J Inf Process Syst, Vol.13, No.6, pp.1459~1466, December 2017
https://doi.org/10.3745/JIPS.03.0083

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Task Management System According to Changes in the Situation Based on IoT

Cao Kerang*, Hyunju Lee**, and Hoekyung Jung**

Abstract

Recently, the development of the smart home field provides a range of services to install and keep the smart home appliance in a user's residential environment pleasantly. However, the conventional system method is not convenient enough to use properly because users have to select a device and manually operate the device on their own. In this paper, we propose a system to set the priority of the devices selected by the user and proceed with the task. When a user selects a device, the system recommends an optimal device associated with the device. The system compares and sets the priority of each device, carrying out the task one by one according to the set priority. Therefore, the proposed system is expected to provide users with increased convenience and more efficient task management.

Keywords

IoT, Priority, Sensor, Smart Home, Task Management System

1. Introduction

One at present, Internet of Things (IoT) aims to provide services considering the convenience and accuracy for users, and focusing on automation systems [1–3]. Accordingly, a study to address the requirements regarding the accuracy and efficiency of the operation of the task has been going on in the world. Home IoT, which incorporates IoT inside home, provides a service that automatically adjusts the environment inside home by performing data communication and linkage task through connection among devices [4–6].

However, there is a problem within the system which operates automatically among existing systems. It has to manually set the operation of the device because the operation of the device does not change according to the changes in the environment inside home [7–9]. In addition, since the user's intervention is not free, there is a problem that it is difficult to change the state of the device at a desired time point [10–12].

In this paper, to resolve this problem, we propose a system that classifies seasons and devices, performing tasks through environmental data in the home. It was also implemented to help the user progress through the task of a remote control device to control the device at the desired point. This can reduce the interference and collision between tasks and also increase the user's convenience.

Manuscript received May 30, 2017; first revision August 18, 2017; accepted October 11, 2017.
Corresponding Author: Hoekyung Jung (hkjung@pcu.ac.kr)
* Dept. of Computer Science and Engineering, Shenyang University of Chemical Technology, Shenyang, China (caokerang@hanmail.net)
**Dept. of Computer Engineering, Paichai University, Daejeon, Korea (hj1005@nate.com, hkjung@pcu.ac.kr)

2. System Design

The user configures the task environment by selecting the device through the application. The set priorities and sensor data are stored in the database and referred to the stored data to establish the relationship between the devices and to proceed with the operation. Fig. 1 shows the structure of the environmental control system.

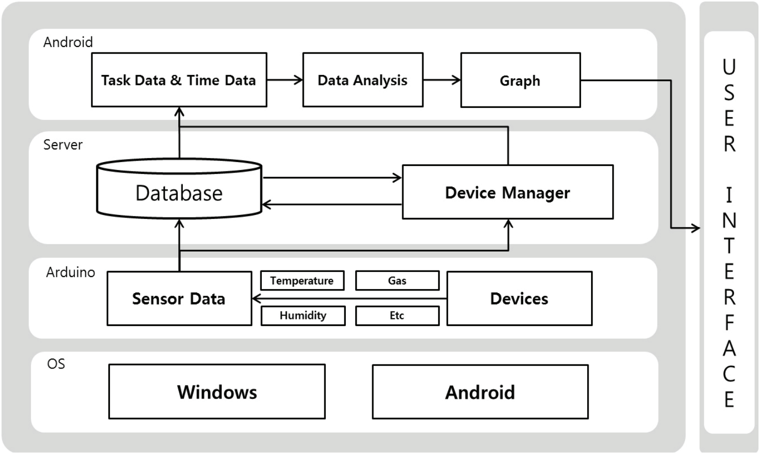


Fig. 1. Environment control system architecture.

At Arduino, each sensor value is measured and sent to the server and stored in the database. The database stores the type of each sensor, sensor value, and device information. The Device Manager in PHP identifies the situation through the received sensor value and the current date. It also selects the job according to the identified situation and sets the priority according to the selection value of the device. The application allows the user to monitor the status of each device and allows manual control at any time. The user's device control is set to have a higher priority than the operation progress, and the device in which the user directly changes the state maintains the changed state until the device switches to the standby state.

