5

Virtual Ubiquitous Robotic Space and Its Network-based Services

Kyeong-Won Jeon*,**, Yong-Moo Kwon* and Hanseok Ko**
Korea Institute of Science & Technology, Korea University***
Korea

1. Introduction

A ubiquitous robotic space (URS) refers to a special kind of environment in which robots gain enhanced perception, recognition, decision, and execution capabilities through distributed sensing and computing, thus responding intelligently to the needs of humans and current context of the space. The URS also aims to build a smart environment by developing a generic framework in which a plurality of technologies including robotics, network and communications can be integrated synergistically. The URS comprises three spaces: physical, semantic, and virtual space (Wonpil Yu, Jae-Yeong Lee, Young-Guk Ha, Minsu Jang, Joo-Chan Sohn, Yong-Moo Kwon, and Hyo-Sung Ahn Oct. 2009).

This chapter introduces the concept of virtual URS and its network-based services. The primary role of the virtual URS is to provide users with a 2D or 3D virtual model of the physical space, thereby enabling the user to investigate and interact with the physical space in an intuitive way. The motivation of virtual URS is to create new services by combining robot and VR (virtual reality) technologies together.

The chapter is composed of three parts: what is the virtual URS, how to model virtual URS and its network-based services.

The first part describes the concept of virtual URS. The virtual URS is a virtual space for intuitive human-robotic space interface, which provides geometry and texture information of the corresponding physical space. The virtual URS is the intermediate between the real robot space and human. It can represent the status of physical URS, e.g., robot position and real environment sensor position/status based on 2D/3D indoor model.

The second part describes modeling of indoor space and environment sensor for the virtual URS.
There were several researches for indoor space geometry modeling (Liu, R. Emery, D.Chakrabarti, W. Burgard and S. Thrun 2001), (Hahnel, W. Burgard, and S. Thrun (July, 2003), (Peter Biber, Henrik Andreasson, Tom Duckett, and Andreas Schilling, et al. 2004).

Here, we will introduce our simple and easy to use indoor modeling method using 2D LRF (Laser Range Finder) and camera. For supporting web service, VRML and SVG techniques are applied. In case of environment sensor modeling, XML technology is applied while coordination with the web service technologies. As an example, several sensors (temperature, light, RFID etc) of indoor are modeled and managed in the web server.

The third part describes network-based virtual URS applications: indoor surveillance and sensor-based environment monitoring. These services can be provided through internet web browser and mobile phone.

In case of indoor surveillance, the human-robot interaction service using the virtual URS is described. Especially, the mobile phone based 3D indoor model browsing and tele-operation of robot are described.

In case of sensor-responsive environment monitoring, the concept of sensor-responsive virtual URS is described. In more detail, several implementation issues on the sensor data acquisition, communication, 3D web and visualization techniques are described. A demonstration example of sensor-responsive virtual URS is introduced.

2. Virtual Ubiquitous Robotic Space

2.1 Concpet of Virtual URS

The virtual URS is a virtual space for intuitive human-URS (or robot) interface, which provides geometry and texture information of the corresponding physical space. Fig. 1 shows the concept of virtual URS which is an intuitive interface between human and physical URS. In physical URS, there may be robot and ubiquitous sensor network (USN) which are real things in our close indoor environment. For example, the robot can perform the security duty and sensor network information is updated to be used as a decision ground of whole devices' operation. The virtual URS is the intermediate between the real robot space and human. It can represent the status of physical URS, e.g., robot position and sensor position/status based on 2D/3D indoor model.

2.2 Concept of Responsive Virtual URS

The virtual URS can be responded according to the sensor status. We construct sensor network in virtual URS and define the space as a responsive virtual URS. In other words, responsive virtual URS is generated by modeling of indoor space and sensors. So sensor status is reflected to space. As a simple example, the light rendering in virtual URS can be changed according to the light sensor information in physical space. This is a concept of responsive virtual URS which provides similar environment model to the corresponding physical URS status. In other words, when event happens in physical URS, the virtual URS responds. Fig. 2 shows that the responsive virtual URS is based on indoor modeling and sensor modeling.

