

# Design and Implementation of IoT-based Intelligent Surveillance Robot

Moon Sun SHIN<sup>1</sup>, Byung Cheol KIM<sup>2</sup>, Seon Min HWANG<sup>3</sup>, Myeong Cheol KO<sup>1</sup>

<sup>1</sup> Department of Computer Engineering, Konkuk University,  
268 Chungwon-daero, Chungju-si, Chungbuk, 27478, Republic of Korea,  
msshin@kku.ac.kr, cheol@kku.ac.kr

<sup>2</sup> Department of Information and Communication, BaekSeok University,  
76 Munam-ro, Dongnam-gu, Cheonan-si, Chungnam, 31065, Republic of Korea,  
bckim@bu.ac.kr

<sup>3</sup> BizForce Co. Ltd,  
Cheongju-si, Chungbuk, Republic of Korea,  
smhwang@kut.ac.kr

**Abstract:** Recently, the continually evolving robot technology has contributed to the development of a variety of robot-using service models. In particular, the robot with the artificial intelligence may be utilized in various fields of the context awareness. Conventional CCTV is installed in a fixed form and it can transfer monitored images to a central server where they are stored in the server system. The monitoring and control processes must be performed at the server side. In this paper, an intelligent surveillance robot for which Internet of Things(IoT) is used to overcome the limitations of these existing CCTV is proposed. The proposed intelligent surveillance robot is called "SMART-I", and it can perform monitoring and control remotely. Also a DVR server is included in the robot design for the transmission and processing of large image data. The prototype of SMART-I has been developed based on IoT, and both the app client and Web client have been implemented to provide the remote monitoring and control of the robot. SMART-I can be used as an active surveillance robot. While it monitors activities, it can detect abnormal contexts that are then notified to a security manager through the sending of alarm messages to a smartphone, allowing for a quick response to the potential risk. The developed IoT-based intelligent surveillance robot can perform monitoring activities in locations such as factories, offices, and industrial sites. SMART-I can be used as a user-friendly mobile robot that provides real-time monitoring and intelligent services according to the context awareness in a variety fields for an active performance of surveillance activities.

**Keywords:** IoT, Surveillance robot, Intelligent service, Context awareness, Active CCTV.

## 1. Introduction

Recently the development of the robot industry has been directed towards the support of intelligent convergence services. As industrial society has changed to the information society, the robot paradigm is changing the trend, as people-friendly, IT-based intelligent-service robots are developed now as an alternative to the traditional work robot. Unlike conventional robots, which were performing passive, repetitive tasks according to people's orders, IT-based intelligent-service robots provide intelligent and proactive services through a convergence of IT and network technologies [8]. The research on intelligent robots has been spotlighted in accordance with the development of convergence-ICT technology; here, the focus is the development of an optimized service in the application domain [8]. In this paper, an intelligent-surveillance robot that is like the social-security robot will be developed using the Internet of Things (IoT) in connection with the recent smart environments. The proposed

intelligent-surveillance robot is capable of actively monitoring and taking pictures in specific areas. Also, the robot performs real-time communication based on the IoT, and controlled and monitored remotely with the use of smart phone.

It is possible for people to enact immediate response and to prevent some kinds of accidents, if mobile CCTV or real-time monitoring exists for the identification of specific place by a smart phone.

An intelligent robot that is camera-equipped and for which the IoT is included, can be used to proactively manage the surveillance and condition monitoring of specific areas like farms, housing, factories, and material warehouses [15]. To support an intelligent-surveillance service, the robot must perform the following functions:

Being equipped with a high-resolution CCTV camera, and it must be capable of DVR and the monitoring of the specific section; the performance of real-time video-transmission

capabilities to proactively recognize situations and remote administrators; and manual and automatic four-wheel driving and line tracing where by it can be controlled by the Web or a smartphone app to increase the ease of use.

The focus of the active-surveillance robot that was developed in this paper is the rapid resolution of an abnormal situation through more than the prevention and detection of intrusion, environmental pollution and the potential disasters of a variety of fields.

The rest of the paper is organized as follows. In section 2, we explain related works of context awareness and intelligent robot. Section 3 presents the design and implementation of the IoT-based intelligent-surveillance robot and section 4 describes the test scenario and the experimental results. Lastly a brief conclusion is presented in section 5.

## 2. Related Works

In recent years, the CCTV (closed circuit TV) video-surveillance system has been continuously introduced into a variety of fields. Also it has developed into a network-based CCTV. The intelligent-CCTV system has been evaluated for its ability to monitor situations very effectively, as images that can be taken with CCTV cameras around the track have automatically detected the characteristics of an object or a person.

Nevertheless most CCTV systems have been installed at fixed positions and they send data to a central server by photographing an image to provide monitoring or searching functionality [11]. However, it is difficult to immediately respond to the risks and anomalies because monitoring can only be performed in the control center after the images have been sent to a remote server. It is therefore useful that intelligent CCTV systems can provide an alarm to trigger an immediate response to an abnormal situation. A context-aware service can recognize the surroundings and circumstances, and it then provides an appropriate service [16]. Recently, the need for context awareness to serve as the intelligent service in ubiquitous environments has increased with the development of the variety of sensor technologies [17]. Many kinds of techniques such as machine learning, Bayesian network,

data mining, and collaborative filtering are applied for the construction of context-aware models to provide customized intelligence services in the variety of domains [14].

