The Ubiquitous Network Robot Platform: Common Platform for Continuous Daily Robotic Services

Miki Sato, Koji Kamei, Shuichi Nishio and Norihiro Hagita

***Abstract*— This paper proposes a common infrastructure for robotic services to support our daily life: the *Ubiquitous Network Robot Platform* (UNR-PF). UNR-PF enables the design and development of robotic services that interact with people throughout our daily activities in various environments by utilizing networked robot systems including robots, sensor networks, smartphones, etc. In this paper, we first describe the requirements for UNR-PF from our past research through various field studies with networked robot systems. These requirements include appropriate abstraction and classification of robotic functions, information sharing and management between different services in distributed areas, and a common management facility for the networked robot devices based on their capabilities and availabilities. On the basis of these requirements, we have implemented a prototype system and have been running several real-world case studies. These field tests showed that the proposed UNR-PF is effective for continuously supporting our daily activities across multiple areas. Recent activities toward standardizing the key elements of UNR-PF are also described.**

I. Introduction

Robotic service that support human daily activities have recently become an emerging topic in the area of robotics research [1]. Initially, researchers had been focusing on constructing stand-alone robots that could perform simple tasks by interacting with people. As a result, various devices and functions have been developed, such as navigation, human detection, and speech recognition. With the advances in such elements that form the basis of human-friendly robots, researchers have started to head toward robotic services that perform many complicated tasks such as house cleaning and route navigation. As the structure and functionality of robotic services grow large and complicated, constructing robots has come to require greater time, money, and effort. Similar to traditional software engineering, people started to seek modularity and reusability of basic functional components, which has led to research and development of common libraries and middleware [2–5]. With such support, developers can utilize existing functional modules in combination with their own software and rapidly develop working robotic services. Such modularized development has accelerated studies on robotic services and development of stand-alone robots.

However, as the population ages quickly, there is growing need for robots to support us in a much wider range of daily activities. Robots for individual household tasks such as cleaning floors or folding laundry are not sufficient. Support for the elderly and disabled in a variety of daily activities is greatly demanded, from morning when they are awaken to the evening when they go to bed, and in some cases, even after they fall asleep. However, such varied demands make it more difficult to develop robots that can provide necessary support. The robot shall accompany people to many different places, such as their homes, shopping malls, and hospitals. Also, the robot needs to assist the person in a variety of ways, such as checking health, showing routes, carrying luggage, and so on. At this stage of the maturity of robotic technology, it is still quite unlikely that a single robot can be developed that can follow us everywhere and perform services in the many different situations in our lives.

One approach to solving this problem is the concept named *Network Robot System,* or *Networked Robots* [6]. The main goal of networked robot systems was to integrate various types of "robots" so that the whole system could provide robotic services with rich functionality. Since its proposal in 2002, many research projects have been carried out accompanied with real-world field experiments, and they have successfully shown the concept to be effective [6–10]. However, network robot systems up to now cannot yet sufficiently provide seamless and continuous support in various aspects of our daily lives. Although they have had success in enriching each robotic service, such as route guidance and garbage collection, these services are still missing features such as continuity to different types of locations and continuation and consistency among different types of services. Moreover, the ability to fulfill demands of different types of users is necessary. A simple example is navigation. Some may have difficulty in walking and need to use wheelchairs. Some may be able to walk but prefer routes suited to them. As such, different kinds of "ubiquity," not just for locations but also various services and ways of performing each service on the basis of user characteristics, shall be focused on, as well as inter-service and inter-location continuity. At the same time, ease of developing and setting up such systems shall be focused on. Therefore, a new concept is now required that improves and extends network robot systems that satisfy such demands.

This paper proposes a common infrastructure, the *Ubiquitous Network Robot Platform* (UNR-PF), which extends the concept of traditional network robot systems. In the following chapters, we first review related works for enhancing the effectiveness of robotic services and developing such systems, including the past works on network robot systems. Next, we explore the requirements for providing

This work was supported in part by by the Ministry of Internal Affairs and Communications, Japan. The authors are with Advanced Telecommunications Research Institute International (ATR), Keihanna Science City, Kyoto 619-0288 Japan. (e-mail: {miki, kamei, nishio, hagita}@atr.jp)

continuous support in our daily activities and then describe the proposed system, as well as the current status of activities to standardize it. We also detail the functionalities of the UNR-PF that have been verified through several field tests and briefly describe an example. Finally, we conclude this paper with a discussion on further research topics.

II. Related Works

A. Development Environments for Robotic Functions

Several development environments have recently become available for developing the robotic functions, such as Microsoft Robotics Developer Studio, ROS [4], and RT-Middleware (RTM) [5]. Development with these common specification and development environments enforces the development of each functional module and improves reusability of and mutual connectivity among those functional modules. Specifically, functional modules developed upon open platforms such as ROS and RTM are often published on the web and available for reuse by everyone freely or commercially. Developers nowadays do not have to implement all features of their robots by themselves. They can search for required modules suitable for their development environment and then compose them for their robots. They sometimes need to improve or customize existing modules for their purposes, but the results, such as new functions or new usage, can also be shared among developers. The number of modules published and shared is growing, and the granularity of modules has spread. The developers, however, currently face difficulty in understanding the structure of different abstraction layers of modules, which prevents module utilization.

B. Multi-Robot Service

The spread of wireless networking using wireless LAN and mobile phones makes it possible to connect robots to the network without cabling problems since they physically move around and perform complicated motions. Proposed applications of robotic services, as a result, have come to address the collaborative operation among a group of robots connected via network. The Mobile-Robotic Fulfillment System proposed by Kiva Systems, for example, successfully improved the efficiency of logistics by deploying many the transportation robots in a warehouse and organizing them on the basis of their location information aggregated via network. Thus, common framework becomes required to aggregate and manage information about robots such as location information for organizing networked robots.

C. Networked Robotic Service

The concept of a network robot system was first proposed in 2002 by a study group organized by the Japanese Ministry of Internal Affairs and Communications. Later, a more general concept including a network robot system was named "Networked Robots" by the IEEE RAS technical committee in 2004 [6]. Networked robot systems extend the concept of multi-robot systems toward collaboration among different, heterogeneous types of robots. These are classified into three types of robots: visible-type, virtual-type, and unconscious-type. Visible-type robots are physically embodied agents that move and perform tasks physically. Virtual-type robots appear on screens of mobile information devices as agents that mainly aim at communicating with users. Unconscious-type robots are deployed into environments mainly for sensing and forming ambient intelligence [6, 14].

A system for information sharing is also proposed so that different types of robots can execute the same task in a way that each robot can execute. RoboEarth [11] and CoTeSys [12] also address information sharing among different types of robots. In those projects, information about robots' tasks, such as operation strategies for the robots and knowledge about the targets of their tasks, are aggregated from and accumulated into web servers so that robots can autonomously generate operation commands required for providing their services by referring to this shared information.

To share and exchange information via a network, RSNP [13] is proposed as a protocol between the robot systems and network information services, such as weather forecasts, traffic information, and news contents. This protocol enables the robotic service to refer to network information services and also enables the information service providers to announce the latest information using a number of robots connected to the network.

III. Requirements for Continuous Daily Activity Support

Networked robotic services place much importance on cooperation among plural robots, sensor environments, and databases on networks. The requirements for robot systems for such next generation networked robotic service, i.e., ubiquitous network robot system, are studied in this section.

For developing networked robotic services, flexible customizability must be provided for service developers. The development process of the services does not only depend on the pre-constructed design by service provides. While developing the services, the system should be modified for each environment and improved in accordance with the requests from service users through field experiments in the real environments. To provide the service at a different place, the system should be customized in accordance with the local environment. Also, robotic functional modules are assumed to be modified for performance improvement. If such improvements enforce modification in the whole system, they obstruct the whole development process. Therefore, the service application layer, which defines the process of the service content, shall be independent of robotic functions. Thus, the abstraction level of robotic functions shall be carefully designed to be controlled from service applications.

Networked robotic services will be provided among not only wide but also non-continuous areas, such as a user's home, shops, hospitals, restaurants, etc., to support a user's life ubiquitously. To coordinate such multi-locational services, a mechanism must be constructed for sharing information among these different areas.

As robot development advances, more and more types of robots will be created to achieve a variety of tasks. The abilities of robots will also increase rapidly across the whole robotics field, and a kind of specialized robot will proliferate. For such a variety of robots to cooperate, the types of robot abilities must be classified and managed properly.

As mentioned above, the design of a ubiquitous network robot system should address the following issues: appropriate abstraction and classification of robotic functions, information sharing and management between distributed locations, and common management platforms for robotic components with their capabilities and availabilities.

A. Multi-Robot Management

The platform is required to classify the robot abilities and allocate appropriate robots in accordance with the service content and the execution environment. By implementing such a management function on the platform, each robot's functionalities can be covered so that the robot coordination can be improved in situations where various robots are available. Also, the compatibility has to be enhanced between similar functional modules on different robots.

B. Multi-Area Management

To provide services over a wide area, it is important to improve coordination among multiple areas. To link several physical points, the platform needs a mechanism to share spatial information of each area such as map information. It is important to share not only static spatial information like map information but also dynamic location information (i.e., location of robots, user, obstruction and target object) that changes its relative and absolute location in accordance with circumstances. For the efficient multi-area coordination, this spatial and location information shall be managed on a multi-layered management structure that contains local area layers, which manage location information of each area, and global layers, which manage the relationships between the local areas.

C. User Management

To provide services suitable for service users, information related to the users has to be managed. Especially in the robotic service that does a physical task for the user, the robot system must select the proper equipment in accordance with the user attributes. The platform shall provide a mechanism for managing such user attributes commonly so that all services can refer to the user information efficiently.

D. Service Management

To execute a number of services across multiple areas, it is important to monitor the state of each area and determine the start of the services so that the service can be executed in appropriate situations. The platform requires a mechanism for managing the state of a service execution environment. Furthermore, the platform requires not only a mechanism to execute each service independently but also a mechanism to share information between each service.

IV. The Ubiquitous Network Robot Platform

To solve these problems, UNR-PF consists of three layers: the functional module layer, the service application layer, and the common platform layer. The first layer consists of reusable functional modules that belong to individual robots, and the second layer manages logic for service contents. The common platform layer should be independent of and serve common functions to the other two layers. Once such a common platform is developed and provided stably, it will separate robotic functions and services so that the whole robotic system will be able to be developed at lower cost and become more dependable in terms of reusability, scalability, and availability. This structure of the UNR-PF is shown in the dotted part of Figure 1. Service applications, which define flow of the service contents, and robots in each environment connect and register to the UNR-PF at first. Then, the interaction between the service applications and the robots is established via the UNR-PF.

As such, UNR-PF serves as middleware between services and robotic devices and is composed of two layers of platform: a local platform (LPF) and a global platform (GPF). The LPF is a platform for configuring the robotic system in a single area. The GPF is a platform for configuring the robotic system in a wider range of areas that includes a number of LPFs These platforms serve as a middle-layer between the service application layer and robotic function layer. The platform has five database functions and three management functions. The database functions consist of robot registry, operator registry, user registry, map registry, and service queue. The management functions consist of a state manager, resource manager, and message manager. Each function is detailed below.

A. Robot Registry

The system contains a registry (database) for managing attributes for robots, operators and robotic-service users. These are used for selecting appropriate robots and/or operators on the basis of each user's demands. The robot registry contains information about the robots available in each area, such as their shapes and capabilities as well as their status. From these pieces of information, UNR-PF can decide what kind of service a certain robot can perform, what kind of route the robot can navigate, and whether the robot is available now or in the near future.

B. Operator Registry

Operators for tele-operation of robots are managed in the operator registry. Such operators are often necessary to compensate for the limitations of artificial intelligence in robot AIs. In UNR-PF, operators are treated as one kind of robot. The same selection mechanism works for the operators on service application demands based on the information stored in the operator registry.

C. User Registry

The user registry holds attributes on each user who wishes to receive robotic support, as well as a history of services

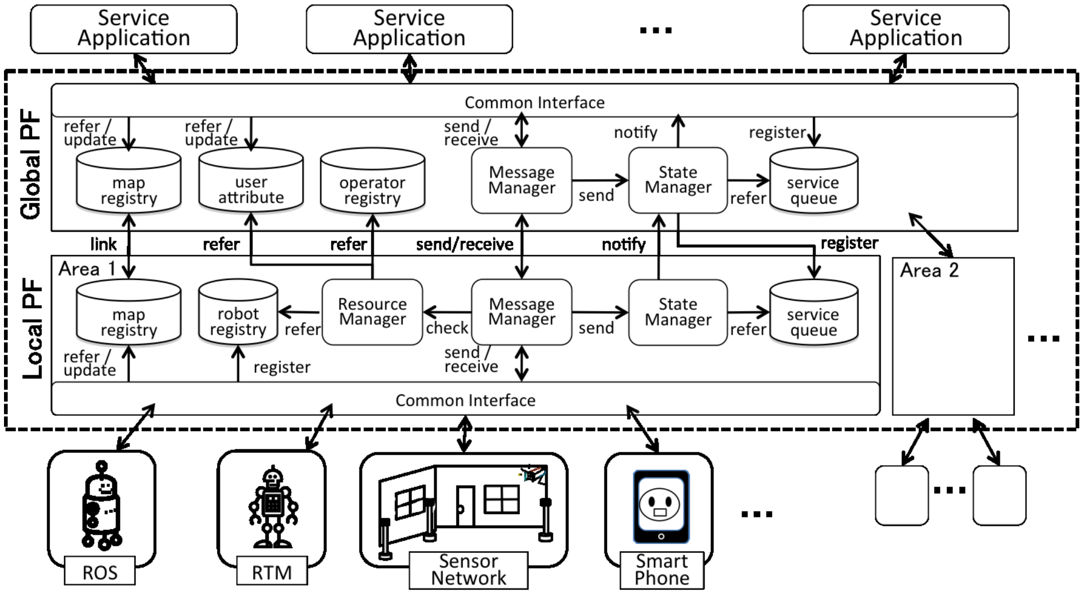

Fig. 1. Overview of the Ubiquitous Network Robot Platform

that user has used. From these pieces of information, UNR-PF can allocate the robots and operators necessary to provide certain services.

D. Map Registry

The map registries in both LPF and GPF are used to improve the linkage between different areas. The map registry of the LPF contains the map information of the service execution environment, the properties of the floor and information about movable zones, and no-go zones. The map registry of the GPF contains the positional relationship of each single area.

E. Service Queue

The service queue is a function implemented in both LPF and GPF. This function is used to manage the start of the service. This database contains IDs of the service and their initiation conditions. At first, the service application registers its ID and initiation condition to the service queue in the GPF and the state manager in the GPF registers the ID and condition to the service queue in the appropriate LPF in accordance with the state notification from the LPFs.

F. State Manager

The state manager is implemented in both LPF and GPF. This function subscribe to the message manager for the state notifications that are registered in the service queue. When receiving the state notification, the manager determines if the state complies with the start conditions in the service queue. If the state complies with the start condition, the manager send a message to the service application to start the service.

G. Resource Manager

The resource manager is implemented in LPF. When receiving the command message for executing the service application, the manager refers the robot registry, the user registry and operator registry and reserves the robot suitable for the user and the operator who can operate the service depending on the situation.

H. Message Manager

The message manager is implemented in both LPF and GPF. This function manages the messages exchanged between the service applications and the robotic functions through the common interface. The robotic functions provide the profile of available messages to the manager. When receiving the message from the service applications, the manager refers to the profiles and selects the robotic functions that fit the service application requirements. In LPF, the manager requires the resource manager to reserve the necessary resources. When receiving the message, i.e., the state notification from the robotic function, the manager checks the delivery addresses, which are given at the time of the state notification subscription, and forwards the message to the appropriate state manager and/or the service applications.

V. STANDARDIZATION

To share the information between the robots and the service applications and achieve interoperability between different robots, the specifications of data structures and interfaces must be standardized. We have focused on four key elements of UNR-PF and have been working on standardizing these elements: map information, location information, common interface, and platform architecture.

Standardization of map information is processed in Open Geospatial Consortium (OGC), a consortium for developing standards associated with the geographical information. We have requested the *CityGML* specification be extended to allow maps to contain robot-specific information. This will be reflected in the next revision, version 1.1.

Fig. 2. Scene from touring support service in shopping mall. The wheelchair robot navigates the mall on the basis of commands and environmental sensor information sent from UNR-PF.

To exchange location/pose information among various networked robot elements and robotic services, the standard specification for describing and exchanging location and pose information for robots have been issued as the *Robotic Localization Service (RLS)* specification [18].

The standardization of common interface between the service application and the robotic functional components are treated in OMG as *Robotic Interaction Service (RoIS) Framework* specification [19]. This specification was approved in June 2010 and is expected to be issued in 2012.

The architecture of UNR-PF is now under discussion at the International Telecommunication Union Telecommunication Standardization Sector (ITU-T). The recommendation, F.USN-NRP, was accepted as a standardization work item in 2011 and is expected to be released in 2012 [20].

VI. Field Experiments

Functionality of the UNR-PF was evaluated through demonstration experiments on six service cases: a remote listening support service [15], community formation service, a healthcare service [16], shopping support service, attracting customer service [17], and touring support service. This section introduces the touring service in the shopping mall (see Figure 2) as an example of networked robotic service that is provided across multiple areas. The touring support service is provided through the three areas: the user's home, the shopping mall, and the operator center. Figure 3 shows the structure and interaction of the touring service.

At first, the user makes a reservation via the virtual-type robot on the mobile device at his/her home. The virtual-type robot is connected to the LPF of the user's home and the LPF notifies the GPF of the reservation information. The touring support service application is connected to the GPF and receives the reservation. When receiving the reservation, the service application registers its service ID and the start condition, i.e., the user's arrival at the shopping mall, to the service queue on the GPF. The GPF refers to its map register to confirm the LPF of the shopping mall and then registers the service and its start condition to the LPF's service queue. When the user approaches the shopping mall, the virtual-type robot on the user's mobile device connects to the LPF of the shopping mall and notifies it of the user arrival. The LPF of the shopping mall determines that the state meets the start condition for the service and notifies the service application of the start via the GPF. Then, the service application requests the resource manager to reserve the robot in the shopping mall and the operator in the operator center via the GPF. To reserve the robot, the resource manager refers to the user registry and selects the robot suitable for the user. In the touring support service, the manager selects the wheelchair robot if the user has difficulty walking. Otherwise, it selects another movable robot. After the allocation of the resources, the service is executed in accordance with the service flow defined in the service application. The service application refers to the map registry in the LPF and instructs the robot to move around the shopping mall.

In the case of a large shopping mall, the touring support service is provided across several LPFs. When the robot comes close to the boundary between the two areas, the robot notifies GPF's state manager of the alert via LPF's state manager of the current area. Then, GPF's state manager refers to its map register to find the LPF of the next area and then registers the service and its start condition, i.e., the user's arrival at the next area or the service queue of the LPF. When the user arrives at the next area, the robot disconnects from the LPF of the first area and connects to the next LPF if the robot can work in the next area. Otherwise, another robot in the next area comes to pick the user up. Thus, the service is executed continuously. In this way, the user can receive the service seamlessly in a wide area regardless of the area segmentation for the robotic system.

VII. Conclusion and Future Works

This paper proposed a Ubiquitous Network Robot Platform, which is a common platform that enables networked robotic services over multiple areas. They are achieved using networked robot systems that consist of robots and/or sensor networks, smartphones, etc. The functionalities required to achieve the networked robot system have been studied through the discussion of issues in recent robot systems. The discussion has induced the requirement of the platform that includes the following: appropriate abstraction and classification of robotic functions, information sharing and management between distributed locations, and common management platform for the instances of robotic components with their capabilities and availabilities. The common functionalities to solve these requirements and the platform including those functionalities have been proposed. The functionalities of the UNR-PF have been verified through the field experimentation of the touring support service in a real environment. Finally, the relevant standardizations have been introduced.

To achieve a truly useful networked robot system, it is important to not only improve robotic functions and service contents but also to enhance the stability of the platform.

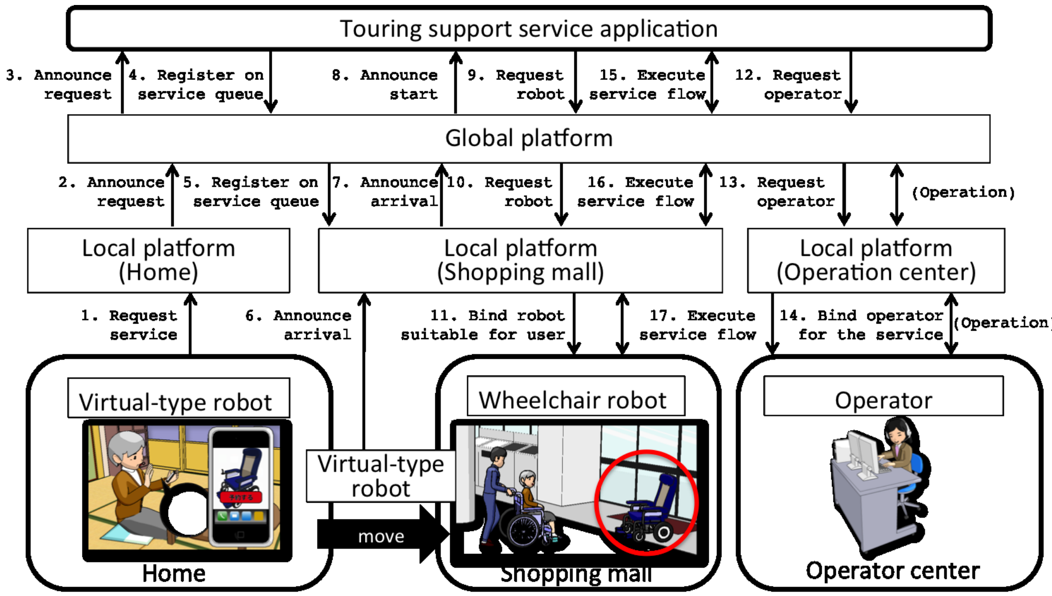

Fig. 3. Command sequence of Touring Support Service

Therefore, the UNR-PF and the development and experiment environment that uses the platform are planned to be made available to all robot developers and service providers to improve the capability and stability of the platform.

References

[1] Fong, T., et al., "A survey of socially interactive robots," Robotics and Autonomous Systems, Vol. 42, pp.143-166, 2003.
[2] Bruyninckx, H., "Open robot control software: the OROCOS project," Proc. ICRA, pp. 2523–2528, 2001.
[3] Park, H. and Han, S., "Development of an Open Software Platform for Robotics Services," Proc. ICCAS-SICE, pp. 4773–4775, 2009
[4] Quigley, M., et al., "Ros: an open-source robot operating system," ICRA Workshop on Open-Source Software, 2009.
[5] Ando, N. et al., "RT-middleware: distributed component middleware for RT (robot technology)," Proc. IROS, pp.3933–3938, 2005.
[6] Sanfeliu. A. et al., "Network robot systems," Robotics and Autonomous Systems, Vol. 56, No. 10, pp. 793-797, 2008.
[7] Shiomi, M. et al., "Field Trial of a Networked Robot at a Train Station," Int. J. Social Robotics, Vol. 3, No. 1, pp. 27-40, 2011.
[8] Sanfeliu, A. et al., "Decentralized Sensor Fusion for Ubiquitous Networking Robotics in Urban Areas," Proc. Sensors, pp. 2274-2314, 2010.
[9] Salvini, P. et al., "Do Service Robots Need a Driving Licence?," IEEE Robotics and Automation Magazine, Vol. 18, No. 2, pp. 12-13, 2011.
[10] Jung, H.-S. et al., "Unified Ubiquitous Middleware for U-City," Proc. ICCIT2007, pp. 2374-2379, 2007.
[11] Waibel, M. et al., "RoboEarth," IEEE Robotics and Automation Magazine, Vol. 18, No. 2, pp. 69-82, 2011.
[12] Tenorth, M. et al., "Web-enabled Robot," IEEE Robotics and Automation Magazine, Vol. 18, No. 2, pp.58-68, 2011.
[13] Uchiyama, T., "Robot Service Network Protocol Developed by Robot Service initiative," Proc. 5th Japan-Korea Service Robot Workshop, 2009
[14] Nakamura, Y., et al., "Framework and service allocation for network robot platform and execution of interdependent services," Robotics and Autonomous Systems, Vol. 56, No. 10, pp.831-843, 2008.
[15] Yamazoe, H. et al., "Estimation of user conversational states based on combination of user actions and feature normalization," Proc. CASEMANS, pp.33-37, 2011.
[16] Sugiyama, O. et al., "Case Study of a Multi-robot Healthcare System: Effects of Docking and Metaphor on Persuasion," Proc. ICSR, pp. 90-99, 2010.
[17] Kanda, T. et al., "A communication robot in a shopping mall, " IEEE Trans. Robotics, Vol. 26, No. 5, pp. 897-913, 2010.
[18] Nishio, S. et al., "Robotic Localization Service Standard for Ubiquitous Network Robots," in Abdellatif, H. ed., Robotics 2010 Current and Future Challenges, pp. 381-400, In-Teh:Vukovar, Croatia, 2010.
[19] Sato, M., Hori, T., Nishio, S. and Chi, S-Y, "Standardizing Framework for Robotic Services and Functions," Proc. ICRA Workshop on Robotics Modular Architecture Design and Standardization, 2011.
[20] Takashima, Y., "Requirements of network robot platform for USN applications and services: initial text of new draft recommendation", Q25/16, ITU-T, 2011.