

Review

A Comprehensive Survey of Recent Trends in Cloud Robotics Architectures and Applications

Olimpiya Saha * and Prithviraj Dasgupta 

Computer Science Department, University of Nebraska Omaha, 6001 Dodge Street, Omaha, NE 68182, USA; pdasgupta@unomaha.edu

* Correspondence: osaha@unomaha.edu

Received: 29 May 2018; Accepted: 17 August 2018; Published: 30 August 2018



Abstract: Cloud robotics has recently emerged as a collaborative technology between cloud computing and service robotics enabled through progress in wireless networking, large scale storage and communication technologies, and the ubiquitous presence of Internet resources over recent years. Cloud computing empowers robots by offering them faster and more powerful computational capabilities through massively parallel computation and higher data storage facilities. It also offers access to open-source, big datasets and software, cooperative learning capabilities through knowledge sharing, and human knowledge through crowdsourcing. The recent progress in cloud robotics has led to active research in this area spanning from the development of cloud robotics architectures to its varied applications in different domains. In this survey paper, we review the recent works in the area of cloud robotics technologies as well as its applications. We draw insights about the current trends in cloud robotics and discuss the challenges and limitations in the current literature, open research questions and future research directions.

Keywords: cloud robotics; collective robot learning; knowledge sharing

1. Introduction

Robotics has seen significant developments over the past decades resulting in its increased applications to several real-world problems including automated manufacturing, extra-terrestrial operations, unmanned search and rescue, disaster robotics, self-driving vehicles, socially assistive robots, and healthcare and medical robots. The robots used in many of these applications are single robots that are limited by their on-board hardware and computational constraints. To address this problem, the field of networked robotics [1] emerged almost two decades ago, which connected a team of robots through a wired or wireless communication network. Networked robots address the problem associated with standalone robot systems by sharing the perceived data with each other and solving a task in a cooperative and/or coordinated manner. However, networked robots also encounter some of the same problems associated with single robot systems.

The main problems [2] associated with networked robots occur due to resource constraints, information and learning constraints, and communication constraints. The hardware constraints of each individual robot limits its computing and storage capacity resulting in a collective limitation of the networked robots. It is technically difficult to change or upgrade the resource configurations once a robot has been designed, built and deployed. Networked robots are also constrained by information, as they have access to only the information accumulated by robots possessing a limited variety of sensors and connected through a network. This can mostly lead to performance gains in static environments, but their performance quickly degrades when the environment drastically changes or when they are placed in new environments. Networked robots maintain machine-to-machine communication using communication protocols like proactive routing or ad hoc routing which

incur high computation and memory requirements and high latency. Complex real-life problems requiring real-time execution demand sophisticated data analysis and computational capabilities that are challenging for networked robots to handle.

To address the discussed challenges, researchers have recently proposed the cloud-enabled robotics technology which utilizes the elastic on-demand resources offered by a ubiquitous cloud infrastructure. Cloud robotics originated as a recent technology integrating the advantages of cloud computing into robotics. Mell et al. define cloud computing as: “a model which enables ubiquitous, convenient, on-demand network access to a shared pool of configurable resources that can be rapidly provisioned and released with minimal management effort or service provider interaction” [3]. The term ‘Cloud Robotics’ was first coined by James J. Kuffner in 2010. It refers to any robot or automation system that utilizes the cloud infrastructure for either data or code for its execution, i.e., a system where all sensing, computation and memory are not integrated into a single standalone system. Integration of cloud computing into robotics results in several advantages which are highlighted below:

- Cloud robotics offers robots increased computational power and storage space [4].
- Cloud-enabled robots are not required to perform complicated tasks on board, but can offload computation-intensive tasks such as object recognition, pattern matching, computer vision, and speech synthesis and recognition to the cloud. These tasks can be solved much faster on the cloud, in real-time, utilizing massively parallel computation or grid computing capabilities [4].
- Along with dynamic offloading of computational tasks, cloud infrastructure in cloud robotics supports elastic and on-demand availability of computational resources [2].
- Many applications in robotics, like simultaneous localization and mapping (SLAM) and other perception systems, rapidly give rise to enormous amounts of sensor data that is difficult to store with the limited onboard storage capacity on most robots. Cloud enabled robots have access to high storage spaces offered by the cloud where they can store all the useful information for utilization in future [2].
- In addition to these advantages, cloud robotics enables robots to access big data such as global maps for localization, object models that the robots might need for manipulation tasks as well as open-source algorithms and code [5].
- Finally, cloud robotics also facilitates cooperative learning between geographically distributed robots by information sharing on how to solve a complicated task. Cloud robotics also allows robots to access human knowledge through crowdsourcing utilizing the cloud. Hence, with the introduction of cloud robotics, robots are no longer self-contained systems limited by their on-board capabilities, but they can utilize all the advantages offered by the cloud infrastructure [5].

In this survey paper, we offer an in-depth review of the current literature and recent trends in cloud robotics. We have mainly reviewed the works published over the time span ranging from 2012–2018 and have included 104 papers in this survey. While there has been a considerable body of literature before this time frame, in this survey, we emphasize the recent works to determine the current trends and limitations in this field. We have retrieved papers available in standard databases of scientific articles like IEEE Xplore, ACM, Elsevier and Springer using different variations and combinations of the keywords ‘cloud robotics’, ‘cloud’, ‘robots’ and ‘automation’. We have also reviewed relevant papers cited in the papers initially retrieved from the databases. In order to offer a holistic and complete overview of the literature, we have included all the works that we have found between 2012–2018 without performing any selective criteria analysis on the retrieved papers. From our study in this area, we found that research in cloud robotics can be broadly classified into two major categories: (a) cloud robotics system architectures, and (b) cloud robotics applications. The first category involves different system architectures to enable cloud robotics including communication protocols and languages between robots and the cloud, database facilities for storage and maintenance of robot data and software architectures of the robots and the cloud. Cloud robotics has also been applied to various applications related to robotics and automation systems. We have further categorized these

applications into six major directions: (i) Perception and Computer Vision applications, (ii) Navigation, (iii) Grasping or Manipulation, (iv) Manufacture or Service robotics and (v) Social, Agriculture, Medical and Disaster robotics (vi) Crowdsourcing applications. Table 1 summarizes our literature classification along with the general contributions and evaluation techniques used.

Table 1. Literature classification of recent works in cloud robotics (CR)

Category	Works	Contribution	Metrics and Evaluation Techniques
Architecture	[6–20]	Development of software architecture for heterogeneous robots to share sensor data between processing nodes for computationally intense algorithms.	Execution and round-trip communication time for SLAM and robotic manipulation tasks.
Applications	[21–26]	(i.) Computer vision tasks like object and face recognition, mapping and video tracking.	Round-trip communication time for computer vision systems and performance analysis for biometric systems like False Rejection Rate (FAR) and False Acceptance Rate (FAR).
	[27–35]	(ii.) Navigation problems in robotics like SLAM, motion and path planning.	Implemented on distributed robots, constructed maps and paths, qualitatively examined for localization accuracy, bandwidth usage, processing speeds and map and path storage.
	[36–46]	(iii.) Manipulation tasks like grasp planning combined with deep learning from publicly available 3D object models like Dex-Net 1.0 and Dex-Net 2.0.	Empirically evaluated for grasp reliability in comparison to naive locally computed strategies.
	[47–52]	(iv.) Automate industrial processes to increase the efficiency of manufacturing	Distributed robots' navigation performance.
	[11,53–62]	(v.) Agriculture, healthcare and disaster management.	Performance gains measured by analyzing standard domain metrics like round-trip time, data loss percentage, mean localization and root mean square error.
	[63–65]	(vi.) Incorporate human knowledge for augmenting vision and speech tasks.	Approximate time for cloud communication and task completion and system feasibility analysis.

It is important to mention that significant literature reviews have been published in the past in the area of cloud robotics [2,5]. However, as discussed by the authors, these surveys have mainly included works until August of 2014. Many new works have been published after that time frame in the area of cloud robotics. Hence, in this survey, we have included all the recent works available in the literature to present an overview of the current state of the art of the field. Although our survey includes few important works from 2012–2013 time frame, we have tried to include the majority of the works in the literature published at a later time frame. Another important point of difference between our work and [5] is the categorization of the literature used to present the survey. Kehoe et al. [5] used cloud robotics aspects like big data, cloud computing, collective robot learning and human computation for the organized representation of their survey. In contrast to their work, in our survey, we have used two broader categorization encompassing cloud robotics architecture and applications and used subcategories within the applications like computer vision, navigation, grasping, manufacturing systems, social, agriculture, medical and disaster robotics and crowdsourcing applications. We believe this detailed categorization will offer a better understanding of the works in specific areas of cloud

robotics applications. We would also like to emphasize that in our survey we have included the works specifically involving the application of cloud to manufacturing/service robotics which were not present in the previously published works. In the next section, we discuss the main works under each of these categories and subcategories along with their main contribution, evaluation technique and results. In the following section, we discuss the open issues, challenges and future directions of research in cloud robotics and finally conclude with our insights and closing remarks.

2. Cloud Computing and Robotics

Cloud computing can be informally defined as a set of services that enable access to distributed and shared computing resources over the Internet, including information processing, information storage and massive datasets. There are several commercial cloud computing service providers at present like Amazon Web Services [66], Elastic Compute Cloud commonly referred as EC2 [67], Google Compute Engine [68] and Microsoft Azure [69]. Transferring robotic applications to the cloud requires special frameworks or service models that support this transition [3]. Cloud computing offers three types of service models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). In the recent past, Robot as a Service (RaaS) has been proposed as a cloud computing unit which facilitates the seamless integration of robot and embedded devices into Web and cloud computing environments [70,71]. Cloud robotics has emerged as an evolutionary concept which combines cloud computing with robotics. Cloud robotics architecture mainly consist of two components—the cloud infrastructure and its bottom facility. The bottom facilities typically include different types of robots ranging from varied unmanned ground vehicles to aerial robots, automated machinery and equipment. The cloud infrastructure consists of several high-performance servers, proxy servers and massive databases that can support high-speed processing and computation along with huge storage capabilities. Figure 1 illustrates the high-level architecture of cloud robotics systems along with some of the applications.

