

# 센서 융합을 이용한 스마트폰 자세 추정

컴퓨터공학부 4학년 구자경

## 차 례

1	들어가기 . . . . .	2
2	스마트폰 센서 . . . . .	3
2.1	가속도 센서 . . . . .	5
2.2	자이로스코프 센서 . . . . .	7
3	칼만 필터 . . . . .	8
3.1	칼만 필터 . . . . .	9
3.2	확장 칼만 필터(EKF) . . . . .	11
4	실험 . . . . .	12
5	맺음말 . . . . .	21
참고 문헌		21

## 1 들어가기

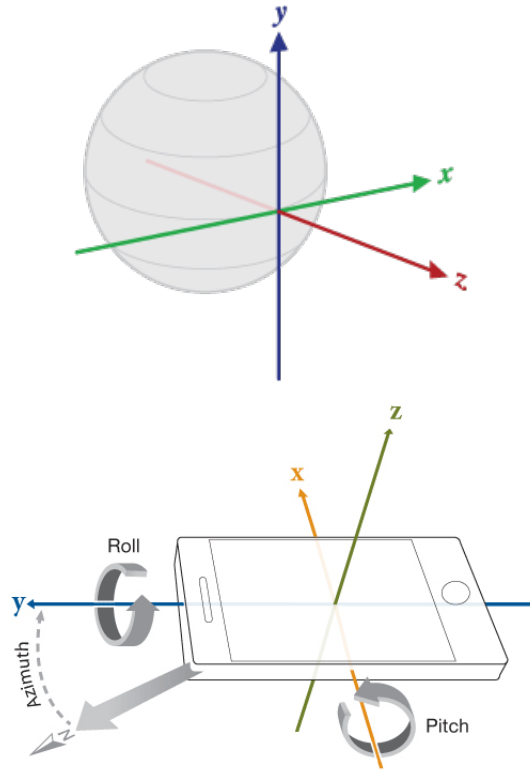


그림 1: (a) 기준 좌표계[1] (b) 센서 좌표계[2]

여기서 하고 싶은 것은 기준 좌표계를 안드로이드폰에 대한 센서 좌표계로 변환하는 회전 행렬을 구하는 것이다. 보통 좌표계 변환에는 다음 수식에서 보듯이 회전행렬과 이동행렬이 필요하다.

$$X_{world} = RX_{sensor} + T$$

회전행렬을 표현하는 방법에는 오일러각, 쿼터니언 등이 있으며 어떤 방법으로든 회전행렬을 구하면 스마트폰의 자세를 알 수 있다.

이를 위해 2장에서는 스마트폰의 센서를, 3장에서는 칼만 필터를 이용하여 자세를 추정하는 방법을, 4장에서는 실험 결과를 다룬다. 특히 칼만 필터를 처음 배우는 입장에서 어떻게 적용했는지에 대해 집중하였다.

기호가 너무 많고 헷갈림을 방지하기 위해 다음 표로 정리하였다. 수식에서

굵은 문자는 벡터를 나타낸다.

기호 및 용어	설명 혹은 동의어
$\phi$	x축을 기준으로 한 오일러각 (pitch)
$\theta$	y축을 기준으로 한 오일러각 (roll)
$\psi$	z축을 기준으로 한 오일러각 (yaw) (책에 따라 $\varphi$ 로 쓰기도 함)
기준 좌표계	세계 좌표계, 관성 좌표계, 항법 좌표계
센서 좌표계	스마트폰 좌표계, 동체 좌표계, 모델 좌표계
$\mathbf{F}_a$	가속도 센서의 출력값 : $F_{ax}, F_{ay}, F_{az}$
$\mathbf{F}_g$	자이로 센서의 출력값 : $F_{gx}, F_{gy}, F_{gz}$
$R$	점의 변환을 나타내는 회전 행렬
$R'$	좌표축의 변환을 나타내는 회전 행렬

또한 기존에 항공학에서 쓰이던 관례와 스마트폰에서 사용하는 좌표축에 대해 정리하면 다음과 같다. 특히 기존에 항공학에서 정리된 문서들을 참고할 때 유용할 것이라고 생각한다.

