Peer-reviewed version available at Appl. Sci. 2019, 9, 4555; doi:10.3390/app9214555

Article

A Function-as-a-Service Based Fog Robotic System for Cognitive Robots

Hyunsik Ahn [1]*

[1] Department of Robot System Engineering, Tongmyong University; hsahn0@gmail.com
* Correspondence: hsahn0@gmail.com; Tel.: +82-10-2518-4777

Received: date; Accepted: date; Published: date

Featured Application: Robot Cloud and Cognitive Robots.

Abstract: Cloud robotics is becoming an alternative to support advanced services of robots with low computing power as network technology advances. Recently, fog robotics has gained attention since the approach has merit relieving latency and security issues over the conventional cloud robotics. In this paper, a Function-as-a-Service based Fog Robotic (FaaS-FR) for cognitive robots is proposed. The model distributes the cognitive functions according to the computational power, latency and security with a public robot cloud and fog robot server. During the experiment with a Raspberry Pi as an edge, the proposed FaaS-FR model shows efficient and practical performance in the proper distribution of the computational work of the cognitive system.

Keywords: Robot Cloud; Cognition as a Service; cognitive robots; sentential cognitive system; cloud service; human–robot interaction

1. Introduction

Cloud computing is a widely advancing information and communications technology service and is a key technology of the advanced industry. Robot clouds applying cloud computing to robots allows robots to connect to a cloud environment, uses a huge computational infrastructure, and obtains results of high level programs from the cloud [3]. The cloud robots share information including environments, actions, and objects and offload heavy computation to a cloud server.

However, such cloud robot services could give rise to security issues of privacy breaches and latency issues of control signals delays for robot motions. Recently, to solve these issues, fog robotics, distributing computing work properly with fog servers and edges, is getting attention as it has the advantages of reducing latency and security matters (Figure 1) [18,19,25].

These merits of fog robotics can accord with cognitive robots to reduce the cost of the robot and its Human-Robot Interaction (HRI) services. If the cognitive robot adopts fog robotics model offloading burdened computing tasks to clouds or fog servers, it also needs to consider privacy, security and latency as well as abundant computing power for advanced intelligent functions. Especially, the cognitive robots can represent experienced cognitive information, store it in a proper form, and retrieve it using a reasoning procedure. It means that the fog robotics model of cognitive robots need to consider the characteristics of the cognitive structure.

In this paper, a Function-as-a-Service based Fog Robotics (FaaS-FR) for the Sentential Cognitive System (SCS) of cognitive robots is proposed. FaaS-FR model includes the edge as the local robot system, the fog robot servers for the private, security and computing power, and the robot clouds for the high performance computation. The previous SCS consists of multiple modules to recognize new events that the robot has experienced and describes them in a sentential form to be stored in a sentential memory and retrieved with a reasoning process in the future [24]. In this approach, the SCS adopts FaaS-FR, and each module of the SCS is classified with the functionality of privacy, security, and latency as well as required computing power. According to the functionality, the computation of modules is executed on the edge or offloaded to fog robot servers or robot clouds [7].

The merit of FaaS-FR is that advanced services utilizing high performance computation is possible even in the edge system of low cost and low computing capability. A module in the SCS acquires and transfers raw data to the fog sever or a cloud. Then, the server processes the data and sends back the results to the SCS. In the implementation and test, we can observe that the FaaS-FR can make cognitive robots more efficient via proper distribution of the computing power and information sharing.

The contributions of this study are as follows: (1) a fog robotics model, FaaS-FR model, is suggested to be applied to a cognitive robot for efficient and advance services at a low cost. (2) with this model, a functionality based modular networking in SCS of a robot is proposed and tested.

This paper is organized as follows. In section 2, related work on robot cloud and fog robotics are described. Section 3 details the theoretical background of the proposed FaaS-FR model. Section 4 describes an application of the model to a SCS for a cognitive robot platform. Section 5 provides the implementation of the proposed approach to a service robot through experimental results. Finally conclusions and future work are presented in section 6.

Figure 1. Fog robotics schematic [27]

2. Related Work

Cloud robot services applying the technology of cloud computing to robots utilize computation, storage, and communication in internet infrastructure. For instance, RoboEarth, a pioneering robot cloud service, supports the service of storing software components, mapping information, behavior information, and object information in a database (DB) and utilizing a cloud engine for services; robots can use it by virtualizing the information [9].

However, there have been different opinions regarding the usefulness of robot clouds. An advocative side, on one hand, agrees that a robot can enhance its capability by combining it with robot cloud services. Kuffner insists that the robot cloud, for a robot system having limited computing power, can carry over burdened tasks to cloud servers [10]. On the other hand, Laumond noted that the approach of robot clouds could result in harmful effects to the performance of robots [11]. He insisted that the rapid advancement of real-time capability of on-board processing could make the concept of cloud robotics meaningless, and the cloud robot depending on the network could weaken the autonomy and reliability of robots.

Nevertheless, a robot cloud has characteristics of cloud computing and robot technology and the unique characteristics of the robot cloud itself. Particularly at the robotics level, robots in the robot cloud can communicate with each other to share the burden of tasks rather than serve as isolated

Peer-reviewed version available at Appl. Sci. 2019, 9, 4555; doi:10.3390/app9214555

devices that allow only data exchange with remote servers. As a result, the robot cloud will optimize the outstanding achievements that exist in robotics.