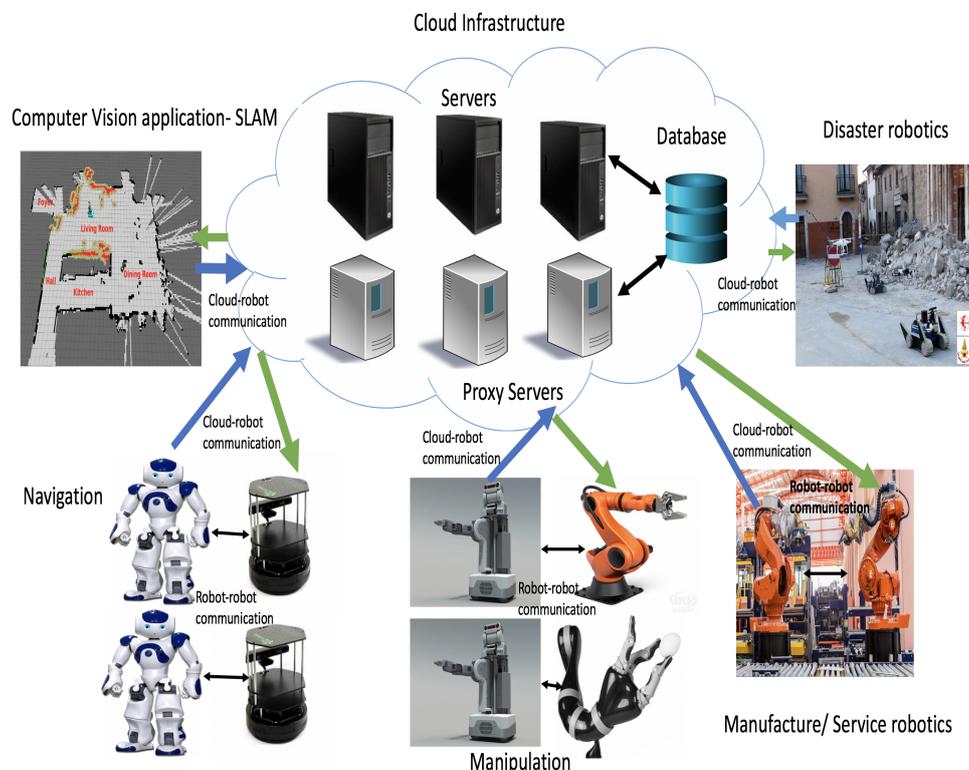


Figure 1. High-level overview of cloud robotics system architecture and applications.

Recent work in cloud robotics can be broadly classified into two categories: Cloud Robotics Architecture and Cloud Robotics Applications. In the following subsections, we discuss the techniques proposed in the literature under each of these categories.

3. Cloud Robotics System Architectures

Among the earlier works in cloud robotics, the works by Arumugam et al., Mohanarajah et al. and Riazuelo et al. [6,16,17] can be considered major works which proposed and implemented the software architecture integrating cloud services with robotics hardware. DAVinCi [6] is a software framework which exploits several advantages of cloud computing for service robots like scalability and parallelism. The DAVinCi software framework is based on the Hadoop cluster combined with the Robot Operating System (ROS) as the messaging framework for robotic ecosystems. The authors implemented the FastSLAM algorithm in Map/Reduce and demonstrated that performance can be significantly improved in terms of execution times to build a map of a large area even while utilizing a small, eight-node Hadoop cluster. The authors empirically evaluated the execution time of their algorithm with change in the number of nodes used in the Hadoop cluster and the number of particles in the sensor data, and demonstrated the corresponding maps created by the robots by traversing the environment.

The RoboEarth project [72] is driven by an open-source cloud robotics platform called Rapyuta [16]. Figure 2 illustrates the system architecture of the RoboEarth system. Rapyuta enables the robots to delegate their intense computational tasks to the cloud through secured customizable computing environments within the cloud. The computing environments in Rapyuta offer access to the RoboEarth knowledge repository that enables robots to download maps, object models and action recipes to perform a specific task. The computing environments in Rapyuta are compactly interconnected, enabling easy deployment of robotic teams. As a working implementation of their framework, the authors discuss three common test cases, some benchmarking and performance results and some working demonstrations of the system in action. The authors provided detailed performance measures of Rapyuta in comparison to rosbridge [73,74] by measuring the round-trip times (RTTs) of different sized messages between two processes.

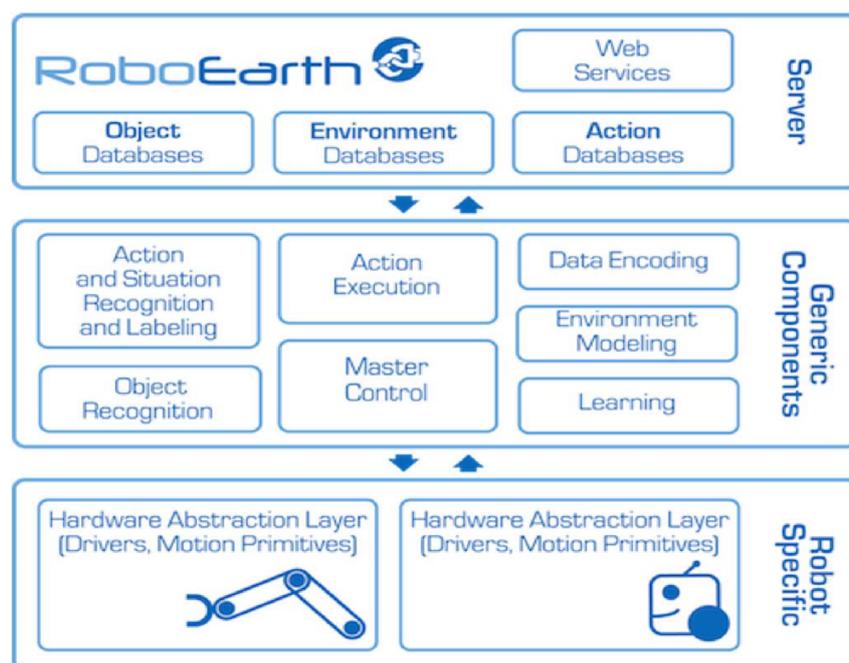


Figure 2. The RoboEarth system architecture [72]. Reused with copyright permissions.

C²TAM [17] is a cloud robotics framework especially developed for cooperative tracking and mapping in the context of simultaneous localization and mapping (SLAM) problems in robotics. C²TAM offers a visual SLAM system based on a distributed framework where the expensive tasks of map optimization and storage are allocated as a service to the cloud, whereas a light camera tracking client is executed on a local computer. Hence, the onboard computers on the robot are freed from the majority of the computations and the only requirement is the availability of a stable Internet connection. The C²TAM system provides access to a map database where a map can be built and stored, stored maps can be reused by other robots, a robot can combine its map on-the-fly with a preexisting map in the database, and several robots can estimate individual maps and combine them together if an overlap is determined. Different experiments were performed to evaluate the cost and bandwidth requirements for tracking, mapping and relocation, in addition to estimating the real-time place recognition and map merging capabilities of their system.

Rosbridge [73,74] was developed as a middleware abstraction layer that offers robotics technology with a standard, minimalist applications development framework that can be accessed by application programmers who are not roboticists themselves. Rosbridge offers a simple, socket-based programmatic access to robot interfaces and algorithms provided by ROS through the use of Web technologies like Javascript. Osentoski et al. [75] proposed an infrastructure to enable Web services for robotics based on a lightweight Javascript binding for ROS, called Rosjs. Rosjs exposes many capabilities of ROS which enables application developers to develop controllers that are executed via a Web browser. Lee utilized rosbridge and the advantages of Web-based solutions for robotic applications in an application development environment called ROSProcessingjs [76] that enables the development of robot applications in a Web browser. Robot Web tools [77] combined rosbridge and Ros.js, an open source library in Javascript which facilitates the communication between the browser and the rosbridge. In a later work, Toris et al. [78] used robot Web tools to advance cloud robotics by developing more efficient ways to transmit high-bandwidth topics. Robot Web tools were compared against standard ROS internal modules in terms of average bandwidth usage for transform subscriptions and transferring point cloud streams, and performance of portable network graphics (PNG) compression for ROS messages was evaluated in terms of data size and compression time. In [79], Sato et al. proposed the Ubiquitous Network Robot Platform (UNR-PF), which realizes the design and development of robotic services that interact with people at different phases of our daily activities in various environments with the help of networked robot systems including robots, sensor networks and smartphones.

The authors in [80] integrated methods from the RoboEarth project with the distributed execution capabilities of UNR-PF to develop a networked robot architecture that can facilitate the development, deployment, management and adaptation of distributed robotic applications. In [81], the authors integrated cloud robotics with software product lines to solve this problem. Software product lines enable end users to deploy and configure complex robotics applications without handling low-level problems like configuring algorithms and designing architectures. As a working example of their approach, the authors provided a case study related to the design of a robotics software product line for cloud-based 3D mapping applications and illustrated how an end user can deploy one of its applications by specifying his or her requirements. The RSi Research cloud (RSi-Cloud) [82] enables the integration of robot services with Internet services. Figure 3 illustrates the system architecture of the Rsi-Cloud. RSi-Cloud features a standardized communication protocol termed as the Robot Service Network Protocol (RSNP). RSi-Cloud can offer various kinds of novel services by combining different service components and by utilizing the robot service specification. A comparative analysis between RSi-Cloud, RoboEarth [72] and Sensor-Cloud [83] indicated that Rsi-Cloud offers a simpler platform for service providers and roboticists to develop robot services and robot applications, respectively.

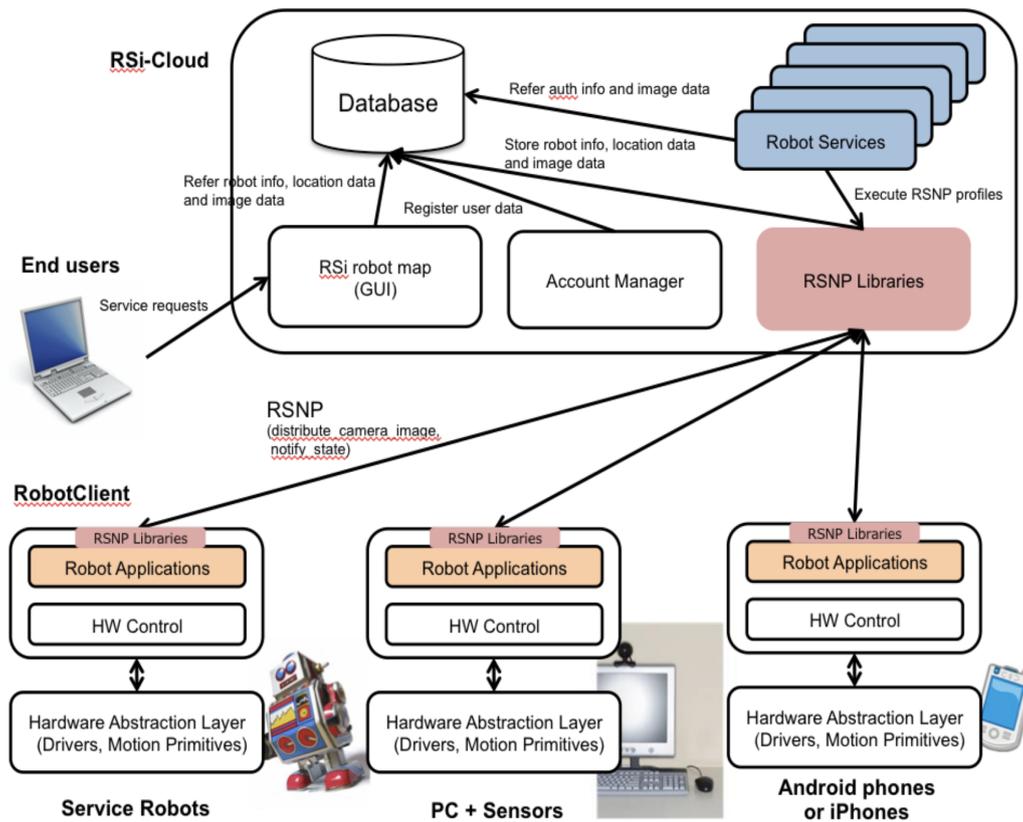


Figure 3. The Rsi-Cloud system architecture [82]. Reused with copyright permissions.