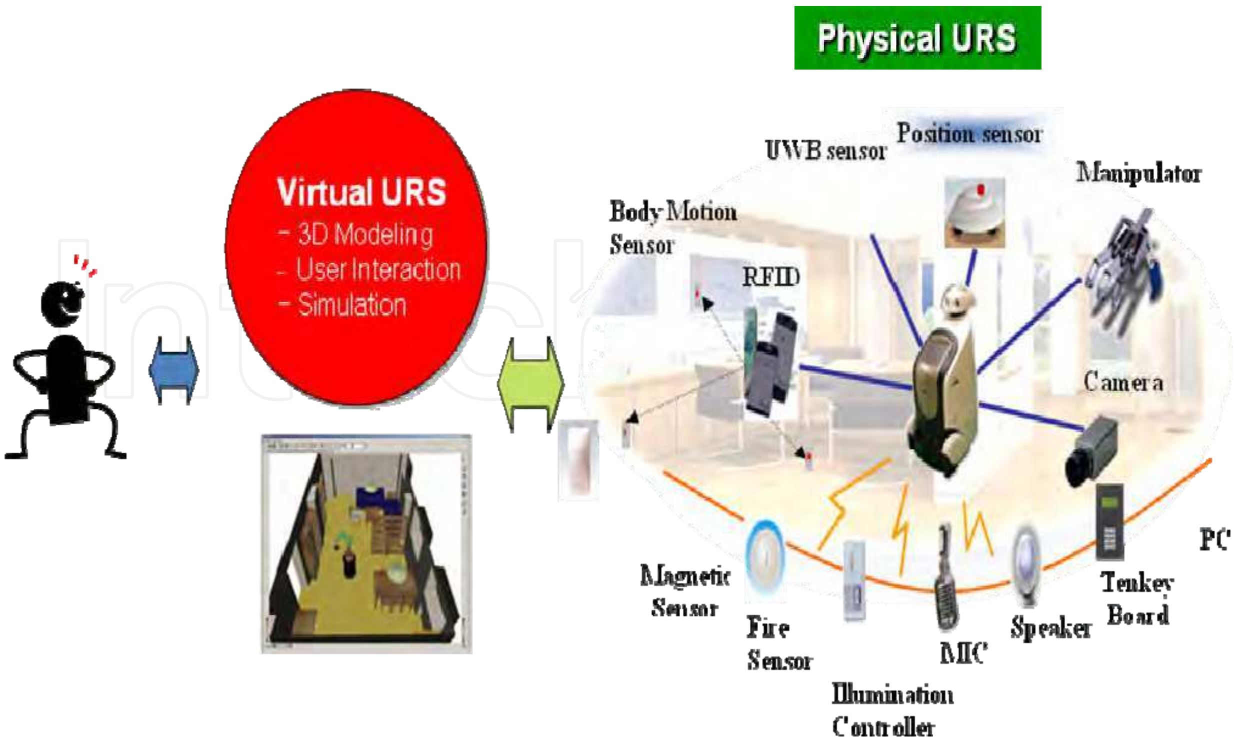

Fig. 1. Concept of physical and virtual URS

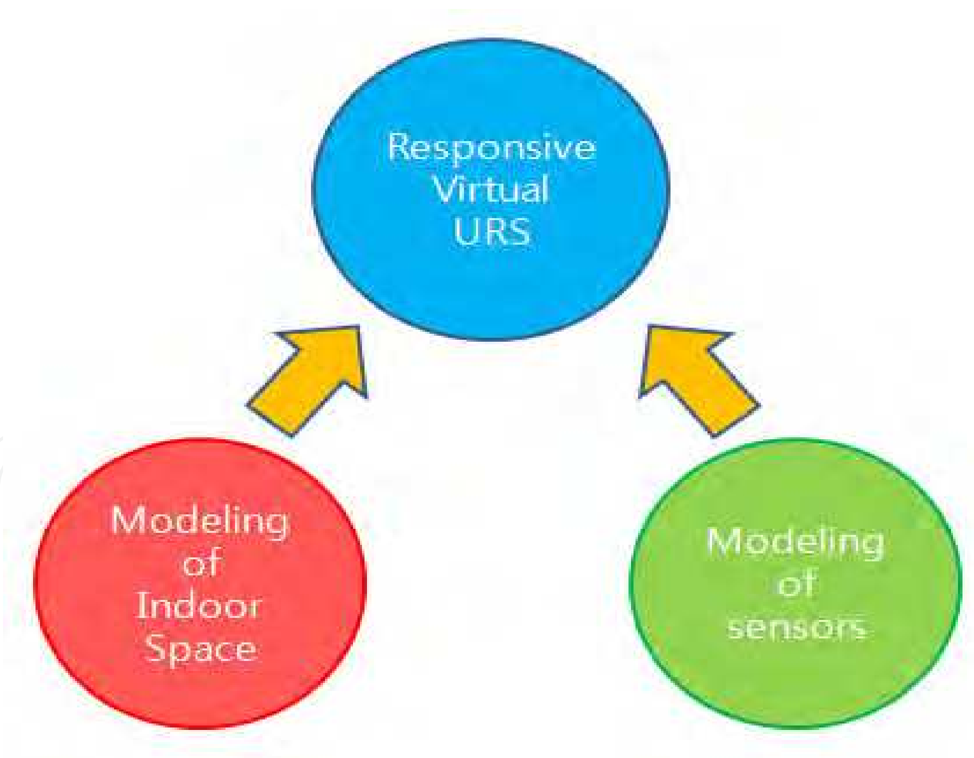

Fig. 2. The concept of responsive virtual URS

3. Modeling Issues

3.1 Modeling of Indoor Space

This section gives an overview of our method to build a 3D model of an indoor environment. Fig. 3 shows our approach for indoor modeling. As shown in Fig. 3, there are three steps, localization of data acuqisition device, acquisition of geometry data, texture image capturing and mapping to geometry model.

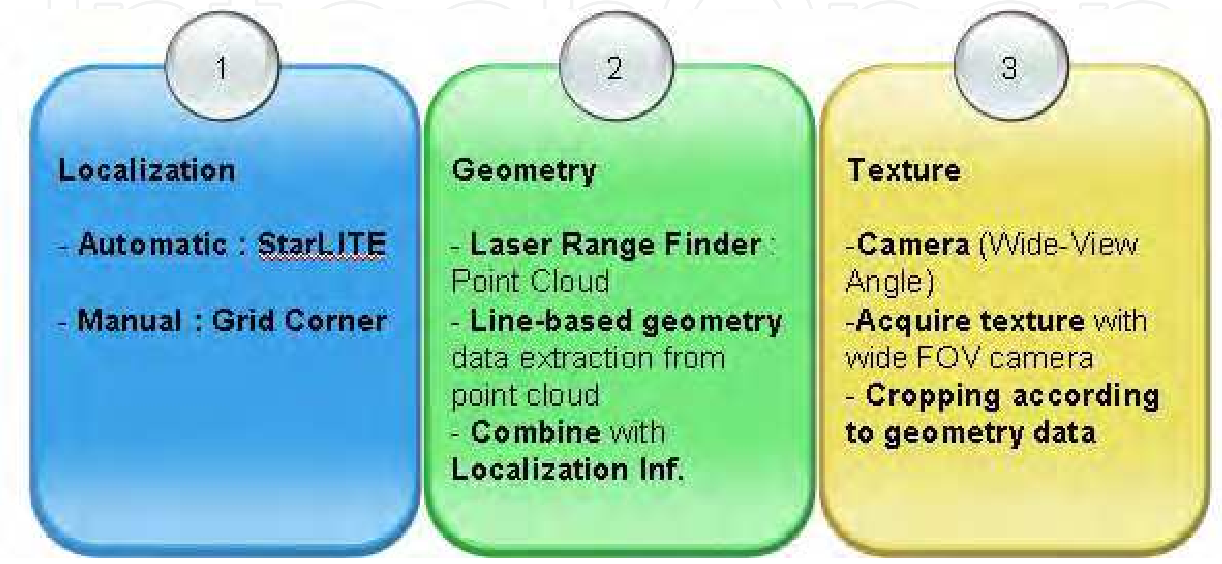

Fig. 3. Indoor modeling process

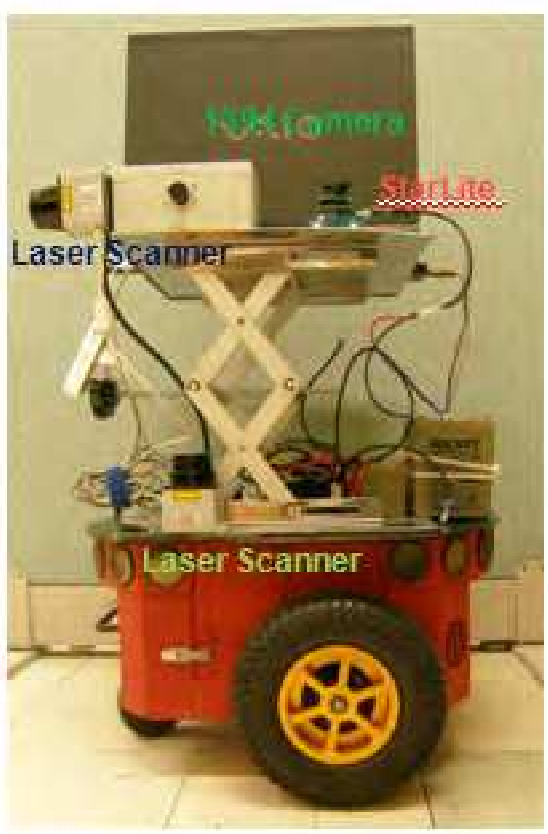

Fig. 4. Indoor 3D modeling platform

The localization information is used for building the overall indoor model. In our research, we use two approaches. One is using IR landmark-based localization device, named as starLITE (Heeseoung Chae, Jaeyeong Lee and Wonpil Yu 2005), the other is using dimension of floor square tile (DFST) manually. The starLITE approach can be used automatic localization. The DFST approach is applied when starLITE is not installed. The DFST method can be used easily in the environment that has reference dimension without the additional cost for the localization device, although it takes times due to the manual localization.

In case of 2D model & 3D model, the geometry data is acquired with 2D laser scanner. In case of 2D laser scanner, we used two kinds of laser scanners, i.e., SICK LMS 200 and Hokuyo URG laser range finder.

Fig. 4 shows our indoor modeling platform using two Hokuyo URG laser range finder (LRF) and one IEEE-1394 camera. One scans indoor environment horizontally and another scans indoor environment vertically. From each LRF, we can generate 2D geometry data by gathering and merging point clouds data. Then, we can get 3D geometry information by merging two LRF 2D geometry data. For texture, the aligned camera is used to capture texture images. Image warping, stitching and cropping operations are applied to texture images.

