future internet

MDPI

Review

Approaches and Challenges in Internet of Robotic Things

Aqsa Sayeed [1], Chaman Verma [2,*], Neerendra Kumar [1,*], Neha Koul [1] and Zoltán Illés [2]

[1] Department of Computer Science & IT, Central University of Jammu, Jammu 181143, India
[2] Department of Media and Educational Informatics, Faculty of Informatics, Eötvös Loránd University, 1053 Budapest, Hungary
* Correspondence: chaman@inf.elte.hu (C.V.); neerendra.csit@cujammu.ac.in (N.K.)

Abstract: The Internet of robotic things (IoRT) is the combination of different technologies including cloud computing, robots, Internet of things (IoT), artificial intelligence (AI), and machine learning (ML). IoRT plays a major role in manufacturing, healthcare, security, and transport. IoRT can speed up human development by a very significant percentage. IoRT allows robots to transmit and receive data to and from other devices and users. In this paper, IoRT is reviewed in terms of the related techniques, architectures, and abilities. Consequently, the related research challenges are presented. IoRT architectures are vital in the design of robotic systems and robotic things. The existing 3–7-tier IoRT architectures are studied. Subsequently, a detailed IoRT architecture is proposed. Robotic technologies provide the means to increase the performance and capabilities of the user, product, or process. However, robotic technologies are vulnerable to attacks on data security. Trust-based and encryption-based mechanisms can be used for secure communication among robotic things. A security method is recommended to provide a secure and trustworthy data-sharing mechanism in IoRT. Significant security challenges are also discussed. Several known attacks on ad hoc networks are illustrated. Threat models ensure integrity confidentiality and availability of the data. In a network, trust models are used to boost a system's security. Trust models and IoRT networks play a key role in obtaining a steady and nonvulnerable configuration in the network. In IoRT, remote server access results in remote software updates of robotic things. To study navigation strategies, navigation using fuzzy logic, probabilistic roadmap algorithms, laser scan matching algorithms, heuristic functions, bumper events, and vision-based navigation techniques are considered. Using the given research challenges, future researchers can get contemporary ideas of IoRT implementation in the real world.

Keywords: IoRT; robotics; sensors; augmented reality and virtual reality; robot navigation techniques; heuristic functions; bumper event; fuzzy logic; trust-based mechanism; IoRT security framework; threat model; trust model; machine learning; IoRT remote server access; IoRT energy efficiency

Citation: Sayeed, A.; Verma, C.; Kumar, N.; Koul, N.; Illés, Z. Approaches and Challenges in Internet of Robotic Things. *Future Internet* **2022**, *14*, 265. https://doi.org/10.3390/fi14090265

Academic Editors: Christoph Stach and Clémentine Gritti

Received: 9 August 2022
Accepted: 8 September 2022
Published: 14 September 2022

1. Introduction

Robotic systems have aided various technological developments during the previous decade. During the 1990s, robotic and network technologies were combined to expand the range of functional values of the robots [1]. IoRT was formulated to determine the structure in which sensor data from various sources are incorporated, and then explicated using local and distributed information. Thereafter, the data are used to monitor and verify things in the physical world [2,3]. According to the IEEE Society of Robotics and Automation, a networked robot is described as "a robotic device associated with a communication network through the internet or local area network (LAN) using standard network protocols such as TCP, UDP, or 802.11". Robotic engineering systems are used widely in the industry today. Robotic systems are seen as critical components for humanity's growth in the new digital era. The robotic systems were turned into industrial IoRT applications when technologies of IIoT, AI, robots, intelligent networking, and electric mobility emerged [4]. Robotic things can now be connected to anything and everyone at any time, at any location, via various paths/networks and services. Due to new advancements in intelligent networking.

Edge nodes, which are formed by networked robotic devices, might act as the pillar for IoRT applications in the future [4,5]. The IoT and robotic technologies focus on two goals: (1) to provide information services for detection, sensing, and tracking, and (2) to create movement and interaction behavior. The development of IoRT has been improved due to the combination of the above two goals. According to Vermesan [4], IoRT is defined as an active global network framework with self-adapting and self-configuring characteristics. The characteristics are based upon the standard communication protocol (rules for data access over the network) and the interoperability protocol (multiple system data exchanges). In this technology, to make decisions and act on various sets of rules, virtual and physical robotic things with varying degrees of mobility and autonomy use intelligent interfaces, cognition, and connectivity. IoRT enables the collaboration of people, devices, processes, and technology with actuators and sensors [1]. IoRT performs various functions including human–robot interactions (HRIs) and robotic interaction services (ROIs). A robotic system requires necessary equipment, commonly a microphone, camera, LIDAR, RADAR, and even sensors for performing interactions and reactions [6]. HRI is built into a robot for assisted living facilities, hotels, etc. Due to IoRT and HRI, various robots are deployed to monitor the work continuously. During IoRT communications, the data leakage problem is a big issue for data exchange. Data leakage affects the privacy of customers. For example, a stage subjected to IoRT security risk is associated with the transmission of data to IoRT systems by sensing units and sensors. Sensing units transmit data to the IoRT system to detect physical environments, while sensors give information to the device [7,8].

IoRT devices suffer from heterogeneity, interoperability, time variance, network inactivity, security, multirobot systems, quality of services, precise navigation, and standardization. This article discusses secure communication for IoRT devices to overcome leakage problems. This manuscript provides a review of the IoRT definition and technologies used in the functionalities of IoRT. The abilities of robotic components are very essential for the autonomous behavior of robotic things; various characteristics are illustrated in this article. Various organizations use architectures as per their requirements, and there are various architectures for IoRT devices. In this review, we discuss many IoRT architectures, among which five-tier architectures are the most advanced and feasible for intelligent IoRT devices. This article describes the IoRT key concept, abilities, evolution, applications, latest architectural designs, robotic navigation techniques for obstacle-free navigation, IoRT security, and technical challenges.

The primary findings of this work are as follows:

a. We present a novel taxonomy for Internet of robotic things strategies.
b. We provide an in-depth study and analysis of several IoRT literature approaches and techniques.
c. We briefly illustrate the security methods for IoRT.
d. We highlight some open research problems, as well as futuristic scope, in this active field of research.

Organization of Paper

The organization of the remainder of this paper is depicted in Figure 1. Section 2 gives an overview of IoRT techniques, architecture, and abilities. Section 3 delivers a summary of the recent literature survey. A focus on security and the taxonomy of security threats is presented in Section 4. Section 5 highlights some open research challenges in this active field of research, and Section 6 concludes the paper, along with the future scope.

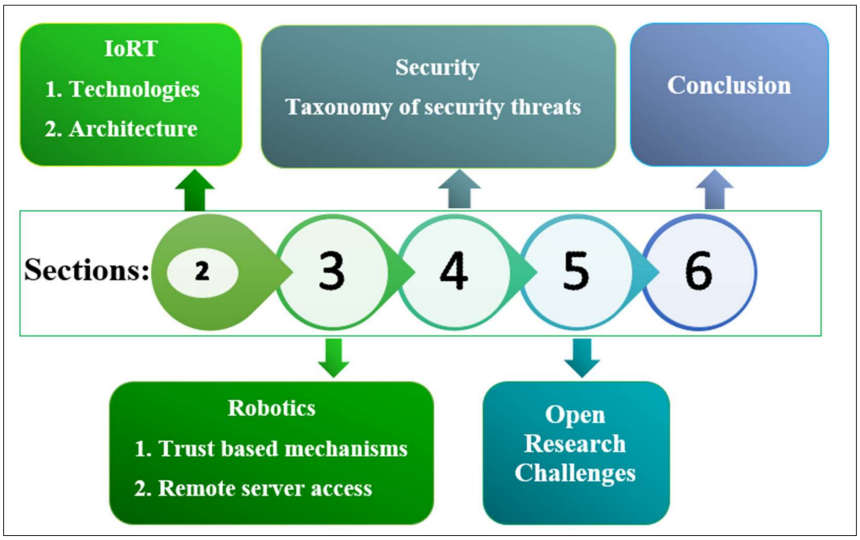

Figure 1. Graphical layout of article.

2. IoRT: An Overview

2.1. Definitions and Concept of IoRT

According to Ray [1], IoRT is described as a global framework for the information sector. IoRT facilitates the improvement of services by robots by affiliating robotic things on the basis of known and emerging compatible information and communication technologies. As per ABI Research [6], IoRT is an intelligent device that monitors procedures and merges sensor data from diverse sources. Robotic devices practice local and distributed intellect to conclude the best way of action.

IoRT is an explicit and dynamic internet framework. The association of IoRT and cloud results in the collection of data from all devices and brings out a report after examining and scrutinizing the data. IoRT allows a large number of distinguishable "things" to share and transfer information with other things over the available Internet or the compatible protocols of the network. Using basic protocols (TCP/IP), IoRT provides a powerful platform for connecting things to assist M2M and M2H data transmission [3,9]. Mark Weiser was the first to mention the idea of IoT in his Scientific American article "The Computer for the 21st Century", based on ubiquitous computing. After that, in 1999, the director of the Auto-ID Center (Kevin Ashton) coined the IoT term. Scientific efforts have enabled the IoRT to pursue real-time decisions by integrating robots and IoT technologies. No study has yet provided a proper and complete definition of IoRT. IoRT is usually proposed as a merger of IoT and robotics (cloud robotics) [3]. IoRT has boosted the IoT application market, as well as advanced the technology, by providing important features such as AI, robotics, and swarm technologies. Earlier robotic technologies relied on computer programs, while more recent robotic technologies rely on AI and ML algorithms, resulting in very effective IoRT technology [6]. Different types of technologies use different types of robots as per their needs. A wired robot is linked to a network (Internet or LAN), and the network (wired or wireless) uses many protocols such as TCP, UDP, and IEEE 802.11 for data transmission among multiple robots. IoRT is a new field, and many more new technologies are currently being developed. The sensing efficiency of robots is enhanced by a network of sensors (installed, repaired, and maintained by robots to increase their reliability and availability). The network sensors result in long-distance robot communications and activity maintenance [3]. A robot is a large-capacity closed system. A cloud robotic system is utilized to overcome the noise, congestion, and time-delay limitations of network robots. In addition to networked and cloud robots, IoRT employs more advanced IoT technologies and robotic devices for expanded capabilities. In addition, depending on the functionality and complexity based on the operability and sophistication of the robot, each robot has a network interface card (NIC) card with a unique NIC address, as well as the remaining

hardware identifiers [10]. IoRT connects a variety of smart devices to a sophisticated IoRT infrastructure that includes cloud and edge technologies. For IoRT computation and control in the cloud, the robotic systems are connected to the cloud via a primary medium known as the "Internet". Cloud robotics is a new branch of robotics based on cloud storage, cloud computing, and other Internet technologies [11–15]. Figure 2 represents the basic ideas of IoRT, IoT, and cloud robotics and mentions their functionalities. Currently, the robotic operating system (ROS) is fully advanced in all aspects.

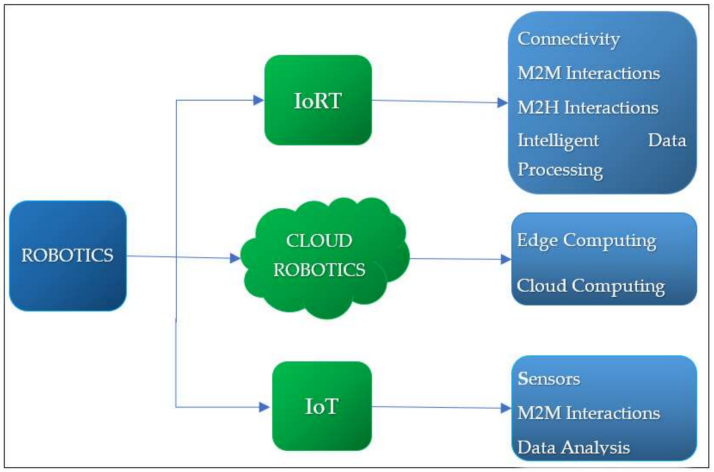

Figure 2. Diagrammatic representation robotics and robotic functionalities.

Hence, there is no threat of complexity in IoRT communication, and a simple API is required for all communication [7,16]. Figure 3, a block diagram of IoRT, mentions the functionalities of robotic things, the latest enabling technologies in robotics, and the application area of IoRT.

Internet of Robotic Things		
	Robots	Smart technologies
Function	ACTING	MONITORING
Technologies	Decision Perception Visualization Multi Agent system Control and Projecting	Cloud Computing IoT Sensing and Actuation Data Analysis Distributed Monitoring Distributed Networking
Applications	Assistive Robot Manipulators Service Robots Mobile Robots	Classrooms Manufacturing Surveillances Smart City

Figure 3. Block diagram of IoRT, including functionalities, technologies, and applications.

2.1.1. How Does Communication Take Place in IoRT?

In H2M interactions, humans provide input to IoT devices in the form of speech, text, and images, among other things. The IoT device, including sensors and actuators, then interprets the input, analyzes it, and reacts to the user via text or a visual display, such as facial recognition or speech recognition. By automating programs, machines may communicate with one another. M2M communication needs machine-level instructions. Communication can happen without human assistance. A point-to-point connection between two network devices is known as an M2M connection, e.g., alerts from a smart washing machine and smart meters. M2H communication is the most prevalent sort of communication utilized when robots assist humans in their regular activities. It is a type of interaction in which humans collaborate with smart systems and other machines to complete a task by using tools or gadgets, e.g., fire alarms and traffic lights [8,17,18]. The IoRT platform maintains the robotic thing's functionalities and technologies. The platform's major capabilities enable robotic things to achieve their main goals, such as communication among robotic things, data flow, IoRT device organization for accessing and maintaining devices, and IoRT device cooperation inside and between the platforms. This is all done to form IoRT applications via the IoRT platform infrastructure. IoRT platform technologies enable elasticity, usability, and productivity [4,9,19,20]. Sharing of data between robots is the responsibility of IoRT platforms to connect data (in the cloud and at control centers) to robotic objects, devices, and people (IoRT environment) [21,22].