Robot Service Network Protocol (RSNP) has also been utilized in [84,85] to develop a distributed service framework that enables the coordination of various devices, robots and service functions. Narita et al. [86] proposed a method to implement reliable cloud-based robot services for RSNP-based communication in the presence of problems like wireless LAN disconnection, robot service problems and system errors on robots. Another work along this direction proposes a survivable cloud multi-robot (SCMR) framework [87] for heterogeneous environments to alleviate the communication challenges associated with cloud robotics mainly resulting from network disconnections between multi-robot systems and the cloud. The SCMR framework leverages the combined effects of a virtual ad hoc network formed between robots and a physical cloud infrastructure formed by robot-to-cloud communications. Koubaa [88] proposed the RoboWeb system, which extended the paradigm of the service-oriented architecture (SOA) to virtualize robotic hardware and software resources so that they can be universally available through the Web. The proposed RoboWeb system relies on a Simple Object Access Protocol (SOAP)-based Web service middleware that embeds robotic computing resources as services and publishes them to the end-users. Miratabzadeh et al. [89] proposed a cloud robotics software architecture specifically for large-scale autonomous robots, which includes three subsystems in the cloud environment called Middleware, Background Tasks and Control. The proposed architecture uses IaaS as a cloud service and can handle computing, storage and networking.

Berezenyak et al. [7] discuss the prototypes of some of the proposed cloud services and elaborates on Actions language, which is a scripting language for programming robotic behavior. In [9], the authors proposed a Web services-based solution specifically for the Nao robot, which enables it to utilize their software package libraries as an on-demand solution in the form of a service. Many recent works in the cloud robotics literature [8,12,13,19,20] investigate quality-of-service (QoS)-aware and other intelligent techniques for efficient work flow allocation, resource allocation and computational offloading to the cloud for real-time execution. In [18], Nandhini et al. proposed a secured framework

that provides authentication and key agreement using Elliptic Curve Cryptography (ECC) for accessing the robotic services that are hosted in the cloud. In order to incorporate cloud-based services into the daily lives of humans, Du et al. [10] proposed the concept of 'Robot cloud' which was developed by using the popular Google App Engine and by adopting the service-oriented architecture (SOA) to augment the functional modules in the Robot Cloud to become more flexible, extensible and reusable. To address the problem of task allocation in the cloud, the authors proposed a Location-based Packing Scheduling (LPS) algorithm that agglomerates multiple requests and allocates them together to a single robot. A comparative analysis was performed between the LPS algorithm and a naive Exclusive Scheduling (ES) algorithm in terms of number of active robots and total traveling cost changes over time. The impact of deadline distribution on scheduling results was also assessed. In order to alleviate the complexities associated with setup and configuration of majority of the cloud robotics systems, Hajjaj et al. [11] proposed an alternative method called Port Forwarding which offers a private, secured and direct ROS-to-ROS connection, thus eliminating the requirement for a dedicated middleware. Luo et al. [14] designed a cloud robotics middleware based on the Web service technology which parses the Cloud Robotics task request and schedules ROS nodes in a distributed network. In [15], the authors introduced an Open Mobile Cloud Robotics Interface (OMCRI) as a Robot-as-a-Service vision-based platform that offers a unified easy access to remote heterogeneous mobile robots. Pereira et al. [90] proposed the ROSRemote framework, which enables the users to work with ROS remotely while allowing them to create several applications.

4. Cloud Robotics Applications

Cloud robotics applications can be classified into the following subcategories: perception and computer vision applications, navigation, grasping or manipulation, manufacturing or service robotics, social and medical applications, crowdsourcing and human-robot interaction.

4.1. Perception and Computer Vision Applications

There have been some works over the past years that utilize cloud-based solutions for enhancing different aspects of perception and object recognition in robotic systems. Beksi et al. [21] proposed a Cloud-based Object Recognition Engine (CORE) that can train on large-scale datasets, perform classification of 3D point cloud data and perform efficient data transfer in a robotic network. CORE was empirically evaluated for round-trip times during point cloud transmission using User Datagram Protocol (UDP), Transmission Control Protocol (TCP) and WebSocket protocol [91], and UDP was found to provide fastest round-trip times but less reliability compared to TCP and WebSocket. In [22], Lei et al. presented the Cloud Robotics Visual Platform (CRVP) to enhance robot vision. Figure 4 illustrates the framework of CRVP. The main advantages of the CRVP system include the utilization of the parallel computation model of Map/Reduce to reduce the image time cycle of image training and recognition, the usage of a Service-oriented architecture (SOA) at its core, the adoption of the H264 encoding algorithm and the real-time protocol to enhance the video transmission process.

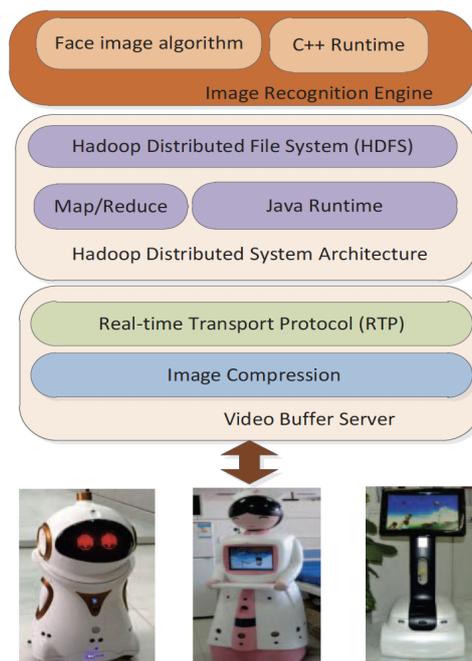


Figure 4. The framework of the Cloud Robot Visual Platform [22]. Reused with copyright permissions.

The authors in [24] present a system that facilitates life-long learning of training data and perception method parameters by using MongoDB, which is a document-oriented, schema-less database technology commonly used in cloud computing applications. The system ensures that a single robot can learn new data with time, and it can also share this dataset with other robots through the available cloud. Tian et al. [25] designed and implemented a cloud-based robot system RC-Cloud that links cloud computing resources for accessing distributed computing resources and big data and executes classic computer vision tasks like face detection and face recognition. The performance efficiency of the system was assessed by evaluating performance metrics for biometric systems like False Rejection Rate (FRR) and False Acceptance Rate (FAR), which respectively measure the likelihood of false rejection rate and false acceptance rate by the cloud-based biometric security system. Multi-sensor data retrieval (MSDR) is a primary task to share resources in robotic systems. In [26], the authors introduced a framework that is explicitly designed to achieve near real-time MSDR by allowing asynchronous access to the cloud from the robots along with a market-based management strategy for efficient data retrieval. Liu et al. [23] proposed a comprehensive framework for information fusion commonly available in cloud robotics that offers desirable characteristics like good scalability and elasticity. The system is validated through an experimental study for a video tracking application where the memory, speed and processors of the system are evaluated.

4.2. Navigation

Many works in the recent past have utilized the cloud for mapping and localization, which are two of the most important tasks for mobile robots. Mapping refers to the robot's process of perceiving the environment with the help of its on-board sensors and creating a map of it, whereas localization refers to the robot's knowledge about its relative position in the perceived environment. The process of simultaneously building a map and localizing a robot, popularly known as SLAM, is a computation-intensive process, and many of the recent works have incorporated cloud technologies for faster and more efficient execution of SLAM.

In [27], the authors developed a simple cloud-based architecture that enables a robot to know its localization information in a dynamic environment. This information could be useful to help navigate the robot in a desired path remotely using a central cloud. Instead of using a single robot, using

the sensory output of multiple robots and combining it efficiently results in globally accurate and consistent metric maps. However, multi-robot map building is a daunting task owing to problems like network latency and low bandwidth. To overcome this problem, Cieslewski et al. [28] proposed a set of novel techniques that lower the bandwidth usage and improve inter-robot communication for distributed mapping by defining a version control system that allows concurrent and consistent access to the map data. Ali et al. [29] proposed an extremely efficient architecture for the Simultaneous Localization and Mapping (SLAM) problem where a recent variant of the SLAM algorithm, FastSLAM 2.0, is parallelized as Map/Reduce tasks via the Hadoop framework. In the proposed approach, the FastSLAM 2.0 algorithm is segmented to execute concurrently the localization and mapping tasks on the cloud to resolve the real-time constraints of the localization task. The FastSLAM 2.0 algorithm was compared against the corresponding sequential algorithm on a single machine with varying number of particles in the Hadoop cluster. In addition to this, latency of the localization process was also compared as a client in the robot and as the cloud service.

Mohanarajah et al. [31] proposed an architecture, protocol and parallel algorithms for collaborative 3D mapping in the cloud with low-cost robots. Figure 5 presents the high-level mapping architecture of the proposed framework. In the proposed collaborative architecture, the robots execute a dense visual odometry algorithm on a smartphone-class processor and keyframes from the visual odometry are subsequently transmitted to the cloud where parallel maps produced by other robots are optimized and merged together. The 3D mapping technique was qualitatively evaluated by analyzing the maps constructed and combined in different environments as well as quantitatively evaluated in terms of network usage, localization accuracy and global map optimization times. In [32], Riazuelo et al. proposed a method that improves a visual SLAM system by utilizing a robot knowledge base. The knowledge base enables the robot to predict which objects can be encountered in a particular environment as well as guide the robot to areas where a novel object can be found given a partial environment map. In a later work [33], Riazuelo et al. proposed the RoboEarth semantic mapping system consisting of an ontology for coding maps and objects, and a SLAM map which provides the scene geometry and object locations relative to the robot. The semantic mapping system boosts the mapping process in RoboEarth by providing a relevant sub-database of object models for a task, sharing of semantic maps between robots and software-as-a-service for mapping in the cloud.

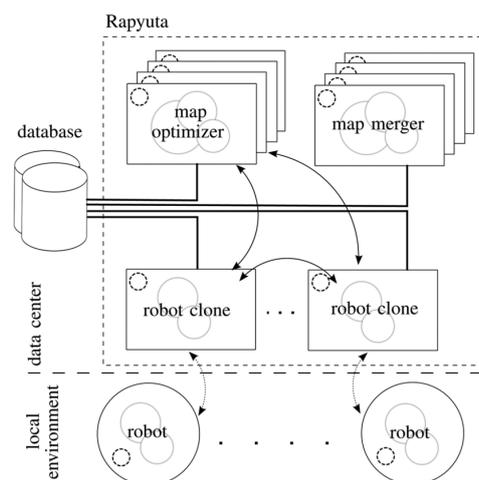


Figure 5. The overview of the proposed mapping architecture proposed in [31]. Reused with copyright permissions.

In the recent past, researchers have explored the advantages offered by the cloud to enhance the path planning and motion planning capabilities in robots. Lam et al. [30] took cloud-based path planning to the next level by offering it as a service in the cloud called Path Planning as a Service (PPaaS). PPaaS is based on a three-layered architectural design consisting of a cloud server layer, cloud engine layer and cloud robot layer. The authors evaluated the feasibility and effectiveness of PPaaS by solving the shortest path problem via parallel processing in the cloud, and path lengths were compared for single and multi-thread processing with time. Salmeron et al. [34] improved motion planning in robots by utilizing computational offloading in the cloud for point cloud extraction from stereo images. In [35], the authors introduced RobotControl as a Service as a new concept for flexible motion planning and control in industrial robots, which is based on an open service-based framework. The services can be executed on the cloud without any form of hardware dependency, which allows a fast reconfiguration of control modules and their repeated usage in diverse tasks.

