International Journal of Bio-Science and Bio-Technology
Vol.7, No.6 (2015), pp.239-248
http://dx.doi.org/10.14257/ijbsbt.2015.7.6.24

Multiple Heart Rates Monitoring using Smart Phones

Sangho Ho[1] and Seo-Young Noh[2,*]

[1]*Department of Computer Science & Engineering, Soonchunhyang University, South Korea*
[2]*Korea Institute of Science and Technology Information, South Korea*
[1]*hsh@sch.ac.kr*, [2]*rsyoung@kisti.re.kr*

Abstract

Heart rate is one of the important vital signs that monitor cardiovascular disease or control exercise intensity. There are a lot of studies to use heart rate to monitor people's health status and provide rapid feedback. However, these studies are usually limited to monitor heart rates for an individual at a time. In this paper, we develop a system to allow simultaneous heart rate monitoring for several users who are wearing a wearable heart-rate monitor using an Android smart phone. It provides several useful functions such as setting the candidates to be monitored, monitoring heart rates by graph views, alternating graph views over several users by window touching, and storing and retrieving heart rates on the fly.

Keywords: *Heart Rate, Wearable Heart-Rate Monitor, Smart phone, Bluetooth, Monitoring*

1. Introduction

Heart rate is one of the important vital signs for monitoring people's health conditions for many cases. It can be used to decide various symptoms of cardiovascular diseases such as myocardial infarction as shown in [1,2]. The higher resting heart rate shows the higher risk factor for mortality and morbidity in healthy individuals as well as in patients with cardiovascular diseases [3-6]. In addition, heart rate can be used to control your exercise intensity because the higher heart rate means the more degree of exercise intensity[7] is. So, the user's guidance for exercise can be given using a maximal heart rate estimated individually [8], avoiding dangerous situations due to exercise excess by a continuous monitoring. Heart rate can be also used to estimate calorie consumption due to exercise[9].

There are a lot of studies[10,11,12,13,14] to use heart rate to monitor people's health status and provide rapid diagnosis in a wearable context. These studies are usually limited to monitor heart rates coming from only an individual at a time. However, it would be helpful if a health trainer can monitor simultaneously multiple heart rates coming from several people under exercise in a fitness center.

In this paper, we are interested in monitoring simultaneously heart rates coming from several people who are wearing a wearable heart-rate monitor. As a related work, Bodypro_PAPS[15] is developed by a company called Dusung technology. It can monitor heart rates coming from several students. Its purpose is to identify students who are doing over-exercise. In the system, a wireless heart rate sensor is worn on the chest of a student. The sensed heart rates are transferred to administrator's notebook by the wireless connectivity of ZigBee technology. However, the system requires its dedicated receivers for getting the sensed data and has the range limitation by the ZigBee connectivity. In

[*] Corresponding Author

ISSN: 2233-7849 IJBSBT

addition, it is less convenient and less portable due to the usage of a notebook rather than a smart phone.

In this paper, we will develop a system to simultaneously monitor multiple heart rates coming from several people using smartphones. We just need a wearable Bluetooth-enabled heart rate monitor as an external device. We thus can monitor multiple heart rates by just using smartphones without the dedicated separate expensive equipment. We use the heart rate monitor, HRM-1000, which is developed by H3system[16]. The monitor uses PPG(Photoplethysmographic) sensor. It is unobtrusive and non-invasive in that it detects PPG signal using an optical sensor applied on the epidermis. So, it is preferred as the usage of sports compared with patch-style ECG or pressure heart-rate monitors which are obtrusive. HRM-1000 also provides a good accuracy by applying motion artifact detection algorithm.

Due to the Bluetooth connectivity between a heart rate monitor and a smartphone, the application of the system is limited to its range. However, we could employ a healthcare server to eliminate the limitation. In that case, user's smartphone has a role of the gateway between heart rate monitors and the healthcare server. Even without employing the healthcare server, the system can be also applied to comparatively small areas such as fitness centers.

In Section 2, we introduce HRM-1000 briefly which is used as a heart rate monitor in this paper. In Section 3, we describe the multiple heart-rate monitoring system using the monitor and a smart phone in detail. In section 4, we describe the details of the system implementation. In Section 5, we apply the system to an Android smart phone. Finally, we mention the conclusions in Section 5.

