International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.9 (2015), pp.299-308
http://dx.doi.org/10.14257/ijmue.2015.10.9.31

Effective Use of Android Sensors Based on Visualization of Sensor Information

Young Jae Lee

Faculty of Smartmedia, Jeonju University, 303 Cheonjam-ro, Wansan-gu, Jeonju-si, 560-759, Korea
leeyj@jj.ac.kr

Abstract

Using acceleration sensor on smart phone, we derive data on x, y, and z axis on spatial coordinate. With the proposed method we signalize this sensor information. The result is inferred and analyzed to detect the exact movement information along with the size of the directional information to the right, to the left, to the up and to the bottom. It is then output on the graph to propose the sensor application algorithm to visualize movement information. To verify the data gotten from the proposed method, we make 'third-person, top-view game' and apply it to the purpose. By using orientation sensor (acceleration sensor and magnetic sensor) we move the object in cyber space, and with the proposed method, we get movement information and directional information and apply them in comparison. After application, the proposed method provided consistent and correct movement information whether the object was moving or not, enabling us to identify correct information on the movements of the object in cyber space. For correct detection of sensor data, we used Kalman filter. To get stable data, we collected the data at regular intervals. We could verify the performance and validity of the proposed algorithm through experiments.

Keywords: *acceleration sensor, orientation sensor, movement information, directional information*

1. Introduction

In recent years, as technologies advance in the areas of sensors, computing power, and interface, smart interaction is getting possible between human and digital devices. Smart phone has established its status as the key success case in having effectively enabled interaction between human and digital devices [1-10].

In the market for smart phone, wearable devices, IoT (internet of things), and healthcare are getting recognized as core contents and emerging as the blue ocean in the new ICT (Information & Communication Technology) market. All of these are possible thanks to innovation in sensor technology including smart sensors. This kind of innovation in sensor technology signifies that communication is getting possible between human and machine including computers, robots, electronic devices, and automobiles. Another name for sensor is 'detector', meaning parts which transform temperature, light, noise, and pressure into certain forms of signals. Sensors are the electronic or mechanized version of human five senses. For example, camera on smart phone represents vision; microphone, hearing; touch screen, touch, and *etc.* In addition, GPS tells objective locations, and acceleration sensor and gyroscope realize the sense of balancing [2-3]. Image sensor, the vision of the camera, makes it possible to take clear pictures with a smart phone, and various functions are being tried including hand-shaking modification. Also, light sensor which senses light in the surrounding area minimizes battery waste and improves screen readability for the user. As for the touch sensor, even a slight touch of the screen activates the input function. In addition, the support of multi-touch function

ISSN: 1975-0080 IJMUE

enables enlarging or reducing along with other various motions. All in all, intuitive control of the device has become a very important outcome. Proximity sensor uses electronic power without physical contact to sense presence/absence, passage, continuous flow, congestion, and controls locations. It performs various applications by detecting if the object is present in the vicinity. There are also GPS sensor with navigation function, geomagnetic sensor which senses the movements of the smart phone, weight sensor, acceleration sensor, gyro sensor, and *etc.* [1-5].

In this paper, among various sensors, acceleration sensor, which senses acceleration rate and impact force of a moving object, has been used to propose various applications and algorithm for natural interface. The effectiveness has been proven through experiments.

2. Android Sensor

Android developers provide functions related to various sensors to supply better and more sensitive applications for users. Roughly speaking, these sensors are divided into motion sensor, location sensor, and environment sensor. Kinds of sensors have each different trait. [5-7].

Motion sensors consist of various sensors which can detect the movements of the device. Since they can detect movements such as tilt, vibration, and rotation, motion sensors are being widely used for interactive games which can provide sense of reality. Each sensor event in motion sensors returns multi-dimensional array value on x, y, and z axis.

Location sensors include magnetic field sensor and orientation sensor which help to identify the physical location of a smart phone. They also provide proximity sensor which helps to judge the nearness to the device. Geomagnetic sensor and acceleration sensor, in particular, provide relative location of the device from the North Pole when they are used together. Therefore, they are frequently used for application programs which require GPS function.