2.1.2. How Does Robot-to-Human Communication Take Place?

The digital twin technique is used for robotic virtual commissioning over the lifespan of robotic things. This may be accomplished by combining data from physical IoRT devices with other inputs. All of this leads to real-time optimization, application scenarios, throughput, and possible issues. As a result, the system's virtual representation invokes and strengthens its ability to serve as a real and physical robotic device, as well as an HRI. Enhanced intelligent cognition at the control of IoRT applications enables the combination of AR and VR into human–robot interconnection [3,4].

2.1.3. Security Importance in Robot Communication

Robots are often wirelessly connected to a file server. The network associations create a subnet with the router's static IP address exposed globally, and this is the main reason for robotic data attacks. The server and robots create a subnet of local IP addresses. On the other hand, each robot possesses a static IP address. Distributed ledger technologies (DLTs) are linked with IoRT frameworks and provide systematic data management concerning security, privacy, and safety [10,23–25] The reliability of the IoRT system is increased by hardening end-to-end security, digital identities, services, and mobile data security. This is prompted by robotic cognition from new AI algorithms [4].

2.2. Abilities of IoRT

IoRT depends on the robot functionalities, which are categorized into basic-level abilities, higher-level abilities, and system-level abilities, as given in Figure 4. Some of the characteristics of IoRT are mentioned below, along with the taxonomy of abilities.

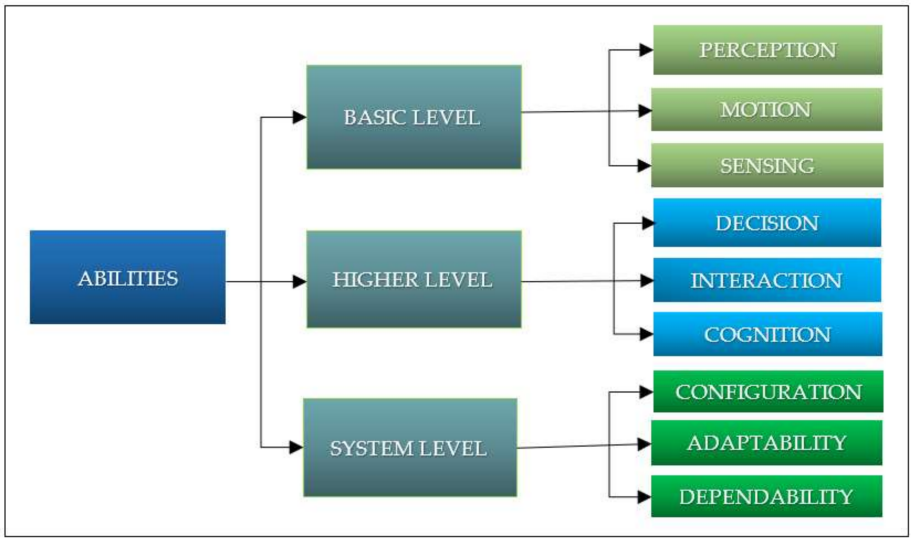

Figure 4. An evaluation of the characteristics of robotic things.

Perception: The performance of the robotic system relies on IoRT sensor information and data analytics technologies. Perception interprets vision, sound, smell, and touch using sensors. Perception is carried out through the utilization of technologies such as software engineering, cloud computing, and big data to accomplish M2M interaction, sensor communications, and AI. IoRT has become more sophisticated as a result of IoT. As a result, robots can sense the real-time world to perform complex tasks [6].

Motion: The important ability of IoRT in all technologies is the ability to travel. The important factor that plays a role in determining the locomotion of devices is mechanical architecture. For navigating the robots, IoT networking also plays an important role [26–28]. A robotic equation of motion defines its motion as a function of time and optional control inputs [29]. Equation (1) is written as

$$F\left(\boldsymbol{q}(t),\ \boldsymbol{q}'\ (t), \boldsymbol{q}''(t),\ \boldsymbol{k}(t), t\right) = 0, \tag{1}$$

where t is the time variable, $\boldsymbol{q}$ is the vector of ordered coordinates, e.g., the vector of combined angles for a manipulator, $\boldsymbol{q}'$ is the first time derivative (velocity) of $\boldsymbol{q}$, $\boldsymbol{q}''$ is the second time derivative (acceleration) of $\boldsymbol{q}$, and k is the vector of control inputs.

Manipulation and sensing: Sensing as a service can be implemented for IoRT and robotic system interactions with IoT devices and people. The responsibility of the IoT is to sense the surroundings. The responsibility for catching, shifting, and directing the shape is taken by robots [13,30,31].

Decisional autonomy (DA): Choosing the best plan for completing a task by a system is called DA. IoT middleware neglects this characteristic and uses API execution (smarts) in its applications, which hides the intrinsic complexity [3,22,30].

Interaction: This is the ability of robots to communicate systematically and cognitively with other systems in an environment. In the industrial context, the interaction potential highlights how IoT technology may boost HRI. For manufacturers, IoT devices can enhance the robustness of HRI [3,22,32,33].

Cognition: This is the ability of IoRT to comprehend a robotic system by sensing the sensor data. IoRT can examine the data from varied systems in the surroundings and take the obligatory way of action. Through this, the intelligence of robots is leveraged [34,35].

Control: Control loops in IoRT can be simply mapped to nearly anything, from virtual things to physical items, from the cloud to multiple networks, granting IoRT autonomy [9,36].

Configurability: Robotic systems are modified for particular tasks or reconfigured for various tasks. IoT is useful in the manufacturing context for software configurability and

the interactive configuration of several computers that contribute different functionality and collaborate to execute complex tasks or jobs. Let f_i be the degrees of freedom of a robot spatial procedure supplied by joint i, and let c_i be the number of constraints given by joint i; it follows that $f_i + c_i = m_i$ for all i [37,38]. Then, Grubler's formulas (Equations (2)–(4)) for the degrees of freedom (dof) of the robot are as follows:

$$dof = m(N-1) - \sum_{i=1}^{j} c_i. \tag{2}$$

$$dof = m(N-1) - \sum_{i=1}^{j} (m - f_i). \tag{3}$$

$$dof = m(N-1-j) + \sum_{i=1}^{j} f_i. \tag{4}$$

The formulas are only retained if all joint constraints are autonomous. If they are not, then the formulas give a lower bound on the number of degrees of freedom. In the above equations, the robot has N links and $N-1$ is the total number of degrees of freedom of the bodies if they are not contrived by joints [39].

Adaptability: The ability of a system to respond to a variety of problems, conditions, etc. is called adaptability. Adaptability adjusts robots in the environment to respond to unexpected circumstances and uncertain human behavior. Adaptability is possible through perception, decision planning, and the configuration of a robot [6].

2.3. Evolution in IoRT

In 1961, robots were first used in the industrial sector to unload parts in a die-casting factory. After 20 years, Japanese manufacturers developed new designs to incorporate robotic manufacturing lines. Robotics and artificial intelligence have advanced rapidly in recent years. Automated machines are now widely utilized in industry, marine exploration, space exploration, the military, and commercialized agriculture to undertake repetitive activities [4]. The IoRT evolution requires many robotic thing activities. The main robotic thing activities for IoRT evolution are secure data, robotic thing cognition, robotic thing collective and collaborative actions, real-time actions, authentic low-latency communication, and energy efficiency. The latest IoRT applications expedite the merging of IoT and autonomous intelligent systems. As a result, collaborative robotic objects may pass on to others, learn autonomously, and have more secure relationships with the environment (people and other things). To improve robotic technologies, future independent IoRT systems may consist of the following qualities: think, learn, sense, act, connect, collaborate, and locate [4,7]. Table 1 illustrates the evolution of IoRT.

Table 1. Evolution of IoRT.

Multidisciplinary Attributes	Evolution In Multidisciplinary Nature of IoRT
Think	Computing, cognition, connectivity, and control
Connect	Connectivity in robotic things and the environment
Locate	High-definition dynamic maps, GPS, GNSS, and location of networks
Learn	AI algorithms are used for learning robotic things
Sense	Collection and processing of data streams from the perception domain radars, LIDARs, cameras, and ultrasound sensors
Collaborate	Activities with their robotic things, autonomous vehicles, edge cloud, etc.
Act	Acting, speed, and stopping

2.4. Applications of IoRT

For the past few years, IoRT has been a rapidly growing field. IoRT applications interlinked with the Internet are found in every field. Examples include transferring resource-intensive activities to the cloud, accessing huge quantities of data, and exchanging data with other robots [3]. Some application fields are manufacturing, agriculture, healthcare, education, and surveillance [24]. The electronics industry is using the IoRT widely. In the modern era, robots do the work of humans in every sector, such as healthcare robots, agricultural robots, and home and hotel robots [23]. IoRT is significantly developing in terms of the revolution of numerous application fields. Hence, new techniques are emerging and required [2]. The human standard of living has been affected by the Internet of robotic technologies in numerous ways. Several manufacturers use robotics to do sophisticated, critical, and difficult jobs, including welding, product assemblage, product testing, packing, and quality control. Preprogrammed robotics has aided and improved industries to never-before-seen levels of precision and 24/7 operational capability. Robotics became more efficient as network technologies were merged, allowing them to perform in unstructured situations [2,40]. Figure 5 describes the overall percentage of IoRT in different fields such as the health sector, agriculture, manufacturing, and surveillance, giving us a brief idea of the latest use of IoRT in all sectors. Figure 6 classifies the robots on the basis of application areas, requirements, and features [4]. The IoRT physical operation classifications used by IoRT include ground and underground, space and planetary exploration, marine and underwater, hybrid location operations, and aerial. Each class has its own set of capabilities [22,41].

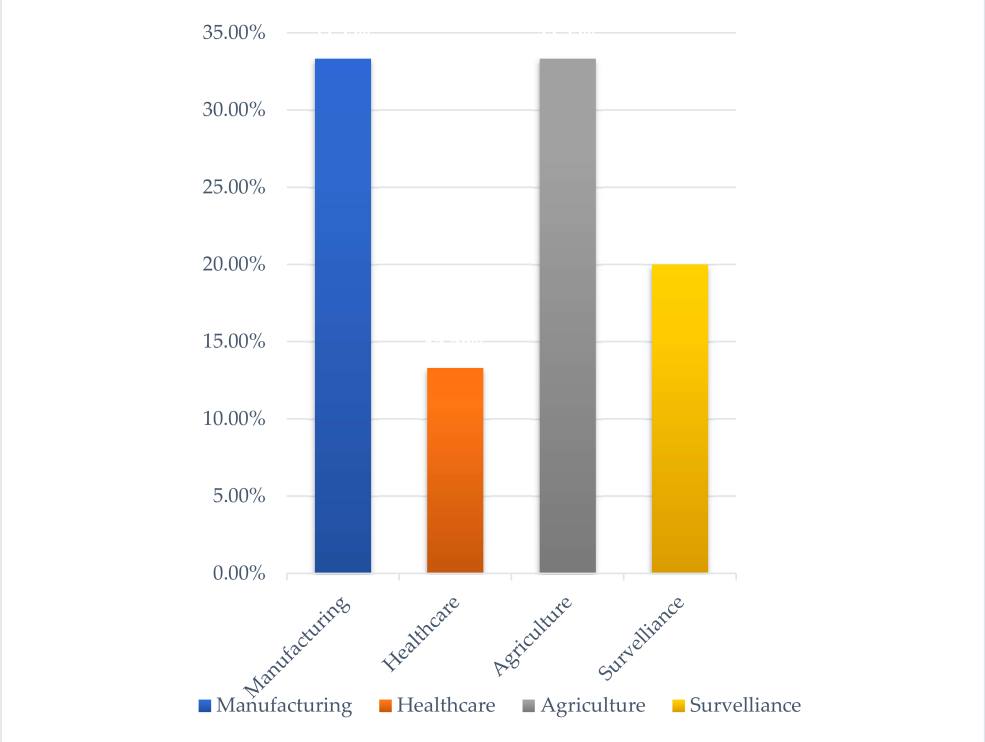

Figure 5. IoRT market usage [1,3].

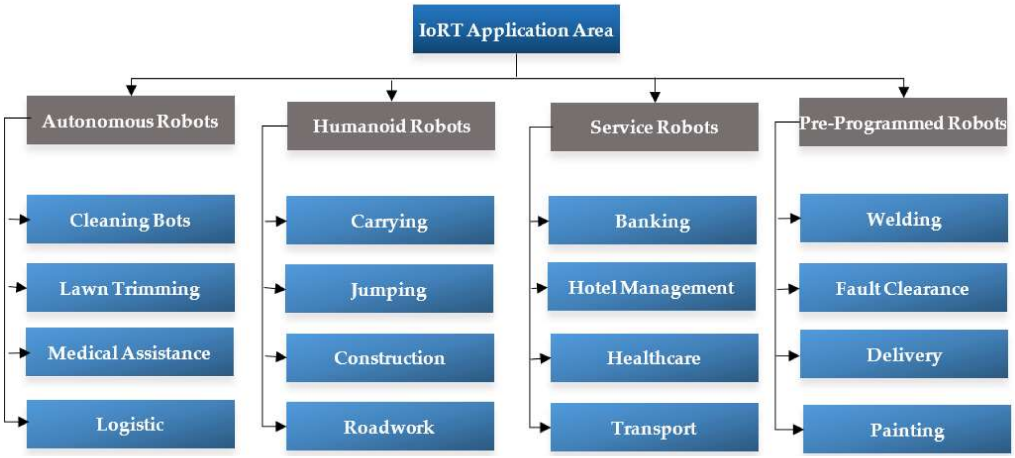

Figure 6. Summary of IoRT application areas.

