6

Design and Implementation of Intelligent Space: a Component Based Approach

Takeshi Sasaki and Hideki Hashimoto
Institute of Industrial Science, The University of Tokyo
Japan

1. Introduction

In recent years, the research field on smart environments, which are spaces with multiple embedded and networked sensors and actuators, has been expanding (Cook & Das, 2004). The smart environments observe the spaces using distributed sensors, extract useful information from the obtained data and provide various services to users. Such an environment is also referred to as smart space, intelligent environment, etc., and many researchers have developed smart environments for providing informative services to the users (e.g. support during meeting (Johanson et al., 2002), health care (Nishida et al., 2000), support of the elderly (Mynatt et al., 2004), information display using a pan-tilt projector (Mori et al., 2004)). On the other hand, smart environments are also used for support of mobile robots that work in complicated human living environments. In this type of smart environments, mobile robots inside the space get necessary information from multiple distributed sensors and various functions such as localization, path planning and human-robot interaction are performed with the support of the system (Mizoguchi et al., 1999), (Sgorbissa & Zaccaria, 2004), (Koide et al., 2004).

Aiming to provide both informative and physical services to the users, we have also been developing a smart environment, called Intelligent Space (iSpace), since 1996 (Lee & Hashimoto, 2002). Fig. 1 shows the concept of iSpace. In iSpace, not only sensor devices but also sensor nodes are distributed in the space because it is necessary to reduce the network load in the large-scale network and it can be realized by processing the raw data in each sensor node before collecting information. We call the sensor node devices distributed in the space DINDs (Distributed Intelligent Network Device). A DIND consists of three basic components: sensors, processors and communication devices. The processors deal with the sensed data and extract useful information about objects (type of object, three dimensional position, etc.), users (identification, posture, activity, etc.) and the environment (geometrical shape, temperature, emergency, etc.). The network of DINDs can realize the observation and understanding of the events in the whole space. Based on the extracted and fused information, actuators such as displays or projectors embedded in the space provide informative services to users. In iSpace, mobile robots are also used as actuators to provide physical services to the users and for them we use the name mobile agents. The mobile agent can utilize the intelligence of iSpace. By using distributed sensors and computers, the

mobile agent can operate without restrictions due to the capability of on-board sensors and computers. Moreover, it can understand the request from people and offer appropriate service to them.

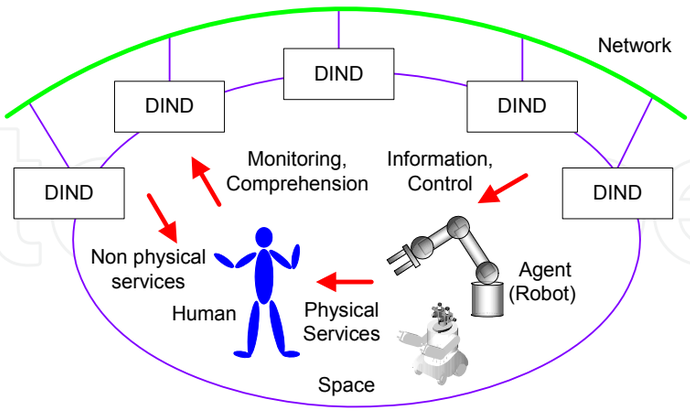


Fig. 1. Concept of Intelligent Space (iSpace)

Smart environments should have flexibility and scalability so that we can easily change the arrangement of embedded devices and switch applications depending on the size of the space, technological advances, etc. Therefore, the system integration becomes an important issue. In this chapter, the problem of implementation of iSpace is addressed. To be more precise, component based system integration of iSpace is described. In order to implement the system efficiently, a development support platform of robot systems is utilized as a RT (Robot Technology) Middleware.

Next section gives a selection of a development support platform. Component design is discussed in section 3. Section 4 describes the implementation of robot technology components. In section 5, experimental results of mobile robot navigation is presented. Conclusion and future work are given in section 6.

2. Development Support Platform of Robot System

2.1 Review of existing platforms

Until now, various types of development support platform of robot systems have been developed. We review some existing platforms in this subsection.

Player/Stage/Gazebo (Player Project, http://playerstage.sourceforge.net/) (Gerkey et al., 2003) is one of the most famous platforms for mobile robot control. Player is a network server for mobile robot control. Player provides interfaces to obtain information from various types of sensors and control actuators. Stage and Gazebo are a 2D and a 3D simulator, respectively. Since the simulators have same interface as that of actual robots, algorithms tested on the simulators can be applied to the actual systems without making major changes in the source codes. Another development support platforms for mobile robot applications is Miro (Neuroinformatik: Robotics, http://www.informatik.uni-ulm.de/neuro/index.php?id=301&L=1) (Utz et al., 2002) that is a middleware for mobile robots. Since Miro uses distributed object technology CORBA (Common Object Request Broker Architecture), the platform can ensure the connectivity of programs that operate on different

operating systems. In addition, Miro provides some basic functions used for mobile robot control including self-localization and map building as class of objects.

Some platforms are also developed for other specific applications. ORiN (Open Resource interface for the Network / Open Robot interface for the Network) (Mizukawa et al., 2004) is a communication interface to provide unified access method to the devices. ORiN also uses DCOM (Distributed Component Object Model) or CORBA as a distributed object middleware. The main target application of ORiN is FA systems. PEIS Middleware (Broxvall, 2007) is a middleware for ubiquitous applications developed in PEIS Ecology project (PEIS Ecology Homepage, http://www.aass.oru.se/˜peis/). PEIS Middleware realizes cooperation of distributed devices by using a tuple space which is a kind of a shared memory space. This project also develops Tiny PEIS kernel for tiny networked embedded devices which do not have enough memory.

The other platforms are designed for multiple purposes so that various kinds of robot elements can be developed. In ORCA project (orca-robotics project, http://orca-robotics. sourceforge.net/) (Brooks et al., 2007), a structure of component is defined and the system is developed based on these components. ORCA uses Ice (Internet Communication Engine) as a distributed object middleware. This project has component repository and various components can be downloaded from the website. OpenRTM-aist (RT-Middleware: OpenRTM-aist Official Website, http://www.is.aist.go.jp/rt/OpenRTM-aist/) (Ando et al., 2005) also supports component based system development. OpenRTM-aist utilizes CORBA for ensuring the connectivity between components on the network. The members of this project are working on standardization of Robotic Technology Component in OMG (Object Managing Group) (OMG, 2008) and the latest version of OpenRTM-aist complies with the specification adopted by OMG. OpenRTM-aist also promotes improvement of the development environment and offers a template code generator which makes a source code of a component from the specification of the component (e.g. number of I/O ports, etc.) and a system design tool which provides graphical user interface to change the connection of components and start/stop the system. In addition, a lightweight middleware RTC-Lite is developed for embedded systems that have insufficient resources to operate OpenRTM-aist.