Environment sensors monitor various environmental factors around the smart phone. With environment sensors, we can detect the temperature of the device and the surroundings. Also, through environment sensors, we can know luminance, humidity, and pressure around the device.

2.1. Acceleration Sensor

Acceleration sensor measures movement (acceleration) within the unit time using various electronic elements bearing various characteristics-either using 'Piezoelectric Element' proportionate to the size of the movement (acceleration or vibration) or using element which changes its capacitance according to movements [5-8]. Detected acceleration is transformed into electric signal proportional to the size, and processed into information which shows movements in certain directions on smart phone. The processed information is measured by two forces-slope and acceleration. Both forces can be defined in terms of acceleration. Gradient is attitude to weight which is the concept in acceleration of gravity, and it appears as weight applied on each axis. With these kinds of acceleration sensors, screen transition such as 'portrait mode' or 'landscape mode' is possible. Besides, we can expect various applications in regard to slope or movement such as 'tilt' or 'shake'. Android platform supports acceleration sensor in the form of 'Sensor. TYPE_ACCELERATION'. Here, the array of the three accelerations measured on x, y, and z axis (value[0], value[1], value[2]) are used to generate the return value in the unit of m/s^2.

3. Proposed Algorithm

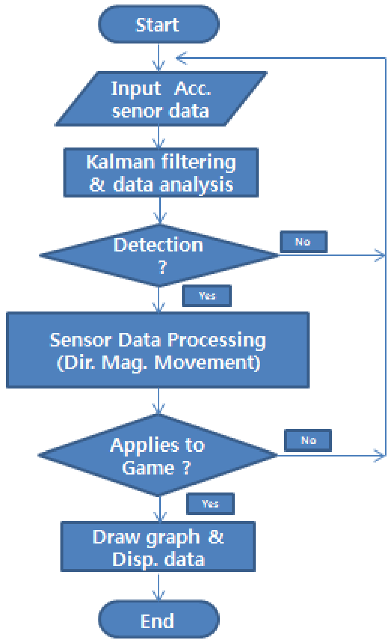


Figure 1. Flowchart of the Proposed Method

Sensor data are prone to variability and frequent changes. Therefore, before being used as data, pre-processing through filters is needed. In addition, for stable inputting of data, it is necessary to get data input at regular intervals. In this paper, we use 'Kalman filter' because it can provide excellent performance in optimization compared to other filters. Through 'Kalman filer', we realize data signal processing on x, y, and z axis. After filtering of data on each axis, threshold is determined to distinguish movement data. Especially, since the acceleration sensor outputs minus data depending on directions of the slope, two threshold values are applied for judgment. The judged data are normalized to 1, and the maximum data output is quantified to 1. Each value is the variation on each axis, and these values are used to normalize the values on x, y, or z axis. We get resultant force vector on each axis and calculate directional vector on each axis (Dir:direction, Mag:magnitude, Disp:display). In this paper, to get screen output on two-dimensional x and y axis, we get resultant force vector on x and y axis and find information for directional angle. To verify the movement volume and the direction information data, we make three-person top view game and compare movements of the virtual object in game background with the directional information. The sensor used here is the orientation sensor based on accelerator sensor and magnetic sensor. Through various experiments, we get the error range of the data and compare the experiment results and visualize the directional information and the size information in graphs if they are within the error range (Error_data < 7). For this acceleration sensor, since movement sensing is possible when the device is parallel to the ground, we assign offset value in the movement direction, especially in the counter-clockwise direction on x axis, and additionally, detailed tuning is necessary for movement information. Even for the movement information of the two virtual objects in game, the convenient angle for the user to handle the device is used as offset to control the movement information on each axis. Through this visualization process, the output of the acceleration sensor can be expressed visually on two-dimensional x and y axis in terms of size and directional information. And this information can be used in various areas including game and education interfaces.