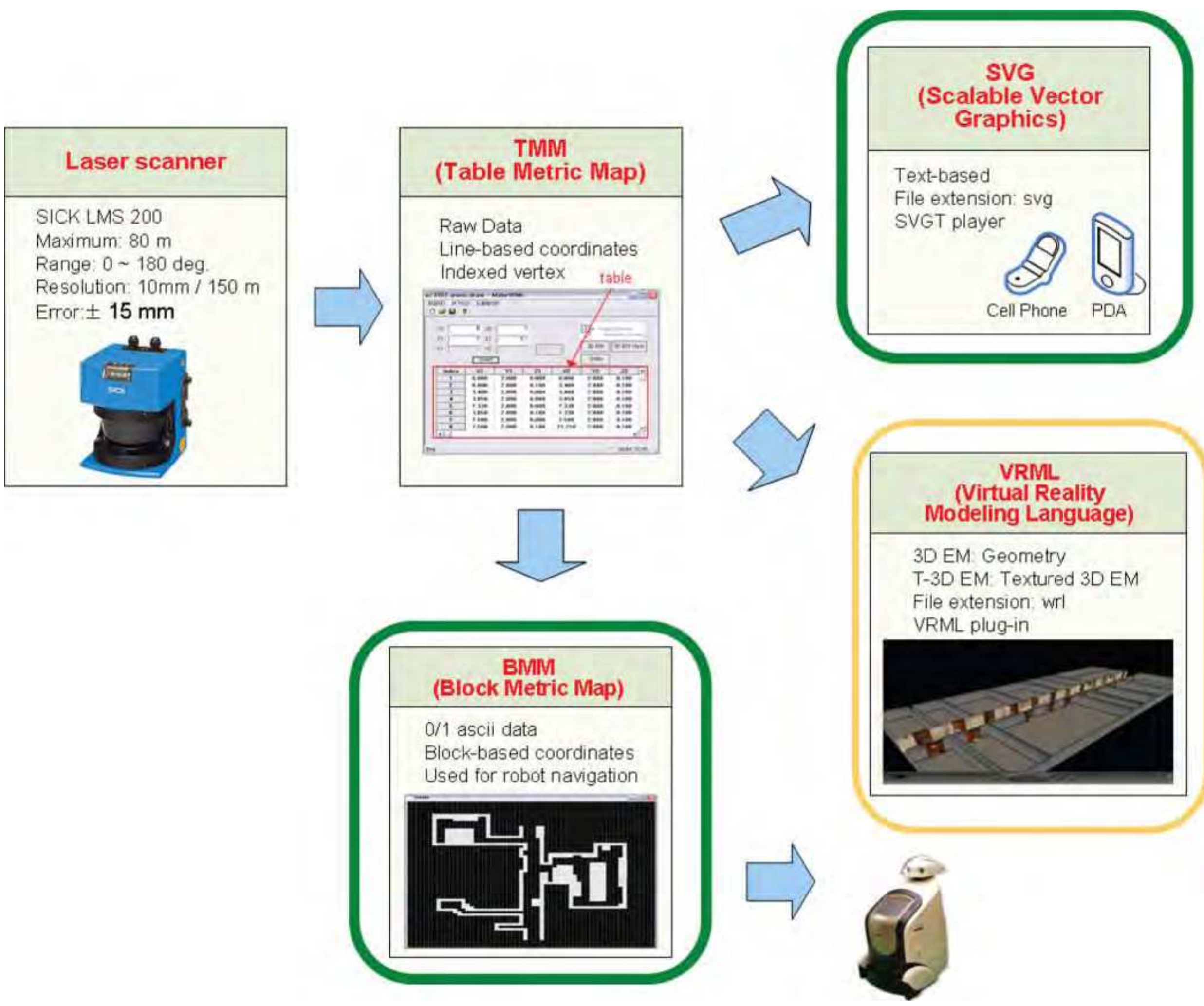

Fig. 5. Data flow for building 2-D and 3-D models of an indoor environment

Fig. 5 shows data flow for building 2-D and 3-D models of an indoor environment. Based on these geometry data acquired from laser scanner, TMM (Table Metric Map), GMM (Graphic Metric Map) and 3D EM (Environment Map) are generated automatically. It is also possible to apply same procedure if there is a metric flat drawing. In this case, the data acquisition with laser scanner is not needed. For the web service, the indoor model is generated based on SVG (2D) and VRML (3D). BMM (Block Metric Map) is also generated for the robot navigation map.

3.2 Modeling of Environment Sensor

(1) Sensor modeling based on XML

We implement XML based environment sensor modeling. Especially we design sensor XML GUI and develop XML sensor file generator according to the input data from GUI. As an example, the sensors we use are light, fire and gas sensor. We design the model data of sensors with sensor id, type, location and status.

Using our sensor XML GUI shown in Fig. 6, user can create the XML file for each sensor. In other words, user can input sensor id, type, location and status using GUI and then the corresponding XML sensor file is generated. For same sensor type, We can describe many sensors while using different sensor ids.

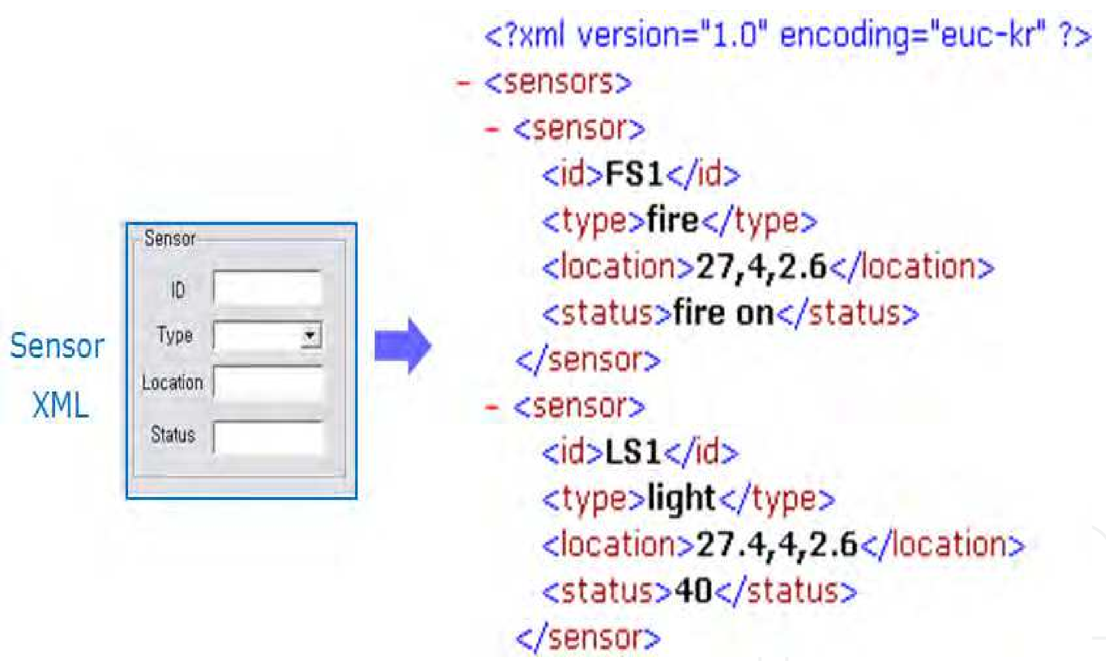

Fig. 6. Sensor XML GUI and generated XML data

(2) Automatic acquisition of sensor installation

We can acquire sensor information automatically. The sensor network is based on zigbee network. Sensor base station detects sensor which is now working and sensor information (sensor id, sensor type, sensor status etc.) based on data logging. So user is able to confirm the sensors that are working and know their id, name and value. All information are saved and managed by XML.

Fig. 7 shows automatic detection of newly installed sensor and addition of new sensor XML data into the previous XML sensor file.

