



A Task-Oriented Hybrid Cloud Architecture with Deep Cognition Mechanism for Intelligent Space

Yongcheng Cui¹, Guohui Tian^{1,*} and Xiaochun Cheng^{2,*}

¹School of Control Science and Engineering, Shandong University, Jinan, China

²Computer Science Department, Swansea University, Wales, UK

*Corresponding Authors: Guohui Tian. Email: g.h.tian@sdu.edu.cn; Xiaochun Cheng. Email: xiaochun.cheng@gmail.com

Received: 10 March 2023; Accepted: 24 May 2023; Published: 30 August 2023

Abstract: Intelligent Space (IS) is widely regarded as a promising paradigm for improving quality of life through using service task processing. As the field matures, various state-of-the-art IS architectures have been proposed. Most of the IS architectures designed for service robots face the problems of fixed-function modules and low scalability when performing service tasks. To this end, we propose a hybrid cloud service robot architecture based on a Service-Oriented Architecture (SOA). Specifically, we first use the distributed deployment of functional modules to solve the problem of high computing resource occupancy. Then, the Socket communication interface layer is designed to improve the calling efficiency of the function module. Next, the private cloud service knowledge base and the dataset for the home environment are used to improve the robustness and success rate of the robot when performing tasks. Finally, we design and deploy an interactive system based on Browser/Server (B/S) architecture, which aims to display the status of the robot in real-time as well as to expand and call the robot service. This system is integrated into the private cloud framework, which provides a feasible solution for improving the quality of life. Besides, it also fully reveals how to actively discover and provide the robot service mechanism of service tasks in the right way. The results of extensive experiments show that our cloud system provides sufficient prior knowledge that can assist the robot in completing service tasks. It is an efficient way to transmit data and reduce the computational burden on the robot. By using our cloud detection module, the robot system can save approximately 25% of the average CPU usage and reduce the average detection time by 0.1 s compared to the locally deployed system, demonstrating the reliability and practicality of our proposed architecture.

Keywords: Service robot; intelligent space; cloud robot; interactive robot system; robot cloud service framework



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

The growing demand for household tasks has led to a broad market prospect for service robots, and performing more types of service tasks has become a new challenge in the service robot field. Moreover, the system of those robots has some drawbacks, such as the high cost of local computation, weak robust activity, poor computing reusability, and few service applications. The above problems are unavoidable in the robotic system design.

- From the architecture design perspective, most robots' operating environment needs to use the specified version of the operating system (i.e., Robot Operating System (ROS)) and software toolkit. This way can ensure the stable operation of the system. However, different brands and models of robots lack interoperability in infrastructure, information interfaces, and low reusability of computing resources.
- From the functional design perspective, to ensure the appearance of service robots and flexible actions, most service robots do not carry deep computing modules. Although the robot systems can complete service tasks, they can only perform basic, low-power operations. Besides, those robots also face many problems, e.g., low task efficiency, slow computational feedback, shallow cognition level, etc.

Adopting a cloud computing model is an effective means of enhancing local computing power [1–4]. In 2018 Qi et al. established a novel privacy-preserving and scalable mobile service recommendation approach based on SOA [1]. Wu et al. implemented multi-objective evolutionary list scheduling by cloud computing [2]. Cloud computing systems also can shine in the field of the Internet of Things (IoT) [3]. Meanwhile, Al-Rakhmi [4] explored methods for evaluating the performance of cloud services. Therefore, we consider adding cloud architecture to the home service robots to enhance the performance of the Intelligent Space.

The concept of Intelligent Space (IS) was first proposed by the Hashimoto laboratory at the University of Tokyo, Japan [5,6]. Its space environment is composed of a variety of distributed sensors, which can be used by the system as hardware resources. In 2010, ASORO Laboratory in Singapore proposed a cloud robot architecture named DaVinci [7], which is the prototype of the robot cloud architecture at present. DaVinci is developed based on the Hadoop cluster and combines communication mechanisms in ROS, which aims to realize information sharing in different robot ecosystems. With the development of cloud computing, the robot cloud platform as an information center can provide scalability and parallelism for service robots. Europe started RoboEarth [8] project in 2011. With the help of cloud computing technology and a huge knowledge base, RoboEarth makes all kinds of robots share and update information about the environment, tasks, and objects. In 2015, Hunziker et al. [9] combined the RoboEarth cloud service architecture with the distributed execution mechanism of the Ubiquitous Network Robot Platform (UNR-PF) to propose the Raptuya Cloud robot platform. In 2019, the semantic knowledge-based robot service mechanism [10] was proposed to solve the problem of heterogeneous service data in IS, which sought a breakthrough for the active service of robots in IS.

The studies mentioned above provide valuable architectural exemplars of robotic cloud computing, which enable robots to have a greater capacity to perceive their environment, but they are mostly inadequate in designing solutions for executing service-oriented tasks. Moreover, these architectures often neglect the impact of computational power on the operation of cloud systems, which may add an unnecessary computational burden to IS. With the diversification and deepening of service tasks, IS is required to have more powerful cognitive and computing capabilities to deal with complex application scenarios. However, there is a lack of an effective software system to carry out unified

task planning during the operation of service robots, which cannot provide users with an intuitive display. In addition, the installation, upgrade, and maintenance costs of the software environment and modules of the robot operating system are extremely high. So far, there is still no general solution to such problems in IS. Therefore, it is essential to design a dedicated service robot cloud architecture system that integrates the necessary modules of service robot operation. This system will help the robot alleviate the computational burden while ensuring efficient information transmission. In addition, it is also important to ensure that the robot has sufficient prior knowledge to support its successful completion of service tasks.

To this end, we design a novel hybrid cloud robot service framework for the IS service robot, which is oriented to low-level computing, task-oriented planning, and execution. It is mainly composed of public cloud, private cloud, and service robots. This service framework is used to solve the problems such as weak scalability of the underlying algorithm, insufficient local computing power, and poor service application ability. To demonstrate that the proposed system can enable the robot to efficiently extract, reason, and utilize knowledge in a home environment, a series of evaluation criteria and validation experiments are designed in this paper. The key contributions of this paper can be outlined as follows.

- We design a hybrid cloud architecture system for intelligent space based on SOA, which consists of the physical resource layer, the service execution layer, the service implementation layer, the SOA interface layer, and the service request layer. To the best of our knowledge, this is the first time that SOA technology has been used in home service robots.
- To maintain greater applicability, versatility, and stability characteristics than the local server architecture while greatly reducing the computing power overhead of the robot itself, we design distributed functional modules in this system.
- To ensure the accuracy and intelligence of service execution, we construct the domain knowledge representation in the hybrid cloud architecture. It simultaneously considers both public service and personal preference knowledge while bridging the gap in knowledge updates and extractions.
- To better support the experimental work of the proposed architecture, we develop an intelligent space visualization system based on B/S architecture with an emphasis on aesthetics and interactive experience.
- The experimental results show that the proposed framework well solves the problems of information interaction between intelligent space systems and users. What's more, it is proven to be a state-of-the-art framework in the field of home service robots.

This paper first introduces the architecture of a constructed cloud system ([Section 3](#)), followed by the intelligent private cloud platform service mechanism modeling and the detailed presentation of the key functional modules involved in the proposed system ([Section 4](#)). The effectiveness and feasibility of the proposed method are then demonstrated through experimental verification of each module ([Section 5](#)), and conclusions are drawn accordingly ([Section 6](#)).

2 Related Works

In this section, we review the methods and techniques commonly used in representative studies related to our work.

2.1 Intelligent Space (IS) and Smart Home (SH)

The initial IS contained a series of computers and vision cameras, subsequently. With the development of IS, its foundational notion has widened more widely. In general, SH is a home environment with a variety of sensors, small devices as well as infrastructures in union with communication techniques and information [11–13]. As a ramification of IS, the core idea of SH is to combine the distributed context-conscious actuators and devices with the home environment. Correspondingly, SH can comprehend and perceive the contextual data and information (i.e., users, objects, etc.) and then perform related service allocation [14]. It is worth mentioning that SH, with a blend of IS and robot, can address the issues which are hardly able to be resolved by a robot alone, including observing the overall home environment or discovering things efficiently when the robot enters a new environment. Lee et al. [15] designed an IS system in which robots could independently recognize the discovery and navigation of the obstacle. Reference [16] designed an indoor IS to correctly and quickly assess the robot's position. Cvitić et al. [17] proposed an ensemble machine-learning method for device classification in IS. The above researches show that hybrid IS can effectively improve the robot's capabilities. However, how to implement and conduct service allocation properly is still lacking in the research on IS and robotics.

2.2 Ontology-Based Knowledge Representation

Cloud-based knowledge representation models are more conducive to knowledge update and utilization [18]. Ontology-based knowledge joint representation models are diffusely applied in robotics and corresponding domains. The ontology knowledge model is widely employed in robotic systems. OpenCyc [19] offered a universal upper ontology recorded in Web Ontology Language (OWL). Open Robots Ontology (ORO) and KnowRob [20] widen the upper ontology proposed in OpenCyc for robots on indoor service. KnowRob offered a diversity of classes presenting cases, things, assignments, motions as well as mathematical notions, etc. KnowRob could present specifics of individuals, including components and abilities of robots and affordances of things [21]. There are a lot of applications and proofs represented, which are on the foundation of KnowRob. For example, web-based and cloud-based knowledge services were employed to conduct assignments [22], e.g., serving a patient and making a pancake. ORO can offer things and performs tasks. Meanwhile, it is also applied to perform tabletop assignments like cleaning and packing.