	항공기	스마트폰
월드 좌표계 (EastNorthUp)	x축:동쪽, y축:북극(true north) z축:지구중력과 반대 방향	x축:y × z 방향, y축:자북 z축:지구중력과 반대 방향
모델 좌표계	x축:비행기가 나아가는 방향(종축) y축:z × x 방향, z축:수직축	x축:스크린 오른쪽, y축:스크린 위쪽 z축:스크린을 뚫고 나오는 방향
pitch $\phi$	y축 회전(-90도에서 90도)	x축 회전(-180도에서 180도)
roll $\theta$	x축 회전(-180도에서 180도)	y축 회전(-90도에서 90도)
yaw $\psi$	z축 회전(0도에서 360도)	z축 회전(0도에서 360도)

한 가지 중요한 사실은 회전각 중에서 한 개는 총 180도의 범위를 가져야 한다. 예를 들어 핸드폰을 책상에 둔 채로 z축, y축, x축 방향으로 180도씩 회전하면 원래대로 돌아옴을 알 수 있다. 각각 축에 대해 180도씩 변화했음에도 말이다. 그렇다면 가장 회전이 적은 축을 선택하면 될텐데 스마트폰에서는 x축보다는 y축 회전이 상대적으로 적은 특성을 이용한 것으로 보인다.

## 2 스마트폰 센서

반도체 기술의 발달로 미소한 요소를 만들고, 그 기계적 동작을 검출하는 MEMS 기술이 진행되었고, 마침내 스마트폰에도 탑재되었다. 특히 아이폰4에서 자이로

스코프 센서를 탑재한 이후로 대부분의 스마트폰에서 가속도 센서, 자지가 센서, 자이로스코프 센서가 장착되었다. 가속도 센서는 속도의 변화량을, 자지가 센서는 방위각의 산출을, 자이로스코프는 각도의 변화량을 측정한다. 이러한 센서로 이루어진 장치를 관성측정장치(Inertial Measurement Unit, IMU)라 하는데, 특히 군사, 해양, 항공 분야 등 다양한 분야에서 연구가 많이 진행되어 왔다.

**관성항법과 복합항법** 항법(Navigation)은 우리가 일상생활에서 쓰듯이 출발점에서 도착점까지 이동하는 기술, 계획을 뜻한다. 관성항법이란 관성계에서 나타나는 물리현상에 기반을 둔 항법이다. 위에서 말한 IMU를 이용한 항법을 관성항법이라 한다. 우리에게 친숙한 GPS장치와 관성 항법을 비교하면 다음과 같다.

	장점	단점
관성 항법	외부 교란에 영향 안 받음	오차 누적, 센서의 성능 중요
GPS 장치	절대적인 위치 측정	전파방해(Jamming)에 취약

관성항법에 의해 누적되는 오차를 GPS 장치를 이용해 중간 중간에 보정함으로써 서로의 장점을 얻을 수 있다. 또한 전파 수신에 이상이 있는 경우에 관성 항법 장치로 보완할 수 있다. 관성항법에 비관성 방법을 결합한 것을 복합항법이라 한다.

**스마트폰에 탑재된 관성측정장치** 스마트폰 센서 탑재 중 가장 유의미한 사건 중 하나는 아이폰4에서 자이로스코프를 장착한 일이다. 대부분 이렇게 빨리 스마트폰에 장착될지 예상하지 못했고, 이로서 저가의 관성항법장치 구현이 가능해졌다. 또한 GPS 센서를 이용하여 복합항법장도 구현할 수 있게 되었다. 스마트폰에 추가된 센서 중 디지털 기압계가 제공하는 고도 정보를 이용하여 GPS의 수신율을 높이기도 한다. 본 문서에서는 관성측정장치에 집중할 예정이고 앞으로 살펴볼 센서를 비교하면 다음과 같다.

자세 측정	장점	단점
자이로 센서	빠른 반응 속도, 비교적 정확한 측정	드리프트 효과, 기준 방향 필요
가속도 센서	기준 방향 필요 없음, 드리프트 효과 없음	부정확, $\psi$ 각도 잴 수 없음

한 가지 특이한 점은 관성 항법으로 상대적인 위치를 알 수 있듯이 자이로 센서를 이용해 상대적인 자세를 알 수 있고, GPS 장치를 이용해 절대적인 위치를 알 수 있듯이 가속도 센서를 이용해 절대적인 자세를 알 수 있다. 이제 가속도 센서와 자이로 센서를 정량적으로 알아 보자.

