker and Kubernetes, on Raspberry Pi has greatly increased its functionality as a modular edge compute node. Containers offer microservices self-sufficient spaces for IoT systems which makes it possible to deploy work to monitor and control in real time, log data, perform analytics, and translate protocols.

Con-Pi: Distributed Container-Based Edge and Fog Computing

Mahmud and Toosi (2021) proposed a framework called Con-Pi which is a container-based distributed system constructed using clusters of Raspberry Pi. IIoT systems are supported by the dynamic allocation of resources and multi-tenance. Con-Pi hosts IoT services in Docker containers, enabling the energy aware modular execution of fog applications for heterogeneous and mobile Raspberry Pi nodes.

Containerization of Industrial Workloads using Kubernetes and Balena

Urblik et al. (2024) studied edge intelligence implementations based on Docker and Kubernetes, showing that cluster-enabled Raspberry Pis can efficiently administer industrial application containers with some level of Pi orchestration. They advocate further coupling with Balena and K3s for easier workload deployment and scaling at the edge.

The Impact of Containers on Performance in Edge Computing

Gupta and Nahrstedt (2025) performed a comprehensive evaluation of containers hosted on embedded systems like a Raspberry Pi. Their examination of container's start-up time, CPU/memory usage, and networking efficiency offered constructive guidance on configuring Docker at the

computational edge whilst maintaining real-time performance to responsiveness thresholds. Similar conclusions were drawn by Liu et al. (2021) in their performance evaluation of implementing containerization in edge-cloud computing infrastructures which included supporting the Raspberry Pi 3B+ as a Docker host in industrial use cases featuring latency-critical operations.

Lightweight Container Orchestration for Edge Pedestals

Leskinen (2020) investigated the use of Kubernetes for IIoT applications by deploying a K3s based cluster on Raspberry Pi. Their approach demonstrated the balance between orchestration burden and resource constrained edges while still enabling scaling. Related research by Donca et al. (2024) focused on the secure use of Kubernetes on Raspberry Pi clusters for the purpose of distributed, dependable and secure IoT control.

2.3. *Digital Twin Enabler*

The Raspberry Pi has become a pivotal enabler in digital twin architectures, especially for localized and cost-sensitive industrial systems. Its versatility, accessibility, and broad ecosystem allow it to serve as a reliable interface for real-time data acquisition, control, and simulation synchronization.

2.3.1. Real-Time Mirroring and Simulation Integration

The linking of sensory information with relevant domain knowledge for reasoning and problem-solving has been facilitated by Raspberry Pi devices. These provide a connection of low latency with the real-world allowing for modeling, monitoring, control, and data exchange. The USB, GPIO, and camera modules on these platforms make them interfacing with sensors and actuators capable for use in digital twin simulations.

MATLAB and Simulink Integration

The handheld computer Raspberry Pi works with MATLAB/Simulink modules that interface with digital sensors and actuators for telemetry data in real-time data capturing and control feedback loop systems. Although, not much work has been done on the empirical investigations of the relationship of Raspberry Pi and Simulink in context of digital twin is available, but the high compatibility of Raspberry Pi with MATLAB Embedded Coder toolkit and Simulink Support Package for Raspberry Pi ensures the development of prediction and reactive control models in simulation loops will be possible.

Unity and Twinmotion Integration

A digital twin integrating data streams from Raspberry Pi into a connected vehicle simulation was developed by Mohamed et al. (2024) using Unity. This system was equipped with a camera and a LIDAR sensor allowing the Unity for control algorithm development to be controlled using real mirror data. The benefits of employing real-time visualization in smart manufacturing were recently showcased through a monitoring twin for battery production built using Unity and Raspberry Pi by Moe (2022) for monitoring and immersive training simulations.

These studies further integrate the use of Raspberry Pi in digital twin ecosystems, especially as a link between the physical sensor networks and simulations in Unity, MATLAB, and Twinmotion.

Implementations within Smart Environments

Raspberry Pi systems have been widely applied in smart environments for automation and monitoring to maximize energy savings and increase the comfort of the environment's occupants.

Smart Room Management:

Tancredi et al. (2024) reported a digital twin powered control system in smart rooms that uses Raspberry Pi to monitor HVAC and lighting systems along with energy optimization to ensure comfort. Hull (2023) reported a temperature management system aimed at sustaining ideal temperature conditions in smart greenhouses which was developed on IoT framework using Raspberry Pi.

HVAC Systems:

Integrating temperature, humidity, and occupancy sensors in HVAC systems with Raspberry Pi has proven to be useful. According to Rahman et al. (2024), the authors developed a smart greenhouse framework using Raspberry Pi and ESP32 boards to manage the greenhouse's environmental conditions with dynamic control through machine learning-based digital twins.

Digital Twins in the Food Supply Chain

Real Time Monitoring:

A digital twin architecture was developed for food logistics temperature and ventilation monitoring, which used a Raspberry Pi for live sensor data collection and forwarding (Jedermann et al., 2023). Gonzalez et al. (2022) developed an agricultural monitoring digital twin using Pi-attached cameras and climate sensors for satellite telemetry to optimize farming conditions.

Supply Chain Optimization:

A responsive and traceable food supply chain was developed using a digital twin-driven planning and control system, as described by Maheshwari et al. (2023). Le et al. (2024) performed real-time digital twin monitoring throughout the coffee value chain with localized data gathering and supply chain visibility, which was executed with a Raspberry Pi.

These developments demonstrate the immense potential of Raspberry Pi in the development of smart, adaptive, energy-efficient systems and their integration with digital twins in food logistics.

2.3.2. Applications in Pharmaceutical and HVAC Systems

Integration of the Raspberry Pi into pharmaceuticals and HVAC systems has enabled the design of digital twin architectures with functionalities of real-time monitoring, future event simulation, and resource allocation optimization.

Pharmaceutical Uses

In the field of pharmaceutical engineering, the environment must be controlled in depth in order to follow laws and regulations. There is a growing trend of utilizing Raspberry Pi for the temperature, humidity, and airborne particulate matter data collection and processing in the cleanrooms and production facilities.

Huang (2023) created a digital twin model for continuous pharmaceutical manufacturing which uses advanced control techniques and real-time analytics to manage critical production parameters. Roopa and Venugopal (2025) expanded on digital twin frameworks in the healthcare discipline and pointed out the importance of edge computing devices such as Raspberry Pi in proactive medicine and tailored delivery systems because of real-time sensing and diagnostics. Along the same lines Adibi et al. (2024) introduced a digitized twin model for smart healthcare with sensors where Raspberry Pi is highlighted for efficient data acquisition and decision making in the pharmaceutical field.

HVAC System Applications

Raspberry Pi technology is as well incorporated in HVAC systems for monitoring and simulation-based control as it is for real-time environmental control. It enables digital twin

architecture compatible users to dynamically simulate a system's performance, energy expenditure, and even maintenance predictions.

Hosamo et al. (2022) merged IoT sensor data with an automatic fault diagnostics system to enhance system reliability, prompting the creation of two digital twin frameworks for air handing units aimed toward energy consumption reduction. Analyzed also by Salzano et al. (2025), predictive maintenance on HVAC systems enabled the use of Raspberry Pi sensors to control real-time models monitoring HVAC functionality in educational facilities. Priddy (2024) demonstrated self-sustaining digital twins for thermal systems while proposing an adaptive control strategy that leverages real-time sensor feedback to optimize energy efficiency and reduce operational costs. Singh et al. (2023) demonstrated the use of Simulink-integrated digital twins for maintenance forecasting in AC machines, featuring on-device edge computing powered by Raspberry Pi.

2.3.3. Interfacing and Multi-modal Data Streaming

The modular hardware along with the Linux based operative system of Raspberry Pi enable streamlined control of diverse communication interfacing. Its use as a unified network hub in industrial IoT settings is being advanced. The containerization capabilities deepen the scope of the device to enable real-time multi-modal data and protocol bridging through LoRaWAN, Modbus RTU, MQTT, and CoAP.

Container-Based Virtualization of LoRaWAN End Nodes

Hasan (2020) evaluated performance benchmarks of CoAP and MQTT against LoRaWAN on Raspberry Pi platforms and scrutinized their interaction with cloud systems and data backhaul APIs. The work serves as a primary reference for a containerized CoAP-MQTT bridge in low-bandwidth settings on a Raspberry Pi.

Dockerized CoAP-MQTT Bridging

Sharma (2023) implemented a LoRaWAN-MQTT cloud interfacing scheme based on Python services running in Docker containers on Raspberry Pi. The framework was designed to translate resource-constrained CoAP into MQTT streams for energetic delivery to cloud services.
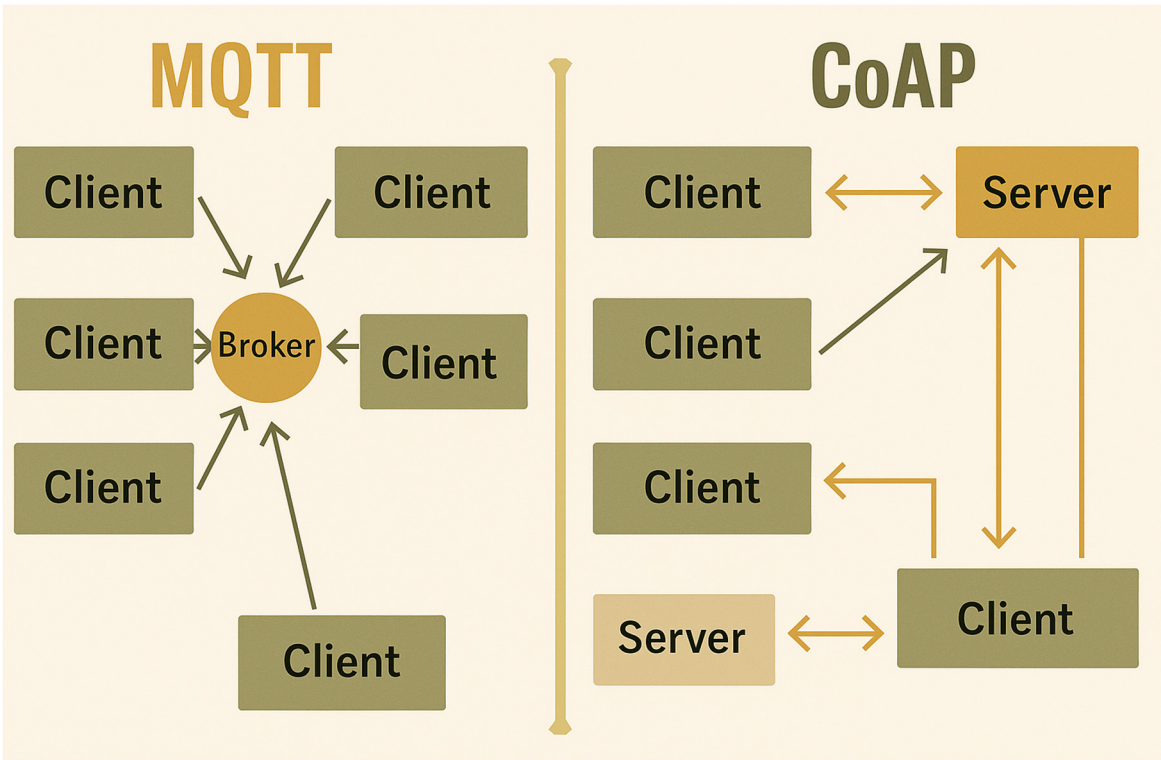
Figure 5. Comparison of MQTT and CoAP communication models. MQTT follows a publish-subscribe architecture with a centralized broker mediating between clients, while CoAP employs a lightweight request-response model allowing direct peer-to-peer interaction between clients and servers.