2.3 Recognition Technology

In recent years, object detection frameworks for robotics are evolving rapidly [23]. The reason is that image recognition has matured, which can accomplish more complex detection tasks. So far, many deep learning-based methods have been proposed in image recognition. One of the most outstanding methods is convolutional neural networks (CNNs) in image recognition. R-CNN employed a selective seek [24] for a region, in which CNN was applied to extract and classify features [25]. Researchers propose Fast R-CNN [26] and Faster R-CNN [27] to enhance detection action as same as the speed of R-CNN. In 2016, Liu et al. [28] proposed the single-shot multi-box detector (SSD), which has the same detection accuracy as Faster RCNN and faster detection speed. The model You Only Look Once (Yolo) [29] fulfilled a quicker detection rate of forty-five fps that is suitable for real-time applications. Nevertheless, it has the disadvantages of poor accuracy and low recall. Yolo-V3 [30] solved this issue by using prepend boxes with variable sizes and enhanced detection action by adopting focal loss. Yolo-V4 [31] and Yolo-V5 [32], which followed in 2020 and 2021, are even more significant in terms of detection performance.

By building a household object detection (HOD) dataset, we propose a deep learning network suitable for home object detection based on R-CNN, which is used for object cloud detection. Meanwhile, the integration of detection-deep cognition is realized, and the detection results are displayed in the intelligent interactive system, which makes use of a data interface and visualization technology.

2.4 Communication Technology

To realize real-time communication between the robot and the IS server and also to ensure the proposed system transmits information efficiently [33]. WebSocket technology is applied [34], which is used to actively send the status information (visual information, location information, task information, robot state, etc.) of the robot to the interactive client, and also receive the echoed information simultaneously transmitted by the client through the protocol, then analyze, process and execute the information.

2.5 Human-Robot Interaction Techniques

In the past few years, Human-Robot Interaction (HRI) in IS has received significant attention. In general, most of the target groups of service robots are people with limited mobility, and most of them are the elderly and the disabled [35]. Therefore, the HRI mode in IS should have effective and real-time interaction. Most research shows that older people are optimistic about the interaction of home service robots [36]. Therefore, we can specifically design the HRI system for the elderly living alone [37] and integrate the interaction core into mobile devices. By designing a remote interaction system [38] and setting some permissions, people other than the service target (relatives, friends, etc.) can obtain the current service status information of the robot online, thereby greatly improving the service quality of the robot.

In this paper, we design an HRI system based on B/S architecture, which can adapt to any TV, PC, and mobile device with different resolutions and display the position, vision, cognition, and state information of the robot system in real-time.

2.6 Probabilistic Latent Semantic Analysis (PLSA)

Specifically, PLSA introduced a latent dependent variable [39] related to different polysemous occurrences to define a semi-generative data model. This model is generated in the following PLSA process:

- (1) Document d is selected, with a probability of $p(d)$;
- (2) A latent class z is picked, with a probability of $p(z|d)$;
- (3) A word w is generated, with a probability of $p(w|z)$.

The PLSA generation process is always considered to be ambiguous since topic mixes are set by documents that simultaneously generate topics. This also leads to no natural way to infer previously unseen documents [40]. Furthermore, the semi-generative data model is treated as a special model considering the linear relationship between the number of training documents and the number of PLSA parameters.

3 Architecture

This section takes the private cloud as an example to introduce and describe the basic SOA architecture and overall hybrid cloud architecture in detail, layer by layer.

3.1 Basic Architecture Based on SOA

Hybrid cloud architecture consists of public, private, and service robots. The structure of the public and private clouds is similar, constructed by using SOA. We take the private cloud as an example. The basic architecture of its model is shown in Fig. 1.

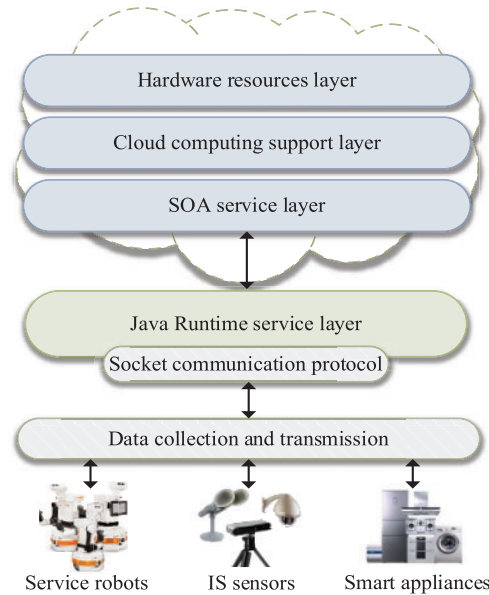


Figure 1: Basic IS private cloud platform based on SOA

The service architecture model based on SOA has been widely applied among the classical models of cloud robot distributed architecture. This model will mainly divide the cloud server into three parts: the physical resource layer, the cloud computing support layer, and the SOA service layer. In the SOA service layer, hardware devices use the unified data acquisition interface, which can eliminate the cloud's dependence on the ROS environment and shields the differences between heterogeneous robot bottom layers. It can provide universal services for heterogeneous robots more conveniently, making service function expansion and management of the system more flexible. In addition, data acquisition and transmission interfaces are designed for robots and ISs (e.g., camera, depth camera, microphone, etc.) respectively. After service invocation planning, robots and IS can realize asynchronous data interaction with private cloud servers.

3.2 Overall Hybrid Cloud Architecture

The public cloud is used to store common resources (i.e., the public knowledge base and deep algorithm model), which provides an interface for users to upload models to service initial downloads on the private cloud. The private cloud directly faces different service task environments and users, which consist of IS, service robots, and private cloud servers. The IS is connected to the local private cloud server through the interface layer to further strengthen the system's cognitive ability to the external environment information. In addition, its platform can act as an execution and interaction mechanism to respond to service requests based on personalized knowledge in the private cloud. Distributed architectures design both public and private clouds. Fig. 2 shows the private cloud architecture, which includes the Physical resource layer, Service execution layer, Service implementation layer, SOA interface layer, and Service request layer. The details of each layer are as follows:

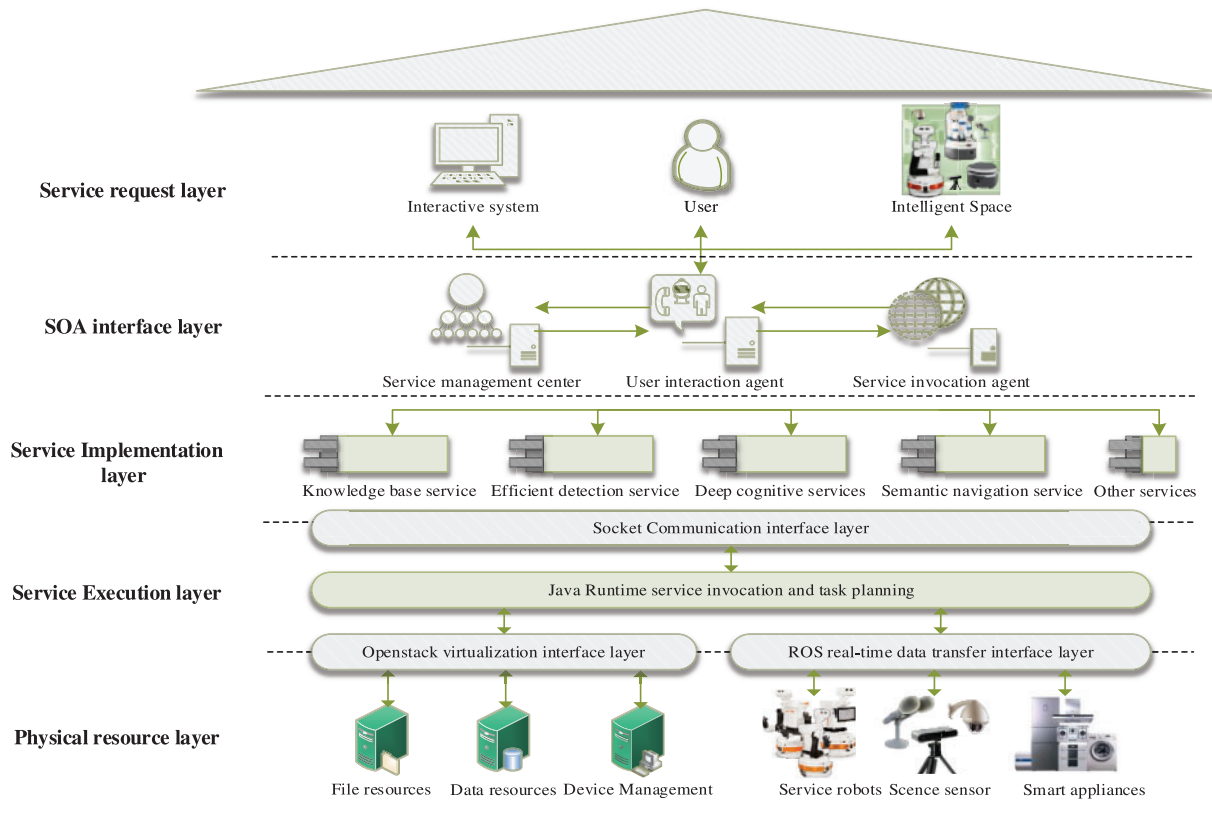


Figure 2: Architecture of intelligent space private cloud. The basic structure can be described as the physical resource layer, service execution layer, service implementation layer, SOA interface layer, and service request layer

Physical resource layer. It mainly realizes base resource and device management, which includes collecting and managing basic information such as file information and ontology knowledge base. Moreover, it also includes data collection and transmission for robots and sensors in IS.