Microsoft Robotics Studio (Microsoft Robotics, http://msdn. microsoft.com/en-us/robotics/default.aspx) (Jackson, 2007) is a development environment for robot systems which adopts service oriented architecture. Microsoft Robotics Studio provides various kinds of tools for implementing robot systems efficiently, for example, a library for asynchronous programming, a visual programming language, a 3D simulator and so on.

2.2 Selection of platform for development of iSpace

In order to determine a platform that is used in this research, we consider following criteria.

1. *Modularity:* In module or component based systems, independent elements (modules or components) of functions of the systems are first developed and the systems are then built by combining the modules. The modularization increases maintainability and reusability of the elements. Moreover, flexible and scalable system can be realized since the system is reconfigured by adding or replacing only related components.
2. *Standardization:* In order to receive the benefit of modularization, it is necessary to ensure the connectivity between components. This means that the interface between components should be standardized. Considering the cooperation of components that are developed by various manufactures, international standardization is desirable.

3. *Suitability for iSpace application:* As mentioned above, some platforms are developed for specific applications, for example, mobile robots and industrial robots. The platforms which are aimed at network sensing or ubiquitous computing are appeared to be suitable for iSpace. However, as also shown in the network robot (Hagita, 2006) and the ubiquitous robot (Kim et al., 2007) concept, iSpace pays attention to cooperation of various types of robots including mobile robots and software robots. Therefore, development support platforms should be applied to various applications.

Table 1 shows a comparison of the existing platforms mentioned in the previous subsection from the view point of modularity, standardization and suitability for iSpace application. We can find that only OpenRTM-aist (AIST, Japan) meets all of our requirements described above. So we decided to use OpenRTM-aist in this research.

Platform	Style of system development	Network	Standardization activity	Target application
Player/Stage/Gazebo	Object oriented	Socket	-	Mobile robot
Miro	Object oriented	CORBA	-	Mobile robot
ORiN (ORiN Ver.2)	Object oriented	DCOM, CORBA	ISO	FA system
PEIS Middleware	Component based	P2P / tuple space	-	Ubiquitous device
ORCA (ORCA2)	Component based	Ice	-	Versatile
OpenRTM-aist	Component based	CORBA	OMG	Versatile
Microsoft Robotics Studio	Service oriented	SOAP	-	Versatile

Table 1. Comparison of development support platforms of robot systems

3. Design of Components

From the point of view of component based system development, the design of functional unit (granularity) of components and interface between components (input and output) is important. In this section, we consider that the observation function of iSpace consists of the information acquisition part and the information integration part. In the following subsections the component design for these two sub-functions and mobile robot navigation function is discussed.

3.1 Design of the information acquisition part

Sensors which can get the same sort of data should be replaced by changing the components without any other modification. Moreover, it is desirable that processing methods can easily be changed depending on the purpose. So we decided to make sensor components and information processing components in the information acquisition part. The outputs of the sensor components are raw sensor data. In addition, the sensor components have interface which provides hardware dependent information used for information processing. The information processing components receive data from the sensor components, process the data and extract required information. Since the outputs of the information processing components are the inputs of the information integration part, the information processing components send reliability information, which is used for information fusion, as well as the extracted information. Fig. 2 shows the configuration of components in the information acquisition part.

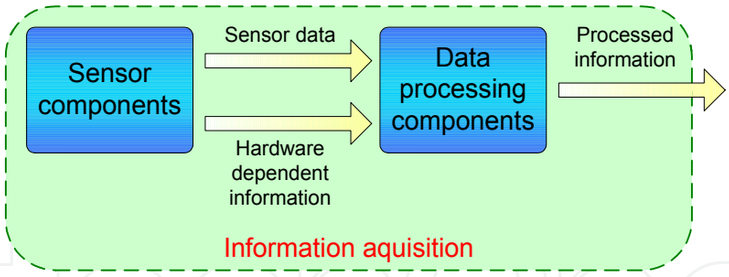


Fig. 2. Component design of information acquisition function

3.2 Design of the information integration part

The inputs and outputs of the information integration part use same data structure as those of the outputs of the information acquisition part so that, in a small system, applications can also receive information from the information acquisition components directly. The information integration part should support various usages of the obtained information, for example, utilizing all obtained information, extracting patterns from the long-term observation information, selecting information which has specific properties and so on. Therefore information from the information acquisition part is fused by the information fusion components and the fused information is stored in the database components. Fig. 3 shows the configuration of components in the information integration part.

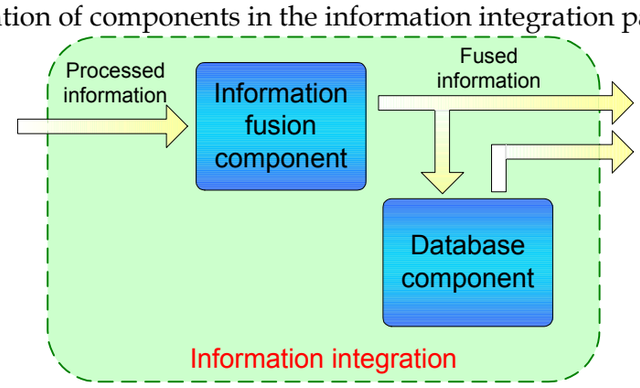


Fig. 3. Component design of information integration function

3.3 Design of mobile robot navigation function

In our previous research (Lee & Hashimoto, 2003), in order to reduce the cost of the system, mobile robots do not have external sensors and iSpace controls the mobile robots completely. However, considering the recent progress of information and robot technology, it is desirable to give autonomy to mobile robots and realize services by cooperation between iSpace and the robots. Therefore, sensors are attached to mobile robots and the mobile robots get information from iSpace according to their request or situations.

Mobile robot navigation can be realized by localization and mapping, path planning and obstacle avoidance, etc. so it is natural to make these functional units as components. We note that we can reuse some sensor and processing components of iSpace as those of mobile robots.

4. Implementation of Components

4.1 Information acquisition components

Table 2 and 3 show the major developed sensor components and data processing components, respectively.

Sensor	Output of sensor data	Output of hardware dependent information	Available devices
CCD camera	RGB image	Image properties (width, height, etc.)	OpenCV compatible cameras
Range imaging camera	Depth image	Image properties (width, height, etc.)	SwissRanger SR-3000 (MESA Imaging)
Laser range finder	Distance to objects, scan area	Observable area information	URG series (Hokuyo automatic)
Microphone	Sound data	Sampling parameters	Any
3D ultrasonic positioning system	3D positions of ultrasound tags	-	ZPS-3D (Furukawa)

Table 2. Sensor components

Sensor	Processing	Input	Output	Method
Camera	Object segmentaion	Camera image	Object positions in image coordinates	Background subtraction, color histogram matching
	3D reconstruction	Object positions in image coordinates	3D positions of objects	Epipolar geometry
Laser range finder	Tracking	Scan data	2D positions of objects Background scan data	Background subtraction
	Mapping	Background scan data	Occupancy grid map	Ray casting

Table 3. Information processing components

The sensors that are distributed in iSpace and mounted on mobile robots - CCD cameras, range imaging cameras, laser range finders, a 3D ultrasonic positioning system and so on were modularized as robot technology components.