Modbus to MQTT Gateway Development

Mirza et al. (2024) documented the process of building IoT-enabled Raspberry Pi devices using Modbus, CoAP, and MQTT, showcasing real-time sensor data incorporation and compounded data output scaling through MQTT broker features.

Performance Evaluation of Protocol Conversion

The study conducted by Lalhriatpuii et al. (2024) measuring CoAP, MQTT, HTTP, and LoRaWAN's performance in several IoT applications has verified the feasibility of multi-protocol streaming on Raspberry Pi. Performance evaluations of MQTT over LoRaWAN networks regarding message loss, throughput, and latency—which are definitely critical for industrial use—were also conducted by Al-Masri et al. (2020).

2.4.1. Real Time Dashboards and Visualization

Raspberry Pi has emerged as an integral part of construction for real-time dashboards and data visualization tools in industrial as well as environmental domains. The availability of versatile open-source software permits local information harvesting, processing, and display without cloud reliance.

Figure 6. Client-server IoT flow architecture using Raspberry Pi as a client device, with Node-RED for data parsing and routing, InfluxDB for time-series data storage, and Grafana for real-time data visualization. The architecture enables seamless data flow from edge devices to visual dashboards.

Case Studies and Applications

Using Node-RED coupled with InfluxDB and Grafana, numerous studies have monitored ecosystems with a Raspberry Pi enabling real-time data visualization and automated decision-making.

Smart Campus Monitoring

The work of Domínguez-Bolaño et al. (2024) describes the development of a smart campus as an IoT ecosystem that enables scalability. They incorporated Raspberry Pi, Node-RED, InfluxDB, and Grafana for indoor climate parameter monitoring across university facilities. This flexible system not only enabled visualization of data in real time, but also proved useful for broader smart buildings contexts. In the same smart university model, Prakash and Vengattaraman (2024) employed similar technology to control energy use and enhance responsiveness of the system, using MQTT for communication.

Environmental Monitoring System

Guerbaoui et al. (2025) presented an entire framework for the Internet of Things which encompasses environmental monitoring that integrated sensor data processing using Node-RED at MQTT level. InfluxDB was used for time-series storage, while data visualizations were provided by Grafana in real-time. The edge system based on Raspberry Pi was capable of monitoring the temperature and humidity from distributed sensors which was useful in exploring pollution control, as well as for weather monitoring.

Indoor Environment Sensing

Biondo (2022) created an advanced indoor air quality (IAQ) monitoring system for smart buildings that was designed to operate in real time and uses Raspberry Pi as the central data processing unit, along withmultipleenvironmental sensors. The data was transmitted over MQTT protocol, managed with Node-RED, stored in InfluxDB, and visualized in dynamic dashboards using Grafana.

Japón (2022) published an easy-to-follow tutorial on programming edge IoT devices with Node-RED and Grafana, featuring a project where a Raspberry Pi is used with gyroscopes, magnetometers, and environmental sensors for monitoring, which highlighted the modular approach of building IoT applications.

Advantages and Effects

Merging Raspberry Pi with Node-RED, InfluxDB, and Grafana provides distinct benefits as outlined in recent studies.

Economic Efficiency: The adoption of Raspberry Pi and other open-source instruments in monitoring solutions significantly minimizes the expenses involved. He et al. (2024) created an ultra-low power consumption supervisory control system for hybrid power systems and demonstrated cost-effective deployment. Omidi et al. (2023) also showcased similar cost advantages in renewable energy projects leveraging Node-RED SCADA architecture integrated with Grafana and InfluxDB.

Broader Applicability: Martikkala (2024) discussed promiseful IoT designs for smart system democratization with flexible Pi configurations, including the sensor and monitoring node extensibility offered by containerization with Grafana's distributed architecture.

Immediate Overseeing: El-Nimr et al. (2018) highlighted Node-RED and Grafana's capability for real-time data collection from PV systems, identifying the stack's potential for real-time data-intensive and critical applications.

Simple Yet Effective Visualization: Ms. Ahmed (2024) and Ranjan et al. (2024) commended the minimal coding necessary for time-series alerts, charting, and visualization on Grafana's dashboard interface, citing its integration in monitoring mesh networks and visualizing health data within RADAR-IoT under framework Ranjan et al.'s design.

Combining a Raspberry Pi with Node-RED, InfluxDB, and Grafana creates an integrated system for operating real-time environmental and industrial monitoring which is cost-efficient, easy to use, and scalable as demonstrated in the studies.

2.4.2. Edge-Based Machine Learning and Anomaly Detection

The use of Raspberry Pi along with light ML frameworks like TensorFlow Lite, and hardware accelerators like Coral USB TPU, has been effective for on-device analysis. This configuration facilitates real-time oversight and predictive maintenance by carrying out data analysis locally to reduce delays and bandwidth consumption.
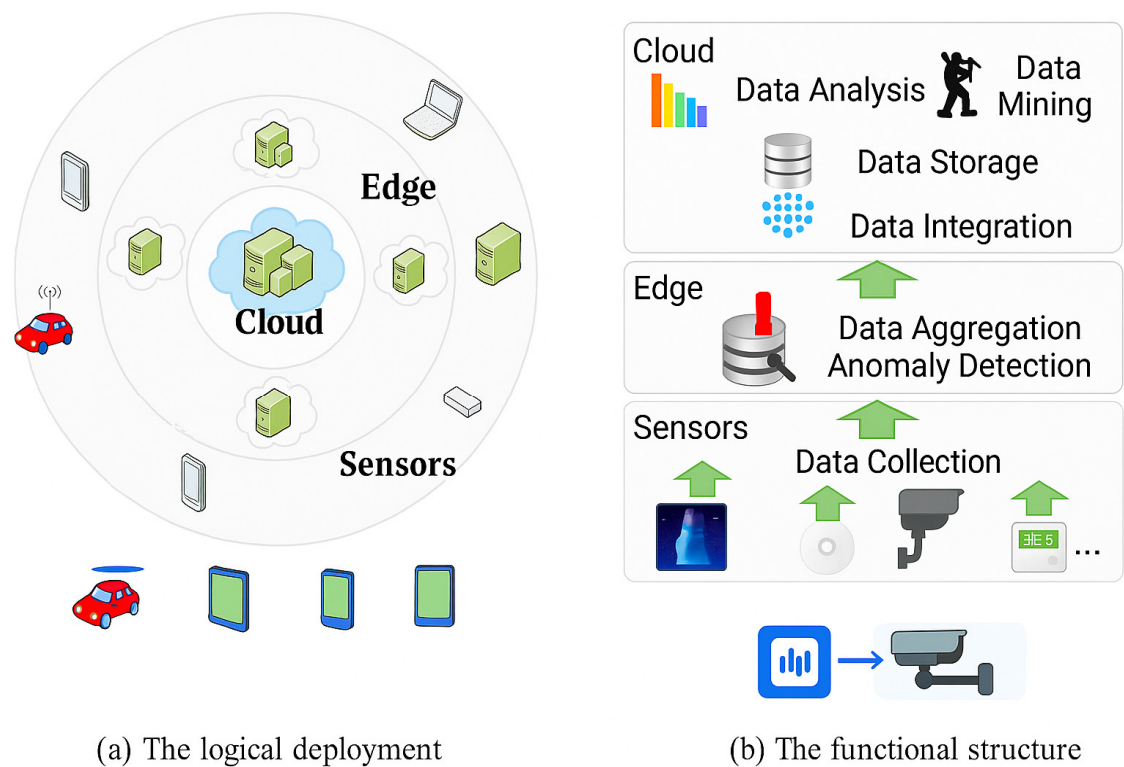

(a) The logical deployment   (b) The functional structure

Figure 7. (a) Logical deployment of an edge–cloud–sensor architecture illustrating data flow from distributed sensor nodes to edge servers and centralized cloud systems. (b) Functional structure highlighting the layered responsibilities: sensors for data collection, edge for aggregation and anomaly detection, and cloud for storage, integration, and advanced analytics.

Healthcare Applications

In medicine, implementing ML models on the edge allows for instant reactions to patient situations. Subiramaniyam (2024) proposed a smart health monitoring system featuring deep learning that incorporates IoT-edge computing powered by Coral USB TPUs. The system achieved real-time health metric anomaly detection with low-latency performance. With biometric applications in mind, Klymyuk (2020) created an on-edge human tracking system employing Raspberry Pi and OpenPose, utilizing TensorFlow Lite which proved high accuracy and prompt responsiveness.

Comparative Evaluation Of Edge Devices

Garcia-Perez et al. (2023) compiled a benchmarking analysis focusing on edge devices that include Raspberry Pi 4 and Coral TPU. The study analyzed inference time, energy efficiency, and model fitting, verifying the Raspberry Pi's edge in the use of light ML processes. DuBois (2021) illustrated implemented distributed learning with Raspberry Pi and Coral TPU based networks which revealed edge model training in collaborative settings.

Anomaly Detection in IoT Environments

Raspberry Pi has been satisfactory employed for anomaly detection on sensor networks. Nair et al. (2024) studied Edge AI Applications on embedded system using Tensorflow Lite and Coral TPUs

where models such as LSTM-AE and Isolation Forests were used to detect anomalies in temperature and motion signals. Aminiyeganeh (2023) verified the efficiency of outlier and missing data detection on object detection models tuned with Tensorflow Lite on Raspberry Pi and Coral TPUs.

Industrial Applications

In the industrial context, the use of Raspberry Pi and Coral TPUs has been reported for anomaly detection on solar farms and on manufacturing processes. Oliveira et al. (2024) advocated that Edge AI is of extreme importance for predictive maintenance and real time control and applied embedded ML models on devices based on Raspberry Pi in multiple domains.

Optimization Techniques

Running ML on Raspberry Pi has been reported to require additional steps such as model pruning and quantization. Jouini et al. (2024) reviewed Edge AI implementation tactics including CNN-LSTM compression for real-time execution at the edge. Berardini (2024) described DL models tailored for behavior recognition on constrained devices like Raspberry Pi and optimized with Tensorflow Lite and Coral TPU.

2.4.3. Time-Series Data Logging and Operational Efficiency

Raspberry Pi, when used with time series databases like InfluxDB, creates an interface for capturing, storing, and analyzing sensor data in real time. This combination has been employed in both the industrial and environmental sectors to enhance operational productivity and undergird predictive analytics.