## 2.1 가속도 센서

가속도 센서가 측정하는 것은 중력 가속도와 선형 가속도이다. 예를 들어 엘리베이터가 상승하면 선형 가속도가 더해져 몸무게가 늘어난 기분이 들고, 엘리베이터가 하강하면 선형 가속도에 의해 몸이 가벼워진 느낌이 든다. 중력 가속도는 기기의 움직임이 거의 없는 상태에서 기울기(자세) 정보를 제공하고, 선형 가속도는 스마트폰이 움직일 때 발생하는 속도의 변화량(가속도)을 측정한다. 그러나 고가의 항법 센서가 아니면 선형 가속도를 측정하기가 힘들다.

그림 1을 보면 안드로이드폰에 대한 세 축이 나와 있고, 각각의 축에 대하여 가속도를 측정한다. 이를 수식으로 나타내면 다음과 같다.

$$\begin{bmatrix} F_{ax} \\ F_{ay} \\ F_{az} \end{bmatrix} = \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} + \begin{bmatrix} 0 & w & -v \\ -w & 0 & u \\ v & -u & 0 \end{bmatrix} \begin{bmatrix} F_{gx} \\ F_{gy} \\ F_{gz} \end{bmatrix} + g \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix}$$

(움직이는 물체의 가속도)

$u, v, w$ 는 이동 속도를,  $F_{gi}$ 는 자이로스코프 센서로 측정한 회전 각속도를,  $g$ 는 중력 가속도를 나타낸다. 위에서 언급했듯이 고가의 항법센서가 아니면  $u, v, w$ 와  $\dot{u}, \dot{v}, \dot{w}$ 를 측정할 수 없다. 그래서 시스템이 정지해 있거나 등속 운동한다는 조건을 가정하면 다음 식을 얻는다. (근사식이라는 사실이 중요하다)

$$\begin{bmatrix} F_{ax} \\ F_{ay} \\ F_{az} \end{bmatrix} = g \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix} \quad (\text{정지 혹은 등속운동 하는 물체의 가속도})$$

한 가지 흥미로운 사실은 (정지 혹은 등속운동 하는 물체의 가속도)에는  $\psi$  성분이 없다. 즉 가속도 센서로는 yaw각을 측정할 수 없다는 말인데, 책상 위에 핸드폰을 둔 상태에서 책상 위에 붙인 상태로 돌리면 가속도 센서는 똑같은 값을 측정한다.

그런데 이 식을 다른 방법으로 유도해 보자. 각 축에 대해 회전하는 행렬을 생각하자. 예를 들어  $z$ 축에 대한 회전은 우리에게 친숙한 2차원 상에서 회전 행렬과 비슷하다.

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}, R_y(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, R_z(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

그런데 이는 점 혹은 위치벡터를 변환 시켜주는 행렬이고, 좌표계를 변환하는 행렬은 음의 각도를 나타내면 된다. 좌표계 변환 회전 행렬은 다음과 같이 나타내고, 이 때 미분 기호와는 상관 없다.

$$R'_x(\phi) = R_x(-\phi), R'_y(\theta) = R_y(-\theta), R'_z(\psi) = R_z(-\psi)$$

기준 좌표계의 경우 선형가속도  $\mathbf{a}$ 를 고려하지 않았을 때  $\mathbf{g}$ 만큼의 중력 가속도를 느낀다. 그렇다면 회전 변환 후 센서 좌표계에서 느끼는 중력 가속도는 다음과 같다.

$$\mathbf{F}_a = \begin{bmatrix} F_{ax} \\ F_{ay} \\ F_{az} \end{bmatrix} = R'(\mathbf{g} - \mathbf{a}) = R'\mathbf{g} = R' \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (1)$$

이때  $R' = R'_x R'_y R'_z$ 이고, 이 식을 이용하면,

$$\mathbf{F}_a = R'\mathbf{g} = \begin{bmatrix} \dots & \dots & -\sin \theta \\ \dots & \dots & \sin \phi \cos \theta \\ \dots & \dots & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} -g \sin \theta \\ g \cos \theta \sin \phi \\ g \cos \theta \cos \phi \end{bmatrix}$$

를 얻는다. 이는 (정지 혹은 등속운동 하는 물체의 가속도)과 같다.

**가속도 센서로  $\phi, \theta$  추정하기** (1) 식을 변형하면,

$$R_y R_x \mathbf{F}_a = R_y'^{-1} R_x'^{-1} \mathbf{F}_a = R_z' \mathbf{g} = \mathbf{g}$$

이 식을 정리하면,

$$\begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} F_{ax} \\ F_{ay} \\ F_{az} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$$

따라서 다음 관계식을 얻는다.

$$\begin{cases} F_{ay} \cos \phi - F_{az} \sin \phi = 0 \\ F_{ax} \cos \theta + F_{ay} \sin \theta \sin \phi + F_{az} \sin \theta \cos \phi = 0 \end{cases}$$

이를 오일러 각에 대해 정리하면 다음과 같고, 이는 (정지 혹은 등속운동 하는 물체의 가속도) 식에서도 유도할 수 있다.

$$\begin{cases} \phi = \text{atan2}(F_{ay}/F_{az}) & -180^\circ \leq \phi \leq 180^\circ \\ \theta = \text{atan}(-F_{ax}/(\sqrt{F_{ay}^2 + F_{az}^2})) & -90^\circ \leq \theta \leq 90^\circ \end{cases}$$

회전을 오일러 각으로 표현했을 때 단점 중 하나는 세 각 중에 하나의 범위를 -90도에서 90도 사이로 해야 한다는 것이다(singularity). 1장에서 보았듯이  $\theta$ 는 -90도에서 90도 범위이므로 atan함수를 이용하고,  $\phi$ 는 -180도에서 180도 범위를 나타낼 수 있는 atan2 함수를 이용한다.

## 2.2 자이로스코프 센서

자이로스코프의 역사와 스마트폰에 장착된 배경에 대해 정리하면 다음과 같다.

1852년	푸코가 처음으로 '커다란 각 운동량을 가진 장치'로 정의
1910년대	독일과 미국에서 나침반 용도로 실용화
1970년대	미소한 요소를 만들고, 그 기계적 동작을 검출하는 기술(MEMS) 연구
1990년대	MEMS기술이 시장에 진출하기 시작 [5]
2010년 6월 24일	아이폰4 부터 자이로스코프 센서 탑재
2010년 12월 6일	안드로이드 2.3 (API 9) 부터 자이로스코프 센서 탑재

(움직이는 물체의 가속도)와 (정지 혹은 등속운동 하는 물체의 가속도) 수식을 다시 보자. 정지 혹은 등속운동을 가정한 상태로 가속도 센서의 출력값을 결정했지만, 이 때 자이로스코프 센서의 출력값을 이용하지 않는다. 그렇다면 자이로스코프 센서와 가속도 센서를 서로 보완할 수 있음을 시사한다.

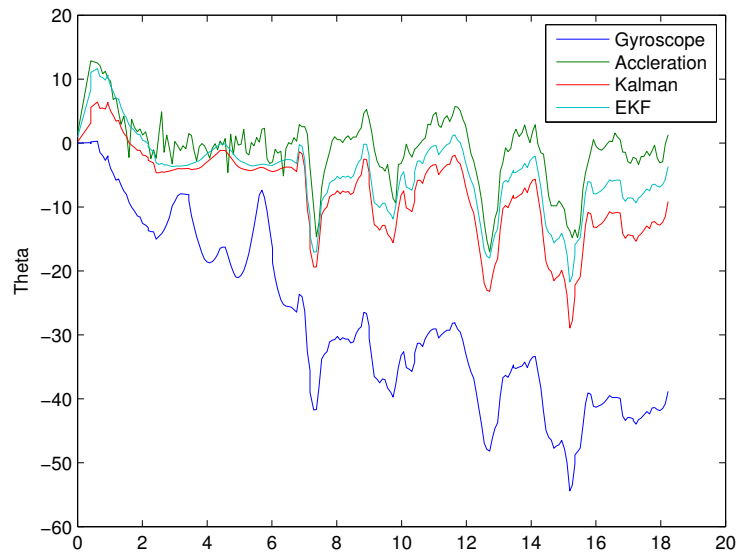
동역학 분야에서 오일러각과 각속도의 관계는 이미 잘 알려져 있다.  $\mathbf{F}_g$ 는 자이로스코프 센서로 측정한 각속도이다.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \begin{bmatrix} F_{gx} \\ F_{gy} \\ F_{gz} \end{bmatrix}$$