2. HRM-1000

Figure 1 shows the HRM-1000 developed by H3System, which is a wrist-worn heart rate monitor. It consists of three parts: main unit, sensing unit, and glove. (a) shows the main unit and the sensing unit. They are connected by cable. The main unit has CPU, display, operation buttons, and charging terminal. In (a), ④ is a button for starting and stopping measuring heart rates, and ⑤ is a button for the Bluetooth connection. ⑦ is the sensing unit which detects the amount of the reflected light compared to the acting light and transmits it to the main unit. The change of the light reflectance is analyzed to calculate heart rate in the main unit. (b) shows the wrist-worn HRM-1000 using the hand glove. It is used to install the main unit and the sensing unit. The sensor unit is positioned flat against the fleshy part of the finger. The display of ③ is used to display the measured heart rates.

(a) **(b)**

Figure 1. HRM-1000

HRM-1000 is connected to its counterpart by Bluetooth, which is a smart phone. Its speed is 115,2000bps. The HR packet is defined to communicate between HRM-1000 and its counterpart. Its structure shown in Figure 2 has the seven parts: 'Start', 'Packet Size', 'command', 'Device', 'DATA', and 'CK'. The number below each part represents its size in bytes. Note that the size of 'DATA' can be arbitrary. 'Start' represents the start of the packet, 'Packet Size' represents its whole size, 'command' represents commands to be used to communicate between HRM-1000 and its counterpart. 'Device' represents who sent the packet between the monitor and its counterpart. 'DATA' is used to send actual data, including the heart rate value as well as the identifier of the monitor. 'CK' represents the check sum to be used to detect the transmission error. The monitor sends the new packet to its counterpart every 5 seconds, and its counterpart analyzes the packet and extracts the heart rate value from it. Thus, heart rates can be monitored every 5 seconds in the counterpart.

Start	Packet Size	command	Device	DATA	CK
(1)	(2)	(1)	(1)	(N)	(1)

Figure 2. HR Packet Structure

3. System

In this section, we first describe the overall architecture of our system. We then describe components of the system in detail, and describe how heart rates coming from several heart rate monitors are manipulated on a smart phone.

3.1. The Overall Architecture

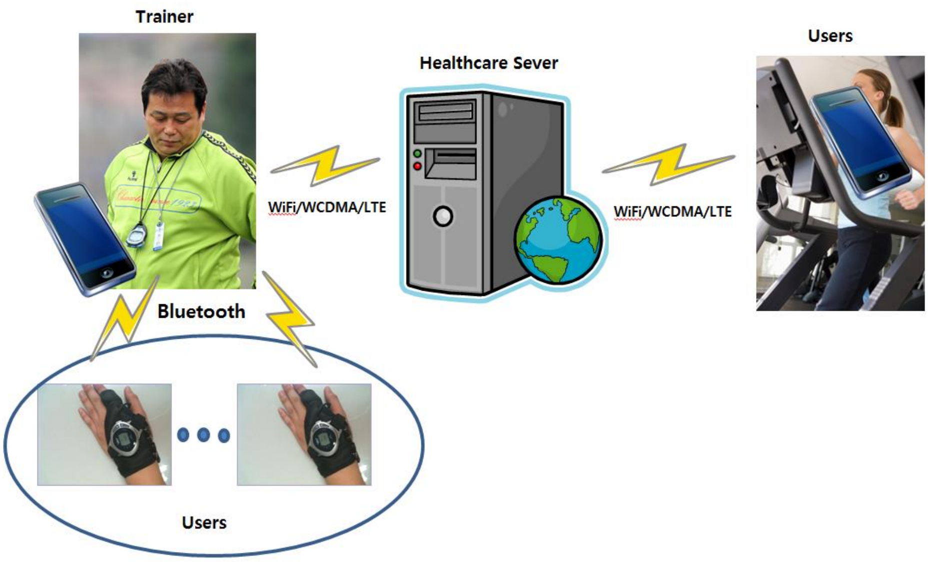


Figure 3. The Overall Architecture of the System

Figure 3 shows the overall architecture of the system developed in this paper. It consists of Health trainer, Healthcare Server, and Users. Each user wears a heart rate monitor on his wrist, HRM-1000. The health trainer sets users to be monitored by identifying their devices on his smartphone. Each heart rate monitor periodically measures heart rates and transfers them to the trainer's smartphone. The system provides the trainer with the function of simultaneously monitoring multiple heart rates coming from several users. It also can provide the users with useful feedback such as alarming upon the occurrences of dangerous heart rates. The measured heart rates can be also transferred to the healthcare server through the trainer's smartphone, and can be stored there. Users can later access the healthcare server to retrieve and review their own heart rates.