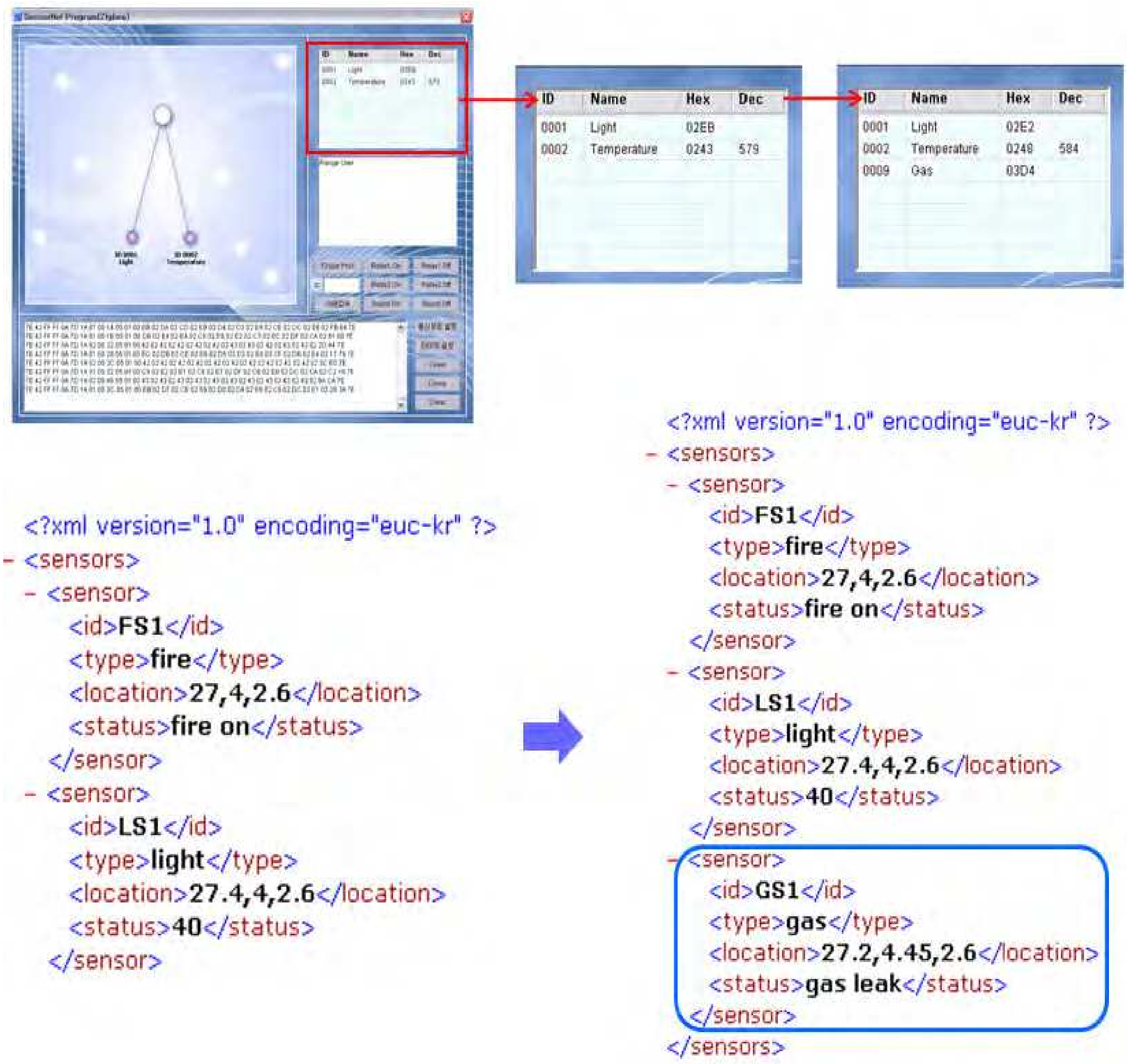

Fig. 7. Detection of sensor module and automatic addition of sensor model data in XML file

(3) Registration of sensor location

It should be noted that when we make a model for each sensor installed, the sensor id, type and status can be inserted automatically in XML sensor file. However, the location information of sensor is not easy to insert automatically, because we need to measure 3D location of sensor.

In this paper, we implement the virtual URS based input of sensor location information into XML file. Here, we assume that sensor is installed at ceiling and the height information is measured once and we know it. User can point out the corresponding sensor location roughly using the virtual URS (2D map or bird eye view 3D model). Then, the (x,y) data of sensor location can be extracted automatically and merged with the known height data for 3D location data of sensor. This 3D location data is saved to XML file automatically and virtual sensor is generated in the virtual URS according to XML file. Fig. 8 shows the basic concept of the virtual URS based insertion of sensor location data into XML file.

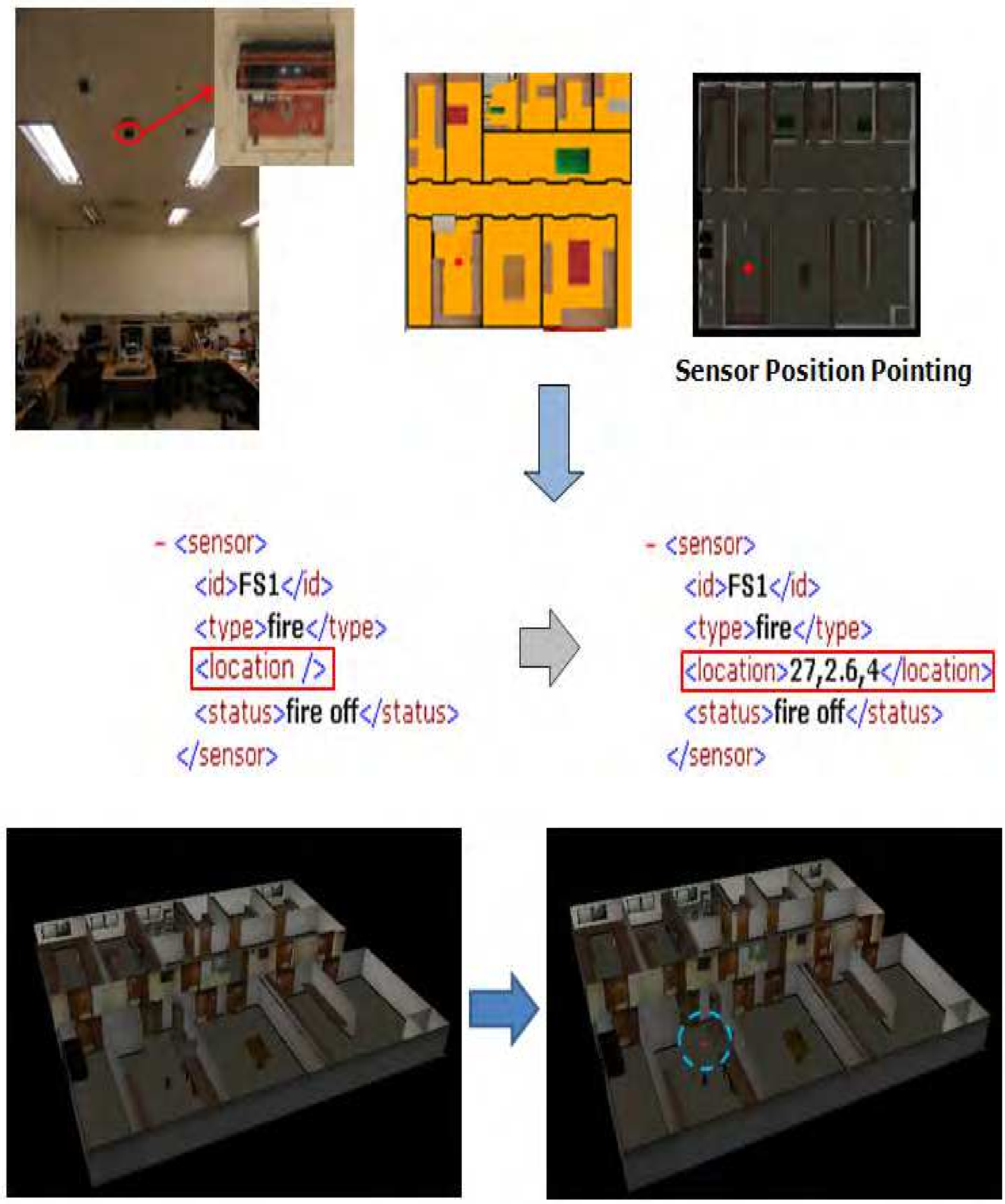

Fig. 8. Virtual URS-based insertion of sensor location to XML file

4. Virtual URS Services

4.1 Network-based Human-Robot Interaction

User can interact with robot through web server as shown in Fig. 9. For example, user can designate the destination point of robot and also receive the current robot position information through network. For interaction with robot, user can use several kinds of terminals like PC, PDA or mobile phone. Here, web server provides services of human-robot interaction.