2.5. Robotic and IoRT Enabling Technologies

Robotic and IoRT technologies are discussed in this section.

2.5.1. Robotic Technologies

The purpose of robotics is to create machines that can support and benefit people. Robotics is the study of creating machines that can replace people and perform human-like tasks. Robotic applications vary according to the environment. In this section, we discuss robots as per IoRT requirements, such as cloud robotics, collaborative robotics, cognitive robotics, fog robotics, network robotics, smart robotics, and swarm robotics [2]. The categorization of different robotic technologies is based on robot functions and numerous interconnected technologies. Cloud robotics uses cloud technology such as processing, storage, and data analysis. HRI is a difficult aspect of robotics, and collaborative robotics aids in the interaction between humans and robots. For intelligent decisions, cognitive behavior is a distinctive trait in robotics, and smart robotics and cognitive robotics play a key part. In communication and computing activities, network robotics and fog robotics are required [1]. Table 2 represents the functions of various robots according to the robotic technologies and purpose.

Table 2. Types of robotics.

Type	Description
Cloud robotics	Robots + cloud infrastructure
Collaborative robotics	Robot–human collaboration
Cognitive robotics	Robots use AI algorithms to learn and respond the complex tasks
Fog robotics	Robots use fog computing to process data and services
Network robotics	To complete a task, multiple robots collaborate and coordinate through networked communication
Smart robotics	AI + robots + ML + DL + cloud computing
Swarm robotics	Multiple robotic systems with physical robots

2.5.2. IoRT Enabling Technologies

IoRT requires many technologies such as sensors and actuators, communication technologies, processing, data fusion techniques, environments, objects, virtual and augmented reality, VR, VC, orchestration, decentralized cloud, adaptation, ML, end-to-end operation, Internet technologies, safety and security frameworks, blockchain, etc. All of these technologies work together to complete various tasks collaboratively. The major IoRT

enabling technologies are defined below, and Table 3 provides a survey on existing robotic technologies.

Actuators and sensors: IoRT and IoT technologies obtain precise and accurate real-time data identification from sensors and actuators. The sensors and actuators are the fundamental gadgets that set the groundwork for the improvement of IoT and robotic systems. The present sensor industry focuses on 2D sensing information. However, with the upcoming IoRT boom, 2D sensing information might change to 4D [1,33].

AR and VR (digital twins): Augmented and virtual reality are counter-reflections of each other. VR provides digital leisure in a real-life scenario. AR provides virtual objects as a cover for the real world. The latest example is Meta's "meta-verse", which merges virtual reality with physical reality and blurs the gap between our interactions online and in real life [4,42].

Voice recognition and control system: For better HRIs, voice control and recognition systems play an important role. For HRI, the IoRT system must be able to communicate between humans and robotic things. Due to the critical nature of VR and VC, such technologies should be versatile and modular to remove the noise using information gathered from the robot's motions and expressions. In addition, the quality of the microphone and speech recognition procedures ought to be able to minimize noise. Multichannel systems with progressive methods such as side-lobe cancellers and feature-space noise clampdown should be included in IoRT systems [4,43].

AI and ML: IoRT technology combines IoT, AI, cloud computing, and other techniques. Due to this, IoRT systems become highly competent in real time and improve the learning experience. These techniques are used in the various layers of the IoRT frameworks to give data and perceptions, as well as maximize the functionality of individual robotic things. Adapting ML and DL techniques and algorithms to IoT-enabled devices enhances the intelligence in IoRT. The primary topics of ML are computational learning and pattern recognition. This provides systems with the capability to acquire data by researching the construction of models to predict and assimilate datasets. In the next few years, ML may be able to replace human learning for data analysis and prediction [4,5].

Connectivity and communication: Communication is the most necessary functionality of the IoRT system. Communication protocols are required to provide layer-by-layer information transmission. IoRT connectivity is preferred over wireless access methods. The new IoRT connectivity strategy permits pooled real-time computing and data stream exchange [19,42,44,45].

Table 3. A summary of enabling technologies in IoRT.

Technologies	Author	Domain	Findings
IoT/IIoT, autonomous robotic system, intelligent connectivity, AI, DL, ML, swarm technology, and VR and AR	Versemen et al. [4]	IoRT—intelligent connectivity and frameworks	• This paper mentions the merging of ML algorithms (CNN and RNN) with IoT and networks for combining the IoRT architecture with edge and fog computing • Role of digital twins, VR and AR, in HRI; collective tasks and efficient data management by swarm technologies and DLT

Table 3. *Cont.*

Technologies	Author	Domain	Findings
Voice recognition and voice control, ML, and security framework	Khalid et al. [3]	IoRT—detailed review	• The author explains how the sensors in different fields are used and how they work • The actuating of sensor data • The improvement in HRI is due to VR and AR; the way in which security attacks occur in networks.
Architecture and network framework, multi-robotic system, computing (edge, fog, cloud), and security	Ilya et al. [46]	IoRT—analysis	• A detailed summary of network layers and their functionality in communication and connectivity • The author mentions the protocols used in different scenarios, as well as the efficiency of multi-robotics

Swarm technology: Swarm robotics may be defined as the integration of multiple robots into a system. Multirobot systems consist of many simple physical robots to perform collective tasks. Combining the swarm robots with IoRT results in scalability, flexibility, and robustness for multirobot systems [1].

2.6. IoRT Architectures

There is no single architectural design that is agreed upon universally because each organization, company, or each user, for that matter, has different requirements. Moreover, the hierarchy of architectures includes three-tier architecture, four-tier architecture, five-tier architecture, and seven-layer architecture. IoRT is an interaction between the physical and digital worlds using sensors, actuators, and robots. In a few years, IoRT has framed so many novel designs, criteria, and platforms. Different architectures of IoRT were illustrated in [1,3,42,47].

2.6.1. Three-Tier Architecture

According to [3], IoRT has a three-tier architecture. The three-tier architecture of IoRT is illustrated in Table 4, featuring the hardware/physical/perception layer, network layer, and application layer, as discussed below.

Hardware layer/physical layer: The physical layer or robotic layer comprises actual IoRT devices. IoRT devices may vary from small sensors to a varied range of robotic devices to produce data [1]. This bottom-most layer comprises various robotic things such as sensors, vehicles, smartphones, home equipment, and actuators. The intelligent IoRT develops a multi-robotic system and delivers innovative features through distributed activities by contacting and integrating them. This layer is in charge of operating in the environment, sensing the data, acquiring information, and transmitting it to the higher layer. Above the robotic layer lies the network layer [48,49].

Table 4. An illustration of various components in a three-tier IoRT architecture.

Layers	Domain
Services and application layer	Smart environments Installation and execution of programs are carried out here by interconnected IoRT
Network and control layer	Routers, switches, local and cloud servers, and network and management protocols
Physical/hardware layer	Sensors, robots, actuators, robot-to-robot communication, and multi-robotic systems

Network layer: The network layer transfers the sensor data between different layers using networks of type 3G, 4G, 5G, RFID, LAN, Bluetooth, and NFC. The network layer contains components that communicate and control operations entailing several robotic things using several protocols. To offer the required connectivity, this layer can comprise routers, controllers, and gateways. Sensor and robot connectivity was explained in [50,51].

Application layer: The application layer is the uppermost layer in the IoRT architecture and defines all applications that use IoRT technology. The application layer interprets and monitors data using various application software. Records are prepared on the basis of data analysis [26]. The physical layer aims to distribute the client experience by investigating the offered sample of robotics-based applications. IoT-connected robots can actively participate in solving a variety of problems in fields [52,53].

2.6.2. Four-Tier Architecture

According to [47], IoRT has a four-tier architecture, divided into four layers for reliable data communication: (i) hardware layer, (ii) support layer, (iii) network layer, and (iv) application layer. The roles of three of the layers were discussed above; the fourth support layer is described below.

Support layer: The support layer provides security in the architecture of IoRT. In a three-tier architecture, data are directly communicated to the network layer, which is susceptible to attacks. The support layer consists of antiviruses and secure computing, overcoming the flaws of the three-layer architecture. Information obtained from the perception layer is sent to the support layer, which provides authenticity to the user. Then, the support layer sends information to the network layer.

2.6.3. Five-Tier Architecture

According to [1], IoRT has a five-tier architecture, which can be further subdivided for a better understanding of IoRT functionalities, thereby minimizing modification requirements to the underlying hardware and software logic: (i) hardware/robotic things layer, (ii) network layer, (iii) Internet layer, (iv) infrastructure layer, and (v) application layer. The five-tier architecture layers are summarized below.

Network layer: The network layer and transport layer are in charge of transmitting data from one end of a network to the other. Both layers are closely linked and are commonly mentioned collectively. Figure 7 depicts a five-tier design, with the network layer referred to as the transport layer.

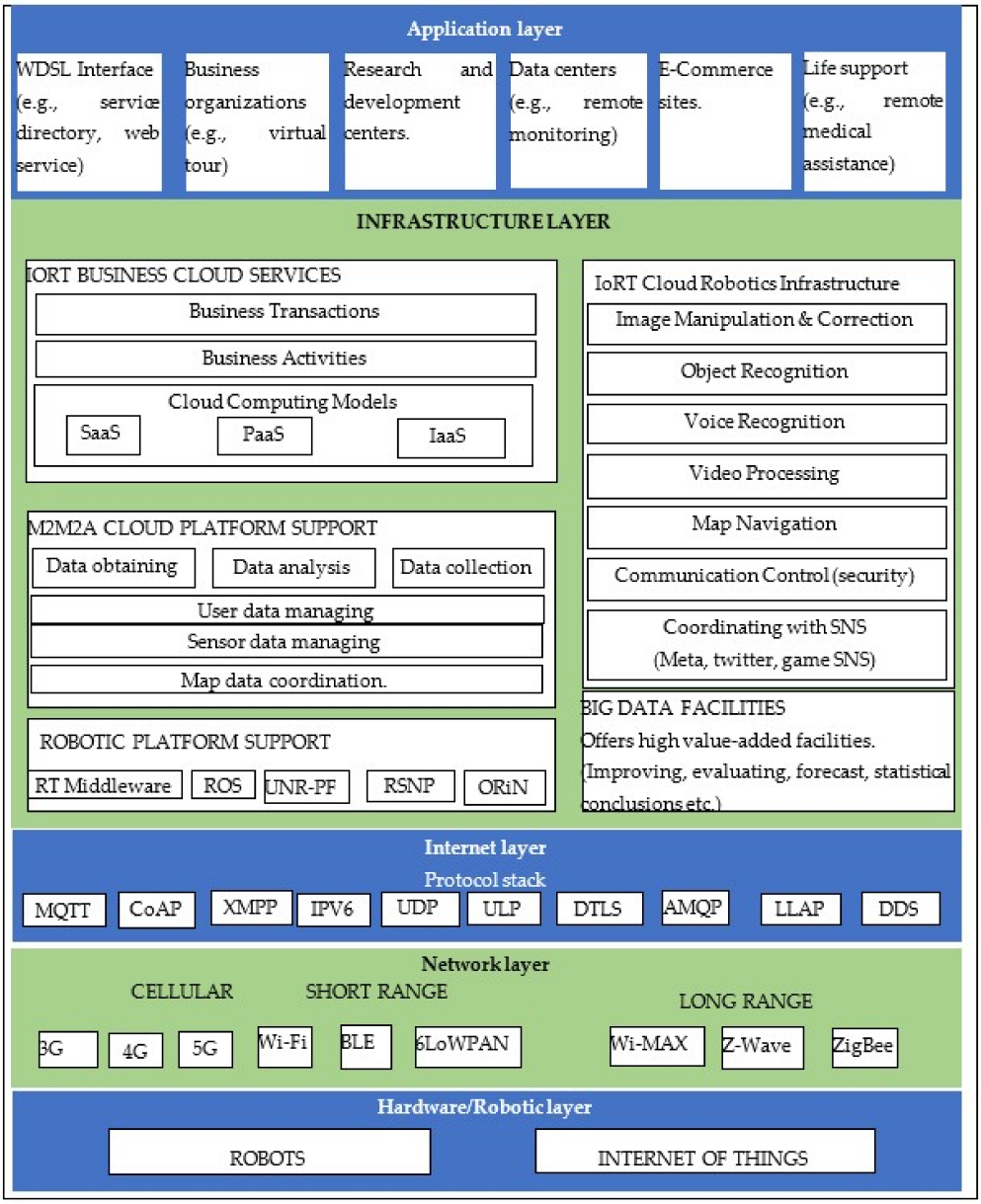

Figure 7. Five-layered IoRT infrastructural architecture.

Internet layer: Network connectivity is an option for facilitating device connectivity and the right to use information from wherever in the world. Internet connectivity provides connectivity for systems and access to data anywhere and anytime. Internet connectedness is regarded as the core part of communication in the IoRT architecture. As the IoRT is constructed on the basis of robotic things, it uses a variety of IoT-defined communication protocols to enable M2M and M2H communication, as well as lightweight processing of information in robotic systems [1,49].

Infrastructure layer: The robotic cloud stack transforms this portion of the architecture into the maximum managed service-centric methods for the cloud, middleware, business processes, and big data. The infrastructure layer is made up of five different but connected modules, including robotic cloud infrastructure, M2M2A cloud infrastructure support, IoT business cloud facilities, big data facilities, and IoT cloud robotics structure. All of these layers are well outlined in the architecture diagram (Figure 8) of IoRT [1,51].