First, the context-aware model must be constructed so that the context-aware technology can be applied [6]. A probability model of the Bayesian network model can be used as the main method for the recognizing of the situations of users. A Bayesian network model can be configured by using either a manual model that has been designed by experts, or a learning model from the trained data. If the sufficient amount of data is given it is possible to obtain a reliable model through learning, but often the data is low or not [4]. Therefore, a systematic approach is required for the construction of an effective manually designed model that is based on the domain knowledge. To solve the problem for the context awareness of the home robot, research of the application of the Bayesian network model has been carried out, and attempt was made to combine the robot technologies [9]. In [9], they tried to configure a Bayesian network model that can be trusted by a manual design that is based on the domain knowledge, as the attainment of sufficient data was not possible [3].

The AI Lab of MIT carried out the “Oxygen” project with support from DARPA and other companies. Oxygen is the future of computing technology projects, and like the oxygen in the air, it must be human-oriented and for use at anytime and anywhere. According to the basic concept of the Oxygen project, a mobile device, a camera, a microphone, a display, a speaker, and various radar apparatuses have been used to acquire the context knowledge of the situation around the person of the movement and action [7].

Various studies regarding context awareness were conducted by Georgia Institute of Technology. The Future Computing Environments Group developed the “Context Toolkit” and they attempted to get the context information from the sensor by using the Context Widget. They analyzed the collected information in the context and derived a high-level context-aware knowledge [1]. The “Aware Home” project was also carried out to develop ubiquitous computing technologies for the provision of a variety of the convenient services that are required by the user according

to the user's context information from a smart-home analysis. A particular house had been built as a test bed for the carrying out of research on the mounting a variety of sensing devices in the camera, the microphone, and so on. The input data from the disparate sensors were gathered, and then the study of the management of the context information was carried out through the user's identity, location, gestures, behaviors, and so on [13].

The "SOCAM" (Service-oriented Context-aware Middleware) was proposed as a middleware to support the development of context-aware service infrastructures. In SOCAM, the use of the ontology concept whereby the OWL was applied to the context model was suggested, and the appropriate status information that was obtained through the ontology reasoning consequently has been provided to the application [2].

The EU established joint-research group called EU Framework Program. The EU Framework Program has run a variety of projects that were related to the cognitive situation and they are ongoing and are in progressive development.

The ETRI developed the "CAMUS" robot system to provide an active robot for the realization of an intelligent-service robot system, and they also proposed an evaluation method for robot-service availability [10].

ETRI also developed "Bossam," which was an intelligent robot with the inference capabilities of the ontology-based semantic Web reasoning [12]. The built-in ontology-based inference engine, the query interface, and the Web service capabilities were all supported by "Bossam". These features ensured high availability in the distributed environments.

The research was carried out to implement a home-security system that is based on the mobile robot and the home server, whereby a variety of sensor nodes were used. The mobile intelligent robots that are called "dynamic objects" could recognize different situations constantly. The home security system based on a sensor network (HSSN), was configured by sensor nodes including those for the radio frequency (RF), the ultrasonic, temperature, light and sound sensors [4]

Many researches on intelligent-robot services regarding the various applications have been in progress [5]. However, the development of

these studies was presented as mostly framework for server-client robot architecture monitoring; although, various attempts have been made to provide a context-aware service that is both active and intelligent [18].

In this paper, we proposed intelligent surveillance robot using Internet of Things. IoT-based surveillance robot was differentiated from the existing robot systems with the server-client robot architecture; that is, the robot has become the server robot itself. It was available anytime, anywhere through the use of a Wi-Fi network. And it could make management of the monitoring and control easy, as well as providing an immediate response to abnormal situations. The proposed intelligent surveillance robot includes its own server for which the IoT is used, and it also provides smartphone-app client. The robot also supports user convenience in terms of the management of smartphone-based control and monitoring.

### **3. Design of IoT based Intelligent Surveillance Robot**

First, the requirement analysis is proceeded to design an IoT-based intelligent-surveillance robot for remote control and monitoring. The following section describes the framework of the proposed robot and the requirements analysis. The design of the IoT-based intelligent-surveillance robot is proposed in Section 3.2.

#### **3.1 Requirement analysis**

In this section, the intention is a description of the requirement analysis regarding the functions that the proposed IoT-based intelligent surveillance robot must perform. After analyzing the requirements, they are specifically considered for the design and implementation processes.

Surveillance robots can be applied as either a mobile or a fixed form.

The application must be carried out real-time learning and context-aware processes depending on whether the situation is for active monitoring.

