Providing Services Using Network-Based Humanoids in a Home Environment

KangGeon Kim, *Member*, IEEE, Young-Su Cha, Jung-Min Park, Ji-Yong Lee and Bum-Jae You, *Member*, IEEE

**Abstract — ***This paper describes the development of network-based humanoids to provide services in the home environment. For successful service in more intelligent and varied environments, various robot sub-systems need to be coordinated effectively. Thus, this paper also introduces a coordinated framework which makes robot-human interaction while executing various tasks by means of various robots. Using a task script, an operator can easily describe tasks and regulate actions of the sub-systems while a robot is performing the task. Furthermore, the control system of a robot must operate efficiently to ensure a coordinated robot system, so the realization of an IEEE-1394 real-time distributed control system and a motor controller for a humanoid robot are introduced. Many algorithms have been developed, and the following technologies are described herein: autonomous biped walking; real-time modification of collision-free paths; and interaction ability with humans and environments, such as face, voice, or object recognition. The results of the demonstration show that humanoids can execute tasks efficiently and are suitable to provide services in human environments, such as restaurants or homes.[1].*

Index Terms — Home service robot, Network-based humanoids, Robot control system, Service providing.

I. INTRODUCTION

Robots will soon become a more familiar part of everyday life. In fact, the robot will serve as an important element of the home environment, much like a computer, car, or refrigerator. Unlike industrial robots, robots which are designed to assist in household chores must be usable in a variety of household environments so that they can contribute to improving the quality of life by offering humans relief from burdensome tasks [1], [2]. The humanoid robot will have its own intelligence beyond simple walking and moving activities [3], [4]. The ideal robot is aimed to be developed which will be able to perceive human beings' intentions to communicate, to express feelings, and to co-exist with humans.

[1] This work was supported by KIST [Development of Intelligent Task-Execution Technology for Robots using Spatial Interaction].

K. Kim is with Interaction and Robotics Research Center, Korea Institute of Science and Technology, Seoul, Korea (e-mail: danny@kist.re.kr).

Y. Cha is with Interaction and Robotics Research Center, Korea Institute of Science and Technology, Seoul, Korea (e-mail: givemong@kist.re.kr).

J. Park is with Interaction and Robotics Research Center, Korea Institute of Science and Technology, Seoul, Korea (e-mail: pjm@kist.re.kr).

J. Lee is with Interaction and Robotics Research Center, Korea Institute of Science and Technology, Seoul, Korea (e-mail: wisrage@kist.re.kr).

B. You is with Interaction and Robotics Research Center, Korea Institute of Science and Technology, Seoul, Korea (e-mail: ybj@kist.re.kr).

Contributed Paper
Manuscript received 10/14/11
Current version published 12/27/11
Electronic version published 12/27/11.

Fig. 1. Network-based humanoids, MAHRU-Z and MAHRU-M

The majority of the present service robots are mobile robots. This is because they offer stable services in indoor environments. Humanoid robots, however, walk with their legs, and this makes the humanoid's manner of movement more delicate and similar to that of people.

Several humanoids have been developed, such as ASIMO of Honda Co. Ltd. Japan [5], HRP of AIST Japan [6], JOHNNIE of TUM Germany [7], HUBO of KAIST Korea [8], and so on. These kinds of robots are still used for performance, but they have not yet been applied in actual daily life.

The Korea Institute of Science and Technology has recently developed a network-based humanoid robot, MAHRU, which is capable of having intelligence similar to that of humans (Fig. 1). The feature that sets MAHRU apart is its cognitive ability, which allows it to have various facial recognition and artificial intelligence functions through an external computer system connected via a network. Its system structure also facilitates easy updating of various application services through the network, so the same robot can be used to provide improved services depending on its environment and commands.

Adapting a communication protocol is an important part of the humanoid control system. For the stable operation of humanoid robots, its many motors and sensors require fast feedback by high-speed communication. IEEE-1394, also known as FireWire, at over 100 Mbps, is one of the best choices for a faster field-bus in a real-time distributed robot control system. IEEE-1394 is a well-known protocol as a high-performance serial bus, which was first standardized by IEEE in 1995 [9]. IEEE-1394 is used primarily in multimedia devices and office automation. Moreover, IEEE-1394 has been applied to MAHRU as a communication protocol in a real-time distributed control system [10]. This robot needs to control a total of 35 motors. The high-data rate of IEEE-1394 provides enough time to control a number of motor controllers and to transmit mass data to a humanoid robot. In this control system, the transmission time is very short, under 100 µs, to send data to all humanoid motors. The realization of an IEEE-1394 real-time distributed control system and a motor controller for a humanoid robot are reported.

For the robot to be able to execute tasks without human intervention, a sequence of terminal actions is automatically generated. R. Alami et al. [11] use a data set consisting of action and cost, and actions sequences that minimize the total cost are generated through an HTN (Hierarchical Task Networks) planner. C. Galindo et al. [12] use semantic maps. They use lisp-like expressions and deal with the case in which information is obtained partially and the case in which a solution exists but it is not identified. In addition, J. Liu [13] uses task grammar. Given a starting state (high-level state), it is broken down into a low-level state using task grammar. Thus, finding a desirable plan for a high-level composite operation command corresponds to the recursively replacement of that operation by the right-hand side of a task grammar rule.

This paper describes the research results about the framework to provide services in a home environment by multiple robots. The core technology is to coordinate the individually developed services without any conflicts, such as biped walking and balancing; self-position recognition and path planning; and recognition of faces, voices, and objects. Also, the problems are investigated which occur during the process of developing a humanoid service robot, such as false-recognition of an object, malfunction, and so on.