To develop information processing components, we consider acquisition of positions of objects since position information is one of the most basic information for iSpace.

For laser range finder, we made components for measuring 2D positions of objects based on background subtraction and clustering. Fig. 4 shows the tracking process. Background subtraction is processes of extracting moving objects. The static parts of scan (background) are subtracted from the scan data in order for determining which parts of the scan are due to moving objects (foreground). The scan points in the foreground are clustered based on the Euclidian distance between them using a nearest neighbor classifier. This divides the foreground to a number of clusters, each belonging to one of the tracked object. Clusters with a small number of scan points are discarded as measurement noise. The details of our laser range finder based tracking method and evaluation of the method are described in (Brscic & Hashimoto, 2006). Fig. 5 shows examples of tracking of a mobile robot and multiple people using laser range finders. In the experiments, three laser range finders were

used and the tracked positions obtained in each laser range finder were sent to a position server mentioned below. The figure shows the fused position information in the server.

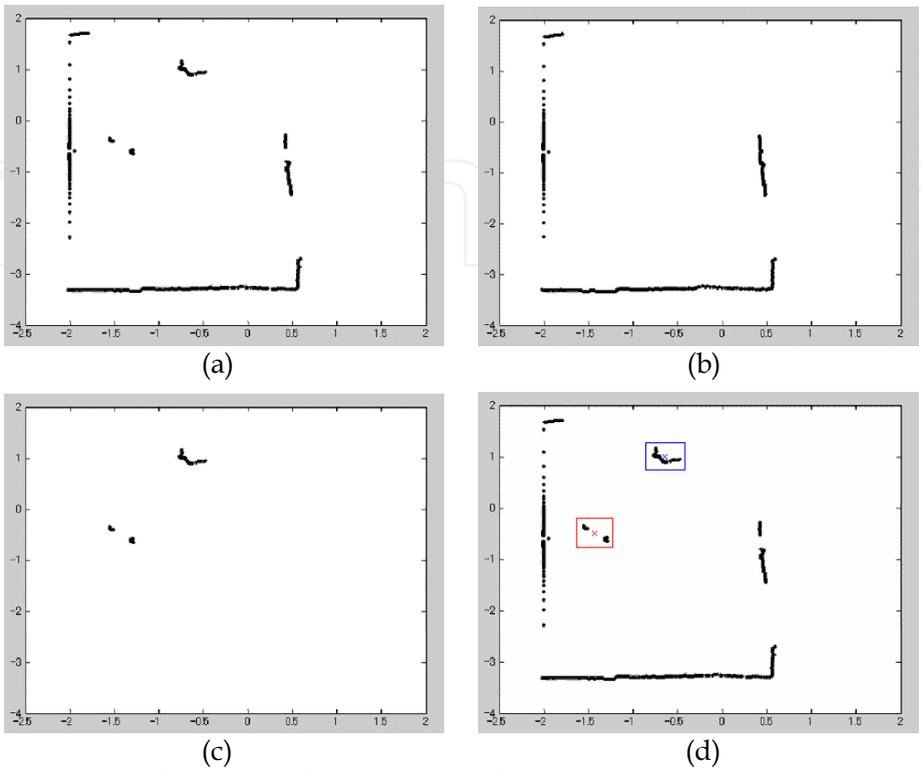


Fig. 4. Process of tracking using laser range finder (a) raw scan data, (b) static part of the scan (background), (c) result of background subtraction (foreground), (d) result of clustering and centers of the clusters. The units of x and y are in meters.

Fig. 5. Examples of mobile robot tracking and multiple people tracking using three laser range finders. The tracking results obtained in a position server mentioned below are overlaid on video images.

For CCD camera, similar algorithm to the one using laser range finder was implemented for detecting moving objects in iSpace. Fig. 6 shows an example of object segmentation based on background subtraction and color histogram matching using a CCD camera. In addition, a

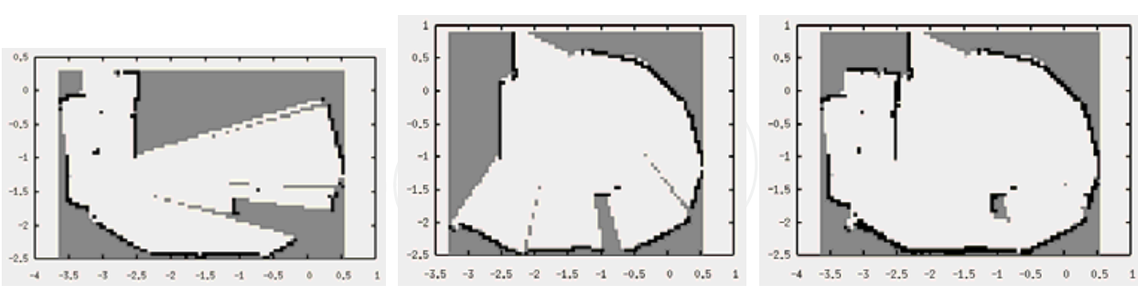
component of 3D reconstruction based on epipolar geometry was developed in order to measure 3D positions of the objects.

Fig. 6. Detection of color markers on a mobile robot using a camera

Furthermore, geometrical structures of environments are also important since it can be used for obstacle avoidance of mobile robots. Therefore static parts of the space extracted from scan data of laser range finders (backround data) are output as an occupancy grid map that represents whether the part of the environment is occupied by the obstacles or not.

4.2 Design of the information integration part

Information fusion components are implemented for information obtained by the information acquisition parts. So we developed a position server component and a map server component to fuse information. Since it is difficult to determine how many sensor nodes are connected to the server components, DynamicPorts which realize dynamic addition and deletion of data ports were also implemented. Fig. 7 shows an integration of occupancy grid maps obtained by two laser range finders. Fig. 7 (a) and (b) are maps from laser range finder 1 and 2, respectively, and Fig. 7 (c) is an integrated map.



(a) (b) (c)

Fig. 7 Integration of occupancy grid maps in the map server. (a) (b) map from each laser range finder, (c) integrated map. The while, black and gray cells denote free (no obstacles), obstacle and unknown regions, respectively. The units of x and y are in meters.

Moreover, database components for position and map information were developed by using PostgreSQL.