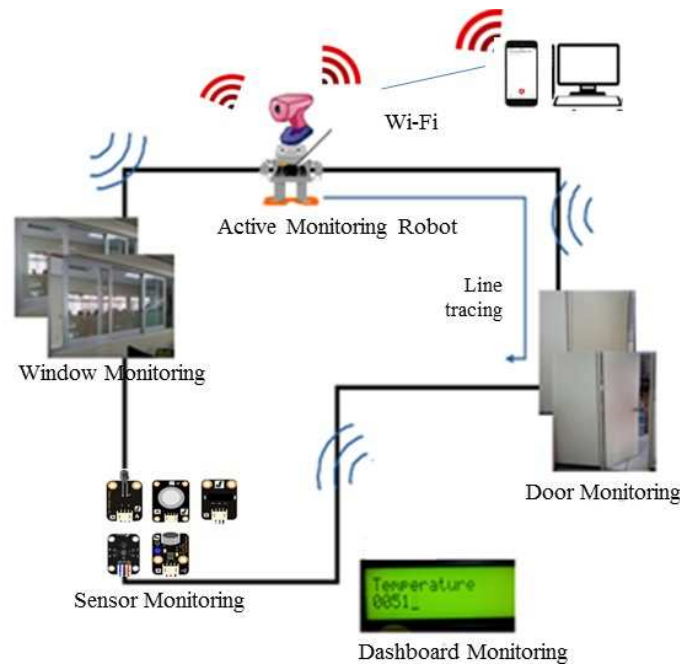
It is also necessary to move the autonomous navigation using a map, or by using a tracing line for the monitoring of a specific area.

The webcam is needed to perform the picture-taking function at particular places.

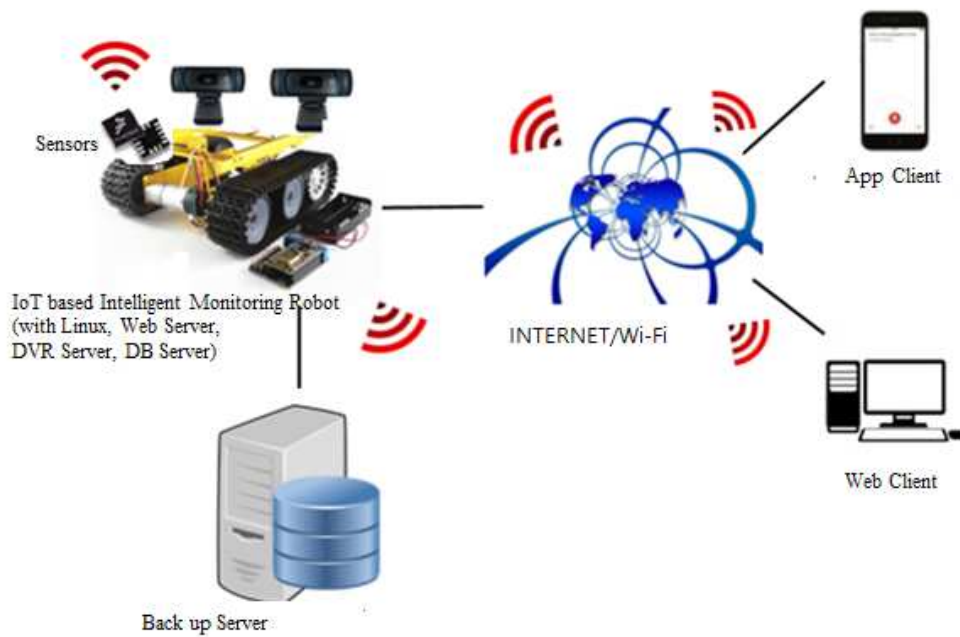
In terms of future utilization, storage management is required for searching images later. To enable the Internet at all times, the IoT technology is needed to apply for remote control and management for smartphones and the Web. The camera may need to be equipped with a 5M-pixel over the image -pickup device so that the objects can be clearly identified. Ideally, low-power products are those that can be operated on battery power for a long time; in addition, the camera operation can be occurred during the daytime as well as the night-time, so an infrared-shooting function and a lighting function must be included. Since the height of the mobile robot could be under 20 cm, some problems regarding a limited shooting area could arise. To solve such problems, a motorized height-adjusting device can be attached to the robot; also this device can adjust the position of the camera so that the recording and shooting area can be expanded. The robot can also perform unattended monitoring functions, detect the light intensity changes, and detect the situation changes and context awareness, the latter two of which are used to actively transmit information to an administrator in a remote location.

The Wi-Fi network and the TCP/IP, protocol-based standardized transmission technology are used for support so that the user interface of

different client environments can be exploited. In addition, the situation must be recognized and the alarm that is the notification of an abnormal occurrence must be sent to the smartphone. The image-processing function for the surveillance of an intrusion can be performed by the robot. A Web server and a DVR server must be built in the main controller that is mounted on the robot to enable the searching of stored images at remote locations. The DVR server should be able to perform a regular image storage function, the event storage function and search function. Web client is necessary for a remote user to receive imaged and to control the movement of the robot. Sensing data transmitted from the sensor in the robot body must be shown to a remote web browser in real time. It must be able to provide the management of the control the robot and image processing through the smartphone, so it requires smartphone video-reception capability and support for the smartphone's robot-control function. Figure 1 shows the application scenario of the intelligent-surveillance robot that reflects the requirements that are listed above. An intelligent robot with a built-in server for which the IoT is used performs active monitoring inside factories or offices and it can be controlled using a smartphone or the Web.