4. Experiment

4.1. Experiment 1

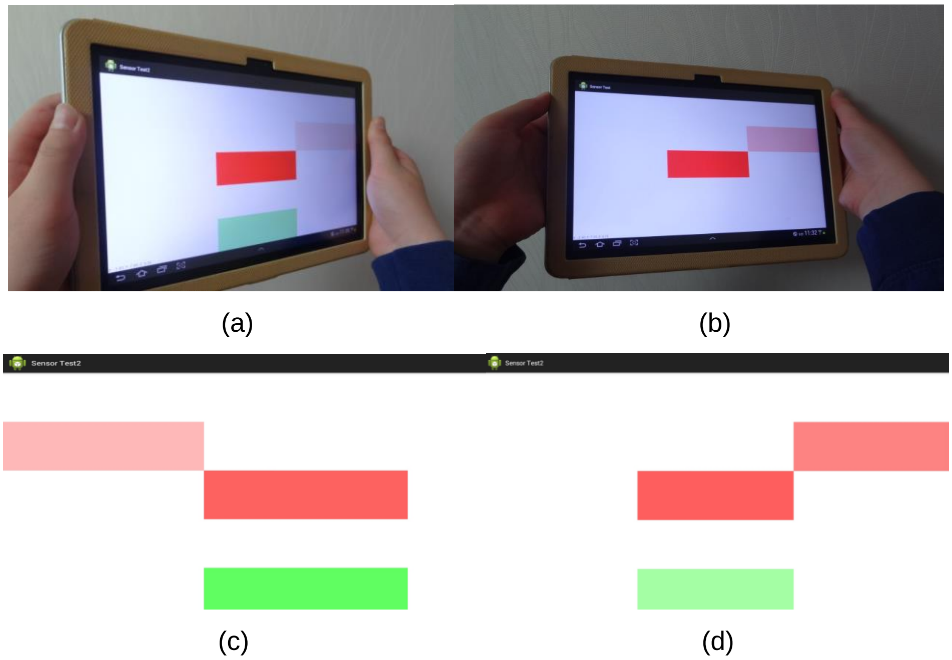
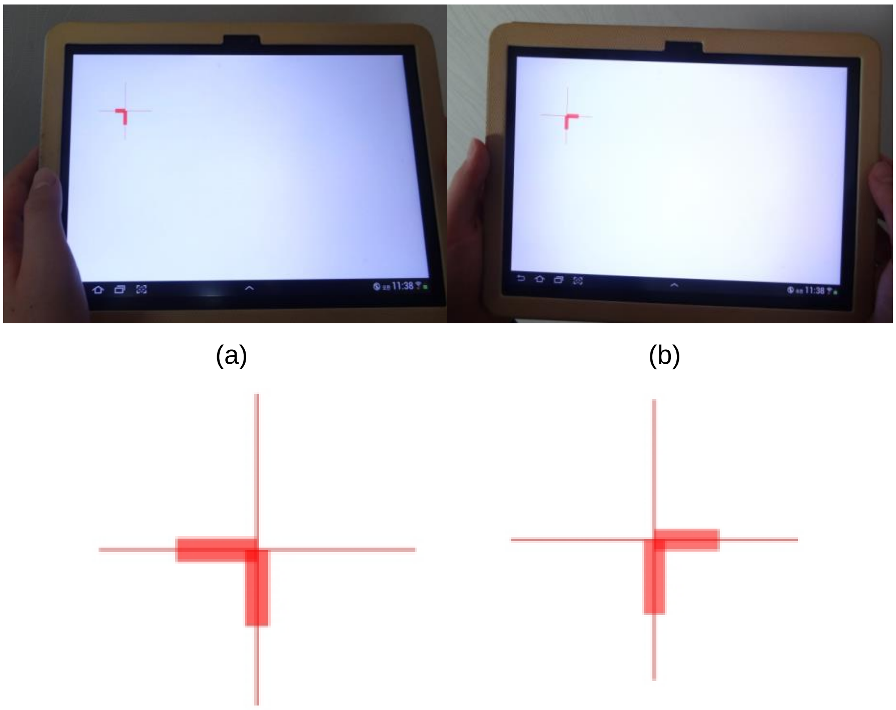