There are different levels of service in the field of cloud computing comprising Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) [12]. These models can be applied to robot clouds and multiple studies regarding them have been introduced. Mourandian et al. [13] highlighted aspects of IaaS in robotic applications. They proposed an architecture that enables cost-effectiveness by delegating virtualization and dynamic tasks to robots, including robots that could belong to other clouds. Gerardi et al. [14] proposed a PaaS approach to a configurable suite of products based on cloud robotics applications. It allowed end users to be free from the low-level decisions needed to construct architectures of complex systems distributed among robots and clouds. For example, if the REALabs platform was built using a PaaS model, it was predicted that many robot applications could be developed in this area [15]. In the framework of SaaS, a robot used a remote server for the training of locations with a neural network [16]. This case was used to establish communication between the cloud and the robot through a wide environment and to identify its location in images transmitted by the robot at the SaaS level. Chen and Hu [17] described Robot-as-a-Service (RaaS) with the Internet of Things. The RaaS was able to create a local pool of intelligent devices using autonomous and intelligent robots and make local decisions without communicating with the cloud.

However, networking robots has given rise to security and latency issues. DeMarinis et. al [26] checked a number of robots can be accessible and controllable from the Internet and dangerous to both the robot and the human, and the robot's sensors can be viewed to be a threat to privacy. Chinchali et. al [19] issued that cloud robotics comes with a key communicating with the cloud over congested wireless networks may result in latency or loss of data, and formulated offloading as a sequential decision making problem, and proposed a solution using deep reinforcement learning.

To overcome security and latency matters of cloud robotics, fog robotics, a new network model, was introduced [27]. The model, as shown in Figure 1, distributes storage, computer and networking resources between the cloud and the edge in a federate manner. Gudi et. al [25] first suggested fog robotics that fog robot servers gather the cloud information and server it to the private robots. Tanwani et. al [18] introduced a fog robotics approach for secure and distributed deep robot learning. It provides that deep learning models are trained on public synthetic images in the cloud, the private real images are adapted at the edge within a trusted network.

In summary, the robot cloud systems aim to improve robot performance with the aid of clouds enhancing computing power and data management capability. However, the robot cloud structure is challenged by the issues of real-time control, synchronizations, and stability risks. Alternatively, the fog robotics aims to improve the performance of robots by distributing computation burden with cloud, fog, and edge computing works, and at same time enhancing security and latency. In the next section, an advanced fog robotics model, the FaaS-FR model, distributing computing tasks according to the cognitive functions based on SCS is detailed.

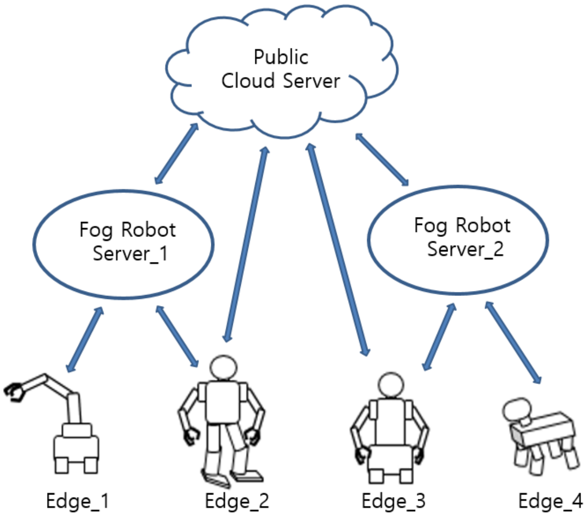


Figure 2. The schematic of FaaS-FR

3. FaaS -FR model for cognitive robots

In the cloud robot paradigm, there are various deployment models, including private, community, public, and hybrid clouds. Wang et al. introduced these models into the robot clouds [6]. This study proposes and evaluates various connection methods that occur in the implementation of cloud robots. Public cloud models exchange large amounts of data and information across networks and clouds. The cloud can be used to share data in a computing environment that everyone can share. However, the exclusive materials of an individual should be served in a proper separate manner. In a personal robot cloud, a server or cloud is privately connected to a home or company. Personal robot clouds can form an external and independent cloud and distribute the robot's computing power through the servers.

In the view of fog robotics, the function of personal cloud servers can be matched with fog robot servers which personally support edges (local robot system) [25]. Therefore, we adopt the term of “fog robotic server” than “cloud robotic server” because the server works not for other’s robots but for specific robots privately.

However, the fog robotics models generally have a hierarchical model consisting of clouds, fog servers, and edges. With these models, it is difficult for edges directly to access to a cloud and get the result of services which have specific functionality with enough computing power.

In this paper, to overcome these matters, an advanced model, FaaS-FR model, is proposed as shown in Figure 2. In the model, all the functional modules of cognitive robot are classified according to privacy, security and computing power, and have their own networking with concept of fog robotics. The functions of the robot are suitably divided for being worked on edges, fog robot servers, and public robot clouds respectively. In the case of information possibly being a violation of privacy, it is computed and stored in an edge or a fog robot server. If the edge need to uses the public robot cloud to utilize an advanced computing service, it can access the cloud directly to reduce latency or go through a fog robot server to the cloud. The reason that a new term is coined, FaaS, is that the classified functions of robot can be offloaded on the cloud or fog servers according to the security, latency, privacy, as well as computing power