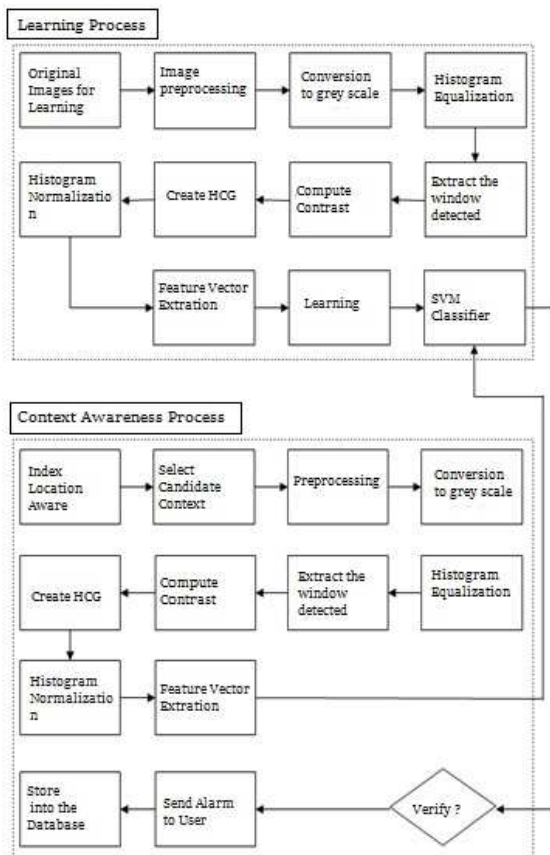
**Figure 1.** Application scenario of intelligent surveillance robot



**Figure 2.** Service Framework of IoT-based Intelligent Surveillance Robot

### 3.2 Design and implementation

Figure 2 shows the service framework of IoT-based intelligent-surveillance robot. The active



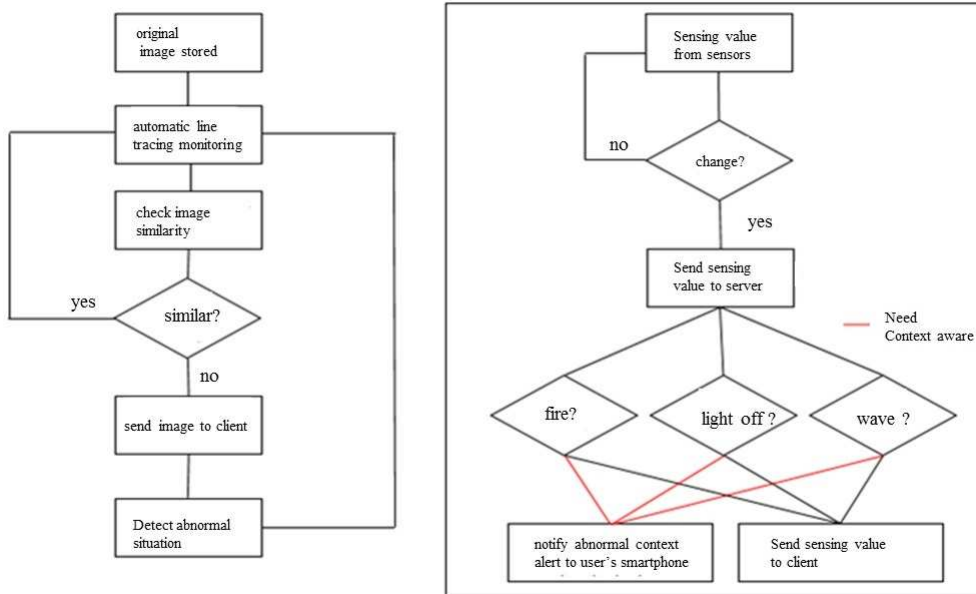
**Figure 3.** Learning Process and Context Aware Process for Surveillance Robot

intelligent robot as shown in the figure has its own server and can communicate with a PC or a smartphone using Wi-Fi. In the framework, camera-captured monitoring images are stored in the database and are streamed to the client in real time.

As shown in the framework, the monitoring images are stored in the database and streamed to the client display. The IoT-based intelligent surveillance robot developed in this paper is SMART-I.

The learning process and the context-aware process for the intelligent surveillance robot is shown in Figure 3. In the learning phase, the image of the normal state is stored in the original\_data table.

