Human Computer Interaction for the Accelerometer-based Mobile Game

Jonghun Baek[1], Ik-Jin Jang[2], KeeHyun Park[3], Hyun-Soo Kang[4], and Byoung-Ju Yun[5]

[1] Dept. of Information and Communications, Kyungpook National University, Daegu, 702-701, South Korea
[2] Samsung Electro-Mechanics
[3] College of Information and Communication, Keimyung University, Daegu, 704-701, South Korea
[4] School of ECE, Chungbuk National University, Chungju, 361-763, South Korea
[5] School of Electrical Engineering and Computer Science, Kyungpook National University, Daegu, 702-701, South Korea
bjisyun@ee.knu.ac.kr

Abstract. As a result of growth of sensor-enabled mobile devices such as PDA, cellular phone and other computing devices, in recent years, users can utilize the diverse digital contents everywhere and anytime. However, the interfaces of mobile applications are often unnatural due to limited resources and miniaturized input/output. Especially, users may feel this problem in some applications such as the mobile game. Therefore, Novel interaction forms have been developed in order to complement the poor user interface of the mobile device and to increase the interest for the mobile game. In this paper, we describe the demonstration of the gesture and posture input supported by an accelerometer. The application example we created are AM-Fishing game on the mobile device that employs the accelerometer as the main interaction modality. The demos show the usability for the gesture and posture interaction.

1 Introduction

Advances in mobile computing and micro electro mechanical systems (MEMS) enable the development of games which lead about user's interest on the mobile phone, PDA, and other portable devices. Presently, mobile devices have been made popular to everyone, and users always carry with them. In this sense, the mobile game is becoming more popular, especially with the recent release of the portable devices such as Sony's PlayStation Portable or Samsung's SCH-G100/SPH-1000. It shows that this area has huge potential for growth.

Traditional desktop-based user interfaces have been developed on the basis that user's activities are static states. User interface design for desktop devices can use all of its visual resources. The representative desktop-based interaction mechanisms are keyboard, mouse and joystick. In general, these are very graphical and still more detailed for desktop-based applications. In contrast, interaction mechanisms of the mobile devices can not utilize all or any of their visual resources by reasons not only that activities of users are dynamic states [1] but also the mobile devices have the limited resource and the small LCD display. Games on the mobile devices rely on key-pad, stylus-pen and input-panel. When using these mechanisms to manipulate mobile devices, it may lead to the time delay and the difficult operation because the small button may be manipulated repeatedly. So, it may make the users to lose their interests in the games [2].

There have been various studies about new input/output methods to solve this problem; voice recognition and gesture recognition are representative cases [1]-[4]. Voice recognition comes hard to apply to it because it is unsuitable for the user interface of the mobile game. The other way to design user interface is to recognize the gesture of the user. The user's gesture-based input by interaction between human and computer enable user to approach easily more than desktop PC. The gesture recognition needs additional devices that can detect user's gesture and must be able to embed in mobile devices. The trend of this field is much using the embedded camera [5]. The camera-based user interface may be uncomfortable for use because user's gesture must be transmitted to the mobile device under condition that user carries it in hand. Plus, the camera is comparative high cost. Therefore, it needs new method that can detect user's gesture easier and faster in mobile environment. In other words, proper sensors are necessary to detect actions or to recognize gestures of the user in the case of mobile and portable devices, and the processing capacity and mobile convenience should be considered before applying sensors. MEMS accelerometers are suitable sensors that coincide in this purpose. There have been various studies about using acceleration in recognizing gestures [6]-[8]. To use the gesture as input/output interface of the mobile devices, each gesture should be definitely discriminated each other. Otherwise, useless situation or inappropriate gestures can cause malfunction. Therefore, we sort context information in context-aware computing.

In this paper, to supplement lacking the user interface of the mobile devices, we propose the novel user interface that recognizes user's gestures such as swing and tilt, and estimates the user's posture. The application example we created is a fishing game. To improve efficiency that is expected by applying the accelerometer, with considering many restriction elements such as processing capacity, we propose the structure of the system that embedded in the accelerometer and the algorithm of the signal processing of the accelerometer for the mobile devices, which is the motion-based interaction technology and context-aware computing technology.

2 Context Information

Unlike designing user interface for traditional desktop applications, mobile devices can sort complex context information. Therefore many kinds of context information in context-aware computing environment are sorted [3][9].