Service execution layer. It realizes the planning and calling of service tasks, responsible for the information interaction and transmission between different layers, avoiding the blocking problem caused by transmission problems.

Service implementation layer. The private cloud is a module of service-oriented tasks built at the service implementation layer. It improves the scalability and robustness of cloud robot services by providing users or robots with more complex service tasks. In addition, it realizes the integrated representation of IS service knowledge and fuses the ontology models from different sources to realize the automatic construction of the smart spatial knowledge base. By applying the foreground interaction system, the constructed intelligent spatial knowledge base can be stored from the public cloud to the private cloud to realize the sharing of IS knowledge. Meanwhile, the private cloud realizes object-deep cognition based on image features and associated service knowledge in the hybrid cloud-distributed knowledge base. It obtains deep knowledge of objects and the environment to improve the robot’s object recognition and comprehensive cognition ability and provides complete auxiliary operation information. For services requiring generic knowledge solutions, the private cloud can

transfer the basic ontology knowledge base in the public cloud to itself and implement corresponding services through the knowledge transfer mechanism. This paper will focus on the service mechanism construction and principle of local private cloud.

SOA interface layer. The main function of this layer is to provide asynchronous data interaction between service modules, IS, and users. Besides it also includes a task planning and scheduling module, which converts the service request into serialized subtasks for execution.

Service request layer. This layer aims to realize user interaction and IS information, interactive systems, users, and intelligent devices in space by a service request and interactive agency for information transmission, thus planning service tasks, i.e., broken down into subtasks or atomic tasks, and calling the implementation layer of each function module to complete the corresponding task. In human-robot interaction, we design an IS cloud display platform to perform a series of visual displays of service information.

4 Intelligent Private Cloud Platform Service Mechanism Modeling

The IS realizes the comprehensive perception of the “human-object-environment” in the home environment. The personalized private cloud platform effectively uses and integrates perceptive information. Moreover, various task-oriented service modules can be deployed in the service implementation layer for invocation. Overall, the introduction of IS is necessary for the whole system. On the one hand, it improves the cognitive and analytical ability of task information in IS, which enables robots to provide more personalized, autonomous, accurate, and stable services. On the other hand, as an information interaction mechanism between robots and users, the IS platform visually displays the response results of personalized knowledge of private cloud service requests.

4.1 Integrated Representation Model of Multi-Domain Knowledge under a Hybrid Cloud

The home environment constantly changes, leading to object diversity and environmental dynamics. The maintenance of domain knowledge is the guarantee for the implementation of service tasks in the hybrid cloud architecture of IS. For the classification of object knowledge, the ontology-based representation method of object knowledge proposed earlier [41] divides the attributes into six categories (i.e., visual, category, physical, functional, and operation), combining the characteristics of home objects and environment. Finally, the systematic and organized description of objects is realized. The integrated representation model of multi-domain and multi-attribute knowledge at the class level is formed, as shown in Fig. 3.

The semantic knowledge represented based on OWL can be stored in MySQL database through the corresponding relationship, which can be used as an ontology knowledge base for cloud invocation, as is shown in Fig. 4. Based on previous work, we design a private knowledge base and a public knowledge base. Specifically, the public knowledge base is used to store common knowledge, which is usually unchanged. The private knowledge base is used to store personalized knowledge, and each private knowledge base can be different. Moreover, the private knowledge bases can access the public knowledge bases through interfaces. As the knowledge is dynamic, to meet the personalized requirements of the family environment, the private knowledge base enables the database table structure to map to element content through the updating mechanism of multi-domain knowledge, which completes the generation of Extensible Markup Language (XML) semantic mapping schema. The Jena tool interface in Python retrieves JSON files from the private knowledge base to add knowledge.

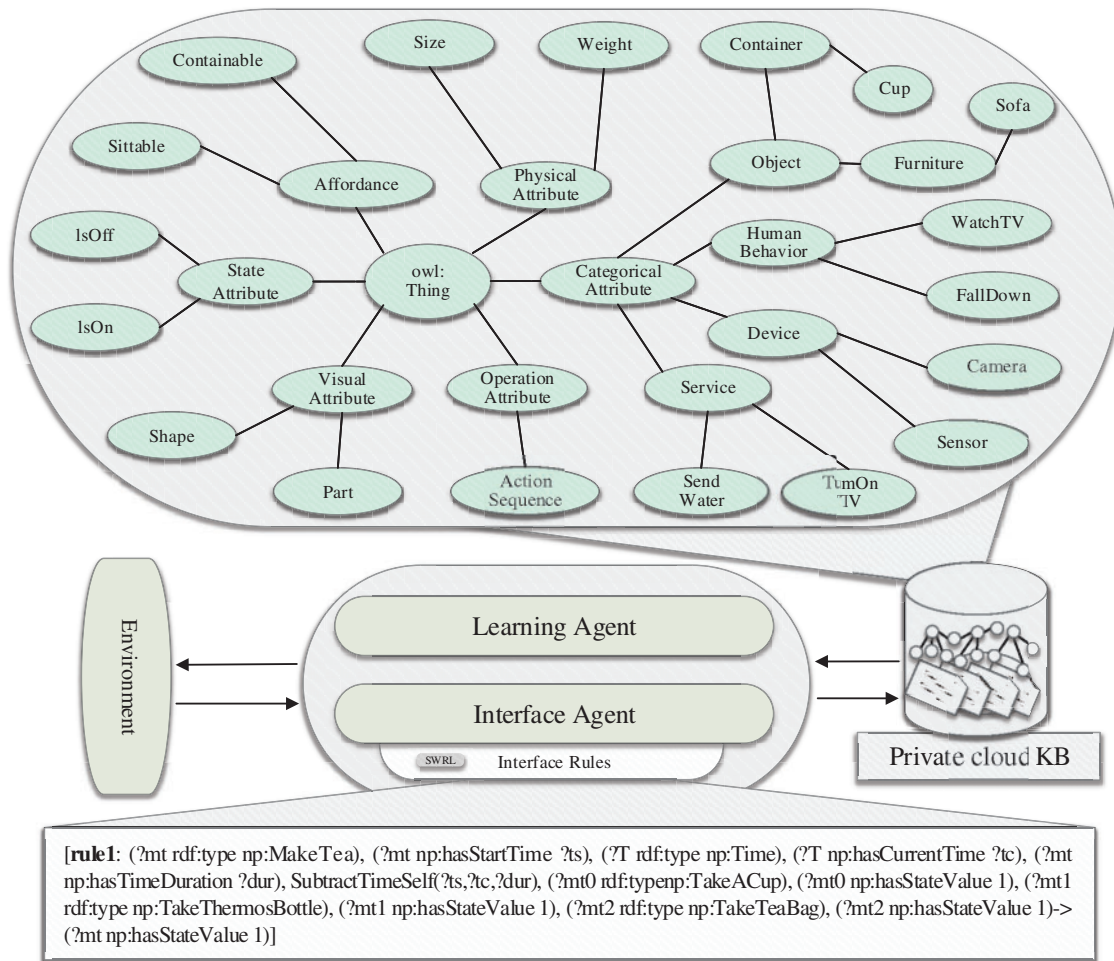


Figure 3: Multi-domain knowledge model on hybrid cloud. An integrated representation of knowledge in multiple fields. Subdivide the attributes into six categories (visual, category, physical, functional, operation, etc.), combining the characteristics of family objects and environment

Although the cloud knowledge base can store and update abundant semantic knowledge, it needs to gain the ability to reason and discover hidden knowledge. To explore the reasoning ability of the cloud knowledge base, we establish Semantic Web Rule Language (SWRL) inference rules in the knowledge base. The basic semantic knowledge in the cloud knowledge base is utilized, which uses the Jess engine to combine reasoning to get the hidden knowledge in the IS.

The basic reasoning form of the SWRL inference rule is *Premise (body) → Conclusion (head)*. The premise and conclusion can include single or multiple basic propositions, with a logical relationship between the basic propositions. For example, SWRL rules in ontology use two constraints $C(a)$ and $C(b)$, where: a and b can be variables or an instance in the ontology. C stands for class; P represents the attribute. The following is the inference for the basic co-occurrence relationship (Rule_In_Location) of objects in IS:

$$\text{container}(?a) \wedge \text{in_location}(?a, ?b) \wedge \text{inside}(?c, ?a) \rightarrow \text{in_location}(?c, ?b). \quad (1)$$

The premise of this rule is that given a belongs to the class *container*, it is stored in place b , and if c is stored in a , it can be concluded that c is stored in place b . Therefore, all the hidden knowledge satisfying the above reasoning rules can be deduced from the public and private knowledge bases.