4.3. Grasping or Manipulation

Manipulation is an important task in robotics, which has wide applications from domestic assistance to robot-assisted surgery. Robotic manipulation poses certain challenges, as it involves grasp planning in a high-dimensional space. Some works in this direction have included the cloud infrastructure to surpass these challenges. In [36], the authors proposed a cloud-based grasp planning framework called *Lightning*, which enables robots to plan grasps for manipulation tasks in high-dimensional spaces. Figure 6 presents the high-level architectural diagram of the *Lightning* framework. This framework consists of two modules running in parallel in the cloud: a planning-from-scratch module and a module that retrieves and modifies paths stored in a library as and when required. Figure 7 illustrates a PR2 robot performing manipulation tasks in kitchen and cupboard environments using the *Lightning* framework. In [38], the authors have proposed a cloud-based grasp planning algorithm that takes as input an approximate object contour along with Gaussian uncertainty around each vertex and center of mass. In addition to this, the authors also present a highly parallelizable algorithm that can evaluate a grasp quality metric using Monte Carlo sampling. In a contemporary work [39], the authors explore how assigning bounds on part tolerances based on an adaptive cloud-based algorithm can help in determining lower bounds on achieving force closure during grasping. Kehoe et al. [40] in a later work illustrated how cloud-based data and computation can improve 3D robot grasping tasks. The proposed architecture for cloud-based robot grasping system included a Willow Garage PR2 robot (Menlo Park, CA, USA) with onboard color and depth cameras, object recognition engine [40], the Point Cloud Library (PCL) [92] for pose estimation, the GraspIt! toolkit [93] and OpenRAVE [94] for 3D grasping. The proposed system was assessed in terms of object recognition, pose estimation and grasping capabilities, and the results indicated a recall rate of 80% for objects in the test set, a failure rate of under 14% and 23%, respectively, for pose estimation and grasping. Another recent work in this direction [41] explores how cloud computing can be utilized to facilitate parallel Monte Carlo sampling of combination actions and shape perturbations to identify a lower bound on the probability of achieving force closure.

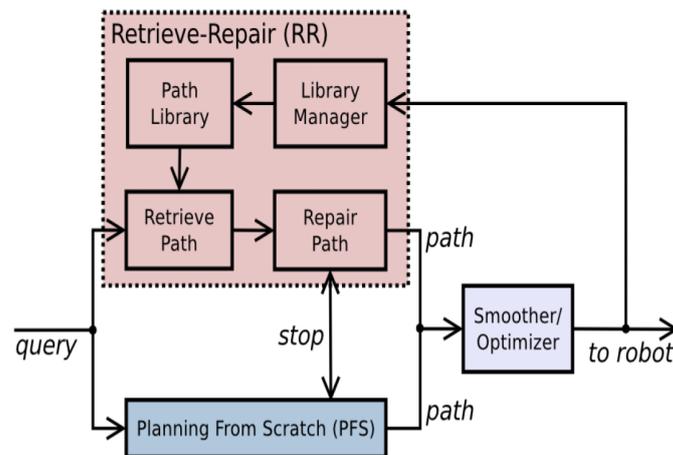


Figure 6. Diagram of the Lightning framework [36]. Reused with copyright permissions.

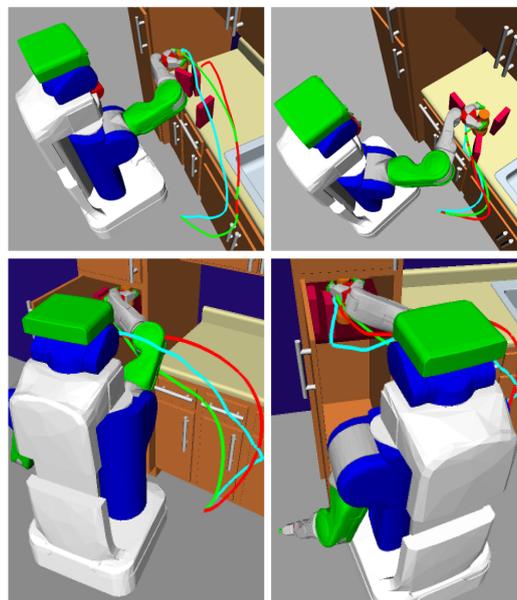


Figure 7. PR2 performing manipulation tasks in different environments using the Lightning framework [36]. Reused with copyright permissions.

Over recent years, cloud resources have been applied in areas like deep learning and deep reinforcement learning to serve the computational requirements associated with these systems and to improve the grasp performance in robots. The Dexterity Network 1.0 [43] consists of a cloud-based dataset of 3D object models that can be used in combination with a sampling-based planning algorithm for robust grasp planning. Dex-Net 1.0 uses Multi-View Convolutional Neural Network (MV-CNN), a new deep learning method for 3D object classification. The authors evaluated the performance of the Dex-Net 1.0 algorithm on robust grasp planning on varying sizes of prior data used from Dex-Net in terms of the force-closure metric along with the sensitivity of the convergence rate to object shape, the similarity kernel bandwidths and uncertainty. In [95], the authors described the Berkeley Robotics and Automation as a Service (Brass) which is a Robotics and Automation as a Service (RAaaS), and enables the robots to access Dex-Net 1.0. Dex-Net 1.0 preserves data on hundreds of candidate grasps on several 3D object meshes and uses perturbation sampling to estimate and update a stochastic robustness metric for each grasp. Later, the authors proposed the Dexterity Network 2.0 [44], which was developed to reduce the data collection time for deep learning of robust robotic grasp plans by

exploring training from a large synthetic dataset consisting of point clouds, grasps and grasp analysis metrics generated on 3D object models available from Dex-Net 1.0. In [45], the authors present a privacy-preserving grasp planning method in the cloud in which a masker version of the part boundary of an object is uploaded, allowing proprietary aspects of the part geometry to remain confidential. This work focuses on the needs of industrial users who might not want to share proprietary details of a product's geometry with external third parties. Levine et al. [42] proposed a learning-based approach to facilitate hand-eye coordination for robotic grasping from monocular images. The efficiency of the learning-based continuous grasping system was quantitatively evaluated in terms of grasping accuracy and success rate, and comparative advantage was measured against alternative approaches by performing experiments with novel objects not seen during training. In [37], the authors illustrated how knowledge can be interchanged between robots by using OpenEASE [96] as the cloud application and providing it with ontologies about the environments, execution logs from different robots and semantic descriptions of both environments. In [46], the authors discussed the collaboration among robots for cooperative adaptive control using the cloud. In this work, the authors introduced the concept of Collective Sufficient Richness, which achieves parameter convergence through teamwork in the group. The recently proposed distributed asynchronous system architecture in deep reinforcement learning [97,98] indicates possibilities for their integration with cloud robotics in future.

4.4. *Manufacture or Service Robotics*

The emergence of cloud robotics has ushered in a new era of Smart Manufacturing Environments (SME) in industry where autonomous robots equipped with cloud resources are deployed in real-world industrial tasks like warehouse management, automated packing, etc. Some works in the recent past have looked in this direction. Cardarelli et al. [47] proposed a centralized cloud service responsible for receiving and merging data acquired by different sensing systems. These data are then distributed to autonomous vehicles, which utilize these for implementing advanced navigation strategies in real industrial environment settings. The system was validated in a real industrial environment to safely perform obstacle avoidance with an autonomously driven forklift. Singhal et al. [49] discuss a detailed implementation of a cloud-based system for managing a fleet of autonomous mobile robots operating in a factory or warehouse environment. The implemented system includes a global planner, which decides the target destination for each robot and along with the ground vehicles comprises a multi agent system (MAS), communicating with each other over a wireless network. In [48], the authors depicted an individual cloud robotics distributed application that was deployed across a high-speed wide-area network. The industrial application was aimed at enabling an industrial robot to perform surface blending. Wan et al. [50] proposed a cognitive industrial entity called Context-Aware Cloud Robotics (CACR), which integrates the benefits of the industrial Internet of Things (IIoT) with cloud robotics. Figure 8 presents the cloud-based architecture for the CACR system in collaboration with the Cognitive Industrial Internet of Things (CIIoT). CACR is characterized by context-aware services and effective load balancing to significantly enhance production processes and intelligent manufacturing in smart factories. In [51], the authors offered a comprehensive survey of the recent works related to the manufacturing systems based on cloud infrastructure. Yan et al. [52] elaborated about the challenges associated with smart manufacturing applications and discusses the countermeasures that can be adopted in the future. In this work, the authors discuss about the internal mechanisms of perception and interaction, intelligent scheduling and control of cloud robot systems oriented to smart manufacturing, and proposes a design of cloud architecture suitable for group learning.

4.5. *Social, Agriculture, Medical and Disaster Robotics*

Owing to its strong computational and storage benefits, cloud robotics has found its applications in different real-life problems. Bhargava et al. [54] proposed a mobile-cloud collaborative approach for context-aware outdoor navigation that utilizes the computational power of resources made accessible by cloud computing providers for real-time image processing. The authors also developed an outdoor

navigation application with integrated support for pedestrian crossing guidance that is believed to be helpful for safe outdoor navigation especially for the blind and visually-impaired in unfamiliar environments. In [99], the authors proposed a cloud robotics framework adapted for optimal task offloading particularly for smart city applications. In this work, task offloading is formulated in the form of a constrained optimization problem that can support a direct acyclic graph (DAG) known as task flow, and a genetic algorithm-based approach is developed to solve the problem. In a later work, Beigi et al. [53] proposed a cloud robotics platform called C2RO, which incorporates real-time stream processing technology to virtually connect the energy-efficient and low-cost mobile devices or sensors. C2RO includes the hybrid cloud robotics computation model as a processing model, which uses both edge and cloud computing technologies in robotics to support smart city applications in real time. The advantage of the C2RO platform was demonstrated experimentally and evaluated in terms of latency and processing rate. In [59], the authors proposed a novel robotic telepresence platform consisting of a semi-autonomous mobile robot based on a cloud robotics framework that can aid mobility impaired people to explore information sites like museums that would otherwise be impossible. The presented system in this paper supports active perception of the user as he can access the robot's vision through its camera in real-time while controlling it to follow a predefined tour or to freely explore the museum. Recently, cloud technology has also been integrated to develop an automated surveillance system [61], which includes a 3D symbolic world model, simple embedded devices capable of computational offloading, mobile devices like "watch" robots, high-bandwidth wireless connections and 3D depth cameras as the sensing technology. In [60], the authors explore the potentialities of using a cloud computing platform to overcome the limitations of hardware existing in simple service robots used in smart homes by integrating the service robot Rovio (WowWee, HongKong, China) and the home automation system Z-wave (Vera, Zensys, Copenhagen, Denmark). Cloud computing improved this integrated system by offering it advanced sensing and interaction capabilities based on image processing and voice recognition.