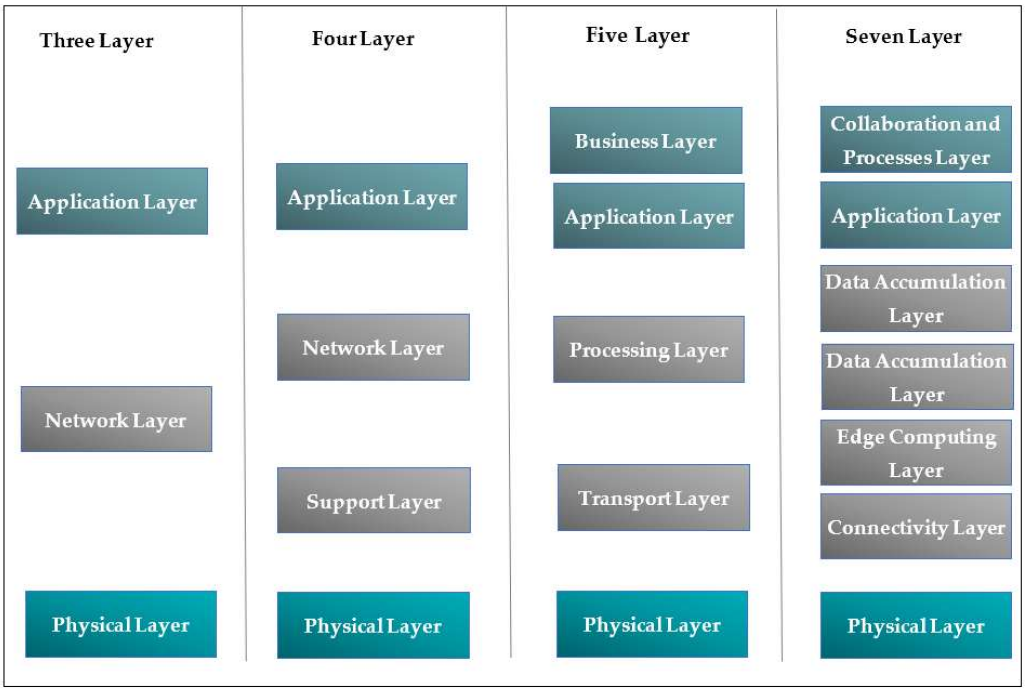

Figure 8. Evolution of IoRT architectures: three-, four-, five-, and seven-tier architectures.

Application layer: The application layer is the uppermost layer in the IoRT architecture. The physical layer aims to distribute the client experience by investigating the offered sample of robotics-based applications. IoT-connected robots can actively participate in solving a variety of problems in fields [53].

A conceptual diagram of the detailed architecture of robotic things and cloud computing is given in Figure 8. This architecture gives an overview of how the robotic platform support gives robot-specific service technologies such as middleware, robotic operating systems, service network protocol, and network interfaces. The M2M2A exchanges the data to the network with resource utilization in the Internet of things business cloud services [1].

2.6.4. Seven-Layer Architecture

According to [47], IoRT has a seven-tier architecture, which breaks down the intricate problem into manageable parts to acquire a complete sense. This IoRT architecture is more realistic rather than just conceptual. Moreover, the data control layer in the IoRT architecture grips data at the edge, fog, and cloud. The seven-layer architecture is summarized below.

Network connectivity layer: The connectivity layer, also called the network layer, performs packet forwarding, requiring virtual connections obtained from infrastructure suppliers to operate virtualization with the required environmental outline, trustworthiness, and efficiency for telecom operators. Studies have illustrated how low-cost IPTV distribution may be achieved via wide-area IP multicast, which tracks on the maximum of a trustworthy virtual network. This layer ensures accurate and consistent data transmission by implementing numerous protocols, switching and routing protocol interpretation, and networking inquiry [16].

Edge computing layer: This layer emphasizes the analysis, processing, and transformation of data.

Data accumulation layer: This layer interprets mobile data as fixed data [54].

Data abstraction layer: This layer is aware of the many languages used to express data where the information is stored. As a result, the layer is able to handle the communication needs of the appropriate information sources. This layer allows multiagent systems entities to access information via Java calls, regardless of the true data representation language. Different application programming interfaces (APIs) plus a new component called the

data access layer make up the DAL. The APIs are a set of Java functions that serve as a link between data stored in one location and the remains of the network. The data abstraction layer uses data stored in various formats to create easy and more performant applications [54].

Collaboration and processes layer: This layer of architecture utilizes and distributes the application information with business processes and people [1].

Each organization requires a specific architecture for the development of a particular product, which means that architectures are used as per the requirement. A detailed architectural diagram is shown in Figure 7, which represents the evolution of architecture. IoRT is made up of several components, such as temperature, motion, light, gas, accelerometer, and pressure sensors. Gateways in IoRT are devices that connect to any network and store data in cloud centers. Analytics or mobile applications analyze the data according to the needs. Several IoRT structures such as three-layer, four-layer, five-layer, and seven-layer architectures are provided to thoroughly analyze these components. Data are acquired from sensors and actuators in the perception layer of a three-layer design. Data are collected and sent to cloud servers for storage and analysis. The application layer is in charge of providing services to users. Layers are further subdivided into a five-layer design for a better understanding of IoRT features. Data are transferred from the physical layer to the network layer in this architecture. A vast volume of data is stored and subsequently analyzed by the processing layer or middleware. Data processed in the application layer are used by users in a human-readable format. The business layer is at the head of IoRT technology, managing the whole system, user policy, profit model, and applications. For a better understanding of IoRT technology, the five layers are divided into seven layers, each of which has been addressed previously. As a result, an evolution of layers occurs as each tiered design is required by the organization. In the evolution diagram, architectural layers are segregated into the next layers for a better understanding of technology [47,48,55,56]. Table 5 mentions the various existing layered architectures. Figure 7 illustrates four-tier, three-tier, five-tier, and seven-tier architectures.

Table 5. A survey of layered IoRT architectures.

Author	IoRT Domain	Architecture
Ray et al. [1]	IoRT—infrastructure	Five-layered
Khalid et al. [3]	IoRT—applications	Three-layered
Anand et al. [6]	Intelligent robotics	Five-layered
Ilya et al. [46]	IoRT—architecture and components	Three-layered
Rana et al. [47]	IoT—energy efficiency and interoperability	Three-, four-, five-, and seven-layered
Sathish et al. [48]	IoRT—security and privacy	Three-layered

3. Related Work

3.1. IoRT: An Outline

IoRT is a fusion of several disciplines such as robotics, cloud computing, AI, and the IoT [2]. IoRT allows robotic objects to participate actively in diverse environments. In diverse surroundings, the robotic objects share data with other robotic devices, IoT devices, and people [3]. IoRT's most recent concepts, technologies, and challenges are useful for the future progress of robotic systems. The application of the IoRT system can be further intensified in industrial production and development, agriculture, and other areas of human importance [6]. Khalid et al. [3] showed a three-layer architecture of IoRT including related technologies such as actuators and sensors, and the Bricks View-RoIS. In addition, the authors of [3] presented HRI challenges and related charts for service robots. In [1], an IoRT architecture was recognized and understood considering five layers: robotic, network, Internet, infrastructure, and application layers. The infrastructure layer includes the robotic and M2M2A cloud platforms, IoT commercial cloud facilities, and IoT

cloud robotics setup. The main capabilities of the five-layered architecture are awareness, interoperability, extensibility, virtualized diversity, dynamic, and self-adaptive behavior. The authors of [22] described the enabling technologies of the robotic system such as robots, AI, ANN, ML, fuzzy logic, and swarm technology, along with their application. The future difficulty of IoRT is data connectivity and security, which require a great deal of attention [7]. Gaze tracking, speech recognition, and biological recognition are HRI issues encountered by IoRT. HRI problems have not yet been put to the test. Instead, HRI issues are mostly being investigated. Computational issues, optimization, security concerns, and ethical concerns are among the IoRT challenges [3,5,11]. In our study, a taxonomy of IoRT, including IoRT technologies and capabilities, is proposed in Figure 9. IoRT, as we know, is a mix of different technologies, such as AI, which aids in intelligent decision making, and cloud robotics, which aids robots by employing cloud infrastructures such as cloud computing, cloud storage, and connectivity technologies. The Internet of things focuses on sensing, monitoring, and tracking, whereas robotics focuses on interactions, navigation, etc. [4]. As seen in the taxonomy graphic, all of these technologies interact with one another.

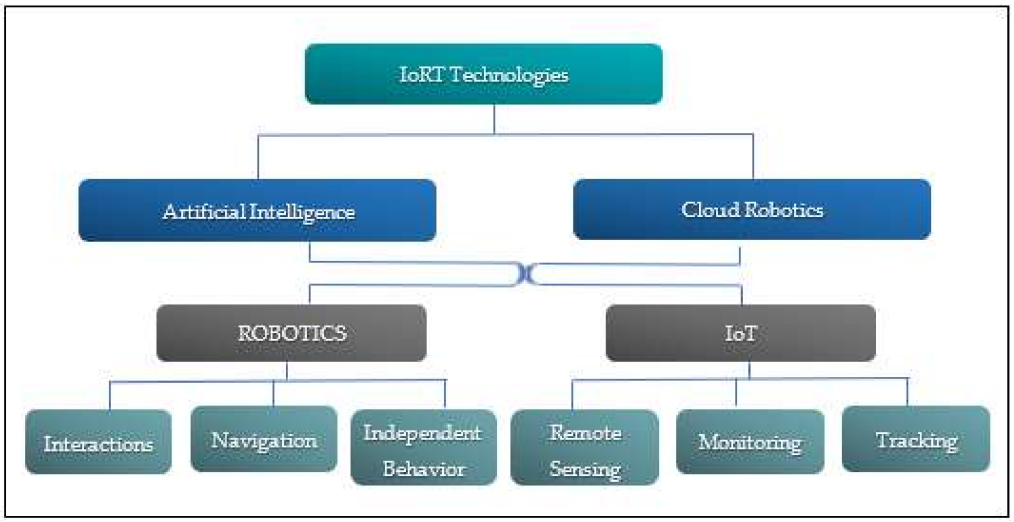

Figure 9. A taxonomy of IoRT technology.

3.2. Secure Communication Mechanisms for IoRT

Secure communication among IoRT devices is one of the primary concerns for industries, as well as society. Commonly used secure communication mechanisms for IoRT are as follows:

- Trust-based mechanisms,
- Encryption-based mechanisms.

3.2.1. Trust-Based Mechanisms for Robotic Devices

A technique for trust-based IoT VANET reveals security issues to make the system secure and trustworthy. The trustworthy cluster is identified as the "cluster head". The cluster head employs statistical models. Trust metrics are calculated by statistical models to identify maliciously infected nodes. RSU is in charge of calculating the clusters in the process. In the process, previous trust values surrounding the nodes are saved in special fixed storage with unique vehicle identification. For analysis of performance, the OMNet++ Simulator is employed. In this mechanism, a Sybil attack is detected by trust-based criteria to provide security. A malicious code is identified as one not being used to earn greater trust levels. It is possible to upgrade the technology to establish a bidirectional clustering technique for VANETs [11]. An enriched, reliable execution environment is employed for IIoT edge devices. The described environment focuses on the real-time and safety features of edge devices. The security features are represented by three CIA elements.

The model demonstrates that security is the most important aspect of most protected systems [9,12–15,40,45].

3.2.2. Encryption-Based Mechanism

Encryption intercepts data using computer algorithms and decodes it using a key provided by the sender. Encryption ensures that confidential information remains confidential, whether it is saved or in transit. Any illegitimate access to the data may produce a jumbled array of bytes. Data security is an issue in cloud computing, and it includes many aspects such as the CIA, surveillance, reliability, and telecommunications. The cloud introduces various types of data security solutions using encryption techniques [17,18,20].

3.3. Robot Navigation Techniques

The ability of a robot to establish its location and orientation within a frame of reference is referred to as robot navigation. Robots use sensors to extract information from their surroundings [57]. A robot navigating in unidentified terrain may encounter an impediment that must be avoided. Probabilistic roadmaps, bumper events, and some algorithms are used for clearing impediments in navigation [58,59]. The robot follows a path with a specific goal, avoiding obstacles along the way. For implementation, a real TurtleBot robot with sensors is used [60]. The navigation model just requires prior information for navigation at the beginning and places the goal. The navigation methods allow the avoidance of both static and dynamic obstacles [61]. A few navigation techniques are mentioned below.

- Robot navigation using fuzzy logic

The robot localization model uses two kinds of controllers, namely, fuzzy logic and pure pursuit. The controllers use labeled data input and output mapping FIS algorithms. The two algorithms control navigation and obstacle avoidance. The former determines the direct path without considering obstacles, while the latter does [62,63]. Using fuzzy logic, the unidentified territory is guessed. The fuzzy logic design, membership functions, and fuzzy rule base are all used in the fuzzy controller. For receiving inputs (minimum range, corresponding angle), the MATLAB-Simulink model is utilized, as well as the gazebo simulator. For pre-navigation, the system does not require data for the obstacles. As a result, a model for navigating robots in an unknown environment is worthy of consideration. With future improvements, the left or right turn can be eliminated [64–66].

- Robot navigation using probabilistic roadmap algorithm

For robot path pursuit, the probabilistic roadmap is implemented; a path is obtained from the beginning to the end of navigation. The phases of the navigation process are as follows:

a. Creating a map of the neighboring world,
b. Storing the map in an intelligible form,
c. Selecting a suitable path from start to finish on the preserved map,
d. Ultimately navigating the robot on the detected path.

The code is written in the MATLAB programming language. To achieve experimental findings, probabilistic roadmaps and path pursuit are employed. In the future, dynamic environments with moving obstacles can be built [58].