3.2. System Components

Figure 4 shows the components of our system. They are divided into two parts: the client and the server. The client part will be installed on the smart phone, and the server part on the healthcare server. These components are represented in Android activities, Java objects, or JSP(Java Server Pages). In the client side, 'Client Controller' gets user's requests, takes the charge of having them processed by invoking other components, and gives the result back to the client. 'HR Monitoring' provides the health trainer with a function of simultaneously monitoring heart rates for several users. He chooses candidates to be monitored using 'Set Candidates'. 'Bluetooth Interface' takes charge of communicating with the heart rate monitors. It also analyzes packets coming from the monitors and extracts the heart rate values from them.

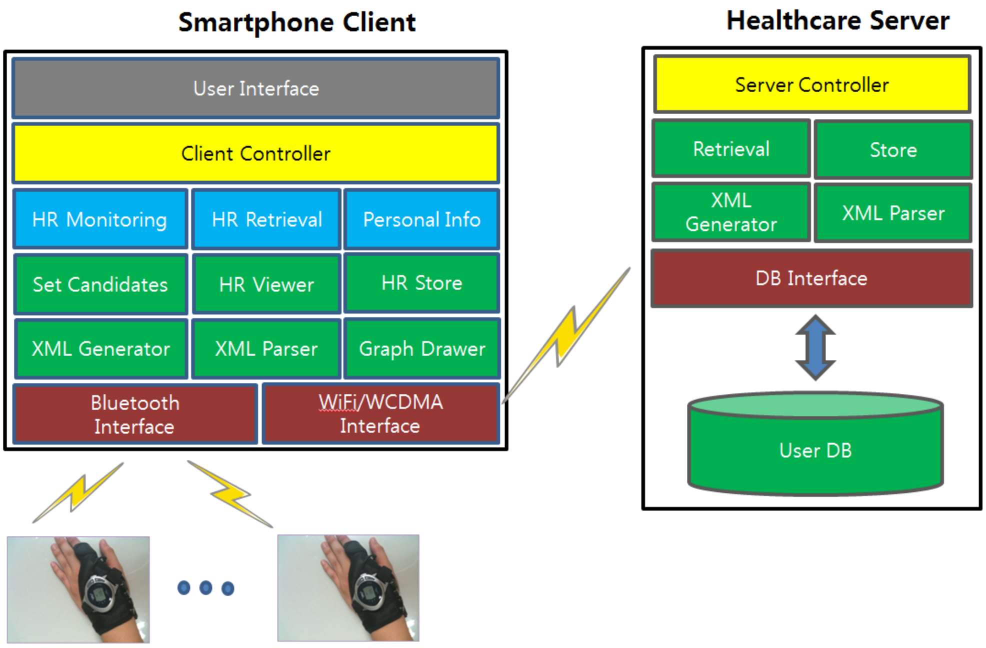


Figure 4. System Components

'HR Viewer' provides a function of viewing heart rates by user in a form of their values or graph trends. The view of trends of heart rates by graph is given using 'Graph Drawer'. Heart rates can be stored to the database of the server through 'WiFi/WCDMA Interface' using 'HR Store' upon the occurrence of abnormal level of heart rates. The stored heart rates can be retrieved later using 'HR Retrieval'. 'XML Generator' and 'XML Parser' are used to transfer messages between the client and the server for efficiency. 'XML Generator' is used to wrap the data to be transferred in a XML document and 'XML Parser' is used to extract the data from the document. Usually, the store/retrieval of heart rate values and personal information include a lot of the structured data. Thus, those data can be represented and transmitted effectively by XML documents.

In the sever side, ‘Server Controller’ mainly gets the requests coming from the client processed and gives the results back to the client. Personal information and heart rate values for users are stored in the user DB. The store and retrieval for this information are done by ‘Store’ and ‘Retrieval’, respectively. As in the client side, ‘Store’ invokes ‘XML Parser’ and ‘Retrieval’ invokes ‘XML Generator’. Either components can access User DB through ‘DB Interface’.