The remainder of this paper is organized as follows. Section 2 introduces a cooperative framework for the humanoid robot. Section 3 introduces a real-time distributed humanoid robot control system based on IEEE-1394. Section 4.1 describes movement and path planning. Section 4.2 describes perception in operation, such as object pose estimation and face tracking for service providing. Section 5 explains error handling and safety, and Section 6 presents the experiment for coordinated task execution.

II. Cooperative Framework for a Humanoid Robot

A block diagram of the coordinated task execution framework is shown in Fig. 2. The framework can help various robots to cooperate with each other and reach a common goal.

Using a task script, scenarios can be scheduled for the following parts: the walking/mobile part, the perception part, and the robotic manipulator. Each part can be controlled by a task coordinator, which is essential to arrange and manage the system effectively in various situations. With the proposed framework, diverse tasks are performed, such as moving to an object, picking up an object, finding a specific person or exchanging an object, through the control of different sub-systems. In the task execution process, the robot sub-system is commanded to perform autonomously, and it sends feedback to the task coordinator.

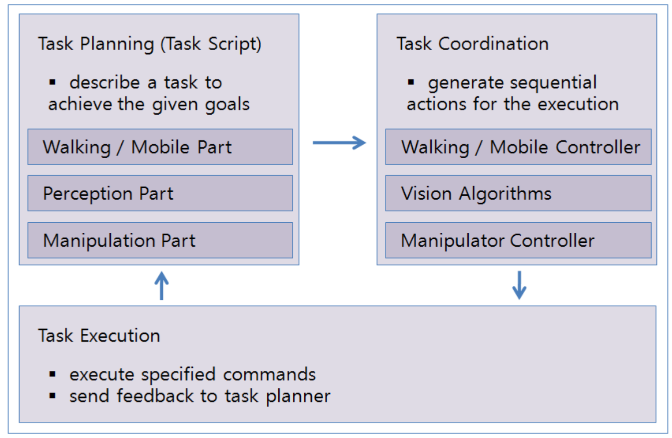


Fig. 2. Framework for coordinated task execution

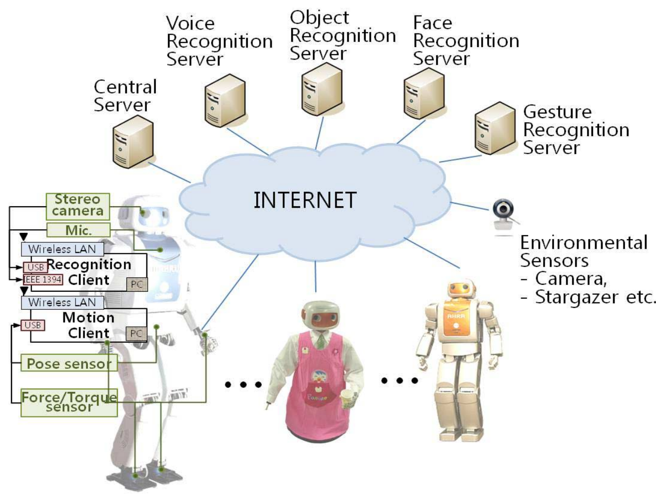


Fig. 3. A Network-based humanoid system

A. *Network-Based Framework*

A network-based humanoid system is composed of various forms of humanoids, external service servers to provide various robotic services for connected humanoids, and environmental sensors, as shown in Fig. 3.

The humanoid equipped with an internal control system is mainly responsible for such aspects as motor control for each robot joint, motion control for whole body control, and the transmission of sensory data (for example image data, voice data, force/torque data, and motion data) from various sensors to external service servers. The external service servers consist of several service servers to provide features such as perception, human-friendly interaction, and intelligence to multiple network-based humanoids by using multi-tasking capability. The central server is responsible for real-time data transmission and action command generation after reasoning for responses from other service servers. In addition, it chooses the proper server to provide service to a network-based humanoid and sends requests for service to selected servers. An object recognition server and a face recognition server identify three-dimensional objects and the faces of human beings, respectively. A voice recognition server recognizes human voices and words.

To efficiently realize the client and server architecture in a distributed environment, a network-based humanoid system should be developed independently and integrated without difficulty. CORBA is the distributed object middleware introduced by ORG [14]. Various CORBA-based service frameworks for a distributed robotic control system have been proposed [15]. CORBA makes it possible to integrate several service software components in a distributed environment and provides mutual operation abilities, such as location transparency, program transparency, and hardware and software, as well as network transparency. All servers are integrated with the CORBA middleware, while service components in each server are modularized by using the CORBA interface.

Services of humanoids can be performed either inside the humanoid robot or on the external service server in the network-based humanoid system. Services integrating vision in feedback control loops, such as whole body control and visual servoing must be performed inside the robot because they require high-speed feedback and are subject to strict time constraints. However, high-level processing services, such as face recognition, object recognition, and voice recognition can be executed in the external service servers. Therefore, the robotic services are divided into two categories: reactive service, which is executed inside the robot, and network service, which is performed on the external service server via a network.

B. Scheduling the Task

The execution of a coordinated task requests a technique for the scheduling of the robot actions. To make this possible in various environments and situations, an XML script is used to describe and plan the tasks efficiently. An individual task makes up the whole operation, and the task script enables this task to be performed with a simple description. There are advantages to using this task script. It makes it possible to construct the actions of the desired task for the users, and it can be changed by a simple description. The information obtained by the robot, rather than a static value, is applied in real time. Moreover, the execution sequence is clearly determined using the condition of each task. The main components of the task script are the task, the action, and the condition. These are defined as follows:

Task A task is a unit to execute a purpose. Individual tasks make up the entire task, which makes it possible for the robot to perform flexibly in various environments. Each task involves an action and a condition.

Action An action is the unit operation that is performed when the task is run. Action types are divided into manipulation, walking/mobility, and perception, depending on the use of each sub-system. An action is executed according to the parameters which are described for each sub-system.

