



Motion Test

IISL

2020.08.12

이소운, 이연주, 유승희, 박은화, 전성운



| Index

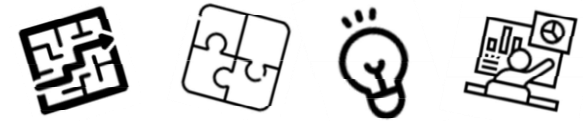
- 데이터 지연
- 타겟 동작
- 알파벳 별 그래프

기본 데이터 지연



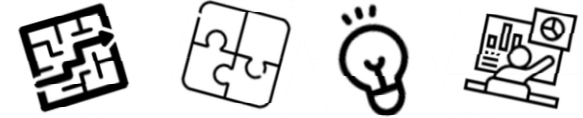
- `SENSOR_DELAY_NORMAL` ($200,000\mu s = 200ms$)
- `SENSOR_DELAY_GAME` ($20,000\mu s = 20ms$)
- `SENSOR_DELAY_UI` ($60,000\mu s = 60ms$)
- `SENSOR_DELAY_FASTEST` ($0\mu s = 0ms$)

타겟 동작(모션 및 근거)



- 갤럭시 노트 20에서 제시한 동작: 뒤로, 최근 앱, 홈 화면, 캡처 메모
- 특정 상황에서 터치 인식이 어려울 때 불편했던 점/위급한 상황 고려

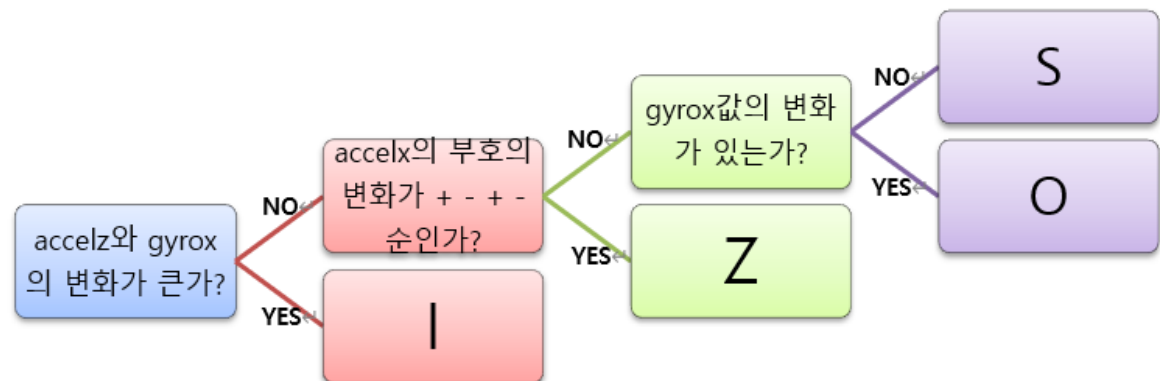
동작	모션	근거
→ 긴급 전화	흔들기	위급한 상황, 정신 차리고 전화번호 누르기에 한계가 있을 것 같다.
와이파이 켜기	I	
삼성 페이	S (Samsung)	갤럭시를 사용하는 사람들이 많이 이용하는 기능 중 하나.
잠금해제	Z (Zip)	비 오는 날 잠금 해제가 잘 안 풀려 비활성화된 경우가 많았다.

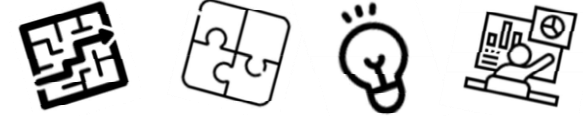


■ 그래프 분석1

➤ 저번 시간에 실험한 것을 바탕으로 I, O, S, Z 각각의 가속도 및 자이로 센서 그래프가 가지는 특징을 분석해본다.

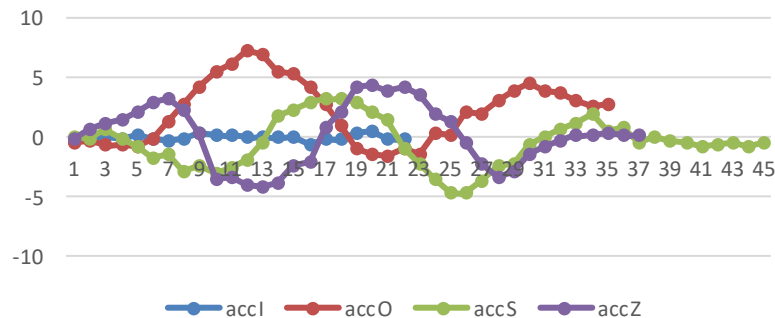
1. I의 경우: O, S, Z와 구별되
합을 구하여 I를 가장 먼저
2. Z의 경우: O와 S에서의 acc
다르다. 예를 들어, accel X는
다. 따라서 부호의 변화를 배
3. O와 S는 그래프에서 큰 차
변화가 크다는 점을 이용하여 :



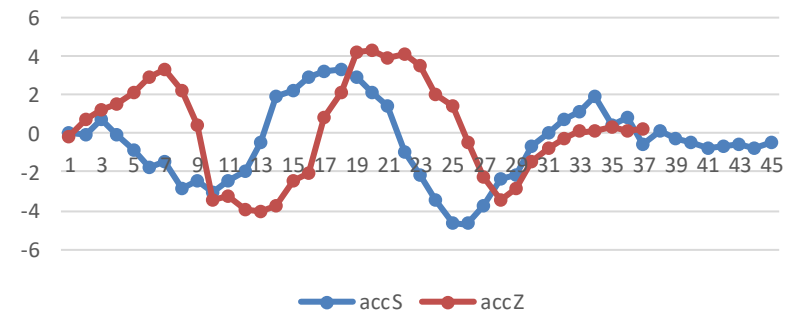


■ 그래프 분석 2

acc x 비교

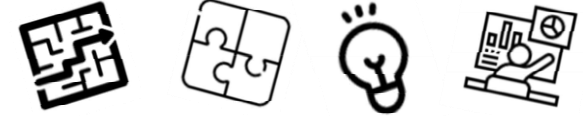


S, Z accx 값 비교



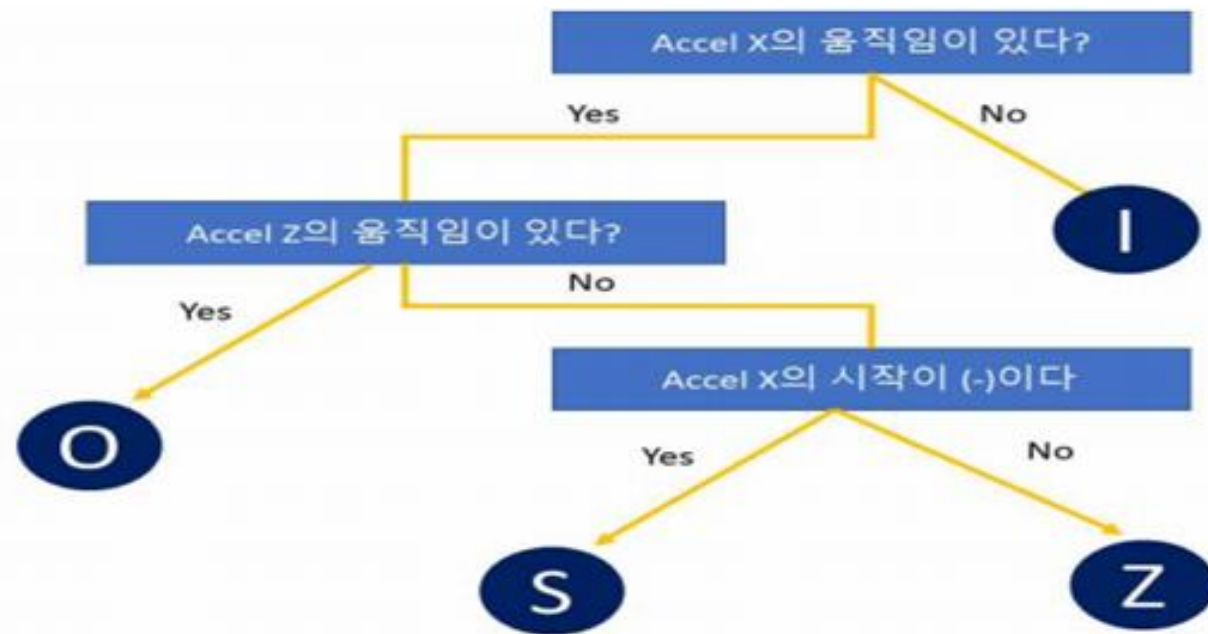
1. | 구별: 측정 속도가 알파벳마다 다르긴 하지만 값 변화 추이를 볼 때, 다른 알파벳에 비해 | 의 accel X 값 변화가 굉장히 작다는 것을 확인할 수 있었다.
2. S와 Z 구별: accel X의 값이 +인지 아닌지에 따라 S, Z를 구별할 수 있다는 가정을 세웠다. 하지만 시작 값 자체가 같은 알파벳에 있어서도 일정하지 않다는 결과를 얻었다. 따라서 구별할 수 있는 다른 방법을 생각해 보았다. S는 극점이 -+-+를 찍는 반면, Z는 +-+-를 찍었다. → 시작값의 부호를 기준으로 삼기보다 최대/최소값의 변화 부호를 보고 S, Z를 구분할 수 있을 것 같다.