3.3. HR Monitoring Flow

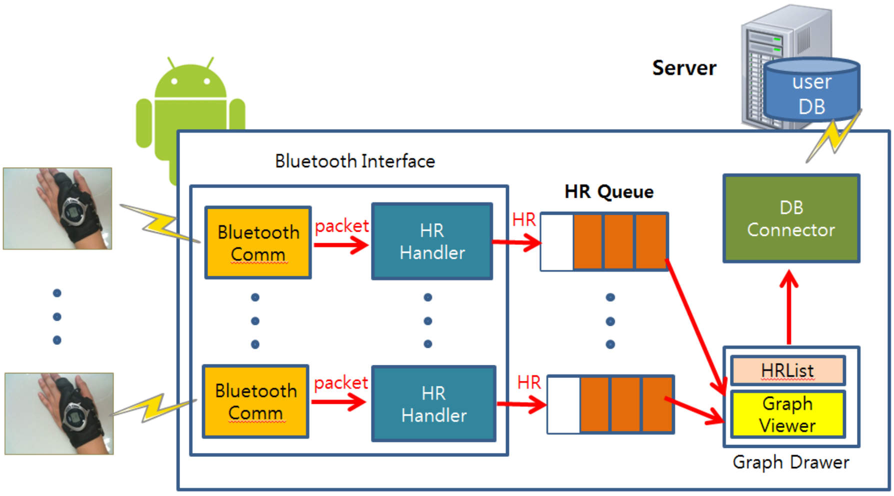


Figure 5. HR Monitoring Flow

Figure 5 shows how heart rates coming from several heart rate monitors are manipulated on the trainer’s smart phone. In this process, ‘Bluetooth Interface’, ‘HR Viewer’, and ‘DB Connector’, which is a subcomponent of ‘HR Store’, are involved. ‘Bluetooth Interface’ and ‘HR Viewer’ are Android activities. Both of them are simultaneously activated when an item of ‘HR Monitoring’ is clicked in the main menu of our application shown in (a) of Figure 6. ‘Bluetooth Interface’ works in the background, taking charge of communicating with the heart rate monitors continuously. It creates two objects of ‘Bluetooth Comm’ and ‘HR Handler’ for each heart rate monitor, and register the handler to ‘Bluetooth Comm’. It has two threads of ConnectThread and CommunityThread, which are involved in the communication with the monitors. ConnectThread takes a part of connecting to its counterpart monitor. When the connection is done successfully, CommunityThread starts to get packets from the monitor. It then passes the packets to the handler which is registered to ‘Bluetooth Comm’. The handler extracts heart-rate values from the packets by analyzing them and put them in the associated HR Queue. Note that each of ‘Bluetooth Interface’ has its own HR Queue and heart-rate values coming from the interface are put in the queue.

‘HR Viewer’ is also activated with the activation of ‘Bluetooth Interface’. Within the viewer, a thread of getHRThread is executed repeatedly every 5 seconds. Note that the associated heart rate monitors send their packets to their counterparts every 5 seconds. For each repetition, the thread accesses each HR queue, gets the new heart rate value from the queue, and put it in HRArrayList, and invokes a method of updateGraph(). HRArrayList consists of an array of HRLists, each of which contains a list of the 12 recent heart rate values for a user. updateGraph() gets an HRList of the currently focused user from HRArrayList, and calculates graphHeightList using the list. graphHeightList represents a list of heights which will be actually drawn on the graph, which is calculated relatively on the absolute height of the graph. updateGraph() then creates an object of ‘Graph Drawer’, and sends three parameters of UserID, HRList, and graphHeightList to the object. UserID represents an identifier of the currently focused user. 'Graph Drawer' draws a graph using its parameters which shows trends of heart rates. Since getHRThread invokes updateGraph() every 5 seconds, the graph is updated accordingly every 5 seconds. Note that the HRList used by the graph drawer is also updated every 5 seconds. You are given a touch event on the graph which allows several users to simultaneously be monitored conveniently. When the graph is touched leftwards(rightwards), the currently focused user is changed to the user on the left(right) of the previous user in order, resulting in that the graph is redrawn for the newly focused user.

While the graph is monitored, some heart rate values can be stored to the server using 'DB Connector' if necessary. They are of some period of time before and after the occurrence of abnormal level of heart rates. Heart rate values aggregated during the period are wrapped in an XML document and sent to the 'Store' component of the server.