Figure 2. Resulting Image of Experiment 1

In Experiment 1, the acceleration sensor is used for output using the conventional algorithm [9]. Here, information on z axis is added for applications. Figure (a) is the result image based on the conventional algorithm where there are four arranged view grids for x and y axis when the movements occur up & down and right & left. According to directions, the grid background rate is adjusted, and the slope information appears in red color. In Figure (b), the information on the z axis has been added and the grid background rate is adjusted based on 6 directions. And the slope information is output in red and green colors. In these experiments, when the device is slanted, the side with low slope starts to change into red and green colors. And at the point where the direction in which one spot of the device stands upright, the perfect red and green colors are output. Figure (c) and (d) show the front output of the experiment device. Figure(c) is the output result of the slope on each axis when the user moves the device to the left. Figure (d) is the sensor output when the device moves to the right. Among the information, the z-axis information gets perfect green color when the front of the device is horizontal. We can see that it does not have a substantial effect on the angle usually adopted by the user. By using alpha-value depending on slope, we could partly visualize the result, but detailed information was hard to get. And even after adding the z-axis data, it did not affect the angle usually adopted by the user. In particular, when the movement is toward the resultant force vector on x and y axis, we could not detect the exact direction.

4.2. Experiment 2

To overcome the shortcomings of experiment 1, in experiment 2, the maximum data value on each axis has been normalized to 1, and we show the output on x and y axis through visualization.



(a) (b)

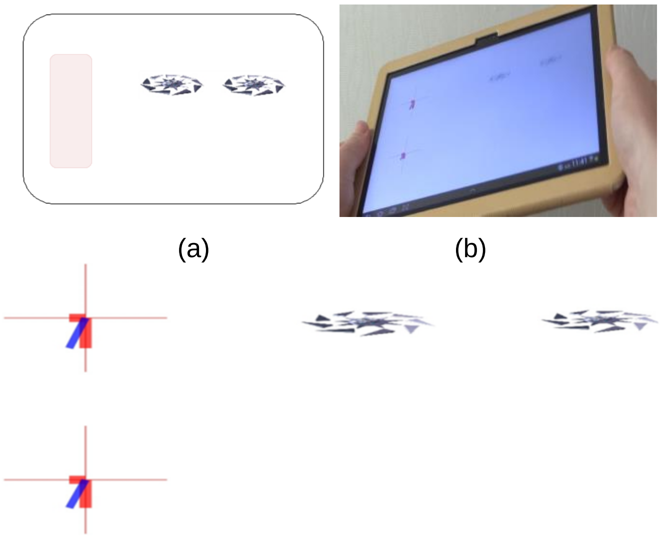
(c) (d)

Figure 3. Experiment Image and Resulting Image

Figure (a) and (b) show experimenting with the device. Figure (c) and (d) are the result images of Figure (a) and (b). As we are seeing in Figure (c) and (d), the slope information on each axis is normalized for output and the entire size was possible to be output at certain rate. We could also identify correct slope information size on each direction. But the slope value is different when the device has been moved in the counter-clockwise direction on the x axis from the clockwise movement where the variation according to the slope directions is small by the trait of the sensor. To display this kind of value, we need offset for effective modification. And we also need to show detailed slope information by using directional vector which is the resultant force vector of the directional vectors on each axis.

4.3. Experiment 3

In Experiment 3, the problems found in experiment 1 and 2 are dealt with by applying the proposed algorithm, and for application purpose, we produce a game. Especially, we express the slope information of the x and y axis as resultant force vector to track the movement direction information and the movement direction of the object induced by the resultant force. By applying this information to the two characters in cyber space, we check out the result.



