

가속도센서 기반 휴대폰 단말기 사용자 인터페이스 구현

백종훈⁰ 이기혁 *

경북대학교 정보통신학과, *한국정보통신대학원대학교 공학부
jhback⁰@inc.knu.ac.kr; geehyuk@icu.ac.kr

Accelerometer-based UI for a Mobile device

Jong-Hun Baek⁰ Gee-Hyuk Lee *

Dept. of Information and Communication, Kyungpook National University,
*Dept. of Engineering, Information and Communication University

요 약

MEMS 가속도센서 기술의 발전은 가속도센서를 이용한 사용자 인터페이스 구현에 대한 연구를 촉진 시키고 있다. 본 논문에서는 가속도센서를 기반으로 한 휴대폰 단말기를 사용하여, HCI 관점에서 센싱 기술을 소개한다. 이러한 기술은 센서 신호로부터 컴퓨터의 변위량을 계산함으로써 Knock & Mute 와 Tilt & Scroll와 같은 인터페이스를 구현할 수 있다.

1. 서론

PIM, PDA, cellular phone과 같은 휴대용 단말기에 대한 시장이 급속히 성장하고 있다. 휴대폰 단말기는 항상 휴대하거나, 사용자의 일과 전부를 PIM을 통해 저장시킬 수 있기 때문에 데스크탑 컴퓨터 보다 사용자에게 더 유용하다. 그래서 현재 우리는 사용자의 경험을 통해 얻어진 결과를 가지고 사용자에게 보다 더 친밀하게 설계되어진 HCI를 제시한다.[1]

본 논문에서는 이러한 휴대폰 단말기에 가속도센서를 장착하여 인간의 움직임을 항상 파악할 수 있는 인터페이스를 제안한다. 이러한 움직임을 통한 대화방식은 인간 대인간의 대화에서의 보다 HCI에 더 유용할 것이다.[2][3]

본 논문에서는 가속도센서의 출력 신호로부터 제스처인식의 응용에 용이한 형태의 신호를 추출하는 신호처리 기법과 각각의 사용자 인터페이스의 구현결과를 설명한다.

2. 가속도센서의신호처리 기법

2.1 Knock & Mute

휴대폰의 벨이 울릴 때 주머니에서 꺼내지 않은 상태에서도 세 번 두드림으로써 벨을 멈추게 하는 기능이다. 본 연구에서는 네 가지 다른 상황, 즉, 평면 위에 둔상태, 손에 든 상태, 주머니에 둔 상태, 가방 안에서 두드림에 의한 가속도센서의 신호를 검출했다. 그림 [2.1-1]에서 한번의 두드림이 하나의 피크를 발생하게 되는 것을 보여준다.

개인마다 두드리는 패턴이 각각 다르고 같은 사용자라도 두드림 때 마다 패턴의 차이는 있지만, 한번의 두드림이 하나의 피크를 발생시키는 것을 그림 2.1-1으로부터 알 수 있다.

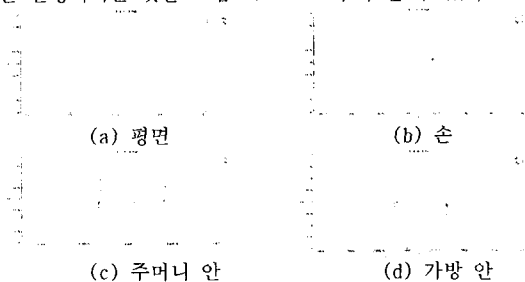


그림 2.1-1 휴대폰을 평면, 손, 주머니 안, 가방 안에 둔 상태에서 세 번 두드렸을 때의 신호

이러한 신호를 처리하기 위해 그림 2.1-2에 나타낸 전형적인 경우를 고려한다. 그림에서 가속도의 절대치, 즉

$$a = \sqrt{a_x^2 + a_y^2}$$

의 시간적 변화를 보여주고 있다.

