

관성 센서를 이용한 위치기반 가상 멀티 타악기

A Position-based Virtual Multi-Percussion using Inertial Sensors

최은석, Eun-Seok Choi, 손준일, Jun-Il Sohn, 방원철, Won-Chul Bang, 김연배, Yeun-Bae Kim
삼성종합기술원 Advanced Systems Research Lab.

요약 관성 센서는 외부 장치의 도움 없이 3 차원 공간상에서 움직임 측정이 가능하다. 최근 MEMS 기술의 발달로 소형 저가 관성 센서(가속도 센서 혹은 각속도 센서) 제작이 가능해져 관성 센서를 소형 휴대 기기에 내장하여 사용자의 움직임을 감지하거나 의도 파악하는 연구가 진행되고 있다. 본 연구에서는 관성 센서가 내장된 휴대 기기를 이용하여 3 차원 공간상에서 6 가지 위치에 따라서 각기 다른 6 가지 소리를 발생하는 가상의 멀티 타악기 시스템을 제안한다. 즉, 휴대 기기를 상/하로 흔들면 가상 타악기의 타점 위치에 왔을 때 비트 음을 발생하고, 6 개의 다른 위치를 구분하여 다른 타점의 위치에서 휴대 기기를 흔들면 각각 그 위치와 미리 지정된 소리가 발생하도록 하였다. 이러한 가상의 멀티 타악기 시스템을 위해서 3 차원 공간상에서 실시간으로 사용자의 움직임을 감지하고 휴대 기기의 위치를 파악하는 것이 필요하다. 저가의 관성 센서를 이용하여 사용자가 휴대 기기를 움직이는 동작이 있는 상황에서 실시간으로 휴대 기기의 위치를 추정하는 것은 쉽지 않지만 본 연구에서는 다양한 사용자의 움직임 동작 분석을 통하여 사용자가 가상의 멀티 타악기를 상/하로 흔드는 동작을 감지하고 다른 위치로 이동하는 동작을 구분하였다. 개발된 동작 감지 알고리즘과 위치 구분 알고리즘을 휴대 기기에 적용되어 실제로 가상의 타악기 시스템을 구현하였다.

핵심어: 가속도 센서, 각속도 센서, 자이로, 타악기, 동작, 움직임

1. 서론

최근의 MEMS 기술의 발달로 관성 센서 (가속도 센서나 각속도 센서)의 크기는 점점 작아지고, 성능은 점점 향상되었다. 이러한 MEMS 센서 제작 기술의 발달로 예전에는 주로 항공기나 미사일과 같은 시스템에 장착되었던 관성 센서가 이제는 휴대폰과 같은 모바일 기기에 실장되어 하드 디스크 드라이브(HDD) 보호, 만보계, 공간 필기 인식, 공간형 입력장치 등에 적용되고 있다 [1-7]. 또한 관성 센서를 이용하여 3차원 공간에서 마우스나 전자기기 리모컨으로 사용하고자 하는 연구도 있었다[2]. 이와 같은 소형 관성 센서와 신호 처리 기술을 이용하여, 사용자가 3차원 공간에 쓴 문자를 인식하는 연구도 진행되었다[5, 6]. 최근에는 소형 관성 센서를 이용하여 사람의 동작 패턴을 이용하여 소리로 변환하는 등의 음악이나 엔터테인먼트를 위한 입력 장치에 대한 연구도 수행 되었다 [7-9]. 흔드는 동작에 따라 비트 음을 발생하는 비트 박스 기능이 있는 휴대전화기도 개발되어 상용화 되었다 [7].

전자 기술을 이용한 다양한 전자 악기 제품이 나와있는 요즘 소형 관성 센서를 이용하여 사람의 동작이나 움직임 패턴을 인식하여 음악 연주에 응용하려는 다양한 시도가 있다. [8]에서는 3축 가속도 센서와 2축 각속도 센서를 이용하여

일본식 북을 치는 북채를 제안하였다. 이 북채는 관성 센서와 마이크로프로세서, 배터리, Bluetooth 등을 이용하여 센서의 신호를 무선으로 PC로 보내도록 되어있다. [9]에서는 가속도 센서를 이용하여 beat boxing 장치를 만들어서 이를 음악 연주에 이용하였다. 가속도 센서를 이용하여 특정 임계치를 넘는 동작을 감지하여 hit 동작을 검출하거나 최대 가속도 및 가속도의 유지 구간 등을 검출하여 특정 악기와 연결하여 동작과 소리를 매핑하는 연구도 진행되었다.

위의 연구에서는 사람이 움직임의 세기와 길이 등을 이용하여 소리를 발생하였기 때문에 다양한 종류의 악기 소리를 낼 수가 없다. 또한 구현을 위해서 사람의 움직임을 감지하는 센싱부와 소리를 발생하는 부분이 분리 되어 있어서 휴대하기가 불편하다는 단점이 있다.

본 연구에서는 3축 가속도 센서와 3축 각속도 센서를 이용하여 사용자가 휴대기기를 흔드는 위치를 감지하고, 그 위치에 따라서 다른 비트 음을 발생하는 휴대기기를 개발하였다. 이는 기존에 개발된 비트 박스 기능보다 더 발전된 형태로 특별한 버튼 조작 없이 하나의 휴대 기기에서 비트 음을 발생시키는 위치에 따라서 다양한 악기 소리 발생이 가능하다. 본 연구에서 개발된 기능을 이용하면 3차원 공간에서 특별한 장치 없이 휴대 기기를 다양한 비트 음을 발생하는 멀티 타악기 시스템으로 이용 가능하다.

2. 전체 시스템 구조

본 연구에서는 3축 가속도 센서와 3축 각속도 센서를 이용하여 가상의 멀티 타악기 시스템의 센싱부를 구성하였다. 그림 1은 각 센서가 측정하는 물리량의 축을 정의한 것이다. 여기서 Ab_x , Ab_y , Ab_z 는 각각 x, y, z 축 방향의 가속도를 나타낸다. 본 연구에서는 하나의 3축 가속도 센서를 이용하여 3축 x, y, z 축 방향의 가속도를 측정한다. Wb_x , Wb_y , Wb_z 는 각각 x, y, z축을 중심으로 회전하는 각의 각속도를 나타내는 것이며, 3개의 1축 각속도 센서로부터 얻은 신호를 이용하여 각축의 각속도 신호를 측정한다.

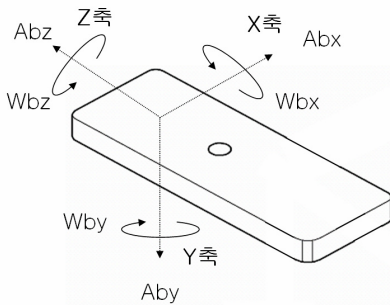


그림 1. 가속도 및 각속도 신호의 축 정의

그림 2는 가속도 센서(Accelerometer)와 각속도 센서(Gyroscope)의 신호를 받아서 처리하는 전체 하드웨어 구조를 나타낸 것이다. 가속도 센서와 각속도 센서 신호는 A/D 변환기에 의하여 디지털 값으로 변환되고 이것을 일정주기로 읽어와서 CPU에서 처리하게 된다. 신호 처리 알고리즘의 결과에 따라서 각 위치에 맞는 소리를 출력하라는 이벤트가 발생하면 최종적으로 스피커에서 미리 저장된 소리를 발생시킨다.

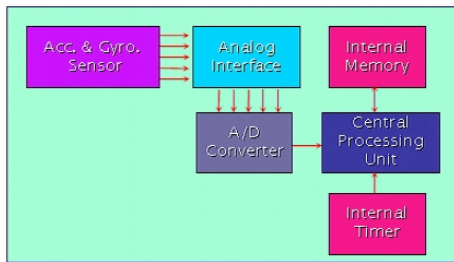


그림 2. 관성 센서 신호처리를 위한 H/W 전체 시스템

그림 3은 전체 시스템 구성의 흐름도를 나타내고 있다. 사용자가 개발된 시스템으로 소리를 발생시키기 위한 동작을 수행하면 일정 주기로 가속도 센서와 각속도 센서 값을 읽어온다. 그런 후에 그림 1에서 정의한 축으로 축 변환을 하게 된다. 이렇게 변환하는 이유는 하드웨어 설계에 따라서 가속도 센서나 각속도 센서의 위치가 일정하지 않을 수 있기 때문에 이것을 통일 시키기 위해서이다. 이렇게 공통으로 정의된 축이 있어야 알고리즘 개발할 때 혼돈되지 않고, 정해진 축에 대해서 알고리즘을 개발할 수가 있다.

이와 동시에 센서로부터 들어온 전압 값을 의미 있는 물리량으로 변환하고 calibration하는 과정을 거친다. 센서로부터

들어온 전압 값을 물리량으로 변환하는 것은 가속도 센서로부터 들어온 전압은 가속도 신호(m/sec^2)로 변환하고, 각속도 센서로부터 들어온 전압은 각속도 신호(rad/sec)로 변환한다. 이렇게 물리량으로 변환해야 센서 신호 이상 유무를 잘 판단할 수 있고, 센서 테스트 및 calibration을 수행하는데에 유리하다. 물리량으로 변환 시, 센서 자체의 오차로 offset과 gain이 센서의 사양서(specification)에 나와 있는 값과 잘 맞지 않는 경우가 많다. 따라서 이를 보정하기 위해서 calibration 과정이 반드시 필요하다.

이렇게 물리량으로 변환된 신호를 이용하여 사용자의 움직임에 따라서 동작을 감지하고 사용자가 어느 위치에서 움직임이 있었는지를 구분하게 된다. 사용자의 움직임이 감지되면 소리를 발생시키는 이벤트를 발생하고 이 이벤트에 따라서 소리가 발생된다. 또한 사용자가 소리를 발생시킨 위치를 구분하게 되는데, 이렇게 구분된 위치에 따라서 특정 위치에 따라 미리 지정 되어있는 저장된 소리를 발생한다.

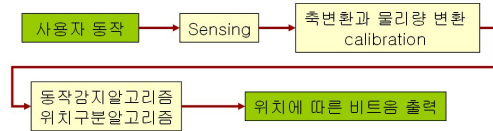


그림 3. 전체 시스템 구성도 흐름도

3. 하드웨어 구성

본 연구를 위한 하드웨어 구성은 그림 4와 같다. 마이크로 프로세서가 있는 CPU 보드에 3축 가속도 센서 1개와 1축 각속도 센서 3개가 연결되어 있다. 음악 파일 저장을 위하여 SD Card slot도 연결되어 있다. 본 연구에서는 가속도 센서와 각속도 센서 그리고 SD Card slot이 있는 센서보드로 따로 제작(그림 5 참조)하여 CPU 보드에 연결하였다.

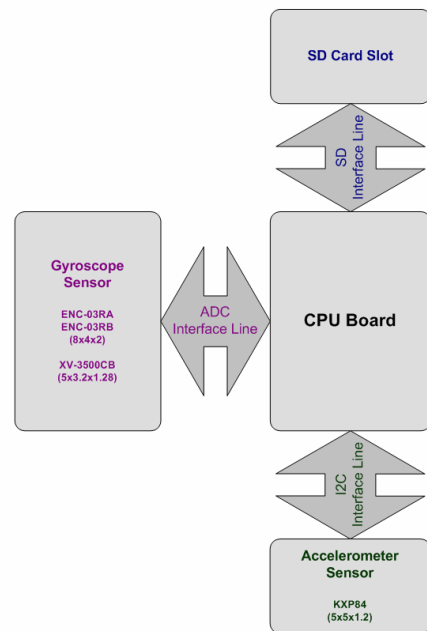


그림 4. 하드웨어 구성도

가속도 센서는 Kionix 사의 3축 가속도 센서인 KXP84를 이용하였다. KXP84의 경우 12bit 아날로그 디지털 변환기 (Analog-Digital Converter, ADC)가 내장되어있고 A/D변환을 한 결과값은 CPU 보드와 I²C로 인터페이스 한다. Kionix 사의 KXP84의 자세한 사양은 표 1과 같다 [10]. 가속도 센서의 경우 신호의 크기가 작기 때문에 증폭기(Amplifier)를 거쳐서 CPU 보드에 내장된 10bit ADC로 연결된다. 사용된 가속도 센서는 muRata 사의 소형 자이로 센서인 ENC-03R과 Epson Toyocom사의 XV-3500CB이다. 각 가속도 센서의 자세한 사양은 표 2와 표 3과 같다 [11, 12].

SD card와 CPU보드와는 SD interface 라인으로 연결되어 있다. SD card는 음악 파일과 비트 음 파일이 저장용으로 이용된다. 여기에 저장된 음악 파일이나 비트 음 파일은 이벤트 신호가 발생하면 출력하도록 프로그램 되어있다.

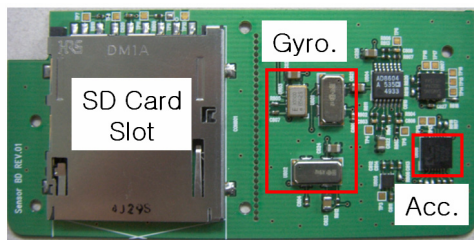


그림 5. 제작한 센서보드

표 1. 가속도 센서 Kionix KXP84 사양 [10]

항목	사양	비고
Range	±2.0 g	g: 중력가속도
Sensitivity	819 count/g	12bit ADC 내장
Output (at 0g)	2048 count	
Power voltage	2.7~5.25 Vdc	
Dimension	5×5×1.2 mm	

표 2. 가속도 센서 muRata ENC-03R 사양 [11]

항목	사양
Range	±300 deg/sec
Sensitivity	0.67 mV/deg/sec
Output (at angular velocity = 0)	1.35 Vdc
Power voltage	2.7~5.25 Vdc
Dimension	8×4×2.0 mm

표 3. 가속도 센서 Epson Toyocom XV-3500CB 사양 [12]

항목	사양
Range	±100 deg/sec
Sensitivity	0.67 mV/deg/sec
Output (at angular velocity = 0)	1.35 Vdc

Power voltage	2.7~3.3 Vdc
Dimension	5×3.2×1.3 mm

4. 신호 처리 알고리즘 구성

가상의 타악기 시스템을 구성을 위해서는 비트 음을 발생하기 위하여 상/하로 흔드는 동작을 감지하는 부분과 위치를 파악하기 위하여 위치를 구분하는 알고리즘이 필요하다.

4.1. 동작 감지 알고리즘

본 연구에서 이야기하는 동작 감지는 그림 6과 같이 사용자가 가상의 멀티 타악기를 손에 들고 상/하 방향으로 흔들었을 때, 흔드는 동작이 있는지를 판단하는 것을 말한다.



그림 6. 가상의 멀티 타악기를 상/하 방향으로 흔드는 모양

사용자가 흔드는 동작을 감지하여 실시간으로 소리를 발생 시키는 장치를 위해서는 동작 감지 시 시간 지연(time delay)이 없어야 하고, 사용자의 동작을 정확히 감지해야 하는 조건이 요구된다. 시간 지연이 없어야 되는 이유는 사용자가 가상의 멀티 타악기를 흔들었을 때, 흔드는 동작과 소리가 잘 동기화 되어야 실제로 타악기를 연주하는 듯한 느낌을 줄 수 있기 때문이다.

그림 6과 같이 상/하 방향으로 흔드는 동작을 감지하기 위해서 본 연구에서는 그림 1의 Y축 가속도 신호인 Aby와 X축 가속도 신호인 Wbx를 이용하였다. 상/하 방향으로 흔드는 동작은 그림 1에서 Y축을 기준으로 직선 운동이 발생하거나, X축을 중심으로 회전이 일어나는 동작이므로 가속도 신호 Aby나 가속도 신호 Wbx를 이용하면 동작을 감지하는 것이 가능하다.

4.1.1. 가속도 신호를 이용한 동작 감지

가속도 신호를 이용하여 동작을 감지하기 위해서 본 연구에서는 Aby 신호를 이용한다. 가속도 신호의 경우 항상 중력 가속도 성분이 신호에 포함되므로 이를 제거하고 순수한 사용자의 움직임에 의한 동작 신호만을 추출하기 위해서 이산 시간의 현재 샘플(k)의 가속도 신호 Aby(k)와 이전 샘플(k-1)의 가속도 신호 Aby(k-1)의 차를 이용하여 동작 감지에 이용한다.

가속도 신호에 의한 동작 감지 알고리즘은 아래와 같다 [7].

$$\xi(k) = \begin{cases} 1, & \text{if } |\delta A_{by}(k)| > \eta_1, \text{ and } A_{by}(k) > \eta_2 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

여기서 $\delta A_{by}(k) = A_{by}(k) - A_{by}(k-1)$ 이고 η_1 과 η_2 는 상수로 미리 결정된 임계치(threshold)이다. $\xi(k)$ 가 0에서 1로 변할 때, 미리 저장된 소리를 발생시킨다.

4.1.2 각속도 신호를 이용한 동작 감지

가속도 신호와 달리 각속도 신호는 중력에 의한 영향 없으므로 움직임에 의해 발생한 각속도 신호를 동작 감지 신호로 바로 이용 가능하다. 각속도 신호에 의한 동작 감지 알고리즘은 다음과 같다.

$$\lambda(k) = \begin{cases} 1, & \text{if } |\omega_{bx}| > \varepsilon_1 \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

여기서 ε_1 은 상수로 미리 결정된 임계치(threshold)이다. $\lambda(k)$ 가 0에서 1로 변할 때, 미리 저장된 소리를 발생시킨다.

4.2. 위치 구분 알고리즘

본 연구에서 구분을 목표로 하는 6개의 위치는 그림 7과 같다. 우선 상단과 하단을 구분해야 하고, 상단과 하단 각각 왼쪽(A와 D), 중간(B와 E), 오른쪽(C와 F)을 구분해야 한다. 사용자마다 움직임의 범위와 행동 패턴이 너무나도 다양하기에 본 연구에서는 사용자가 움직이는 데에 불편함이 없는 범위 안에서 아래와 같은 제약 조건을 두었다.

(제약 조건 1) 왼쪽, 중간, 오른쪽 위치 구분은 사용자의 팔꿈치를 중심으로 약 30도씩의 간격 차이가 있다고 가정한다.

(제약 조건 2) 하단을 기준으로 할 때, 팔꿈치를 중심으로 상단과 하단의 각도는 약 45도라고 가정한다.

(제약 조건 3) 가상의 멀티 타악기 장치를 정해진 방향으로 잡고 동작 한다. 즉, 그림 1에서 Y축을 중력 방향으로 Z축을 앞을 향하도록 유지하여 잡는다.

(제약 조건 4) 처음 시작할 때의 위치는 항상 하단의 왼쪽, 즉 A 위치 이다.

(제약 조건 5) 하나의 위치에서 다른 하나의 위치로 변경하면 반드시 1회 이상 상/하로 흔들여 비트 음 발생 동작을 수행한다.

그림 7에서 A/B/C는 하단에 위치 하였고, D/E/F는 상단에 위치하였다. 각각의 A/B/C/D/E/F 위치에서 그림 6과 같이 상/하로 비트 음을 발생시키기 위한 동작을 수행하면, 해당 위치에 맞는 비트 음을 식 (1)이나 식 (2)를 이용해서 발생시킨다.

위치 구분을 위해서 앞에서 언급한 ‘제약 조건 1~5’ 상황에서 획득한 사용자 data를 분석하여 알고리즘이 설계되었다. 알고리즘은 상단/하단의 위치를 구분하는 수직 위치 구분

고리즘과 왼쪽/중간/오른쪽의 위치를 구분하는 수평 위치 구분 알고리즘으로 구성되어 있다.

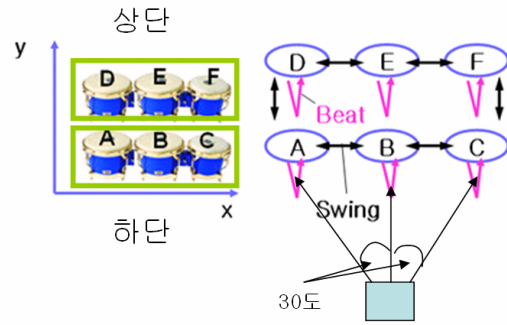


그림 7. 위치 구분을 위한 6개의 위치

4.2.1. 수직 위치 구분 알고리즘

본 연구에서 각속도 신호 W_{bx} 를 이용한 수직 위치 구분을 위한 알고리즘은 다음과 같다.

A. 초기 위치 설정

수직 위치 구분 알고리즘에서 이산 시간 k 에 대하여 상단과 하단의 위치 구분 결과를 $V(k)$ 에 저장할 때, $V(k) = 0$ 이면 하단 위치, $V(k) = 1$ 이면 상단 위치라고 정의하면, ‘제약 조건 4’로부터 시작 지점인 $k = 0$ 에서의 하단 위치이므로 $V(0) = 0$ 이다.

B. 상단 위치 검출

$V(k-1) = 0$ 이라는 가정하에서 아래의 조건을 만족하는지를 판단한다.

$$V(k) = \begin{cases} 1, & \text{if } m(k) < \mu_1, \text{ and } \theta_{bx}(k) > \mu_2 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

여기서 $m(k) = \min_{i=k-l, \dots, k} (W_{bx}(i))$ 이므로 l 샘플 이전부터 현재 샘플 k 까지 $W_{bx}(k)$ 의 최소값이다. 또한 $\theta_{bx}(k) = \sum_{i=0}^k W_{bx}(i)$ 는 $W_{bx}(k)$ 를 이산 시간 k 에 대하여 적분한 값으로 각도를 나타낸다, μ_1 과 μ_2 는 상수로서 미리 결정된 임계치(threshold)이다.

C. 하단 위치 검출

$V(k-1) = 1$ 이라는 가정하에서 아래의 조건을 만족하는지를 판단한다.

$$V(k) = \begin{cases} 1, & \text{otherwise} \\ 0, & \text{if } \mu_3 < m(k) < \mu_4 \end{cases} \quad (4)$$

여기서 μ_3 과 μ_4 는 상수로서 미리 결정된 임계치(threshold)이다.

D. 위치 고정

‘계약 조건 5’로 인해서 식 (3)과 식 (4)의 조건을 만족하더라도 동작 감지가 발생하지 않으면 현재의 위치가 변하지 않는다.

4.2.2. 수평 위치 구분 알고리즘

본 연구에서 각속도 신호 W_{by} 를 이용한 수평 위치 구분을 위한 알고리즘은 다음과 같다.

A. 초기 위치 설정

‘계약 조건 4’로부터 초기에는 위치는 왼쪽이라고 가정한다. 수평 위치 구분 알고리즘에서 이산 시간 k 에 대하여 왼쪽, 중간, 오른쪽의 위치 구분 결과를 $H(k)$ 에 저장할 때, $H(k)=0$ 이면 왼쪽 위치, $H(k)=1$ 이면 중간 위치, $H(k)=2$ 이면 오른쪽 위치라고 정의하면 시작 위치는 왼쪽 위치이므로 $H(0)=0$ 이다.

B. 수평 위치 변화 검출

$H(k-1)=0$ 이라는 가정에서 수직 위치 구분을 위한 알고리즘은 다음과 같다.

$$H(k) = H(k-1) + dH(k) \quad (5)$$

여기서 $H(k)$ 의 최대 최소값은 각각 0과 2이다. 즉 $H(k)$ 값이 0보다 작으면 0이고, 2보다 크면 2이다. 또한 $dH(k)$ 는 다음과 같다.

$$dH(k) = \begin{cases} -2, & \text{if } \theta_{by}(k) < \alpha_1 \\ -1, & \text{if } \alpha_1 \leq \theta_{by}(k) < \alpha_2 \\ 0, & \text{if } \alpha_2 \leq \theta_{by}(k) < \alpha_3 \\ 1, & \text{if } \alpha_3 \leq \theta_{by}(k) < \alpha_4 \\ 2, & \text{if } \theta_{by}(k) \geq \alpha_4 \end{cases}$$

여기서 $\theta_{by}(k) = \sum_{i=0}^k W_{by}(i)$ 는 이산 시간 k 에 대하여 $W_{by}(k)$ 를 적분한 값으로 각도를 나타낸다. $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ 는 상수로서 미리 결정된 임계치(threshold)이다.

위와 같이 $dH(k)$ 의 값을 결정 짓는 θ_{by} 의 구간을 +

향과 -방향을 구분해 놓은 이유는 각속도 센서의 오차 혹은 비선형 성으로 인하여 +방향과 -방향의 각속도가 비대칭을 이루지 않기 때문이다.

C. 위치 고정

‘계약 조건 5’로 인해서 식 (3)의 조건을 만족하더라도 동

작 감지가 발생하지 않으면 현재의 위치가 변하지 않는다.

5. 실험

5.1. 실험 장치 및 방법

5.1.1. 실험 장치

알고리즘 개발 및 사용자 테스트를 위한 데이터 수집 장치는 그림 8과 같다. 가상의 멀티 타악기 장치를 컴퓨터와 유선 직렬 통신으로 연결하여 3축 가속도 신호 데이터와 3축 각속도 신호 데이터를 제작한 데이터 수집 프로그램을 이용하여 수집하였다. 데이터 수집 주기(T)는 0.01초이다.

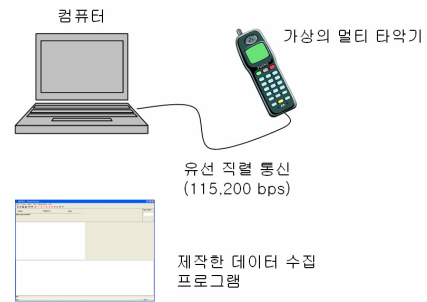


그림 8. 센서 데이터 수집 시스템

5.1.2. 실험 방법

본 연구에는 ‘계약 조건 1~5’를 만족하도록 하면서 다음과 같은 다양한 시나리오로 데이터를 수집하여 동작 감지 알고리즘과 수직/수평 위치 구분 알고리즘의 성능을 검증하였다.

■ 수직 위치 구분 검증을 위한 시나리오

- (SV1) A2=>D1=>A2=>D1=>A2=>D1=>A2=>D1
- (SV2) A2=>F1=>A2=>F1=>A2=>F1=>A2=>F1
- (SV3) C2=>D1=>C2=>D1=>C2=>D1=>C2=>D1
- (SV4) C2=>F1=>C2=>F1=>C2=>F1=>C2=>F1
- (SV5) A2=>D1=>C2=>F1=>A2=>D1=>C2=>F1
- (SV6) A2=>F1=>C2=>D1=>A2=>F1=>C2=>D1

■ 수평 위치 구분 검증을 위한 시나리오

- (SH1) A1B1=>A1B1
- (SH2) A4B4=>A4B4
- (SH3) A1C1=>A1C1
- (SH4) A4C4=>A4C4
- (SH5) A1B1C1=>A1B1C1
- (SH6) A4B4C4=>A4B4C4

여기서 X_n 은 그림 7에 있는 X 위치에서 n회 반복해서 비

트 음을 발생시키는 동작을 하라는 의미이다. 또한 ‘=>’ 는 다음 동작으로 넘어가라는 의미이다.

5.2. 알고리즘 테스트

그림 9는 ‘수직 위치 구분을 위한 시나리오’에서 (SV2)를 수행했을 때, 동작 감지 알고리즘(식 (2))과 수직 위치 구분 알고리즘을 통하여 동작을 감지하고 위치를 구분한 결과를 보여준다. 시작 위치는 그림 7의 하단 A이며 비트 음 발생 시키는 동작을 2회하고 상단 위치 F로 옮겨서 비트 음 발생 동작을 1회 하는 것을 4회 반복하였다. 동작 감지와 위치 구분 모두 잘 되는 것을 볼 수 있다.

그림 10은 ‘수평 위치 구분을 위한 시나리오’에서 (SH5)를 수행했을 때, 동작 감지 알고리즘(식 (2))과 수평 위치 구분 알고리즘을 통하여 동작을 감지하고 위치를 구분한 결과를 보여준다. A위치에서 시작하여 비트 음 발생 동작을 1회 수행하고, B위치 C위치에서도 각각 1회 비트 음 발생 동작을 수행한 후에 다시 이 과정을 반복하는 시나리오를 수행한 것으로 동작 감지와 수평 위치 구분이 잘 되는 것을 볼 수 있다.

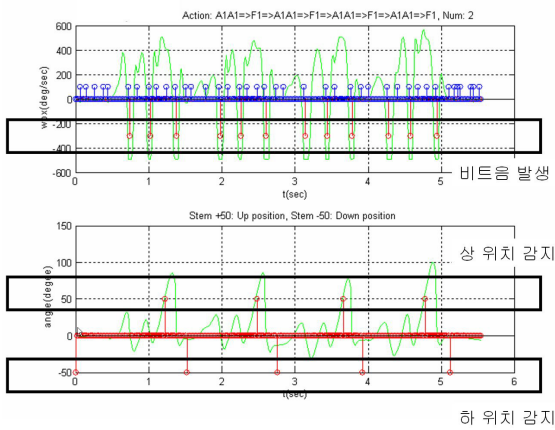


그림 9. 수직 위치 구분 결과

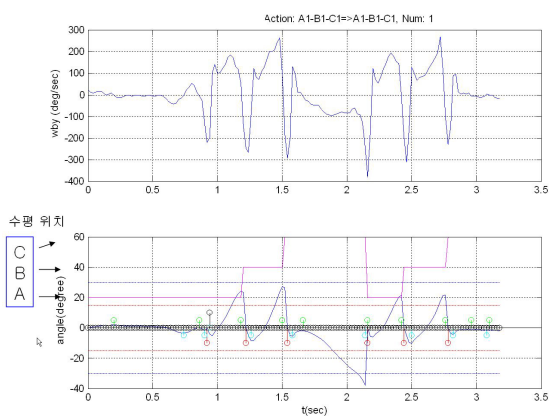


그림 10. 수평 위치 구분 결과

5.3. 사용자 테스트

본 연구에서 개발한 시스템을 ‘제약 조건 1~5’의 상황에서 수직/수평 위치 구분 검증을 위한 시나리오 (SV1~SV6,

SH1~SH6)를 수행하여 피실험자 4명에게 데이터를 수집하여 본 연구의 성능을 검증하였다.

표 3은 수직 위치 구분 시나리오와 피실험자에 따른 수직 위치 구분의 인식률을 나타낸다. 수직 위치 구분의 경우 전체 인식률 평균은 100%로 수직 위치 부분의 경우 상단과 하단 위치 구분이 잘 됨을 알 수 있다.

표 4는 수평 위치 구분 시나리오와 피실험자에 따른 수평 위치 구분의 인식률을 나타낸다. 수평 위치 구분의 경우 전체 인식률 평균은 92.4%이다. 표 4를 보면 수평 2개의 위치 구분하는 것이 수평 3개의 위치를 구분하는 것보다 인식률이 더 좋은 것을 알 수 있다. 또한 비트 음 발생 동작을 4회 수행하는 것보다 1회 수행하는 것이 인식률이 높음을 알 수 있다. 이는 사용자가 비트 음 발생 동작을 여러 번 수행할수록 센서의 오차로 인하여 각도 오차가 커짐에서 기인한 것이다.

표 3과 표 4의 결과는 미리 저장된 센서 데이터를 인식률이 최고가 되도록 각 알고리즘의 파라미터를 최적화 하여 얻은 것이다.