Condition After the individual task is executed, the task to be performed next is determined according to the condition. This plays the role of deciding the direction of the flow. For example, the condition decides which task is executed when the former task has succeeded or, when it has failed, it decides whether to terminate the entire task or to execute the same task repeatedly.

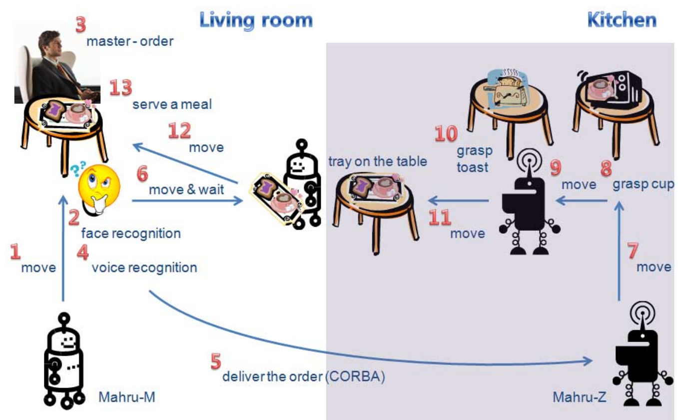


Fig. 4. Demonstration scenario for coordinated system

The demonstration process is illustrated in Fig. 4. It shows the flow of the task. The real parameter, which includes the robot operation information, is given in each action. The robot can then move on using the parameter value already determined or the one obtained during the operating process, and the parameter reflects the newly obtained information simultaneously. On the other hand, if each task is not performed successfully, the robot can terminate the operation according to the condition or can carry out the operation repeatedly.

III. Humanoid Control System based on IEEE-1394

A. Hardware Platform, MAHRU

The humanoid robot MAHRU-Z, shown in Fig. 5, was developed on a real-time distributed system using IEEE-1394. The following parts describe specifications of the humanoid robot MAHRU-Z. The robot has 1500 mm height and 60 kg weight, including batteries. This robot has a total of 35 degrees of freedom (DOF): 12 DOF in two legs, 14 DOF in two arms, 6 DOF in two hands, 2 DOF in the neck, and 1 DOF in the waist. They are operated by DC servo motors through harmonic drive gears to provide compactness and better controllability. The ankles and wrists are equipped with 6-axis force/torque sensors for elaborated control such as walking and visual servoing.

The control system consists of one main control PC and several motor controllers as shown in Fig. 6. The main control PC comprises a 1.8 GHz Intel CPU and an industrial single-board computer (SBC) which is mini-ITX size. Additionally, it uses the IEEE-1394 PCI card on the SBC for communication. The PCI card has three IEEE-1394 ports and includes the IEEE 1394 chip. It is able to transfer data between the 33 MHz PCI bus and the IEEE 1394 bus at 100 Mbps, 200 Mbps, and 400 Mbps. This performance exercises IEEE standard 1394a-2000 [9].

The humanoid uses nine motor controllers for the control network in MAHRU-Z. One motor controller can control a maximum of four motors. These motor controllers are adequately distributed to 3 ports to reduce the communication load from a connecting tree. The controllers include DSP, IEEE-1394 control chip, and FPGA as shown in Fig. 7. The DSP chip is a main processor in the board, and it is in charge of the control of motors. The IEEE-1394 control chip is for the IEEE 1394a-2000 communication protocol physical layer. Then, the FPGA is used for other accessory operations.

Also, the humanoid can be supplied power by external source. This uses 48 V for motor operation and 24 V for motor controllers. The power of PC uses 12 V, transformed by DC-DC converter from 24 V. At normally standing state, the continuous current is 0.7 A on 48 V and 9.8 A on 24 V. The humanoid consumes 268.8 W.

Brief specifications of this robot are shown in Table 1.