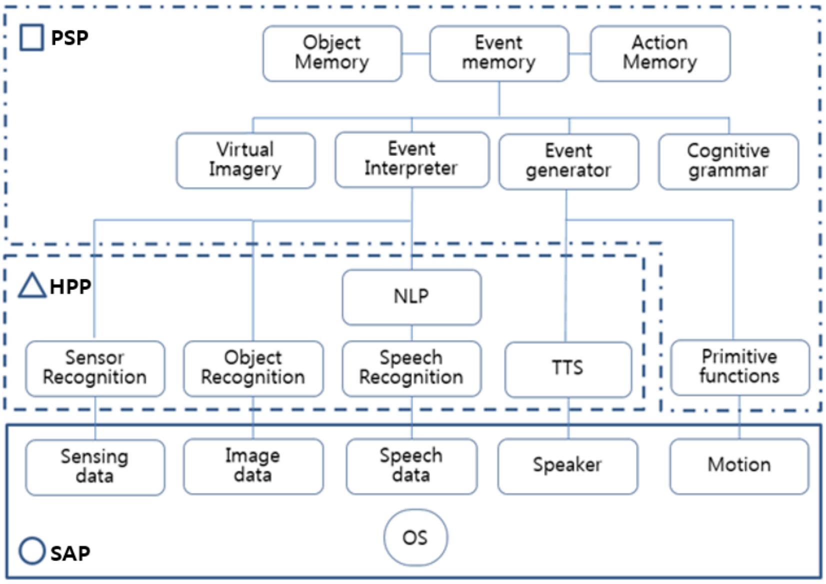


Figure 3. A schema of cognitive robot functionality

Figure 3 shows a schema of modular cognitive functions of a general structure of cognitive robots. It has perception modules comprising sensing, object recognition, and speech recognition. Therefore, the robot can talk with humans regarding the visual situation that the robot recognizes. The robot also has behavior modules such as utterance and motion. In the higher part, there are interpretation and generation modules which can produce descriptive cognitive information from the perceptional. Memory modules are used to store the descripted cognitive information, and can be retrieved for the future by a reasoning procedure with virtual imager and cognitive grammar. In the view of functionality including privacy, security, safety, latency, and computing power, the functions can be divided with 3 parts; Sensing and Actuation Part (SAP), Privacy and Security Part (PSP), and High Performance Part (HPP).

The SAP which covered by solid line and marked with a circle, is the essential part including OS, sensing and actuation which are indispensable for the robot. This part should have OS, perception functions including sensors, cameras and microphones, and behavior functions including speakers and actuators. These functions are dependent on the hardware of the robot and cannot be taken over by others.

The PSP covered with dashed single-dotted line and marked with a rectangle should be installed on edges or fog servers. For low cost robot services with minimum computing power and network infrastructure, this part needs to be offloaded to clouds; however, these functions can be related to private and security information. Therefore, it is reasonable that this part is working on the edges or a fog servers. In the case that the private information is not serious and well secured, it could be offloaded to a public cloud.

The HPP which has dashed line with a triangle marked is dependent on public robot cloud tools that can supply high-quality performance such as speech recognition, Natural Language Processing (NLP), 2D and 3D object recognition, and Text-To-Speech (TTS). The Google Cloud Application Programming Interface (API) supports multiple modules of deep learning in its public cloud [8].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 October 2019 doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at *Appl. Sci.* **2019**, *9*, 4555; doi:10.3390/app9214555

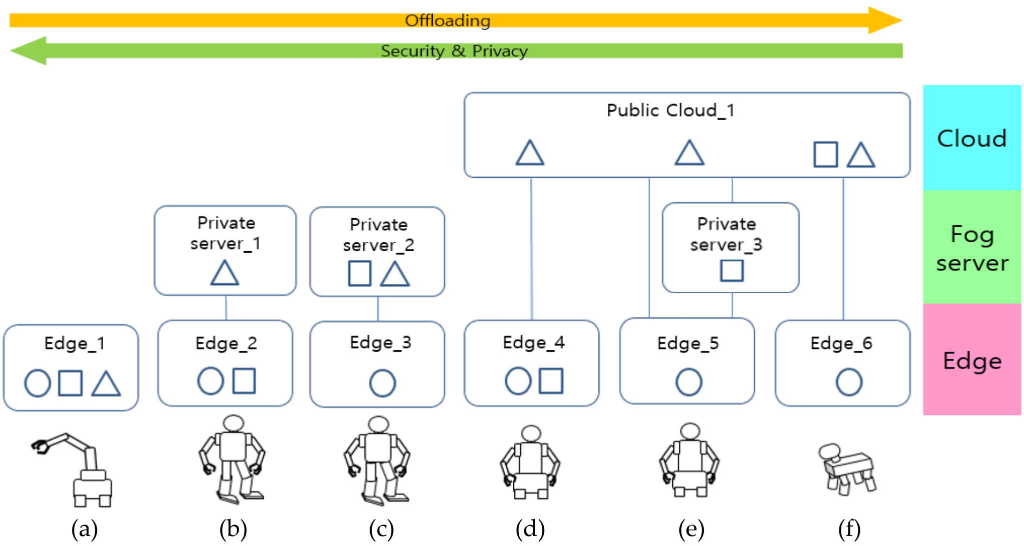


Figure 4. Various distribution of robot functions and offloading levels
(a) Stand-alone type, (b) and (c) are using private servers for offloading computation burden, (d), (e) and (f) adopt clouds. The proposed FaaS-FR adopts model (e).

Figure 4 shows various distribution of functions and offloading levels. As shown in the top, according from the left to right side, the functions of the robot is dependent of the servers and clouds. On the contrary, the security and privacy could be weaken with the direction.

The stand-alone type (a) is a conventional type of robot computing in which the edge covers all parts (SAP, PSP and HPP). (b) and (c) both use private servers for the functional parts. Edge_3 of (c) offloads the PSP to the private server, then the edge can work with lower computing power. In these cases, the HPP and PSP should be developed and installed on the server by developers for the functions of the robots.