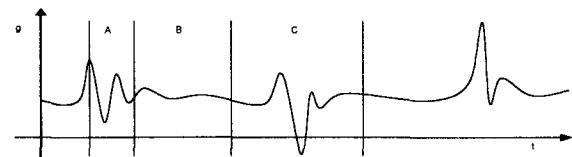


그림 2.1-2 전형적인 두드림 신호

여기서 두드림을 검출하기 위해서 고려해야 할 점을 정리하면 다음과 같다.

- 사용자의 움직임과 휴대폰의 위치변화에 따른 중력가속도의 영향으로 인한 저주파 영역을 highpass filter로 제거해야 한다.
- 위 그림의 A 구간에 나타난 바와 같이 휴대폰 사용자가 100ms 간격으로 두드릴 수 없는 것이므로, 이 구간을 dead period로 둬으로써 일종의 Debouncing을 구현하여 멀티 피크를 해결한다.
- 사용자가 휴대폰을 심하게 움직이고 있을 때는 멀티 피크의 문제뿐 아니라 계속되는 진동을 야기시킬 수 있다. 이러한 진동을 규칙적인 두드림과 구분하기 위하여 위그림의 B 구간을 silent period로 둔다.
- 규칙적인 두드림과 간헐적인 충격을 구분하기 위하여 위 그림의 C 구간을 두어, 한번 두드림이 검출된 이후 다음 두드림 까지 최대시간을 정한다.

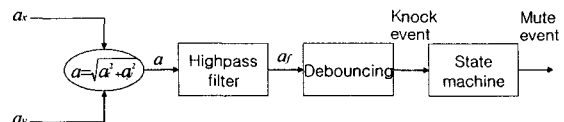


그림 2.1-3 가속도센서의 신호로부터 세 번의 규칙적인 두드림을 검출하기 위한 신호처리 및 detection 과정.

Highpass filter 과정은 가속도센서의 신호에서 사용자의 음

직임이나 휴대폰의 위치변화에 따른 중력가속도신호의 변화를 제거하기 위한 과정이다. 이것은 실험결과 DC 성분만 제거하여도 원하는 결과를 얻을 수 있었다. 이 실험에서 사용한 highpass filter는 지정한 주파수 사이에 통과 영역을 갖는 프로토타입 필터의 대표적인 first order butterworth filter를 이용하였다. 필터의 pole-zero 위치와 amplitude 특성을 그림 2.1-4에 나타내었다.[4]

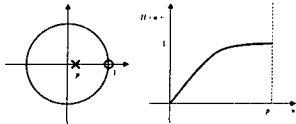


그림 2.1-4 1-st order Butterworth filter (p=0.2)

필터의 transfer function과 해당하는 recurrence equation은 다음과 같다.

$$H(z = e^{j\omega}) = A \frac{1 - z^{-1}}{1 - pz^{-1}}, \quad A = \frac{p+1}{2}$$

$$a_f[n] = pa_f[n-1] + A(a[n] - a[n-1])$$

그림 2.1-4에서는 p=0.2의 경우를 보여주고 있는데, 이 값은 실제 실험을 통하여 선택된 적절한 값이다.

그림 2.1-5는 필터링 전·후 신호의 예를 보여주고 있다.

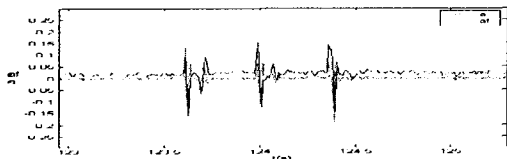


그림 2.1-5 Signal before & after the highpass filter

Debouncing 과정은 한번의 두드림에 의해 발생하는 여러 개의 피크를 각각 검출하는 것을 피하기 위한 과정이다. 구현은 피크가 한번 검출되고 나면 지정한 시간 동안 들어오는 신호를 무시한다.