Sensor data and current date data are needed to identify the situation. We use two kinds of data in order to classify seasons and tasks and to judge the operation of the task. Fig. 2 shows the task classification algorithm.

When the sensor data is entered into the decision tree, classification begins. In the decision tree of each sensor, the seasons are classified by date and sensor values, and the tasks are classified by season. The reasons for classifying the seasons through decision trees are that the operation of selected devices and devices must be changed according to changes in the internal environment of the home, and that the types of devices and operations are different for each season. The threshold value of the classified task is compared with the current sensor value to judge the operation of the task then the device is selected and the status is set. By setting thresholds for each season, the system allows the environment inside the home to be maintained according to the seasonal changes.

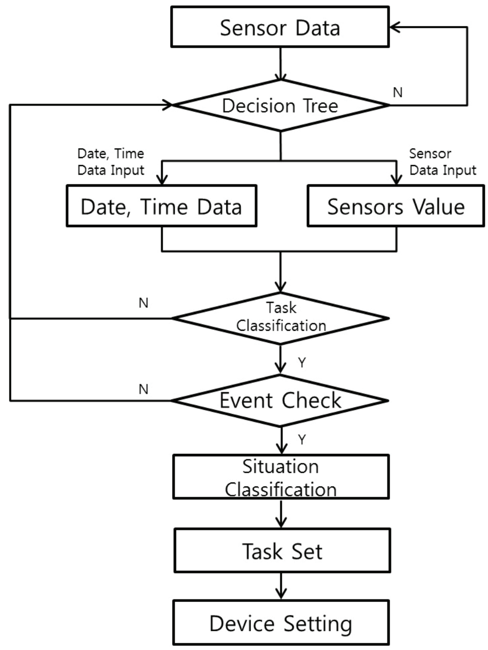


Fig. 2. Task classification algorithm.

Fig. 3 is a flow chart of device selection and relationship setting. To prevent malfunctions and bottlenecks caused by the simultaneous operation of the selected devices, one needs to set up the relationships among devices. After updating the status of the selected devices through the job classification of device database, the relation setting module compares the operation states of the devices and sets the relationship. It is possible to prevent malfunction caused by simultaneous operation of the apparatus by setting the priority of the apparatus and sequentially operating the apparatus.

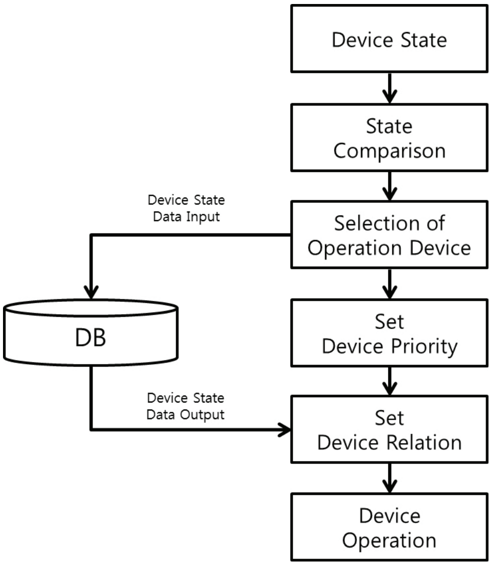


Fig. 3. Device selection and relation set flow chart.

3. System Implementation

The application is a module for the user to monitor the status of the device and remotely change the status of the device. In the application, the user can check the operation state of the current devices and change the state of the corresponding device. Fig. 4 shows the application status monitoring and user remote control implementation screen.