Traditional desktop applications wait until user inputs data through the keyboard or mouse before execute especial action. Context-aware applications with sensors, however, can offer or require information even through non-conscious input of user by estimating motion of user at specific time and in particular place. Context-aware computing analyzes changing contexts in ubiquitous computing environment, and discriminates whether the information is valid or not, and then it generates the control event that requests or delivers information to execute the application.

In this paper, we detect context information called the "gesture" and "behavior", and calculate displacement amount of computer from the acceleration in order to gesture recognition and posture estimation. These user's gestures and behaviors are very important factors in ubiquitous computing environment. For examples, pull & push can be used to control the TV (e.g zoom in and out the display, etc.), a tilt can be used to control the TV (e.g. volume and channel control, etc.) and scroll the display of the mobile device, a snatch can be used to control the bell of the cellular phone and to converter the windows of the PDA, etc., and a posture can be used to automatically turn on the mobile devices or receive any information from the object that attached RFID tag when user puts the mobile devices toward their chest in ubiquitous computing environment. Especially, the user's posture estimation technology can correct the various postures of the user such as walking, running and sitting in daily life as well as be applied to the sport science by the fusion of the sensors.

3 System Description

User interface for the mobile devices is designed by a general-structure that is available to the sensor's output at the same time in diverse control interface to increase the expected efficiency by applying the sensor, because the mobile devices have many restriction elements such as processing capability, limited resources (e.g. memory, battery). Fig. 1 is our system that is designed to use the accelerometer by a general-purpose in mobile device.

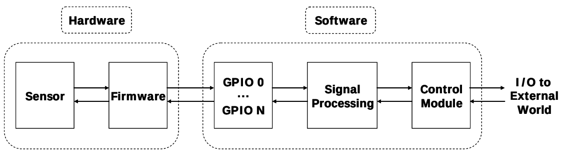


Fig. 1 An architecture of the mobile device that embedded the accelerometer.

Fig. 2 shows that the system generates the control event by processing the signal detected from the accelerometer in software structure of the mobile devices with the RTOS (real time operation system) environment. The accelerometer's output is passed through a dispatcher that asks the GPIO for the signal of the accelerometer and passes through two parallel filters; a lowpass filter for a static acceleration such as tilt and gravity and a highpass filter for a dynamic acceleration such as vibration, shock, and impulse. The lowpass and highpass filtered output then goes to a signal processing for each application and then it generates a gesture-based control event and controls or executes the applications. To decrease overheads that are generated by attaching the accelerometer to the mobile device, we designed the structure that control events do not pass to the control module but to a task module.

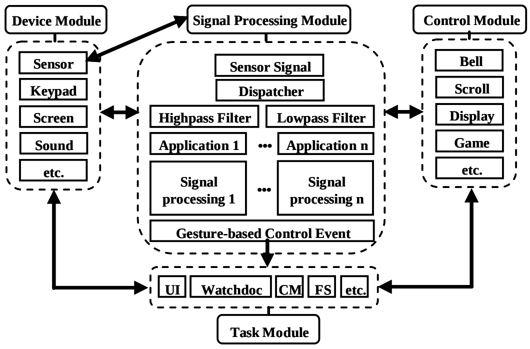


Fig. 2 A block diagram of the mobile device with the RTOS environment.

4. Accelerometer signal processing for the mobile game

In the game of the existing mobile devices, control of the direction and action of a character is done by using the key-pad. In our hand gesture- and posture-based mobile interaction technique, we show the usability for the use of the accelerometer that is embedded in the mobile device and the effect of the input.

In the process of the user's motion recognition, actually, there are several cases that the acceleration signal for the user's action often misrecognized. For examples, there are malfunction by the user's hand tremble, the non-necessary gesture, and misrecognition. In addition, in the case that several gestures are occurred almost at the same time in one action, how we can distinguish each gesture. To resolve these limitations and problems, we proposed that the signal processing method for the gesture recognition and posture estimation is to define each user's posture and gesture stage by stage as considering a typical accelerometer signal for the action. Namely, a representation scheme for user's action is necessary for the description of each posture and gesture. Therefore we use a finite state machine that has a finite number of possible stages.