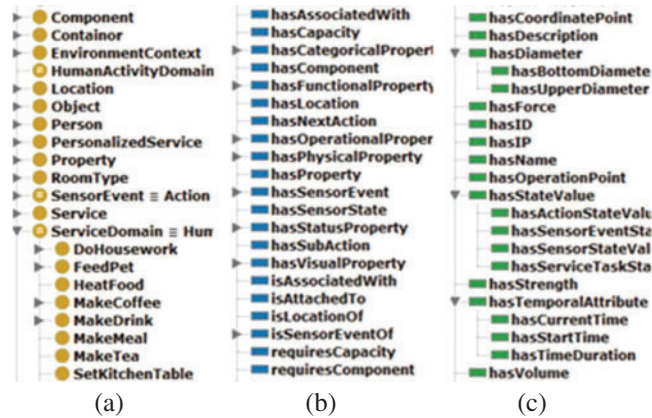


Figure 4: Example of a private cloud knowledge base. All service knowledge is stored in the private cloud ontology database, including ontology description knowledge of six attributes (a), task planning knowledge (b), and action sequence knowledge (c). The private cloud knowledge base can be updated online by the file management system through the public cloud

4.2 Object Cloud Detection Model

The computing power of private cloud servers far exceeds that of service robots. By deploying the recognition module in the private cloud, it can effectively solve the problem of high cost and slow computing resources. And with the Socket communication interface, the private cloud server can effectively improve the call efficiency of different function modules to ensure real-time communication, which also can operate the bottom layer of the binary data stream of files, then send the image pixel information into memory through the form of the data stream, eventually, the data are transmitted to a cloud server through the communication protocol. Private cloud applications enter the server's address to get a keyframe image of the robot camera. Frame image is preprocessed through the designed Socket interface call recognition service. Our detection model is based on Region Convolutional Neural Network (R-CNN) guided by prior knowledge under input selective attention guidance [42], which is trained by the HOD dataset mentioned below to implement cloud detection of household objects. The model structure is shown in Fig. 5.

4.3 Household Object Detection (HOD) Dataset

We construct the HOD dataset by combining multiple methods (i.e., manual annotation method, automatic tagging method, and extraction from a public dataset) to train the detection model of objects in the family scene. The manual annotation method real scene instance diversity is good, but the manpower cost is high. Although the automatic labeling method has a lower time cost, it generates relatively poor instance diversity. Images extracted from public datasets are of high quality. The quantity is large, but the number of categories is limited. The advantages of the three methods complement each other, thus enriching the dataset scale and improving the dataset quality.

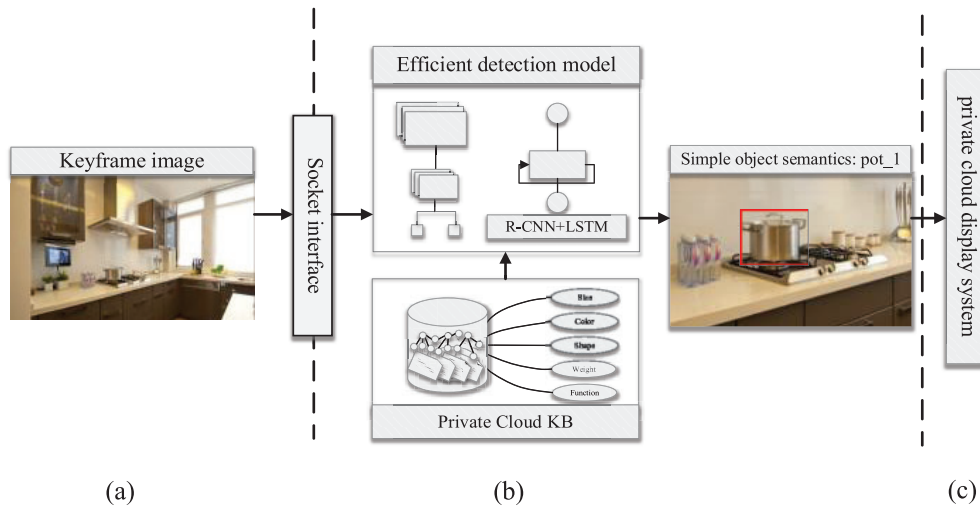


Figure 5: Schematic diagram of the detection model. (a) Robot system. (b) Private cloud platform. (c) Interactive system

The dataset is constructed in VOC format. Images and annotation files are stored separately according to categories, and datasets constructed in different ways are stored separately. Datasets are integrated as needed when used. Among them, the data source extracted from the public dataset is the COCO dataset, which contains a large number of family objects. The completed HOD has been extended so far, including a total of 158,648 images in 99 categories of household objects. Table 1 shows the statistics of the current household goods detection dataset.

Table 1: Statistics on the HOD dataset

Annotation methods	ImageNum	ClassNum
Manually annotated	29200	60
Automatic annotated	81888	47
Open datasets extraction	47560	31

4.4 Deep Cognitive Model

Based on the result of item cloud detection, the deep cognitive model can be realized by using the private knowledge base built with PLSA technology. In semantic knowledge, PLSA can inference potential semantic structures between contextual words [39,43]. With the multi-attribute knowledge model on the private cloud, as well as semantic description text on the Internet, the semantic structure of the object is elicited. Meanwhile, based on this structure, other semantic knowledge is associated and deduced to form the deep cognition information associated with service tasks.

The topic model proposed in this chapter is a latent variable model for co-occurrence data jointly with an unobserved topic model $z_k \in \{z_1, z_2, \dots, z_k\}$, which z_k is an observation representing the occurrence of a particular word in the corresponding document. $P(d_i)$ characterizes the probability that a word will be observed to occur in a particular document. d_i . $P(w_j|z_k)$ characterizes the

conditional probability of a topic, conditional on the unobserved z_k for a particular word, and finally $P(z_k|d_i)$ characterizes the probability distribution of a document in the latent variable space.

Based on the above representational design, thus obtaining an observation pair (d_i, w_j) and discarding the topic z_k , we can transform the data generation process into a joint probabilistic model \mathcal{L} .

$$\begin{aligned} \mathcal{L} &= \sum_{i=1}^N \sum_{j=1}^M d(d_i, w_j) \log P(d_i, w_j) \\ &= \sum_{i=1}^N n(d_i) \left[\log P(d_i) + \sum_{j=1}^M \frac{n(d_i, w_j)}{n(d_i)} \log \sum_{k=1}^K P(w_j|z_k) P(z_k|d_i) \right] \end{aligned} \quad (2)$$

PLSA-based methods are mainly realized with the maximum likelihood principle. This method performs statistical analysis of a large number of task-related texts to look for potential semantic structure. It has the advantage that no specific semantic encoding is required. As shown in Fig. 6, this method uses semantic structure to represent words and text by relying on the relationship between things mentioned in the context, eliminating the correlation between words and simplifying the text vector. Underlying topic reasoning is performed on the current service task to define an appropriate generative sequence model for semantic knowledge. Finally, it combines the constructed semantic knowledge base with the result of object detection. It comprehensively uses the latent semantic analysis model to retrieve the state and operation of the returned object to realize the deep cognition of the object.

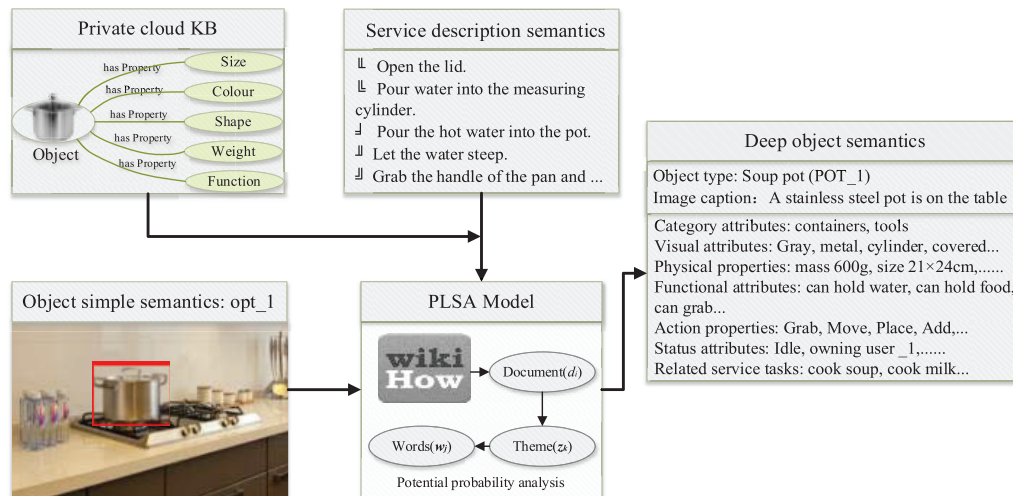


Figure 6: PLSA objects deep cognition model. Relying on the private cloud knowledge base, the PLSA model based on robotics-oriented service tasks can extract the theme of service tasks and key elements

5 Experimental Evaluation

To evaluate the feasibility and efficiency of the proposed framework, we conduct experiments on cloud platform performance in both simulation and real environments, the experimental robot is shown in Fig. 7. This section is devoted to revealing the details of these experiments and discussing the results.