Lab



■ 그래프 분석3

1. I는 회전 성분이 없
2. O는 accel X와 c
3. S는 accel Z의 변
4. Z는 accel Z의 변

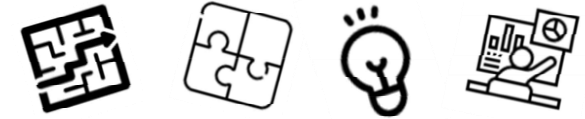


Lab

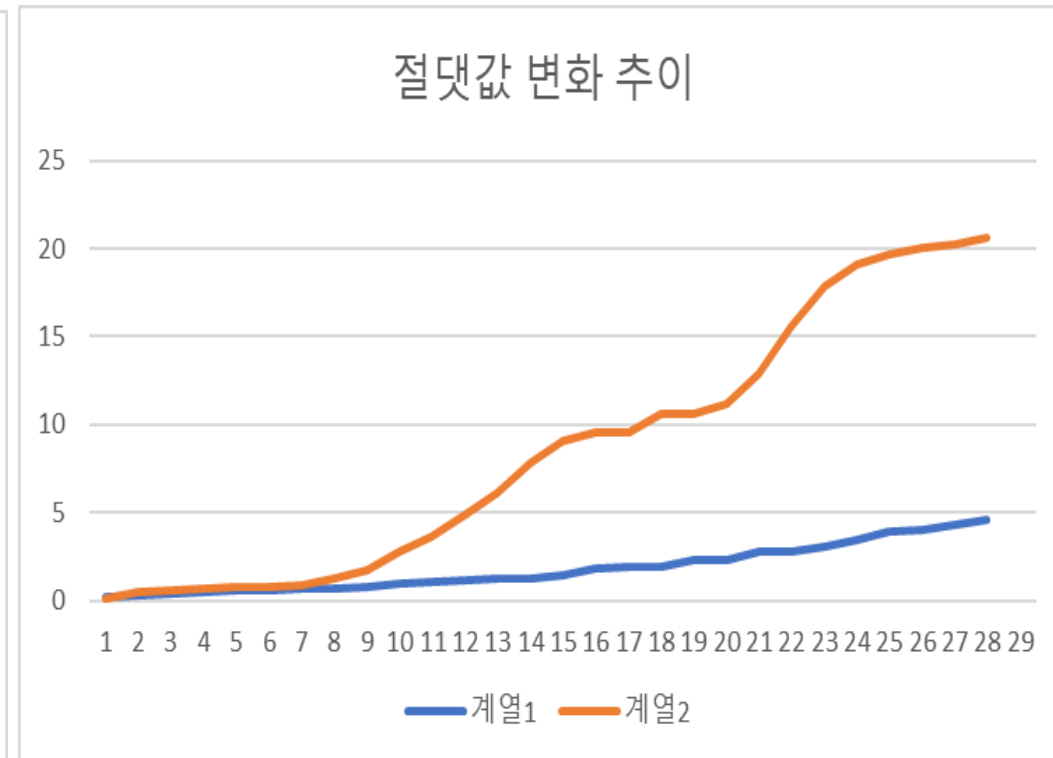
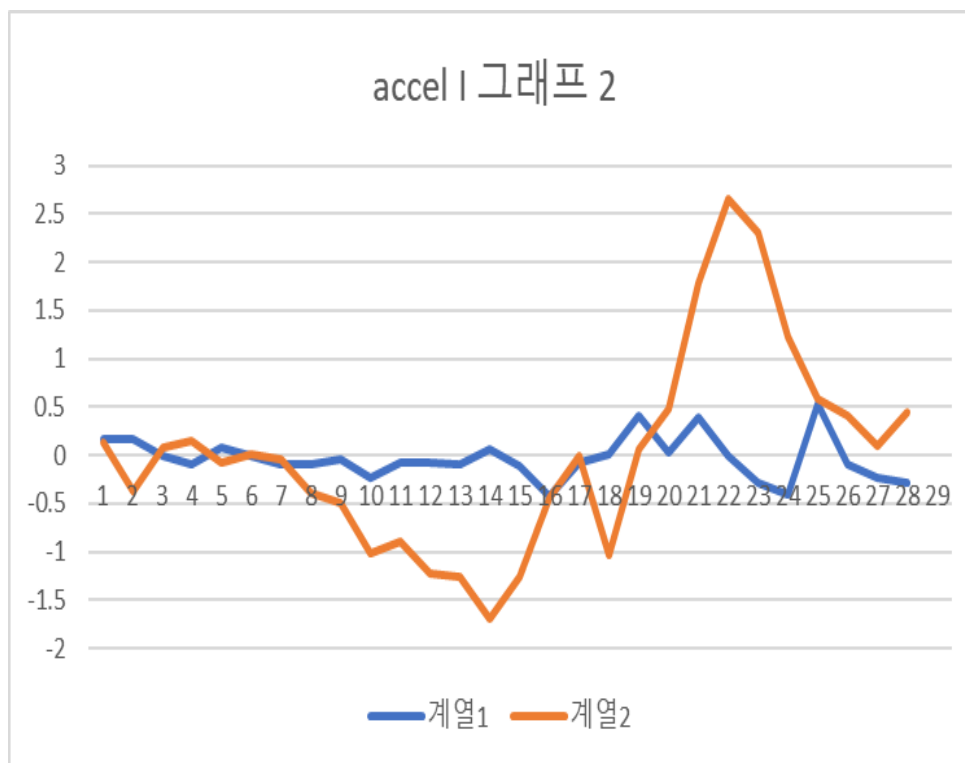