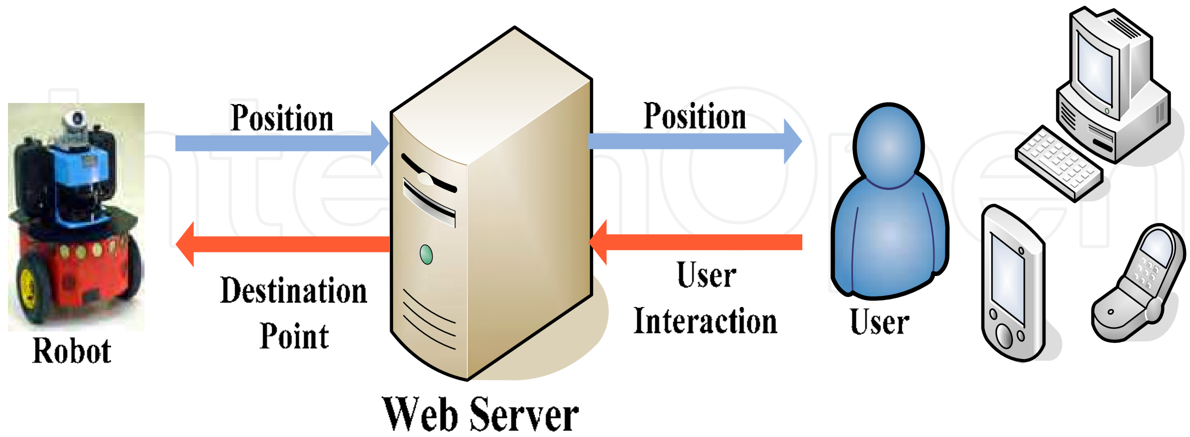

Fig. 9. Basic concept of network-based human-robot interaction

Here, we introduce two kinds of network-based human-robot interaction (HRI) services, i.e., HRI through web-browser and mobile phone. Fig. 10 shows overview of network-based human-robot interaction service system.

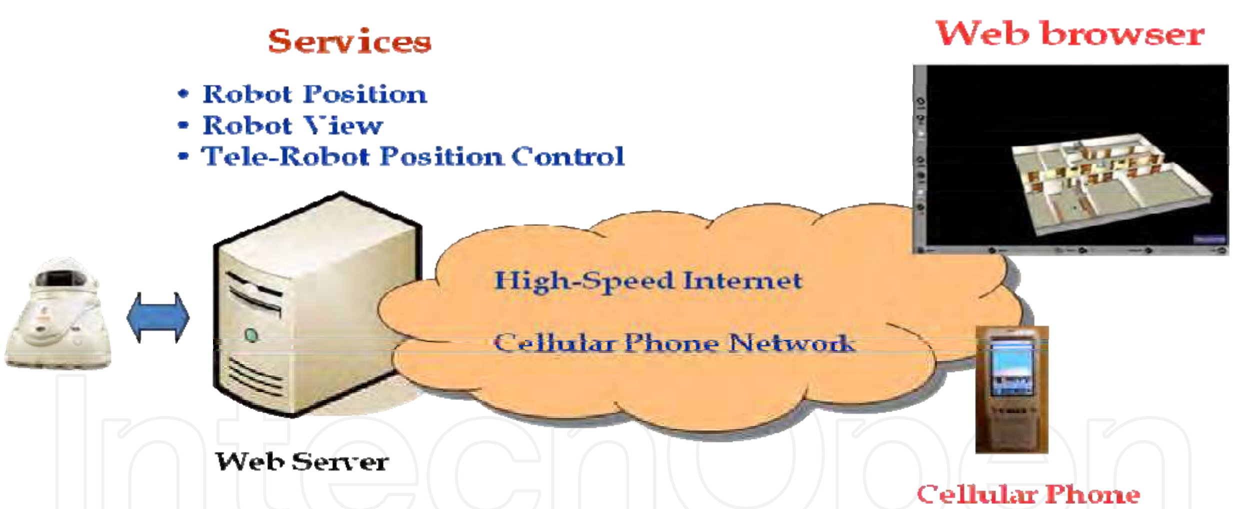

Fig. 10. Overview of network-based human-robot interaction service system

(1) Web browser -based interactive service

Basically, the virtual URS provides 2D/3D Model of URS, which supports indoor space model, object model and status update of robot or some object in space according to event occurred in physical URS. The virtual URS also provides the function such as display of robot location of physical URS, designating point and robot path planning. Through bridging between the virtual URS and the physical URS, user is able to command robot to move. Moreover, the user can pick many destinations to decide the robot path so that the robot will move according to the designated path. The functions are possible in remote environment through web.

Fig. 11 shows web browser-based interactive tele-presence through network, for example, tele-presence between KIST in Seoul, Korea and ETRI in Daejon, Korea. As shown in Fig. 11, our system can provide telepresence with the virtual URS including 3D indoor model, robot, sensor and map information. Moreover, user can interact with remote site robot through the virtual URS.