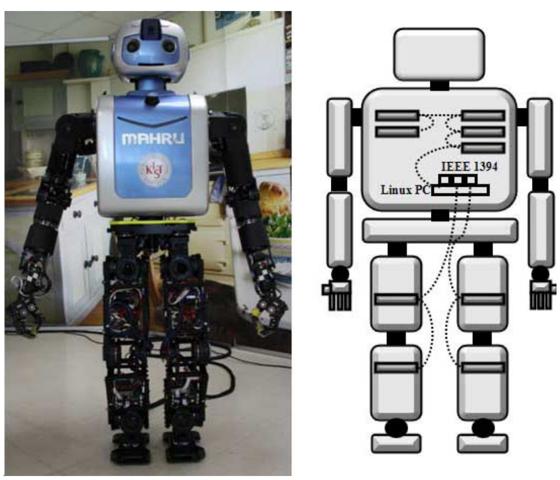


Fig. 5. MAHRU-Z based on real-time distributed system using IEEE-1394

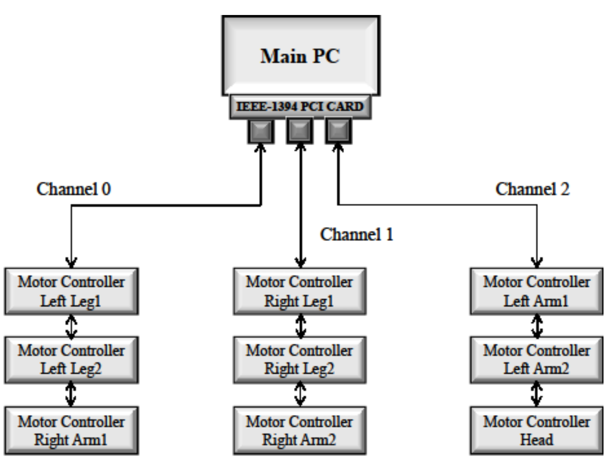


Fig. 6. Control system architecture

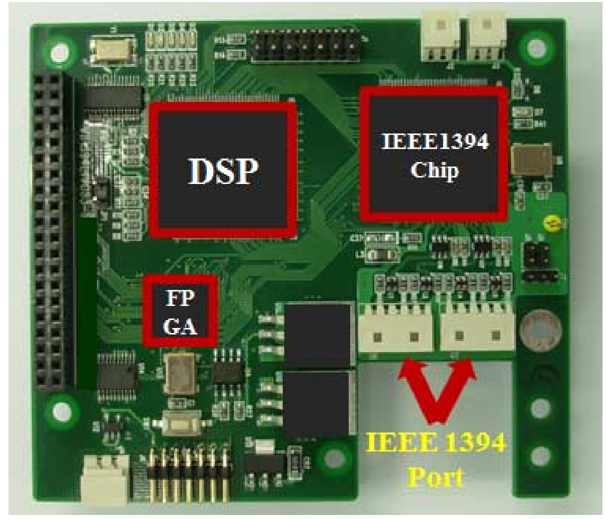


Fig. 7. IEEE-1394 motor controller board

TABLE I
MAHRU-Z SPECIFICATIONS

Height	1500 mm
Weight	60 kg(with battery)
Degree of Freedom	35 DOF Head : 2 DOF Arm : 7 DOF x 2 Leg : 6 DOF x 2 Hand : 3 DOF x 2 Waist : 1 DOF
Sensors	6-axis F/T sensors in ankle & wrist
Main PC	1.8 GHz CPU Mini-ITX SBC
I/O Communication	IEEE 1394
Operating System	Linux + Xenomai + RT FireWire
Power Supply	48 V for motor operation 24 V for PC & controller
Consuming Power	268.8 W

B. Software Platform

A software platform of the main control PC in the humanoid robot has a dual-kernel real-time operating system (RTOS), Linux and Xenomai for its obvious advantages in hard real time support in user space programs. When Xenomai modules are uploaded, it attains two domains, the primary real-time Xenomai domain and the secondary non-real time Linux domain. Most real-time tasks are executed in the primary domain. The first Xenomai module, *xeno-hal* offers this basic hardware abstraction, and the summarized RTOS core and *xeno-nucleus* provides all basic real-time services. In the execution, the native skin, *xeno-native*, for user space real-time support and posix skin, *xeno-posix*, to port a posix submissive application from real time Linux have been applied. Another skin, called Real-Time Driver Model (RTDM), *xeno-rtdm*, is provided by the real-time FireWire (RT FireWire) device driver to interface with Xenomai. The RT FireWire driver operates on top of the RTOS layer. The real-time IEEE-1394 driver core, constituting the modules *rt-pkbuff*, *rt-serv,* and *rtpc*, provides the real-time memory management, task arrangement, and interrupt management offer the high layer of the driver. The modules connect directly with Xenomai. Real-time IEEE-1394 driver execution, which consists of *rt-ieeel394* and *rt-ohcil394*, offers an IEEE-1394 character device interface to user applications. The driver execution supports open host controller interface boards [10].

C. Operation Sequence

To operate MAHRU-Z, The main control PC transmits commands to the motor controllers as shown in Fig. 8. In the control network, each controller recognizes the commands as a pre-programmed order in their DSP memory. First of all, the main PC executes to check ID to identify a connected board; at this time, the motor controllers reply with their ID numbers. On that basis, the IEEE-1394 network starts to organize a node tree. Second, the command to set the parameters of the motor controllers is transmitted to the motors. Next, the ready-state command is executed to activate the motors. To indicate the origin of the motor encoder, a home-search command is sent to the motor controllers as a final setting. After that, the network starts the control feedback loop. Periodically, data read/write commands, which are the most important part, communicate between the main control PC and the motor controllers after setting. The commands are to read the current degree of motors and to write the desired degree.

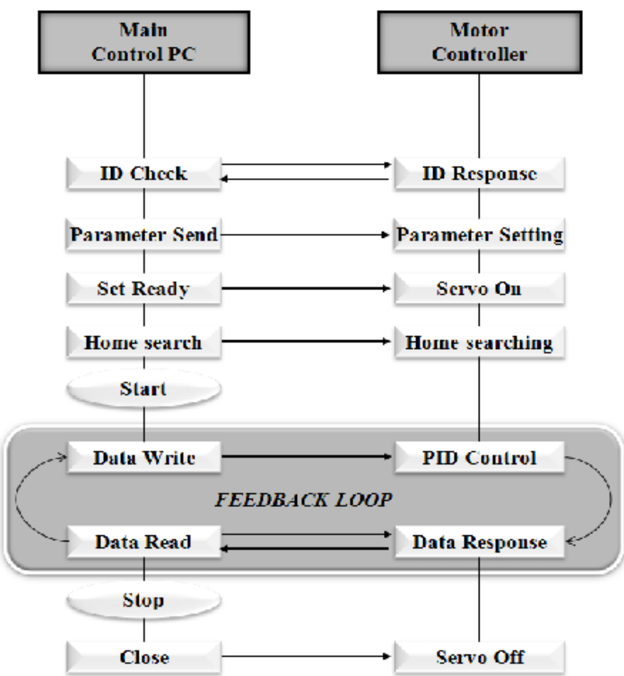


Fig. 8. Command sequence diagram

IV. INDIVIDUAL TASK FOR PROVIDING A SERVICE

A. Movement and Path Planning

The robot should be able to assess its location in order to move to a specific location to carry out its command. It plans its path, taking into consideration the location of an obstacle, and moves itself by biped walking or wheeling. To measure its location, a StarGazer sensor system is used. A landmark is printed with a pattern reflecting an infra-red light and is attached to the ceiling. The landmark plays the role of an independent coordinate system with an identity number. The StarGazer consists of an infra-red beam emitter and an infra-red image camera. Generally, the sensor system is equipped on the head of the robot. It analyzes the infra-red image of the pattern on the ceiling and calculates its position and the heading angle of the robot.