State machine 과정은 앞에서 한번 두드림을 검출하는 과정은 실제로 사용자의 움직임에 의한 가속도신호를 잘못 인식하는 경우가 많다. 따라서 세 번의 규칙적인 두드림이 필요하며, 이러한 규칙적인 두드림을 검출하기 위해 state machine이 필요하다.

State machine으로의 입력은 debouncing 단계에서 보내지는 Knock event가 되며 출력은 Mute event가 된다. State machine의 구조를 그림 2.1-6에 나타내었다.

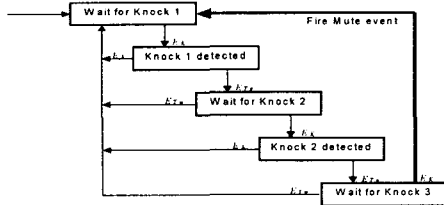


그림 2.1-6 State diagram

실제 testbed에서의 구현 예에서 세 번 두드림이 아니라 임의의 동작에 의해서도 "Wait for Knock2"나 "Wait for Knock3"까지 transition하는 경우를 가끔 볼 수 있었다. 그러나 임의의 동작에 의하여 Mute event가 발생하는 경우는 거의 관찰할 수 없었다.

2.2 Tilt & Scroll

키나 마우스 등의 일반적으로 사용자 인터페이스를 이용하기 어려운 휴대폰과 같은 소형 휴대용 장치에서 화면을 기울임으로써 스크롤링을 구현하는 방법이다. 예를 들어 큰 이미지를 소형 휴대폰이나 PDA 스크린으로 볼 때 원하는 부분으로 스크롤링하기 위해서 그 방향으로 디스플레이를 기울이면 자동으로 스크롤이 된다.

이 기능을 구현하는데 있어서 고려해야 할 점을 정리하면 다음과 같다.

- 적절한 lowpass filtering과 dead-zone의 설정으로, 손 떨림이나 차량의 진동 등에 의한 스크롤링 문제점을 해결한다.
- 휴대용 장치의 평행상태를 선택할 수 있는 적절한 방법이 고안되어야 한다. 가속도센서의 두 축이 디스플레이의 평면상에 있다고 할 경우, 디스플레이가 수평으로 놓일 때 가속도 센서의 출력이 최소가 되며 자연스런 평행상태가 된다고 볼 수 있다. 그러나 사용자의 입장에서 디스플레이를 수평으로 유지하면서 사용하는 것이 자연스러운 자세가 아니다. 대부분의 경우 디스플레이가 사용자 쪽으로 기울어진 상태에서 사용하는데, 이러한 상태가 평행상태가 될 수 있어야 하고, 이 상태에서 상대적인 기울임에 의해 스크롤링이 될 필요가 있다.
- Tilting의 정도에 따른 스크롤링의 변화에 대한 연구가 필요하다. 예를 들면, 기울이는 정도에 관계없이 일정한 속도로 스크롤링을 할 수 있고, 기울임 각도에 비례하는 속도로 스크롤링을 할 수도 있을 것이다.

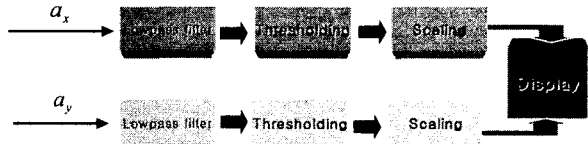


그림 2.2-1 Tilt & Scroll 신호처리 단계

그림[2.2-1]은 중력에 의한 가속도 성분을 손의 떨림이나 차량의 진동 등에 의한 신호로부터 분리하기 위한 과정을 나타내었다.

X축과 Y축 방향의 가속도는 독립적으로 각각의 단계를 거친다. 왜냐하면 이렇게 독립적으로 처리하여 수평·수직 스크롤링 양을 계산해도 되는 이유를 그림 [2.2-2]에서 설명하고 있다.