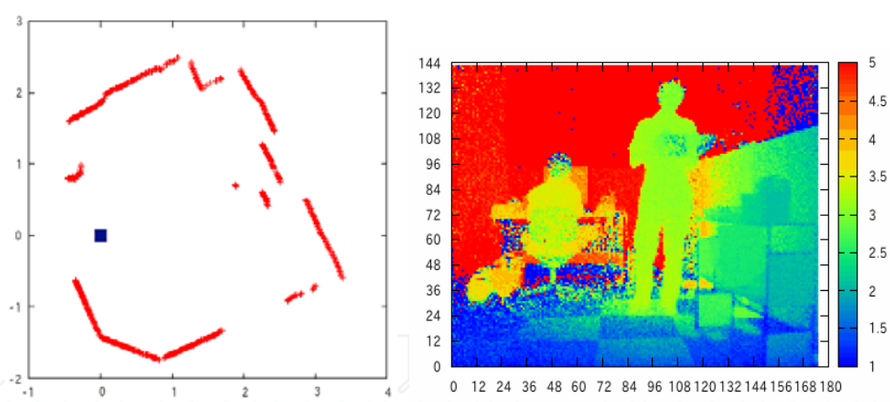
4.3 Utility components

We also developed some components which were considered to be useful for component based system integration. These components are listed in Table 4.

Component	Input	Output	Operation
Console input	-	Data from console	Get data from console
Console output	Data to console	-	Output data to console
Data viewer	Data to be plotted	-	Visualize data
Coordinate transformation	Data to be transformed, transformation parameters	Transformed data	Perform coordinate transformation
Data conversion	Data to be converted	Converted data	Convert data type, e.g. from int to double, etc.
Calibration	Set of corresponding points	Calibration parameters	Calibration support (camera, laser range finder)

Table 4. Utility components

Some of them are implemented for efficient operation of the system. These components include coordinate transformation components, data conversion components, input/output components and so on. Fig. 8 shows results of visualization of sensor data, scan data from a laser range finder component and a depth image obtained from a range imaging camera component, by using the developed viewer component. Such visualized information of outputs of a component can be used for, for example, monitoring of the space and failure detection of the component.



(a) (b)

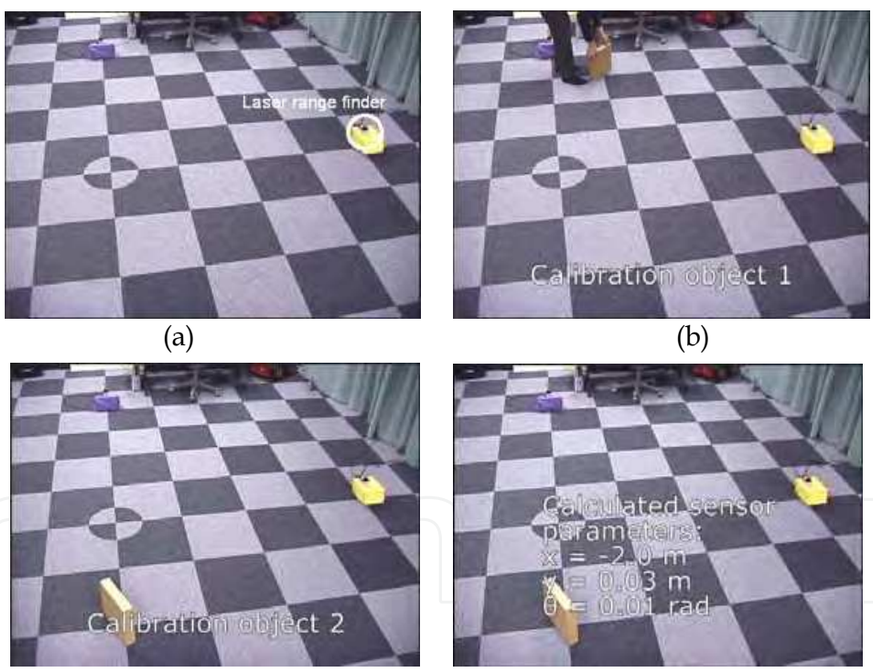
Fig. 8. Examples of visualization of sensor data (a) scan data from a laser range finder URG-04LX. The blue square shows the position of the laser range finder. The units of x and y are in meters. (b) depth map from a 3D camera SwissRanger SR-3000. The bar represents the distance from the sensor to the object in meters. The units of x and y are in pixels.

These tools also contain the components that focus on the development support of iSpace. The representative example of them is calibration support components. Although calibration is needed for proper calculation from the local coordinate system to the world coordinate system, it takes a great deal of time and effort to calibrate many sensors distributed in iSpace. The developed camera and laser range finder calibration support components provide various types of calibration methods including the conventional

manual calibration and the automated calibration based on object tracking (Sasaki & Hashimoto, 2006), (Sasaki & Hashimoto, 2009).

Fig. 9 shows manual calibration of a laser range finder. In the case of manual calibration, in order to estimate the pose of a laser range finder placed in the space (Fig. 9 (a)), a calibration object (an object which can be well detected by a laser range finder) is placed in turn on several points with known global coordinates (Fig. 9 (b) (c)). The calibration parameters are then calculated based on the positions of the calibration object in the global and local (laser range finder's) coordinate system (Fig. 9 (d)).

Fig. 10 shows an example of automated calibration of two laser range finders based on human tracking. In this experiment, the coordinate system of laser range finder 1 was considered as the reference coordinates (Fig. 10 (a)). The result of object tracking in each laser range finder is overlaid on the video image. Before calibration (Fig. 10 (b)), the positions of the person in each sensor's coordinates were stored as corresponding points. At that time, the tracked positions of two laser range finders were not coincident since the pose of laser range finder 2 was unknown. The calibration process was then performed based on the set of corresponding points (Fig. 10 (c)). As a result, almost the same tracked positions were obtained in laser range finder 1 and 2. By fusing these measurements, more reliable estimations could be acquired (Fig. 10 (d)).



(a) (b) (c) (d)

Fig. 9. Manual calibration of laser range finder using a calibration object (a) arrangement of a laser range finder, (b) (c) placement of a calibration object, (d) calculated calibration parameters