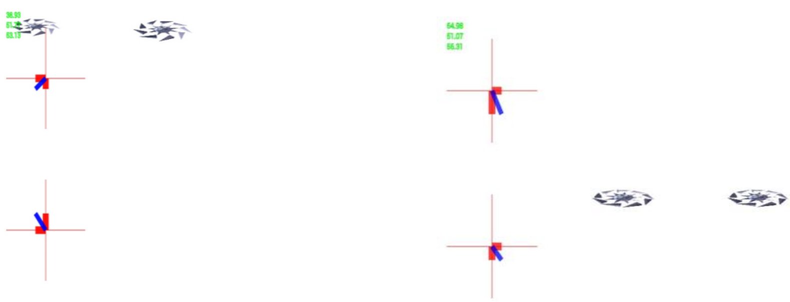
(a) (b)

(c) (d)

Figure 4. Image of Game Design and Motion Detection

Figure (a) is the game configuration to realize the proposed algorithm. We output two virtual objects in the white background and apply to them the sensor information on x and y axis from the orientation sensor using acceleration sensor and magnetic sensor. Figure(b) shows the user experimenting with the device in which when the device is moved to the x and y axis from the user position, the proposed algorithm detects movement information on each axis along with the resultant force vector and the output is visualized. Here, the virtual object moves according to the movement data of the orientation sensor. The graph on the upper left in Figure(c) considered directionality and size of the sensor output. The graph on the bottom left considered the characteristics of the sensor output and visualized direction and size by using the proposed algorithm. Where the movement is in counterclockwise direction on x and y axis, both output graphs well show the movement information of the device. And we could see the virtual object also moving in the same direction. The movement data information on the x and y axis is marked in red, and the resultant force vector is visualized in blue. The object used in the game is composed of 7 frames so that dynamic animation can possibly be materialized. Also, to check out the exact movement in cyber space, two identical objects have been designed.

4.4. Experiment 4

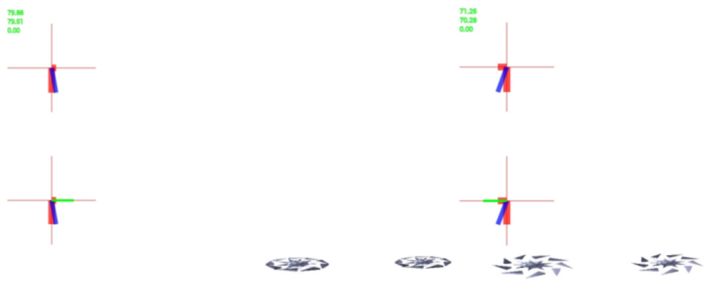


(a) (b)

Figure 5. Movement Detection from the Upper Left to the Bottom Right Direction

In experiment 4, the directional information (the tangent value) has been added to the result image of the object movement in cyber space. Experiment 4 detects the movement information of the object and the sensor where the movement is from the upper left to the bottom right. In Figure (a), the user moves the device to the upper left. The state of the slope detection through sensor is visualized, and we can see that the object in the game also moves to the upper left. Here, in the first graph on which the sensor output is immediately visualized, the angle is at 38.93 degrees. But there appears an error in directionality. The second graph outputs 51.77 degrees based on the proposed algorithm, and the direction is identical with the object. The movement angle of the third virtual object is at 53.13 degrees, which shows 1.4 degrees of error which is within the allowable limit. In Figure (b), the user moved the device to the bottom right. The first graph is at 64.98 degrees which shows great errors in direction and angle. In the second graph, the angle is at 51.07 degrees, and the third virtual object of the game is at 56.31 degrees, showing the error (5.2 degrees) which is within the allowable limit. The experiment 4 measured the sensor slope with the proposed algorithm. We could see that the slope value shows the error within the allowable limit compared with the angle of the movement direction of the virtual object in game. The result is visualized in two-dimensional graph.