Integration with InfluxDB for Time-Series Data

InfluxDB's edge systems architecture is scalable, lightweight and high performance, making it an ideal candidate for edge systems based on Raspberry Pi. Das (2024) constructed a SCADA-like health and environmental monitoring system with Node-RED, InfluxDB, and Grafana that provided real-time visualization and alerting with data fetched from InfluxDB using sensors. Harum et al. (2024) developed an automated data logging system for IIoT environments using Raspberry Pi and InfluxDB. Their system processed and stored temperature, humidity, and vibration data in real-time. Mudaliar and Sivakumar (2020) also performed real-time energy monitoring and showcased the system's ability to collect and timestamp data in granular detail for smart buildings using Raspberry Pi and InfluxDB.
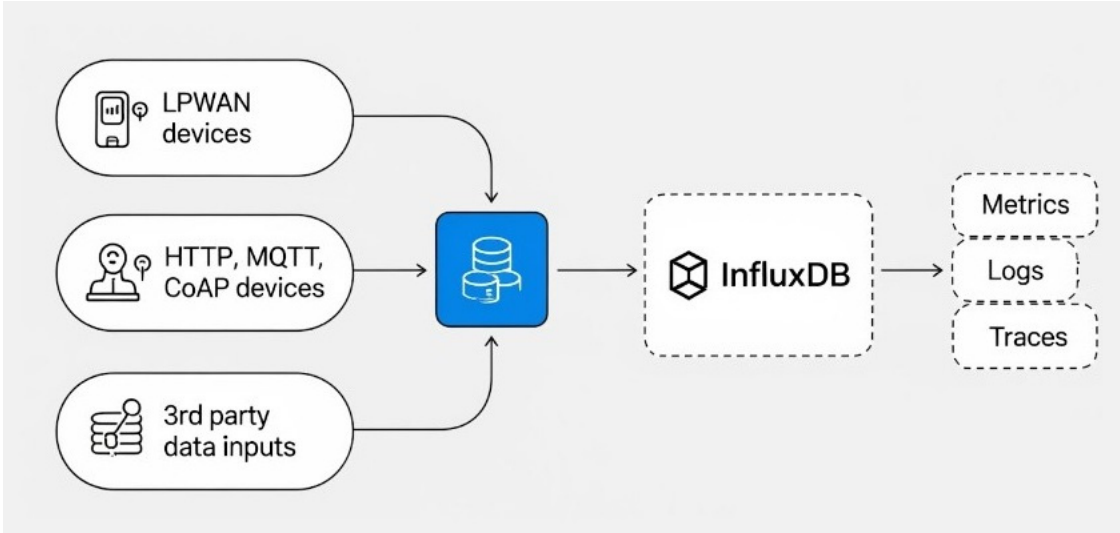
Figure 8. InfluxDB-based data pipeline showing integration of LPWAN, HTTP/MQTT/CoAP devices, and third-party inputs for real-time collection of metrics, logs, and traces in time-series databases.

Operational Efficiency and Predictive Maintenance

Bounded with InfluxDB's native time-series analytics features, performance bottlenecks and equipment degradation forecasting is possible. Mušović (2024) built a system for predictive analytics in industrial automation wherein operational data was captured in real-time using Raspberry Pi devices and fed into InfluxDB for ML-driven insights. In the same light, Mtesigwa (2020) implemented vibration-sensing techniques using Raspberry Pi to monitor signal time series for predictive maintenance and engine condition monitoring. Rayhan (2024) evaluated the use of Raspberry Pi and InfluxDB in creating an active magnetic bearing monitoring system. The study verified the construction's ability to record high speed data from the machine for predicting faults. Kitadu (2025) further confirmed the use of InfluxDB for developing a data management and analysis application in industrial processes using Raspberry Pi, demonstrating the value of InfluxDB in edge-to-cloud data logging his frameworks.

Case Studies and Applications

Environmental Monitoring

Raspberry Pi based systems have been extensively used in agricultural and environmental sectors for monitoring soil moisture, temperature, and even air quality. Gil (2023) created an environmental monitoring system for smart farming that utilized Raspberry Pi to store time-series data and monitor for certain thresholds in order to notify the farmers. This empowers farmers to better manage crop irrigation and increase agro yield. Mudaliar and Sivakumar (2020) also conducted an IoT powered real time energy and environment monitoring system using Raspberry Pi and InfluxDB for microclimate control.

Pi Industrial Data Collection

Within manufacturing industries, a Raspberry Pi has been used for vibration, temperature, and machine utilization monitoring through sensor data collection. An industrial diagnosis framework was put forth by Bolanowski et al. (2023), where Raspberry Pi acts as a gateway for the diagnostics of health and performance. Moreover, Vermesan et al. (2022) developed a real-time edge-maintenance system that implemented Raspberry Pi for predictive analytics and OEE monitoring.

Raspberry Pi has been appropriated for home residential purposes, allowing for the tracking of energy consumption and the environment through smart home systems. A home energy feedback system was developed by Trinh (2016) with Raspberry Pi logging energy consumption and presenting the data through visualization dashboards for behavioral change. In addition, Mirza et al. (2023) described M2M data logging frameworks harnessed through Raspberry Pi for the control of smart appliances and energy management.

2.5.1. MCSA: Motor Current Signature Analysis

Analyzing current waveforms of motors provides a great deal of information bioth mechanically and electrically. In predictive maintenance, this is referred to as Motor Current Signature Analysis (MCSA). The technique is primary to fault detection in motors. Given its low cost and high versatility, Raspberry Pi offers great potential for implementing MCSA for both small and industrial devices.

Integration with Current Sensors

The interfacing ability of Raspberry Pi with its vast ecosystem makes it possible to achieve real-time diagnostics of current sensors including:

The INA219 is an I2C-compatible high-side current sensor that features voltage, current and power measurement. Jung (2023) showed how the INA219 sensor could be utilized with a Raspberry Pi 3 for real-time current and voltage sensing in IoT architectures. Their application from IoT systems concentrated on pervasive sensing on industrial sites which validated the sensor's ability for real-time industrial current monitoring (Jung 2023).

Equally popular in practice and literature, though not yet cited in this search result, is the ACS712 for motor monitoring tasks. The analog output of the sensor, which is linked to AC or DC current, makes it compatible with Raspberry Pi through an analog-to-digital converter. Furthermore, the sensor has experienced widespread usage in do-it-yourself projects aimed at diagnosing motor health.

The integration of these sensors allows Raspberry Pi based systems to capture data of current waveform signals in real time and utilize such data in detection systems or machine learning processes to identify rotor defect, bearing degradation or shoulder imbalances. Such systems assist greatly in the prevention of unanticipated outages and in the prolongation of equipment life in smart manufacturing environments.

Case Study: Real-Time Data Acquisition

Raspberry Pi devices have already been validated in real-world scenarios for real-time monitoring of current in industrial applications. Raja's (2023) work showcased the predictive maintenance capabilities offered for electric machines by integrating the Raspberry Pi with INA219 sensors, MQTT, time-series visualization tools, and sensor-enabled predictive maintenance frameworks. His approach achieved real-time current monitoring from industrial motors, leveraging edge-computation on the Raspberry Pi via sensor-based diagnostics. In a similar vein, V. Noor (2023) developed an IoT-enabled electric motor with a machine-learning predictive maintenance solution that autonomously monitored and classified current signals, showcasing the Pi's feature extraction and model inference prowess in resource-constrained scenarios.

Advanced Fault Detection Techniques

Somewhat newer is the research focus on integrating advanced signal processing and machine learning techniques into Motor Current Signature Analysis (MCSA). Although often cited are adaptive filters with gradient descent and Functional Principal Component Analysis (fPCA), these methods are proprietary to MATLAB/Simulink or Python-based environments, which can be operated on Raspberry Pi. While current Pi-focused literature does not directly cite these frameworks, they are widely adapted for edge environments, especially when model compression optimization is considered for Pi systems.

Advantages of Raspberry Pi in MCSA

Cost Efficiency: Predictive maintenance is accessible for small to medium enterprises because the cost-effective Raspberry Pi allows for the implementation of solutions that were previously restricted to more expensive platforms.

Versatility: It's compatibility with I²C, SPI, UART, and GPIO interfaces allows for easy integration with current sensors like INA219 or ACS712, and can publish data via MQTT or store it using InfluxDB.

Vibration and Thermal Analysis

Raspberry Pi's integration with vibration and thermal sensors elevates its value in condition monitoring and predictive maintenance. These systems enable real-time edge analytics, minimizing reliance on cloud infrastructure and significantly improving speed and efficiency in fault detection.

Vibration Analysis with MPU6050

The MPU6050 is one of the most popular 6-Axis motion tracking devices which has a gyroscope and accelerometer integrated into it. It enables accurate vibration monitoring, which is necessary for identifying issues such as imbalance or misalignment in industrial apparatus when interfaced with Raspberry Pi through the I²C interface. Turkin and his colleagues (2023) designed a system which used Raspberry Pi and MPU6050, to compute frequency domain features using FFT and time-series analysis to assess the state of rotating machinery. The system enabled predictive maintenance using condition monitoring based vibration diagnostics.

Also, Bleboo and Aboagye-Otchere (2024) developed an extensive IoT based condition monitoring system, collecting 65,536 samples of vibrations using Raspberry Pi 4 for remote assessment of induction motors which can be used remotely. This work underscored the importance of lower-cost MEMS sensors for high-frequency vibration measurements.

Thermal Analysis with AMG8833

Another important device for condition analysis is the AMG8833 thermal camera sensor which can create an 8x8 pixelated thermal image. Akpo (2021) created an edge capable diagnosis framework employing AMG8833 together with Raspberry Pi which enabled monitoring thermal signatures of rotating machines. This system made it possible to detect overheating and irregular distributions of heat at comparatively lower temperatures. An IoT system was created by Firmanasyh et al. in 2023 where a Raspberry Pi and AMG8833 sensor could monitor the temperature and locate thermal anomalies in industrial machinery. An overheating problem was also identified using an inexpensive real time thermal array (Firmansyah et al., 2023) which assisted in maintenance work. For rotating machinery, Akpro (2021) focused on implementing AMD8833 with Raspberry Pi and created an edge enabled frame work to observe thermal patterns. Their work also included real-time fault detection which was conducted using infrared imaging for predictive maintenance.
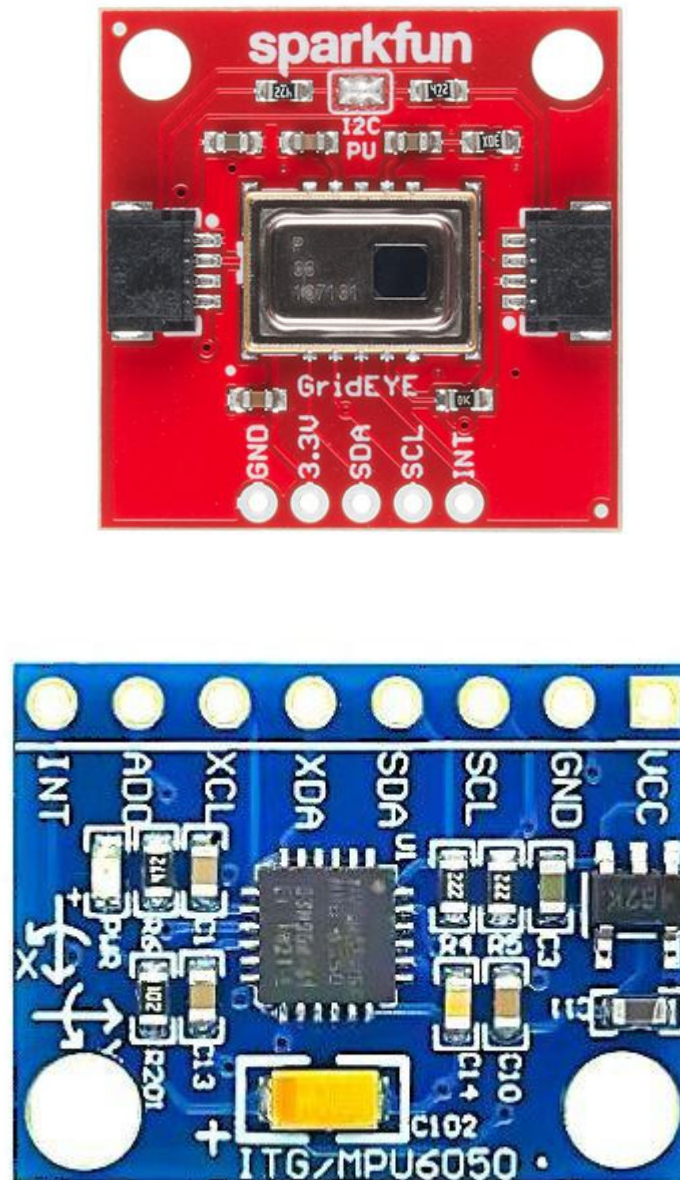
Figure 9. Sensor modules commonly used in edge-based predictive maintenance systems—(top) Panasonic AMG8833 Grid-EYE thermal infrared array sensor for thermal imaging, and (bottom) MPU6050 IMU module for capturing motion and vibration data through integrated accelerometer and gyroscope.