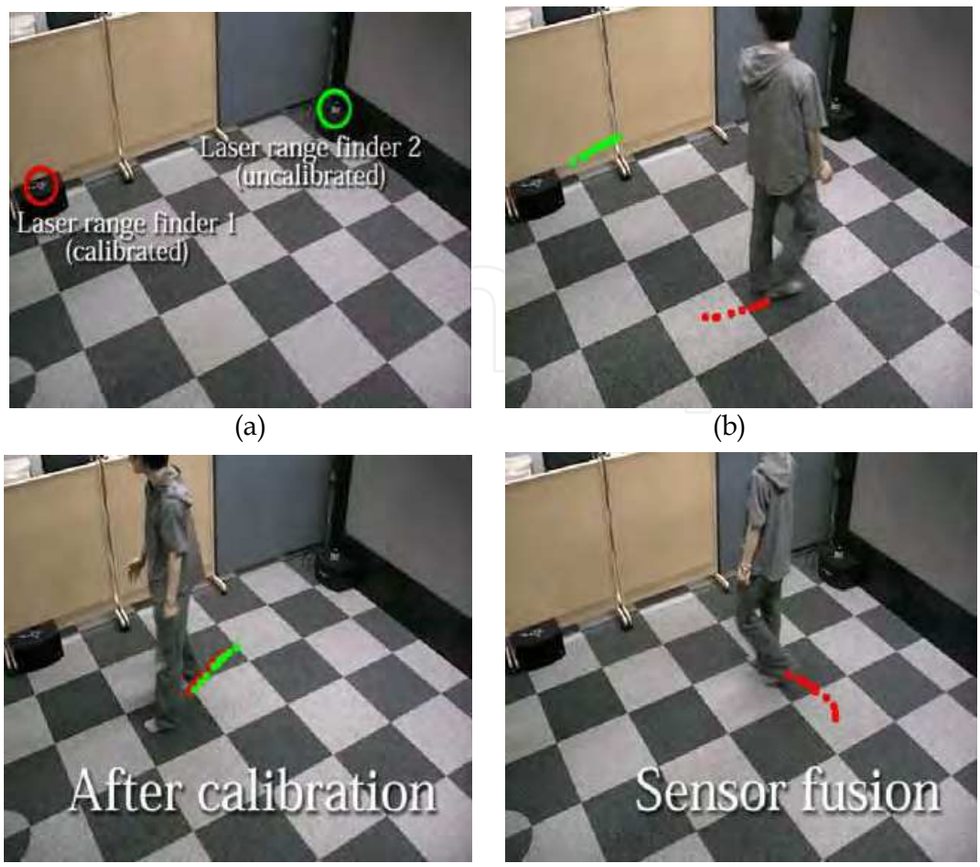


Fig. 10. Automated calibration of two laser range finders based on a calibration object (a) arrangement of laser range finders, (b) tracked positions of a person before calibration, (c) tracked positions after calibration, (d) result of sensor fusion.

4.4 Mobile robot navigation components

In addition to a mobile robot component, localization, mapping and path planning and obstacle avoidance components were developed as mobile robot navigation components. In our mobile robot localization method, the position of the mobile robot measured by distributed sensors in iSpace and the mobile robot on-board sensor data (wheel encoder) are fused using EKF (Extended Kalman Filter) to minimize the position error. To find the best way for the robot to move through the space towards a goal (path planning), we use the Field D* method (Ferguson & Stentz, 2005). Moreover, in order for the robot to follow the calculated path and at the same time avoid bumping into obstacles, the Dynamic Window Approach (DWA) (Fox et al., 1997) is used as a local control algorithm. Table 5 shows the developed mobile robot navigation components.

Component	Input	Output	Method
Localization	On-board sensor data, position information from iSpace	Robot pose	Extended Kalman filter
Path planning and obstacle avoidance	Current pose, occupancy grid map, goal position	Control input	Dynamic Window approach, Field D*
Mobile robot platform	Control input	Encoder data	-

Table 5. Mobile robot navigation components

5. Experiments

5.1 Component based system integration

Fig. 11 shows the procedure of system integration based on RT components. Building a map of the environment by using laser range finders is illustrated in the figure. First, the components needed for the designed system are selected from the component repository and the data (input and output) ports are connected. Here a laser range finder component, a mapping component and a map server component are picked up and connected (Fig. 11 (a) (b)). These components are then activated (Fig.11 (c)) and the tasks are executed: each component receives data from the input ports, processes the data and sends the obtained results to the output ports. In this case, the laser range finder component gets scan data, the mapping component extracts a map of the space from the scan data and the map is stored in the map server component (Fig.11 (d)). As discussed above, one of the major advantages of component based integration is scalability and flexibility of the developed system. By adding second laser range finder and mapping components, the observation area can readily be expanded (Fig. 11 (e) (f)). Moreover, the system can be modified by changing the connection of the components. For example, the output of a laser range finder component can be monitored by connecting it to a viewer component (Fig. 11 (g) (h)).

5.2 Development of mobile robot navigation system

We also present a mobile robot navigation system based on the developed components as an application. The configurations of components for mobile robot navigation in iSpace and on a mobile robot are shown in Fig. 12 and Fig. 13, respectively. The information obtained by distributed devices is sent to the corresponding information servers. The information servers fuse the information and store it to the database. A mobile robot moves to a goal point that is sent from an application or a user while getting necessary information from the databases and onboard sensors.

Fig. 14 shows the experimental environment. In this experiment, a mobile robot moved from the point (-1.5, 1), and made the rounds of three points (-1, 0), (1, -1) and (1.5, 1) and returned to the point (-1, 0). In the room, two laser range finders (LRF 1 and 2) were placed at around (0.5, -2) and (2.5, 0), respectively, and sent the observed map information to the map server. These laser range finders were calibrated by using calibration components in advance. Moreover, a 3D ultrasonic positioning system was installed in the environment. This system involves 96 ultrasonic receivers installed on the ceiling.