In the case of (d) and (f), the conventional robot cloud models, all the high performance computations is offloaded on the cloud. In the case of (d), the edge has computing power to cover PSP. However, the model of (f) even transfers the PSP to the cloud to reduce the computational load of the edge. We will be able to see this case for mass robot service providers. The public cloud should safely manage private information for the robot

The model of (e) shows the typical characteristics of fog robotics. It provides flexibility in adopting applications among the edge, the server, and the cloud. Specifically, Private Server_3 can be a fog robot server to do the job of PSP. This is applicable to achieve high quality functions with low computing power by offloading them to clouds. In this case, there are two kinds of services: The first method is that the edge directly access the cloud for the HPP service. The other method is that the fog robot server mediates for receiving the service from the cloud and additive processing, and then transfer it to the edge. In this paper, the model (e) is adopted as FaaS-FR.

Table 1. The characteristics computing of FaaS-FR

Computing type	Functional parts	Service levels
Public robot cloud	HPP	PaaS, SaaS
Fog robot server	PSP	PaaS, SaaS
Edge	OS, SAP	-

The FaaS-FR model can be applied according to the specified functionality and the service level. Table 1 shows the level of FaaS in the fog robotics model. In the cloud, the function is offloading high performance computation. In the case of fog robot server, it is used to privacy and security as well computing power distribution. Edges, shallow computing systems, covers OS and elementary data acquisition and actuation. For the clouds and fog servers, both PaaS and SaaS can be adopted

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 October 2019 doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at *Appl. Sci.* **2019**, *9*, 4555; doi:10.3390/app9214555

according to the functionality and the computing power of the edge. In the PaaS case, the user should develop an application using APIs supported by the PaaS [22]. On the contrary, SaaS supports applications without any application development.

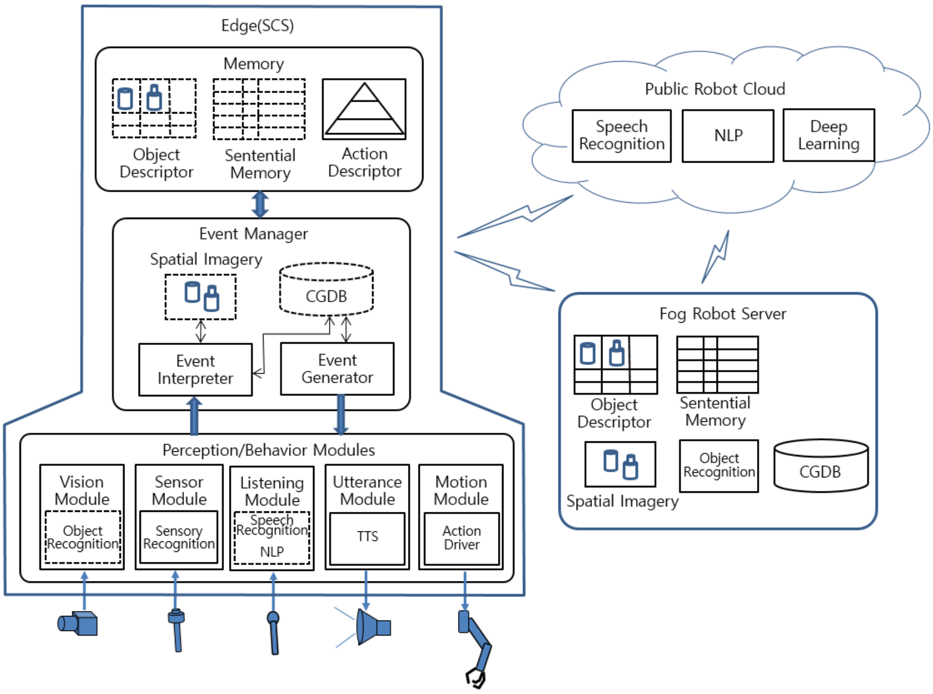


Figure 5. The block diagram of the SCS based on FaaS-FR that offloads the computation of modules.

Table 2. Examples of sentences stored in sentential memory

#	Time	Module	Sentence					
			VERB	AUX	ARG1	ARG2	SPACE	TIME
1	t_1	VM	appeared	-	(NP A cup)	-	(PP at (NP *x*, *y*, *z*))	
2	t_2	LM	move	-	-	(NP the cup)	(PP to (NP the front) (PP of (NP the bottle)))	-
3	t_3	MM	picked up	-	(NP I)	(NP the cup)	-	-

VM: Vision module, LM: Listening module, MM: Motion module

4. A SCS model based on FaaS-FR

Figure 5 shows that the proposed FaaS-FR model which is applied to the SCS of a service robot. The functions are categorized according to cognitive functions for allocation according to functionality. The modular functions with dotted line are offloaded to a public robot cloud or a fog robot server. The applications for the functions use APIs of PaaS or SaaS of the cloud and the fog server.

In the memory of the SCS, the sentential memory stores a series of sentences describing cognitive information of events as shown in Table 2 [24]. When an event occurs in a module, the system converts the cognitive information of the event to a sentential form and stores it in a sentential memory. Each sentence has a modular and time tags for being used to query the memory for reasoning. SCS uses an object descriptor to store the features of objects, such as labels, shapes, and current poses, for expressing visual events. The motion descriptor stores the information of physical actions of the robot hierarchically [7]. Each module of the memory is related to the privacy and security and indispensable for the essential functions of robots. In the view of FaaS-FR, the memory modules can be worked on the edge when the computing power is enough. If the computing power of the edge is limited, the task of the module can be moved to the fog robot server

 doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at Appl. Sci. 2019, 9, 4555; doi:10.3390/app9214555

The event manager controls the interpretation and reasoning of events. The event interpreter interprets the cognitive information obtained from the modules and creates sentences. The event manager stores the sentences in the sentential memory. Schematic imagery is an imitation of a human mental model for spatial reasoning. If the SCS needs a reasoning of the visual situation at a certain time, it produces a virtual scene by placing the models of the objects and derives sentences that express the spatial context of the scene. The cognitive grammar DB (CGDB) has grammar rules to generate sentences from the cognitive information of the events. For the purpose of FaaS-FR, the function of the event manager is essential for the robots to work properly. Therefore, the modules can be work on the edge or in the robot fog servers.