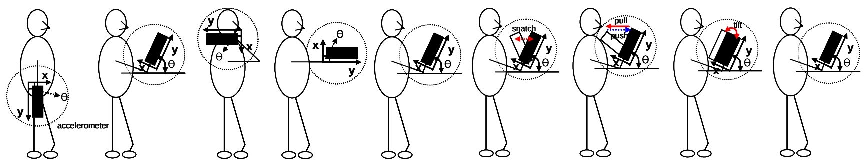
4.1 The scenario for the fishing game

We assume that the action for fishing game consists of the nine stages as followings:

- In the initial stage, the position of the mobile device is assumed that user is gripping the mobile device in an attention state (see Fig. 3(a) and section A in the Fig. 5(b)).
- First, in the beginning stage, the user takes the mobile device toward their chest (see Fig. 3(b) and section B in the Fig. 5(b)). It informs the beginning of the game by the estimating of the user's posture.
- Second, in the ready stage, the mobile device (fishing rod) is located above the user's shoulder before throwing the fishing rod. It informs the throwing gesture

by estimating of the user's posture (see Fig. 3(c) and section C in the Fig. 5(b)).

- Third, in the throwing stage, we determine the movement distance, velocity, power, and path-direction of the fishing rod for the user's action (see Fig. 3(d) and section D in the Fig. 5(b)).
- Fourth, in the waiting stage, at the time, the user takes the mobile device toward their chest and then waits a bite of the fishes (see Fig. 3(e) and section E in the Fig. 5(b)).
- Fifth, in the snatching stage, snatching motion that user tried to land it immediately when the fish are on the feed (see Fig. 3(f) and section F in the Fig. 5(b)).
- Sixth, in the pull & push stage, pulling & pushing gesture indicate that user pulls the fishing rod toward user's chest like user reels in fish in real fishing to consume the power of the fish, and then user watches the mobile phone to check the state of the fish (ex. power of the fish) (see Fig. 3(g) and section G in the Fig. 5(b)).
- Seventh, in the right and left tilting stage, tilting gesture indicate that user tilts the fishing rod to take the fish by their chest direction and to consume the power of the fish (see Fig. 3(h) and section H in the Fig. 5(b)).
- Finally, in the finishing stage, the user takes the mobile device toward their chest again and then user verifies the experimental result (see Fig. 3(i) and section I in the Fig. 5(b)).



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 3 Examples of the fishing action and the location of the accelerometer in the each stage; (a) initial stage, (b) beginning stage, (c) ready stage, (d) throwing stage, (e) waiting for a bite stage, (f) snatching stage, (g) pull & push stage, (h) tilting stage, and (i) finishing stage. A θ is the angle between the mobile device and the ground.

4.2 Accelerometer signal analysis for the fishing action

An experiment was conducted to assess the usability of the accelerometer as interaction technique for the mobile games. Thirty subjects (27 males and 3 females, ages 23 to 33, all were colleagues who volunteered for the study) participated in this study. First, we observed the output patterns of the accelerometer for the fishing action as the above scenario (see Fig. 4).

Fig. 4 was the result of the experiment for the representative eight subjects among of the thirty subjects. Though the amplitudes are different each other for their action, the all output patterns are almost similar in all the experiments. To analysis the acceleration signal of the gestures for the fishing action, we considered the typical signal type for the fishing action (see Fig. 5(b)).

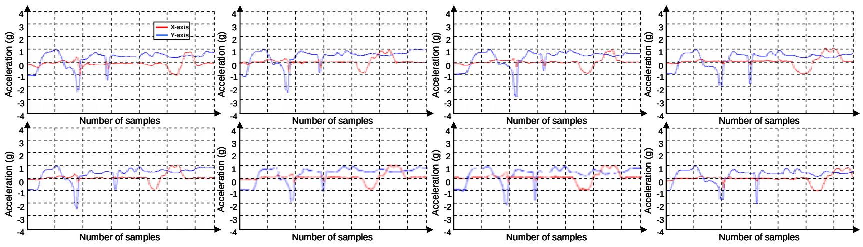
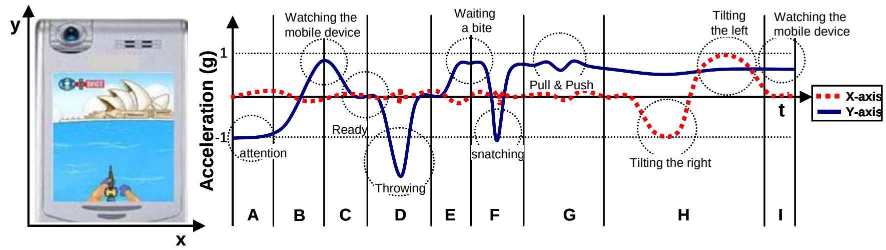


Fig. 4 Signal patterns of the eight participants for the fishing game.