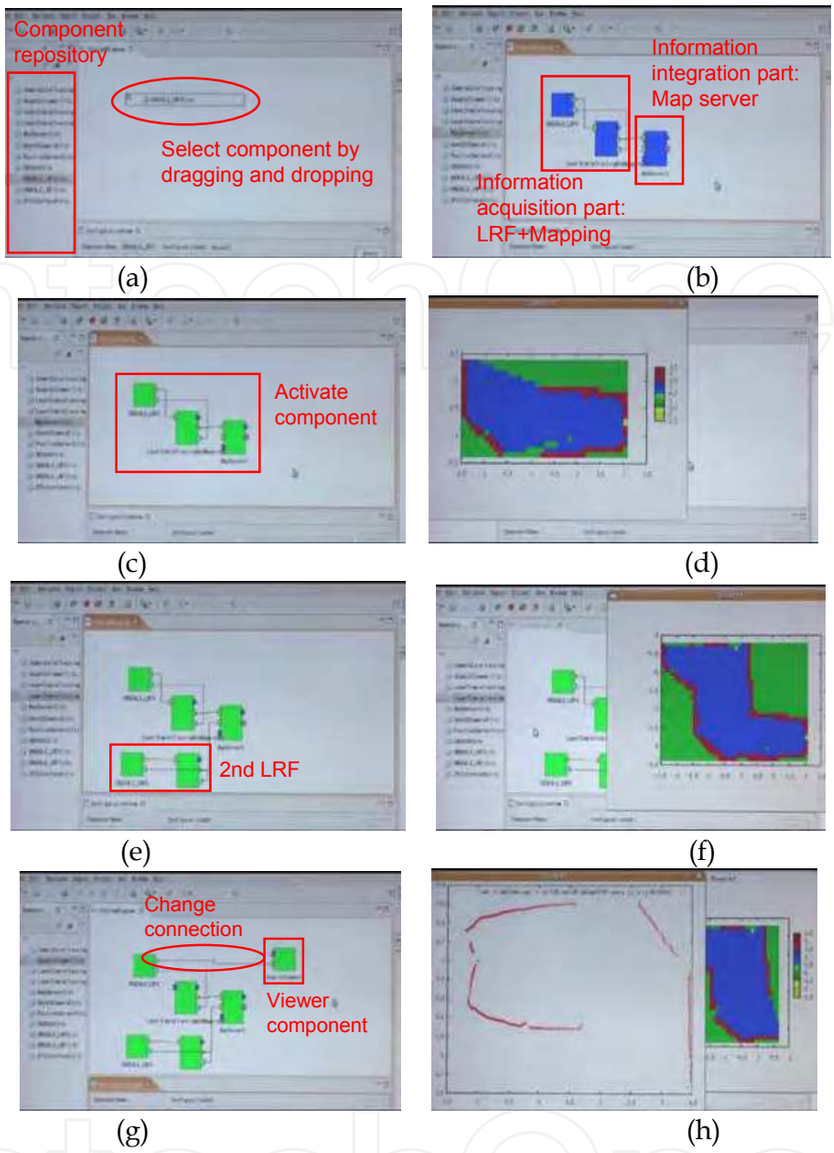


Fig. 11. Map building using laser range finders (a) selection of components from the repository, (b) connection of data ports, (c) activation of components, (d) built map of the space, (e) addition of a second laser range finder, (f) map from two laser range finders, (g) modification of the system , (h) raw sensor data

This system can measure the three dimensional position of an ultrasonic transmitter attached on the top of the mobile robot to an accuracy of 20-80 millimeters using triangulation. So the mobile robot could get the position and map information from the corresponding servers. The mobile robot also had a laser range finder and could detect obstacles around the robot. This means

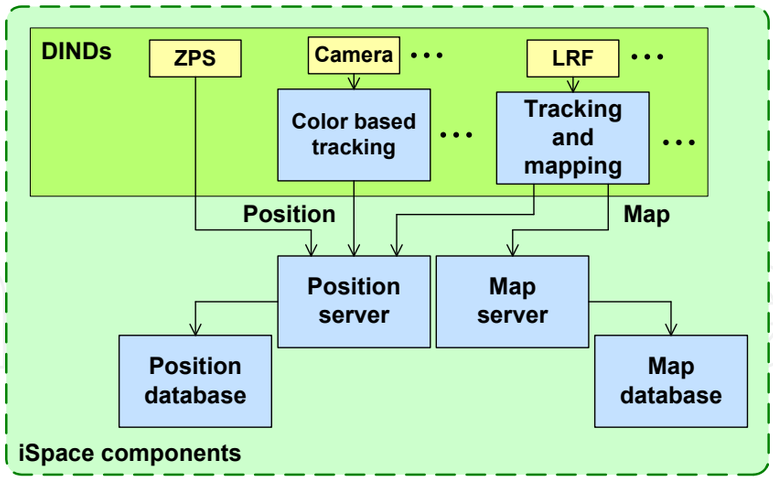


Fig. 12. Component configuration of iSpace

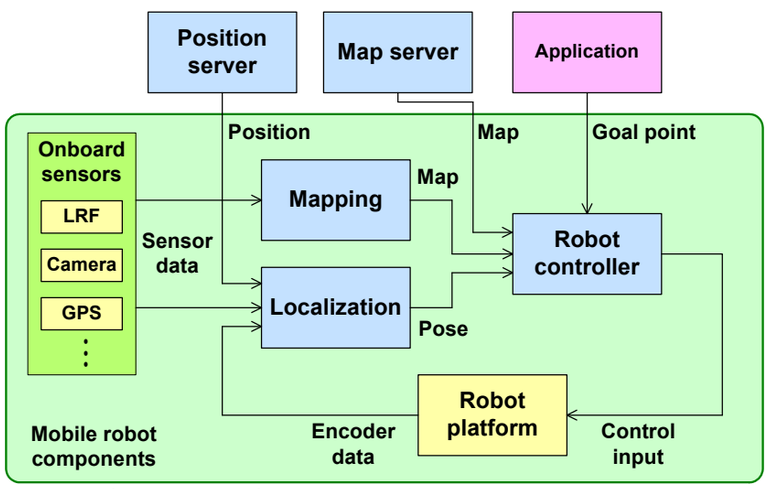


Fig. 13. Component configuration of a mobile robot

that, in this experiment, the 3D ultrasonic positioning system and laser range finders were used as sensors - for localization of mobile robot and map building, respectively.
On the other hand, the mobile robot has a laser range finder to detect obstacles around the robot. For estimating the position of the robot by the 3D ultrasonic positioning system, an ultrasound transmitter is installed on the top of the mobile robot. It is also equipped with a wireless network device to communicate with iSpace. Fig. 15 shows occupancy grid maps of the environment obtained by distributed laser range finders. Since the observable areas of the distributed laser range finders cannot cover the whole space, for example, an obstacle placed around (-0.3, 0.7) is not observed. Therefore, the mobile robot has to detect obstacles in the region by using onboard sensors.
Fig. 16 and 17 show the result and the snapshots of mobile robot navigation. We can find that the mobile robot planed the path by using the map from iSpace and passed through first three subgoals (-1, 0), (1, -1), (1.5, 1). Moreover, on the way from (1.5, 1) to (-1, 0) where the global map was not given, the mobile robot built the local map based on data from the onboard laser range finder and reached the goal point successfully.

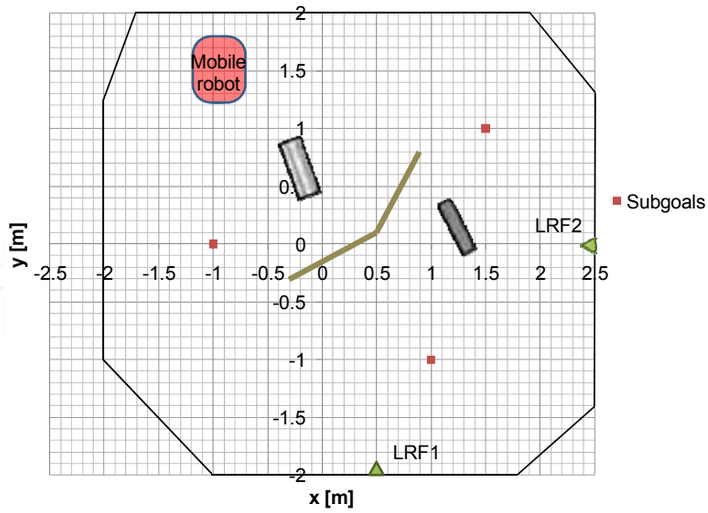


Fig. 14. Experiment environment

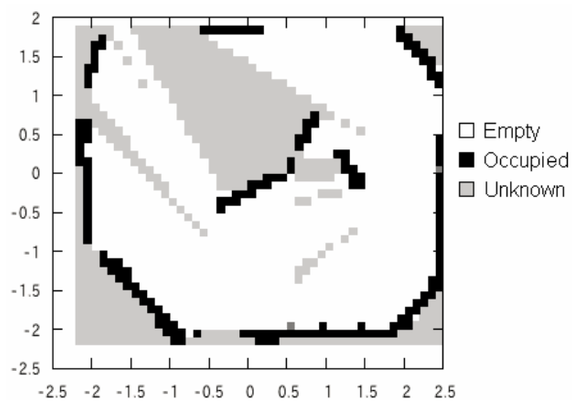


Fig. 15. Occupancy grid maps obtained by distributed laser range finders. The units of x and y are in meters.