There are perception and behavior modules linked to and from the external world in the lower part of the Figure 5. The vision module is used to recognize visual events by capturing scenes by a camera and recognizing objects. The sensor module includes all sensing functions of the robot acquired by data acquisitions that include physical contacts, sound, and temperature. The listening module captures human speeches and transfers it to the cloud to use a speech recognition application to get sentences. Then, it analyzes the acquired sentences via an NLP including syntactic and semantic parsing. The utterance module generates sentences using a sentence generator and utters them with TTS application.

The action module controls the motion of the robot. A physical emergency situation could be happened and therefore the motion must be managed and controlled in the edge to keep security and privacy. This approach adopts a hierarchical motion model to provide effective handling of objects by using predefined primitive actions (Table 3). It comprises three levels: episodes, primitive actions, and atomic functions [4]. Episodes could be human commands asking the robot to perform a task via a series of primitive actions. The primitive action calls the predefined atomic functions with the atomic functions in the motion descriptor of an SCS, and physically performs them in the motion module. For example, as shown in Table 3, if a user orders "bring o_i to p_{oi}," it can consist of a series of primitive actions: "identify o_i," "pick up o_i," "move the hand to p_{oi}," and "place o_i." A primitive action, such as "pick up o_i," calls the atomic functions: extend (o_i), grasp (o_i), and retract (). The motion descriptor of the SCS stores the elements of each level of the hierarchical model, sustains their linkage, and physically responds to the human speech commands.

Table 3. An example of action events with the action descriptor.

Episodes	Primitive actions	Atomic functions
bring o_i to p_{oi}	identify o_i	(search o_i in the object descriptor)
	pick up o_i	extend(o_i), grasp(o_i), retract()
	move the hand to p_{oi}	move_hand_to(p_{oi})
	place o_i	open_hand(o_i), retract()

 doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at *Appl. Sci.* **2019**, *9*, 4555; doi:10.3390/app9214555

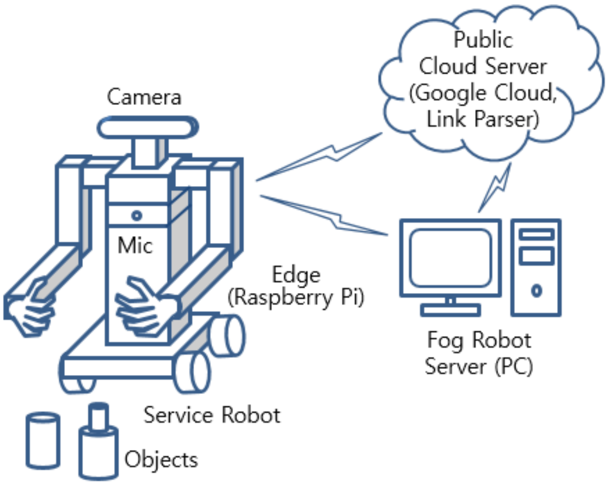


Figure 6. The schematic of the FaaS-FR service implementation with an edge of low computing power.

5. Implementation and experimental results

In this paper, the proposed FaaS-FR model was implemented in a mobile robot and tested object recognition, speech recognition, and object handling motion. Figure 6 shows the schematic of implemented FaaS-FR. The functions of the robot service were distributed with SAP, PSP, and HPP which worked for their own tasks. Figure 7 shows a two handed mobile robot as a testbed of the edge. The system of the edge was Raspberry Pi 3 using Linux OS (Ubuntu) that has low computing power, and a desktop computer with Windows 10 was used as the fog robot server. Table 4 shows the system specifications of the edge and the fog robot server. In the vision module, there is an Xtion sensor made by ASUS for acquiring color and depth (RGB-D) image. For the listening model, it had a microphone on the edge system and captured the human speech, and transferred it to Google Cloud to get the recognized text data. The acquired text data was transferred to a parsing cloud, Link parser server, and got the parsed results.

The test scenario of the FaaS-FR was that a user asks a speech order to the robot to move an object and place it a specific position. For the execution of the order, the robot used the listening module for understanding the human speech, the vision module for 3D object recognition, and the motion module to bring the object. The FaaS-FR based SCS distributed tasks on the edge, a robot fog server, and Clouds. Table 5 shows the functions and fog computing types.

For the listening module, the speech recognition was executed with the cloud (Google Cloud), but the NLP was done on the fog server (Link parser server). For the speech recognition, the edge first acquired a human speech and transfer it to the Google Cloud to get the text of the speech. The speech recognition application utilized Google Cloud speech API as a PaaS. And the result of speech recognition was transferred to the fog robot server to recognize the meaning of the sentence with syntactic and semantic parsing. The event interpreter requested a motion to execute the order of the human.

In the case of vision module, if the scene is changed, the module transfers the capture RGB-D data to the fog server, then the server runs an object recognition application. In this paper, for the object recognition, You Only Look Once (Yolo), a Convolution Neural Network (CNN), was adopted [20]. It produced Bounding Boxes (BBX) and labels of the objects in the RGB image. The trained weight files were brought from a cloud (Yolo server), but the object recognition was done on the fog robot server as a SaaS. The depth data were converted to XYZ coordinates for being used for 3D segmentation and extracting the real coordinates of the objects to be handled by the robot.

 doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at Appl. Sci. 2019, 9, 4555; doi:10.3390/app9214555

(a) (b)

Figure 7. A test bed of FaaS-FR using Raspberry Pi.
(a) The service robot, (b) a Raspberry Pi and perception and behavior modules

Table 4. The system specification of an edge and a fog robot server.