4.5. Experiment 5



(a) (b)

Figure 6. Detection of Consistent Positioned the Device to the Bottom Right and to the Bottom Left

In experiment 5, we detected the movement information of the virtual object and the sensor where the user consistently positioned the device to the bottom right and to the bottom left. In Figure (a), the virtual object moves at the directional angle of 0 degree to the +X axis direction. When the virtual object reaches the end of the bottom, the movement stops since there is nowhere to go in the bottom direction. In the first graph, the directional angle of the sensor slope is consistent at 79.88. In the second graph, the sensor slope is at 79.61, and the movement direction is to the +X axis where the directional angle is at 0 degree, which appears in green color. The directional angle of the third virtual object is at 0 degree. Figure (b) is the experiment to detect continuous movement to the bottom left. In the first graph, the sensor slope is output at 71.26 degrees, showing successful detection of the movement angle. Even when the virtual object is not moving, having reached the left wall, the same angle is consistently being output. The proposed second graph shows in green color that the movement directional angle of the virtual object is at 79.62 degrees and the directional angle is at 0 degree when the movement is toward the +X axis. The third virtual object is at 0 degree in directional angle. As we can see in experiment 2, with the proposed algorithm, we could detect the slope information and visualize the result in graph whether the virtual object is at 0 degree or not.

4.6. Experiment 6

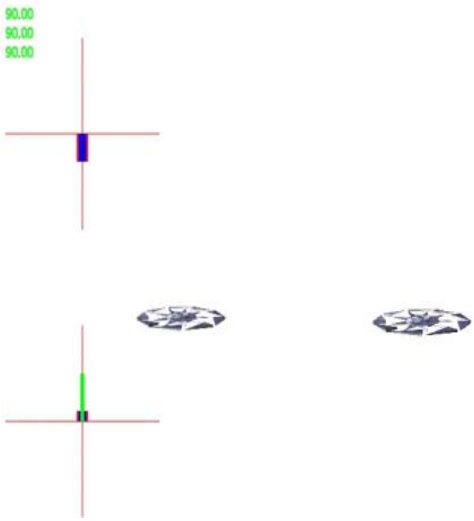
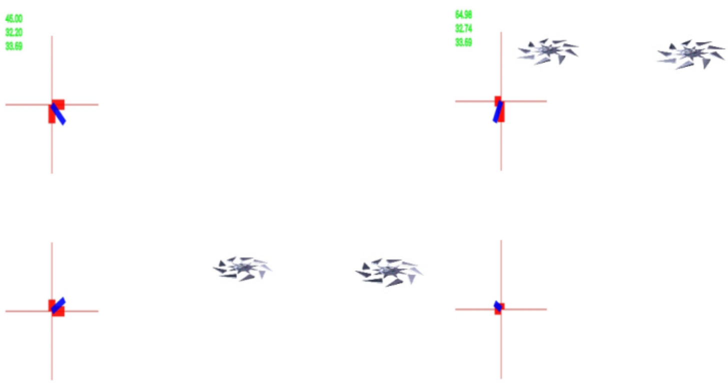


Figure 7. Detection of Consistent Positions the Device to the Upper Location

In experiment 6, while the user consistently positions the device to the upper location, the movement information of the virtual object and the sensor has been detected. In experiment 3, the virtual object moves to the +Y axis at 90 degrees and reaches the end of the upper location and stops. In the first graph, the sensor slope is at 90.00 degrees, but the direction is to the bottom. In the proposed second graph, the green color shows that the sensor slope is at 90.00 degrees, and the directional angle is at 0 degree when the movement is toward the +Y axis direction. The third virtual object is at 0 degree of directional angle. In experiment 3, as in experiment 2, by applying the proposed algorithm, we could visualize the graph using the slope information whether the slope is at 90 degrees or not.