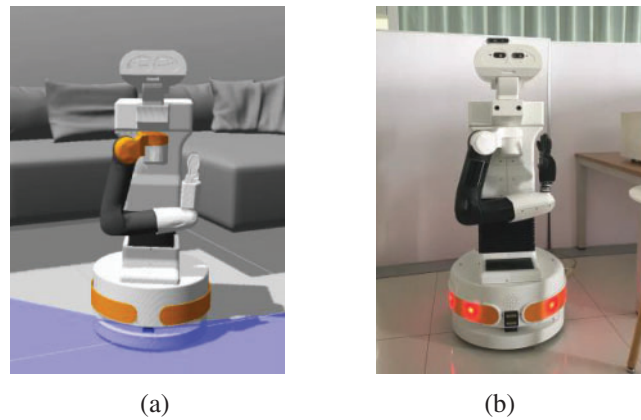


Figure 7: Experimental testbed: TIAGo in the simulation environment (a) and real environment (b)

5.1 Intelligent Space Cloud Display System

We built the Intelligent space cloud display system (ISCDS) based on B/S architecture to verify the proposed model in IS experiment scene, as shown in Fig. 8. The scenes simulating the home environment include robots, various household appliances, and furniture. Common objects in the environment include tools for robot operation and commodities to be identified. We modeled the objects in each position in the home environment at an equal scale and used 3D projection technology to generate the scene map, which was used as a display map in ISCDS to indicate the robot’s position and posture in real-time.

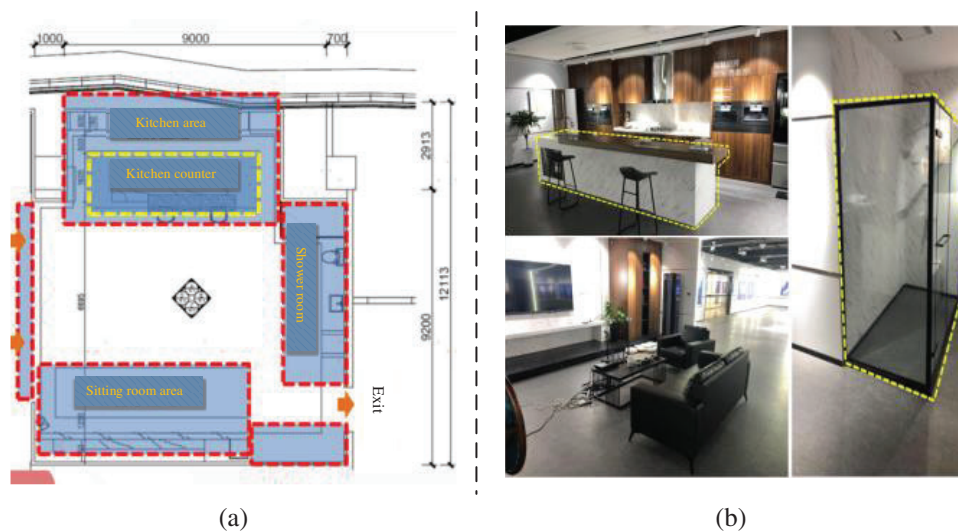


Figure 8: Real home scene layout. (a) The home scene area plan. (b) Some photographs of the scene

The backend of the experimental platform in the private cloud server is developed in the Java Spring Boot framework with IntelliJ IDEA software in Java language. The development of the front end involved Websocket and HTML5 front-end technology to design interactive pages, showing various information about the robot, including real-time position, running state, task execution state, camera detection, deep cognition, etc.

With the visualized map as the background and the top view of the 24-way robot as the direction marker, the position and orientation information of the robot can be displayed on the interface by superposition of the two. Fig. 9 shows that the current task of the robot is *Open refrigerator*. The last task of the robot is to *Bring cola*.

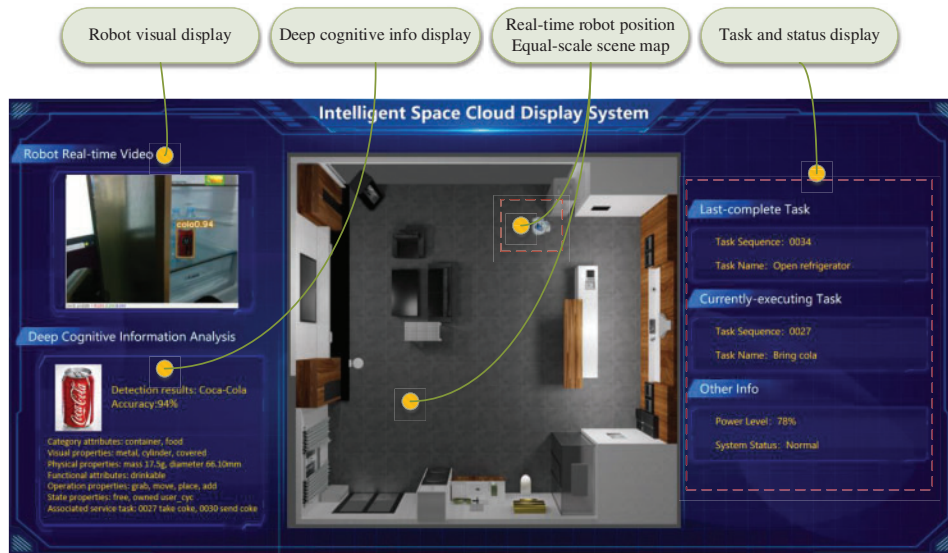


Figure 9: Intelligent space display system interface. The real-time display system visualizes the robot-path process, task process, and deep cognitive information intuitively

5.2 Private Cloud Service Network Stress Comparison Experiment

We use the Siege stress testing tool to simulate high concurrent network access to test the operation stability of private cloud services. This stress test takes the service module as the target detection and is divided into four scenarios for comparison: 100 connections, 200 connections, 500 connections, and 800 connections. Each scenario is tested six times. Meanwhile, the object detection service has 100, 200, 500, and 800 requests at the same time, and each round lasts for 15 s of service call pressure. Table 2 counts the number of successful service calls, service call failures (or timeout times), and QPS (the number of successful service calls that return correct responses per second) in the 15 s.

Table 2: Private cloud object detection stress test comparison results

	100 connections			200 connections			500 connections			800 connections		
	Success	Failure	QPS	Success	Failure	QPS	Success	Failure	QPS	Success	Failure	QPS
1	16927	2912	1128.46	28538	5263	1902.53	38726	6858	2581.73	52034	9181	3468.93
2	15479	2685	1031.93	26971	5349	1798.06	40129	7260	2674.86	51371	9064	3424.73
3	15634	2896	1042.26	29493	5180	1966.20	37272	6697	2484.80	52923	8784	3528.20
4	16349	2949	1089.93	28217	4957	1881.14	39324	6799	2621.60	50172	8931	3344.80
5	15934	2752	1062.26	27861	5513	1857.45	38435	6831	2562.33	53432	9212	3562.13
6	17106	2845	1140.40	28196	5072	1879.78	37238	7032	2482.53	50579	8674	3371.93
Avg.	16238	2840	1082.54	28213	5222	1880.86	38521	6913	2567.98	51752	8974	3450.12

By comparison, as connections increase, the total number of successful and failed accesses to the system grows. Even with 800 connections, the average QPS is about 3450 connections, thus verifying that the system can operate efficiently with a large number of visits. The results show that this paper’s private cloud service architecture has a certain access carrying capacity when simulating high concurrent access—specifically, the more access times, the higher the QPS.

5.3 Knowledge Representation and Reasoning Comparison Experiment

For knowledge utilization by robots, we designed comparative experiments on knowledge representation and reasoning to verify the cloud’s knowledge extraction and updating capability. As shown in Fig. 10, *BodyCup* class is part of the instance object knowledge representation template and contains the following knowledge:

- *CategoryAttribute, VisualAttribute, AffordanceAttribute, PhysicalAttribute, StateAttribute, LocationAttribute and its subclasses;*
- *HasAffordance, hasColor, isLocatedIn, hasMaterial, hasPart, hasShape, hasState object properties and their sub-properties;*
- *HasAreasqcm, hasVolumecucm, hasSizecm, hasWeightN data attributes and their sub-attributes;*
- *The class description of the functional components of each object.*

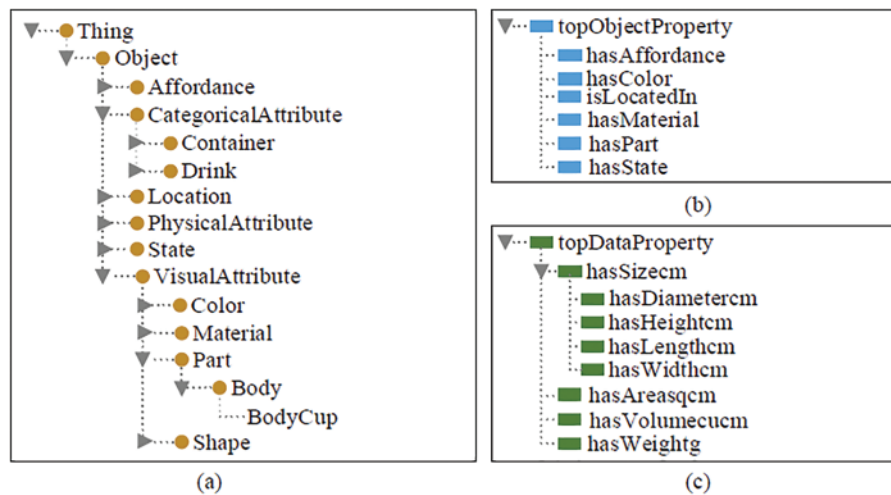


Figure 10: Example of object knowledge representation template. Add instance objects to the private cloud knowledge base according to this template. The systematic and deep representation of object knowledge is carried out by using its category attributes, physical attributes, state attributes, visual attributes, functional attributes, and operational attributes

We take *BodyCup* class as an example. The class description of *BodyCup* may have the colors, shapes, materials, sizes, weights, functions, positions, and states. In the Open-World Assumption, DL as the logical basis of OWL, says that if you do not explicitly state that something does not exist, then it does not exist. For example, *hasShape some Cylinder* declares that the shape of *BodyCup* is *Cylinder*, but this does not mean it has no other shape. Additionally, to claim that the cup is only cylindrical, it can be declared by the closed axiom *hasShape only Cylinder*. The *BodyCup* class description enables the robot to conceptualize the entire *BodyCup* class. Meanwhile, all instances of the *BodyCup* class can inherit this description.