Integrated Vibration & Thermal Monitoring

A comprehensive edge based condition monitoring system can be achieved by incorporating AMG8835 thermal camera and MPU6050 accelerometer. A system that combines these features was created Bleboo and Aboagye-Otchere (2024) for induction motors which used Raspberry Pi to collect an astonishing 65,000 vibration and thermal images. These two allowed real time assessment of operation using both vibration fault classification and heat mapping.

Real-time monitoring of the vibration and thermal parameters of bearing housings was studied by Cardona et al. (2021) using MPU6050 and AMG8833 sensors. Their work proved the feasibility of monitoring MEMS sensors in real time to vibration and thermal metrics for predictive diagnostics.

By applying machine learning on a Raspberry Pi, early warning signs, fault predictions, and automated maintenance alerts can be detected to enhance these integrated sensor systems. This

transforms traditional asset monitoring into proactive asset management, which is more advanced and efficient than traditional systems.

2.5.3. AI and Sensor Fusion for Predictive Maintenance

The ability of the Raspberry Pi to support different sensor interfaces alongside its compatibility with machine learning frameworks TensorFlow Lite and Scikit-learn add to its edge platform capabilities for predictive maintenance tasks. A Raspberry Pi-based system can integrate data from vibration, thermal, and current sensors to achieve autonomy in real-time anomaly detection, fault classification, and health prognosis, thus eliminating the need for secondary cloud computation.

Fusion of Sensors

Sensor fusion combines data from multiple sources in a single sensor to improve reliability, thus making the monitoring of equipment more efficient. Mtesigwa (2020) implemented a neural network-based fault classification model using time-series data from MEMS sensors for an engine condition monitoring project with a Raspberry Pi-based vibration sensing platform (Mtesigwa, 2020).

Vasilache et al. (2024) discussed the role of sensor fusion in improving classification accuracy in acoustic, thermal, and vibration predictive maintenance of low-powered systems by deep neural networks (Vasilache et al., 2024).
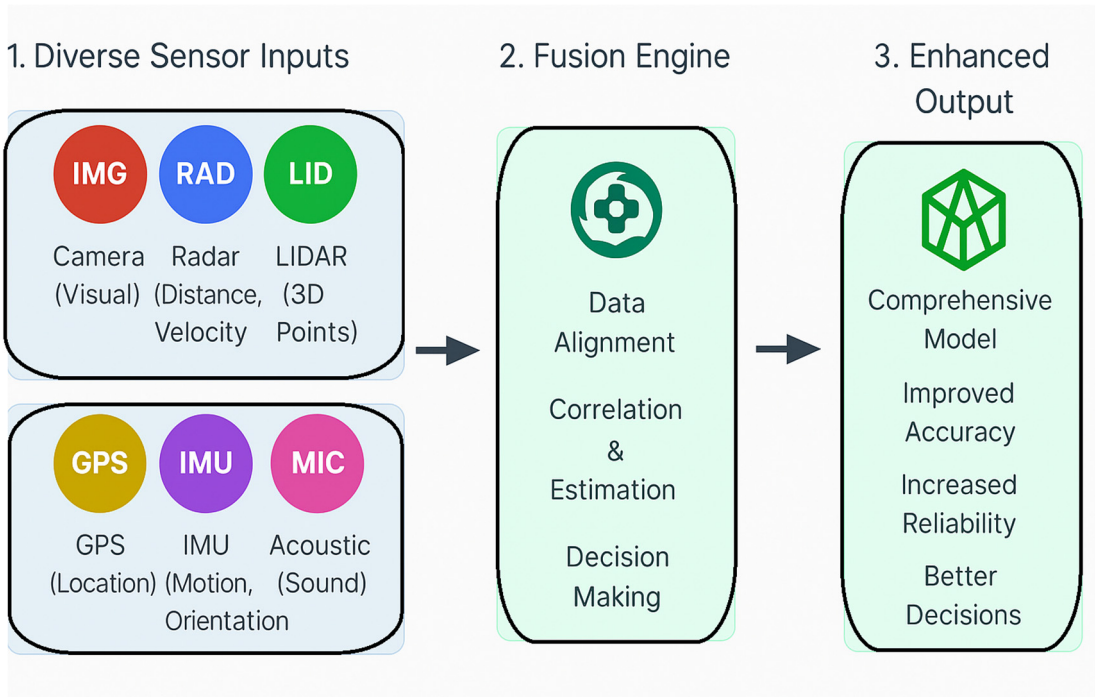

Figure 10. Sensor fusion pipeline integrating diverse sensor inputs (e.g., camera, radar, LiDAR, GPS, IMU, microphone) into a unified fusion engine for data alignment, estimation, and decision making, resulting in a comprehensive model with enhanced accuracy, reliability, and decision quality.

Machine Learning Models on Raspberry Pi

The deployment of machine learning models on Raspberry Pi for on-device analytics has attracted various researchers' attention. Abdulkareem (2024) discussed classifying PC fan failures using a vibration-derived neural network and TensorFlow Lite, achieving triumphant classification outcomes through edge inference on the Raspberry Pi (Abdulkareem, 2024). Akyaz and Engin created a predictive maintenance system for artificial yarn machines which incorporated a Pt100 and IR sensor, as well as a vibration sensor. The data was processed using Keras and TensorFlow on a Raspberry Pi (Akyaz & Engin, 2024). Moreover, Zhou et al.l presented a deployment strategy for edge intelligence-enabled motor fault diagnosis, demonstrating how neural network execution with

TensorFlow Lite quantization on a Raspberry Pi provided real-time diagnostic capability (Zhou et al., 2024).

This increasing line of research supports the idea that Raspberry Pi works for low-power, cost-sensitive platforms for advanced predictive maintenance using sensor fusion and edge AI.

Benefits and Applications

AI and sensor fusion embedded on a Raspberry Pi have considerable benefits for predictive maintenance in industrial and environmental fields:

Cost-Effectiveness: The versatility offered by a Raspberry Pi allows its use for developing AI-enabled maintenance systems for budget-constrained small and medium-sized enterprises. Mourtzis and Angelopoulos highlight the cost saving advantages of remote monitoring provided by Raspberry Pi-based predictive maintenance systems, which reduces deployment and maintenance costs (Mourtzis & Angelopoulos, 2020).

Scalability: The modular design allows Raspberry Pi to be used for the expansion of sensor networks over several machines or areas. Anitha et al. showcased this in the maritime area where health monitoring using neural networks for marine propulsion systems aboard multiple vessels leveraging sensor fusion on Raspberry Pi performed well (Anitha et al., 2024).

Real-Time Processing: With Raspberry Pi, there is low data transfer and latency with devices, especially in marine contexts, where quick decisions need to be made. Algarni et al. presented an edge-computing predictive maintenance framework for maritime communication systems which emphasized the need for immediate reaction from the Raspberry Pi inference unit (Algarni et al., 2024).

Energy efficiency: Khan et al. developed a low-cost sustainable environmental monitoring system using Raspberry Pi powered by solar energy which highlighted the non-dependence on power sources and flexibility provided by Raspberry Pi, showing its value in continuously remote and resource-poor locations (Khan et al., 2024).

These applications have further advanced within the following sectors:

Manufacturing:

The combination of edge computing with your good old smart sensor networks has done wonders for real-time motor diagnostics and fault detection systems. These systems require little infrastructure, thus allowing motors and machinery to be monitored on an almost-continuous basis. This significantly lowers maintenance costs and downtime. Predictive analytics is now able to flag the earliest mechanical wear or electrical faults, helping manufacturers schedule maintenance in a timely manner. This not only enhances equipment lifespan but optimizes production lines safety alongside operational efficiency (Mourtzis & Angelopoulos, 2020).

Marine Engineering:

The smart marine applications combine powerful neural networks with other sensors that provide real-time insights into mechanical and operational anomalies and engine performance through intelligent propulsion monitoring systems. These technologies help with optimizing fuel consumption, preserving the mechanical backbone, and maintaining the vessel within the environmental regulations. Understanding the dynamics of the engine with the data from various onboard sensors allows predicting problems before they surpass certain thresholds, improving efficiency and safety (Anitha et al., 2024).

Environmental Monitoring

The deployment of low-power autonomous sensing platforms that operate in remote resource constrained environments greatly enhanced environmental monitoring. These platforms track critical climate variables such as temperature, humidity, and overall air pollution levels. Powered by

advances in energy-efficient IoT and edge computing, these systems provide real-time data that can aid decision-making in urban planning, agriculture, and even disaster management. These technologies are crucial in addressing the emerging concerns of environmental sustainability due to their scalability and adaptability (Nayar & Dubey, 2025).

Raspberry Pi in Industrial Applications: Advantages and Disadvantages

The Raspberry Pi has emerged as a key component of the ongoing Industry 4.0 and IIoT evolution, due to its low-cost, modular, and flexible design which enables real-time data capture, control, and analysis. Nonetheless, as industrial automation enhances, its deployment in industrial settings needs careful consideration of optimizing capability limitations.

Comparative Analysis of Raspberry Pi's Industrial Benefits and Limitations

Aspect	Benefits	Limitations
Cost	Raspberry Pi devices are extremely cost-effective, making them ideal for SME adoption and prototyping (Vieira et al.).	However, they lack the computational capabilities of higher-end industrial platforms like Jetson or Intel NUC, which can limit use in demanding applications (Karthikeyan et al.).
Size	Their compact footprint enables integration into constrained enclosures and mobile or embedded systems (Ramzey et al.).	But with only 26–40 GPIO pins, they can be insufficient for large-scale I/O applications (Salah, 2021).
Software	Broad OS support (e.g., Raspbian, Ubuntu Core) makes the Pi flexible for various edge software stacks (Behnke & Austad, 2023).	Yet, standard Raspberry Pi OS does not support hard real-time constraints, which may hinder its use in safety-critical environments (Nguyen & Kortun, 2021).
Networking	The Pi includes Ethernet, Wi-Fi, and Bluetooth for connectivity, sufficient for many IIoT deployments (Babayigit & Abubaker, 2023).	However, its components lack industrial-grade durability and may be less robust in high-vibration or extreme temperature environments (Xia et al., 2024).