Specification	Edge	Fog robot server
System	Raspberry Pi 3 B	Desktop
CPU	Quad Cortex A54	Intel® Core™ i7-770
Clock	1.2GHz	3.6GHz
RAM	1GB SDRAM	8GB RAM
Network	Wifi	LAN
Bandwidth	14.4 Mbit/sec	94.5 Mbit/sec

Table 5. The functions of the robot and service types.

Modules	Tasks	Servers	Fog computation types	Functionality
Listening module	Speech recognition	Google Cloud	Cloud-Fog-Edge	High performance computation
	Semantic parsing	Link parser	Fog-Edge	Computing power
Vision module	Object recognition	Yolo	Fog-Edge	Computing power
Motion module	Object handling	Motion application	Edge	Security & privacy

Figure 8 shows the results of vision module with a fog server. When the module of SCS of the robot captures the RGB-D data by using an Xtion sensor, the SCS executes the vision processing in the fog cloud by transferring the acquired data. The fog server receives the data and processes an object recognition algorithm that needs a relatively higher computing power and then transfers the results of the processing to the cognitive system. Figure 8 (a) and (b) show the RGB and depth data of the Xtion sensor. Figure 8 (c) shows the result of object recognition using Yolo. The vision module had the result of BBX and label of the object with offloaded on the cloud fog server. Figure 8 (d) shows the 3D view of scene using OpenGL library to get x, y, z coordinates of the cloud points from the acquired RGB-D data, which were processed in the fog robot server. Figure 8 (e) shows the result of 3D object recognition obtained by using 3D segmentation on the BBX area of the object. It provided x, y, z coordinates of the object to be used for the handling of the robot hand

For the action to handle objects, the action module analyzed the meaning of the ordered sentence. The argument of the sentence was linked with the cup in the object descriptor and search the position and pose of the object. Figure 9 shows the motion executing an episode, "bring the cup to the front of the bottle." The order was an episode and divided with primitive actions: (a) "identify the cup,"

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 14 October 2019 doi:10.20944/preprints201910.0155.v1

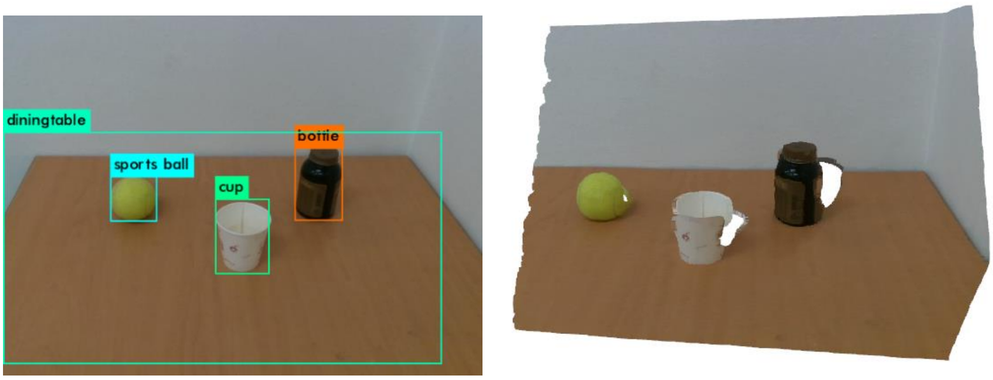
Peer-reviewed version available at Appl. Sci. 2019, 9, 4555; doi:10.3390/app9214555

(b) “pick up the cup,” (c) “move the hand to the front of the bottle,” and (d) “place the cup.” These primitive actions are executed with atomic functions.

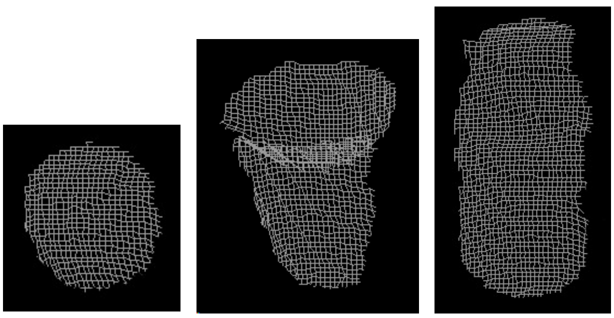
In this paper, FaaS-FR model was tested by comparing the two fog computing types. Table 6 shows two sentences and their speech signals for testing FaaS-FR. Two sentences, “bring the cup” and “bring the cup to the front of the bottle” were tested with the Link parser cloud in textual syntactic parsing and Google Cloud for speech recognition. Figure 10 shows the average computing times with two types of FaaS-FR models to utilize the Google Cloud for speech recognition. The graph shows that the Cloud-Fog-Edge type is better in the response time. Figure 11 shows Link parser response time. We can see Fog-Edge type is best to reduce the computation time.

From the results of two cases, the response time of the services are not proportional to the size of the data. The syntactic parsing shows that the response time performance between Cloud-Fog-Edge and Cloud-Edge is largely different, but the speech recognition produces relatively small difference. It could be related to the bandwidth, computing time, and overhead of files. Therefore, when one selects a fog computation type, a previous performance test is needed.