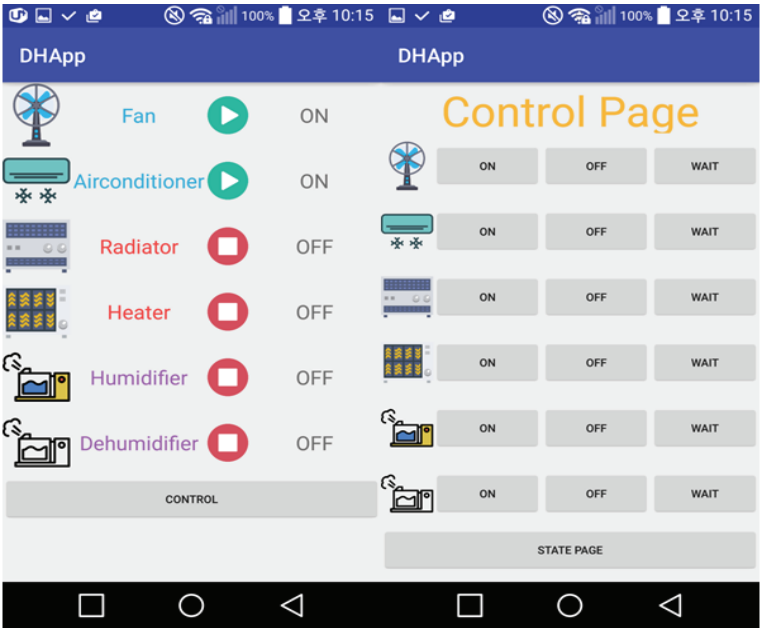


Fig. 4. Device state monitoring & remote control page.

The user can check whether the current device is operating by utilizing the application or not. The operation state of the current device stored in the database can be transmitted to the application through the server in order to confirm the change of the device operation. The user can also check whether the unnecessary operation of the devices in the home is operated by using the monitoring function. The user can control ON/OFF and WAIT device status by using the application. If ON is selected, the user's control command precedes the task instruction. Then the device will operate even when the link operation is in progress. On the other hand, when OFF is selected, the user's operation is prioritized as in the case of ON, and the operation of the apparatus will stop. WAIT is a state in which the user stops manual control of the device, changes the state to include the device in the job, and automatically operates the device. This allows the user to use the customized services in an automatic operation system.

Decision trees are used to classify tasks and devices, to establish relationships among devices, and then to operate the device. The sensor attached to Arduino measures the domestic environmental data and transmits it to the server. The server loads the received sensor value into database and transmits it to each sensor-based tree model to use as input data. The data of the operation and operation device, which are the result values classified through the decision tree, are transmitted to the automatic relation module. As a result, the priority and relationship among the devices are set.

In the prototype of this system, the sensor consists of temperature sensor, humidity sensor and gas sensor. The devices used in the process are windows, fan, air conditioner, heater, radiator, humidifier and dehumidifier. When the job classification and relationship setting are completed, the job starts, and the device operates in sequence. The window changes among operation depending on the operation of the temperature sensor. In this system, the air conditioner was replaced with a fan, and the dehumidifier was replaced with a humidifier in the humidity control operation. One connects the SSR to the Arduino to control the unit and turns the unit on and off. Sequential operation is performed according to the set priority and relation among the connected devices.

4. Review

The proposed system classifies the branch and season by the current date, selects the job, and uses the sensor threshold to determine the operation of the job. In addition, since the sequential operation is performed by automatically setting the relationship between the selected devices, it is possible to prevent interference and collision between tasks. When the task is in progress, the user can freely intervene and change the state of the device to manually control it. This allows users to receive customized services and provide convenience due to accurate seasonal behavior. Experiments were conducted using a data set to verify the performance of the system.

Table 1. Experimental dataset

System type	Description		
	Temperature (℃)	Date	Season
Test Dataset A	17	December 10, 2016	Winter
Test Dataset B	29	December 20, 2016	Winter

Table 1 show the dataset used in the experiment. Test Dataset A is the data used for operation verification and it is winter at 17℃. The Test Dataset B is the experimental data to compare the malfunction of the proposed system with that of the existing system. Since the temperature is set to 29℃, the operation of the air conditioner and the fan would not be operated. Fig. 5 is a graphical representation of device operation according to Test Dataset A.

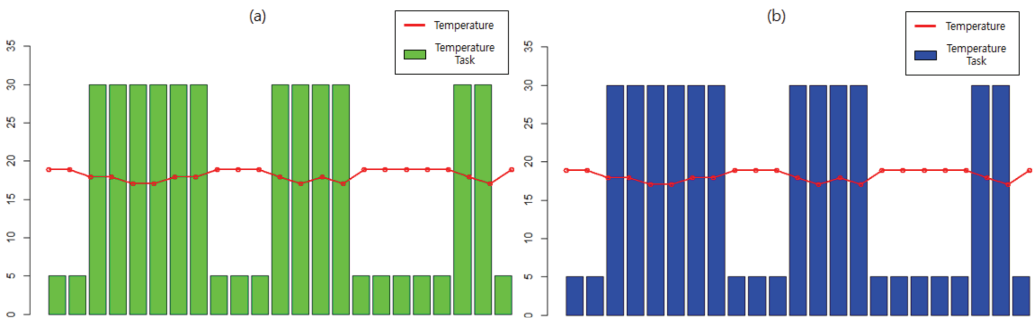


Fig. 5. Temperature control operation graph using Dataset A. (a) the existing system, (b) the proposed system.

Experiments using Test Dataset A showed that both the existing system and the proposed system preceded when the temperature reached the threshold value. Fig. 6 is graph shows the occurrence of malfunction using Test Dataset B.

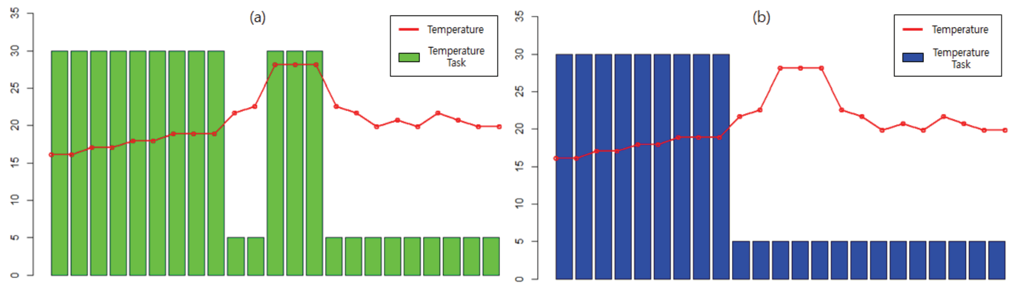


Fig. 6. Temperature control operation graph using Dataset B. (a) the existing system, (b) the proposed system.

In the case of the existing system, the temperature-rise operation starts when the temperature is below 17℃, and when the temperature reaches 29℃ or more, the temperature-decrease operation will be activated. Ultimately the air conditioner and the fan are operated. However, the proposed system only performs the temperature control operation when the temperature is below 17℃, and while the temperature lowering operation does not proceed when the temperature reaches 29℃.

Existing systems can only work with all devices regardless of the season because the task is done only through the threshold of the device. As shown in the experimental results, malfunctions such as the operation of the air conditioner and the fan in winter might be able to occur. As a consequence, unnecessary task may be initiated and electric power can be wasted caused. However, the proposed system classifies the seasons and tasks, thus prevents the malfunctions mentioned above. Through this experiment, it is verified that the proposed system can prevent the malfunction of the existing system and the power dissipation.

5. Conclusions

In the existing smart home system, all domestic devices in the home are connected to one network and operate automatically. However, this system causes a problem, causes a problem in which users must manually change the configuration and operation of the device by themselves. In specific, when the threshold value reaches a particular state, the existing system cannot cope with a changing environment properly.

In order to solve the problems of existing systems, this paper proposes a system that can manage task depending on the changing environment by using decision tree. In this system, the decision tree model for each sensor is applied to classify seasonal tasks and devices, selecting the devices to be operated sequentially. This method minimizes the conflicts and interference among tasks. The user can intervene at a desired time and change the state of the device. If users want to change manually the state of the device while the task is in progress, priority needs to be set appropriately. In addition, when users change the state of the device to the standby state, the changes are automatically included in the operation.

Future research need to expand the classification tree of the decision trees in order to enhance the proposed system. Then, it will contribute to construct a more convenient and accurate smart home system by changing the IoT system depending on various environmental changes.

Acknowledgement

This research was supported by The Leading Human Resource Training Program of Regional Neo industry through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (No. 2016H1D5A1911091). This work was supported by the research grant of Paichai University in 2017.

References

[1] J. A. Jeon, N. S. Kim, J. G. Go, T. J. Park, H. Y. Kang, and C. S. Pyo, "IoT devices product and technology trends," *Journal of the Korean Institute of Communication Sciences*, vol. 31, no. 4, pp. 44-52, 2014.