그림 [2.2-2]에서 x, y 평면을 수평으로 보여주고 있으나 실제상황에서는 g vector가 수직하방향이 된다. 디스플레이면이 기울어져 있을 경우, 디스플레이에 나타난 화상이 g_m 방향으로 (또는 그 반대방향으로) 움직이는 것이 가장 직관적일 것이다. 이는 마치 화상이 중력의 힘을 받아서 움직이는 것 같이 보이게 할 것이다. 이런 경우의 화면 스크롤링은 g_m 방향이어야 하고, 이를 구현하기 위해서는 x축, y축 방향으로의 스크롤링 양이 각각 g_x와 g_y에 비례하도록 하여야 할 것이다. 따라서 그림 2.2-1에 나타낸 바와 같이 g_x와 g_y의 독립적인 처리가 가능한 것이다.

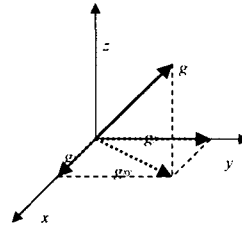


그림 2.2-2 중력의 방향과 스크롤링 방향의 관계

Lowpass filter 과정은 가속도 신호에서 사용자의 손떨림 등에 의한 스크롤링의 반응을 줄이기 위한 과정이다. 실험을 통하여 아주 적은 저주파만 남겨도 이용에 문제가 없었다. 따라서 Tilt & Scroll 기능이 휴대폰과 같이 제한된 계산능력을 가진 장치에 적용하려면 간단한 필터를 이용하는 것이 바람직하다. 따라서 본 연구에서는 first order butterworth filter를 이용하였다. 필터의 pole-zero 위치와 amplitude 특성을 그림 2.2-3에 나타내었다.[4]

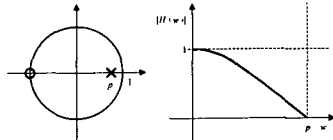


그림 2.2-3 1-st order Butterworth filter (p=0.8)

필터의 transfer function과 해당하는 recurrence equation은 다음과 같다.

$$H(z = e^{j\omega}) = A \frac{1 - z^{-1}}{1 - pz^{-1}}, \quad A = \frac{1-p}{2}$$

$$a_f[n] = pa_f[n-1] + A(a[n] + a[n-1])$$

그림 2.2-3에서는 p=0.8의 경우를 보여주고 있는데, 이 값은 실제 실험을 통하여 선택된 적절한 값이다.

그림 2.2-4는 필터링 전, 필터링 후 신호의 예를 보여주고 있다.



그림 2.2-4 Signals before & after the lowpass filtering

Thresholding 과정은 사용자가 평행상태를 유지하고 정지된 좌상을 유지하는 것을 돕기 위한 과정이다. 구현은 미리 설정된 값 이하의 g_x 와 g_y 를 무시함으로써 쉽게 구현할 수 있다. 그림 2.2-5는 이 과정을 의미하는 것을 g_x 와 g_y 평면에서 dead zone이란 개념으로 설명한 것이다. Lowpass filtering 단계를 거친 g_x 와 g_y 가 그림 2.2-5에서 나타낸 $g_x - g_y$ 평면에서 active zone에 들어갈 때만 스크롤링으로 변환이 되고, dead zone에 들어갈 때는 스크롤링이 일어나지 않도록 한다. 따라서 그림 2.2-5에서 나타낸 dead zone의 폭은 어느정도 디스플레이를 기울여야 스크롤링이 시작되는지를 조절하는 파라미터가 된다.

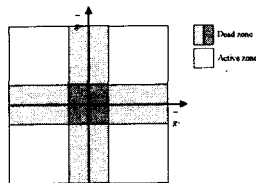


그림 2.2-5 Dead-zone and active-zone

Scaling 과정은 ω 와 α 에 비례하여 어느 정도의 스크롤링을 하는 것이바람직한 것인지를 결정하는 과정이다. 본 연구에서는 아래와 같이 세가지 경우를 고려하였다.