We transform XML structured data into OWL form using the mapping relationship between elements and knowledge representation templates in item semantic mapping schema. As shown in Fig. 11, Protégé5.1 is used to visually display the knowledge of Cola (Cocacola1) in the object ontology model, which verifies the correctness of the constructed object knowledge.

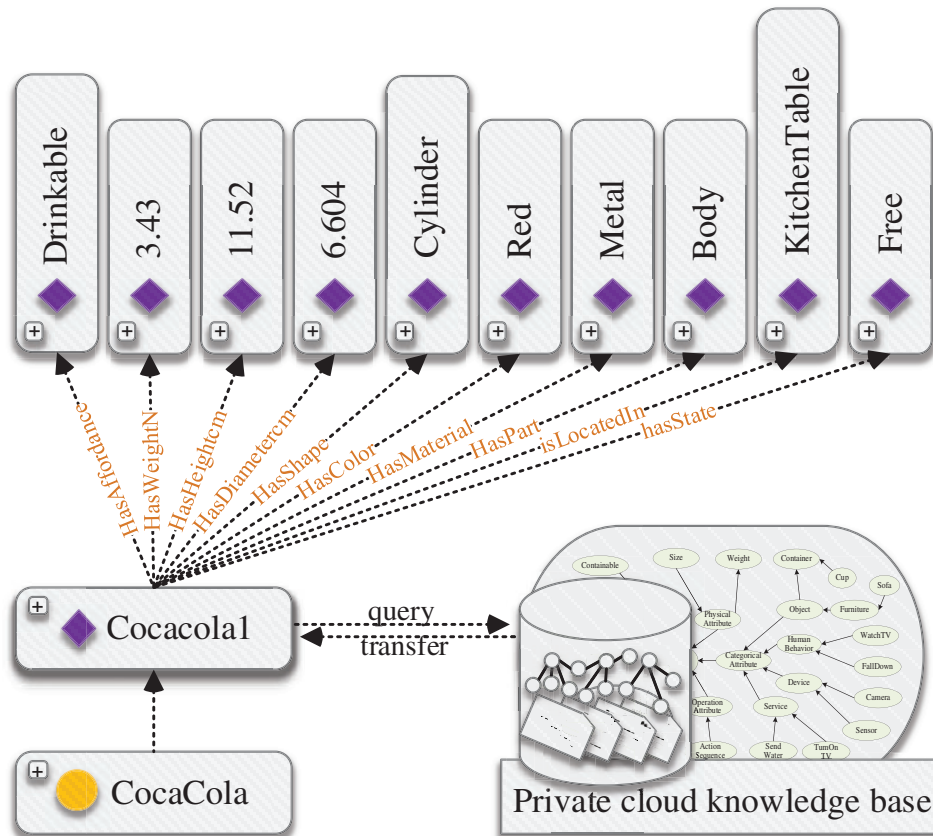


Figure 11: Example of knowledge representation. Representation form of ontology instance CocaCola in Protégé5.1 on the private cloud knowledge base

The private cloud ontology model constructed in this paper contains 50 categories of objects, 568 instances, 5654 axioms, and 117 rules. This model takes 0.6 s below to complete query, reasoning, and transmission on the ontology model. Fig. 12 shows that the size of the ontology model and the number of inference rules affect the efficiency of inference and query. Overall, with the increase of axioms and rules, the required time will also increase, but it is still at the level of seconds, which has almost no influence on the execution of service tasks.

We also tested the reasoning ability of the implicit axioms. Fig. 13 shows the number of implicit axioms obtained by inference rules is more than or close to the number of existing axioms. This result indicates that the constructed SWRL rules can obtain a large number of implicit axioms. The number of implicit axioms depends on whether SWRL rules can obtain missing attributes of ontology instances.

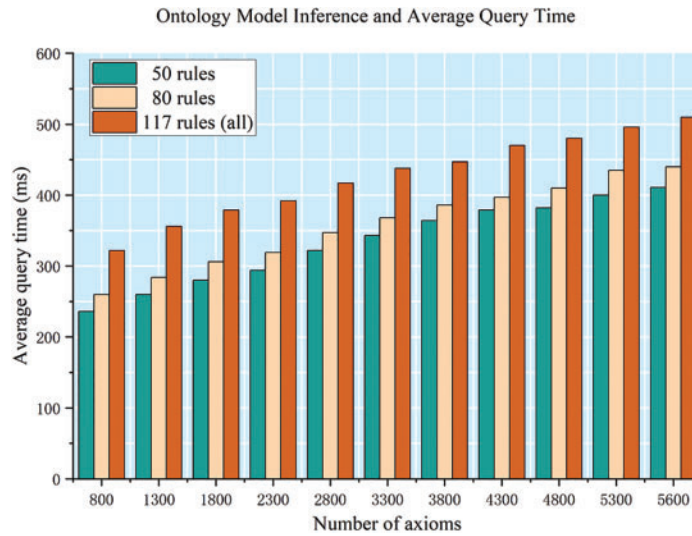


Figure 12: Ontology model inference and average query time. The average query time is statistics under 50, 80, and 117 rules. In the case of the same number of axioms, the more rule reasoning, the longer the runtime on the cloud, but all within 0.6 s

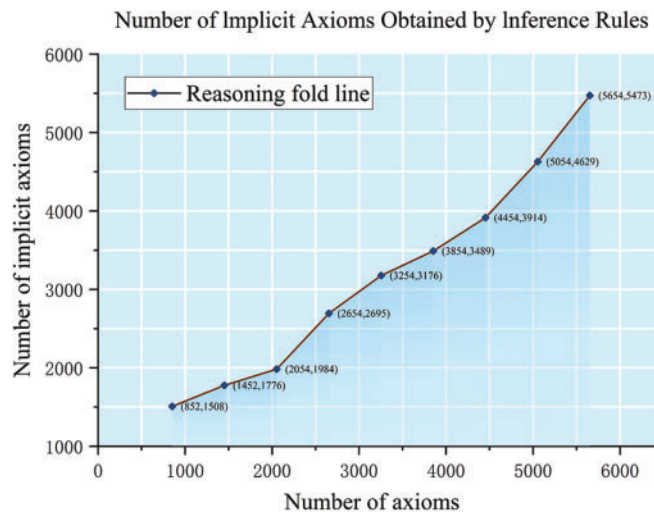


Figure 13: Number of implicit axioms obtained by inference rules. The number of implicit axioms obtained by inference rules is more than or close to the number of existing axioms. This method can deduce a large number of implicit axioms

5.4 Object Cloud Detection Experiment

In this paper, the dataset used for the detection cloud detection task is the HOD dataset which is constructed independently. The dataset is constructed in VOC format. Images and annotation files are stored separately according to categories, and constructed differently. So far, the HOD dataset contains a total of 158,648 pictures in 99 categories of household objects.

We first fine-tuned the Faster R-CNN model to improve its performance in detecting household objects. Then, we trained and validated the model using the HOD dataset with 300 epochs and a

batch size of 32. We conducted a comparative experiment with several exemplary models to verify the benefits of the fine-tuned model. The experimental results for on-cloud detection of HOD datasets are presented in Table 3, which allows for comparing the detection performance among different methods.

Table 3: Comparative experimental results of on-cloud detection of HOD datasets

Models	Precision	Recall	mAP
Faster RCNN	92.6	94.9	93.4
SSD	90.2	92.1	90.8
YOLOv3	91.4	93.9	92.5
YOLOv4	92.0	94.8	93.2
YOLOv5	92.8	95.3	94.1
Ours	93.7	96.1	95.3

According to the provided table, the proposed model achieves an average accuracy rate (mAP) above 0.95 for all class labels after 300 epochs. This represents a significant improvement, as the mAP increased by 1.9 compared to the representative two-stage detection algorithm, Faster RCNN. Additionally, compared to the widely used YOLOv5 model, the proposed model outperforms it by 1.2 in terms of detection accuracy. Therefore, the proposed improved object detection model is the top-performing model in all metrics and effectively meets the practical requirements for robotic tasks. Fig. 14 depicts the detection results of various objects in a home environment, demonstrating the proposed model's accurate object detection capabilities.

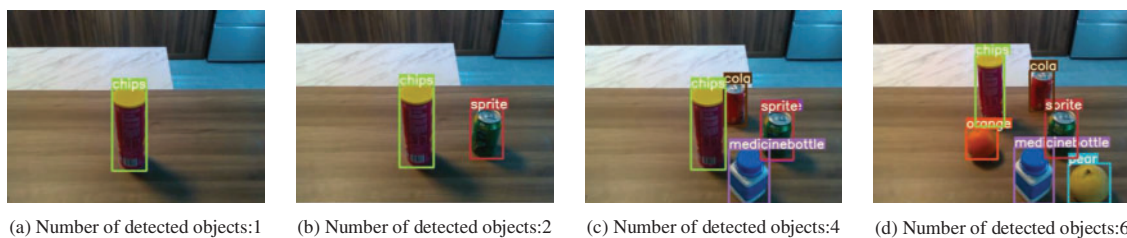


Figure 14: Single and multiple object detection results. The target detection model is evaluated on cloud detection under 1, 2, 4, and 6 common objects simultaneously. The results demonstrate that the model performs well in accuracy and response speed

To test the operational efficiency of item detection of the cloud, three different items (cola, sprite, orange, pill bottle, chips) are tested on basis of the hardware environment. As shown in Table 4, we count five items separately. The same object is tested ten times for each item to calculate the average runtime. The results show that the image recognition speed can reach between 0.4 and 0.6 s, achieving real-time detection of images.