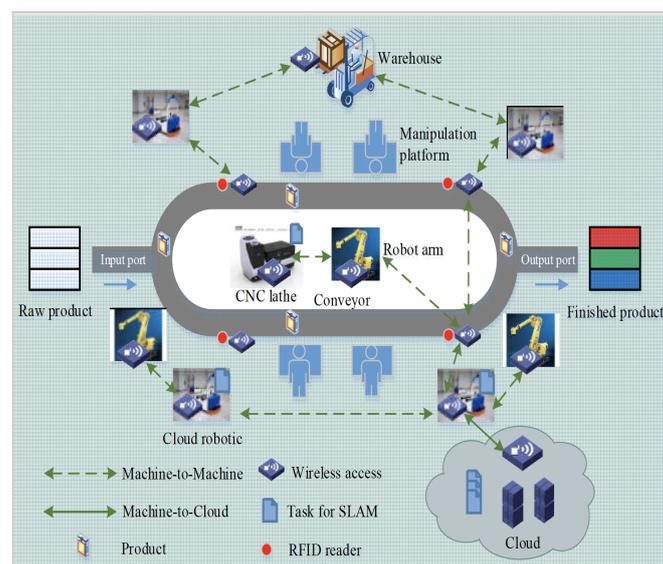


Figure 8. Cloud-based architecture for Context-Aware Cloud Robotics in the Cognitive Industrial Internet of Things [50]. Reused with copyright permissions.

In [62], the authors proposed the construction of a rehabilitation database inspired with medical cloud technologies where physical therapy robots can digitally record and save information to the database and can help physical therapists gather information about patient's health conditions and progress. The rehabilitation database consists of the physical therapy robots and the rehabilitation server connected through a network where the latter saves and maintains a database of movement therapy and can also extract knowledge based on statistical processing. The proposed system was

empirically evaluated by performing experiments with ten healthy male adults using an artificial pneumatic muscle and the subjects moved their arm to follow a target after the robot arm started position and force control. The experimental results indicated that the improvement of tracking property of subjects is superior compared to maneuverability. Bonaccorsi et al. [55] aimed to investigate and examine the usefulness of a cloud robotic system for offering the much-required assistive services essential to facilitate healthy and active aging among the elderly population. The proposed system consisted of two smart environments located in Sweden and Italy where a service robot was connected to a cloud platform for offering localization-based service to the elderly. In order to assess the responsiveness and reliability of the database, the round-trip time and data loss percentage was evaluated and the user localization module in the system was evaluated for mean localization error and root mean square error. In [11], the authors proposed a unique application of a Robot Operating System (ROS) that can be combined with cloud or IoT to solve agricultural problems through automation.

Robotics and automation systems can serve a critical role in disaster management as they can perform unmanned search and rescue operation in areas which are dangerous and difficult for humans to access. The European project SHERPA: Securing a Hybrid Environment for Research Preservation and Access [58] has been dedicated to developing a robotic platform to support search and rescue activities in hostile environments like the Alpine scenario. In a recent extension of this project, Botta et al. [56] envision the potential of incorporating cloud technologies to enhance the smart collaboration between humans and ground-aerial robots involved in SHERPA. In this work, the authors investigated how different network technologies can facilitate fast, reliable and available network connections to utilize maximum advantage of cloud robotics in SHERPA. Jangid et al. [57] explained in detail about the utility and tremendous benefits that can be offered by a Real-Time Cloud (RTC) for efficient disaster management. RTC can help intelligent robots perform complex processing on RTC via a request and response model. The authors discuss some of the salient features that can be used for developing real-time disaster management applications on the cloud-like multi-core CPU architecture with network-on-chip, parallel and distributed algorithms and gigabit networking.

4.6. Crowd-Sourcing and Human–Robot Interaction

Cloud computing corroborates the idea of crowd-sourcing and facilitates human–robot interaction for efficiently solving robotic missions. The Human–Robot Cloud (HRC) [63] extended cloud computing by exploring two important directions. First, it promoted the utilization of distributed sensing and actuator networks, thus extending cloud computing to the physical world. Secondly, in addition to electronic components, HRC also integrates the human physical and cognitive components and hence provides crowd-sourcing for the cloud in the form of brief, non-expert interaction with the cloud. In [65], the authors proposed SIGVerse, which is a cloud computing architecture simulation platform developed for realizing social human–robot interaction. Tan et al. described the design and development of the SIGVerse system based on its function as the central server, as the agent controller for user applications, as the service provider and as the client terminal and web-based development interface. These design factors of SIGVerse collectively handle the distributed development platform and large scale of human robot simulation, which are the two primary components of human–robot interaction. Sugiura et al. [64] proposed the cloud robotics platform, Rospeex, which was designed to enhance human–robot spoken dialogue experiences and can be used by the robotics community without any payment or authentication. In order to validate their proposed system, the authors analyzed the feasibility of the speech logs from the Rospeex system collected over the span of a year. Logs were analyzed to investigate the feasibility of the proposed platform in terms of processing speeds by cloud servers and utterances of words.

Table 2 highlights some of the important works in the main application areas of cloud robotics like perception and computer vision, SLAM and robot motion planning and robot manipulation. Table 2 also presents the experimental settings, empirical analyses and the main results of these works.

Table 2. Selective analysis of major applications of cloud robotics.

Papers	Application Area	Experimental Setup	Results
[21]	Perception & computer vision applications	This paper presents the experimental results of the measured round-trip times when point cloud data is transmitted through CORE using UDP, TCP and Websocket protocols.	Experimental results illustrated that UDP provides the fastest round-trip times although there is no guarantee that the server will receive all point cloud messages in order with no duplication.
[23]	Perception & computer vision	To evaluate the performance of offloading image processing in video tracking tasks to the cloud, a cloud enabled distributed robotics prototype consisting of a remote robot network and a cloud testbed were designed. Performance differences were evaluated between a local machine and three instances in the cloud—small, medium and large.	The medium and large instance showed higher performance than the others owing to the availability of higher number of CPUs.
[28]	Robot localization and map building	To demonstrate and validate proposed Map API system, a visual inertial mapping application was developed. For experimental purposes, the runtime for running the registration operation was evaluated about on three emulated ground robots vs. a centralized instance with serialized map processing.	Experimental results indicate that even using less computational power, the decentralized entities finished registering the data within 10 minutes while the central entity takes 25 minutes to finish all the processing.
[31]	Collaborative 3D mapping	The experimental setup for evaluation consisted of two robots and the cloud based architecture running in data center. Experimental analysis involved qualitative evaluation of building and merging of maps created in different environments in addition to quantitative evaluation of network usage, localization and accuracy and global map optimization times.	The experiments analyzing network usage illustrate that the bandwidth usage is proportional to the velocity of the robot. Empirical evaluation of map optimization revealed that cloud-based optimization significantly reduced the error especially in presence of loop closures.
[34]	Mobile robot motion planning	Intensive performance testing was done using different stereo streams, cloud states and connection technologies. Experiments were performed to evaluate scalability, communication technology and time delay.	A significant speedup was obtained for scalability measures which sustained an average frequency of 4 frame pairs per second.
[40]	Robot manipulation	Two sets of experiments were performed. The first set included a set of six objects and end-to-end testing of object recognition, pose estimation and grasping. The second set of experiments evaluated the confidence measure for object recognition using a set of 100 objects and pose estimation using the first set of objects.	For the first set of experiments, higher recall was achieved for image recognition through multiple rounds of hand-selected training images. Pose estimation indicated a maximum failure rate of 0.13 and grasping experiments illustrated a maximum failure rate of 0.23. The second set of experiments demonstrated a recall rate of 0.90 for image recognition and a recall rate of 0.90 for pose estimation.
[42]	Robotic grasping	The proposed approach was compared to an open-loop method which observes the scene prior to the grasp, a random baseline method and a hand-engineered grasping system. Two experimental protocols were used to evaluate the methods based on grasping with and without replacement.	The success rate of the proposed continuous servoing method exceeded the baseline and prior methods in all cases. For the experiments with no replacement, it removed almost all objects successfully after 30 grasps. In the experiments with replacement, the grasp success rate achieved ranged from 20% at the beginning to 70% at the end.
[44]	Robot grasp planning	Classification performance was evaluated on both real and synthetic data. Physical evaluations were performed on an ABB YuMi robot with custom silicon gripper tips. Performance of grasp planning methods were evaluated on known and novel objects. Generalization ability was also evaluated.	The proposed approach planned grasps 3 times faster compared to the baseline approach for known objects and achieved a success rate of high 93% and a precision rate of 94%. For novel objects, success rate achieved was a high of 80% and precision rate was 100%.

5. Open Issues, Challenges and Future Directions

Cloud robotics has evolved tremendously from its inception in 2011, with ample works in the literature emphasizing its architecture as well as applications. However, there still exist several open challenges and problems in cloud robotics systems, which should be addressed for improving the

performance of cloud-based robotic systems. In this section, we identify some of the open issues and challenges in this area. Figure 9 highlights the main challenges and future directions in cloud robotics.