In the current method, the robot moves to an object on the assumption that the robot already knows the position of every object in the home. This movement is based on the above sensor system, so it can move to the coordinate on the already known map. When the robot is in reachable space, then it starts operating. By the experiment, the work space to be covered by the robot's arm is decided. The path is planned globally while avoiding overlapping with obstacles or other objects. When an unexpected situation is encountered, the path is replanned. The humanoid walking pattern is classified as "move forward", "move backward", "move left", "move right", "turn left", "turn right", and so on. A path planning algorithm based on the A-star algorithm is developed to design the optimal path considering different weights on moving forward and turning based on the given map.

B. Perception in Operation

To grasp an object, the robot should know the object's location and orientation. To detect the object, the object should be in the view of the robot camera. However, due to the constraints of the robot camera field of view (FOV), the robot has to stand exactly in front of the target object. To complement this weakness, the robot's head needs to be moved, but it required a lot of operating time to recognize an object, so the robot decides the angle of its head using the object information from the map.

After detecting the object in the camera image, by the transformation between the robot coordinate and the camera coordinate, the humanoid robot is able to operate in the error level of a 3 mm boundary. For example, the process of seeking and pushing an oven button with a radius of only 15 mm, is more exact due to the perfect recognition and the delicate calibration process. Here, it should be considered how the robot can execute a task when a certain object is not in its designated place but has moved to another location. The robot should decide whether to execute the task again, skip the task, or quit the task.

The object's location and orientation can be obtained by performing object pose tracking based on the model. This is done using a particle filter. This algorithm is used in many areas due to its suitable performance [16]-[19]. A conventional particle filter works as follows:

1 **Sampling,** in which a weighted particle set is transformed to an evenly weighted particle set in which particles are concentrated on special areas in accordance with the posterior density of the particles.

2 **Prediction,** in which the particles are propagated depending on the motion model that represents the movement tendency of the target object.

3 **Measurement,** in which the new posterior density (weight) is calculated by the likelihood function of each particle.

However, a conventional particle filter is not suitable for object pose tracking because the search space is considerably large and a long computational time is required to find the solution. Consequently, back projection-based sampling is introduced, which has a notable advantage in that the search space is reduced.

Back projection-based sampling reduces the search space using a depth map; if the depth information is known, one point on the image corresponds to one point in 3D space.

The back projection method is shown in Fig. 9. One point on the depth map corresponds to one point on the surface patches in 3D space. Therefore, 3D space volumes are diminished to the 3D surface patches, resulting in the reduction of the search space.

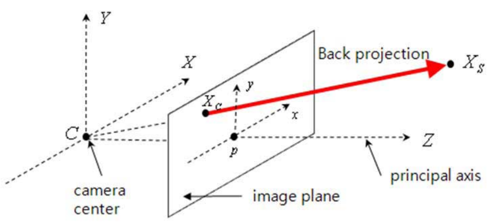
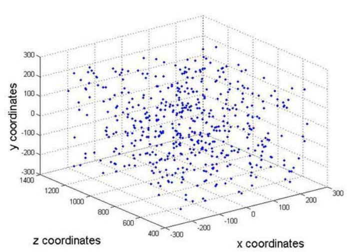
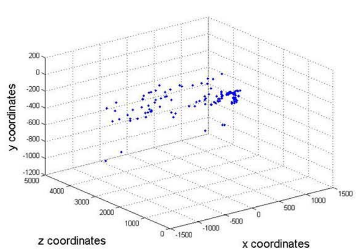


Fig. 9. Back projection. It is possible to map one point on a 2D image plane to one point in a 3D space if the depth is known; the point X_C on the image plane corresponds to the point X_S in the 3D space.



(a)



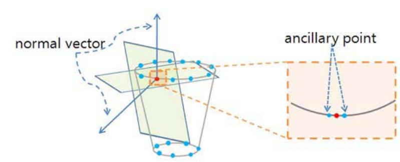
(b)

Fig. 10. Sample distribution of pseudo-random sampling and back-projection-based sampling. The particles of the back-projection-based sampling (b) are scattered in a smaller space than those created by pseudo random sampling (a) with respect to the translation.

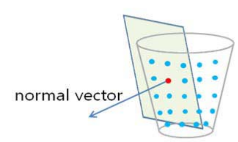
The sample distribution is shown in Fig. 10. The particles generated into 3D space directly by pseudo-random sampling are scattered in a broad area, whereas those generated through back projection are aggregated.

In the sampling step, the initial particles are generated by back-projection-based sampling. To cope with sudden changes in shape and motion and to avoid becoming stuck in local spurious minima, random particles at a rate of 10% are introduced at each iteration based on the back-projection-based sampling.

The model of the target object for the likelihood computation consists of points, and each model point is endowed with either of two attributes: an edge or a depth (Fig. 11). This occurs because the two attributes are difficult to obtain at the same position of the object in the case of a fast algorithm, which is suitable for running in real time. All points with edge attributes have two ancillary points and two normal vectors of the faces, including the model point. The ancillary points are used in calculation of the orthogonal direction of the model points after projection onto an image plane, and the normal vectors are used to determine whether a current point is visible using backface culling.



(a)



(b)

Fig. 11. Model points. (a) edge points and (b) depth points of model

The likelihood function for a posterior density, $p(X \mid Z_t)$, is constructed by two image feature: the edges and the depth. In a sample set $\{s^1,...,s^N\}$ generated from the prior density $p(X)$, the likelihood of the samples is calculated as follows.

$$p(s^i) = p(e \mid X = s^i)\, p(d \mid X = s^i) \tag{1}$$

Here, e represents the points of the edges, and d denotes the points of the depth. The measurements are calculated separately according to the attributes that serve to determine the weight of the samples.