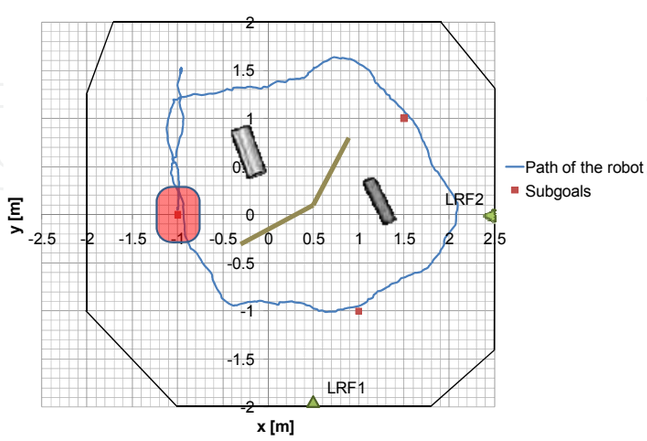
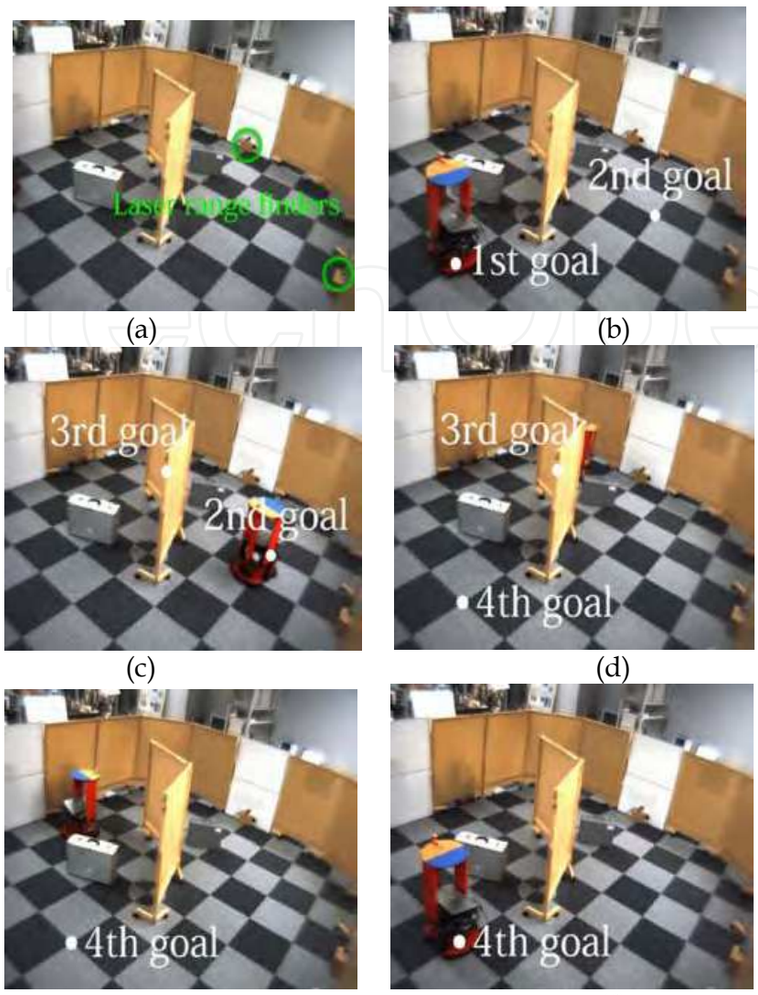


Fig. 16. Result of mobile robot navigation



(a) (b) (c) (d) (e) (f)

Fig. 17 Snapshot of mobile robot navigation experiment (a) placement of laser range finders, (b)-(d) passage of the first, second and third subgoals, (e) avoidance of a previously undetected obstacle, (f) passage of the fourth subgoal

6. Conclusion

Intelligent robot systems are developed by integration of mechatronics and software technologies. However, the systems are getting more complicated since the cooperation of various types of robots is necessary to realize advanced services for users. Therefore, the system integration becomes an important issue. In order to realize a flexible and scalable system, Intelligent Space (iSpace) is implemented using RT (robot technology) middleware. First we discussed the component design of the information acquisition function and the information integration function. The information acquisition part consists of sensor components and data processing components whereas the information integration part is composed of fusion components and database components. The developed components were then introduced and the operations of them were demonstrated. As an application, a mobile robot navigation system which can utilize information obtained from both

distributed and onboard sensors is developed. For future work we will develop sensor and processing components as well as application and actuator components to provide iSpace platform that can realize various types of services to users.

7. References

Ando, N.; Suehiro, T.; Kitagaki, K.; Kotoku, T. & Yoon, W.-K. (2005). RT-Middleware: distributed component middleware for RT (Robot Technology), *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems,* pp.3555-3560, ISBN 0-7803-8912-3, Alberta Canada, Aug. 2005.

Brooks, A.; Kaupp, T.; Makarenko, A.; Williams, S. & Oreback, A. (2007). Orca: a component model and repository, In: *Software Engineering for Experimental Robotics* (*Springer Tracts in Advanced Robotics 30),* Brugali, D. (Ed.), pp.231-251, Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, ISBN 978-3-540-68949-2.

Broxvall, M. (2007). A middleware for ecologies of robotic devices, *Proceedings of the First International Conference on Robot Communication and Coordination,* 30 (1)-(8), ISBN 978-963-9799-08-0, Athens, Greece, Oct. 2007.

Brscic, D. & Hashimoto, H. (2006). Tracking of objects in Intelligent Space using laser range finders, *Proceedings of the 2006 IEEE International Conference on Industrial Technology,* pp.1723-1728, ISBN 1-4244-0726-5, Mumbai, India, Dec. 2006.

Cook, D. J. & Das, S. K. (2004). *Smart Environments: Technologies, Protocols, and Applications (Wiley Series on Parallel and Distributed Computing),* Wiley-Interscience, ISBN 0-471-54448-7, USA.

Ferguson, D. & Stentz, A. (2005). The Field D* algorithm for improved path planning and replanning in uniform and non-uniform cost environments, *Technical Report CMU-RI-TR-05-19,* Robotics Institute, Jun. 2005.