• Constant scrolling

$$v_x = v_0 \operatorname{sgn}(\bar{a}_x), \quad v_y = v_0 \operatorname{sgn}(\bar{a}_y)$$

여기서 v_x 와 v_y 는 각각 x축과 y축 방향으로의 스크롤링 속도를 나타내고, v_0 는 상수이다.

• Linear scrolling

$$v_x = \alpha \bar{a}_x, \quad v_y = \alpha \bar{a}_y$$

여기서 v_x 와 v_y 는 각각 x축과 y축 방향으로의 스크롤링 속도를 나타내고, α 는 비례상수이다.

• Quadratic scrolling

$$v_x = \beta \bar{a}_x^2, \quad v_y = \beta \bar{a}_y^2$$

여기서 v_x 와 v_y 는 각각 x축과 y축 방향으로의 스크롤링 속도를 나타내고, β 는 비례상수이다.

위 세가지 방법 중 어느 방법이 적절한 지를 결정하는 것은 신호처리적인 관점에서 해결할 수 있는 것이 아니라, 사용자 인터페이스의 연구에 일상적으로 이용되는 사용자 테스트에 의해 결정될 수 있다.

실제 testbed에서의 구현 예에서는 linear scaling을 이용하였고, 실험을 통하여 dead zone의 폭은 0.2g가 적절함을 알 수 있었다. 그리고 파라미터들은 사용자에게 따라 다를 수 있으므로 설정 가능하여야 할 것이다.

3. 예제 어플리케이션의 구현 및 결론

본 연구의 최종 목표는 실제 휴대폰에서 각 기능들의 구현을 목적으로 하나, 현실적으로 휴대폰의 소스 프로그램이 공개되기 힘들 뿐 아니라, 현재 생산되고 있는 휴대폰의 계산능력의 한계 등을 고려할 때 최종 목표의 달성이 용이 하지 않았다. 차선의 방법으로 가속도센서를 내장한 휴대폰 장치와 PC에서 동작하는 휴대폰을 시뮬레이션하는 프로그램을 연결하여 본 연구에서 제안한 기능들을 테스트 하였다.

휴대용 장치는 ADXL202 가속도센서 evaluation board를 내장하고, evaluation board는 ADXL202로부터 신호를 읽어 RS232 프로토콜로 출력하여 PC의 serial port로 연결하여 실시간 가속도센서 신호를 전달한다.[5]

가속도센서를 이용한 여러가지 기능들을 휴대폰에 적용할 경우, 각 기능들간의 간섭을 연구하는 목적이 중요하다. 본 실험에서는 Tilt & Scroll을 이용하는 동안 Knock & Mute가 오동작하는 경우는 없었다.

현재는 가속도센서 기반으로 한 Pedometer와 Slide & Scan 기능을 연구 중에 있으며, 성공적인 결과가 예상된다.

참고문헌

[1] K. Hinckley, J. Pierce, M. Sinclair, E. Horvitz, Sensing Techniques for Mobile Interaction, ACM UIST 2000 Symposium on User Interface Software and Technology, CHI Letters 2 (2), pp.91-100, 2000.
 [2] H. Sawada, S. Hata, S. Hashimoto, Gesture Recognition for Human-Friendly Interface in Designer-Consumer Cooperate Design System, RO-MAN '99, 8th IEEE International Workshop on Robot and Human Interaction, pp.400-405, 1999.
 [3] K. Hinckley, E. Horvitz, Toward More Sensitive Mobile Phones, ACM UIST 2001, Symposium on User Interface Software & Technology, pp.191-192, 2001.
 [4] J.G.Proakis, D.G.Manolakis, Digital Signal Processing, Third Edition, Prentice hall, pp.682-683, 1996.
 [5] H. Weinberg, Using The ADXL202 In Pedometer And Personal Navigation Applications, Analog Decive, TECHNICAL NOTE.