The edge is calculated using a gradient-based mask, and the depth is obtained from a stereo camera. The measurements of the edge for sample X are calculated using the difference between the transformed edge model points and the nearest edge pixel. Similarly, the measurements of the depth point are calculated using the difference between the depth value and the depth point of the model transformed by the sample X.

When a robot hands an object to the designated person correctly, the robot should recognize the position of the designated person. To track the face which is detected and recognized by the AdaBoost [20] and PCA [21] methods, a mean-shift algorithm using bilateral filtering [22] is used due to its robustness to illumination changes.

The mean-shift algorithm is an efficient technique for object tracking [23]. Object tracking is achieved by applying it to the Bhattacharyya coefficient to find the object position.

A bilateral filter [24] is used to overcome a number of problems, such as occlusion by an object with a similar color distribution or a background with a similar color distribution when an object is tracked using color information only. The bilateral filter consists of two support maps in the depth domain and the color domain. The use of the depth domain allows for easy segmentation of the user from a scene with other people and background objects. It is quite useful under changing illumination conditions to combine the advantages of color and depth data.

V. Emergency Situations and Safety Issues

Error handling is necessary to manage any emergency situation in order for robots to live together with human beings in real life. In such a situation, unknown obstacles can hit humanoid robots and humanoid robots are able to hurt humans or damage other objects. There are many causes of malfunction:

1. When a robot fails to walk properly due to obstacles
2. When a robot fails to recognize a mark which can help it to know its own position
3. When a robot fails to find an object. In other words, when the object is not in the designated position or when the robot fails in recognition

One approach to solve this problem is that the robot stops immediately. However, biped humanoid robots generally fall over easily because their stability margin is small according to a restriction of support polygon and a high COG position. Therefore, it is necessary to improve fail-safe technology to prevent them falling over, to control the fall when falling is unavoidable, and standing by themselves in any state in addition to the walking mobility for working in a living environment.

Morisawa et al. [25] suggested an emergency stop algorithm of a walking humanoid robot. Humanoid robots need to be able to stop quickly without falling. Since the timing of emergencies is unpredictable and they can occur at any state of robots, the stopping motion must be generated in real-time.

In any case, securing the safety of human users is essential, so the safest and most stable measures need to be taken. When a robot falls down, it will suffer physical damage or be destroyed. However, if there are people near, and if there is any possibility for people to be harmed, that is the biggest problem. Regarding the integrated system configuration, the ability to judge the situation and handle errors is a very good criterion to estimate the intelligence of a robot.

In actual applications, various algorithms are executed. As the success rate of each algorithm is not guaranteed, the ability to judge the result and react in each case is needed.

VI. Experiment

A demonstration of network-based humanoid operation was performed using humanoid robots. In the system, “MAHRU-Z” is used as the humanoid robot (Fig. 12) and “MAHRU-M” is used as the mobile robot.

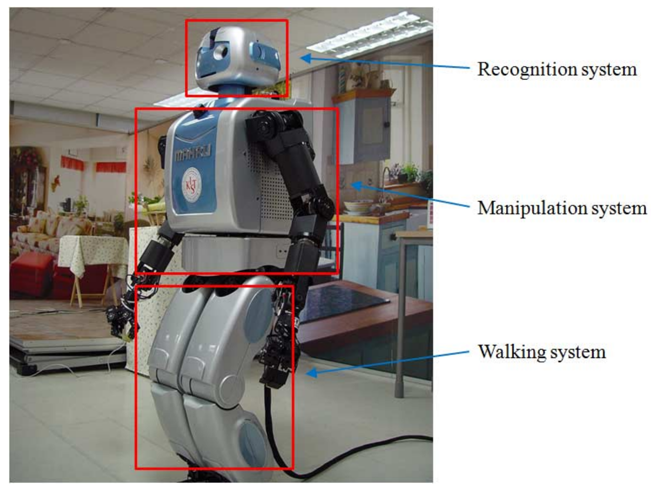


Fig. 12. MAHRU-Z, a humanoid manipulation platform

Now, the specific sequence of the task, “Service in Home Environment”, is as follows. A demonstration stage which represents a home environment was set up, and the process of the task is shown in Fig. 13. First of all, the mobile robot, MAHRU-M in the living room moved to a person to receive an order (a). The robot recognized the face and voice of the person to take the order (b). The order was delivered to the humanoid robot, MAHRU-Z in the kitchen by the network. The robot moved to the oven. The robot already knew the position of every object in home environment using a StarGazer sensor system.

Fig. 14 illustrates a landmark array used in the experiment, which is composed of 30 landmarks and covers an approximate area of 5 m × 4 m. Each landmark functions as a local coordinate system, and the transform function is used to transform each local coordinate system into the global coordinate system. The path was planned by the proposed algorithm, and the robot was able to approach to the oven.

The robot detected the location of a button (Fig. 13) (c) and pushed the button to open the door (d). The robot detected the location and orientation of a milk cup in the oven, and the robot arm grasped the milk cup exactly in the correct position (e). Real-time object localization and location estimation were possible.

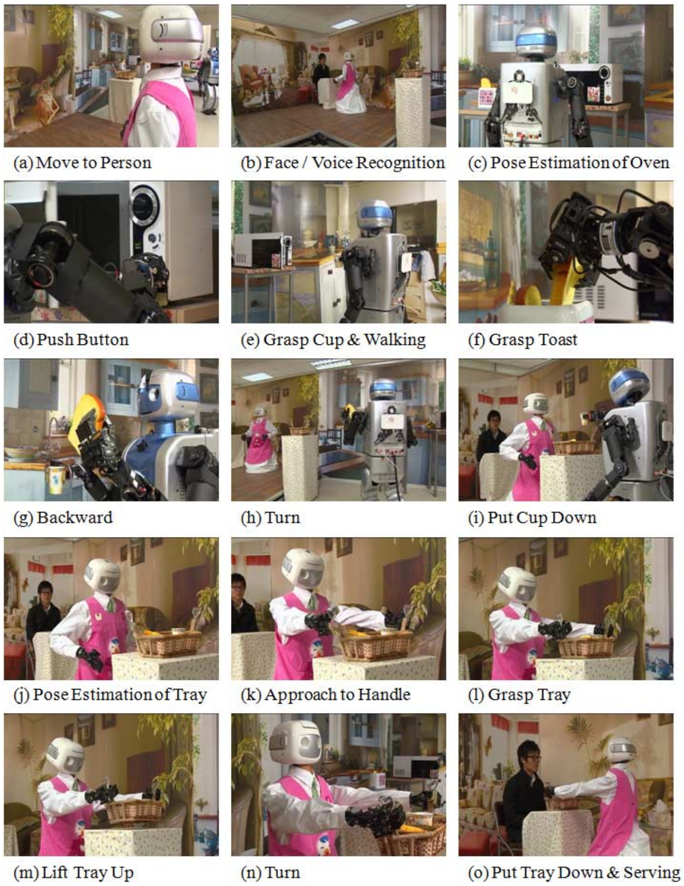


Fig. 13. The process of the task, "Service in Home Environment"

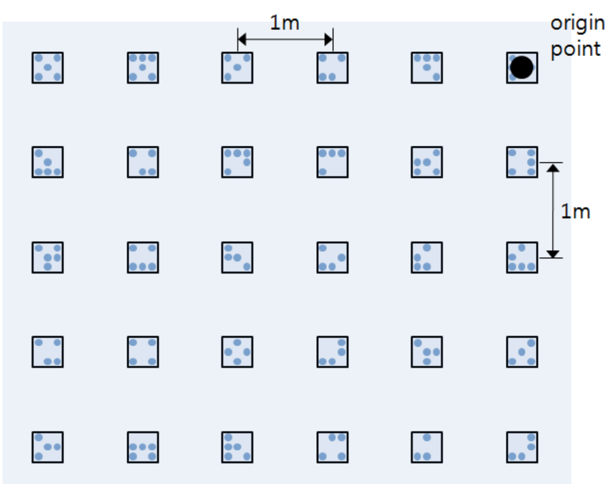


Fig. 14. An operation area and a landmark array

Using a stereo camera mounted on the robot's head, 3D information was precisely calculated for grasping the object. The grasping position had been already set in modeling and was obtained when the location estimation of the object was assumed. When the robot moved its hand to this position, it could grasp an object such as a milk cup or toast. When the robot did so, it was essential to control the grasp strength. The strength of grasp had been set experimentally according to the object type.

After grasping the milk cup and toast (f), MAHRU-Z moved to the food distribution table (g, h), and dropped them down onto a tray (i). Then, MAHRU-M detected the pose of the tray (j) and lifted it up (k, l, m). The robot moved to the person (n) and handed the milk and toast to the person who ordered them (o). A demonstration is successfully performed more than 50 times over the last 12 months.

VII. Conclusions

This paper introduced a coordinated framework for network-based humanoids. The humanoid robot consists of three sub-systems: the walking/mobile platform, and the perception system. The sub-system should be scheduled according to the task sequence. The task planner scheduled the actions for the diverse sub-systems, and the XML script had been presented for describing tasks. The design of a real-time distributed robot control system based on IEEE-1394 for the humanoid robot was presented. Based on the whole-body control framework, the humanoid robot is able to walk and balance its body. It is also able to plan its path, taking into consideration the location of an obstacle, and move itself by biped walking or wheeling. The location of an object was estimated with the particle filter; thus, real-time object tracking was possible. To track the face which was detected and recognized by AdaBoost and PCA methods, a mean-shift algorithm using bilateral filtering was used according to its robustness to illumination changes. Task operation by a humanoid robot was successfully performed in an experimental situation.

Humanoids which are made to function similarly to humans are a futuristic technology that can reduce the work load in work and home environments. This is a core technology of human welfare, and humanoids must be developed to be human-friendly, efficient, and similar to humans, so that they may co-exist with humans in a common environment. The ultimate aim of the humanoid research is to develop a humanoid which can provide services in real home environments utilizing a network.

Acknowledgment

The authors gratefully acknowledge the contribution of Interaction and Robotics Research Center and reviewers' comments.

References

[1] H. Ahn, I. Sa, and J. Choi, "PDA-Based Mobile Robot System with Remote Monitoring for Home Environment", *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, Aug. 2009.
[2] Y. Oh, J. Yoon, J. Park, M. Kim, and H. Kim, "A Name Recognition Based Call-and-Come Service for Home Robots", *IEEE Transactions on Consumer Electronics*, vol. 54, no. 2, May 2008.
[3] A. Morales, T. Asfour, D. Osswald, S. Schulz and R. Dillmann, "Towards an anthropomorphic manipulator for an assistant humanoid robot", *Robotics: Science and Systems-Workshop on Humanoid Manipulation*, MIT, USA, 2006.
[4] R. Dillmann, Teaching and learning of robot tasks via observation of human performance, *Journal of Robotics and Autonomous Systems*, vol. 47, no. 2-3, 2004, pp. 109-116.
[5] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki and K. Fujimura, "The intelligent ASIMO: System overview and integration", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2002, pp. 2478-2483.

[6] F. Kanehiro, H. Hirukawa and S. Kajita, OpenHRP: Open Architecture Humanoid Robotics Platform, *The International Journal of Robotics Research*, vol. 23, no. 2, 2004, pp. 155-165.

[7] K. Loffler, M. Gienger and F. Pfeiffer, Sensors and Control Concept of Walking 'Johnnie', *The International Journal of Robotics Research*, vol. 22, no. 3-4, 2003, pp. 229-239.

[8] J. Kim, I. Park, J. Lee, M. Kim, B. Cho and J. Oh, "System Design and Dynamic Walking of Humanoid Robot KHR-2", *Proceedings of the IEEE International Conference on Robotics and Automation*, 2005, pp. 1431-1436.

[9] IEEE. "IEEE standard for a high performance serial bus-amendment, IEEE Std 1394-2000", 2000.

[10] M.O.F. Sarker, C. Kim, S. Baek and B.-J. You, "An IEEE 1394 based real-time robot control system for efficient controlling of humanoids", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 1416-1421.

[11] R. Alami, A. Clodic, V. Montreuil, E. Sisbot and R. Chatila, "Task planning for Human-Robot Interaction", *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence*, 2005, pp. 81-85.

[12] C. Galindo, J. F. Madrigal, J. Gonzalez and A. Saffiotti, "Robot task planning using semantic maps", *The international Journal of Robotics and Autonomous Systems*, vol. 56, no. 11, 2008, pp. 955-966.

[13] J. Liu, "Robot assembly planning using a task grammar augmented with heuristic knowledge", *The international Journal of Intelligent Manufacturing*, vol. 7, 1996, pp. 201-215.

[14] E. Colon and H. Sahli, "CoRoBA, an open framework for multi-sensor robotic system integration", *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 2005, pp. 463-468.

[15] I. Song, F. Guedea-Elizalde and F. Karray, CONCORD: A control framework for distributed real-time systems, *IEEE Sensor Journal*, vol. 7, no. 7, 2007, pp. 1078-1089.

[16] M. Isard and A. Blake, Conditional density propagation for visual tracking, *International Journal of Computer Vision*, vol. 29, no. 1, 1998, pp. 5-28.

[17] C. Dubois and M. Davy, Joint Detection and Tracking of Time-Varying Harmonic Components: a Flexible Bayesian Approach, *IEEE Transaction on Audio, Speech and Language Processing*, vol. 15, no. 4, 2007, pp. 1283-1295.

[18] M. Pupilli and A. Calway, "Real-Time Camera Tracking Using a Particle Filter", *Proceedings of the British Machine Vision Conference*, 2005, pp. 519-528.

[19] K. Najim, E. Ikonen and E. Gomez-Ramirez, Trajectory Tracking Control Based on a Genealogical Decision Tree Controller for Robot Manipulators, *International Journal of Innovative Computing and Control*, vol. 4, no. 1, 2008, pp. 53-62.

[20] P. Viola and M. Jones, Robust Real-Time Face Detection, *International Journal of Computer Vision*, vol. 57, no. 2, 2004, pp. 137-154.

[21] R. Duda, P. Hart and D. Stork, *Pattern Classification*, 2nd Edition. Wiley Interscience; 2000.

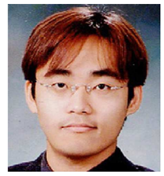
[22] Y. Lee, M. Jeong, J. Lee and B. You, "Robust Face Tracking Using Bilateral Filtering", *International Conference on Intelligent Computing*, vol. 1, 2008, pp. 1181-1189.

[23] D. Comaniciu and P. Meer, Mean shift: a robust approach toward feature space analysis, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, 2002, pp. 603-619.

[24] C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images", *Proceedings of the IEEE International Conference on Computer Vision*, 1998, pp. 839-846.

[25] M. Morisawa, K. Kaneko, F. Kanehiro, S. Kajita, K. Fujiwara, K. Harada and H. Hirukawa, "Motion Planning of Emergency Stop for Humanoid Robot by State Space Approach", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 2986-2992.

BIOGRAPHIES



KangGeon Kim (M'10) received the B.S. and M.S. degrees in Mechanical Engineering from Seoul National University, Seoul, Korea, in 2004 and 2008, respectively. Since 2008, he has been a research engineer at the Interaction and Robotics Research Center in the Korea Institute of Science and Technology (KIST). His current research interests include humanoid robotics, robot/machine vision, image processing, and intelligent robotics.

Young-Su Cha received the B.S. degree in Electrical Engineering from Korea University, Seoul, Korea, in 2004, and the M.S. degree in Electrical Engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2007. He is currently a research engineer at the Interaction and Robotics Research Center in the Korea Institute of Science and Technology (KIST). His current research interests include humanoid robotics, real-time control, and embedded system.

Jung-Min Park received the B.S. and M.S. degree in Computer Science from Ewha Womans University, Seoul, Korea, in 1989 and 1991, respectively, and the Ph.D. degree in Computer Science and Engineering from Ewha Womans University, Seoul, Korea, in 2008. In 1991, she joined the Korean Institute of Science and Technology (KIST), Seoul, Korea, where she is currently a senior research scientist at the Interaction and Robotics Research Center. Her current research interests include security issues in wireless/mobile networks and intelligence and software architecture in networked robot.

Ji-Yong Lee received the B.S. degree in Computer Science from Chonnam National University, Gwangju, Korea, in 2007, and the M.S. degree in HCI & Robotics from University of Science & Technology, Daejeon, Korea, in 2009. Since 2009, he has been a research engineer at the Interaction and Robotics Research Center in the Korea Institute of Science and Technology (KIST). His current research interests include object detection and recognition.

Bum-Jae You (M'89) received the B.S. degree in Control and Instrumentation Engineering from Seoul National University, Seoul, Korea, in 1985, and the M.S. and Ph.D. degree in Electrical and Electronic Engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1987 and 1991, respectively. He was a visiting fellow in Yale University, USA, in 1997, and was a visiting research scholar in Stanford University, USA, in 2006. He joined Korea Institute of Science and Technology (KIST) in 1994 after industrial experience in Turbo-Tek Co. Ltd., Korea, for three years. Now, he is the head of the Interaction and Robotics Research Center and a principal research scientist in KIST. His current research interests include humanoid robotics, network-based robotics, vision-based control, and digital signal processing applications.