Overall, the Raspberry Pi offers enormous value for prototyping, educational purposes, and within the bounds of simple to moderately complex industrial applications. However, true responsiveness, additional I/O capabilities, and toughness still present challenges that need to be solved with additional measures or other systems.

Advantages

1. Affordability and Accessibility

For small- to medium-sized enterprises of $50 or less, the Raspberry Pi is particularly economical. This price encourages risk-taking and invention when it comes to educational and industrial contexts (Tarjan et al., Waddell & Fry).

2. Compact Form Factor

The compact design of the Raspberry Pi means that it can be employed in areas with very little space, such as for embedded industrial monitoring, edge computing, or retrofitting. These benefits were described in the context of integration systems where weight and space reduction are of utmost importance (Karthikeyan et al.).

3. Versatile Operating System Support

It supports a variety of Linux-based operating systems, such as Raspberry Pi OS and Ubuntu Core, which broadens the scope of industry applications developers can design or implement solutions with (Culic et al.). This diversity of operating system stems encourages application development and makes changes in production easier (Culic et al.).

4. Connectivity Options

The capability to support Ethernet, Wi-Fi, and Bluetooth makes the integration of Raspberry Pi into industrial networks seamless. Establishing these measures allows for reliable real-time data collection as well as remote monitoring. Research indicates that its implementation in IIoT paradigms demonstrates its versatility for edge-cloud communication (Alasmary, 2023; Kelkar, 2024).

Limitations

1. Performance Constraints

Raspberry Pi is satisfactory for basic monitoring and control, however, processing power limitations make it insufficient for compute intensive tasks like advanced AI inference or large scale real-time analytics. His comparisons with edge devices such as NVIDIA Jetson brought to light the Pi's shortcomings in terms of CPU/GPU throughput and memory bandwidth (Karthikeyan et al., 2023).

2. Limited GPIO and I/O Capabilities

The Pi has 26-40 GPIO pins, which is significantly lower than many of their competitors and limits the possibilities of integrating into an I/O rich system. In addition, the ports associated with industrial communication standards such as RS-485, CAN, or 4-20 mA current loops require some additional hardware interfaces or HATs (hardware add-ons) for compatibility (Salah, 2021).

3. Real-Time Operating System Limitations

The current OS for Raspberry Pi, Raspbian, does not provide real time scheduling. Alternative solutions such as real-time Linux kernels do exist but come with additional PREEMPT-RT patches that require a lot of work to set up which is, frankly, daunting for many prospective users targeting high-reliability control systems (Nguyen & Kortun, 2021).

4. Endurance and Dependability in Extreme Conditions

The Raspberry Pi cannot tolerate high temperature ranges or extreme dust, vibration, and humidity. For industrial use-case scenarios, additional sturdiness is often required by means of rugged cases or I/O peripherals, such as the Compute Module Industrial Kit or the external I/O switches (Xia et al., 2024).

Defensive Measures

For those bounding constraints, a number of actions can be taken:

1. Implementation of Industrial Raspberry Pi Models

The Strato Pi and Revolution Pi optimize the Raspberry Pi's operational efficiency further by having greater resilience to damaging environmental conditions. More stringent IIoT environments can be supported with these devices due to the integration of industrial-grade I/O ports, larger power and energy mitigated heat dissipation, increased operational temperature range, and wider applicable temperature range. As noted by Tomzik (2022), it is \crucial to add adequate ruggedized covers with certain boundaries to ensure reliable Pi-based hardware for industrial purposes.

2. Addition of Real-Time Operating Systems (RTOS)

RTOS addition, or the implementation of real-time Linux patches, allow for better responsiveness in time-crucial tasks. Connects to control frameworks that use dedicated microcontrollers based on the Raspberry Pi platform exist within the realm of RTOS powered control

systems, as shown in Abbas's (2011) work on power-efficient task management for embedded systems. Khalil (2023) further argues that RTOS deployment is essential in ensuring system responsiveness by decreasing delays in response/decision time, thereby guaranteeing predictable performance in IoT gateways.

3. External I/O Expansion

The use of HATs (Hardware Attached on Top) or expansion racks can alleviate the issues concerning GPIO limitations. For instance, Wan (2016) illustrates the advantages RT controllers gain from modular I/O expansion for control cabinets, akin to Raspberry Pi HAT implementations in distributed IIoT systems. The addition of Raspberry Pi compatible CAN shields, RS-485 modules, and analog input boards augment interfacing capabilities without needing alterations to the base board.

4. Protective Enclosures

The sonic overwhelming effects of dust, moisture, vibration, temperature changes, and other destructive forces are reliably eliminated through encasing Raspberry Pi devices in rugged enclosures. These custom shells serve to extend the life of the devices by protecting them and preserving compliance with safety standards which is mandated in physical deployments of lower cost systems integrated into civil infrastructure.

In addition to addressing the constraints of Raspberry Pi in industrial settings, these methods enable new possibilities for implementing scalable IIoT systems that are robust and cost-efficient.

Conclusion

The Raspberry Pi has developed into one of the most remarkable and affordable options in industrial automation and IIoT, blending unique versatility with affordability and modularity. Its value as a flexible computing platform is evident in its overwhelming acceptance throughout almost all areas like SCADA systems, predictive maintenance, digital twins, or edge computing. Notably, the Raspberry Pi must be further evaluated in regards to the limitations of its processing power, available interfaces, lack of native real-time features, and environmental vulnerability. However, deploying strategies like real-time OS modifications, industrial-grade design, increased I/O, and harsh environmental enclosures are proven to solve the vast majority of the challenges. The cost-efficient and easily adjustable Raspberry Pi is transformative when strategically deployed in borderlining industrial-grade applications, enabling rapid evolution towards smart, adaptive, and interconnected manufacturing environments.

References

Zare and M. Tariq Iqbal, "Low-Cost ESP32, Raspberry Pi, Node-Red, and MQTT Protocol Based SCADA System," *2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, Sep. 2020, doi: https://doi.org/10.1109/iemtronics51293.2020.9216412.

L. Aghenta and M. T. Iqbal, "A Low-Cost, Open Source IoT-Based SCADA System Design, and Implementation for Photovoltaics," *ResearchGate*, Dec. 17, 2020. https://www.researchgate.net/publication/347439941_A_Low-Cost_Open_Source_IoT-Based_SCADA_System_Design_and_Implementation_for_Photovoltaics (accessed May 28, 2025).

M. A. Sheba, D. A. Mansour, and N. H. Abbasy, "A new low-cost and low-power industrial internet of things infrastructure for effective integration of distributed and isolated systems with smart grids," *IET generation, transmission & distribution*, vol. 17, no. 20, pp. 4554–4573, Aug. 2023, doi: https://doi.org/10.1049/gtd2.12951.

Z. J. Hussain and R. Krishnaveni, "Multi-Level Security Architecture for Edge Intelligent Devices with Modbus Tcp/Ip Communication Protocols," *SSRN Electronic Journal*, 2024, doi: https://doi.org/10.2139/ssrn.4835969.

I.-V. Nițulescu and A. Korodi, "Supervisory Control and Data Acquisition Approach in Node-RED: Application and Discussions," *IoT*, vol. 1, no. 1, pp. 76–91, Aug. 2020, doi: https://doi.org/10.3390/iot1010005.

P. H. M. Pereira, G. Cainelli, C. E. Pereira, and E. P. de Freitas, "Interoperability Middleware for IIoT Gateways based on Standard Ontologies and AAS," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 9831–9836, Nov. 2023, doi: https://doi.org/10.1016/j.ifacol.2023.10.403.

"Design of an Edge to Cloud IIoT Middleware Architecture - ProQuest," *Proquest.com*, 2022. https://www.proquest.com/openview/ef0e1fcc83b32a2177e412514e346af3/1?pq-origsite=gscholar&cbl=18750&diss=y (accessed May 28, 2025).

P. Henrique, "Interoperability middleware for IIoT gateways based on international standard ontologies and standardized digital representation," *Ufrgs.br*, 2022, doi: http://hdl.handle.net/10183/248690.

K. Ferencz, J. Domokos, and Levente Kovács, "Cloud Integration of Industrial IoT Systems. Architecture, Security Aspects and Sample Implementations," *Acta polytechnica Hungarica*, vol. 21, no. 4, pp. 7–28, Jan. 2024, doi: https://doi.org/10.12700/aph.21.4.2024.4.1.

S.-Y. Chuang, N. Sahoo, H.-W. Lin, and Y.-H. Chang, "Predictive Maintenance with Sensor Data Analytics on a Raspberry Pi-Based Experimental Platform," *Sensors (Basel, Switzerland)*, vol. 19, no. 18, Sep. 2019, doi: https://doi.org/10.3390/s19183884.

D. Mourtzis, J. Angelopoulos, and N. Panopoulos, "Design and development of an IoT enabled platform for remote monitoring and predictive maintenance of industrial equipment," *Procedia Manufacturing*, vol. 54, pp. 166–171, 2021, doi: https://doi.org/10.1016/j.promfg.2021.07.025.

W. Villegas-Ch, Joselin García-Ortiz, and S. Sánchez-Viteri, "Towards Intelligent Monitoring in IoT: AI Applications for Real-Time Analysis and Prediction," *IEEE Access*, vol. 12, pp. 1–1, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3376707.

Kaushik Sathupadi, Sandesh Achar, S. V. Bhaskaran, N. Faruqui, M. Abdullah-Al-Wadud, and J. Uddin, "Edge-Cloud Synergy for AI-Enhanced Sensor Network Data: A Real-Time Predictive Maintenance Framework," *Sensors*, vol. 24, no. 24, pp. 7918–7918, Dec. 2024, doi: https://doi.org/10.3390/s24247918.

R. Atassi and Fuad Alhosban, "Predictive Maintenance in IoT: Early Fault Detection and Failure Prediction in Industrial Equipment," *Journal of Intelligent Systems and Internet of Things*, vol. 9, no. 2, pp. 231–238, Jan. 2023, doi: https://doi.org/10.54216/jisiot.090217.

M. N. Jamil, Olov Schelén, A. A. Monrat, and K. Andersson, "Enabling Industrial Internet of Things by Leveraging Distributed Edge-to-Cloud Computing: Challenges and Opportunities," *IEEE Access*, vol. 12, pp. 127294–127308, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3454812.

"Design of an Edge to Cloud IIoT Middleware Architecture - ProQuest," *Proquest.com*, 2022. https://www.proquest.com/openview/ef0e1fcc83b32a2177e412514e346af3/1?pq-origsite=gscholar&cbl=18750&diss=y

S. R. Poojara, C. K. Dehury, P. Jakovits, and S. N. Srirama, "Serverless data pipeline approaches for IoT data in fog and cloud computing," *Future Generation Computer Systems*, vol. 130, pp. 91–105, May 2022, doi: https://doi.org/10.1016/j.future.2021.12.012.

Sun, K. Guo, Z. Xu, J. Ma, and D. Hu, "Design and Development of Modbus/MQTT Gateway for Industrial IoT Cloud Applications Using Raspberry Pi," *2019 Chinese Automation Congress (CAC)*, Nov. 2019, doi: https://doi.org/10.1109/cac48633.2019.8997492.