Fox, D.; Burgard, W. & Thrun, S. (1997). The dynamic window approach to collision avoidance, *IEEE Robotics and Automation Magazine,* Vol.4, No.1, (Mar. 1997) pp.23-33, ISSN 1070-9932.

Gerkey, B.; Vaughan, R. T. & Howard, A. (2003). The Player/Stage project: tools for multi-robot and distributed sensor systems, *Proceedings of the 11th International Conference on Advanced Robotics,* pp.317-323, Coimbra, Portugal, Jul. 2003.

Hagita, N. (2006). Communication robots in the network robot framework, *Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision,* pp.1-6, ISBN 1-4244-0341-3, Singapore, Dec. 2006.

Jackson, J. (2007). Microsoft Robotics Studio: a technical introduction, *IEEE Robotics and Automation Magazine,* Vol.14, No.4, (Dec. 2007) pp.82-87, ISSN 1070-9932.

Johanson, B.; Fox, A. & Winograd, T. (2002). The Interactive Workspaces project: experiences with ubiquitous computing rooms, *IEEE Pervasive Computing,* Vol.1, No.2, (Apr.-Jun. 2002) pp.67-74, ISSN 1536-1268.

Kim, J.-H.; Lee, K.-H.; Kim, Y.-D.; Kuppuswamy, N. S. & Jo, J. (2007) Ubiquitous robot: a new paradigm for integrated services, *Proceedings of the 2007 IEEE International Conference on Robotics and Automation,* pp.2853-2858, ISBN 1-4244-0601-3, Rome, Italy, Apr. 2007.

Koide, Y.; Kanda, T.; Sumi, Y.; Kogure, K. & Ishiguro, H. (2004). An approach to integrating an interactive guide robot with ubiquitous sensors, *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.3, pp.2500–2505, ISBN 0-7803-8463-6, Sendai, Japan, Sep.-Oct. 2004.

Lee, J-H. & Hashimoto, H. (2002). Intelligent Space - concept and contents, *Advanced Robotics,* Vol.16, No.3, (Apr. 2002) pp.265-280, ISSN 0169-1864.

Lee, J.-H. & Hashimoto, H. (2003). Controlling mobile robots in distributed intelligent sensor network, *IEEE Transaction on Industrial Electronics,* Vol.50, No.5, (Oct. 2003) pp.890-902, ISSN 0278-0046.

Mizoguchi, F.; Ohwada, H.; Nishiyama, H. & Hiraishi, H. (1999). Smart office robot collaboration based on multi-agent programming, *Artificial Intelligence,* Vol.114, No.1-2, (Oct. 1999) pp.57-94, ISSN 0004-3702.

Mizukawa, M.; Sakakibara, S. & Otera, N. (2004). Implementation and applications of open data network interface 'ORiN', *Proceedings of the SICE 2004 Annual Conference,* Vol.2, pp.1340-1343, ISBN 4-907764-22-7, Sapporo, Japan, Aug. 2004.

Mori, T.; Hayama, N.; Noguchi, H. & Sato, T. (2004). Informational support in distributed sensor environment sensing room, *Proceedings of 13th IEEE International Workshop on Robot and Human Interactive Communication,* pp.353-358, ISBN 0-7803-8570-5, Kurashiki, Japan, Sep. 2004.

Mynatt, E. D.; Melenhorst, A.-S.; Fisk, A.-D. & Rogers, W. A. (2004). Aware technologies for aging in place: understanding user needs and attitudes, *IEEE Pervasive Computing,* Vol.3, No.2, (Apr.-Jun. 2004) pp.36-41, ISSN 1536-1268.

Nishida, Y.; Hori, T.; Suehiro, T. & Hirai, S. (2000). Sensorized environment for self-communication based on observation of daily human behavior, *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems,* Vol.2, pp.1364-1372, ISBN 0-7803-6348-5, Takamatsu, Japan, Nov. 2000.

OMG (2008). Robotic Technology Component Specification Version 1.0, OMG Document Number: formal/2008-04-04, Apr. 2008.

Sasaki, T. & Hashimoto, H. (2006). Camera calibration using mobile robot in Intelligent Space, *Proceedings of SICE-ICASE International Joint Conference 2006,* pp.2657-2662, ISBN 89-950038-4-7, Busan, Korea, Oct. 2006.

Sasaki, T. & Hashimoto, H. (2009). Calibration of laser range finders based on moving object tracking in Intelligent Space, *Proceedings of the 2009 IEEE International Conference on Networking, Sensing and Control,* pp.620-625, ISBN 978-1-4244-3491-6, Okayama, Japan, Mar. 2009.

Sgorbissa, A. & Zaccaria, R. (2004). The artificial ecosystem: a distributed approach to service robotics, *Proceedings of the 2004 IEEE International Conference on Robotics and Automation,* Vol.4, pp.3531-3536, ISBN 0-7803-8232-3, New Orleans, LA, USA, Apr.-May 2004.

Utz, H.; Sablatnog, S.; Enderle, S. & Kraetzschmar, G. (2002). Miro - middleware for mobile robot applications, *IEEE Transaction on Robotics and Automation,* Vol.18, No.4, (Aug. 2002) pp.493-497, ISSN 1042-296X.

Mechatronic Systems Applications

Edited by Annalisa Milella Donato Di Paola and Grazia Cicirelli

ISBN 978-953-307-040-7
Hard cover, 352 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

Mechatronics, the synergistic blend of mechanics, electronics, and computer science, has evolved over the past twenty five years, leading to a novel stage of engineering design. By integrating the best design practices with the most advanced technologies, mechatronics aims at realizing high-quality products, guaranteeing at the same time a substantial reduction of time and costs of manufacturing. Mechatronic systems are manifold and range from machine components, motion generators, and power producing machines to more complex devices, such as robotic systems and transportation vehicles. With its twenty chapters, which collect contributions from many researchers worldwide, this book provides an excellent survey of recent work in the field of mechatronics with applications in various fields, like robotics, medical and assistive technology, human-machine interaction, unmanned vehicles, manufacturing, and education. We would like to thank all the authors who have invested a great deal of time to write such interesting chapters, which we are sure will be valuable to the readers. Chapters 1 to 6 deal with applications of mechatronics for the development of robotic systems. Medical and assistive technologies and human-machine interaction systems are the topic of chapters 7 to 13.Chapters 14 and 15 concern mechatronic systems for autonomous vehicles. Chapters 16-19 deal with mechatronics in manufacturing contexts. Chapter 20 concludes the book, describing a method for the installation of mechatronics education in schools.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Takeshi Sasaki and Hideki Hashimoto (2010). Design and Implementation of Intelligent Space: a Component Based Approach, Mechatronic Systems Applications, Annalisa Milella Donato Di Paola and Grazia Cicirelli (Ed.), ISBN: 978-953-307-040-7, InTech, Available from: http://www.intechopen.com/books/mechatronic-systems-applications/design-and-implementation-of-intelligent-space-a-component-based-approach

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen