

가속도 센서를 이용한 동작 인식 알고리즘 개발

Development of motion-detective algorithm using accelerometer

임영철*, 박지호*, 김상동*, 정우영*

Young-Chul Lim*, Chi-Ho Park*, Sang-Dong Kim*, Woo-Young Jung*

Abstract - In this thesis, we propose a algorithm that is able to detect a user's motion. A minute noise of accelerometer arises cumulative error in case of converting velocity and distance, which makes it difficult to detect movement of sensor. We use a lowpass filter, ALS algorithm, and motion detection block to minimize such cumulative error. we experiment using a motion-detective module which is composed of accelerometer, micro-controller, and serial interface. Our scheme is capable of detecting such as up, down, left, right, forward, and backward movement of the module. It is expected that our scheme is applied to a game controller or user interface of a next generation PC.

Key Words : 동작 인식, 가속도 센서, 신호 처리, 관성 센서, 잡음 제거

1. 서론

최근 사용자의 동작이나 움직임을 인식하여 사용자의 다양한 명령을 시스템에 전달하는 연구가 많이 진행되고 있다. 동작 인식을 통한 사용자 인터페이스는 컴퓨터 비전 기술을 이용한 방법[1]과 관성 센서를 이용한 방법[2-5] 등이 있다. 컴퓨터 비전 기술을 이용한 동작 인식 방법은 카메라의 위치에 따라 사용자의 공간적 제한을 많이 받는 단점과 영상 처리를 위한 메모리와 복잡한 연산이 필요한 단점이 있다. 관성 센서를 이용한 동작 인식 방법은 사용자의 몸에 착용하거나 손을 이용하여 조작할 수 있기 때문에 공간적 제한이 비교적 자유롭고, 센서의 출력 신호만을 처리하기 때문에 메모리나 복잡한 연산을 요구하지 않는 장점이 있다. 그러나 센서와 시스템이 갖고 있는 잡음으로 인하여 정확한 측정이 어렵고, 다양한 동작을 인식하기 위해서는 여러 가지 센서를 융합하여 구현해야 하기 때문에 시스템이 복잡해질 수 있다. 최근 동작 인식에 많이 사용되는 센서로는 가속도 센서, 각속도 센서, 지자기 센서 등이 있다. 최근에는 비교적 가격이 싸고, 초소형인 가속도 센서를 사용하여 모바일 기기나 가전 기기에 적용하려는 연구가 진행되고 있다[3-5]. 가속도 센서의 출력 신호는 중력 가속도, 운동 가속도, 잡음 등으로 구성되어 있어서, 각각에 대한 분리가 어렵다. 그러므로 가속도 센서만을 이용하여 사용자의 다양하고, 복잡한 동작을 정확하게 인식하는 것은 매우 어렵다. 그러나 게임과 같이 제한된 사용자 인터페이스를 요구하는 응용 분야에서는 콘텐츠에 따라

게임에 필요한 동작만을 인식하게 함으로써, 게임에 효율적으로 적용될 수 있을 것이다.

본 논문에서는 3축 가속도 센서의 출력 신호를 저역 통과 여파기, Adaptive Level Shifting(ALS) 알고리즘, 동작 구간 감지 블록을 이용하여 상, 하, 좌, 우, 전진, 후진 등의 동작을 인식한다.

본 논문은 2장에서는 제한한 동작 인식 알고리즘과 실험 결과를 설명하고, 3장에서는 결론을 맺는다.

2. 제안한 알고리즘 및 실험 결과

2.1 제안한 동작 인식 알고리즘

본 논문에서는 사용자의 동작을 인식하기 위하여, 3축 가속도 센서를 이용하여, 구현 동작 인식 모듈의 움직임을 인식한다. 가속도 센서는 X, Y, Z축으로 각각 아날로그 신호를 출력하며, 센서의 출력 신호(S_{out})는 아래와 같이 나타낼 수 있다.

$$S_{out}(t) = A_m(t) + A_g(t) + N_w(t) \quad (1)$$

식 (1)에서 $A_m(t)$ 는 센서의 운동 가속도, $A_g(t)$ 는 센서의 중력 가속도, $N_w(t)$ 는 센서의 잡음을 나타낸다. 가속도 센서를 이용하여 동작을 인식하기 위해서는 식 (1)에서 센서의 출력 신호(S_{out})에서 운동 가속도(A_m)를 분리한 후, 운동 가속도(A_m)에 대하여 적분을 하면, 속도 신호가 되고, 다시 적분하면 이동 거리를 얻을 수 있다. 센서가 고정되어 있다는 가정에서, 즉 중력 가속도 신호(A_g)가 고정되어 있다는 가정에서도 센서의 잡음 성분을 제거하고, 센서의 운동 가속도 성분만을 분리해내는 것은 어렵다. 모듈의 이동 거리를 구하기 위하여, 센서의 출력 신호를 적분하게 되면 센서의 미세한 잡음 성분은 누적된 오차에 의하여, 시간이 지남에 따라 실제 이동 거리와 큰 차이가 나게 된다. 그러므로 본 논문에서는 게임

저자 소개

- * 임영철 : 대구경북과학기술연구원 IT 연구부
- * 박지호 : 대구경북과학기술연구원 IT 연구부
- * 김상동 : 대구경북과학기술연구원 IT 연구부
- * 정우영 : 대구경북과학기술연구원 IT 연구부

※ 본 연구는 2006년 과학기술부/기관고유과제 연구지원사업의 일환으로 진행되었음. (과제번호 : 06CII-01)

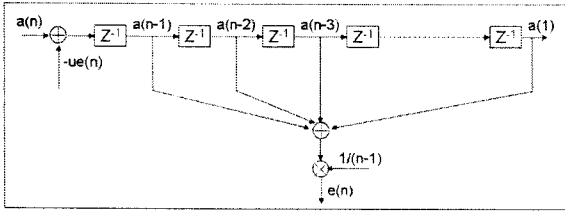


그림 1. ALS 알고리즘 블록도

기의 컨트롤러나 차세대 PC의 사용자 인터페이스와 같은 응용 분야에 적용하기 위하여, 모듈의 제한된 움직임 대한 동작 인식에 필요한 신호 처리 방법을 제안한다. 먼저 센서의 아날로그 출력 신호는 마이크로 컨트롤러에서 디지털로 변환되고, 변환된 디지털 신호는 저역 통과 여파기를 이용하여 필요 없는 고주파의 잡음 성분을 제거한다. 본 논문에서 사용한 이상적인 디지털 저역 통과 여파기의 주파수 응답 특성을 아래와 같다.

$$G_p(e^{i\Omega}) = \begin{cases} 1 & \text{for } |\Omega| \leq \Omega_c \\ 0 & \text{for } |\Omega| > \Omega_c \end{cases} \quad (2)$$

식(2)에서 IDFT를 이용하여 임펄스 응답을 구하면 아래식과 같다.

$$g_p(n) = \frac{1}{2\pi} \int_{-\Omega_c}^{\Omega_c} G_p(\Omega) e^{i\Omega n} d\Omega = \frac{\Omega_c}{\pi} \text{sinc}\left(\frac{\Omega_c n}{\pi}\right) \quad (3)$$

식 (3)에서 $g_p(n)$ 은 기간이 무한히 긴 신호이므로, FIR 필터로 구현할 수 없다. 그러나 임펄스 응답에서 크기가 두드러진 부분($n=0$)을 중심으로 적절한 길이(N_w)의 계수만 이용하면 저역 통과 여파기로 사용하는 데 큰 문제가 없다. 식 (3)에서 구한 N_w 길이의 저역 통과 여파기와 센서의 출력 신호를 컨볼루션하면, 센서의 고주파 잡음 신호를 제거할 수 있다.

$$S_p(n) = S_{out}(n) \otimes g_p(n) \quad (4)$$

식 (4)에서 구한 S_p 신호를 적분하면, 아래 식에서와 같이 센서의 속도 신호를 얻을 수 있다.

$$V_p(n) = \sum_{k=1}^n S_p(k) \quad (5)$$

그러나 S_p 신호에는 미세한 잡음성분이 남아 있기 때문에, 식 (5)에와 같이 적분을 하면, 시간이 지남에 따라 속도에서 누적된 오차를 받게 된다. 이러한 누적된 오차를 최소화하기 위하여 아래 식에서와 같이 ALS 알고리즘을 적용한다. 그림 1은 ALS 알고리즘의 블록도를 보여준다.

$$\epsilon(n) = \frac{V_p(n)}{n} = \frac{1}{n} \sum_{k=1}^n S_p(k) \quad (6)$$

n 길이 구간에서 평균 속도는 식 (6)에서와 같이 구할 수 있다. 사용자 인터페이스를 위한 응용에서는 센서를 움직이는 이동 거리가 제한적이기 때문에, 식 (6)에서 같이 n 이 크게 되면, 평균 속도의 기대값은 0에 가까워진다. 그러므로 식 (6)에서 n 이 크게 되면, $\epsilon(n)$ 은 가속도 측정 오차가 된다. ALS 알고리즘을 적용한 가속도 신호는 아래와 같다.

$$A_{abs}(n) = S_p(n) - u\epsilon(n) \quad (7)$$

식 (7)에서 구한 가속도 신호에는 센서의 동작이 발생하지 않은 구간에서도 미세한 가속도 값이 남아 있다. 이러한 가속도 값은 속도나 이동거리로 변환 시에 적분으로 인한 누적

된 오차를 발생시킨다. 이러한 오차를 최소화하기 위하여 아래 식에서와 같이 N 개의 지연 버퍼를 만들고, 동작이 발생하는 구간을 감지하여, 동작이 발생한 구간 $[i, i+m-1]$ 에서만 유효 가속도로 인식한다.

$$Buf(1:N) = A_{abs}(i:i+N-1) \quad (8)$$

$$|A_{buf}|_{max} = \max(abs(Buf)) \quad (9)$$

식 (9)에서 $|A_{buf}|_{max}$ 값이 임계값 이상이 되면 버퍼에 있는 모든 구간을 동작이 발생한 구간으로 간주하고, 나머지 구간에서는 정지된 상태로 간주한다. 동작 발생 구간을 고려한 센서의 속도는 아래와 같이 구할 수 있다.

$$\begin{cases} V(n) = \sum_{k=i}^n V_{abs}(k), & \text{if } E_{[i,n]} = 1 \\ V(n) = 0, & \text{else} \end{cases} \quad (10)$$

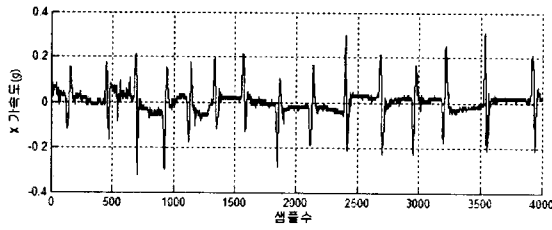
식 (10)에서 $E_{[i,n]}$ 가 1인 경우는 동작이 발생한 구간 $[i,n]$ 을 나타낸다. 동작 발생 구간을 고려한 센서의 이동 거리는 아래와 같이 구할 수 있다.

$$\begin{cases} D(n) = \sum_{k=i}^n V(k), & \text{if } E_{[i,n]} = 1 \\ D(n) = 0, & \text{else} \end{cases} \quad (11)$$

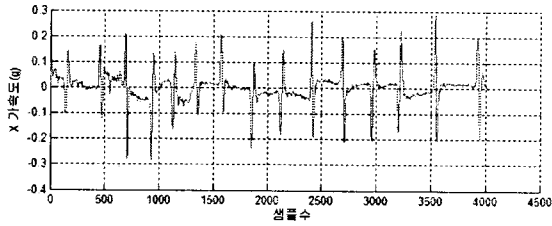
12 실험 결과

본 논문에서는 저가의 3축 가속도 센서 블록과 8051 마이크로 컨트롤러 블록, 호스트 인터페이스를 위한 시리얼 인터페이스 블록으로 구성하여, 동작 인식 모듈을 구현하였다. 가속도 센서의 아날로그 출력은 마이크로 컨트롤러에서 50Hz의 샘플링 주파수로 디지털로 변환된다. 디지털로 변환된 가속도 값은 시리얼 인터페이스를 통해 호스트 PC로 전송된다. 호스트 PC의 응용 프로그램에서는 수신된 가속도 신호를 처리하여, 제안한 동작 인식 알고리즘을 검증한다.

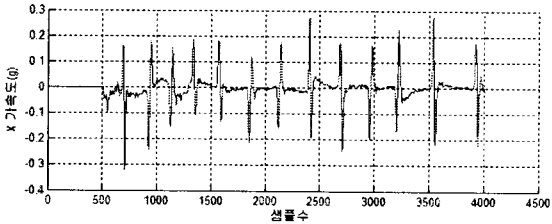
먼저 모듈을 수평으로 고정하고, X축으로 좌, 우 흔들었을 때, 측정된 X축의 가속도 값은 그림 2. (가)와 같다. 센서의 출력 신호에서 고주파 잡음 성분을 제거하기 위하여, 저역 통과 여파기를 사용한다. 식 (2)에서 저역 통과 여파기의 차단 주파수(Ω_c)는 $1/10 \pi$ 로 설정하고, 식 (3)에서의 창 길이(N_w)는 10으로 설정하였다. 저역 통과 여파기를 통과한 가속도 신호는 그림 2. (나)와 같다. 저역 통과 후 500개의 샘플수, 즉 10초 후의 평균 속도의 기대값은 0에 가깝게 된다. 식 (6)에서 $\epsilon(n)$ 값은 가속도 센서의 오차가 되고, 센서의 오차가 최소화 될 수 있도록 조절하는 ALS 알고리즘을 적용한다(그림 2. (다)). 그림 3은 ALS 적용 후의 평균 속도가 0에 가깝게 수렴하는 것을 보여준다. 동작이 발생하지 않는 시점에서의 가속도의 작은 값은 속도나 이동거리로 변환될 때에 적분을 해야 하기 때문에 결과적으로 누적된 오차를 발생시킨다. 이러한 누적된 오차를 최소화하기 위하여, 동작이 발생한 구간을 감지하여, 동작 발생 구간에서의 가속도 신호만을 유효 신호로 인정하고, 나머지 구간에서는 가속도 신호를 0으로 만든다(그림 2. (라)). 동작이 발생한 유효 구간에서 가속도 신호를 적분하면, 상대 속도를 구할 수 있고(그림 2. (마)), 유효 구간에서 구해진 상대 속도를 적분하면, 상대적인 이동 거리를 얻을 수 있다(그림 2. (바)).



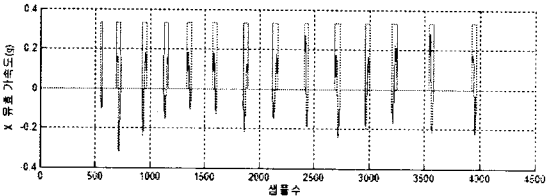
(가) 센서 출력 신호



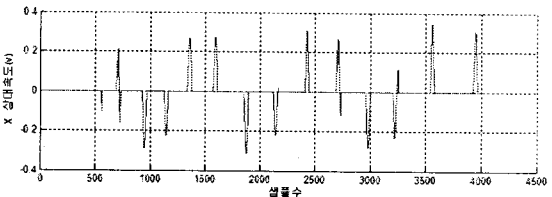
(나) 지역 통과 후 신호



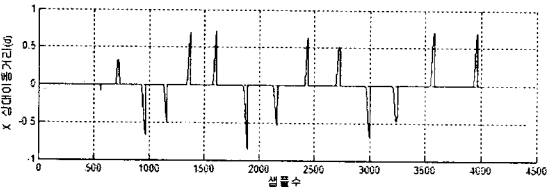
(다) ALS 처리 후 신호



(라) 동작 발생 구간 및 유효 가속도 신호



(마) 상대 속도



(바) 상대 이동 거리

그림 2. 동작 인식을 위한 신호 처리 과정

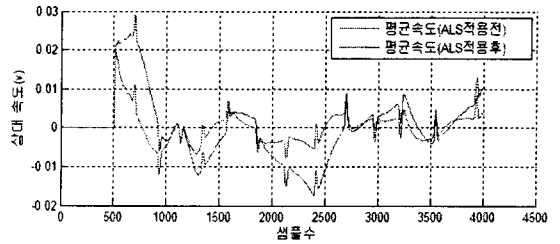


그림 3. 평균 속도

3. 결론

본 논문에서는 가속도 센서를 이용하여 사용자의 동작을 인식할 수 있는 알고리즘을 제안하였다. 동작 인식을 위하여 3축 가속도 센서와 마이크로 컨트롤러, 사리얼 인터페이스로 구성된 모듈을 구현하여, 호스트 PC에서 제안한 알고리즘을 검증하였다. 모듈의 움직임에 대한 오차를 최소화하기 위하여, 지역 통과 여파기, ALS 알고리즘, 동작 인식 블록을 사용하였다. 실험 결과 모듈의 상, 하, 좌, 우, 전진, 후진 등의 동작에 대하여 인식할 수 있었다. 제안한 방법은 게임기의 컨트롤러나 차세대 PC의 사용자 인터페이스로 적용될 수 있는 것이다. 향후 연속 동작이나 복잡한 움직임에 대한 연구와 정확한 동작의 인식에 대한 연구가 필요할 것이다.

참 고 문 헌

- [1] T. Kirishima, K. Sato, and K. Chihara, "Real-time gesture recognition by learning and selective control of visual interest points," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 27, no. 3, pp. 351-364, March 2005
- [2] J. Kela, P. Korpipää, J. Mäntyjärvi, S. Kallio, G. Savino, L. Jozzo and D. Marca, "Accelerometer-based gesture control for a design environment," *Personal and Ubiquitous Computing*, Vol. 10, no. 5, pp. 285-299, July 2006
- [3] S. Kallio, J. Kela, J. Mäntyjärvi and J. Plomp, "Visualization: short papers: Visualization of hand gestures for pervasive computing environments," *Proceedings of the working conference on AVI '06*, pp. 480 - 483, May 2006
- [4] J. Payne, P. Keir, J. Elgoyhen, M. McLundie, M. Naef, M. Horner and P. Anderson "Work-in-progress: Game play issues in the design of spatial 3D gestures for video games," *Conference on Human Factors in Computing Systems'06*, pp. 1217-1222, April 2006
- [5] E.-S. Choi, W.-C. Bang, S.-J. Cho, J. Yang, D.-Y. Kim and S.-R. Kim, "Beatbox music phone: gesture-based interactive mobile phone using a tri axis accelerometer," *ICIT 2005*, pp. 97-102, 14-17 Dec. 2005