[2] J. H. Kang, H. J. Kim, and M. S. Jun, "Market and technical trends of internet of things," *Korea Contents Association Review*, vol. 13, no. 1, pp. 14-17, 2015.

[3] S. Husain, A. Prasad, A. Kunz, A. Papageorgiou, and J. S. Song, "Recent trends in standards related to the internet of things and machine-to-machine communications," *Journal of Information and Communication Convergence Engineering*, vol. 12, no. 4, pp. 228-236, 2014.

[4] Y. T. Lee, W. H. Hsiao, C. M. Huang, and S. C. T. Chou, "An integrated cloud-based smart home management system with community hierarchy," *IEEE Transactions on Consumer Electronics*, vol. 62, no. 1, pp. 1-9, 2016.

[5] X. Zeng, S. K. Garg, P. Strazdins, P. P. Jayaraman, D. Georgakopoulos, and R. Ranjan, "IOTSim: a simulator for analysing IoT applications," *Journal of Systems Architecture*, vol. 72, pp. 93-107, 2017.

[6] S. Zhao, L. Yu, and B. Cheng, "An event-driven service provisioning mechanism for IoT (internet of things) system interaction," *IEEE Access*, vol. 4, pp. 5038-5051, 2016.

[7] W. Lee, S. Cho, P. Chu, H. Vu, S. Helal, W. Song, Y. S. Jeong, and K. Cho, "Automatic agent generation for IoT-based smart house simulator," *Neurocomputing*, vol. 209, pp. 14-24, 2016.

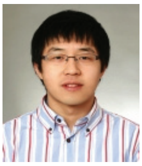
[8] C. Perera, A. V. Vasilakos, "A knowledge-based resource discovery for Internet of Things," *Knowledge-Based Systems*, vol. 109, pp. 122-136, 2016.

[9] Y. Evchina, J. Puttonen, A. Dvoryanchikova, and J. L. M. Lastra, "Context-aware knowledge-based middleware for selective information delivery in data-intensive monitoring systems," *Engineering Applications of Artificial Intelligence*, vol. 43, pp. 111-126, 2015.

[10] C. Kamienski, M. Jentsch, M. Eisenhauer, J. Kiljander, E. Ferrera, P. Rosengren, J. Thestrup, E. Souto, W. S. Andrade, and D. Sadok, "Application Development for the Internet of Things: A Context-aware Mixed Criticality Systems Development Platform," *Computer Communications*, vol. 104, pp. 1-16, 2017.

[11] M. Elkhodr, S. Shahrestani, and H. Cheung, "A smart home application based on the Internet of Things management platform," in *Proceedings of 2015 IEEE International Conference on IEEE Data Science and Data Intensive Systems (DSDIS)*, Sydney, Australia, 2015, pp. 491-496.

[12] Y. T. Lee, W. H. Hasiao, C. M. Huang, and S. C. T. Chou, "An integrated cloud-based smart home management system with community hierarchy," *IEEE Transactions on Consumer Electronics*, vol. 62, no. 1, pp. 1-9, 2016.



Cao Kerang

He received the B.S. degree in 2006 from Computer Science and Application of North Eastern University, China and Ph.D. degree in 2011 from the Department of Computer Engineering of Paichai University. From 2016 to the present, he worked for Shenyang University of Chemical Technology as a Lecturer. His current research interests include multimedia document architecture modeling, information processing, IoT, big data, and embedded system.

Hyunju Lee

She received the Bachelor of Engineering from the Department of Computer Engineering Education of Andong University, Korea, in 1999, Master degree from the Department of Computer Education of Daejeon University, Korea, in 2003. She is currently a Doctorate course in Department of Computer Engineering of Paichai University. She current research interests include data analysis, machine learning.

Hoekyung Jung https://orcid.org/0000-0002-7607-1126

He received the B.S. degree in 1987 and Ph.D. degree in 1993 from the Department of Computer Engineering of Kwangwoon University, Korea. From 1994 to 2005, he tasked for ETRI as a researcher. Since 1994, he has tasked in the Department of Computer Engineering at Paichai University, where he now tasks as a professor. His current research interests include multimedia document architecture modeling, information processing, information retrieval, and databases.