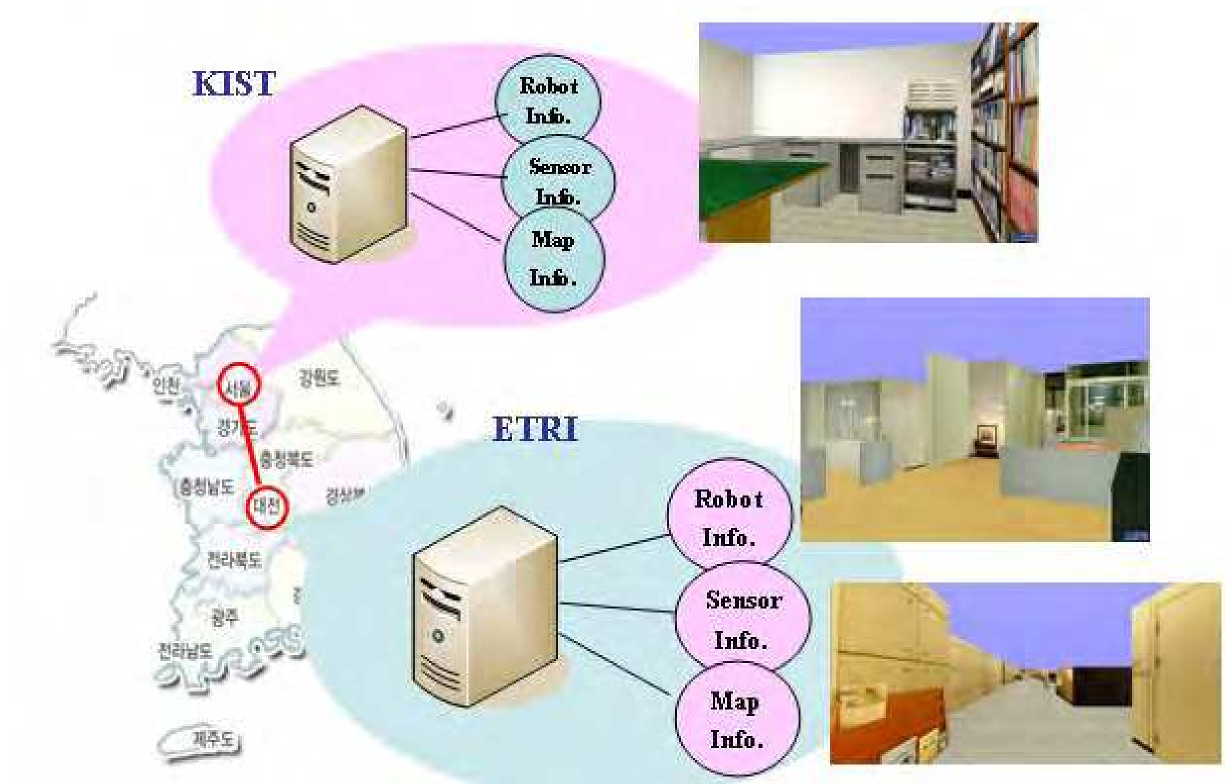

Fig. 11. Web browser-based interactive tele-presence

(2) Mobile phone-based interactive service

If user designates robot position in the virtual URS through web, the remote physical robot moves to the designated location. This function is also possible through mobile phone. It is possible that user can see the robot position, robot view in the physical URS through the 3D virtual URS while using mobile phone. Moreover user can control the robot in physical URS on mobile phone.

The 3D robot view service is impossible on general mobile phone without 3D engine. So we design a service platform for 3D mobile phone service (Kyeong-Won Jeon, Yong-Moo Kwon, Hanseok Ko, 2007). The service platform for the 3D virtual URS service on mobile phone is composed of 3D model server, 3D view image generation, mobile server and mobile phone.

- The 3D model server manages 3D model (VRML). Several 3D models exist in 3D model server.
- The 3D view image generation part is composed of 3D model browser and 3D model to 2D image converting program. 3D model browser is to render 3D view in 3D model. So user can see 3D view through the 3D model browser. Then, the rendered image is converted to 2D image (jpg).

- The mobile server manages the saved 2D image and sends it to mobile phone by TCP/IP communication. Another role of mobile server is to transfer interaction information from mobile phone to 3D model browser for interaction service between mobile phone and 3D model browser

Fig. 12 shows the architecture of mobile phone-based interactive service. Fig. 13 shows mobile phone based interaction to the virtual URS. Fig. 14 shows 3D view image on the mobile phone.

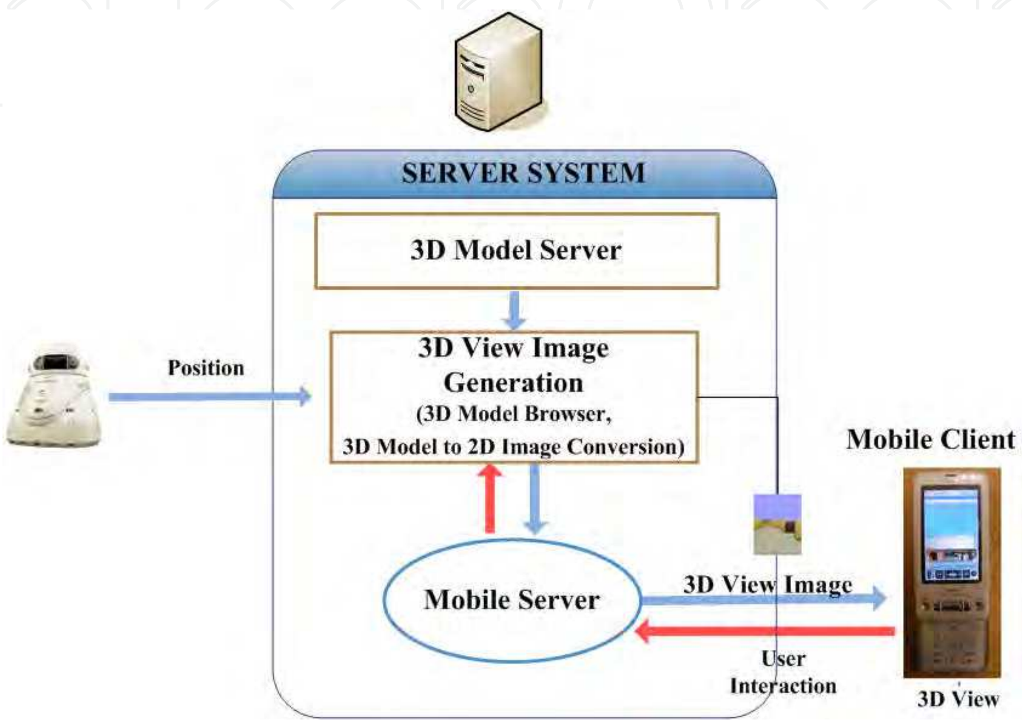

Fig. 12. Architecture of mobile phone-based interactive service

Fig. 13. Mobile phone based interaction to the virtual URS

Fig. 14. 3D view image on the mobile phone

4.2 Sensor-Responsive Virtual URS

We provide sensor-responsive virtual URS service by bridging between the physical URS and the virtual URS. When an event happens in physical space, the sensor catches the event. Then the sensor id, sensor status information are delivered to the web server through the wireless network (for example, zigbee network). Upon receiving sensor status change information, the XML data is also updated automatically. In case of the robot position, it is continuously detected by sensor and then the XML robot data (robot position information) is updated. The XML robot data is reflected to robot in the virtual URS. Here, the XML file acts like a virtual sensor in the virtual URS. Then, the virtual URS also responds according to the virtual sensor status.

For example, if the status of fire sensor is activated, this information is transferred to the virtual URS and then the fire status in XML data is changed. Fig. 15 shows an automatic robot sensor status update in XML file.

The merit of VR technology is that user can experience virtually without experiencing actually. Because the virtual URS provides visual service, user can feel realistically by virtual experience. That is, visualization of the situation of physical URS is the role of virtual URS. User can confirm the status and position of robot and the situation of environment. Moreover, when event happens, robot view service is possible according to robot movement. Fig. 16 shows XML-based bridging between the physical URS and the virtual URS. Fig. 17 shows visualization service of senor in the virtual URS. Fig. 18 and Fig. 19 show a responsive virtual URSs accoring to the fire and light sensors, repectively.