(a) (b)

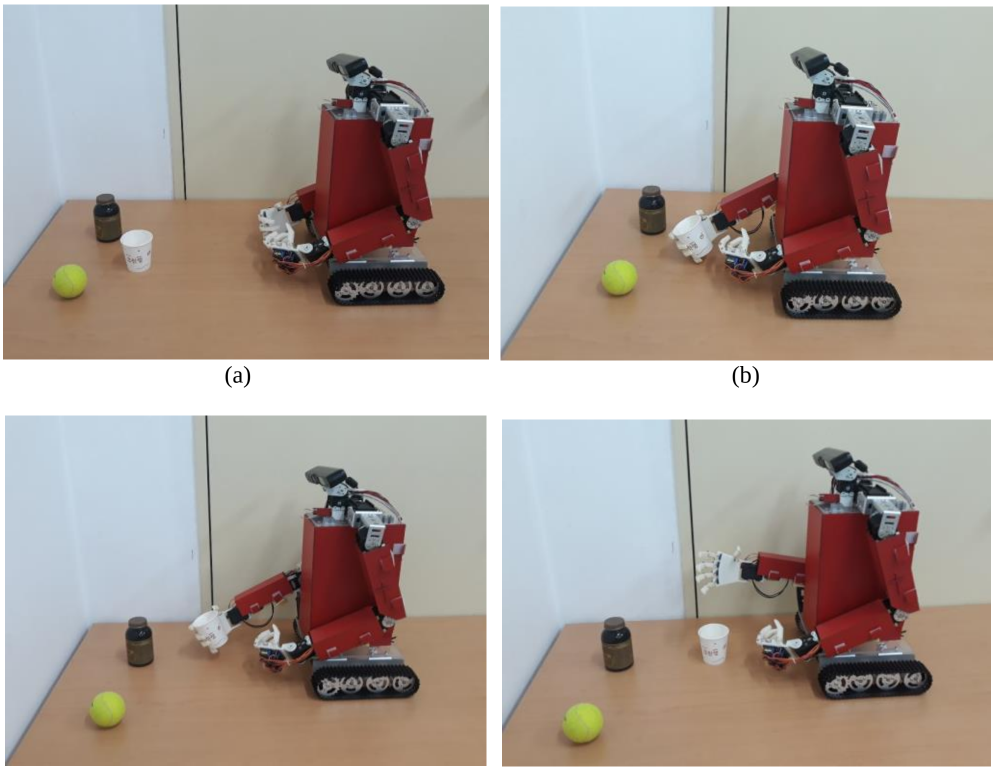


(c) (d)



(e)

Figure 8. The result of object recognition with the fog robot service: (a) an RGB image, (b) a depth image, (c) the result of object recogntin with BBX and label, (d) a 3D view of scene, (e) the result of 3D object recognition after 3D segmenatation on the BBX areas.



(a) (b)

(c) (d)

Figure 9. The motion of the service robot excuting "bring the cup to the front of the bottle": (a) identify the cup, (b) pick up the cup, (c) move the hand to the front of the bottle, (d) place the cup.

Table 6. The sentences and speech signals for testing FaaS-FR.

	Sentence_1	Sentence_2
Text	"bring the cup"	"bring the cup to the front of the bottle"
Time (sec)	1.55	2.93
File size (Kbyte)	135	180

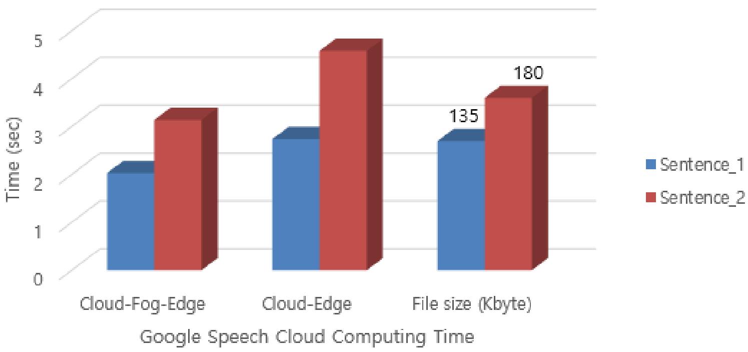


Figure 10. Google speech cloud computing time according to the fog service model

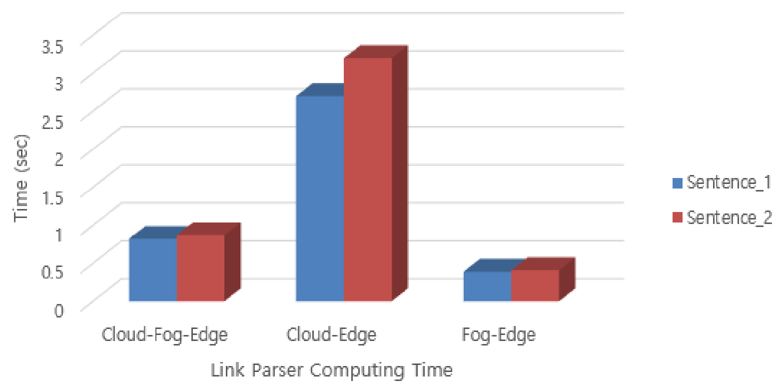


Figure 11. Link parser computing time according to the fog service model

6. Conclusion

In this paper, a FaaS-FR model for cognitive robots is proposed. The functions of cognitive system are categorized as SAP, PSP and HPP according to functionality of security, privacy, high performance computation, and needed computing power. The modular functions of SCS of the robot are divided into classes apt to be proper to edges, fog robot servers, and public robot clouds. FaaS-FR was implemented on Raspberry Pi as an edge, and PCs as a fog robot server, and Google Cloud and Link parser server as robot clouds. From the test of objects handling, the edge system of the robot worked successfully even it had a low cost Raspberry Pi in speech recognition, 3D object recognition, and object handling motion. The test showed that the robot can work more efficiently even in the cases of low specification edges by properly selecting the computation types. The proposed FaaS-FR model can be an alternative selection for low cost but high performance service robots. In the future, the issue of an autonomous selecting of fog computation types needs to be studied to produce the best performance even low cost edges of cognitive robots.