- Robot navigation using laser scan matching algorithm

A laser scan is executed for concurrent positioning and mapping in robot steering. The method is fulfilled by using two normal distribution transform algorithms [67]. The laser scan data from the robot are collected and kept using one algorithm. The other algorithm scans the matching and mapped buildings. To avoid obstacles, the laser sensor receives input that is converted into angular velocity. Neural network training parameters are required for scanning acceptable data quality. Laser scan measurements acquired at two places during navigation can be positioned using the rotation and translation of the robot's

two coordinate frames [32,68,69]. Equation (5) displays the 3D plotting (f) mid (x1, y1) and (x2, y2) coordinate frames of the robot.

$$f: \begin{matrix} x2 \\ y2 \end{matrix} = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} \begin{matrix} x1 \\ y1 \end{matrix} + \begin{matrix} \delta x \\ \delta y' \end{matrix} \tag{5}$$

where ϕ is the rotation between the two frames [δx, δy]; T is the transformation between *(x1, y1)* and *(x2, y2)* [67].

- Robot navigation using heuristic functions

Three heuristic functions are used to independently navigate a robot. A navigational map is obtained. Among the three functions, Euclidean distance yields the most nonuniform global path planner time. The octile distance yields the most uniform time throughout the navigation procedure [70,71]. The Manhattan distance between two points $\{p_1(x_1, y_1)$ and $p_2(x_2, y_2)\}$ is given in Equation (6), while the Euclidean distance between two points and octile distances are represented in Equations (7) and (8), respectively.

$$h_1(p_1,\ p_2) = \mathbb{C}(|x_1 - x_2|,\ |y_1 - y_2|), \tag{6}$$

$$h_2(p_1,\ p_2) = \mathbb{C}\sqrt{(x1 - x2)^2 + (y1 - y2)^2}, \tag{7}$$

$$h_3(p_1,\ p_2) = \mathbb{C}(|x_1 - x_2| + |y_1 - y_2|) + (\mathcal{D} - 2\mathbb{C})\,Min(|x_1 - x_2|, |y_1 - y_2|), \tag{8}$$

where C and D are constants.

- Robot navigation using bumper event

The bumper event is used to remove obstacles from the robot's navigation. The bumper event algorithm is applied to the TurtleBot in the gazebo simulator. Bumper and state fields comprise the robot. The bumper sensor coupled with a TurtleBot is used to manage the hurdles. The robot is moved and turned using two different ROS velocities (linear and angular). C++ code is used to implement the algorithm. Because it reduces complexity, this approach is very beneficial in unfamiliar contexts. As a drawback, the algorithm does not give collision-free navigation; hence, the camera gets priority over the bumper sensor for collision-free navigation [23,60,72].

- Vision-based navigation

The robot's gaze direction can be chosen from a variety of directions according to the inclination of angles. More gaze directions necessitate more computational time. For vision-based navigation, an assessment function M is used to calculate the corresponding connection of feature lines between two images as defined in Equation (9) [70,72–75].

$$\mathrm{M} = \alpha \sum_{i=1}^{\mathrm{N1}} \mathrm{Di} + \beta \sum_{i=1}^{\mathrm{N1}} \mathrm{Li} + \gamma \sum_{j=0}^{\mathrm{N_2}} \mathrm{Pj}, \tag{9}$$

where L_i is the absolute variance between two location intervals. D_i is the absolute horizontal variance value of feature lines i in the first image and the parallel candidate image. α, β, and γ are the weights for each term; α + β + γ = 1. P_j is the penalty value when a feature line does not have a communicator in the second image. N1 is the number of feature lines, with contenders in the second image. N2 is the number of feature lines that do not have a communicator in the second image.

3.4. Remote Server Access in IoRT

Computer servers contain important data and software. The servers can be accessed by IoRT devices remotely. However, data exchange between the server and IoRT devices should be secure enough. Local ORM vehicle work is performed on VEC servers. This type of model aids in the execution of tasks such as distributed and trustworthy reputation maintenance, precision reputation updating, and accessible reputation usage [75].

A software update makes use of the MEC for high processing capability in the access network, despite the limited resources of IoT devices. IoT devices can use MEC's software functionalities [44,69]. Remote software update performed over trusted connections is done in five steps [44]:

1 Record the service profile on a cloud server,
2 Request to ASP server for the service package,
3 Send service package using ASP server,
4 Control function codes using the data core network,
5 Update function codes.

To achieve higher energy efficiency in IoT, fog data analytics of data has been stressed more than cloud to minimize latency [11,25,45]. The energy consumption model is concerned with calculating the total quantity of energy utilized by all nodes throughout the transmission. The major reasons for energy usage in an IoRT network are receiving and sending a packet on a trustworthy channel. The IEEE 802.15.4 communication standard accomplishes the entire process. The energy usage at the node p on link e [47], with E for packet rectifying, is given by Equation (10).

$$E^p{}_c = E_l{}^p + E^p{}_{tx} + E^p{}_{rx} + E^p{}_{sl} = \left(t_l{}^p I_l + \frac{(I_{tx} + I_{rx})L}{R} + t^p{}_{sl} I_{sl} \right) V, \quad (10)$$

where V is the node voltage, L is the packet size, and R is the data packet rate. I_l and $E_l{}^p$ are the current drawn and energy consumption during listening. I_{tx} and $E^p{}_{tx}$ are the current drawn and energy consumption during transmitting. I_{rx} and $E^p{}_{rx}$ are the current drawn and energy consumption during receiving. I_{sl} and $E^p{}_{sl}$ are the current drawn and energy consumption during sleeping.

Assumptions:

$$\begin{cases} E^p{}_{tx} = 0; \text{ if } p \text{ is } a \text{ transmitter} \\ E^p{}_{rx} = 0; \text{if } p \text{ is } a \text{ receiver} \end{cases}.$$

Then,

$$E^p{}_c = \begin{cases} \left(t_l{}^p I_l + \frac{(I_{tx})L}{R} + t^p{}_{sl} I_{sl} \right) V, & \text{if } p \text{ is } a \text{ transmitter} \\ \left(t_l{}^p I_l + \frac{(I_{rx})L}{R} + t^p{}_{sl} I_{sl} \right) V, & \text{if } p \text{ is } a \text{ receiver} \end{cases}. \quad (11)$$

4. IoRT Security

Security is a major concern in the connectivity of robotic things [46]. IoRT has significant challenges in terms of security and protection to enable effective collaboration with networks, sensors, and robots. Companies that collect data from robotic systems face the biggest risk from IoRT. Because IoRT networks are still connected to the Internet, new sorts of data breach attacks can be launched against them [3]. There is always a need for communication protocols for data transmission and processing. Therefore, the communication between robotic things must be encrypted, which often does not occur. The Dieffie–Hellman concept is also used for data encryption for the security of a system's communication [24]. To address security concerns, a secure method has been developed that includes a requirement to register IoRT devices using a digital certificate, as well as a user to the cloud server. For a cloud-based IoRT network, we need a three-way (CIA triad) security architecture [76]. To convey information in a secure approach accumulated by robots, secure frameworks are needed with respect to integrity and confidentiality. The IoRT system should be encompassed with physical access security frameworks for verifying data, maintaining trust and privacy, and keeping the data confidential [6,19]. A security taxonomy is given in Figure 10, which describes the generic security threats and threats at the architecture level of IoRT. In addition, the Internet is the basic source of threats and vulnerabilities to robotic things because it is the basic building block of the IoRT device's communication and connection. Non-standardization of IoT technologies has increased

the frequency of security breaches daily, which has increased the vulnerabilities. Some machinery or physical and boot process vulnerabilities are generic issues that apply to the whole IoRT system. Security assaults are also a result of the HRI. IoRT companies supply some security and data protection mechanisms for the safety of user data. However, the effectiveness of protection against vulnerabilities is uncertain and may or may not be guaranteed. Phishing and security breaches are also caused by users' and employees' lack of awareness. IoRT devices are also responsible for a large percentage of denial-of-service assaults (96%) [77]. Threats to the IoRT architectural layers exist as well. Eavesdropping, battery exhaustion, hardware crashes, data breaches, and unauthorized access to IoRT systems are all possible threats to the physical layer. Spoofing, node replication, and fraudulent message bombardment to gateways for denial-of-service assaults are all threats at the network layer. Because this layer connects numerous private LANs, the MAC or network layer is extremely vulnerable to attacks. The risks of brute-force attacks on encrypted data and malicious code at the application layer are also risks to IoRT devices. Thus, there is a necessity for a dependable data transfer service for IoRT [76,78–80]. There are several well-known attacks on ad hoc wireless networks, as listed below, including network attacks (a to d) and the trust model itself (e to g) [33].

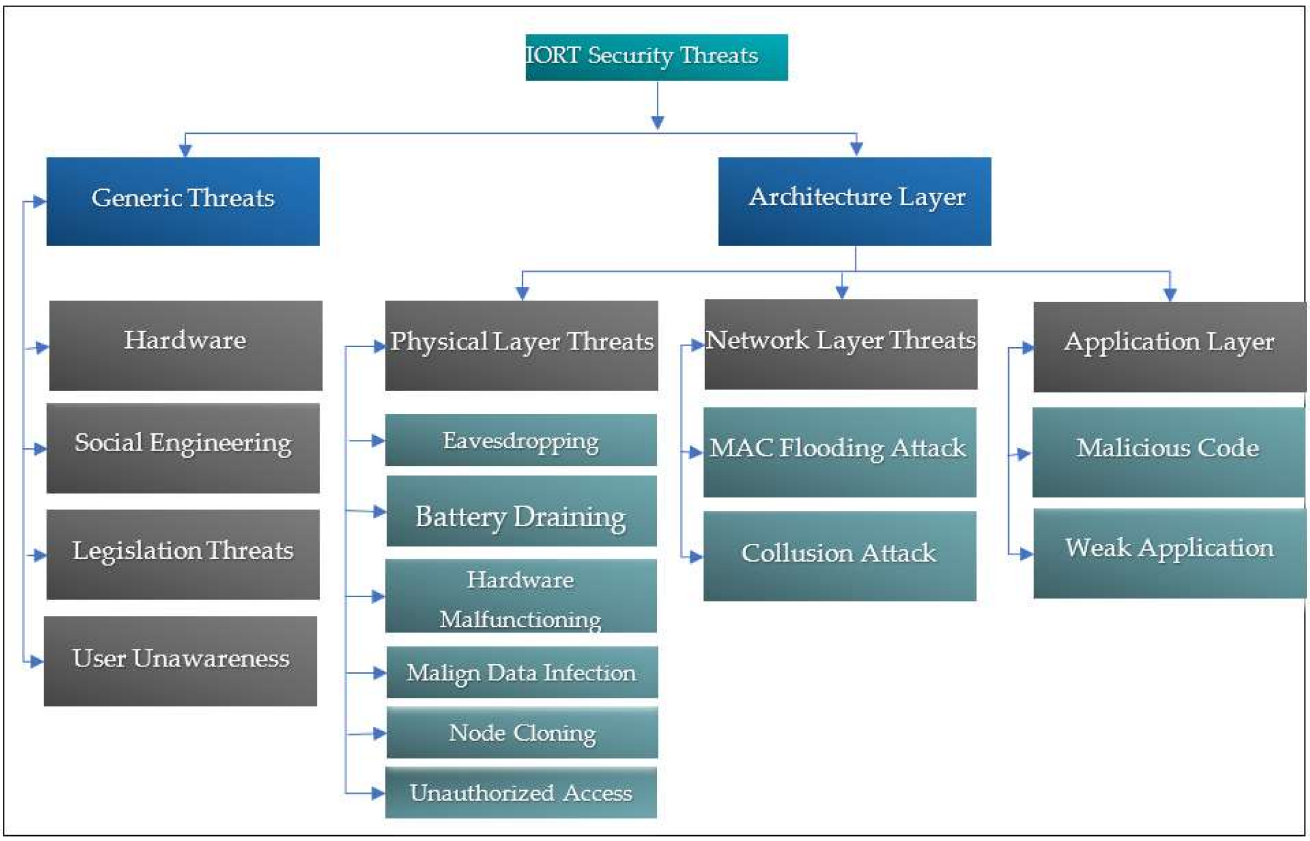

Figure 10. A taxonomy of security threats in data exchange.

a. Packet dropping or modification attacks—black hole and gray hole,
b. Wormhole attack,
c. Sybil attacks,
d. Newcomer attacks,
e. Badmouthing attacks,
f. On–off attacks,
g. Collusion attacks.

IoRT makes use of trust-based techniques to protect the system from vulnerabilities and threats. One of the trust-based mechanisms is threat modeling, which is used to launch

data-sharing security attacks. A threat model is similar to assumptions about an intruder. The threat model's mechanism ensures that the data policy (data should have confidentiality, integrity, and availability) is followed as long as the intruder follows the threat model, which means that, if the threat model is right, it should be able to follow the policy. When security fails, the threat model mechanism is usually to blame. Furthermore, various approaches are used for the screw-up system's policy, such as "recovery questions" [80]. For example, when the threat model goes wrong, it is upgraded over time to ensure its effectiveness. In the 1980s, Kerberos was based on cryptography 56 keys; however, in this cypher-DES, the plausible size is less secure and not reasonable. Later on, it was advanced by applying 256 keys that are more secure [20]. Figure 11 describes the threat model security method in communication to depict secure data flow between the two nodes. The threat model ensures that this communication channel is secure since hackers are always attempting it [81,82]. A threat model is a logical representation of all the data that influence an application's security. Threat modeling (system or data) is the understanding of how a threat actor (external or internal, hostile or abusive) might target a certain asset. Threat modeling differs from application testing [33,58]. The threat model examines the ecosystem, processes, and the circumvention of ecosystem safeguards. If applied effectively, it is one of the finest prospects in solutions, systems, and data security [83]. In successful threat modeling, the following steps are implemented:

a. Uncovering the illegitimate mastermind in the organization,
b. Figuring out the breaking-in method,
c. Choosing the priority method,
d. Portraying the countermeasures,
e. Implementing the solution and testing it.