H. Amiri, "Investigation on Modbus protocol to develop an interoperable IIoT platform using Node-RED with MQTT gateway," *Polimi.it*, Sep. 2023, doi: http://hdl.handle.net/10589/186336.

R. M. Silva and F. A. C. M. Silva, "An IoT Gateway for Modbus and MQTT Integration," *2019 SBMO/IEEE MTT-S International Microwave and Optoelectronics Conference (IMOC)*, Nov. 2019, doi: https://doi.org/10.1109/imoc43827.2019.9317637.

P. Nguyen-Hoang and P. Vo-Tan, "Development An Open-Source Industrial IoT Gateway," *IEEE Xplore*, Sep. 01, 2019. https://ieeexplore.ieee.org/abstract/document/8905157 (accessed Oct. 29, 2022).

Mancheno and S. Gamboa, "Development of an Industrial IoT Gateway Prototype," *Lecture notes in electrical engineering*, pp. 61–75, Jan. 2022, doi: https://doi.org/10.1007/978-3-031-08942-8_5.

Silva, L. I. Carvalho, J. Soares, and R. C. Sofia, "A Performance Analysis of Internet of Things Networking Protocols: Evaluating MQTT, CoAP, OPC UA," *Applied Sciences*, vol. 11, no. 11, p. 4879, May 2021, doi: https://doi.org/10.3390/app11114879.

P. Wiener, P. Zehnder, and D. Riemer, "Managing geo-distributed stream processing pipelines for the IIoT with StreamPipes edge extensions," pp. 165–176, Jul. 2020, doi: https://doi.org/10.1145/3401025.3401764.

J. Lipp, M. Rath, Maximilian Rudack, Uwe Vroomen, and A. Bührig-Polaczek, "Flexible OPC UA Data Load Optimizations on the Edge of Production," *Lecture notes in business information processing*, pp. 43–61, Jan. 2021, doi: https://doi.org/10.1007/978-3-030-75418-1_3.

Dimitry, "Architecting the edge-cloud continuum for IoT-based monitoring applications," *Unibo.it*, Jul. 2024, doi: https://amsdottorato.unibo.it/id/eprint/11496/1/zyrianoff_ivan_tesi.pdf.

D. Silva, L. I. Carvalho, J. Soares, and R. C. Sofia, "A Performance Analysis of Internet of Things Networking Protocols: Evaluating MQTT, CoAP, OPC UA," *Applied Sciences*, vol. 11, no. 11, p. 4879, May 2021, doi: https://doi.org/10.3390/app11114879.

Njor, M. A. Hasanpour, J. Madsen, and Xenofon Fafoutis, "A Holistic Review of the TinyML Stack for Predictive Maintenance," *IEEE Access*, pp. 1–1, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3512860.

Saraan, H. Nizam, and S. M. Zahara, "Tiny Machine Learning (TinyML) Systems," *ResearchGate*, Dec. 10, 2024. https://www.researchgate.net/publication/386579238_Tiny_Machine_Learning_TinyML_Systems

N. Exell, "Applicability of TinyML for maintenance predictability," *Doria.fi*, 2023, doi: https://www.doria.fi/handle/10024/188060.

Z. Chen, Y. Gao, and J. Liang, "LOPdM: A Low-Power On-Device Predictive Maintenance System Based on Self-Powered Sensing and TinyML," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–13, Jan. 2023, doi: https://doi.org/10.1109/tim.2023.3308251.

R. de, L. Radrigan, and A. S. Morales, "Enhancing Predictive Maintenance in Mining Mobile Machinery through a TinyML-enabled Hierarchical Inference Network," Nov. 11, 2024. https://www.researchgate.net/publication/385722233_Enhancing_Predictive_Maintenance_in_Mining_Mobile_Machinery_through_a_TinyML-enabled_Hierarchical_Inference_Network

Pietro D'Agostino, M. Violante, and G. Macario, "A Scalable Fog Computing Solution for Industrial Predictive Maintenance and Customization," *Electronics*, vol. 14, no. 1, pp. 24–24, Dec. 2024, doi: https://doi.org/10.3390/electronics14010024.

Jouini, K. Sethom, A. Namoun, N. Aljohani, M. H. Alanazi, and M. N. Alanazi, "A Survey of Machine Learning in Edge Computing: Techniques, Frameworks, Applications, Issues, and Research Directions," *Technologies*, vol. 12, no. 6, p. 81, Jun. 2024, doi: https://doi.org/10.3390/technologies12060081.

1 Karras et al., "TinyML Algorithms for Big Data Management in Large-Scale IoT Systems," *ProQuest*, p. 42, 2024, doi: https://doi.org/10.3390/fi16020042.

R. de, L. Radrigan, and A. S. Morales, "Enhancing Predictive Maintenance in Mining Mobile Machinery through a TinyML-enabled Hierarchical Inference Network," Nov. 11, 2024. https://www.researchgate.net/publication/385722233_Enhancing_Predictive_Maintenance_in_Mining_Mobile_Machinery_through_a_TinyML-enabled_Hierarchical_Inference_Network

E. Njor, M. A. Hasanpour, J. Madsen, and Xenofon Fafoutis, "A Holistic Review of the TinyML Stack for Predictive Maintenance," *IEEE Access*, pp. 1–1, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3512860.

Rakhee Kallimani, K. Pai, Prasoon Raghuwanshi, S. Iyer, and L. Alcaraz, "TinyML: Tools, Applications, Challenges, and Future Research Directions," *arXiv (Cornell University)*, Mar. 2023, doi: https://doi.org/10.1007/s11042-023-16740-9.

M. Arif and M. Rashid, "A Literature Review on Model Conversion, Inference, and Learning Strategies in EdgeML with TinyML Deployment," *Computers, Materials & Continua*, vol. 83, no. 1, pp. 13–64, 2025, doi: https://doi.org/10.32604/cmc.2025.062819.

Cilfone, L. Davoli, and G. Ferrari, "LoRa Meets IP: A Container-Based Architecture to Virtualize LoRaWAN End Nodes," *IEEE Transactions on Mobile Computing*, vol. 23, no. 10, pp. 9191–9207, Oct. 2024, doi: https://doi.org/10.1109/tmc.2024.3359150.

Dimitry, "Architecting the edge-cloud continuum for IoT-based monitoring applications," *Unibo.it*, Jul. 2024, doi: https://amsdottorato.unibo.it/id/eprint/11496/1/zyrianoff_ivan_tesi.pdf.

"Edge Computing Patterns for Solution Architects," *Google Books*, 2024. https://books.google.it/books?hl=en&lr=&id=0-XvEAAAQBAJ&oi=fnd&pg=PP1&redir_esc=y#v=onepage&q&f=false (accessed May 28, 2025).

"Design And Implementation Of Wireless Sensor Networks Using LoRa WAN, MQTT and Cloud Computing : Vishu Sharma : Free Download, Borrow, and Streaming : Internet Archive," *Internet Archive*, May 15, 2023. https://archive.org/details/design-and-implementation-of-wireless-sensor-networks-using-lo-ra-wan-mqtt-and-cloud-computing/mode/2up?q=%22rong+xiang%22 (accessed May 28, 2025).

"Deliverables | Nephele," *Nephele-project.eu*, 2023. https://nephele-project.eu/library/deliverables (accessed May 28, 2025).

R. Mahmud and A. N. Toosi, "Con-Pi: A Distributed Container-based Edge and Fog Computing Framework," *IEEE Internet of Things Journal*, pp. 1–1, 2021, doi: https://doi.org/10.1109/jiot.2021.3103053.

Lubomir Urblik, E. Kajati, P. Papcun, and Iveta Zolotová, "Containerization in Edge Intelligence: A Review," *Electronics*, vol. 13, no. 7, pp. 1335–1335, Apr. 2024, doi: https://doi.org/10.3390/electronics13071335.

R. K. Gupta, "Performance Characterization of Containers in Edge Computing," *arXiv e-prints*, p. arXiv:2505.02082, May 2025, doi: https://doi.org/10.48550/arXiv.2505.02082.

Y. Liu, D. Lan, Z. Pang, M. Karlsson, and S. Gong, "Performance Evaluation of Containerization in Edge-Cloud Computing Stacks for Industrial Applications: A Client Perspective," *IEEE Open Journal of the Industrial Electronics Society*, vol. 2, pp. 153–168, 2021, doi: https://doi.org/10.1109/ojies.2021.3055901.

Leskinen, "Applicability of Kubernetes to Industrial IoT Edge Computing System," *aaltodoc.aalto.fi*, Oct. 2020, Available: https://aaltodoc.aalto.fi/items/e20c6ec2-090f-4c46-81e1-0bb140df6a38

Ionut-Catalin Donca, Ovidiu Petru Stan, Marius Misaros, A. Stan, and Liviu Miclea, "Comprehensive Security for IoT Devices with Kubernetes and Raspberry Pi Cluster," *Electronics*, vol. 13, no. 9, pp. 1613–1613, Apr. 2024, doi: https://doi.org/10.3390/electronics13091613.

M. S. Elbakry, T. Ismail, A. M. Mahmoud, S. Hassan, and Toqa Hesham, "Digital Twin Simulation and Implementation of Connected and Automated Vehicles with the Unity Game Engine," Jun. 20, 2023. https://www.researchgate.net/publication/382622614_Digital_Twin_Simulation_and_Implementation_of_Connected_and_Automated_Vehicles_with_the_Unity_Game_Engine

R. Andre, "Digital twin for monitoring, optimization and training in battery production," *Openarchive.usn.no*, 2022, doi: no.usn:wiseflow:6583421:50226158.

Rahman, Uzair Muzamil Shah, Syed Morsleen Riaz, Kashif Kifayat, Syed Atif Moqurrab, and J. Yoo, "Digital twin framework for smart greenhouse management using next-gen mobile networks and machine learning," *Future Generation Computer Systems*, Mar. 2024, doi: https://doi.org/10.1016/j.future.2024.03.023.

R. Jedermann, K. Singh, W. Lang, and P. Mahajan, "Digital twin concepts for linking live sensor data with real-time models," *Journal of Sensors and Sensor Systems*, vol. 12, no. 1, pp. 111–121, Apr. 2023, doi: https://doi.org/10.5194/jsss-12-111-2023.

J. Pablo, D. Sanchez-Londono, and G. Barbieri, "A Monitoring Digital Twin for Services of Controlled Environment Agriculture," *IFAC-PapersOnLine*, vol. 55, no. 19, pp. 85–90, Jan. 2022, doi: https://doi.org/10.1016/j.ifacol.2022.09.188.

P. Maheshwari, S. Kamble, A. Belhadi, M. Venkatesh, and M. Z. Abedin, "Digital twin-driven real-time planning, monitoring, and controlling in food supply chains," *Technological Forecasting and Social Change*, vol. 195, p. 122799, Oct. 2023, doi: https://doi.org/10.1016/j.techfore.2023.122799.

A. Le et al., "Study with us," *University of Greenwich*, Nov. 2024, doi: http://gala.gre.ac.uk/id/eprint/48636/7/48636%20LE_Digital_Twins_For_Real-Time_Monitoring_And_Operation_Of_Coffee_Value_Chain_And_Supply_Chain_%28OA%29_2024.pdf.