4. Implementation

As mentioned earlier, the components of the system designed in this paper are divided into two parts: the server side and client side. Components on the client side were designed as Android activities and Java Objects, and implemented using Android SDK 2.2 plugged in Eclipse. The server was constructed in Windows Server 2008 using Apache Tomcat 5.5. Components on the server side were designed based on the MVC(Model-View-Controller) architecture and implemented using Java objects and JSP(Java Server Page) under an eclipse Galileo package. The user database was implemented using MySQL 5. The retrieval of user's heart rate record requires the transmission of the large amount of data between the server and the client. This data transmission is efficiently done using XML. The SAX parser in J2SE 6.0 was used for processing XML documents.

5. Execution

The system components on the client side were installed and executed on the Galaxy Note. We tested the system over the two people, each of whom wears HRM-1000. Figure 6 shows an execution example for the system. Note that Figures shown here are captured by screenshots on the smart phone. (a) shows the main view of the system. When an item of 'HR Monitoring' is clicked, the view of (b) comes up. In (b), we can find and select candidates to be monitored. You can find and select some candidates by filling their name in the blank box. Alternatively, you can just click a button of 'Search'. The view of (c) then comes up, showing a list of members registered using 'Personal Info'. You can check some candidates in the member list. After that, you click the button of 'Add' at the bottom of the member list. The view of (d) then comes up, showing the selected member list. You can delete some candidates by checking them and clicking the button of 'Delete'. After confirming the candidate list, you click the button of 'Next'. The view of (e) then comes up, showing heart-rate values for all the selected candidates. Each candidate is associated with a single box. It is divided into two parts: the up and the down. The name of the candidate is displayed at the down part. The heart-rate value is displayed at the up part. In this way, we can monitor heart-rates for several people simultaneously. We also monitor trends of heart rates in a form of graph by clicking the graph icon at bottom in (e). In the form, only one user is displayed at a time. However, the graph views can be alternated by touching the graph leftwards or rightwards, as shown in (f). The graph view can be temporarily stopped and resumed using the menu window in (g). The window is popped up the graph view when the menu button is pushed on the smart phone. If necessary, some heart rate values can get stored to the server by clicking an item of 'Store' in the window. They can be retrieved by clicking an item of 'HR Retrieval'. The storage and retrieval of heart rate values can be useful for checking user's health status during exercise.