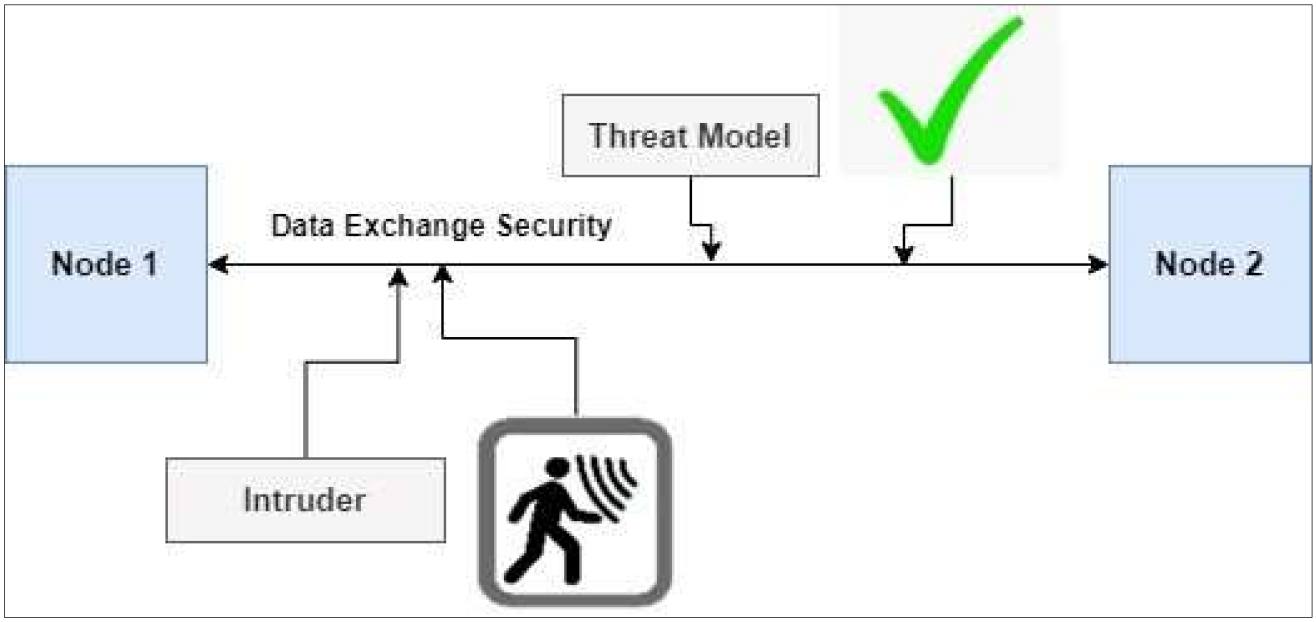

Figure 11. Threat model in secure data exchange.

The term "trust" refers to a set of relationships between the parties involved in a particular protocol. Trust is a belief or trust in other nodes or objects that are based on a defined protocol. Trust is dynamic and is not necessarily transitive. Trust is asymmetrical and dependent on the situation [33]. The computation of trust may be achieved in two ways: distributed or centralized. In distributed systems, we have direct trust, indirect trust, and

hybrid trust. Trust agents can be found from the local standpoint in the network, just as in the centralized system [33,84]. A trusted operator can be denoted by the following formula:

$$T_{i,k}^{m}=M_i\left(T_{i,k}^{n},T_{j,k}^{n}\right). \quad (12)$$

Prefix M models the trust of the *i*-th agent. $T_{i,k}^{n}$ and $T_{j,k}^{n}$ are the trust values of *i*-th and *j*-th agents toward the *k*-th agent. Similarly, *n* is the pre-operation value, and m is the resulting value of the operation [85].

Threats in an application can be mitigated by using countermeasures in threat modeling. Table 6 illustrates various security techniques used in various application domains, with a description. Table 7 illustrates various mitigations (countermeasures) corresponding to various security services. Service-level agreement is a way of transferring risk to another company, such as hosting data in a third-party data center to prevent the risk within the facility. The Internet of things infrastructure, operations, cloud computing, and business technologies all work together and require end-to-end communication mechanisms to assure the security. IoRT is a growing technology, and it is necessary to use security evaluations on Internet-connected platforms, devices, and protocols on a regular basis. Currently, the security measures of products throughout the world contain security-related patterns. The security posture of the IoRT product can be evaluated by using the Common Vulnerability Scoring System (CVSS) standard. CVSS is used to rate the severity of each IoRT product's security vulnerabilities [21]. Some popular secure service products are the following:

i. ARMbed for ARM to develop IoT products,
ii. Brillo and Weave connectivity for IoT/IoRT devices by Google,
iii. Homekit by Apple,
iv. Kura Eclipse offersing application program interface access to hardware interfaces of IoT/IoRT ports,
v. Secure operations for robotic automation by BILA.

Table 6. An illustration of IoRT security techniques.

Security Techniques	Author	Domain	Description
Secure IoRT network for data transmission	Khalid et al. [3]	IoRT—analysis	The paper mentions the security challenges and the reasons for data breaches
Integrity, trust, and confidentiality of secure data.	Ray et al. [1]	IoRT—architecture, technologies	The author discusses the security issues, the trustworthy IoRT VM, and the idea of the protection of secure data.
IoT protocols	Neerendra et al. [59]	Modern communication protocols for IoT	On the basis of six key factors of protocols, IoT protocols are analyzed and compared for optimal communication
Automated key update mechanism for M2M communication, preshared key	Tsai et al. [53]	IoT security enhancement	This paper focuses on a technique for increasing security performance for IoT devices in M2M communication
Privacy filter framework, probabilistic model	Zahir et al. [7]	IoRT—applications	A privacy filter framework is designed for attacks in IoRT-HRI applications
Mobile phone security	Liao et al. [86]	Mobile computing used to evaluate IoT device security	The author discusses the security, accuracy, and limitations of IoT devices and mobile phones
Software-defined network	Waseem et al. [77]	IoT security requirements, challenges	This paper mentions the security challenges, the threats of various layers of the IoT architecture, and approaches to network security

Table 6. *Cont.*

Security Techniques	Author	Domain	Description
Three-way system authentication	Nida et al. [76]	Three-way security structure for cloud-based IoT network	This framework can offer the ability to register IoT devices using digital certificates and users on cloud servers
Cyber-security, encryption	Ilya et al. [46]	IoRT architecture analysis	The author draws attention to the authentication mechanism of data.
Blockchain, software-defined networking	Djamel et al. [87]	IoRT survey—securities, privacy, the blockchain	The effective mechanisms in IoT and the security issues surrounding the safety of systems
UML extension for IoT system security modeling	David et al. [88]	IoT security	According to the author, IoT security is a UML extension; to describe IoT systems, the extension attempts to encapsulate security knowledge
AI, DL algorithms, security	Hui-WU et al. [89]	IoT security—using AI	Different algorithms are employed in this study to improve secure networking
Intelligent community security system (ICSS)	Sathish et al. [90]	IoRT—security and privacy issues	The author discusses various ICSS and their subsystems

Table 7. An illustration of security services and their mitigations.

Security Services	Countermeasures
Authentication	Encryption, trusted server authentication
Authorization	Access controls are required
Data validation	Output encoding
User session management	Encrypted authentic cookies, secure sessions

5. Open Research Challenges

IoRT is a new research field and is in the early stages of development, with many obstacles to overcome. This in-depth and critical investigation of the state of the art in IoRT led to various open research challenges that may be carried out further by researchers in the field of IoRT. The major challenges or gaps that emerged from our study are listed in this section, as well as in Table 8, along with future tasks.

Table 8. A description of IoRT limitations and future tasks.

Author	Paper Focus	Limitations	Future Task
Burghart et al. [35]	Cognitive framework for an intelligent humanoid robotic system	A multimodal fusion of speech and motions	Access to active models through tight integration
Nagarajan et al. [91]	Physical HRI mechanism	One-wheeled, continuous position displacements of ballbot	Laser range finders and stereo cameras are needed for accurate localization
Yoo et al. [51]	Gaze control-based localization for mobile robots	The main issue is how to transmit and display various types of data at the same time	The presented design can be expanded to deal with arbitrarily formed and equally sized objects traveling in peculiar ways
Ariffin et al. [32]	ACI used to build a humanoid-led navigation mobile platform within an obstacle in the surroundings by integrating exterior laser sensing with a humanoid	Security concerns	Path planning and trust-based mechanisms can be involved to overcome navigation and security issues

Computational problems: Due to the competence of IoRT, the transfer of resource-intensive computational tasks for execution to the IoT cloud is possible. However, this

process requires a more rigid and merged architectural framework and can handle several complex issues. To solve the above problem, the system's global area (shared pool) can be supported. The novel shared offloading policy can examine so many factors, such as vast data exchange by several robotic things and real-time retard limits, to conclude the specific task in a fixed order. Moreover, the IoRT should be able to determine the competence of performing tasks within the IoRT or not [1,78].

Data security: The most considerable challenges in IoRT are data processing and security [1]. The IoRT-VM environment must be reliable. Without the assistance of a real robot, a malicious IoRT-VM can effortlessly erode a critical mission. For example, in military exercises, IoRT-approved robotic objects must be able to distinguish between trustworthy IoRT-VM infrastructure and harmful IoRT-VM infrastructure to connect to respectable infrastructure. Robotic objects should avoid the dangerous IoRT-VM infrastructure. To address this issue, three approaches can be used: trust establishment, trust measurement, and reputation-based trust. Future robotic systems must have the confidence to commence computing tasks on IoRT-based clouds. In such a manner, the robotic system's owner or controller may perform verification. It must be ensured that no harmful code is operating in the background of these outsourced activities. Simultaneously, secret data can be permanently kept on IoT-enabled cloud servers with reasonable data being cloned to private cloud servers. To safeguard IoRT data, stringent approaches are required to preserve integrity, trust, and confidentiality [23,40].

Ethical issues: Robotics has been working on resolving this critical problem. Sir Isaac Asimov's three renowned laws should be followed in robotics. A robot may not harm a human person or cause injury to a human being through its actions. Except where such directives clash with the first law, a robot must obey directions given by humans. As long as this shielding does not clash with the first or second laws, a robot must defend its own existence [1,3].

Human–robot interactions: According to recent research, HRI is facing a variety of problems in gaze tracking, voice interactions, and biological recognition, but these problems have not yet been tested and are mostly being studied by researchers. HRI-defined human movements must be adapted by intelligent robots [3,42,92–94].

Emotional robots: Emotional robots, bring their emotional relationships to reality. Recent advances in the field of emotional computing involve intervening in the design and development of "emotional robots" to create an emotional attachment between humans and robots. Nevertheless, there are huge gaps that need to be corrected in the future. The artificial software agents (bots) of "Pepper" are paving the way for emotional interactions to become a reality [36,95,96]

Remote computation problems: Remote working has provided enormous advantages in recent years, especially in the COVID pandemic, as it helps to increase productivity through the best work/life balance. Remote education is also an important advantage of using remote education robots. The educational relationship between people and robots has to be further developed. For a better means of managing industrial operations, additional improvement in such IoRT technology is required [44].

Energy consumption by devices: Industrial technologies are facing a problem of energy demand. In smart environments, the assessment and optimization of energy quality lack a detailed understanding of energy consumption. To address this issue, smart sensor energy utilization should be prioritized [14,47,91].

Data processing: Robotic things are facing enormous IoRT security threats in data exchange. The security of the IoT and the safety of robots are big issues. Large amounts of data are processed in IoRT systems, causing cybersecurity issues. To overcome this, we need an advanced network for IoRT communication to avoid insecure communication between robots and users. The security issue needs to be further investigated [17].

Authorization to industrial IoT: In industrial IoT, data must be shared using the same encrypted protocol with any other compatible system anywhere in the world. There should be proper authorization and privacy for industrial output and management applications

and the internal information of the company. Authorization plays an important role in data security. Authorization is required for sensitive data, as many IoRT programs usually gather data from both labs and engaged clients. This matter should be investigated [11,13,46].

Localization problems: The navigational duties performed by robots remain restricted to motion modeling and position analysis, with little discussion of trajectory planning [27].

Noise problems: Noise is a serious problem in robotic movement, depending on the surface resistance and pushback in the joints.

Accurate localization: The measurements produced by the small sensors that are frequently used with humanoid robots are noisy and inconsistent. As a result, precise navigation, which is thought to be mostly addressed for wheeled robots, remains a difficult challenge for humanoid robots.

6. Conclusions

IoRT technology is relatively a new research area. IoRT has boomed in the market due to its rapid growth and demand in the e-commerce manifesto, the education section, consumer arcade, and research areas in just a few years. The IoRT industry is expected to be worth 21.44 billion USD by 2022, with a compound annual growth rate of 29.7% between 2016 and 2022.

This review focuse on IoRT abilities, evolution, applications, enabling technologies, and IoRT architectures. It was found that collaboration between robots and IoT sensors results in a more advanced IoRT technology. Furthermore, collaboration assists in sensitive data transmission and connectivity. A detailed review of the architectures of IoRT was presented. The study provided an outline of the latest enabling technology of IoRT infrastructure based on M2M2A cloud platforms, IoT business cloud services, and big data analysis. Various methods for navigation of robotic things were reviewed. For robot navigation, different algorithms were studied to overcome the impediment of the robotic surroundings. A security method was presented for secure data transmission between robotic devices. IoRT systems require enormous quantities of data to be transmitted among robots, cloud storage, and other devices. The transmission can lead to data leaks and cyber-attacks. Security issues are becoming more serious. For secure data transmission, secure and trusted data-sharing mechanisms were proposed to eliminate existing research gaps. To handle security threats, future systems can be prepared by considering the proposed security methods and trusted data sharing mechanisms.