Y.-S. Huang, "Digital Twin Development and Advanced Process Control for Continuous Pharmaceutical Manufacturing," *figshare*, Jul. 2023, doi: https://doi.org/10.25394/PGS.23735379.v1.

"Digital Twins for Cyber-Physical Healthcare Systems: Architecture, Requirements, Systematic Analysis and Future Prospects," *Ieee.org*, 2025, doi: https://doi.org/10.1109/ACCESS.2025.3547991.

H. Hosamo, P. R. Svennevig, K. Svidt, D. Han, and H. K. Nielsen, "A Digital Twin predictive maintenance framework of air handling units based on automatic fault detection and diagnostics," *Energy and Buildings*, vol. 261, p. 111988, Apr. 2022, doi: https://doi.org/10.1016/j.enbuild.2022.111988.

Salzano, S. Cascone, E. P. Zitiello, and M. Nicolella, "HVAC System Performance in Educational Facilities: A Case Study on the Integration of Digital Twin Technology and IoT Sensors for Predictive Maintenance," *Journal of Architectural Engineering*, vol. 31, no. 1, Jan. 2025, doi: https://doi.org/10.1061/jaeied.aeeng-1855.

R. Raja Singh, G. Bhatti, Dattatraya Kalel, V. Indragandhi, and Faisal Alsaif, "Building a Digital Twin Powered Intelligent Predictive Maintenance System for Industrial AC Machines," *Machines*, vol. 11, no. 8, pp. 796–796, Aug. 2023, doi: https://doi.org/10.3390/machines11080796.

M. Hasan, "CoAP and MQTT Measurements over LoRaWAN," *Urn.fi*, 2020, doi: https://oulurepo.oulu.fi/handle/10024/17008.

Mr. A. U. Mirza, Mr. M. Irfan, and Mr. P. K. Chandapeta, "Internet Of Things-Based Applied System Development Cookbook," Mar. 2024, doi: https://doi.org/10.59646/iot/140.

None Lalhriatpuii, None Ruchi, and V. Wasson, "Comprehensive Exploration of IoT Communication Protocol: CoAP, MQTT, HTTP, LoRaWAN and AMQP," *Communications in computer and information science*, pp. 261–274, Nov. 2024, doi: https://doi.org/10.1007/978-3-031-75861-4_23.

E. Al-Masri et al., "Investigating Messaging Protocols for the Internet of Things (IoT)," *IEEE Access*, vol. 8, pp. 94880–94911, 2020, doi: https://doi.org/10.1109/access.2020.2993363.

R. Das, "Enhancing Health and Environmental Monitoring," *Chapman and Hall/CRC eBooks*, pp. 166–189, Apr. 2024, doi: https://doi.org/10.1201/9781003391456-9.

M. Guerbaoui et al., "From Data to Decisions: A Smart IoT and Cloud Approach to Environmental Monitoring," *E3S Web of Conferences*, vol. 601, p. 00008, 2025, doi: https://doi.org/10.1051/e3sconf/202560100008.

K. Deepika and B. R. Prasad, "IoT-Based Dashboards for Monitoring Connected Farms Using Free Software and Open Protocols," *Algorithms for intelligent systems*, pp. 529–543, Dec. 2022, doi: https://doi.org/10.1007/978-981-19-6004-8_43.

W. Antus, "Plan a dashboard for energy measuring, improve overview of energy consumption, and increase energy recovery," *Uwasa.fi*, 2023, doi: https://osuva.uwasa.fi/handle/10024/15928.

S. Williams, M. Short, and A. Arockia Selvakumar, "An internet of things development framework for smart city pilot study," *AIP conference proceedings*, vol. 3216, pp. 040004–040004, Jan. 2024, doi: https://doi.org/10.1063/5.0226516.

T. Domínguez-Bolaño, V. Barral, C. J. Escudero, and J. A. García-Naya, "An IoT system for a smart campus: Challenges and solutions illustrated over several real-world use cases," *Internet of Things*, vol. 25, p. 101099, Feb. 2024, doi: https://doi.org/10.1016/j.iot.2024.101099.

P. A and V. T, "IoT-based Smart University Using MQTT Protocol: A Scalable and Sustainable Design Approach," *2024 5th International Conference on Communication, Computing & Industry 6.0 (C2I6)*, pp. 1–7, Dec. 2024, doi: https://doi.org/10.1109/c2i663243.2024.10895475.

M. Guerbaoui et al., "From Data to Decisions: A Smart IoT and Cloud Approach to Environmental Monitoring," *E3S Web of Conferences*, vol. 601, p. 00008, 2025, doi: https://doi.org/10.1051/e3sconf/202560100008.

"Real-Time Indoor Air Quality (IAQ) Monitoring System for Smart Buildings - ProQuest," *Proquest.com*, 2022. https://www.proquest.com/openview/44e3b5a64af9decc188ff9d150fc1f5f/1?pq-origsite=gscholar&cbl=2026366&diss=y (accessed May 28, 2025).

"Learn IoT Programming Using Node-RED," *Google Books*, 2022. https://books.google.it/books?hl=en&lr=&id=03BcEAAAQBAJ&oi=fnd&pg=PT19&redir_esc=y#v=onepage&q&f=false (accessed May 28, 2025).

S. Arash Omidi, Mirza, and M. Tariq Iqbal, "Design and Implementation of Node-Red Based Open-Source SCADA Architecture for a Hybrid Power System," *Energies*, vol. 16, no. 5, pp. 2092–2092, Feb. 2023, doi: https://doi.org/10.3390/en16052092.

Martikkala, "Towards Utilization of Low-cost Technologies in Democratizing IoT Development," *Tuni.fi*, Apr. 2024, doi: https://doi.org/978-952-03-3399-7.

M. A. Radia, K. El, M. M. Kamel, and A. S. Atlam, "Implementation of a Real-Time Data Acquisition System," *ResearchGate*, Sep. 2024, doi: https://doi.org/10.13140/RG.2.2.29086.75847.

S. T. Ahmed, "Implementation And Optimization Of A Secure, Scalable, And Robust Long-Range Low-Power Mesh Network For Enhanced Geo-Location Accuracy And Efficient Data Aggregation," *Digital Commons @PVAMU*, 2024. https://digitalcommons.pvamu.edu/pvamu-dissertations/57/ (accessed May 28, 2025).

Y. Ranjan et al., "RADAR-IoT: An Open-Source, Interoperable, and Extensible IoT Gateway Framework for Health Research," *Sensors*, vol. 24, no. 14, pp. 4614–4614, Jul. 2024, doi: https://doi.org/10.3390/s24144614.

Antony Pradeesh D and N. P. Subiramaniyam, "Enhanced Real-Time Analysis and Anomaly Detection in Smart Health Monitoring Systems through Integration of Deep Learning Algorithm with IoT-Edge Computing," Apr. 2024, doi: https://doi.org/10.1109/iccds60734.2024.10560367.

L. Klymyuk, "On-edge device for human tracking and identification based on RGB camera input data," *Polimi.it*, Sep. 2023, doi: http://hdl.handle.net/10589/177528.

Garcia-Perez, R. Miñón, A. I. Torre-Bastida, and E. Zulueta-Guerrero, "Analysing Edge Computing Devices for the Deployment of Embedded AI," *Sensors (Basel, Switzerland)*, vol. 23, no. 23, p. 9495, Nov. 2023, doi: https://doi.org/10.3390/s23239495.

E. Dubois, "Shared Learning Among Distributed Edge Devices Using Coral Edge TPU Machine Learning Engines," *Dtic.mil*, Jun. 2021. https://apps.dtic.mil/sti/html/trecms/AD1150924/index.html (accessed May 28, 2025).

R. R. Nair, T. Babu, and P. M. Ebin, "Developing Edge AI for Embedded Systems," *Chapman and Hall/CRC eBooks*, pp. 32–54, Jan. 2025, doi: https://doi.org/10.1201/9781003481089-4.

Kasra Aminiyeganeh, "Performance Evaluation of the Object Detection Algorithms on Embedded Devices," *Concordia.ca*, Jan. 2025, doi: https://spectrum.library.concordia.ca/id/eprint/993064/1/Aminiyeganeh_MSc_S2024.pdf.

F. Oliveira, D. G. Costa, F. Assis, and I. Silva, "Internet of Intelligent Things: A convergence of embedded systems, edge computing and machine learning," *Internet of Things*, vol. 26, p. 101153, Jul. 2024, doi: https://doi.org/10.1016/j.iot.2024.101153.

Jouini, K. Sethom, A. Namoun, N. Aljohani, M. H. Alanazi, and M. N. Alanazi, "A Survey of Machine Learning in Edge Computing: Techniques, Frameworks, Applications, Issues, and Research Directions," *Technologies*, vol. 12, no. 6, p. 81, Jun. 2024, doi: https://doi.org/10.3390/technologies12060081.

M. Rossi, G. Minicozzi, G. Pascarella, and A. Capasso, "ESG, Competitive Advantage and Financial performances: a Preliminary Research," *Handle.net*, vol. 1, no. 1, pp. 969–986, Sep. 2020.

R. Das, "Enhancing Health and Environmental Monitoring," *Chapman and Hall/CRC eBooks*, pp. 166–189, Apr. 2024, doi: https://doi.org/10.1201/9781003391456-9.

N. Harum et al., "AN AUTOMATED DATA LOGGER SYSTEM FOR REAL-TIME MONITORING AND ANOMALY DETECTION IN INDUSTRIAL IOT ENVIRONMENT," *Journal of Advanced Manufacturing Technology (JAMT)*, vol. 18, no. 3, 2024, Available: https://jamt.utem.edu.my/jamt/article/view/6798

M. D. Mudaliar and N. Sivakumar, "IoT Based Real Time Energy Monitoring System Using Raspberry Pi," *Internet of Things*, p. 100292, Sep. 2020, doi: https://doi.org/10.1016/j.iot.2020.100292.

Mušović, "SYSTEM DESIGN FOR PREDICTIVE ANALYSIS AND MACHINE LEARNING IN INDUSTRIAL AUTOMATION," *DIVA*, 2024. https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1882079&dswid=-3162 (accessed May 28, 2025).

Masinde Mtesigwa, Masinde, "VIBRATION SENSING FOR ENGINE CONDITION MONITORING AND PREDICTIVE MAINTENANCE," *Doria.fi*, 2020, doi: https://www.doria.fi/handle/10024/177442.

M. Rayhan, "Evaluating real-time monitoring and data-storage systems for active magnetic bearings based high-speed machines," *Lutpub.lut.fi*, 2024, doi: https://lutpub.lut.fi/handle/10024/168532.

E. Kitadu, "Cloud-Based Platform for Industrial Data Management and Analysis," *Theseus.fi*, 2025, doi: http://www.theseus.fi/handle/10024/883432.

M. D. Mudaliar and N. Sivakumar, "IoT Based Real Time Energy Monitoring System Using Raspberry Pi," *Internet of Things*, p. 100292, Sep. 2020, doi: https://doi.org/10.1016/j.iot.2020.100292.

M. Bolanowski, A. Paszkiewicz, T. Żabiński, G. Piecuch, M. Salach, and K. Tomecki, "System Architecture for Diagnostics and Supervision of Industrial Equipment and Processes in an IoE Device Environment," *Electronics*, vol. 12, no. 24, p. 4935, Jan. 2023, doi: https://doi.org/10.3390/electronics12244935.