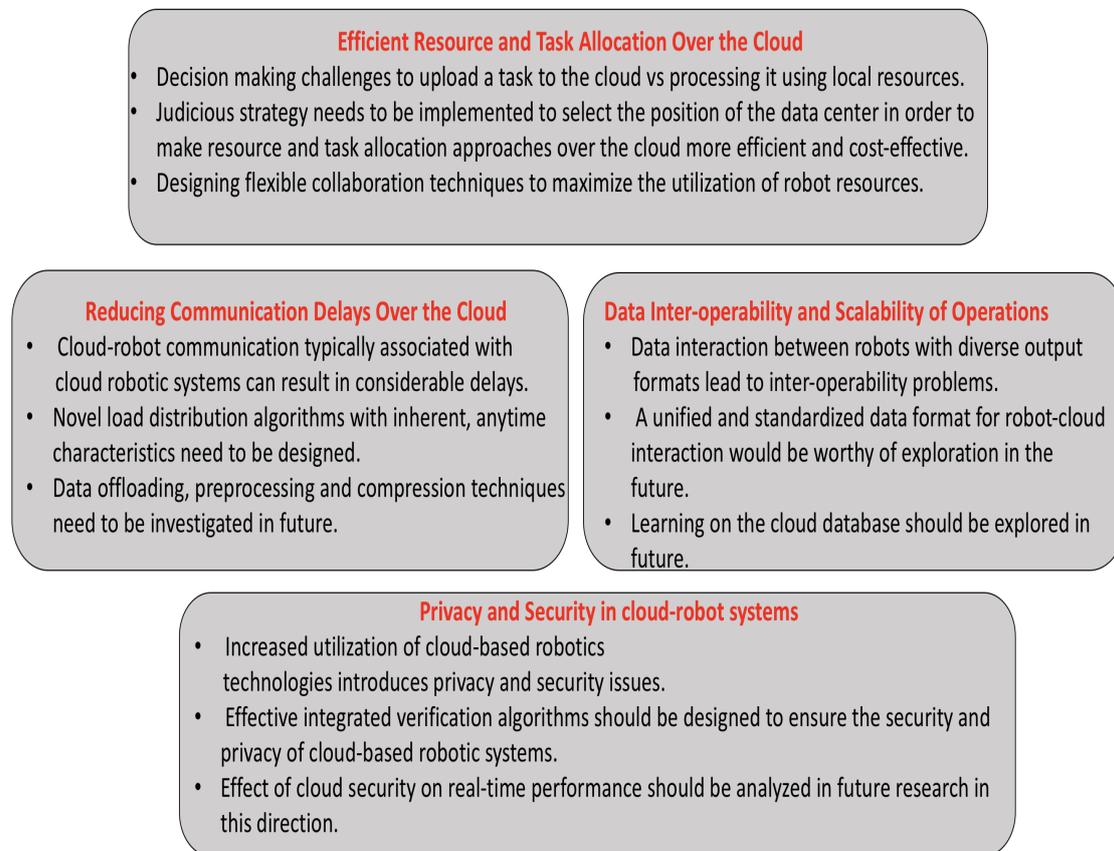


Figure 9. Cloud robotics-challenges and future directions

- **Efficient Resource and Task Allocation Over the Cloud.** One of the crucial determinants of performance of cloud-based robotic systems is making a decision to upload a task to the cloud versus processing it using local resources. Recent works have proposed coordination mechanisms including auctions [100], sequential games [101] and belief propagation techniques [102] for real-time resource allocation and retrieval in cloud-based mobile robotic systems. However, the performance of these techniques is susceptible to the dynamic changes in the network topology as well as delays during the coordination process. A problem worthy of future research is to enable resource allocation methods in the cloud to gracefully handle dynamic changes in the network that might occur during the resource allocation process. Issues related to truthful revelation of preferences through the bids with an auction mechanism and dynamic changes in bids are also relevant in this connection. Within the game-based resource allocation strategy, auto-tuning of variables subject to available information conditions like willingness payment and corresponding response time is another potential future research direction. To address the problem of task allocation in the cloud, authors proposed a communication scheduling technique called Location-based Packing Scheduling (LPS) [10] that agglomerates multiple resource or service requests from different robots and allocates them together to a single robot. As a future direction, a judicious strategy needs to be implemented to select the position of the data center in order to make resource and task allocation approaches over the cloud more efficient and cost-effective. Future work should also involve designing flexible collaboration techniques to maximize the utilization of robot resources to handle challenging tasks.

- **Reducing Communication Delays over the Cloud.** Cloud-robot communication typically associated with cloud robotic systems can result in considerable delays. The massive amounts of data that usually arise in many robotics applications like perception, SLAM and navigation aggravate communication delays. In fact, the central issue in cloud robotics is the computation–communication delay that persists between robots and the cloud for most of the applications. Over recent years, wireless technology has made steady improvements, yet delays are difficult to avoid in the presence of network connection problems between the robot and the cloud services. To overcome this limitation, novel load distribution algorithms with inherent, anytime characteristics need to be designed so that when a computation request from a robot cannot be uploaded properly to the cloud, backup mechanisms are available to dynamically allocate the task and minimize the robot’s delay time in performing operations. Network latency gives rise to another challenge in the real-time deployment of cloud robotics applications. A service quality assurance system and effects analysis dedicated for a specific bandwidth can facilitate uninterrupted network flow by retrieving a balance between limited resource and real-time demands. New algorithms and methods should be investigated to handle time-varying network latency and Quality-of-Service (QoS). In a recent work [103], authors proposed the RoboCloud, which introduces a task specified mission cloud with controllable resources and predictable behavior. However, the RoboCloud was mainly tested for a semantic mapping task based on cloud-assisted object recognition. In the future, it would be worthwhile to test the proposed approach in other types of cloud robotics scenarios like navigation, manipulation, etc. Another work along this direction [87] developed a survivable cloud multi-robot framework for heterogeneous environments, which leverages the advantages of a virtual ad hoc network formed by robot-to-robot communications and a physical cloud infrastructure consisting of robot-to-cloud communication. Future directions of this research could involve the investigation of different offloading schemes to reduce the robot’s energy consumption along with pre-processing and compression of images prior to offloading the data to the cloud server.
- **Data Inter-Operability and Scalability of Operations between Robots in the Cloud.** The data interaction between robots and a cloud platform gives rise to another challenge in cloud robotics. Different robotics applications output data using diverse data formats. As cloud-based services can preserve and operate with data only with specific structures, the data uploaded by the robots to the cloud needs to be properly preprocessed before uploading. Similarly, the data output by the cloud-based services has to be transformed by the robot in robot-specific format for its utilization by the robotics application. This data format conversion to match cloud and robot requirements creates considerable overhead, especially for huge amounts of sensor data involved in robotic tasks. Hence, a unified and standardized data format for robot–cloud interaction would be worthy of exploration in the future. Refs. [104,105] described a formal language for knowledge representation between robots connected via a cloud under the RoboEarth and KnowRob projects. A limitation of the proposed system arises owing to scalability concerns related to management of the quality of stored information, as the RoboEarth system includes many users. In order to make the system scalable, new techniques similar to human crowdsourcing or feedback sharing among the robots need to be investigated to autonomously manage the quality of stored information. The robots can be provided with the ratings of the downloaded information, and they can upload the details of their experience while performing the task and update the rating of the downloaded information. Another interesting future direction that will arise with substantial increase of data will involve learning on the database, which will enable robots to learn typical object locations, success models of plans based on the context, common execution failures, timing information or promising plans for a

given robot platform.

- **Privacy and Security in Cloud–Robot Systems.** The increased utilization of cloud-based robotics technologies introduces privacy and security issues. Cloud-based services involve robotic data to be stored and processes to be performed remotely in the cloud, making these applications vulnerable to hackers and malicious users. Remote storage of data in the cloud can lead to inappropriate access, manipulation and deletion of valuable data by hackers. Remote execution of robotic services in the cloud makes it easier for hackers to access and modify these services, thus changing the behavior of robot tasks in malicious ways. In this regard, researchers have used the term *cryptorobotics* as a unification of cyber-security and robotics [106]. To resolve the security and privacy threats associated with cloud robotics, proper authentication techniques with a layered encryption mechanism are currently used for an entity to access cloud data and services [18]. In the future, more effective integrated verification algorithms should be designed to ensure the security and privacy of cloud-based robotic systems. An important consideration for developing secured cloud robotic systems is whether the integration of security measures like encrypted communication and data validation will affect the real-time performance of these systems. The architectural design in these systems should consider the hardware, the software, the network and the specific application under consideration to achieve a trade-off decision.

6. Conclusions

In this paper, we have provided a comprehensive review of the recent works in the area of cloud robotics. We classified the literature into two broad categories. The first category involves the implementation and development of system-level architectures facilitating the collaboration of cloud computing with advanced robotics. These architectures can be applied to any generalized robotics problem that requires on-demand, real-time execution. On the other hand, the second category comprises works that were proposed to be suitable for specific robotics applications, and the domain knowledge of the application was thoroughly considered while designing these systems. The literature in the latter category was further organized into sub-categories of different robotics applications for ease of understanding. We have also discussed the existing gaps in the literature and future research directions that can lead to higher utilization and inclusion of cloud-based robotic systems in real-life problems. We believe that our survey will offer a detailed knowledge about the state-of-the-art of recent research in this field and help researchers to pursue research to address open issues and existing challenges in this direction.

Author Contributions: Conceptualization, O.S. and P.D.; Methodology, O.S. and P.D.; Investigation, O.S.; Writing—Original Draft Preparation, O.S.; Writing—Review & Editing, P.D.; Visualization, O.S.; Supervision, P.D.

Funding: This research received no external funding.

Acknowledgments: We thank Anna Buhman for her comments which substantially improved the quality of this paper.

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

References

1. McKee, G.T.; Schenker, P.S. Networked robotics. Sensor Fusion and Decentralized Control in Robotic Systems III. *Int. Soc. Opt. Photonics* **2000**, *4196*, 197–210.
2. Hu, G.; Tay, W.P.; Wen, Y. Cloud robotics: Architecture, challenges and applications. *IEEE Netw.* **2012**, *26*, 21–28. [[CrossRef](#)]
3. Mell, P.; Grance, T. *The Nist Definition of Cloud Computing*; Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology: Gaithersburg, MD, USA, 2011; pp. 1–3.

4. Wan, J.; Tang, S.; Yan, H.; Li, D.; Wang, S.; Vasilakos, A.V. Cloud robotics: Current status and open issues. *IEEE Access* **2016**, *4*, 2797–2807. [[CrossRef](#)]
5. Kehoe, B.; Patil, S.; Abbeel, P.; Goldberg, K. A survey of research on cloud robotics and automation. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 398–409. [[CrossRef](#)]
6. Arumugam, R.; Enti, V.R.; Liu, B.; Wu, X.; Baskaran, K.; Kong, F.F.; Kumar, A.S.; Meng, K.D.; Kit, G.W. DAVinCi: A cloud computing framework for service robots. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA), Anchorage, AK, USA, 3–7 May 2010; pp. 3084–3089.
7. Bereznyak, I.; Chepin, E.; Dyumin, A. The actions language as a programming framework for cloud robotics applications. In Proceedings of the 2016 6th International Conference on Cloud System and Big Data Engineering (Confluence), Noida, India, 14–15 January 2016; pp. 119–124.
8. Chen, W.; Yaguchi, Y.; Naruse, K.; Watanobe, Y.; Nakamura, K. QoS-aware Robotic Streaming Workflow Allocation in Cloud Robotics Systems. *IEEE Trans. Serv. Comput.* **2018**. [[CrossRef](#)]
9. Bouziane, R.; Terrissa, L.S.; Ayad, S.; Brethe, J.F.; Kazar, O. A web services based solution for the NAO robot in cloud robotics environment. In Proceedings of the 2017 4th International Conference on Control, Decision and Information Technologies (CoDIT), Barcelona, Spain, 5–7 April 2017; pp. 0809–0814.
10. Du, Z.; He, L.; Chen, Y.; Xiao, Y.; Gao, P.; Wang, T. Robot Cloud: Bridging the power of robotics and cloud computing. *Future Gener. Comput. Syst.* **2017**, *74*, 337–348. [[CrossRef](#)]
11. Hajjaj, S.S.H.; Sahari, K.S.M. Establishing remote networks for ROS applications via Port Forwarding: A detailed tutorial. *Int. J. Adv. Robot. Syst.* **2017**, *14*. [[CrossRef](#)]
12. Hu, B.; Wang, H.; Zhang, P.; Ding, B.; Che, H. Cloudroid: A Cloud Framework for Transparent and QoS-aware Robotic Computation Outsourcing. In Proceedings of the 2017 IEEE 10th International Conference on Cloud Computing (CLOUD), Honolulu, CA, USA, 25–30 June 2017; pp. 114–121.
13. Liu, H.; Liu, S.; Zheng, K. A Reinforcement Learning-Based Resource Allocation Scheme for Cloud Robotics. *IEEE Access* **2018**, *6*, 17215–17222. [[CrossRef](#)]
14. Luo, J.; Zhang, L.; Zhang, H.Y. Design of a cloud robotics middleware based on web service technology. In Proceedings of the 2017 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 10–12 July 2017; pp. 487–492.
15. Merle, P.; Gourdin, C.; Mitton, N. Mobile cloud robotics as a service with OCCiware. In Proceedings of the 2017 IEEE International Congress on Internet of Things (ICIOT), Honolulu, CA, USA, 25–30 June 2017; pp. 50–57.
16. Mohanarajah, G.; Hunziker, D.; D’Andrea, R.; Waibel, M. Rapyuta: A cloud robotics platform. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 481–493. [[CrossRef](#)]
17. Riazuelo, L.; Civera, J.; Montiel, J. C2tam: A cloud framework for cooperative tracking and mapping. *Robot. Auton. Syst.* **2014**, *62*, 401–413. [[CrossRef](#)]
18. Nandhini, C.; Doriya, R. Towards secured cloud-based robotic services. In Proceedings of the 2017 International Conference on IEEE Signal Processing and Communication (ICSPC), Coimbatore, India, 28–19 July 2017; pp. 165–170.
19. Rahman, A.; Jin, J.; Cricenti, A.; Rahman, A.; Panda, M. Motion and Connectivity Aware Offloading in Cloud Robotics via Genetic Algorithm. In Proceedings of the IEEE GLOBECOM 2017-2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
20. Wang, L.; Liu, M.; Meng, M.Q.H. A hierarchical auction-based mechanism for real-time resource allocation in cloud robotic systems. *IEEE Trans. Cybern.* **2017**, *47*, 473–484. [[CrossRef](#)] [[PubMed](#)]
21. Beksi, W.J.; Spruth, J.; Papanikolopoulos, N. Core: A cloud-based object recognition engine for robotics. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4512–4517.
22. Yin, L.; Zhou, F.; Wang, Y.; Yuan, X.; Zhao, Y.; Chen, Z. Design of a Cloud Robotics Visual Platform. In Proceedings of the 2016 Sixth International Conference on IEEE Instrumentation & Measurement, Computer, Communication and Control (IMCCC), Harbin, China, 21–23 July 2016; pp. 1039–1043.
23. Liu, B.; Chen, Y.; Blasch, E.; Pham, K.; Shen, D.; Chen, G. A holistic cloud-enabled robotics system for real-time video tracking application. In *Future Information Technology*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 455–468.