In the Fig. 5, the direction of the mobile device is parallel to the Earth's surface, and X- and Y-axis are pointing the right and forward direction, respectively (see Fig. 5(a)). Therefore, the output values of the X- and Y-axis are both 0g. The section A in the Fig. 5(b) indicates that the user's posture is "attention". The direction of the mobile device of this case is that X- and Y-axis are pointing the right and the Earth's gravitational field, respectively, and the output values are 0g (0º) and -1g (-90º), respectively. The section B in the Fig. 5(b) indicates that user's posture changed from "attention" to beginning stage. In this case, the output values of Y-axis gradually increase because the Y-axis is changed from direction of the Earth's gravitational field to its reverse. And X-axis nearly remains the same. The section C in the Fig. 5(b) indicates that user's posture changed from beginning stage to ready stage. In this case, the output values of Y-axis are decreasing up to about 0g because the direction of the mobile device is parallel to the Earth's surface (see Fig. 3(c)). The section D in the Fig. 5(b) is the throwing stage, and the accelerometer signal is a dynamic acceleration. The section E in Fig. 5(b) is the waiting stage and indicates that user watches the mobile device so that the user checks the bite of the fishes. Therefore, the output values of Y-axis gradually increase the same with beginning stage. The section F in the Fig. 5(F) is the snatching stage, and the accelerometer signal is the dynamic acceleration the same throwing stage. The section G in the Fig. 5(g) is the pull & push stage, and the output values of the accelerometer gradually decrease and then increase. The section H in the Fig. 5(b) is the tilting stage, and the output values of the accelerometer decrease to -1g when user tilt to the right and increase to +1g when user tilt to the left. Finally, the section I in the Fig. 5(b) is the finishing stage that user ascertains the practice result, and the user's posture is the same with beginning stage about the output of the accelerometer.



(a) (b)

Fig. 5 (a) the direction of the accelerometer that is embedded to the mobile device, (b) the typical user's action in fishing game.

Generally, the output acceleration of this accelerometer has been converted to the acceleration that varies between ±1g. However, in the fishing action, the acceleration having over ±1g is detected because the dynamic acceleration such as throwing and snatching exists. Namely, in the fishing action, all of the gesture and posture are a static acceleration except for the throwing and snatching gestures (see Fig. 5(b)).

4.3 Posture Estimation

To estimate user's posture carrying the mobile device, we calculate θ for each state as shown in Fig. 3. The accelerometer uses the force of gravity as an input vector to determine orientation of an object in space. The X- and Y-axis of the accelerometer are pointing the right and forward direction, respectively. If the accelerometer is ideal state, the accelerometer's digital outputs (X- and Y-axis) are 0g when it is parallel to the earth's surface and a scale factor is 12.5% duty cycle change per g. However, the accelerometer may not always be level when the applications using the accelerometer are executed on the mobile device, and calibration is recommended by the manufacturer. The method to do a more accurate calibration is to slowly rotate the accelerometer 360º and to make measurements at ±90º. The following table 1 shows the calibration results with the changes in the X- and Y-axis as the accelerometer that embedded in the mobile device is tilted ±90º through gravity. These experimental results were recorded at each Y-axis orientation to horizon. If the accelerometer is ideal state, the duty cycle is changed 62.5% into 37.5% both X- and Y-axis when angle is changed ±90º. However, in this experiment, the duty cycles of the X- and Y-axis are (66.2-40.6)/2=12.8%/g and (60.1-36.4)/2=11.9%/g, respectively [9].

Table 1 Output of X- and Y-axis respond to changes in tilt.