Author Contributions: Conceptualization, A.S.; methodology, A.S. and N.K. (Neha Koul); formal analysis and investigation, A.S., N.K. (Neerendra Kumar) and N.K. (Neha Koul); writing—original draft preparation, A.S. and N.K. (Neha Koul); writing—review and editing, A.S., N.K. (Neha Koul), N.K. (Neerendra Kumar), C.V. and Z.I.; resources, N.K. (Neerendra Kumar), C.V. and Z.I. All authors have read and agreed to the published version of the manuscript.

Funding: The work of Chaman Verma, and Zoltán Illés was financially supported by Faculty of Informatics, Eötvös Loránd University (ELTE), Budapest, Hungary.

Data Availability Statement: Not applicable.

Acknowledgments: The work of Chaman Verma was supported under ÚNKP, MIT (Ministry of Innovation and Technology) and the National Research, Development, and Innovation (NRDI) Fund, Hungarian Government. Furthermore, the work of Chaman Verma and Zoltán Illés was supported by the Faculty of Informatics, Eötvös Loránd University (ELTE), Budapest, Hungary.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Acronym	Description	Acronym	Description
IoRT	Internet of robotic things	**AR**	Augmented reality
IoT	Internet of things	**VR**	Virtual reality
AI	Artificial intelligence	**BLE**	Bluetooth Low Energy
ML	Machine learning	**BGAN**	Broadband global area network
VR	Voice recognition	**6LowPAN**	Low-power wireless area network
DT	Distributed technologies	**ROS**	Robotic operating system
DLTs	Distributed ledger technologies	**VC**	Voice control
TCP	Transmission control protocol	**LORA**	Long-range transmission with low power
IP	Internet protocol	**MQTT**	Message Queueing Telemetry Transport
M2H	Machine to human	**CoAP**	Constrained Application Protocol
LAN	Local area network	**XMPP**	Extensible Messaging and Presence Protocol
M2M	Machine to machine	**IPV6**	IP Version 6
UDP	User datagram protocol	**DTLS**	Datagram Transport Layer Security
HRI	Human–robot interfaces	**AMQP**	Advanced Message Queuing Protocol
RoIS	Robotic interface services	**LLAP**	Live Long and Process
M2M2A	Machine to machine to actuator	**DDS**	Data Distribution Service
VANET	Vehicular ad hoc network	**WSDL**	Web Services Description Language
ORM	Online reputation management	**ULP**	Upper Layer Protocol
CIA	Confidentiality, integrity, availability	**SNS**	Simple Notification Service
API	Application programming interface	**UNR-PF**	Open Source of Cloud Robotics
ANN	Artificial neural networks	**RSNP**	Robot Service Network Protocol
VEC	Vehicular edge computing	**ORiN**	Standard Network Interface for Factor Automation
MEC	Mobile edge computing	**RPL**	Robot Programming Language
ASP	Active server pages	**CORPL**	Cobalt-RPL

References

1. Ray, P.P. Internet of Robotic Things: Concept, Technologies, and Challenges. *IEEE Access* **2017**, *4*, 9489–9500. [CrossRef]
2. Simoens, P.; Dragone, M.; Saffiotti, A. The Internet of Robotic Things: A review of concept, added value and applications. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1729881418759424. [CrossRef]
3. Khalid, S. Internet of Robotic Things: A Review. *J. Appl. Sci. Technol. Trends* **2021**, *2*, 78–90. [CrossRef]
4. Vermesan, O.; Bahr, R.; Ottella, M.; Serrano, M.; Karlsen, T.; Wahlstrøm, T.; Sand, H.E.; Ashwathnarayan, M. Internet of Robotic Things Intelligent Connectivity and Platforms. *Front. Robot. AI* **2020**, *7*, 104. [CrossRef]
5. Vandewinckele, L.; Claessens, M.; Dinkla, A.; Brouwer, C.; Crijns, W.; Verellen, D.; Elmpt, W. Van Overview of artificial intelligence-based applications in radiotherapy: Recommendations for implementation and quality assurance. *Radiother. Oncol.* **2020**, *153*, 55–66. [CrossRef]
6. Nayyar, A. Internet of Robotic Things: Driving Intelligent Robotics of Future—Concept, Architecture, Applications and Technologies. In Proceedings of the 2018 4th International Conference on Computing Sciences (ICCS), Jalandhar, India, 30–31 August 2018; 2020; pp. 151–160. [CrossRef]
7. Alsulaimawi, Z. A Privacy Filter Framework for Internet of Robotic Things Applications. In Proceedings of the 2020 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 21–21 May 2020; pp. 262–267, ISBN 9781479966646. [CrossRef]
8. Yuan, B.; Lin, C.; Zhao, H.; Zou, D.; Yang, L.T. Secure Data Transportation with Software-defined Networking and k-n Secret Sharing for High-confidence IoT Services. *IEEE Internet Things J.* **2020**, *7*, 7967–7981. [CrossRef]
9. Cao, Q.H.; Khan, I.; Farahbakhsh, R.; Madhusudan, G.; Lee, G.M.; Crespi, N. A Trust Model for Data Sharing in Smart Cities. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 22–27 May 2016; ISBN 9781479966646.
10. Yfantis, E.A.; Fayed, A. Authentication and secure robot communication. *Int. J. Adv. Robot. Syst.* **2014**, *11*, 10. [CrossRef]
11. Romeo, L.; Petitti, A.; Marani, R.; Milella, A. Internet of Robotic Things in Smart Domains: Applications and Challenges. *Sensors* **2020**, *20*, 3355. [CrossRef]
12. Goh, S. Three architectures for trusted data dissemination in edge computing. *Data Knowl. Eng.* **2006**, *58*, 381–409. [CrossRef]
13. Pinto, S.; Pereira, J.; Cabral, J. IIoTEED: An Enhanced, Trusted Execution Environment for Industrial IoT Edge Devices. *IEEE Internet Comput.* **2017**, *21*, 40–47. [CrossRef]
14. Khan, Z.A.; Herrmann, P.; Ullrich, J.; Voyiatzis, A.G. A trust-based resilient routing mechanism for the internet of things. In Proceedings of the 12th International Conference on Availability, Reliability and Security, Reggio Calabria, Italy, 29 August–1 September 2017; Volume Part F1305. [CrossRef]
15. Chahal, R.K.; Kumar, N.; Batra, S. Trust management in social Internet of Things: A taxonomy, open issues, and challenges. *Comput. Commun.* **2019**, *150*, 13–46. [CrossRef]

16. Zhu, Y.; Sampath, R.Z.; Jennifer, R. Cabernet: Connectivity architecture for better network services Cabernet: Connectivity Architecture for Better Network Services. In Proceedings of the 2008 ACM CoNEXT Conference, Madrid, Spain, 9–12 December 2008. [CrossRef]
17. Zhao, F.; Li, C.; Liu, C.F. A cloud computing security solution based on fully homomorphic encryption. In Proceedings of the 16th International Conference on Advanced Communication Technology, Pyeongchang, Korea, 16–19 February 2014; pp. 485–488. [CrossRef]
18. Hemalatha, N.; Jenis, A.; Cecil Donald, A.; Arockiam, L. A Comparative Analysis of Encryption Techniques and Data Security Issues in Cloud Computing. *Int. J. Comput. Appl.* **2014**, *96*, 1–6. [CrossRef]
19. Gulzar, M.; Abbas, G. Internet of Things Security: A Survey and Taxonomy. In Proceedings of the 2019 International Conference on Engineering and Emerging Technologies (ICEET), Lahore, Pakistan, 21–22 February 2019; pp. 1–6.
20. Kumar, N.; Chaudhary, P. Performance evaluation of encryption/decryption mechanisms to enhance data security. *Indian J. Sci. Technol.* **2016**, *9*, 1–10. [CrossRef]
21. Alqahtani, A.; Li, Y.; Patel, P.; Solaiman, E.; Ranjan, R. End-to-End Service level Agreement Specification for IoT Applications. In Proceedings of the 2018 International Conference on High Performance Computing & Simulation (HPCS), Orleans, France, 16–20 July 2018. [CrossRef]
22. Wu, H.; Han, H.; Wang, X.; Sun, S. Research on Artificial Intelligence Enhancing Internet of Things Security: A Survey. *IEEE Access* **2020**, *8*, 153826–153848. [CrossRef]
23. Abbasi, M.H. Deep Visual Privacy Preserving for Internet of Robotic Things. In Proceedings of the 2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI), Tehran, Iran, 28 February–1 March 2019; pp. 292–296.
24. Su, J. Authentication and Encryption for a Robotic Ad Hoc Network using Identity-Based Cryptography. In Proceedings of the 2018 4th International Conference on Big Data Innovations and Applications (Innovate-Data), Barcelona, Spain, 6–8 August 2018. [CrossRef]
25. Kumar, N.; Jamwal, P. Analysis of Modern Communication Protocols for IoT applications. *Karbala International Journal of Modern Science* **2021**, *7*, 390–404. [CrossRef]
26. Zhong, C.; Zhu, Z.; Huang, R. Study on the IOT Architecture and Access Technology. In Proceedings of the 2017 16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science (DCABES), Anyang, China, 13–16 October 2017. [CrossRef]
27. Li, Y.; Tan, D.; Wu, Z.; Zhong, H.; Zu, D. Dynamic stability analyses based on ZMP of a wheel-based humanoid robot. In Proceedings of the 2006 IEEE International Conference on Robotics and Biomimetics, Kunming, China, 17–20 December 2006; pp. 1565–1570. [CrossRef]
28. Yanjie, L.; Zhenwei, W.; Hua, Z. The dynamic stability criterion of the wheel-based humanoid robot based on ZMP modeling. In Proceedings of the 2009 Chinese Control and Decision Conference, Guilin, China, 17–19 June 2009; pp. 2349–2352. [CrossRef]
29. URL-Robot Motion. Available online: https://scaron.info/robot-locomotion/equations-of-motion.html (accessed on 2 May 2022).
30. Rostami, M.; Koushanfar, F.; Karri, R. A primer on hardware security: Models, methods, and metrics. *Proc. IEEE* **2014**, *102*, 1283–1295. [CrossRef]
31. Ariffin, I.M.; Rasidi, A.I.H.M.; Yussof, H.; Mohamed, Z.; Miskam, M.A.; Amin, A.T.M.; Omar, A.R. Sensor Based Mobile Navigation Using Humanoid Robot Nao. *Procedia Comput. Sci.* **2015**, *76*, 474–479. [CrossRef]
32. Ariffin, I.M.; Baharuddin, A.; Atien, A.C.; Yussof, H. Real-Time Obstacle Avoidance for Humanoid-Controlled Mobile Platform Navigation. *Procedia Comput. Sci.* **2017**, *105*, 34–39. [CrossRef]
33. Muzammal, S.M.; Murugesan, R.K.; Jhanjhi, N.Z. A Comprehensive Review on Secure Routing in Internet of Things: Mitigation Methods and Trust-Based Approaches. *IEEE Internet Things J.* **2021**, *8*, 4186–4210. [CrossRef]
34. Munawar, A.; De Magistris, G.; Pham, T.H.; Kimura, D.; Tatsubori, M.; Moriyama, T.; Tachibana, R.; Booch, G. MaestROB: A Robotics Framework for Integrated Orchestration of Low-Level Control and High-Level Reasoning. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 527–534. [CrossRef]
35. Burghart, C.; Mikut, R.; Stiefelhagen, R.; Asfour, T.; Holzapfel, H.; Steinhaus, P.; Dillmann, R. A cognitive architecture for a humanoid robot: A first approach. In Proceedings of the 5th IEEE-RAS International Conference on Humanoid Robots, Tsukuba, Japan, 5 December 2005; pp. 357–362. [CrossRef]
36. Pessoa, L. Do Intelligent Robots Need Emotion? *Trends Cogn. Sci.* **2017**, *21*, 817–819. [CrossRef]
37. Zaraki, A.; Pieroni, M.; De Rossi, D.; Mazzei, D.; Garofalo, R.; Cominelli, L.; Dehkordi, M.B. Design and evaluation of a unique social perception system for human-robot interaction. *IEEE Trans. Cogn. Dev. Syst.* **2017**, *9*, 341–355. [CrossRef]
38. Chen, F.; Cao, L.; Tian, M.; Du, G. Research and Improvement of Competitive Double Arm Wheeled Humanoid Robot. In Proceedings of the 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 27–29 September 2020; pp. 599–601. [CrossRef]
39. Mordern Robotics. Available online: https://modernrobotics.northwestern.edu/nu-gm-book-resource/2-2-degrees-of-freedom-of-a-robot/ (accessed on 2 May 2022).
40. Mohammadi, V.; Rahmani, A.M.; Darwesh, A.M.; Sahafi, A. Trust-based recommendation systems in Internet of Things: A systematic literature review. *Hum. Cent. Comput. Inf. Sci.* **2019**, *9*, 21. [CrossRef]