Since target detection is one of the high-demand modules for the arithmetic power of the robot hardware. To reflect the advantages of the detection module of our designed architecture, a comparison experiment is also designed to highlight the advantages of this architecture by counting the CPU usage rates of local and cloud when running the detection program. The results are shown in Fig. 15.

Table 4: Average runtimes for object on-cloud detection at different steps

	Transmission(s)	Detection(s)	Return(s)	Published(s)	Total(s)
Cola	0.082	0.277	0.0065	0.063	0.4285
Sprite	0.079	0.352	0.0072	0.089	0.5272
Orange	0.076	0.316	0.008	0.094	0.494
Pill-bottle	0.087	0.304	0.0068	0.075	0.4728
Chips	0.074	0.347	0.0079	0.083	0.5119

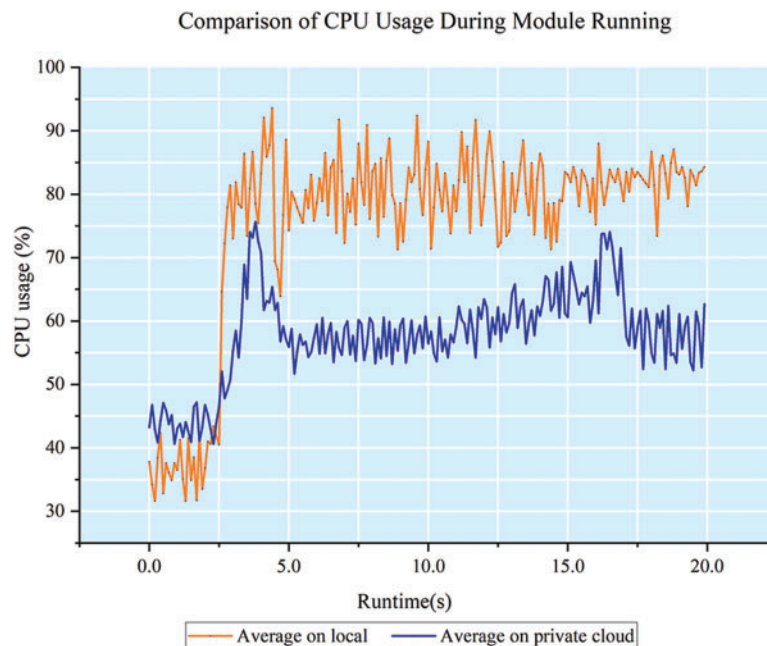


Figure 15: Comparison of line graphs of model runs in 20 s. It can be reflected that the detection module running in the cloud takes up less CPU, requiring about 60% when running. In comparison, the robot itself requires up to 85% occupancy to run, and this part of the usage could hinder the smooth operation of the robot

Meanwhile, as shown in [Fig. 16](#), we determine the detection time of common home scenarios between the local end and the private cloud by conducting a comparison test. Specifically, we test the average detection time for fifty items at both ends and analyze the results. The comparison revealed that the hardware equipment used in the cloud is superior to that of the local end of the robot, resulting in a significant computing advantage that shortened the item detection time. As seen in the result, cloud detection saves approximately 0.1 s.

Comparison of Average Detection Runtime

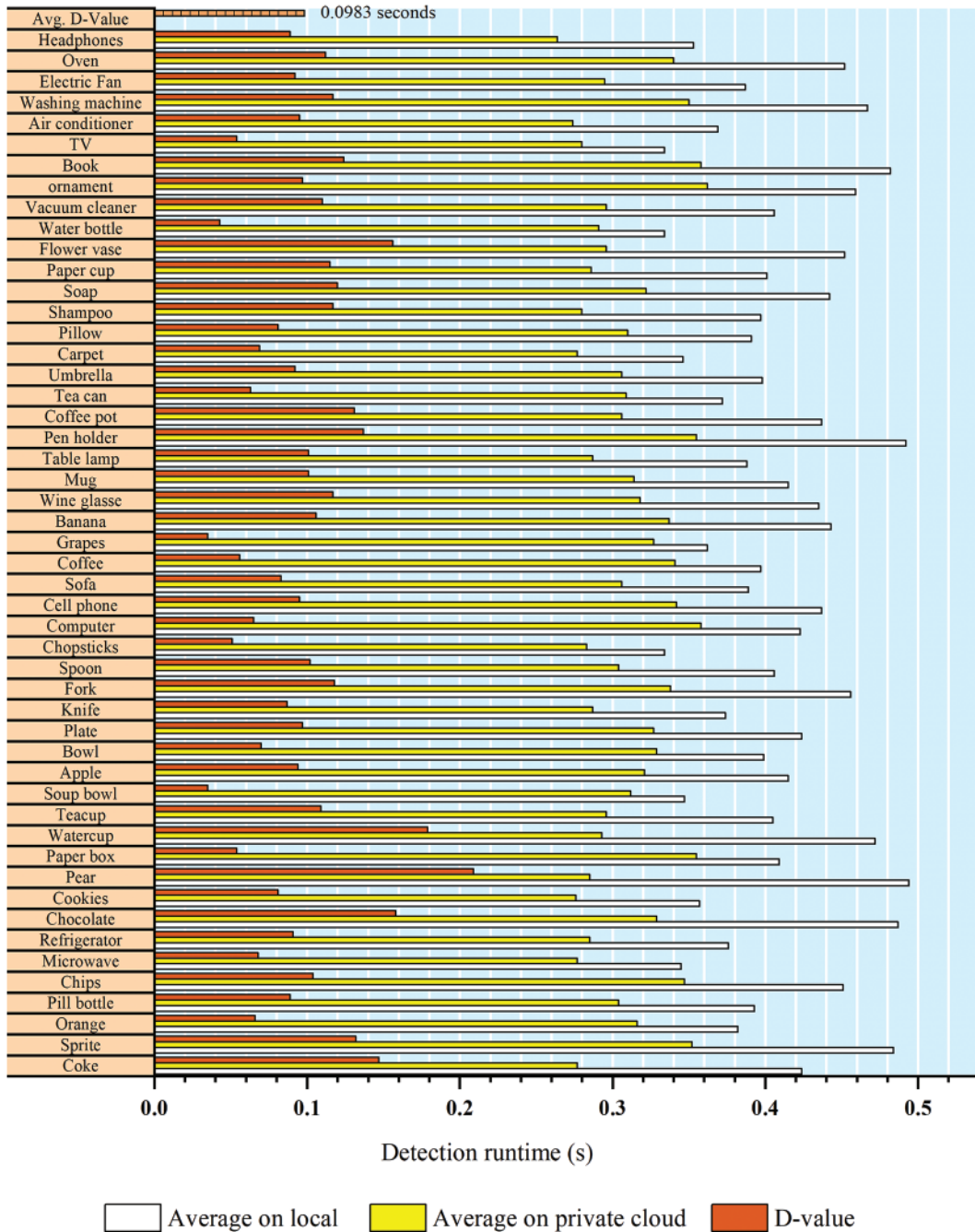


Figure 16: The comparison between the detection runtimes of the local and cloud versions, as well as their difference. The average of 0.0983 s faster detection time for the cloud-based system

Despite the increased time of data transmission and release, the computing power advantage of the cloud compensates for it, achieving the total process time of cloud detection being comparable to

that of robot local detection. This result fully confirms the practicality of the system. The experiments conducted in this study demonstrate that with a guaranteed detection accuracy of 98% or higher, the average total detection time does not exceed 0.54 s, including image transmission, information return, and result release. These results indicate that although images must be uploaded to the cloud for detection, the operation of this model can ensure real-time and efficient detection performance with a reliable network connection.

6 Conclusion and Future Work

In recent years, IS has shown great potential for improving the quality of life. However, the IS architecture designed for service robots has the problems of fixed functional modules and low scalability. To this end, we propose a hybrid cloud architecture of distributed IS based on service-oriented architecture, which aims to provide computing support for robot cognition and realize the overall perception of the “human-object-environment”. Specifically, we develop an intelligent space display system based on B/S architecture, which solves the problem of information interaction between the client and the server in the local and remote access of the intelligent space system. Next, according to deploying the ontology knowledge base in the cloud, we combined the domain knowledge representation model and SWRL rules to realize service reasoning, which ensures the accuracy and intelligence of service execution. Then, a cloud detection model based on Socket communication technology combined with ontology technology and a deep cognitive model of objects is deployed in the private cloud platform, which can proactively retrieve and return the status and operation of objects and other attributes. Moreover, it is also proved that the proposed model improves the cognitive ability of the system for service tasks and satisfies the user’s interactive experience.