4.7. Experiment 7



(a) (b)

Figure 8. Experiment of Movement Detection from the Right Bottom to the Upper Left Direction

In experiment 7, we detect the movement information of the virtual object and the sensor when the object in cyber space moves from the right bottom to the upper left. In Figure(a), the user moves the device to the upper right. We can visually check the state of the sensor detection and see that the game object is also moving to the upper right. In the first graph of the sensor output, visualization shows 45.00 degrees, but the direction shows an error. In the second graph, with the proposed algorithm, the graph is output at 32.00 degrees, and the direction appears the same with that of the object. The movement angle of the third virtual object is at 33.00, showing almost the same angle with the proposed algorithm. Here, the user moved the device to the upper left. The first graph is at 64.98 degrees, showing great errors in direction and angle. In the second proposed graph, the angle is at 32.74 degrees, well expressing the sensor slope information, differently from the first graph. The third virtual object in game is at 33.69 degrees, showing similar data value with the slope value gotten from the proposed algorithm. In experiment 4, the proposed algorithm has been applied to measure the sensor slope value, the result of which showed almost identical value with the directional angle of the movement of the virtual object in game. The result has been visualized in two-dimensional graph.

5. Conclusion

In this paper, we used the acceleration sensor on smart phone and applied the data on each coordinate axis to visualize the movements, directions, and size information through the proposed algorithm. Kalman filter and regular data collection were selected for preprocessing to get stable input data. Also, to validate the effectiveness of the preprocessed data, we made the game and designed two objects in game space and applied orientation sensor data (with acceleration sensor and magnetic sensor) to realize natural and diverse movements. We calculated the movement directional information value and applied the proposed method. The result showed similar direction and size information with errors within the permissible limit. In particular, the proposed method used the natural angle for the user in handling the device, the offset value in the counter clockwise direction on the x axis, so that we can realize the natural movement of the device. When there were movements in the same direction for a certain duration, this has been calculated in angles, which has been visualized to represent directions so that we can see more exact movements of the device. We could verify the effectiveness of the suggested method through experiments. The suggested method is effective as interface in contents such as games and also as information for reacting to emergencies.

References

[1] B. S. Park and H. H. Choi, “Design and Implementation of Interactive-typed Bluetooth Device interact with Android Platform-based Contents Character”, Journal of The Korea Society of Computer and Information, vol. 19, no. 11, November **(2014)**, pp. 127-135.
[2] http://www.sciencetimes.co.kr/?news=%EC%9D%B8%EA%B0%84%EC%97%90%EA%B2%8C%EB%8F%84%EC%A0%84%ED%95%98%EB%8B%A4%EC%8A%A4%EB%A7%88%ED%8A%B8-%EC%84%BC%EC%84%9C
[3] http://www.bodnara.co.kr/bbs/article.html?num=102137
[4] Y. J. Lee, “Detection of Movement and Shake Information using Android Sensor”, Advanced Science and Technology Letters, vol. 90, **(2015)**, pp. 52-56.
[5] J. W. Lee, “Android Sensor Story”, Freelec Press, **(2014)**.
[6] N. Shin, H. Lim, J. Kim and G. Choi, “Socket Network Controller based on Android”, HCI Korea Conference, Gwangwon, Korea, **(2013)**, pp. 1121-1124.
[7] http://developer.android.com/guide/topics/sensors/sensors_overview.html
[8] http://developer.android.com/guide/topics/sensors/sensors_motion.html#sensors-motion -accel
[9] D. Smith and J. Frisen, “Android Recipes: A problem-solution approach”, A press, **(2011)**.
[10] S. P. Hong, S. Kang and J. Kim, “Design and Implementation of Reliable Content Transaction System in Smartphone Environment”, IJSH, vol. 7, no. 6, November **(2013)**, pp. 333-342.

Author

Young Jae Lee, received the B.S. degree in Electronic Engineering from Chungnam National University in 1984, the M.S. degree in Electronic Engineering from Yonsei University in 1994, and the Ph.D. in Electronic Engineering from Kyung Hee University in 2000. He is presently a Professor in the College of Culture and Creative Industry at Jeonju University. His research interests include Smart Media, Computer Vision, AR (Augmented reality), and Computer Games.