In context-aware process, the robot compared the monitoring image of a particular location with the original image. Then, if the image similarity is lower than 80 %, an abnormal state is identified, and the robot can notify the abnormal state to the user. Intelligent anomaly-detection logic is shown in Figure 4. The image of the normal state must have been stored together with the position value that is called the “index”. The image-similarity test logic is preceded by the original picture-taking mode, when the line tracing is performed and the index of the location is figured out. The DVR manager loaded an image of normal state with



**Figure 4.** Logic of Abnormal Detection by Image Similarity and Sensing Value Threshold

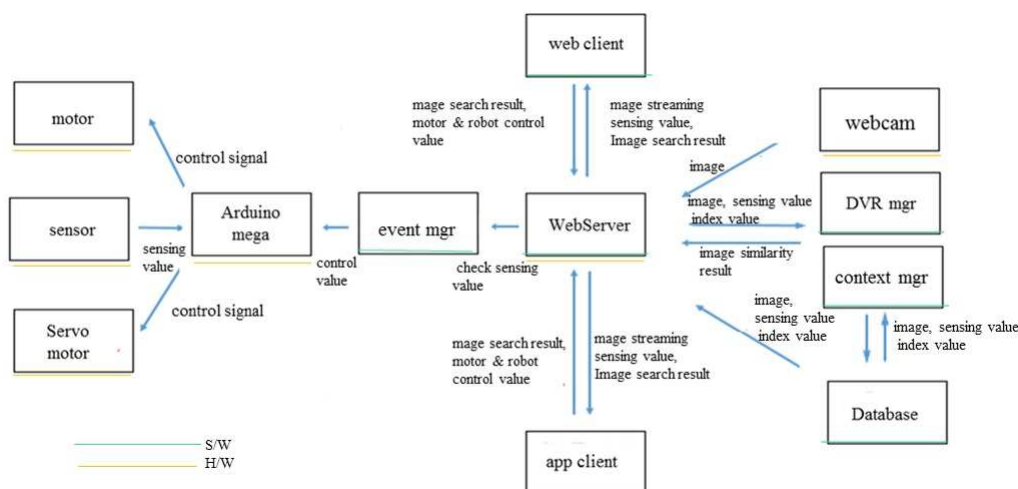
the exact index and a histogram-based image-similarity test had been carried out. If the result value of the similarity test is over 0.2 errors, an image socket is sent to the client; otherwise, the driving is maintained to proceed. As the sensor logic is shown in Figure 4, the combination of the string values that are sent from the sensors is sent to the Web server by serial communication. The Web server, then, performs a parsing of the string with special characters that are based on the characters that are split into the socket communications to the client. If there values that were over the threshold had existed, the robot send a push notification to the user smartphone. Figure 5 showed the data flow of SMART-I. The data

received by the client were the various sensing values and video streams. The Web server parsed the various sensing values from the client and sent them to OpenCV.

OpenCV then stored the sensor values and images in the database. The client could request the image search, the remote control of the robot, and real-time monitoring.

### 3.3 SMART-I server prototype

Raspberry Pi and Arduino were used to build the SMART-I server. The robot was composed of motor drivers for moving, servo motors, sensors, and an infrared sensor that was connected to the Arduino using a



**Figure 5.** Data Flow Diagram for SMART-I

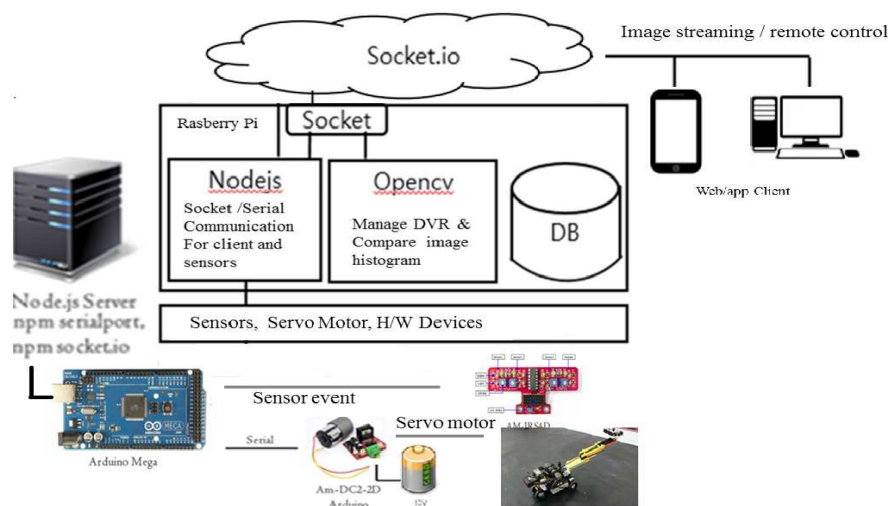


Figure 6. SMART-I System Architecture

separate battery for line tracing. Arduino can be controlled via a serial communication with the fuselage Raspberry Pi and the USB ports. The surveillance camera for the image capturing was a webcam that used Apache server port 8080.

The Node.js server platform was built and OpenCV and C++ were used for the image processing. Image processing was the function for notification of the abnormal states that were detected by performing inspections of the image storage and the image similarity. Figure 6 showed the intelligent-robot monitoring SMART-I system framework and the process for the detection of the status information of a specific area from among the ultrasound, temperature, light, carbon monoxide and various vibration sensors.

The data sent from the hardware equipment were the infrared sensor, the video streaming and various sensor values. The Node.js server performed a parsing the sensing values, the result of which was and that was sent to OpenCV as a file.

Database schema was shown in Figure 7. The images taken with the camera were all stored in the database table named compare\_data, and the original\_data table had been used to store the pictures of the normal state. The primary key is the index of the bottom surface has decided to recognize the Infrared Sensor. Each attribute represents each sensor value, as follows: “path” for image of path, “temp” for temperature sensor value, “light” for light sensor value, “sound” for sound sensors value, “wave” for vibration sensor values, “fire” for

flame sensor value, and “co” for carbon monoxide sensor values.

original_data				compare_data			
ind	VarChar(10)	NI	(PK) (AK1)	num	Int	NI	(PK)
path	VarChar(300)	NI		ind	VarChar(10)	NI	
regdate	Timestamp	NI		path	VarChar(300)	NI	
temp	VarChar(10)			regdate	Timestamp	NI	
light	VarChar(10)			temp	VarChar(10)		
sound	VarChar(10)			light	VarChar(10)		
wave	VarChar(10)			sound	VarChar(20)		
fire	VarChar(10)			wave	VarChar(20)		
co	VarChar(10)			fire	VarChar(20)		
				co	VarChar(20)		

Figure 7. Database Schema for SMART-I

While SMART-I was driving and monitoring, the sensing value from the various sensors, could be used to make an event. If the event occurred and context value was not under the threshold value set by the user, it could be sent to the user as a notification.

Figure 8 presented the software components of SMART-I. The IoT-based Robot Server consisted of five components and each component performed the following functions. The monitoring manager performed automatic or passive movement of the SMART-I. DVR Manager stored the images taken with the camera and supported the streaming service.

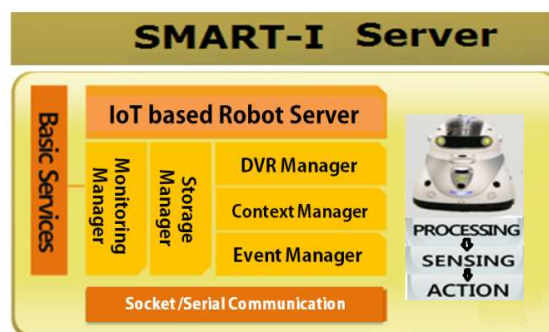


Figure 8. Software Architecture for SMART-I

In addition it performed the search and check of the stored images. Context Manager performed the comparison of the current image with the stored image of the normal state at the specific location, and it could send notifications to the user according to the image -similarity check. The Event Manager could inform the user of an abnormal state if the sensed value from the sensor was not properly compatible with the reference value.

The Storage Manager performed storage management and database backup once per month.

A prototype of SMART-I with four wheels for manual control and automatic control was developed. The prototype of SMART-I was shown in Figure 9. SMART-I had been equipped with a touch screen in the front portion, which could be easily and conveniently adjusted to access the Web client and the various control and monitoring functions in the fuselage itself. With a built-in touch-screen, it could be easy and simplified to configure and control by touch.



Figure 9. Intelligent Surveillance Robot SMART-I

### 3.4 Web/app client

SMART-I can be monitored and controlled both on the Web and with a smartphone app. The Web client of SMART-I was shown in Figure 10. It must be possible to control and monitor status of the robot in the Web, so the Web client provided a View&Control menu for the active mode, camera control and an image search. The active mode was a menu for the operation of the robot and the controlling of the direction. The camera control was a menu for controlling the angle of the camera.

The camera control menu carried out to send the angle value of the servo motor using socket communication and serial communication. The

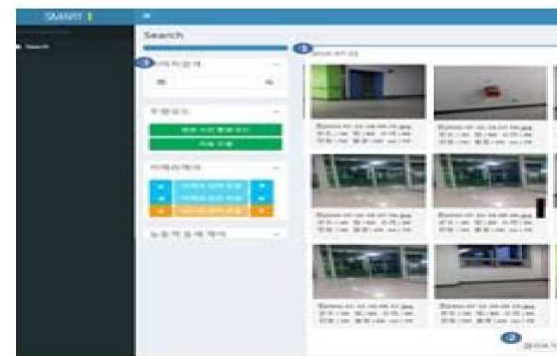
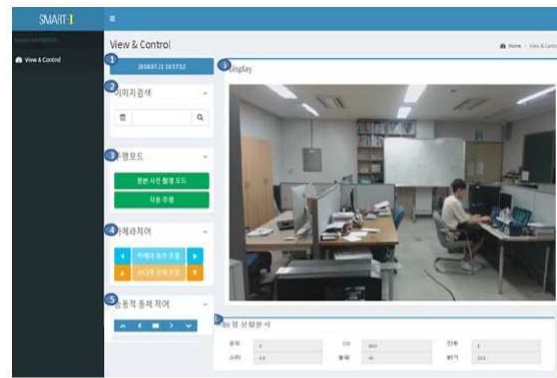


Figure 10. Web Client for SMART-I

camera left and right menu for the adjustment of the servo control so that the camera can be moved from side to side.

The up and down control menu could be used to adjust the ladder height of the camera so that recording from 20 cm to 100 cm height is possible. Therefore the user could intuitively adjust the vertical position of the ladder via the buttons.

The smartphone could also be used to control SMART-I remotely. The developed app client for which Android Studio 2.0 was used as the development tool was shown in Figure 11.



Figure 11. App Client for SMART-I

Figure 11 showed the main menu of the SMART-I app. When running the app the first time, an IP-address must be set to establish connection to SMART-I.