Y-axis	X output		Y output	
Orientation to horizon	PWM (%)	Acceleration (g)	PWM (%)	Acceleration (g)
90	53.40	0.000	60.10	1.000
75	50.43	-0.232	59.69	0.961
60	47.41	-0.468	58.56	0.866
45	44.80	-0.672	56.73	0.713
30	42.77	-0.831	54.31	0.509
15	41.51	-0.929	51.54	0.276
0	40.60	-1.000	48.25	0.000
-15	41.36	-0.941	45.39	-0.240
-30	42.55	-0.848	42.58	-0.476
-45	44.72	-0.678	40.02	-0.692
-60	47.25	-0.481	38.17	-0.847
-75	50.19	-0.251	36.98	-0.947
-90	53.40	-0.000	36.40	-1.000

To determine a range of the θ for the user's posture, with the various postures of the thirty participants, we collected data a two-dimensional time series for about three seconds at the sampling rate of the 140 samples/s from the outputs of the accelerometer (ADXL202EB). We examined the optimal threshold value to determine the convergence of the user's posture. Here, the convergence of the user posture was determined using the threshold valued and following form,

$$\mu - z \times \frac{\sigma}{\sqrt{n}} < \text{Threshold value} < \mu - z \times \frac{\sigma}{\sqrt{n}} \quad (1)$$

where, μ is mean of the population which is output of the accelerometer, z is normal distribution, σ is variance for the population, and n is number of samples. The 90% (z=1.65) of the total samples is used for the confidence interval. The following table 2 shows the result of the confidence interval.

Through the experiment mentioned above (Table 1), the angle of the mobile device for each user's posture is estimated: it is about -90º for attention, about 39.5º ~ 46.8º for watching the mobile device, about -16º ~ 7º for ready, about 35.5º ~ 54º for waiting the bite, and about 37.6º ~ 40º for watching the mobile device after tilting gesture.

Table 2 Threshold value for estimating user posture; min, max, mean and standard deviation values of Y-axis for each posture.

Posture	Min	Max	Mean	Standard deviation	T(Threshold value)
Attention	-1.0000	-0.9730	-0.9852	0.0040	-0.9918 < T < -0.9786
Watching the mobile device	0.6260	0.7580	0.6853	0.0282	0.6388 < T < 0.7318
Ready	-0.3540	0.1620	-0.0703	0.1168	-0.2631 < T < 0.1225
Waiting the bite	0.5840	0.8520	0.6724	0.0784	0.5840 < T < 0.8017
Watching after tilting	0.5960	0.6500	0.6290	0.0098	0.6127 < T < 0.6452

4.4 Implementation

The gestures used in the fishing game are the throwing, snatching, and tilt; the throwing gesture is expressed as the movement distance and the velocity of the fishing rod by measuring the acceleration for the user's throwing gesture, the path-direction of the fishing rod is expressed by X-axis of the accelerometer when user throw the fishing rod. We consider the range of the power and the tilt in the throwing stage in order to determine the movement distance, the velocity, and the path-direction of the fishing rod. Because it is difficult to distinguish between the dynamic acceleration like

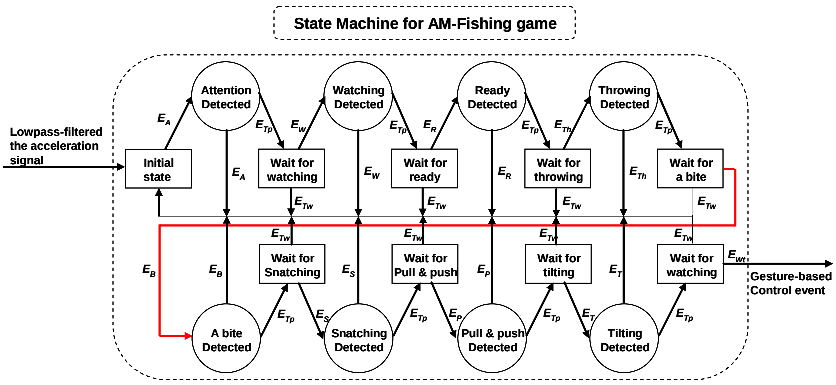


Fig. 6 Finite state machine for the AM-Fishing game; E_A for attention event, E_W for watching event, E_R for ready event, E_{Th} for throwing event, E_B for a bit event, E_S for snatching event, E_P for pull & push event, E_T for tilting event, E_{Wt} for watching event after tilting, and E_{Tp} and E_{Tw} for timer event.

the throwing and the static acceleration like the posture, and the noises as the unnecessary user's action and hand tremble have to be removed, the state machine is needed. The state machine for the mobile fishing game recognizes and processes the user's action by the designed scenario by treating only valid gesture or posture in each state. The input of the state machine is the accelerometer signal by lowpass filter and the output is gesture-based control event. Fig. 6 is structure of the state machine.