- 실험 개요: 기존 실험까지는 가속도 센서를 이용하여 중력 가속도가 포함된 결과 값을 얻었다. 하지만 분석 결과 선형 가속도를 사용하는 것이 각 축의 움직임을 같은 기준점인 0으로부터 떨어진 정도로 보다 쉽게 비교할 수 있고, 중력 가속도 값을 제거하여 더 높은 정확도를 보일 수 있다고 생각했다.
- 실험 방법: 동일
- 실험 분석 방법: 알파벳에 따라 실험을 진행, accel X와 accel Z의 절대값 합 변화 추이를 출력하여 알파벳을 구분할 수 있는 결과를 얻고자 하였다.

Lab



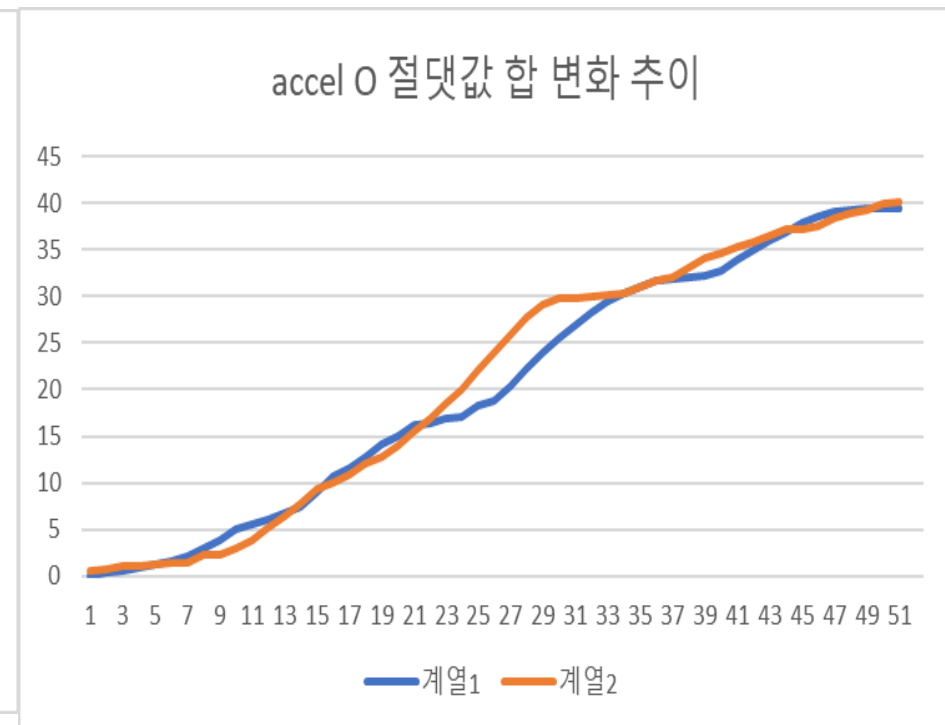
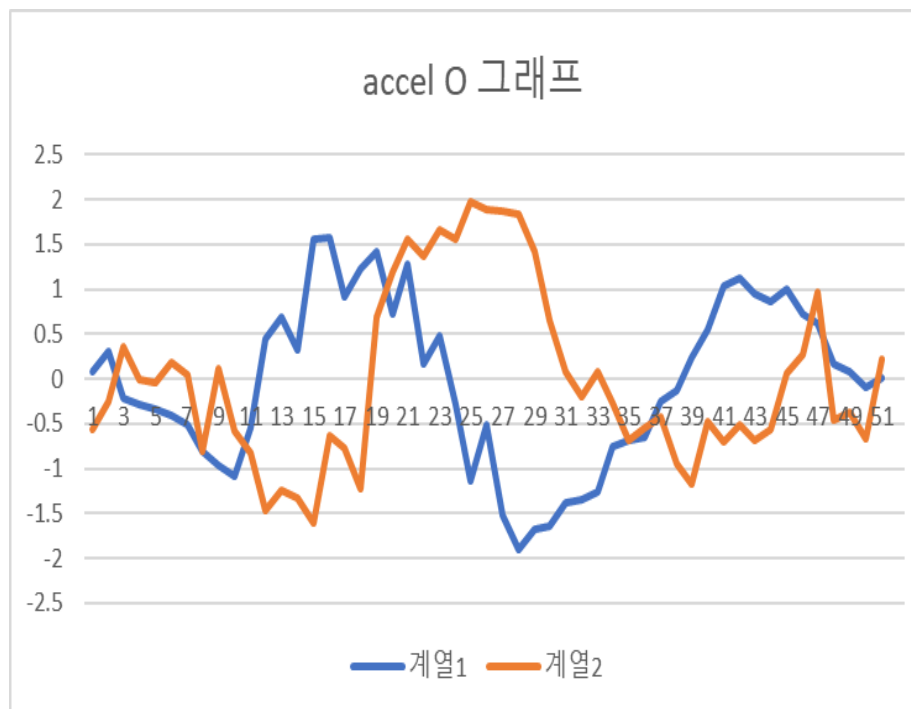
■ 실험 결과 1 – ‘1’ (계열 1: accel X, 계열 2: accel Z)



Lab



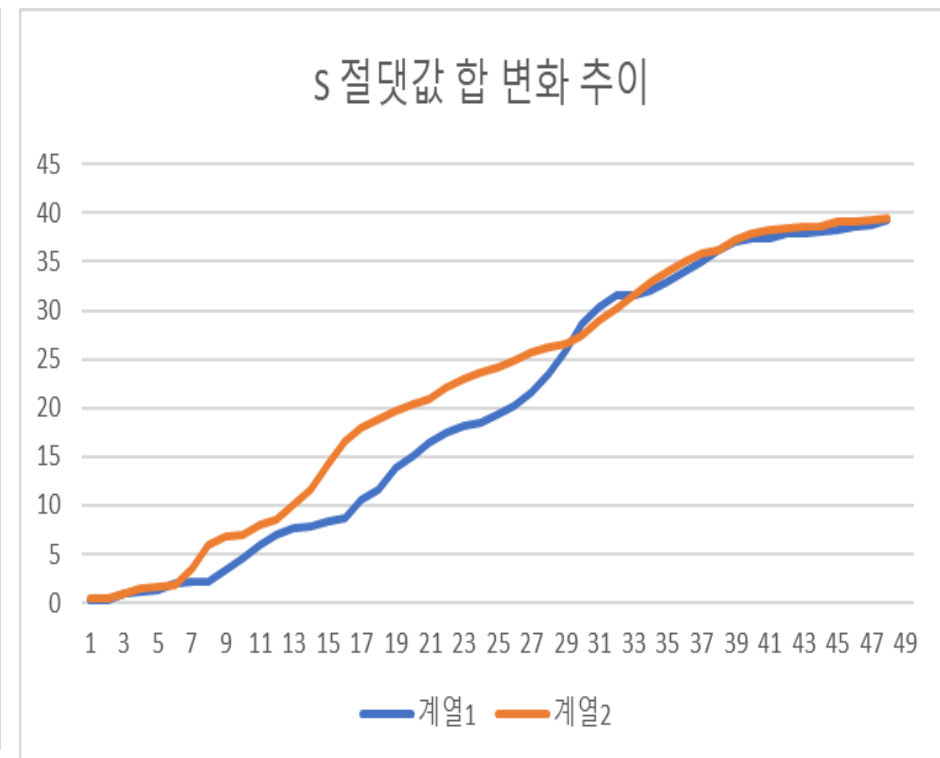
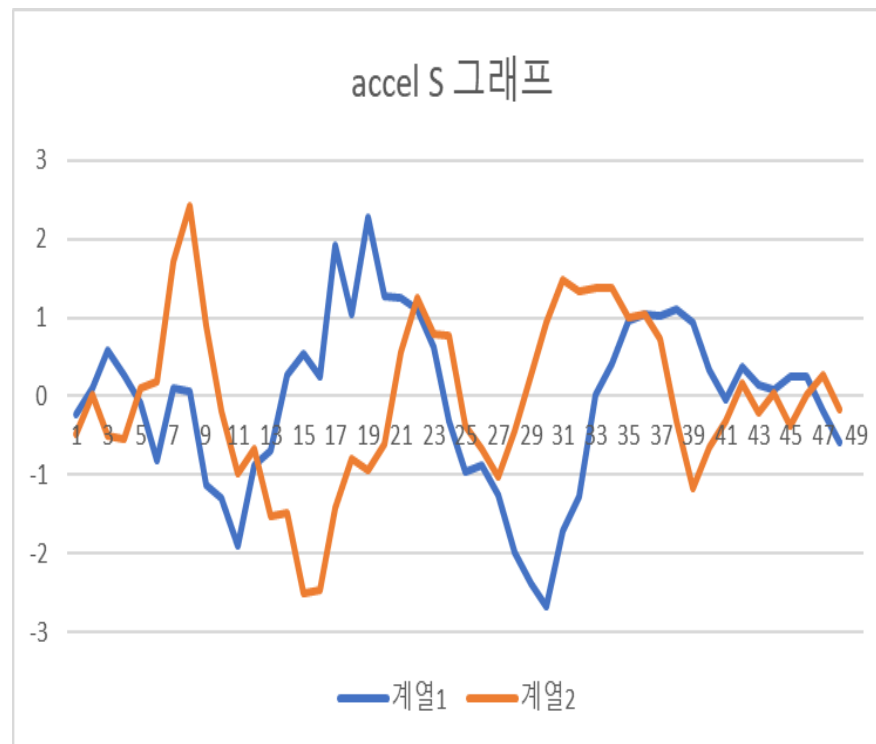
■ 실험 결과 2 – ‘O’ (계열 1: accel X, 계열 2: accel Z)