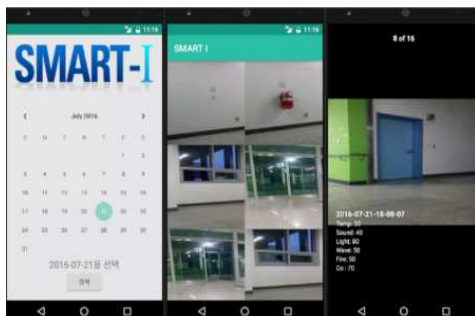
The four main menus such as View & Control, Search, Settings, and SignOut were supported in the app. The configuration of the menu performed the same function with the Web client in order to maintain an easy UI and consistency.

The View & Control menu provides a view of the screen that was being streamed to the robot, and the robot could be controlled using the manual mode of the View & Control menu. Also it allowed users to check the sensor value that were sent to the robot in real time.



**Figure 12.** View & Control menu

In the Search menu, the photos taken on that date could be searched and checked. Figure 13 showed the result from the searching of the stored images by date when the date was chosen.



**Figure 13.** Image Search menu

Using the Setting menu it is possible to set the IP address, streaming port, and control port.

The Sign Out menu could be used to log out of the menu and to clear IP address and port number.

When the abnormal state would be detected during surveillance, a push-alarm service was provided through the smartphone.

## 4. Test Scenario and Experiments

The experiments of three categories were tested to verify the operations and functions of SMART-I. The first test was that the most basic function for the active monitoring of the as-specified driving routes as specified could be performed precisely. Secondly, it was tested whether SMART-I is capable of context awareness and the detection of abnormal states at a specific place. The third test was whether SMART-I could perform various forms of sensing and provide proper alarms according to the threshold value.

The test scenarios were planned and were used to verify the test results. For the test results, the experiments were repeated 20 times to verify the executions, and the results were verified with more than 82% accuracy. Test cases and test scenarios are described as follows:

### Test Case 1: Manual Control

1. Select the <Manual Control> button in the app.
2. Select the <Active Robot Control> menu on the Web.
3. Control the robot using up, down, left, right, and stop button.

### Test Case 2: Automatic Control

1. Select the automatic-run button
2. The robot receives command of automatic run then starts automatic line tracing along the black line.
3. While running the robot meets the indexing line and then stops to capture a picture at the indexed line.
4. Checks the image similarity and provides a notification in the case of an abnormal state.

### Test Case 3: Normal Image Shooting Mode

1. Select the automatic run button
2. The robot receives the command of automatic run and then starts automatic line tracing along the black line.
3. While running the robot meets the indexing line and then stops to capture a picture at the indexed line.
4. Six index lines exist, so to repeat the basic flow 2-3.

**Test Case 4:** During running, the robot check whether the entrance door is open or not, and sends an alert message depending on the condition of the door.

1. The robot is running, and the original photos were stored in the robot
2. During running, the robot meets the indexing number 1.
3. Continue monitoring if the captured picture is the same as the original.
4. If the recorded picture is different from the original, the robot provides an alert.



Normal image



Abnormal:  
the door is open

**Test Case 5:** During running, the robot checks whether the window is open or not, and sends an alert message depending on the condition of the window.

1. The robot is running and the original photos were stored in the robot.
2. During running, the robot meets the indexing number 2.
3. Continue monitoring if the captured picture is the same as the original.
4. If the recorded picture is different from the original, the robot provides an alert.



Normal image



Abnormal:  
the window is opened

**Test Case 6:** During running, the robot checks whether a special object exists or not, and sends an alert message depending on the existence of the object

1. The robot is running and the original photos were stored in the robot.

2. During running, the robot meets the indexing number 3.
3. Continue monitoring if the captured picture is the same as the original.
4. If the recorded picture is different from the original, the robot provides an alert.



Normal image



No fire extinguishers

**Test Case 7:** During running, the robot check whether the sensing values exceeds the specified threshold or not, and sends an alert message depending on the excess.

1. The robot is running and the specific threshold had been set in the robot.
2. During running, the robot check whether the sensing values exceed the specified threshold or not
3. Continue monitoring if the sensing value is under the threshold.
4. If the sensing value exceeds the threshold, the robot provides an alert.

Test cases 1 to 3 tested the function of the intelligent monitoring robot. After having done 20 repeat experiments, it was verified that the operation of SMART-I worked normally. However, battery problems were identified. The consumption of battery power and the charging of the battery must therefore be considered. For the test cases 4 to 7, the tests were performed to see whether SMART-I carried out proper context-aware services according to the context information. The results verified the accuracy rate of the context-aware services performed by SMART-I is over the average accuracy of 85 %.

The graph in Figure 14 presented the experimental results for the accuracy of context-aware services according to the setting of different reference value of the image similarity.