(a) (b) (c)

Fig. 7 AM-Fishing game; (a) throwing mode, (b) snatching mode, and (c) tilting mode on the mobile device.

To demonstrate our application in games we have developed the mobile game called AM-Fishing game (see Fig. 7). In the game the user has to throwing the mobile device with his real arm into a virtual fishing site.

5 Experimental Results

We measured the recognition rate of the gesture and posture for the performance evaluation. The sampling rate is set 140ms. We define practiced man and non-practiced man as the participant and the malfunction includes non-function, mis-recognition, and overtime. The table 3 shows the recognition rate for each gesture and posture. This result is useful in terms of the gesture- and posture-based control interface and technique for inducing the interest and realism for the mobile game as well as providing the convenience of the control. For example, in the AM-Fishing game, we can improve the realism by using the gesture and posture in the control interface. Also, this enables to induce extra-interest through the dynamic motion for the control of the game. This is because that the interface for controlling the motion of game characters enables user to feel higher realism as we prefer to use joystick instead of keyboard in the personnel computer-based game.

Table 3 Recognition rate of the gesture and posture in the AM-Fishing game

User	Number of plays	Number of functions	Number of malfunctions	Recognition rate (%)
Practiced man	75	71	4	94.7%
Non-practiced man	75	68	7	90.7%

6 Conclusion

We implemented the new interaction by estimating and detecting the context information as user's various gestures and postures with 2-axis accelerometer. We considered the typical fishing action by the output type of the accelerometer for 30 participants. Because it is difficult to distinguish between the dynamic acceleration and the static acceleration, and the noises as the unnecessary user's action and hand tremble have to be removed, the state machine is needed. The state machine for the mobile fishing game recognizes and processes the user's action by the designed scenario by treating only valid gesture or posture in each state. As a result, this will enhance not only the convenience to use but also the interest and realism for the mobile game.

Acknowledgement. This work was supported by the Ministry of Education, Korea, under the BK21 project.

References

[1] M. Ehreumann, T. Lutticke and R. Dillmann, Dynamic Gestures as an Input Device for Direction a Mobile Platform, IEEE International Conference on Robotics and Automation, vol. 3, pp. 2596-2601, 2001.
[2] V. Paelke, C. Reimann and D. Stichling, Foot-based mobile Interaction with Games, ACM SIGCHI International Conference on Advances in Computer Entertainment Technology, pp. 321-324, 2004.
[3] P. Indrajit, S. Togesh, O. Ercan and S. Rajeev, Toward Natural Gesture/Speech HCI: A Case Study of Weather Narration, Workshop on Perceptual User Interfaces, pp. 1-6, 1998.
[4] B. N. Schilit, N. Adams, and R. Want, Context-Aware Computing Applications, In Proceedings of the 1st International Workshop on Mobile Computing Systems and Applications, pp. 85-90, 1994.
[5] L. Gupta and S. Ma, Gesture-Based interaction and Communication: Automated Classification of Hand Gesture Contours, IEEE Transactions on Systems, Man, and Cybernetics, vol. 31, No. 1, pp. 114-120, 2001.
[6] S. Sotiropoulos, K. Papademetriou and A. Dollas, Adaptation of a Low Cost Motion Recognition System for Custom Operation from Shrink-Wrapped Hardware, Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications, pp. 107-114, 2003.
[7] A Wilson and S. Shafer, "XWand: UI for Intelligent Spaces, In Proc. CHI, pp. 545-552, 2003.
[8] K. Partrige, S. Chatterjee, V. Sazawal, G. Borriello and R. Want, TiltType: Accelerometer-Supported Text Entry for Very Small Devices, In UIST02: In Proceedings of the 15th annual ACM Symposium on User Interface Software and Technology, pp. 201-204, 2002.
[9] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith and P. Steggles, Towards a Better Understanding of Context and Context-Awareness, In HUC'99: In Proceeding of 1st International Symposium on Handheld Ubiquitous Computing, pp. 304-307, 1999.
[10] Low-cost ±2g/±10g Dual Axis iMEMS Accelerometers with Digital Output ADXL202/ADXL210 Data Sheet, Analog Devices Corporation, 1999.