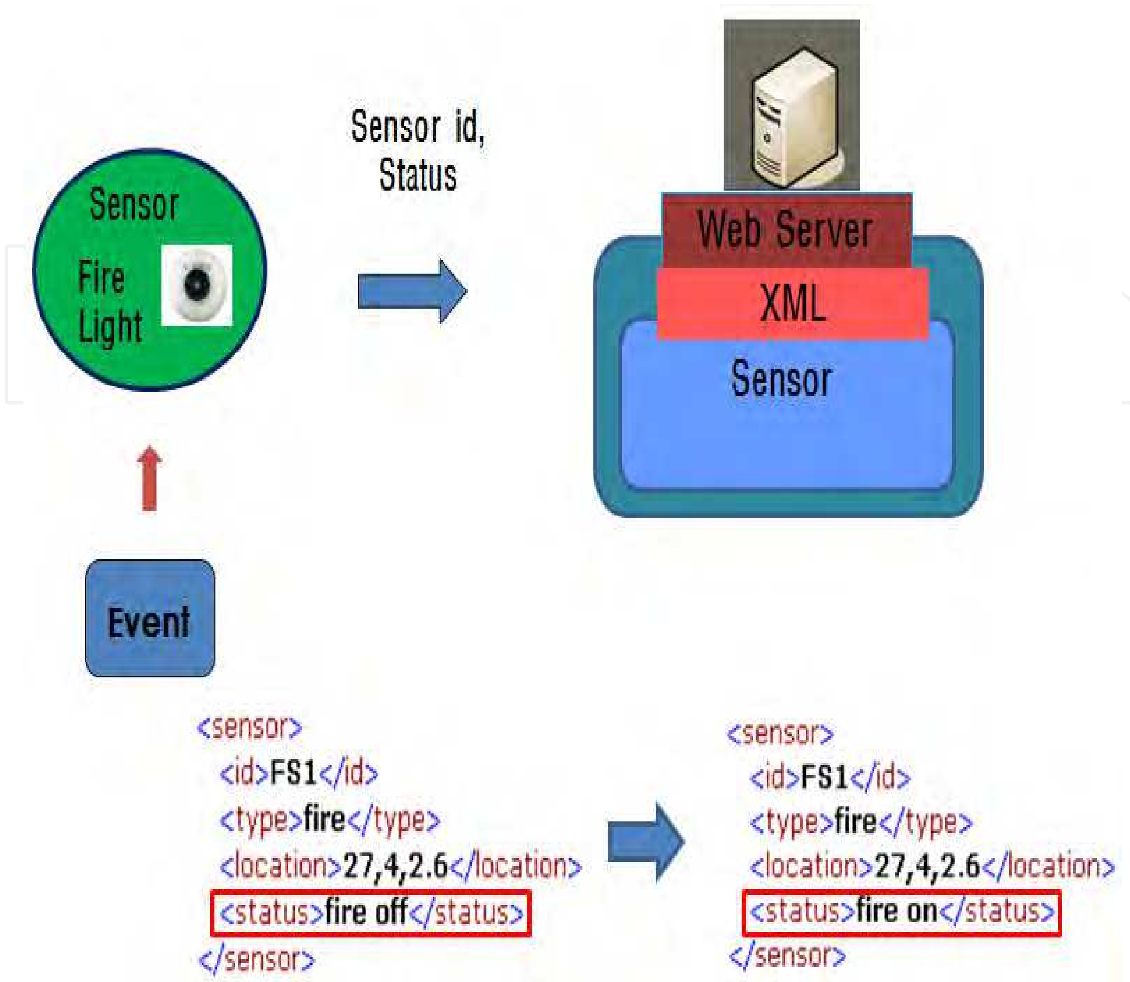

Fig. 15. Automatic robot sensor status update in XML file

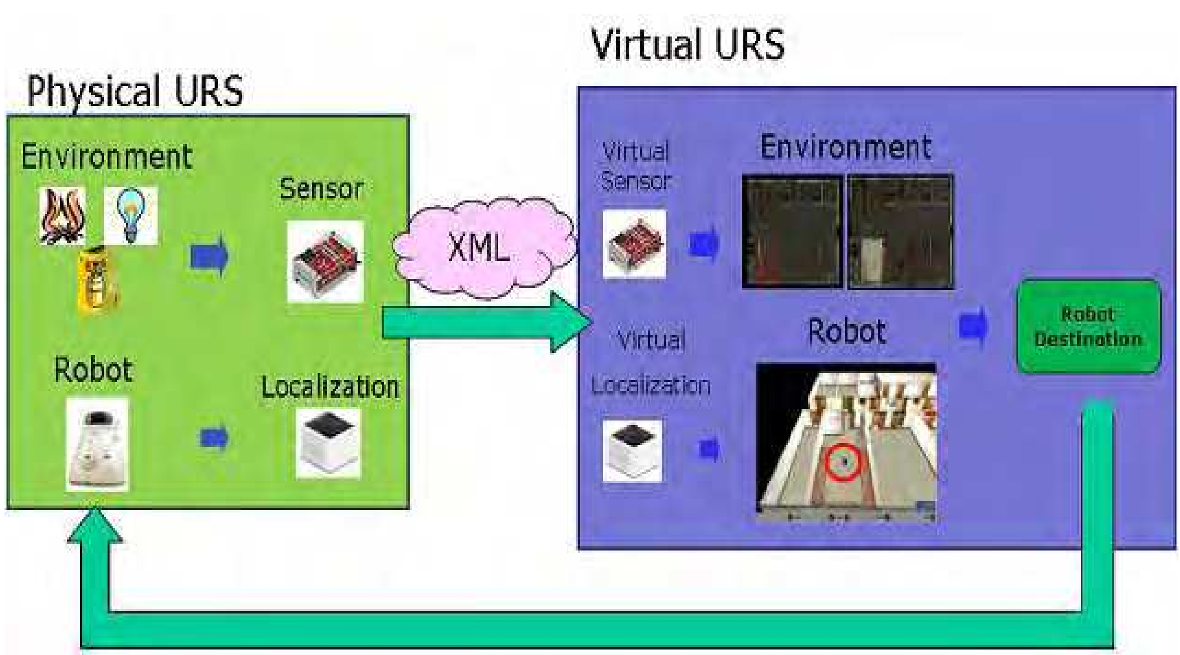

Fig. 16. XML-based bridging between the physical URS and the virtual URS

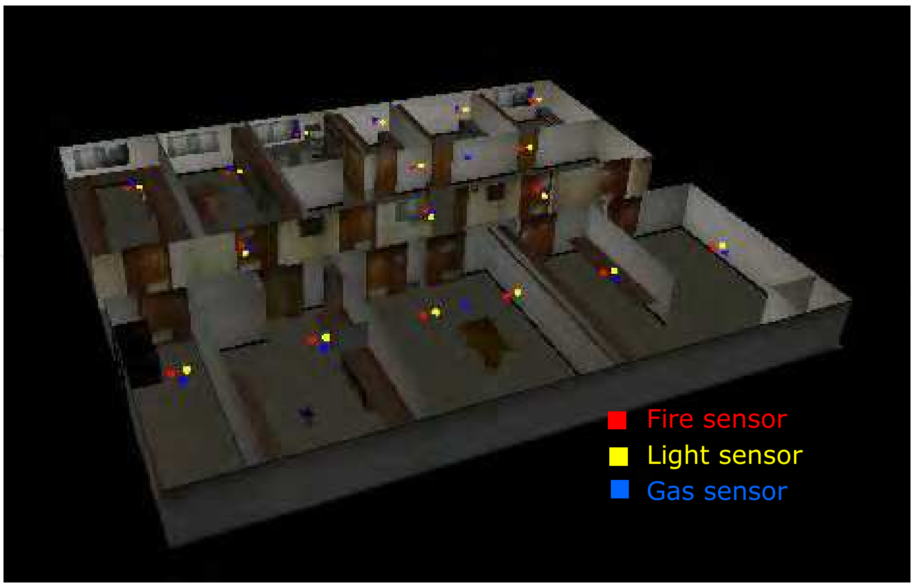

Fig. 17. 3D Responsive virtual URS – 3D visualization of sensor distribution

Fig. 18. Fire sensor-based event visualization

Fig. 19. Light sensor-based visualization

Fig. 20 shows an application scenario of the virtual URS while bridging with physical URS. When fire event occurs, Fig. 20 shows how to coordinate between the physical URS and the virtual URS. Here, the virtual URS visualizes the status of indoor space and a robot will be moved to the fire place for extinguishing fire.