Extensive experimental results show that:

- The deep information of the proposed model can be stably transmitted and shown in the display system in real-time.
- The proposed cloud architecture system runs efficiently and consistently (average QPS of 3450 connections) even under overloaded access (800 connections)
- The proposed knowledge representation and inference model runs in less than 0.6 s and can reason about a sufficient number of implicit axioms.
- The proposed cloud-based robot system can save about 25% of the average CPU usage and about 0.1 s of the average detection time in the detection module compared to the locally deployed robot system.

This paper’s hybrid cloud architecture system has great potential for enabling intelligent spaces and robots to complete related service tasks efficiently. Meanwhile, robots can enhance their performance and capabilities by applying cloud-based technologies. Overall, this paper provides valuable insights into the potential benefits of utilizing cloud-based systems in robotics applications. In the future, we can fully explore the capabilities and limitations of these systems and optimize their performance for specific use cases.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China under Grant 62273203, Grant U1813215, and in part by the Special Fund for the Taishan Scholars Program of Shandong Province (ts201511005).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] L. Qi, X. Zhang, W. Dou, C. Hu, C. Yang *et al.*, “A two-stage locality-sensitive hashing based approach for privacy-preserving mobile service recommendation in cross-platform edge environment,” *Future Generation Computer Systems*, vol. 88, pp. 636–643, 2018.
- [2] Q. Wu, M. Zhou, Q. Zhu, Y. Xia and J. Wen, “MOELS: Multiobjective evolutionary list scheduling for cloud workflows,” *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 166–176, 2019.
- [3] B. Seok, J. Park and J. H. Park, “A lightweight hash-based blockchain architecture for industrial IoT,” *Applied Sciences*, vol. 9, no. 18, pp. 3740, 2019.
- [4] M. Al-Rawajbeh, “Performance evaluation of a computer network in a cloud computing environment,” *ICIC Express Letters*, vol. 13, no. 8, pp. 719–727, 2019.
- [5] J. H. Lee and H. Hashimoto, “Intelligent space—concept and contents,” *Advanced Robotics*, vol. 16, no. 3, pp. 265–280, 2002.
- [6] K. Morioka, J. H. Lee and H. Hashimoto, “Human-following mobile robot in a distributed intelligent sensor network,” *IEEE Transactions on Industrial Electronics*, vol. 51, no. 1, pp. 229–237, 2004.
- [7] R. Arumugam, V. R. Enti, B. Liu, X. Wu, K. Baskaran *et al.*, “DAVINCI: A cloud computing framework for service robots,” in *Proc. 2010 IEEE Int. Conf. on Robotics and Automation*, Anchorage, AK, USA, pp. 3084–3089, 2010.
- [8] M. Waibel, M. Beetz, J. Civera, R. D’Andrea, J. Elfring *et al.*, “Roboearth,” *IEEE Robotics & Automation Magazine*, vol. 18, no. 2, pp. 69–82, 2011.
- [9] D. Hunziker, M. Gajamohan, M. Waibel and R. D’Andrea, “Rapyuta: The roboearth cloud engine,” in *Proc. 2013 IEEE Int. Conf. on Robotics and Automation*, Karlsruhe, Germany, pp. 438–444, 2013.
- [10] Y. Zhang, G. Tian and H. Chen, “Exploring the cognitive process for service task in smart home: A robot service mechanism,” *Future Generation Computer Systems*, vol. 102, pp. 588–602, 2020.
- [11] C. F. Freitas, J. Barroso and C. Ramos, “A survey on smart meeting rooms and open issues,” *International Journal of Smart Home*, vol. 9, no. 9, pp. 13–20, 2015.
- [12] P. C. Roy, S. R. Abidi and S. S. R. Abidi, “Possibilistic activity recognition with uncertain observations to support medication adherence in an assisted ambient living setting,” *Knowledge-Based Systems*, vol. 133, pp. 156–173, 2017.
- [13] M. Yu, A. Rhuma, S. M. Naqvi, L. Wang and J. Chambers, “A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 6, pp. 1274–1286, 2012.
- [14] A. N. Aicha, G. Englebienne and B. Kröse, “Unsupervised visit detection in smart homes,” *Pervasive and Mobile Computing*, vol. 34, pp. 157–167, 2017.
- [15] J. H. Lee and H. Hashimoto, “Controlling mobile robots in distributed intelligent sensor network,” *IEEE Transactions on Industrial Electronics*, vol. 50, no. 5, pp. 890–902, 2003.
- [16] J. Rodriguez-Araujo, J. J. Rodriguez-Andina, J. Farina and M. Y. Chow, “Field-programmable system-on-chip for localization of UGVs in an indoor iSpace,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1033–1043, 2013.
- [17] I. Cvitić, D. Peraković, M. Periša and B. Gupta, “Ensemble machine learning approach for classification of IoT devices in smart home,” *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 11, pp. 3179–3202, 2021.
- [18] J. Li, Y. Bian, J. Guan and L. Yang, “Construction and application of cloud computing model for reciprocal and collaborative knowledge management,” *Computers, Materials & Continua*, vol. 75, no. 1, pp. 1119–1137, 2023.
- [19] D. B. Lenat, “CYC: A large-scale investment in knowledge infrastructure,” *Communications of the ACM*, vol. 38, no. 11, pp. 33–38, 1995.
- [20] M. Tenorth and M. Beetz, “Representations for robot knowledge in the KnowRob framework,” *Artificial Intelligence*, vol. 247, pp. 151–169, 2017.

- [21] S. Lemaignan, M. Warnier, E. A. Sisbot, A. Clodic and R. Alami, "Artificial cognition for social human-robot interaction: An implementation," *Artificial Intelligence*, vol. 247, pp. 45–69, 2017.
- [22] M. Tenorth, A. C. Perzylo, R. Lafrenz and M. Beetz, "Representation and exchange of knowledge about actions, objects, and environments in the robearth framework," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 643–651, 2013.
- [23] K. J. Singh, D. S. Kapoor, K. Thakur, A. Sharma and X. Z. Gao, "Computer-vision based object detection and recognition for service robot in indoor environment," *Computers, Materials & Continua*, vol. 72, no. 1, pp. 197–213, 2022.
- [24] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, pp. 154–171, 2013.
- [25] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and Their Applications*, vol. 13, no. 4, pp. 18–28, 1998.
- [26] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, USA, pp. 1440–1448, 2015.
- [27] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 91–99, 2015.
- [28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.*, "SSD: Single shot multibox detector," in *Proc. Computer Vision–ECCV 2016: 14th European Conf.*, Amsterdam, Netherlands, pp. 21–37, 2016.
- [29] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, pp. 779–788, 2016.
- [30] A. Farhadi and J. Redmon, "Yolov3: An incremental improvement," in *Proc. Computer Vision and Pattern Recognition*, Berlin/Heidelberg, Germany, pp. 1–6, 2018.
- [31] A. Bochkovskiy, C. Y. Wang and H. Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv, Computer Vision and Pattern Recognition*, arXiv:2004.10934, 2020.
- [32] Z. Wang, Y. Zheng, X. Li, X. Jiang, Z. Yuan *et al.*, "DP-YOLOv5: Computer vision-based risk behavior detection in power grids," in *Proc. 2021 Chinese Conf. on Pattern Recognition and Computer Vision (PRCV)*, Zhuhai, China, pp. 318–328, 2021.
- [33] N. Hamid, V. Chang, R. Walters and G. Wills, "A Multi-core architecture for a hybrid information system," *Computers & Electrical Engineering*, vol. 69, pp. 852–864, 2018.
- [34] V. Pimentel and B. G. Nickerson, "Communicating and displaying real-time data with websocket," *IEEE Internet Computing*, vol. 16, no. 4, pp. 45–53, 2012.
- [35] M. Mast, M. Burmester, B. Graf, F. Weisshardt, G. Arbeiter *et al.*, "Design of the human-robot interaction for a semi-autonomous service robot to assist elderly people," in *Proc. Ambient Assisted Living: 7. AAL-Kongress 2014*, Berlin, Germany, pp. 15–29, 2015.
- [36] A. Bicchi, "Of robots, humans, bodies and intelligence: Body languages for human robot interaction," in *Proc. Tenth Annual ACM/IEEE Int. Conf. on Human-Robot Interaction*, Portland, USA, pp. 107–107, 2015.
- [37] B. A. Myers, "A brief history of human-computer interaction technology," *Interactions*, vol. 5, no. 2, pp. 44–54, 1998.
- [38] K. W. Lee, H. R. Kim, W. C. Yoon, Y. S. Yoon and D. S. Kwon, "Designing a human-robot interaction framework for home service robot," in *Proc. IEEE Int. Workshop on Robot and Human Interactive Communication, 2005*, Nashville, USA, pp. 286–293, 2005.
- [39] J. Wang and M. She, "Probabilistic latent semantic analysis for multichannel biomedical signal clustering," *IEEE Signal Processing Letters*, vol. 23, no. 12, pp. 1821–1824, 2016.
- [40] R. Fernandez-Beltran, J. M. Haut, M. E. Paoletti, J. Plaza, A. Plaza *et al.*, "Remote sensing image fusion using hierarchical multimodal probabilistic latent semantic analysis," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 12, pp. 4982–4993, 2018.
- [41] Z. Wang, G. H. Tian and X. Shao, "Home service robot task planning using semantic knowledge and probabilistic inference," *Knowledge-Based Systems*, vol. 204, pp. 106174, 2020.

- [42] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Columbus, USA, pp. 580–587, 2014.
- [43] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, vol. 42, no. 1–2, pp. 177–196, 2001.