We will going to additional experiments to increase the accuracy even at night. Also, an

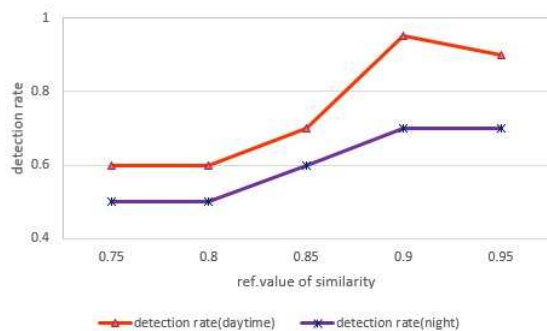


Figure 14. Experimental Results

additional test that will provide an appropriate reference value regarding the similarity for the user are being considered.

The experimental results verified that the IoT-based intelligent surveillance robot, SMART-I, performed remote control and monitoring functions normally. The improvement of both the battery issue and the performance of the abnormal detection are planned as future works.

## 5. Conclusion

In this paper, an intelligent surveillance robot called SMART-I, for which IoT technology was applied, was proposed, and the prototype of the SMART-I was implemented. SMART-I consisted of an IoT-based robot server and a Web/app client. The IoT-based robot server had main components such as monitoring manager, storage manager, event manager, DVR manager and context manager.

The proposed intelligent surveillance robot performed the functions of four-wheel driving and line tracing, the intelligent monitoring of specific places, real-time video transmission and processing, context-aware of the abnormal situation by comparing the image similarity. The client of the SMART-I is supported by both the Web and a smartphone app for ease of use, and SMART-I consisted of its own touch screen display.

The evaluation of the developed robot system was conducted through the construction of some test cases. First, the basic experiments of the tracing and the sensor data were carried out to verify the basic function of the robot. Also the experiments regarding the context-awareness of abnormal situations were performed so that it was possible to figure out the availability and validation of the robot. For

a future work, a solution of battery problem will be sought out, and additional robustness tests will be performed.

## Acknowledgements

This paper was supported by Konkuk University 2016.

Myeong Cheol KO is a corresponding author.

## REFERENCES

1. ANTONIOU, G., H. BOLEY, **RuleML**, LNCS 3323, 2004, pp. 128-138.
2. BIEGEL, G., V. CAHILL, **A Framework for Developing Mobile, Context-aware Applications**, IEEE International Conference on Pervasive Computing and Communications, 2004.
3. DOLGUI, A., O. AMMAR, F.HNAIEN, M. ALY, O. LOULY, **A State of the Art on Supply Planning and Inventory Control under Lead Time Uncertainty**, Studies in Informatics and Control, ISSN 1220-1766, vol. 22(3), 2013, pp. 255-268.
4. GO, Y. C., J. C. SOHN, **Context Modelling for Intelligent Robot Services using Rule and Ontology**, ICACT 2005, Volume 2, 2005, pp. 813-816.
5. GU, T., H. K. PUNG, D. Q. ZHANG, **A Middleware for Building Context-Aware Mobile Services**, In Proceedings of IEEE Vehicular Technology Conference, 2004.
6. <http://awarehome.imtc.gatech.edu/>
7. <http://oxygen.csail.mit.edu/>
8. ICT Standard Weekly, **Design and Construction of Intelligent Monitoring System for CCTV**, IITTA, 2013.
9. KANG, S., J. SOHN, **A Method of Feasibility Estimation for Proactive Robot Services**, KISS 2009.
10. KIM, H., K. W. LEE, Y. H. SUH, J. M. CHO, **Context-Aware Server Framework for Network-based Intelligent Robot**, Proceedings of Conference of Korea CAD/CAM, 2006.
11. KIM, J. W., J. J. YANG, **Agent based Context Awareness Model and Applications**, Journal of KISS, No. 24, vol. 10, 2006.

12. KIM, S., Y. Joo, Y. Lee, **Sensible Media Simulation in an Automobile Application and Human Responses to Sensory Effects**, ETRI Journal, vol. 35(6), December 2013, pp. 1001-1010.
13. KIM, Y. G., H. K. KIM, S. H. YOON, S. G. LEE, K. D. LEE, **Home Security Robot based on Sensor Network**, SICE-ICASE International Joint Conference, 2006.
14. KOVACS, G., D. BOGDANOVA, N. YUSS UPOVA, M. BOYKO, **Informatics Tools, AI Models and Methods Used for Automatic Analysis of Customer Satisfaction**, Studies in Informatics and Control, ISSN 1220-1766, vol. 24 (3), 2015, pp. 261-270.
15. LEE, K. H., C. J. SEO, **Development of User-friendly Intelligent Home Robot Focused on Safety and Security**, International Conference on Control, Automation and Systems 2010, October, 27-30, 2010.
16. PARK, H. S., S. B. CHO, **Context-aware Service of Home Robot using Domain Knowledge-based Bayesian Network Modelling**, KISS conference 2010.
17. RODIC, D., E. ANDRIES, **Social Network as a Coordination Techniques for Multi-Robot Systems, Intelligent Systems Design and Applications**, Springer, Berlin, 2013, pp. 503-13.
18. SHIN, M., M. KO, Y. JU, Y. JUNG, B. LEE, **Implementation of Context-Aware Based Robot Control System for Automatic Postal Logistics**, Studies in Informatics and Control, ISSN 1220-1766, vol. 22(1), 2013, pp. 71-80.