Lab



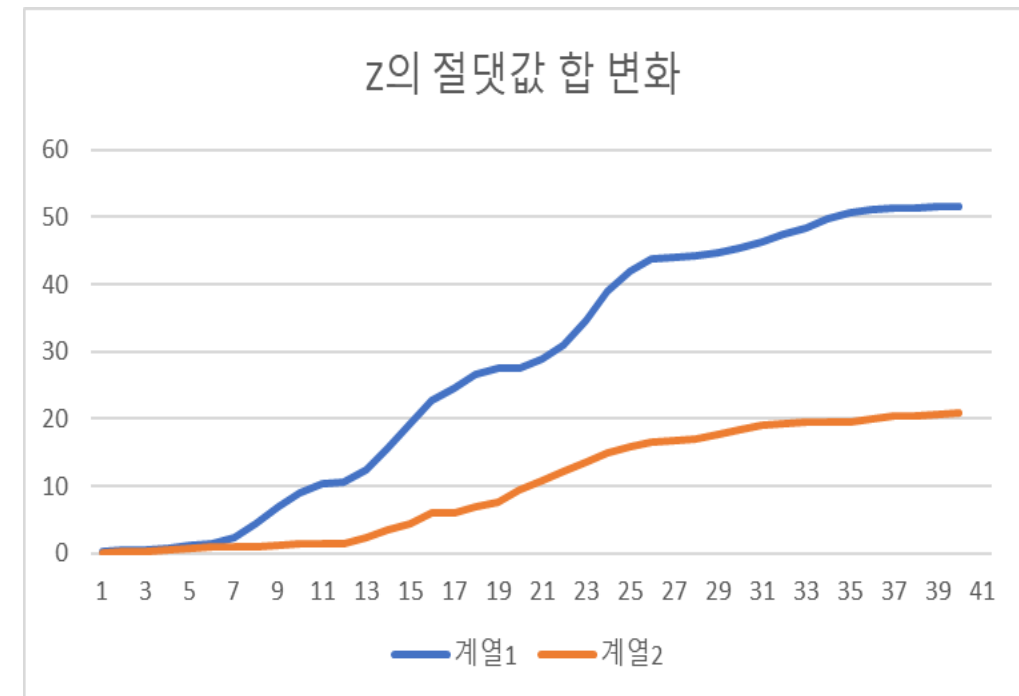
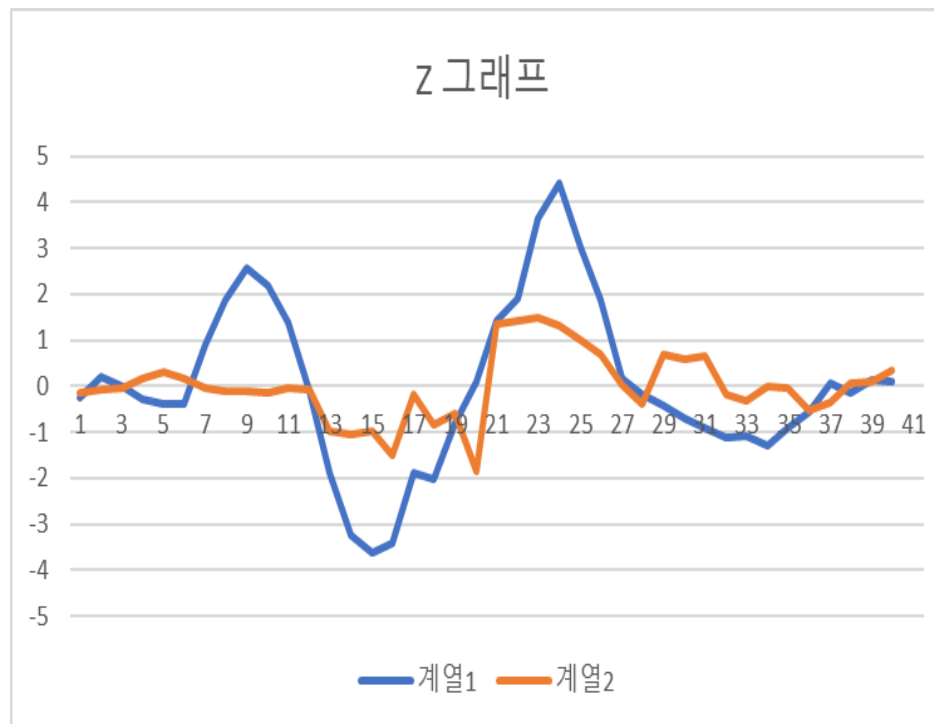
■ 실험 결과 3 – ‘S’ 계열 1: accel X, 계열 2: accel Z

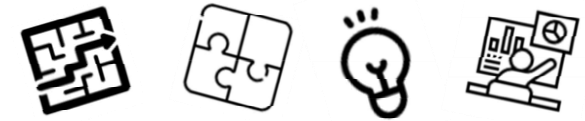


Lab



■ 실험 결과 4 – ‘Z’ (계열 1: accel X, 계열 2: accel Z)s





■ 실험 결과 분석 (1)

➤ 절대값 합을 그대로 쓴 이유는 평균을 구하는 것 보다 훨씬 값 차이가 뚜렷할 것이라고 생각했다.

<절대값 합>

←	I←		O←		S←		Z←	
1회←	4.37←	11.44←	39.82←	32.33←	45.05←	13.16←	57.52←	16.65←
2회←	6.46←	16.20←	43.19←	30.74←	42.79←	20.82←	61.44←	16.53←
3회←	4.48←	22.14←	34.73←	33.70←	41.45←	25.13←	57.96←	17.96←

<절대값 평균>

←	I←		O←		S←		Z←	
1회←	0.15←	0.39←	0.75←	0.61←	1.07←	0.31←	1.30←	0.37←
2회←	0.29←	0.73←	0.79←	0.56←	0.95←	0.46←	1.50←	0.40←
3회←	0.19←	0.92←	0.81←	0.78←	1.01←	0.61←	1.61←	0.50←



■ 실험 결과 분석 (2)

- 결과를 분석하면 I의 accel X의 절댓값 합은 다른 알파벳과 다르게 움직임이 거의 없고(0에 근접), accel Z의 움직임이 크므로 Z의 절댓값이 항상 크다. 따라서 $(X\text{의 절댓값 합}) < (Z\text{의 절댓값 합})$ 인지를 통해 I를 구분할 수 있다.
- O와 S는 X의 움직임과 Z의 움직임이 시간의 흐름만 다르고 비슷하게 흘러가는 모습이다. 따라서 X와 Z의 절댓값의 합의 최종 값이 유사한 모습을 보인다. 둘의 그래프 차이를 조금 더 살펴보거나 다른 방법을 이용해야 한다는 생각이 든다.
- Z의 움직임은 X축 움직임의 절댓값 합에서 I와 반대로 된 모습이다. 따라서 $(X\text{의 절댓값 합}) > (Z\text{의 절댓값 합})$ 인지를 통해 Z를 구분할 수 있다.

Lab -



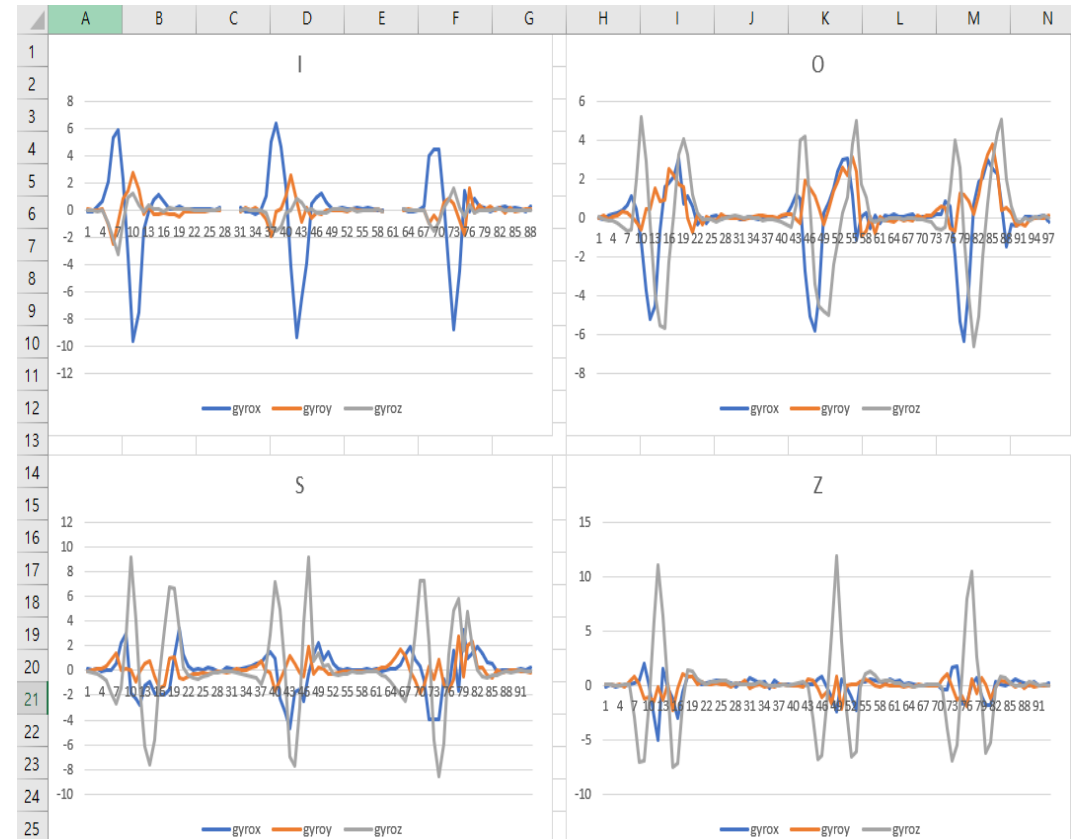
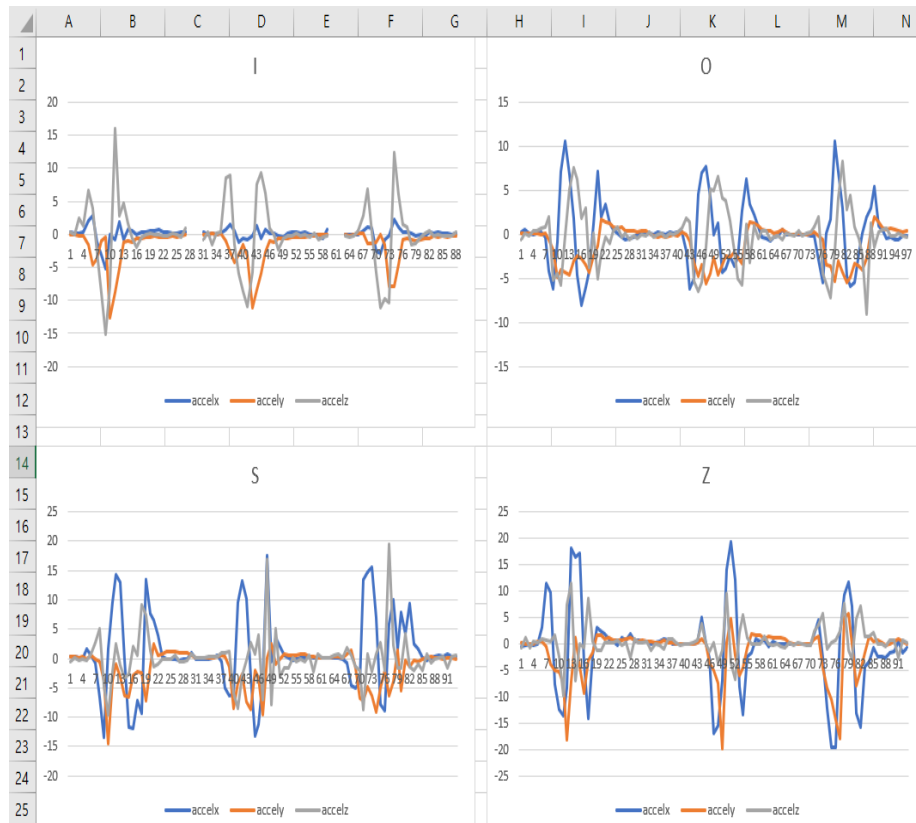
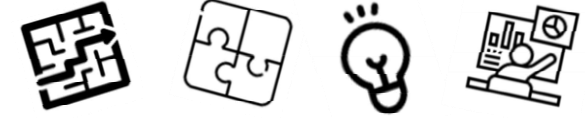
■ 실험 결과 (3)



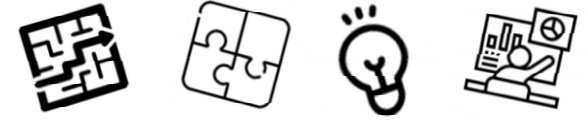
If문을 사용한 전문가 시스템 형식으로
구현 중, 현재 I를 걸러내는 부분까지
작성하였음.

```
if(max_i_accel==2 && max_i_gyro==3){  
    |    tv_motion.setText("동작 : I");  
}  
else if(){  
  
}
```

Lab – 참고 그래프



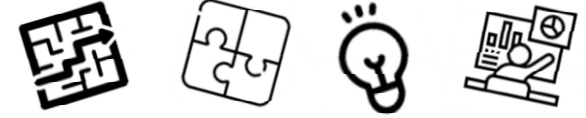
Future Work



■ Future Work

➤ 각 알파벳 별 실험 결과를 비교하여 차이점을 분석하고 알파벳을 인식해 본다.

History



Test environment

-컴퓨터 개발환경

Windows 버전

Windows 10 Home

© 2019 Microsoft Corporation. All rights reserved.

시스템

프로세서:	Intel(R) Pentium(R) CPU 4415U @ 2.30GHz 2.30 GHz
설치된 메모리(RAM):	4.00GB(3.86GB 사용 가능)
시스템 종류:	64비트 운영 체제, x64 기반 프로세서
펜 및 터치:	이 디스플레이에 사용할 수 있는 펜 또는 터치식 입력이 없습니다.

-안드로이드 스튜디오 버전 version 3.5.3

Minimum API level : API 15:Android 4.0.3(IceCreamSandWich)