24. Niemueller, T.; Schiffer, S.; Lakemeyer, G.; Rezapour-Lakani, S. Life-long learning perception using cloud database technology. In Proceedings of the IROS Workshop on Cloud Robotics, Tokyo, Japan, 3–7 November 2013.
25. Tian, S.; Lee, S.G. An implementation of cloud robotic platform for real time face recognition. In Proceedings of the 2015 IEEE International Conference on IEEE Information and Automation, Lijiang, China, 8–10 August 2015; pp. 1509–1514.
26. Wang, L.; Liu, M.; Meng, M.Q.H. Real-time multisensor data retrieval for cloud robotic systems. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 507–518. [[CrossRef](#)]
27. Ansari, F.Q.; Pal, J.K.; Shukla, J.; Nandi, G.C.; Chakraborty, P. A cloud based robot localization technique. In *International Conference on Contemporary Computing*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 347–357.
28. Cieslewski, T.; Lynen, S.; Dymczyk, M.; Magnenat, S.; Siegwart, R. Map api-scalable decentralized map building for robots. In Proceedings of the 2015 IEEE International Conference on IEEE Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 6241–6247.
29. Hammad, A.; Ali, S.S.; Eldien, A.S.T. A novel implementation for FastSLAM 2.0 algorithm based on cloud robotics. In Proceedings of the 2017 13th International IEEE Computer Engineering Conference (ICENCO), Cairo, Egypt, 27–28 December 2017; pp. 184–189.
30. Lam, M.L.; Lam, K.Y. Path planning as a service PPaaS: Cloud-based robotic path planning. In Proceedings of the 2014 IEEE International Conference on IEEE Robotics and Biomimetics (ROBIO), Bali, Indonesia, 5–10 December 2014; pp. 1839–1844.
31. Mohanarajah, G.; Usenko, V.; Singh, M.; D’Andrea, R.; Waibel, M. Cloud-based collaborative 3D mapping in real-time with low-cost robots. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 423–431. [[CrossRef](#)]
32. Riazuelo, L.; Tenorth, M.; Marco, D.; Salas, M.; Mosenlechner, L.; Kunze, L.; Beetz, M.; Tardos, J.; Montano, L.; Montiel, J. Roboearth web-enabled and knowledge-based active perception. In Proceedings of the IROS Workshop on AI-Based Robotics, Tokyo, Japan, 3–7 November 2013.
33. Riazuelo, L.; Tenorth, M.; Di Marco, D.; Salas, M.; Gálvez-López, D.; Mosenlechner, L.; Kunze, L.; Beetz, M.; Tardós, J.D.; Montano, L.; et al. RoboEarth semantic mapping: A cloud enabled knowledge-based approach. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 432–443. [[CrossRef](#)]
34. Salmeron-Garcia, J.; Inigo-Blasco, P.; Diaz-del Rio, F.; Cagigas-Muniz, D. Mobile robot motion planning based on Cloud Computing stereo vision processing. In Proceedings of the 41st International Symposium on Robotics, ISR/Robotik 2014, Munich, Germany, 2–3 June 2014; pp. 1–6.
35. Vick, A.; Vonásek, V.; Pěnička, R.; Krüger, J. Robot control as a service—Towards cloud-based motion planning and control for industrial robots. In Proceedings of the 2015 10th International Workshop on IEEE Robot Motion and Control (RoMoCo), Poznan, Poland, 6–8 July 2015; pp. 33–39.
36. Berenson, D.; Abbeel, P.; Goldberg, K. A robot path planning framework that learns from experience. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN, USA, 14–18 May 2012; pp. 3671–3678.
37. Bozcuoğlu, A.K.; Kazhoyan, G.; Furuta, Y.; Stelter, S.; Beetz, M.; Okada, K.; Inaba, M. The Exchange of Knowledge Using Cloud Robotics. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1072–1079. [[CrossRef](#)]
38. Kehoe, B.; Berenson, D.; Goldberg, K. Toward cloud-based grasping with uncertainty in shape: Estimating lower bounds on achieving force closure with zero-slip push grasps. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN, USA, 14–18 May 2012; pp. 576–583.
39. Kehoe, B.; Berenson, D.; Goldberg, K. Estimating part tolerance bounds based on adaptive cloud-based grasp planning with slip. In Proceedings of the 2012 IEEE International Conference on Automation Science and Engineering (CASE), Seoul, Korea, 20–24 August 2012; pp. 1106–1113.
40. Kehoe, B.; Matsukawa, A.; Candido, S.; Kuffner, J.; Goldberg, K. Cloud-based robot grasping with the google object recognition engine. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany, 6–10 May 2013; pp. 4263–4270.
41. Kehoe, B.; Warrier, D.; Patil, S.; Goldberg, K. Cloud-based grasp analysis and planning for toleranced parts using parallelized Monte Carlo sampling. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 455–470. [[CrossRef](#)]

42. Levine, S.; Pastor, P.; Krizhevsky, A.; Quillen, D. Learning hand-eye coordination for robotic grasping with large-scale data collection. In *International Symposium on Experimental Robotics*; Springer: Cham, Switzerland, 2016; pp. 173–184.
43. Mahler, J.; Pokorný, F.T.; Hou, B.; Roderick, M.; Laskey, M.; Aubry, M.; Kohlhoff, K.; Kröger, T.; Kuffner, J.; Goldberg, K. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1957–1964.
44. Mahler, J.; Liang, J.; Niyaz, S.; Laskey, M.; Doan, R.; Liu, X.; Ojea, J.A.; Goldberg, K. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv* **2017**, arXiv:1703.09312.
45. Mahler, J.; Hou, B.; Niyaz, S.; Pokorný, F.T.; Chandra, R.; Goldberg, K. Privacy-preserving grasp planning in the cloud. In Proceedings of the 2016 IEEE International Conference on Automation Science and Engineering (CASE), Fort Worth, TX, USA, 21–25 August 2016; pp. 468–475.
46. Wensing, P.M.; Slotine, J.J.E. Cooperative Adaptive Control for Cloud-Based Robotics. *arXiv* **2017**, arXiv:1709.07112.
47. Cardarelli, E.; Sabattini, L.; Secchi, C.; Fantuzzi, C. Cloud robotics paradigm for enhanced navigation of autonomous vehicles in real world industrial applications. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4518–4523.
48. Rahimi, R.; Shao, C.; Veeraraghavan, M.; Fumagalli, A.; Nicho, J.; Meyer, J.; Edwards, S.; Flannigan, C.; Evans, P. An industrial robotics application with cloud computing and high-speed networking. In Proceedings of the IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 10–12 April 2017; pp. 44–51.
49. Singhal, A.; Pallav, P.; Kejriwal, N.; Choudhury, S.; Kumar, S.; Sinha, R. Managing a fleet of autonomous mobile robots (AMR) using cloud robotics platform. In Proceedings of the 2017 European Conference on Mobile Robots (ECMR), Paris, France, 6–8 September 2017; pp. 1–6.
50. Wan, J.; Tang, S.; Hua, Q.; Li, D.; Liu, C.; Lloret, J. Context-aware cloud robotics for material handling in cognitive industrial internet of things. *IEEE Internet Things J.* **2017**, *5*. [[CrossRef](#)]
51. Wang, X.V.; Wang, L.; Mohammed, A.; Givehchi, M. Ubiquitous manufacturing system based on cloud: A robotics application. *Robot. Comput. Integr. Manuf.* **2017**, *45*, 116–125. [[CrossRef](#)]
52. Yan, H.; Hua, Q.; Wang, Y.; Wei, W.; Imran, M. Cloud robotics in Smart Manufacturing Environments: Challenges and countermeasures. *Comput. Electr. Eng.* **2017**, *63*, 56–65. [[CrossRef](#)]
53. Beigi, N.K.; Partov, B.; Farokhi, S. Real-time cloud robotics in practical smart city applications. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017; pp. 1–5.
54. Bhargava, B.; Angin, P.; Duan, L. A mobile-cloud pedestrian crossing guide for the blind. In Proceedings of the International Conference on Advances in Computing & Communication, Bali Island, Indonesia, 1–3 April 2011.
55. Bonaccorsi, M.; Fiorini, L.; Cavallo, F.; Saffiotti, A.; Dario, P. A cloud robotics solution to improve social assistive robots for active and healthy aging. *Int. J. Soc. Robot.* **2016**, *8*, 393–408. [[CrossRef](#)]
56. Botta, A.; Cacace, J.; Lippiello, V.; Siciliano, B.; Ventre, G. Networking for Cloud Robotics: A case study based on the Sherpa Project. In Proceedings of the 2017 International Conference on Cloud and Robotics (ICCR), Saint Quentin, France, 22–23 November 2017.
57. Jangid, N.; Sharma, B. Cloud Computing and Robotics for Disaster Management. In Proceedings of the 2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), Bangkok, Thailand, 25–27 January 2016; pp. 20–24.
58. Marconi, L.; Melchiorri, C.; Beetz, M.; Pangercic, D.; Siegwart, R.; Leutenegger, S.; Carloni, R.; Stramigioli, S.; Bruyninckx, H.; Doherty, P.; et al. The SHERPA project: Smart collaboration between humans and ground-aerial robots for improving rescuing activities in alpine environments. In Proceedings of the 2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), College Station, TX, USA, 5–8 November 2012; pp. 1–4.