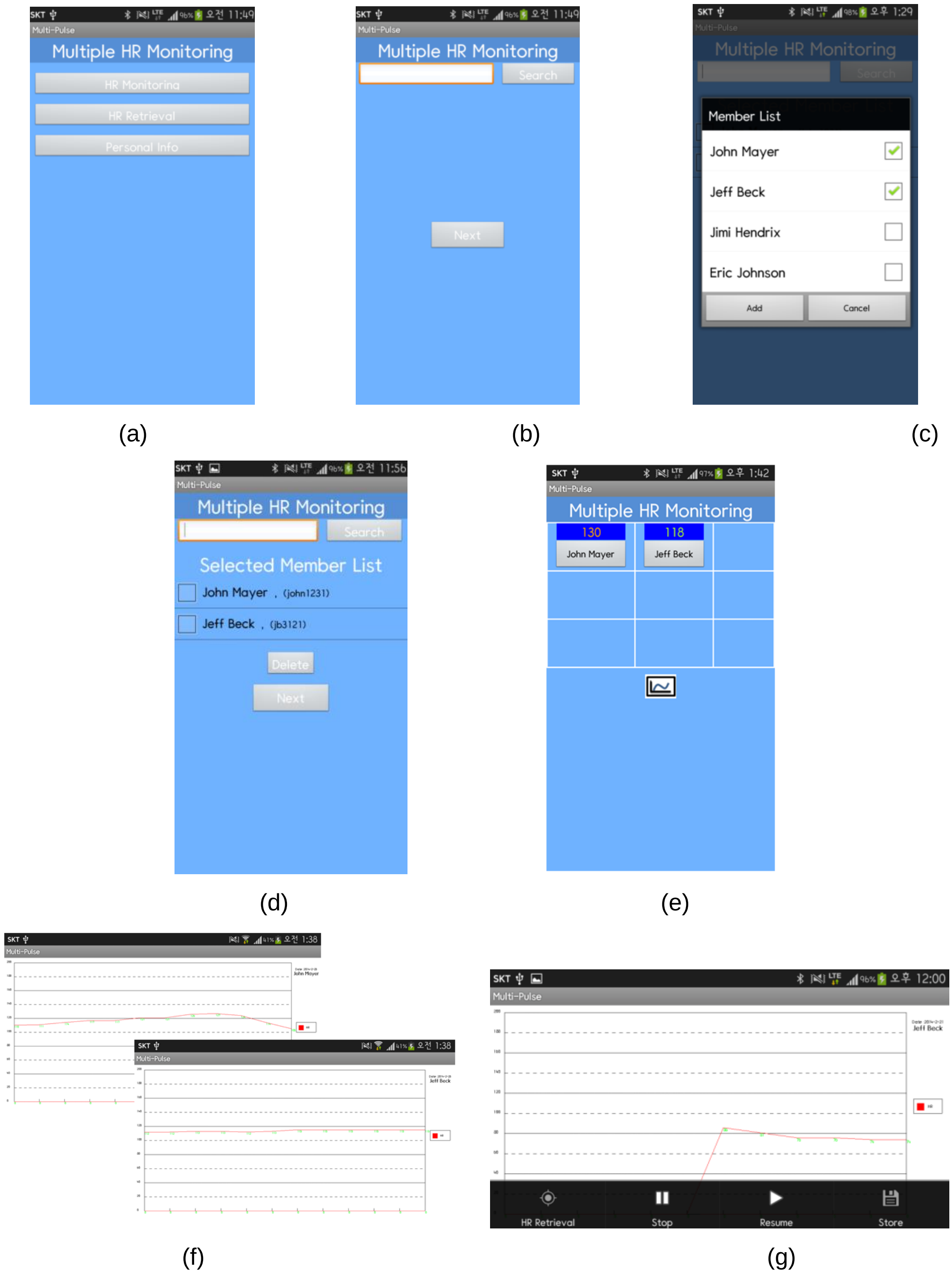


Figure 6. Examples by Execution

6. Conclusions

We developed a system to simultaneously monitor heart rates coming from several people using smartphones without the dedicated separate expensive equipment. In the current system, due to the Bluetooth connectivity between the heart rate monitor and smartphone, the application of the system is limited to the small area such as a fitness center, and the number of candidates to be simultaneously monitored is limited to just seven people. However, this limitation could be resolved by having heart rate monitors sending the measured heart rates to the healthcare server via their user's smartphone and having the trainer getting user's heart rates through the access to the healthcare server. It is realized when the connectivity between the heart rate monitor and a smart phone is replaced by WiFi or WCDMA/LTE via the healthcare server. In that case, the heart rate monitoring would be given in a quasi-realtime mode.

References

[1] K. A. Eagle, M. J. Lim, O. H. Dabbous, K. S. Pieper, R. J. Goldberg, F. V. Werf, S. G. Goodman, C. B. Granger, P. G. Steg, J. M. Gore, A. Budaj, A. Avezum, M. D. Flather and K. A. Fox, "A validated prediction model for all forms of acute coronary syndrome: estimating the risk of 6-month postdischarge death in an international registry," The Journal of the American Medical Association, vol. 291, no. 22, **(2004**), pp.2727-2733.

[2] R. Marchioli, F. Avanzini, F. Barzi, C. Chieffo, A. D. Castelnuovo, M. G. Franzosi, G. Geraci, A. P. Maggioni, R. M. Marfisi, N. Mininni, G. G. Nicolosi, M. Santini, C. Schweiger, L. Tavazzi, G. Tognoni and F. Valagussa, "Assessment of absolute risk of death after myocardial infraction by use of multiple-risk factor assessment equations: GISSI-Prevention mortality risk chart," European Heart Journal, vol. 22, no. 22, **(2001)**, pp. 2085-2103.

[3] A. Diaz, M. G. Bourassa, M-C Guertin, and J-C Tardif, "Long-term prognostic value of resting heart rate in patients with suspected or proven coronary artery disease," European Heart Journal, vol. 26, no. 10, **(2005)**, pp. 967-974.