표 3. 수직 위치 구분 시나리오의 인식률

피실험자	인식률(%)					
	SV1	SV2	SV3	SV4	SV5	SV6
1	100	100	100	100	100	100
2	100	100	100	100	100	100
3	100	100	100	100	100	100
4	100	100	100	100	100	100
평균	100	100	100	100	100	100

표 4. 수직 위치 구분 시나리오의 인식률

피실험자	인식률(%)					
	SH1	SH2	SH3	SH4	SH5	SH6
1	100	100	100	81.5	71.4	63.64
2	100	87.5	100	100	100	90.9
3	100	75	100	100	100	87.9
4	100	100	100	100	71.4	87.9
평균	100	90.6	100	95.4	85.7	82.6

6. 결론

본 연구에서는 가속도 센서와 각속도 센서가 내장된 휴대 기기를 이용하여 3차원 공간에서 사용자의 상/하로 흔들리는 동작을 감지하고 휴대 기기의 위치를 구분하는 알고리즘을 제안하였다. 이렇게 제안된 알고리즘은 실제 시스템에 적용하여, 사용자의 비트 음 발생을 위하여 상/하로 흔들리는 동작을 감지하고 6 곳의 위치를 구분하여 위치에 따른 각기 다른 비트 음을 발생 시켜 가상의 멀티 타악기 시스템을 구현하였다.

본 연구에서 저가의 관성 센서를 이용하여 3차원 공간상에서 흔들리는 움직임이 있는 상황에서 휴대 기기의 위치를 구분하는 것이 가능함을 보였다. 본 연구에서 얻은 결과는 인식률이 최고가 되도록 각 알고리즘의 파라미터를 최적화 한 것이다. 하지만 관성 센서의 오차와 사용자의 다양한 사용 패턴 등이 있는 상황에서는 최적화가 쉽지 않을 것이다. 향후 좀더 정밀하고 안정적인 측정이 가능한 관성 센서가 개발 되고, 알고리즘을 좀더 정교화한다면 좀더 정확도 높은 위치 구분이 가능할 것이며, 더 많은 수의 위치 구분이 가능할 것으로 기대된다.

참고문헌

- [1] T. Miyagawa, Y. Yonezawa, K. Itoh, , and M. Hashimoto, "Handwritten pattern reproduction using 3d inertial measurement of handwriting movement," Transactions of the Society of Instrument and Control Engineers, vol. 38, pp. 1~8, Jan. 2002.
- [2] W.-C. Bang, W. Chang, , K.-H. Kang, E.-S. Choi, A. Potanin, and D.-Y. Kim, "Self-contained spatial input device for wearable computers," in Proceedings of the 7th IEEE International Symposium on Wearable Computers (ISWC 2003), White Plains, NY, October 21-23 2003, pp. 26~34.
- [3] S.-J. Cho, J.-K. Oh, W.-C. Bang, W. Chang, E.-S. Choi, J. Yang, J.-K. Cho, and D.-Y. Kim, "MagicWand: A hand-drawn gesture input device in 3-d space with inertial sensors," in In Proceedings of International Workshop of Frontiers Handwriting Recognition (IWFHR 2004), Tokyo, Japan, October 26-29 2004, pp. 106~111.
- [4] E.-S. Choi, W. Chang, W.-C. Bang, J. Yang, S.-J. Cho, J.-K. Oh, J.-K. Cho, and D.-Y. Kim, "Development of the gyro-free handwriting input device based on inertial navigation system (INS) theory," in Proceeding of SICE Annual Conference 2004 (SICE2004), Sapporo, Japan, August 4-6 2004.
- [5] J. Mantyjarvi, J. Kela, P. Korpipaa, and S. Kallio, "Enabling fast and effortless customisation in accelerometer based gesture interaction," in The 3rd International Conference on Mobile and Ubiquitous Multimedia (MUM '04). New York: ACM Press, October 2004, pp. 25~31.
- [6] S.-J. Cho and J.-H. Kim, "Bayesian network modeling of strokes and their relationships for on-line handwriting recognition," Pattern Recognition, vol. 37, no. 2, pp. 253~264, 2004.
- [7] 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", In Proceedings of IEEE International Conference on Industrial Technology 2005 (ICIT2005), Hong Kong, December 14-16, 2005, pp. 97~102.
- [8] D. Young, and I. Fujinaga, "AoBachi: A New Interface for Japanese Drumming," in Proceedings of the 2004 International Conference on New Interface for Musical Expression (NIME04), Hamamatsu, Japan, 2004, pp. 23~26
- [9] R. Lugo, and D. Jack, "Beat Boxing: Expressive Control for Electronic Music Performance and Musical Applications," in Proceedings of the 2005 International Conference on New Interface for Musical Expression (NIME05), Vancouver, BC, Canada, 2005, pp. 246~247.
- [10] Kionix Inc., "<http://www.kionix.com/>".
- [11] muRata Inc., "<http://www.murata.com/>".
- [12] Epson Toyocom Inc., "<http://www.epsontoyocom.co.jp/>".