59. Ng, M.K.; Primatesta, S.; Giuliano, L.; Lupetti, M.L.; Russo, L.O.; Farulla, G.A.; Indaco, M.; Rosa, S.; Germak, C.; Bona, B. A cloud robotics system for telepresence enabling mobility impaired people to enjoy the whole museum experience. In Proceedings of the 2015 10th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS), Naples, Italy, 21–23 April 2015; pp. 1–6.
60. Ramírez De La Pinta, J.; Maestre Torreblanca, J.M.; Jurado, I.; Reyes De Cozar, S. Off the shelf cloud robotics for the smart home: Empowering a wireless robot through cloud computing. *Sensors* **2017**, *17*, 525. [[CrossRef](#)] [[PubMed](#)]
61. Salmerón-García, J.J.; van den Dries, S.; Díaz-del Río, F.; Morgado-Estevez, A.; Sevillano-Ramos, J.L.; van de Molengraft, M. Towards a cloud-based automated surveillance system using wireless technologies. *Multimedia Syst.* **2017**, 1–15, 0.1007/s00530-017-0558-5. [[CrossRef](#)]
62. Yokoo, T.; Yamada, M.; Sakaino, S.; Abe, S.; Tsuji, T. Development of a physical therapy robot for rehabilitation databases. In Proceedings of the 2012 12th IEEE International Workshop on Advanced Motion Control (AMC), Sarajevo, Bosnia and Herzegovina, 25–27 March 2012; pp. 1–6.
63. Mavridis, N.; Bourlai, T.; Ognibene, D. The human–robot cloud: Situated collective intelligence on demand. In Proceedings of the 2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Bangkok, Thailand, 27–31 May 2012; pp. 360–365.
64. Sugiura, K.; Zettsu, K. Rospeex: A cloud robotics platform for human–robot spoken dialogues. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 6155–6160.
65. Tan, J.T.C.; Inamura, T. Sigverse—A cloud computing architecture simulation platform for social human–robot interaction. In Proceedings of the 2012 IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN, USA, 14–18 May 2012; pp. 1310–1315.
66. Cloud, A.E.C. Amazon Web Services. Available online: <https://aws.amazon.com/> (accessed on 29 August 2018).
67. Cloud, C. Amazon Elastic Compute Cloud. Available online: <https://aws.amazon.com/> (accessed on 29 August 2018).
68. Krishnan, S.; Gonzalez, J.L.U. Google compute engine. In *Building Your Next Big Thing with Google Cloud Platform*; Springer: Berlin, Germany, 2015; pp. 53–81.
69. Copeland, M.; Soh, J.; Puca, A.; Manning, M.; Gollob, D. Microsoft azure and cloud computing. In *Microsoft Azure*; Springer: Berlin, Germany, 2015; pp. 3–26.
70. Chen, Y.; Du, Z.; García-Acosta, M. Robot as a service in cloud computing. In Proceedings of the 2010 Fifth IEEE International Symposium on Service Oriented System Engineering (SOSE), Nanjing, China, 4–5 June 2010; pp. 151–158.
71. Chen, Y.; Hu, H. Internet of intelligent things and robot as a service. *Simul. Model. Pract. Theory* **2013**, *34*, 159–171. [[CrossRef](#)]
72. Waibel, M.; Beetz, M.; Civera, J.; d’Andrea, R.; Elfving, J.; Galvez-Lopez, D.; Häussermann, K.; Janssen, R.; Montiel, J.; Perzylo, A.; et al. Roboearth. *IEEE Robot. Autom. Mag.* **2011**, *18*, 69–82. [[CrossRef](#)]
73. Crick, C.; Jay, G.; Osentoski, S.; Jenkins, O.C. ROS and rosbridge: Roboticians out of the loop. In Proceedings of the Seventh Annual ACM/IEEE International Conference on Human–Robot Interaction, Boston, MA, USA, 5–8 March 2012; ACM: New York, NY, USA, 2012; pp. 493–494.
74. Crick, C.; Jay, G.; Osentoski, S.; Pitzer, B.; Jenkins, O.C. Rosbridge: Ros for non-ros users. In *Robotics Research*; Springer: Cham, Switzerland, 2017; pp. 493–504.
75. Osentoski, S.; Jay, G.; Crick, C.; Pitzer, B.; DuHadway, C.; Jenkins, O.C. Robots as web services: Reproducible experimentation and application development using rosjs. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 6078–6083.
76. Lee, J. *Web Applications for Robots Using Rosbridge*; Brown University: Providence, RI, USA, 2012.
77. Alexander, B.; Hsiao, K.; Jenkins, C.; Suay, B.; Toris, R. Robot web tools [ros topics]. *IEEE Robot. Autom. Mag.* **2012**, *19*, 20–23. [[CrossRef](#)]
78. Toris, R.; Kammerl, J.; Lu, D.V.; Lee, J.; Jenkins, O.C.; Osentoski, S.; Wills, M.; Chernova, S. Robot web tools: Efficient messaging for cloud robotics. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4530–4537.
79. Sato, M.; Kamei, K.; Nishio, S.; Hagita, N. The ubiquitous network robot platform: Common platform for continuous daily robotic services. In Proceedings of the 2011 IEEE/SICE International Symposium on System Integration (SII), Kyoto, Japan, 20–22 December 2011; pp. 318–323.

80. Tenorth, M.; Kamei, K.; Satake, S.; Miyashita, T.; Hagita, N. Building knowledge-enabled cloud robotics applications using the ubiquitous network robot platform. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Tokyo, Japan, 3–7 November 2013; pp. 5716–5721.
81. Gherardi, L.; Hunziker, D.; Mohanarajah, G. A software product line approach for configuring cloud robotics applications. In Proceedings of the 2014 IEEE 7th International Conference on Cloud Computing (CLOUD), Anchorage, AK, USA, 27 June–2 July 2014; pp. 745–752.
82. Kato, Y.; Izui, T.; Tsuchiya, Y.; Narita, M.; Ueki, M.; Murakawa, Y.; Okabayashi, K. Rsi-cloud for integrating robot services with internet services. In Proceedings of the IEEE IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society, Melbourne, Australia, 7–10 November 2011; pp. 2158–2163.
83. Yuriyama, M.; Kushida, T. Sensor-cloud infrastructure-physical sensor management with virtualized sensors on cloud computing. In Proceedings of the 2010 13th International Conference on Network-Based Information Systems (NBIS), Takayama, Japan, 14–16 September 2010; pp. 1–8.
84. Nakagawa, S.; Ohyama, N.; Sakaguchi, K.; Nakayama, H.; Igarashi, N.; Tsunoda, R.; Shimizu, S.; Narita, M.; Kato, Y. A distributed service framework for integrating robots with internet services. In Proceedings of the 2012 IEEE 26th International Conference on Advanced Information Networking and Applications (AINA), Fukuoka, Japan, 26–29 March 2012; pp. 31–37.
85. Nakagawa, S.; Igarashi, N.; Tsuchiya, Y.; Narita, M.; Kato, Y. An implementation of a distributed service framework for cloud-based robot services. In Proceedings of the IEEE IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society, Montreal, QC, Canada, 25–28 October 2012; pp. 4148–4153.
86. Narita, M.; Okabe, S.; Kato, Y.; Murakawa, Y.; Okabayashi, K.; Kanda, S. Reliable cloud-based robot services. In Proceedings of the IECON 2013-39th Annual Conference of the IEEE Industrial Electronics Society, Vienna, Austria, 10–13 November 2013; pp. 8317–8322.
87. Ramharuk, V. Survivable Cloud Multi-Robotics Framework for Heterogeneous Environments. Ph.D. Thesis, University of South Africa, Pretoria, South Africa, 2015.
88. Koubâa, A. A service-oriented architecture for virtualizing robots in robot-as-a-service clouds. In *International Conference on Architecture of Computing Systems*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 196–208.
89. Miratabzadeh, S.A.; Gallardo, N.; Gamez, N.; Haradi, K.; Puthussery, A.R.; Rad, P.; Jamshidi, M. Cloud robotics: A software architecture: For heterogeneous large-scale autonomous robots. In Proceedings of the World Automation Congress (WAC), Rio Grande, Puerto Rico, 31 July–4 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–6.
90. Pereira, A.B.; Bastos, G.S. ROSRemote, using ROS on cloud to access robots remotely. In Proceedings of the 2017 IEEE 18th International Conference on Advanced Robotics (ICAR), Hong Kong, China, 10–12 July 2017; pp. 284–289.
91. Fette, I. The WebSocket Protocol. 2011. Available online: tools.ietf.org/html/rfc6455 (accessed on 25 August 2018).
92. Rusu, R.B.; Cousins, S. 3D is here: Point cloud library (pcl). In Proceedings of the 2011 IEEE International Conference on Robotics and automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 1–4.
93. Miller, A.T.; Allen, P.K. Graspit! a versatile simulator for robotic grasping. *IEEE Robot. Autom. Mag.* **2004**, *11*, 110–122. [[CrossRef](#)]
94. Diankov, R.; Kuffner, J. *Openrave: A Planning Architecture for Autonomous Robotics*; Tech. Rep. CMU-RI-TR-08-34; Robotics Institute: Pittsburgh, PA, USA, 2008; Volume 79.
95. Tian, N.; Matl, M.; Mahler, J.; Zhou, Y.X.; Staszak, S.; Correa, C.; Zheng, S.; Li, Q.; Zhang, R.; Goldberg, K. A cloud robot system using the dexterity network and berkeley robotics and automation as a service (Brass). In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 1615–1622.
96. Tenorth, M.; Winkler, J.; Beßler, D.; Beetz, M. Open-EASE: A Cloud-Based Knowledge Service for Autonomous Learning. *KI-Künstliche Intell.* **2015**, *29*, 407–411. [[CrossRef](#)]
97. Gu, S.; Holly, E.; Lillicrap, T.; Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3389–3396.
98. Yahya, A.; Li, A.; Kalakrishnan, M.; Chebotar, Y.; Levine, S. Collective robot reinforcement learning with distributed asynchronous guided policy search. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 79–86.

99. Rahman, A.; Jin, J.; Cricenti, A.; Rahman, A.; Yuan, D. A cloud robotics framework of optimal task offloading for smart city applications. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; pp. 1–7.
100. Wang, L.; Liu, M.; Meng, M.Q.H. Hierarchical auction-based mechanism for real-time resource retrieval in cloud mobile robotic system. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 2164–2169.
101. Wang, L.; Liu, M.; Meng, M.Q.H. A pricing mechanism for task oriented resource allocation in cloud robotics. In *Robots and Sensor Clouds*; Springer: Cham, Switzerland, 2016; pp. 3–31.
102. Kong, Y.; Zhang, M.; Ye, D. A belief propagation-based method for task allocation in open and dynamic cloud environments. *Knowl.-Based Syst.* **2017**, *115*, 123–132. [[CrossRef](#)]
103. Li, Y.; Wang, H.; Ding, B.; Shi, P.; Liu, X. Toward QoS-aware cloud robotic applications: A hybrid architecture and its implementation. In Proceedings of the 2016 Intl IEEE Conferences Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCOM/IoP/SmartWorld), Toulouse, France, 18–21 July 2016; pp. 33–40.
104. Tenorth, M.; Perzylo, A.C.; Lafrenz, R.; Beetz, M. Representation and exchange of knowledge about actions, objects, and environments in the roboearth framework. *IEEE Trans. Autom. Sci. Eng.* **2013**, *10*, 643–651. [[CrossRef](#)]
105. Tenorth, M.; Beetz, M. Representations for robot knowledge in the KnowRob framework. *Artif. Intell.* **2017**, *247*, 151–169. [[CrossRef](#)]
106. Morante, S.; Victores, J.G.; Balaguer, C. Cryptobotics: Why robots need cyber safety. *Front. Robot. AI* **2015**, *2*, 23. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).