Trinh, "A Demonstration of an Energy Feedback Research Platform in a Field Study of Real-Time Social Comparisons," *Escholarship.org*, Oct. 2015. https://escholarship.org/uc/item/2sr794kx (accessed May 28, 2025).

"Learning-Based Ubiquitous Sensing for Solving Real-World Problems - ProQuest," *Proquest.com*, 2023. https://www.proquest.com/openview/c776daee6da12a137fc070ee8d2fa245/1?pq-origsite=gscholar&cbl=18750&diss=y (accessed May 28, 2025).

"Machine Learning Approach for Predictive Maintenance in an Electric Motor; A Classifier Approach - ProQuest," *Proquest.com*, 2024. https://www.proquest.com/openview/743428fe822f5e96a1c812e70b4e37e0/1?pq-origsite=gscholar&cbl=2026366&diss=y (accessed May 28, 2025).

Turkin, В'ячеслав Андрійович Лезновскій, А. А. Зеленков, Agil Nabizade, L. Volobuieva, and Viktoriia Turkina, "The Use of IoT for Determination of Time and Frequency Vibration Characteristics of Industrial Equipment for Condition-Based Maintenance," *Computation (Basel)*, vol. 11, no. 9, pp. 177–177, Sep. 2023, doi: https://doi.org/10.3390/computation11090177.

Heri Firmansyah, Widyaningrum Indrasari, and Ridho Achmadi, "Development of Temperature Measurement and Monitoring System on Machine with Thermal Camera AMG8833 based Internet of Things (IoT)," *International Journal on Smart Material and Mechatronics*, vol. 10, no. 1, 2023, Accessed: May 28, 2025. [Online]. Available: https://ijsmm.id/index.php/ijsmm/article/view/68

Heri Firmansyah, Widyaningrum Indrasari, and Ridho Achmadi, "Development of Temperature Measurement and Monitoring System on Machine with Thermal Camera AMG8833 based Internet of Things (IoT)," *International Journal on Smart Material and Mechatronics*, vol. 10, no. 1, 2023, Accessed: May 28, 2025. [Online]. Available: https://ijsmm.id/index.php/ijsmm/article/view/68

M. Cardona, M. Cifuentes, B. Hernandez, and W. Prado, "A Case Study on Remote Instrumentation of Vibration and Temperature in Bearing Housings," *Journal of Low Power Electronics and Applications*, vol. 11, no. 4, pp. 44–44, Nov. 2021, doi: https://doi.org/10.3390/jlpea11040044.

Masinde Mtesigwa, Masinde, "VIBRATION SENSING FOR ENGINE CONDITION MONITORING AND PREDICTIVE MAINTENANCE," *Doria.fi*, 2020, doi: https://www.doria.fi/handle/10024/177442.

Vasilache et al., "Low-Power Vibration-Based Predictive Maintenance for Industry 4.0 using Neural Networks: A Survey," *arXiv.org*, 2024. https://arxiv.org/abs/2408.00516

H. Abdulkareem, "Predictive Maintenance in Industrial Equipment Using IoT Devices," *Theseus.fi*, 2024, doi: http://www.theseus.fi/handle/10024/876252.

T. Akyaz and D. Engın, "Machine Learning-Based Predictive Maintenance System for Artificial Yarn Machines," *IEEE Access*, vol. 12, pp. 125446–125461, 2024, doi: https://doi.org/10.1109/access.2024.3454548.

Z. Zhou, Y. Qiao, X. Lin, P. Li, N. Wu, and D. Yu, "A Deployment Method for Motor Fault Diagnosis Application Based on Edge Intelligence," *Sensors*, vol. 25, no. 1, pp. 9–9, Dec. 2024, doi: https://doi.org/10.3390/s25010009.

Mourtzis, J. Angelopoulos, and N. Panopoulos, "Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment Based on Mixed Reality," *Frontiers in Mechanical Engineering*, vol. 6, Dec. 2020, doi: https://doi.org/10.3389/fmech.2020.578379.

Anitha, R. Vijayakumar, J. Gnana Jeslin, K. Elangovan, M. Jagadeeswaran, and C. Srinivasan, "Marine Propulsion Health Monitoring: Integrating Neural Networks and IoT Sensor Fusion in Predictive Maintenance," *2024 2nd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT)*, Mar. 2024, doi: https://doi.org/10.1109/dicct61038.2024.10533037.

Abdulmohsen Algarni, Tayfun Acarer, and Z. Ahmad, "An Edge Computing-based Preventive Framework with Machine Learning-Integration for Anomaly Detection and Risk Management in Maritime Wireless Communications," *IEEE access*, vol. 12, pp. 53646–53663, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3387529.

Mourtzis, J. Angelopoulos, and N. Panopoulos, "Intelligent Predictive Maintenance and Remote Monitoring Framework for Industrial Equipment Based on Mixed Reality," *Frontiers in Mechanical Engineering*, vol. 6, Dec. 2020, doi: https://doi.org/10.3389/fmech.2020.578379.

Anitha, R. Vijayakumar, J. Gnana Jeslin, K. Elangovan, M. Jagadeeswaran, and C. Srinivasan, "Marine Propulsion Health Monitoring: Integrating Neural Networks and IoT Sensor Fusion in Predictive Maintenance," *2024 2nd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT)*, Mar. 2024, doi: https://doi.org/10.1109/dicct61038.2024.10533037.

J. McLaughlin, A. Mathur, and S. M. Tauseef, *Emerging Sensors for Environmental Monitoring*. Elsevier, 2025.

G. Vieira, J. Barbosa, P. Leitao, and L. Sakurada, "Low-Cost Industrial Controller based on the Raspberry Pi Platform," *2020 IEEE International Conference on Industrial Technology (ICIT)*, Feb. 2020, doi: https://doi.org/10.1109/icit45562.2020.9067148.

"A Systematic Analysis on Raspberry Pi Prototyping: Uses, Challenges, Benefits, and Drawbacks | IEEE Journals & Magazine | IEEE Xplore," *ieeexplore.ieee.org*. https://ieeexplore.ieee.org/abstract/document/10091882

H. Ramzey, M. Badawy, M. Elhosseini, and A. A. Elbaset, "I2OT-EC: A Framework for Smart Real-Time Monitoring and Controlling Crude Oil Production Exploiting IIOT and Edge Computing," *Energies*, vol. 16, no. 4, p. 2023, Jan. 2023, doi: https://doi.org/10.3390/en16042023.

B. Salah, "Real-Time Implementation of a Fully Automated Industrial System Based on IR 4.0 Concept," *Actuators*, vol. 10, no. 12, p. 318, Nov. 2021, doi: https://doi.org/10.3390/act10120318.

Behnke and H. Austad, "Real-Time Performance of Industrial IoT Communication Technologies: A Review," *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 7399–7410, Nov. 2023, doi: https://doi.org/10.1109/jiot.2023.3332507.

Nguyen and A. Kortun, "Real-time Optimisation for Industrial Internet of Things (IIoT): Overview, Challenges and Opportunities," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, p. 167654, Dec. 2020, doi: https://doi.org/10.4108/eai.16-12-2020.167654.

Bilal Babayigit and M. Abubaker, "Industrial Internet of Things: A Review of Improvements Over Traditional SCADA Systems for Industrial Automation," pp. 1–14, Jan. 2023, doi: https://doi.org/10.1109/jsyst.2023.3270620.

L. Xia, B. Zhang, and J. Wang, "Research on Industrial Internet Data Collection Application Based on Raspberry Pi," *2024 4th International Conference on Electronic Information Engineering and Computer Communication (EIECC)*, pp. 689–693, Dec. 2024, doi: https://doi.org/10.1109/eiecc64539.2024.10929562.

L. Tarjan, I. Šenk, J. Erić Obućina, S. Stankovski, and G. Ostojić, "Extending Legacy Industrial Machines by a Low-Cost Easy-to-Use IoT Module for Data Acquisition," *Symmetry*, vol. 12, no. 9, p. 1486, Sep. 2020, doi: https://doi.org/10.3390/sym12091486.

R. K. Waddell and T. W. Fry, "The affordably connected factory: A brief evaluation of sensors and hardware deployed in industrial applications," *Procedia Manufacturing*, vol. 53, pp. 741–747, 2021, doi: https://doi.org/10.1016/j.promfg.2021.06.084.

"A Systematic Analysis on Raspberry Pi Prototyping: Uses, Challenges, Benefits, and Drawbacks | IEEE Journals & Magazine | IEEE Xplore," *ieeexplore.ieee.org*. https://ieeexplore.ieee.org/abstract/document/10091882

"Commercial and Industrial Internet of Things Applications with the Raspberry Pi," *Google Books*, 2020. https://books.google.it/books?hl=en&lr=&id=2yjfDwAAQBAJ&redir_esc=y (accessed May 28, 2025).

H. Alasmary, "RDAF-IIoT: Reliable Device-Access Framework for the Industrial Internet of Things," *Mathematics*, vol. 11, no. 12, p. 2710, Jun. 2023, doi: https://doi.org/10.3390/math11122710.

"A Systematic Analysis on Raspberry Pi Prototyping: Uses, Challenges, Benefits, and Drawbacks | IEEE Journals & Magazine | IEEE Xplore," *ieeexplore.ieee.org*. https://ieeexplore.ieee.org/abstract/document/10091882

B. Salah, "Real-Time Implementation of a Fully Automated Industrial System Based on IR 4.0 Concept," *Actuators*, vol. 10, no. 12, p. 318, Nov. 2021, doi: https://doi.org/10.3390/act10120318.

L. Nguyen and A. Kortun, "Real-time Optimisation for Industrial Internet of Things (IIoT): Overview, Challenges and Opportunities," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, p. 167654, Dec. 2020, doi: https://doi.org/10.4108/eai.16-12-2020.167654.

L. Xia, B. Zhang, and J. Wang, "Research on Industrial Internet Data Collection Application Based on Raspberry Pi," *2024 4th International Conference on Electronic Information Engineering and Computer Communication (EIECC)*, pp. 689–693, Dec. 2024, doi: https://doi.org/10.1109/eiecc64539.2024.10929562.

A. Tomzik, "Distributed and cloud-based machine tool control," *Handle.net*, 2022. https://hdl.handle.net/2292/58052 (accessed May 28, 2025).

H. Khalil, "PoT: bridging IoT with phone technology," *Scholaris.ca*, Sep. 2023. https://ontariotechu.scholaris.ca/items/0936d0fe-ab8c-4759-b144-d9039d168bce (accessed May 28, 2025).

W. Khaled, 2016. Accessed: May 28, 2025. [Online]. Available: https://core.ac.uk/download/pdf/71285685.pdf

P. Mathur, *IoT Machine Learning Applications in Telecom, Energy, and Agriculture*. 2020. doi: https://doi.org/10.1007/978-1-4842-5549-0.

V. Noor Mohammed, O. F. Abdulateef, and A. H. Hamad, "An IoT and Machine Learning-Based Predictive Maintenance System for Electrical Motors," *Journal européen des systèmes automatisés*, vol. 56, no. 4, pp. 651–656, Aug. 2023, doi: https://doi.org/10.18280/jesa.560414.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.