[4] X. Jouven, J-P Empana, P.J Schwartz, M. Desnos, D. Courbon and P. Ducimetiere, "Heart-rate profile during exercise as a predictor of sudden death", New England Journal of Medicine, vol. 352, no. 19, **(2005)**, pp. 1951-1958.

[5] K. Fox, J. S. Borer, A. J. Camm, N. Danchin, R. Ferrari, J. L. Sendon, P. G. Steg, J-C Tardif, L. Tavazzi and M. Tendera, , "Resting heart rate in cardiovascular disease," Journal of the American College of Cardiology, vol. 50, no. 9, **(2007)**, pp. 823-830.

[6] P. Palatini, "Heart rate: a strong predictor of mortality in subjects with coronary artery disease," European heart journal, vol. 26, no. 10, **(2005)**, pp. 943-945.

[7] C. Knoepfli-Lenzin, B. Haenggli and U. Boutellier, "Optimised heart rate formulae to monitor endurance training in sedentary individuals", Journal of Sports Science, vol. 32, no. 6, **(2014)**, pp. 557-562.

[8] C. E. Garber, B. Blissmer, M. R. Deschenes, B. A. Franklin, M. J. Lamonte, I. M. Lee, D. C. Nieman and D. P. Swain, "Quantity and quality of exercise for developing and maintaining cardiorespiratory, musculoskeletal, and neuromotor fitness in apparently healthy adults: Guidance for prescribing exercise," Medicine and Science in Sports and Exercise, vol. 43, no 7, **(2011)**, pp. 1344-1359.

[9] Calories Calculator, http://www.calories-calculator.net/

[10] Y. Wen, R. Yang and Y. Chen, "heart rate monitoring in dynamic movements from a wearable system", Proceeding of the 5th International Workshop on Wearable and Implantable Body Sensor Networks, Hong Kong, China, **(2008)** Jun 1-3.

[11] H. Weghorn, "Applying mobile phone technology for making health and rehabilitation monitoring more affordable", Proceedings of the 4th Biosignals and Biorobotics Conference, Rio de Janeiro, Brazil , **(2013)** Feb 18-20.

[12] S. Dagtas, G. Pekhteryev and Z. Sahinoglu, "Multi-stage real time health monitoring via zigbee in smart homes", Proceeding of the 21th International Conference on Advanced Information Networking and Applications Workshops, Niagara falls, Canada, **(2007)** May 21-23.

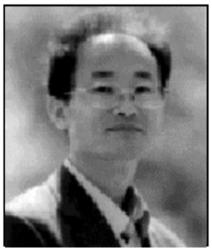
[13] A. Pantelopoulos and N. G. Bourbakis, "Prognosis-a wearable health-monitoriing system for people at risk: methodology and modeling. IEEE Trans. on information technology in biomedicine, vol. 14, no. 3, **(2010)**, pp. 613-621.

[14] R. Sukanesh, P. Gautham, P.T. Arunmozhivarman, S.P. Rajan and S. Vijayprasath, "Cellular phone based biomedical system for health care", Proceeding of the International Conference on Communication Control and Computing Technologies, Ramanathapuram, India, **(2010)** October 7-9.

[15] Bodypro_PAPS, http://www.bodypro.co.kr.

[16] HRM-1000, http://www.h3system.co.kr.

Authors



Sangho Ha received B.S., M.S., and Ph.D. degree in the department of computer science from Seoul National university in Korea in 1988, 1991 and 1995, respectively. He is a professor in the department of computer science & engineering at Soonchunhyang university since 1997. He was a postdoctoral scientist in the computation structure group of the laboratory for computer science at Massachuetts Institute of Technology in 1996. He also was a visiting professor at Iowa State university and university of Connecticut in 2005 and 2014, respectively. His interests include u-Healthcare, context-aware services and ubiquitous computing.

Young Noh is a principal researcher in National Institute of Supercomputing and Networking at Korea Institute of Science and Technology Information and an associate professor at Korea University of Science and Technology. He is leading the development of virtual cluster system called vcluster in conjunction with KISTI-FNAL joint project. Before joining the institutes, he worked for LG Electronics in the fields of embedded database systems and Linux mobile platforms. He received his B.E. and M.E. in Computer Engineering from Chungbuk National University in Korea and his M.S. and Ph.D. in Computer Science from Iowa State University, respectively. His research interests are including scientific data management, cloud & scientific computing, Linux platforms, databases, and natural language processing.