Funding: This research was funded by Tongmyong University Research Grants 2016.

Acknowledgments: This research was supported by Tongmyong University Research Grants 2016 (2016A017).

Conflicts of Interest: The author declares no conflict of interest..

References

1. Roy D., Semiotic Schemas: A Framework for Grounding Language in Action and Perception, *Artificial Intelligence* **2005**, 167, 170–205.
2. Coradeschi, S.; Saffiotti, A., An introduction to the anchoring problem, *Robotics and Autonomous Systems* **2003**, 43, 85–96.
3. RoboEarth. Available online: http://roboearth.ethz.ch/ Accessed: 2019-09-01.
4. Ahn, H., A CaaS Model Based on Cloud/IoT Service for Cognitive Robots, *ICGHIT* **2016**. 1, 106–107.
5. ROS. Available online: https://www.ros.org/ Accessed: 2019-09-01.
6. Wang, X. V.; Wang, L.; Mohammed, A. A.; Givehchi, M., Ubiquitous manufacturing system based on Cloud: A robotics application, *Robotics and Computer-Integrated Manufacturing* **2017**, 45, 116–125.
7. Ahn, H.; Ko, H., Natural-Language-Based Robot Action Control Using a Hierarchical Behavior Model, *IEIE Transactions on Smart Processing & Computing* **2012**, 1(3), 192–200.
8. Google Cloud. Available online: https://cloud.google.com/ Accessed: 2019-09-01.
9. Ansari, F. Q.; Pal, J. K. ; Shukla, J.; Nandi, G. C.; Chakraborty, P. ; A Cloud Based Robot localization technique, *Contemporary Computing* **2012**, 347–357.
10. Kuffner, J., Cloud enabled humanoid robots, *Humanoids 2010 Workshop Talks*, https://www.scribd.com/doc/47486324/Cloud-Enabled-Robots. Accessed: 2019-09-01.
11. Guizzo, E., Robots with their heads in the clouds, *IEEE Spectr*um **2011**, 48(3) 16-18.
12. Mell, P.; Grance, T., The NIST Definition of Cloud Computing, NIST Special Publication, Vol. 800, 2011, p. 7.

doi:10.20944/preprints201910.0155.v1

Peer-reviewed version available at *Appl. Sci.* **2019**, *9*, 4555; doi:10.3390/app9214555

13. Mouradian, C.; Errounda, F. Z.; Belqasmi, F.; Glitho, R., An infrastructure for robotic applications as cloud computing services, WF-IoT, IEEE, 2014, pp. 377–382.
14. Gherardi, L.; Hunziker, D.; Mohanarajah, G., A software product line approach for configuring cloud robotics applications, Proceedings of the 7th International Conference on Cloud Computing, IEEE, 2014, pp. 745–752.
15. Guizzo, E.; Deyle, T., Robotics trends for 2012, *IEEE Robotics and Automation Magazine*, **2012**, Vol. 19, 119–123.
16. DeMarinis, N.; Tellex, S.; Kemerlis, V.; Konidaris, G.; Fonseca, R., Scanning the internet for ROS: A view of security in robotics research, *CoRR* **2018**, vol. abs/1808.03322.
17. Chen, Y.; Hu, H.; Internet of intelligent things and robot as a service, *Simulation and Modelling Practice and Theory* **2013**, 34, 159–171.
18. A. K. Tanwani, N. Mor, J. Kubiatowicz, J. E. Gonzalez, K. Goldberg, A Fog Robotics Approach to Deep Robot Learning: Application to Object Recognition and Grasp Planning in Surface Decluttering 2019
19. Chinchali, S.; Sharma, A.; Harrison, J.; Elhafs, A.; Kang, D. ; Pergament, E.; Cidon, E.; Katti, S.; Pavone, M., Network Offloading Policies for Cloud Robotics: a Learning-based Approach, Robotics: Science and Systems 2019, Freiburg im Breisgau, June 22-26, 2019.
20. Redmon, J.; Farhadi, A., YOLO9000: Better, Faster, Stronger, CVPR 2017, https://pjreddie.com/publications/. Accessed: 2019-09-01.
21. Verma, H., Cloud Computing: Levels (IaaS, PaaS, SaaS) and Deployment models (Public, Private, Hybrid), KnowledgeBlob, http://www.hostingadvice.com/how-to/iaas-vs-paas-vs-saas/. Accessed: 2019-09-01.
22. Stamey L., IaaS vs. PaaS vs. SaaS Cloud Models, http://www.hostingadvice.com/how-to/iaas-vs-paas-vs-saas/, 2017**.** Accessed: 2019-09-01.
23. Wang, X. V.; Wang, L.; Mohammed, A.; Givehchi, M., Ubiquitous manufacturing system based on Cloud: A robotics application, *Robotics and Computer-Integrated Manufacturing* **2017**, 45, 116–125.
24. Ahn, H., A sentential cognitive system of robots for conversational human–robot interaction, *Journal of Intelligent & Fuzzy Systems* **2018**. 35, 6047–6059
25. Gudi, S. Ojha, B. Johnston, J. Clark, and M.-A. Williams, Fog robotics: An introduction, in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2017.
26. N. DeMarinis, S. Tellex, V. Kemerlis, G. Konidaris, and R. Fonseca, Scanning the internet for ROS: A view of security in robotics research, CoRR, vol. abs/1808.03322, 2018.
27. Fog Robotics, https://sites.google.com/view/fogrobotics. Accessed: 2019-09-01.