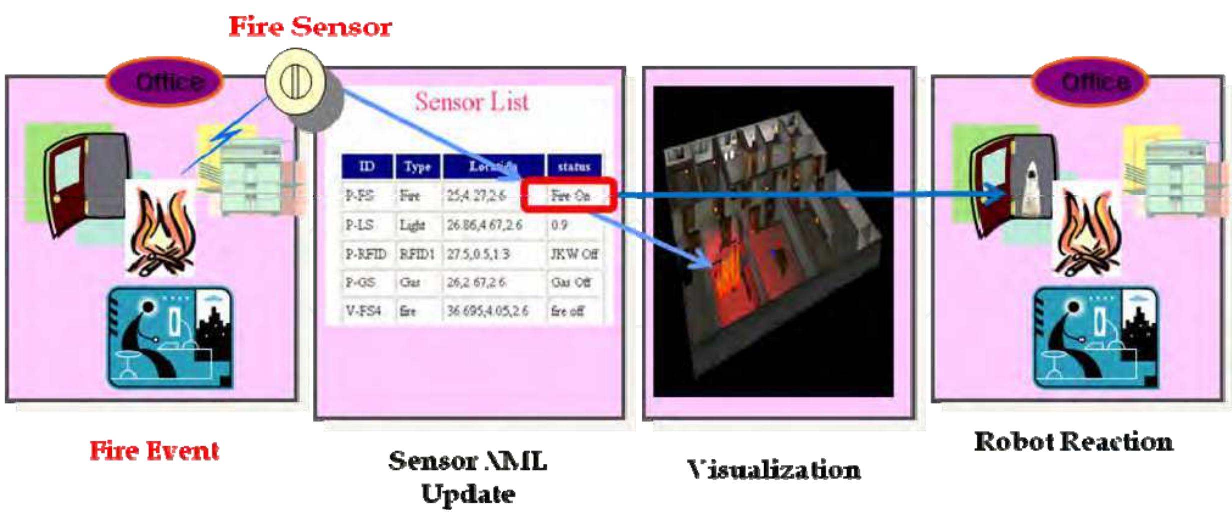

Fig. 20. Application scenario of the virtual URS when fire event occurs

Fig. 21 shows a real implementation of bridging service between the physical URS and the virtual URS. In Fig. 21, when temperature becomes over 50 degree, the virtual URS is responding and the robot moves to the fire place.

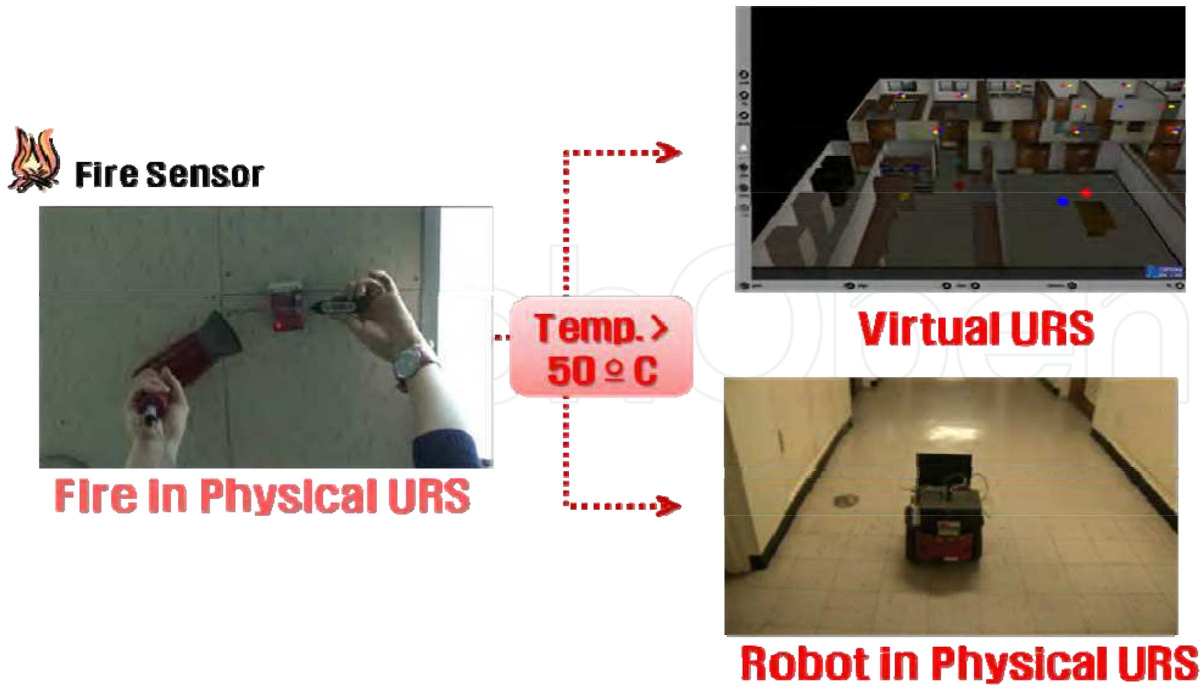

Fig. 21. Implementation of bridging service between the physical URS and the virtual URS

5. Summary

This chapter presents the modeling technique of indoor space and XML-based environment sensor and the robot service technique while bridging between the physical space and the virtual space. This chapter describes our approaches of indoor space and environment sensor modeling. Our sensor modeling system provides sensor XML GUI, sensor XML file generation, zigbee based detection of sensor module and automatic addition of sensor model data into XML file. The bridging system between the physical URS and the virtual URS is also implemented using web server while sensor status is reflected into XML file automatically. Sensors detect the robot position and situation and the detected information is reflected to the virtual URS. This chapter also describes the interactive robot service. User is able to control robot through the virtual URS. The interactive service is possible on mobile phone as well as web.

Acknowledgment

This work was supported in part by the R&D program of the Korea Ministry of Knowledge and Economy (MKE) and the Korea Evaluation Institute of Industrial Technology (KEIT) [2005-S-092-02, USN-based Ubiquitous Robotic Space Technology Development].

6. References

Peter Biber, Henrik Andreasson, Tom Duckett, and Andreas Schilling, et al. (2004), "3D Modeling of Indoor Environments by a Mobile Robot with a Laser Scanner and Panoramic Camera," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004)*

Heeseoung Chae, Jaeyeong Lee and Wonpil Yu (2005), "A Localization Sensor Suite for Development of Robotic Location Sensing Network," *(ICURAI 2005)*

Hahnel, W. Burgard, and S. Thrun (July, 2003), "Learning Compact 3D Models of Indoor and Outdoor Environments with a Mobile Robot," Elsevier Science, *Robotics and Autonomous Systems*, Vol. 44, No. 1, pp. 15-27

Kyeong-Won Jeon, Yong-Moo Kwon, Hanseok Ko (2007), Interactive 3D Virtual URS Service based on USN on Mobile Phone," International Conference on Control, Automation and Systems 2007, Oct. 17-20, 2007 in COEX, Seoul, Korea

Y. Liu, R. Emery, D. Chakrabarti, W. Burgard and S. Thrun (2001), "Using EM to Learn 3D Models of Indoor Environments with Mobile Robots", *18th Int'l Conf. on Machine Learning, Williams College*, June 28-July 1, 2001

Wonpil Yu, Jae-Yeong Lee, Young-Guk Ha, Minsu Jang, Joo-Chan Sohn, Yong-Moo Kwon, and Hyo-Sung Ahn (Oct. 2009), "Design and Implementation of a Ubiquitous Robotic Space," IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, VOL. 6, NO. 4, pp. 633-640

Remote and Telerobotics
Edited by Nicolas Mollet

ISBN 978-953-307-081-0
Hard cover, 220 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Any book which presents works about controlling distant robotics entities, namely the field of telerobotics, will propose advanced technics concerning time delay compensation, error handling, autonomous systems, secured and complex distant manipulations, etc. So does this new book, Remote and Telerobotics, which presents such state-of-the-art advanced solutions, allowing for instance to develop an open low-cost Robotics platform or to use very efficient prediction models to compensate latency. This edition is organized around eleven high-level chapters, presenting international research works coming from Japan, Korea, France, Italy, Spain, Greece and Netherlands.