자이로스코프가 측정한 값을 위 식에 대입한 후 적분하면 초기 자세에서 변화된 자세 정보를 추정할 수 있다. 하지만 각속도를 적분하는 과정에서 어쩔 수 없이 발생하는 드리프트 오차로 인해 짧은 시간 동안에만 유효한 값을 갖는 단점이 있다.



위 그림과 같이 스마트폰의 x축을 기준으로 Pitch각(혹은  $\Phi$ 각)만 변화시켰을 때, 가속도 센서와 자이로 센서를 이용하여 얻은 결과는 다음과 같다.



시간이 지남에 따라 자이로 센서는 드리프트 효과 때문에 오차가 누적되지만 가속도 센서는 그렇지 않다. 따라서 두 센서의 장단점을 융합하면 더 좋은 결과를 얻을 수 있다.

### 3 칼만 필터

칼만 필터로 들어 가기 전에 간단한 필터들과 비교해 보자.  $\hat{x}_k$ 는 추정값을,  $z_k$ 는 측정값을,  $\hat{x}_k^-$ 는 칼만필터에서의 예측값을 나타낸다.



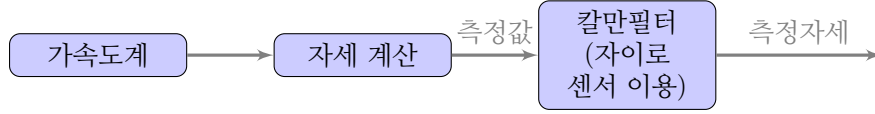


그림 2: 센서 융합

평균의 배치식	$\bar{x}_k = \frac{x_1 + x_2 + \dots + x_k}{k}$	실시간으로 데이터를 처리하기는 비효율적
평균의 재귀식	$\bar{x}_k = \frac{k-1}{k}\bar{x}_{k-1} + \frac{1}{k}x_k$	측정량이 시간에 따라 변하면 비효율적
이동 평균 필터	$\bar{x}_k = \bar{x}_{k-1} + \frac{x_k - x_{k-n}}{n}$	오래된 측정값과 최신 측정값에 같은 가중치
1차 저주파 통과 필터	$\bar{x}_k = \alpha\bar{x}_{k-1} + (1-\alpha)z_k$	최근 측정값에 더 높은 가중치 부여
칼만 필터 ( $H = I$ 인 경우)	$\hat{x}_k = (1 - K_k)\hat{x}_k^- + K_k z_k$ $\hat{x}_k^- = A\hat{x}_{k-1}$	가중치와 추정값의 오차 공분산을 매번 계산 예측 과정 추가

칼만 필터를 이용해서 센서 융합을 통해 스마트폰의 자세를 측정해 보자.

### 3.1 칼만 필터

칼만 필터에 대한 자세한 설명은 [4]을 참고하고, 여기서는 요약해서 살펴 보자.

1. 초기값 설정  $\hat{x}_0, P_0$ .
2. 추정값과 오차 공분산 예측  $P$

$$\hat{x}_k^- = A\hat{x}_{k-1}$$

$$\hat{P}_k^- = AP_{k-1}A^T + Q$$

3. 칼만 이득 계산

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1}$$

4. 추정값 계산

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$$

5. 오차 공분산 계산 후 2. 로 돌아감.

$$P_k = P_k^- - K_k H P_k^-$$

칼만 필터의 입력과 출력은 다음과 같다.