41. Liu, R.; Yu, G.; Qu, F.; Zhang, Z. Device-to-Device Communications in Unlicensed Spectrum: Mode Selection and Resource Allocation. *IEEE Access* **2016**, *4*, 4720–4729. [CrossRef]
42. Razafimandimby, C.; Loscri, V.; Vegni, A.M. A neural network and IoT based scheme for performance assessment in Internet of Robotic Things. In Proceedings of the 2016 IEEE first international conference on internet-of-things design and implementation (IoTDI), Berlin, Germany, 4–8 April 2016. [CrossRef]
43. Möller, R.; Furnari, A.; Battiato, S.; Härmä, A.; Farinella, G.M. A survey on human-aware robot navigation. *Rob. Auton. Syst.* **2021**, *145*, 103837. [CrossRef]
44. Kim, D.; Kim, S.; Park, J.H. Remote Software Update in Trusted Connection of Long Range IoT Networking Integrated with Mobile Edge Cloud. *IEEE Access* **2018**, *6*, 66831–66840. [CrossRef]
45. Zhu, C.; Rodrigues, J.J.P.C.; Leung, V.C.M.; Shu, L.; Yang, L.T. Trust-based communication for the industrial internet of things. *IEEE Commun. Mag.* **2018**, *56*, 16–22. [CrossRef]
46. Afanasyev, I.; Mazzara, M.; Chakraborty, S.; Zhuchkov, N.; Maksatbek, A.; Yesildirek, A.; Kassab, M.; Distefano, S. Towards the Internet of Robotic Things: Analysis, Architecture, Components and Challenges. In Proceedings of the 2019 12th International Conference on Developments in eSystems Engineering (DeSE), Kazan, Russia, 7–10 October 2019; pp. 3–8. [CrossRef]
47. Rana, B. A systematic survey on internet of things: Energy efficiency and interoperability perspective. *Trans. Emerg. Telecommun. Technol.* **2020**, *32*, e4166. [CrossRef]
48. Kumar, J.S. A Survey on Internet of Things: Security and Privacy Issues. *Int. J. Comput. Appl.* **2014**, *90*, 20–26.
49. Mukherjee, A. Physical-Layer Security in the Internet of Things: Sensing and Communication Confidentiality Under Resource Constraints. *Proc. IEEE* **2015**, *103*, 1747–1761. [CrossRef]
50. Dizdarević, J.; Carpio, F.; Jukan, A.; Masip-Bruin, X. A survey of communication protocols for internet of things and related challenges of fog and cloud computing integration. *ACM Comput. Surv.* **2019**, *51*, 1–29. [CrossRef]
51. Jeong, J.; Yang, J.; Baltes, J. Robot magic show as testbed for humanoid robot interaction. *Entertain. Comput.* **2022**, *40*, 100456. [CrossRef]
52. Althumali, H.; Othman, M.; Member, S. A Survey of Random Access Control Techniques for Machine-to-Machine Communications in LTE/LTE-A Networks. *IEEE Access* **2018**, *6*, 74961–74983. [CrossRef]
53. Tsai, W.; Wang, T. An Automatic Key-update Mechanism for M2M Communication and IoT Security Enhancement. In Proceedings of the 2020 IEEE International Conference on Smart Internet of Things (SmartIoT), Beijing, China, 14–16 August 2020; pp. 354–355. [CrossRef]
54. Batet, M.; Gibert, K.; Valls, A. The data abstraction layer as knowledge provider for a medical multi-agent system. In *AIME Workshop on Knowledge Management for Health Care Procedures*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 87–100. [CrossRef]
55. Hendrich, N.; Bistry, H.; Zhang, J. Architecture and Software Design for a Service Robot in an Elderly-Care Scenario. *Engineering* **2015**, *1*, 27–35. [CrossRef]
56. Ankele, R.; Marksteiner, S.; Nahrgang, K. Requirements and Recommendations for IoT/IIoT Models to automate Security Assurance through Threat Modelling, Security Analysis and Penetration Testing. In Proceedings of the 14th International Conference on Availability, Reliability and Security, Canterbury, UK, 26–29 August 2019.
57. Kumar, N.; Vámossy, Z.; Szabó-Resch, Z.M. Robot Path Pursuit Using Probabilistic Roadmap. In Proceedings of the 2016 IEEE 17th International Symposium on Computational Intelligence and Informatics (CINTI), Budapest, Hungary, 17–19 November 2016; pp. 139–144.
58. Kumar, N.; Vámossy, Z.; Szabó-Resch, Z.M. Robot Obstacle Avoidance Using Bumper Event. In Proceedings of the 2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 12–14 May 2016; pp. 485–490.
59. Kishi, T.; Shimomura, S.; Futaki, H.; Yanagino, H.; Yahara, M.; Cosentino, S.; Nozawa, T.; Hashimoto, K.; Takanishi, A. Development of a Humorous Humanoid Robot Capable of Quick-and-Wide Arm Motion. *IEEE Robot. Autom. Lett.* **2016**, *1*, 1081–1088. [CrossRef]
60. Ravankar, A.; Ravankar, A.A.; Kobayashi, Y.; Hoshino, Y.; Peng, C.C. Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. *Sensors* **2018**, *18*, 3170. [CrossRef]
61. Kumar, N.; Takács, M.; Vámossy, Z. Robot Navigation in Unknown Environment using Fuzzy Logic. In Proceedings of the 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 26–28 January 2017; pp. 279–284.
62. Rath, A.K.; Parhi, D.R.; Das, H.C.; Muni, M.K.; Kumar, P.B. Analysis and use of fuzzy intelligent technique for navigation of humanoid robot in obstacle prone zone. *Def. Technol.* **2018**, *14*, 677–682. [CrossRef]
63. Muni, M.K.; Parhi, D.R.; Kumar, P.B.; Sahu, C.; Kumar, S. Towards motion planning of humanoids using a fuzzy embedded neural network approach. *Appl. Soft Comput.* **2022**, *119*, 108588. [CrossRef]
64. Kashyap, A.K.; Parhi, D.R.; Pandey, A. Multi-objective optimization technique for trajectory planning of multi-humanoid robots in cluttered terrain. *ISA Trans.* **2021**, *125*, 591–613. [CrossRef]
65. Kumar, N.; Vámossy, Z. Laser Scan Matching in Robot Navigation. In Proceedings of the 2018 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 17–19 May 2018; pp. 241–246.
66. Kastner, L.; Lambrecht, J. Augmented-Reality-Based Visualization of Navigation Data of Mobile Robots on the Microsoft Hololens—Possibilities and Limitations. In Proceedings of the 2019 IEEE International Conference on Cybernetics and Intelligent

Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Bangkok, Thailand, 18–20 November 2019; pp. 344–349. [CrossRef]
67. Oh, H.S.; Lee, C.W.; Mitsuru, I. Navigation control of a mobile robot based on active vision. In Proceedings of the IECON'91: 1991 International Conference on Industrial Electronics, Control and Instrumentation, Kobe, Japan, 28 October–1 November 1991; Volume 2, pp. 1122–1126. [CrossRef]
68. Kumar, N.; Vámossy, Z.; Szabó-resch, Z.M. Heuristic Approaches in Robot Navigation. In Proceedings of the 2016 IEEE 20th Jubilee International Conference on Intelligent Engineering Systems (INES), Budapest, Hungary, 30 June–2 July 2016; pp. 219–222.
69. Tang, L. Vision Based Navigation for Mobile Robots in Indoor Environment bly Teaching and Playing-back Scheme. In Proceedings of the 2001 ICRA, IEEE International Conference on Robotics and Automation, Seoul, Korea, 21–26 May 2001; pp. 3072–3077.
70. Al-Mutib, K. Smart stereovision based gaze control for navigation in low-feature unknown indoor environments. In Proceedings of the 2014 5th International Conference on Intelligent Systems, Modelling and Simulation, Langkawi, Malaysia, 27–29 January 2014; 2015; Volume 2015, pp. 121–126. [CrossRef]
71. Yoo, J.K.; Kim, J.H. Gaze Control-Based Navigation Architecture with a Situation-Specific Preference Approach for Humanoid Robots. *IEEE/ASME Trans. Mechatron.* **2015**, *20*, 2425–2436. [CrossRef]
72. Adachi, Y.; Tsunenari, H.; Matsumoto, Y.; Ogasawara, T. Guide robot's navigation based on attention estimation using gaze information. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, 28 September–2 October 2004; Volume 1, pp. 540–545. [CrossRef]
73. Awan, K.A.; Ud, I.; Senior, D.I.N.; Almogren, A.; Member, S.; Fellow, M.G.; Khan, S. StabTrust—A Stable and Centralized Trust-based Clustering Mechanism for IoT enabled Vehicular Ad-hoc Networks. *IEEE Access* **2020**, *8*, 21159–21177. [CrossRef]
74. Zeeshan, N.; Member, M.R. Three-way Security Framework for Cloud based IoT Network. In Proceedings of the 2019 International Conference on Computing, Electronics & Communications Engineering (iCCECE), London, UK, 22–23 August 2019; pp. 183–186.
75. Iqbal, W.; Abbas, H.; Daneshmand, M.; Rauf, B.; Abbas, Y. An In-Depth Analysis of IoT Security Requirements, Challenges and their Countermeasures via Software Defined Security. *IEEE Internet Things J.* **2020**, *7*, 10250–10276. [CrossRef]
76. Wu, A.; Guo, J.; Yang, P. Research on Data Sharing Architecture for Ecological Monitoring Using Iot Streaming Data. *IEEE Access* **2020**, *8*, 195385–195397. [CrossRef]
77. Rostami, M.; Koushanfar, F.; Rajendran, J.; Karri, R. Hardware security: Threat models and metrics. In Proceedings of the 2013 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), San Jose, CA, USA, 18–21 November 2013; pp. 819–823. [CrossRef]
78. Hussain, S.; Erwin, H.; Dunne, P. Threat modeling using Formal Methods: A New Approach to Develop Secure Web Applications. In Proceedings of the 2011 7th International Conference on Emerging Technologies, Islamabad, Pakistan, 5–6 September 2011.
79. Bradbury, M.; Jhumka, A.; Watson, T.I.M.; Burton, J.; Butler, M.; Data, M.M. Threat-modeling-guided Trust-based Task Offloading for Resource-constrained Internet of Things. *ACM Trans. Sens. Netw.* **2022**, *18*, 1–41. [CrossRef]
80. Maciel, R.; Araujo, J.; Dantas, J.; Melo, C.; Guedes, E.; Maciel, P. Impact of a DDoS Attack on Computer Systems: An Approach Based on an Attack Tree Model. In Proceedings of the 2018 Annual IEEE International Systems Conference (SysCon), Vancouver, BC, Canada, 23–26 April 2018.
81. Fei, Y.; Ning, J.; Jiang, W. A quantifiable Attack-Defense Trees model for APT attack. In Proceedings of the 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 12–14 October 2018; pp. 2303–2306. [CrossRef]
82. Trček, D. A formal apparatus for modeling trust in computing environments. *Math. Comput. Model.* **2009**, *49*, 226–233. [CrossRef]
83. Liao, B.I.N.; Ali, Y.; Nazir, S. Security Analysis of IoT Devices by Using Mobile Computing: A Systematic Literature Review. *IEEE Access* **2020**, *8*, 120331–120350. [CrossRef]
84. Kouicem, D.E.; Bouabdallah, A.; Lakhlef, H.P.T. Internet of things security: A top-down survey. *Comput. Netw.* **2018**, *141*, 199–221. [CrossRef]
85. Robles-Ramirez, D.A.; Escamilla, P.J.; Tryfonas, T. IoTsec: UML extension for Internet of things systems security modelling. In Proceedings of the 2017 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE), Cuernavaca, Mexico, 21–24 November 2017. [CrossRef]
86. Huang, X.; Yu, R.; Kang, J. Distributed Reputation Management for Secure and Efficient Vehicular Edge Computing and Networks. *IEEE Access* **2017**, *5*, 25408–25420. [CrossRef]
87. Tandon, A.; Srivastava, P. Trust-based Enhanced Secure Routing against Rank and Sybil Attacks in IoT. In Proceedings of the 2019 Twelfth International Conference on Contemporary Computing (IC3), Noida, India, 8–10 August 2019; pp. 1–7. [CrossRef]
88. Nagarajan, U.; Kantor, G.; Hollis, R. The ballbot: An omnidirectional balancing mobile robot. *Int. J. Rob. Res.* **2014**, *33*, 917–930. [CrossRef]
89. Gurunath, R.; Agarwal, M.; Nandi, A.; Samanta, D. An Overview: Security Issue in IoT Network. In Proceedings of the 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 30–31 August 2018; pp. 1–4.
90. Shiomi, M.; Shatani, K.; Minato, T.; Ishiguro, H. How Should a Robot React before People's Touch?: Modeling a Pre-Touch Reaction Distance for a Robot's Face. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3773–3780. [CrossRef]
91. Aucouturier, J.-J. Cheek to Chip: Dancing Robots and AI's Future. *IEEE Intell. Syst.* **2008**, *23*, 74–84. [CrossRef]

92. Velásquez, J.D. When robots weep: Emotional memories and decision-making. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI-98)*; The AAAI Press: Menlo Park, CA, USA, 1998; pp. 70–75. Available online: https://www.aaai.org/Papers/AAAI/1998/AAAI98-010.pdf (accessed on 2 May 2022).
93. Dorner, D.; Hille, K. Artificial Souls: & Motivated Emotional Robots. In Proceedings of the 1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century, Vancouver, BC, Canada, 22–25 October 1995; pp. 3828–3832.
94. Hornung, A.; Wurm, K.M.; Bennewitz, M. Humanoid robot localization in complex indoor environments. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 1690–1695. [CrossRef]
95. Taylor, C.; Ward, C.; Sofge, D.; Lofaro, D.M. LPS: A Local Positioning System for Homogeneous and Heterogeneous Robot-Robot Teams, Robot-Human Teams, and Swarms. In Proceedings of the LPS: A Local Positioning System for Homogeneous and Heterogeneous Robot-Robot Teams, Robot-Human Teams, and Swarms, Jeju, Korea, 24–27 June 2019; pp. 200–207. [CrossRef]
96. Huang, K.; Xian, Y.; Zhen, S.; Sun, H. Robust control design for a planar humanoid robot arm with high strength composite gear and experimental validation. *Mech. Syst. Signal Process.* **2021**, *155*, 107442. [CrossRef]