입력	$z_k$
출력	$\hat{x}_k$
시스템 모델	$A, H, Q, R$
내부 계산용	$\hat{x}_k^-, P_k^-, P_k, K_k$

자이로와 가속도계의 상호 보완적인 특성을 이용해 가속도계로 자이로를 보상하는 센서 융합을 하려고 한다. 먼저 상태 변수를 어떻게 설정할 지가 관건이다. 만약 오일러각을 상태 변수로 잡는다면 다음 식을 이용해야 한다.

$$\begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$$

그런데 이 식으로부터 바로 칼만 필터를 이용할 수 없다. 그래서 쿼터니언을 고려한 다음 시스템 모델을 사용해 보자.

$$x_{k+1} = Ax_k + w_k$$

$$z_k = Hx_k + \nu_k$$

$Q$ : covariance matrix for  $w_k$

$R$ : covariance matrix for  $\nu_k$

쿼터니언과 각속도의 관계 각속도:(p,q,r)

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}$$

위 식을 이산 시스템으로 바꾸면,

$$\underbrace{\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}}_{x_{k+1}} = \underbrace{\left( I + \Delta t \cdot \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \right)}_A \underbrace{\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}}_{x_k}$$

$$H = I, \quad Q = 0.001I, \quad R = 10I$$

$$A = \left( I + \Delta t \cdot \frac{1}{2} \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \right)$$

상태변수를 오일러각에서 쿼터니언으로 바꾸고, 측정값도 오일러각에서 쿼터니언으로 바꿨다. 그랬더니 시스템 모델이 칼만 필터를 쓸 수 있는 형태가 되었다.

### 3.2 확장 칼만 필터(EKF)

확장 칼만 필터는 선형 시스템에만 적용할 수 있는 칼만 필터를 확장해서 비선형 시스템에도 적용할 수 있다.  $f(x)$ 의 자코비안을 구함으로써 비선형 모델을 선형 모델로 근사한다.

$$x_{k+1} = f(x_k) + w_k$$

$$z_k = h(x_k) + \nu_k$$

$Q$ : covariance matrix for  $w_k$

$R$ : covariance matrix for  $\nu_k$

$$A = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}_k}, \quad H = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}_k^-}$$

$$\begin{aligned}
\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} &= \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \\
&= \begin{bmatrix} p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ q \cos \phi - r \sin \phi \\ q \sin \phi \sec \theta + r \cos \phi \sec \theta \end{bmatrix} \\
&= f(x) + w
\end{aligned}$$

$$z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} + \nu = Hx + \nu$$

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial \phi} & \frac{\partial f_1}{\partial \theta} & \frac{\partial f_1}{\partial \psi} \\ \frac{\partial f_2}{\partial \phi} & \frac{\partial f_2}{\partial \theta} & \frac{\partial f_2}{\partial \psi} \\ \frac{\partial f_3}{\partial \phi} & \frac{\partial f_3}{\partial \theta} & \frac{\partial f_3}{\partial \psi} \end{bmatrix}$$

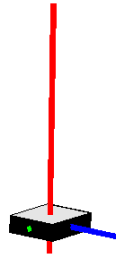
#### 4 실험

쿼터니언을 이용해 스마트폰의 자세를 시각화 하는 것은 [3]의 코드를 이용했다. 또한 스마트폰의 센서 정보는 Android Sensor Support from MATLAB [2]을 이용했다. 책상을 스마트폰에 올려 놓고, z축을 향해서만 회전한 다음 x축에 대해 회전시켰다. 이를 시각화한 것은 다음 그림과 같다. 순서대로 가속도 센서, 칼만 필터, 확장 칼만 필터로 측정한 스마트폰 자세를 보여 준다. 가속도 센서만 가지고는 z축에 대한 회전각(Psi각)을 알 수 없음을 알 수 있다.

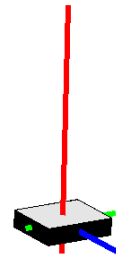
Accel



Kalman(Accel+Gyro)



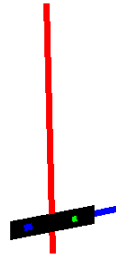
EKF



Accel



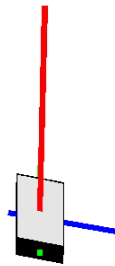
Kalman(Accel+Gyro)



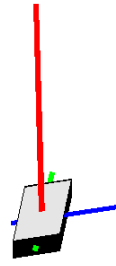
EKF



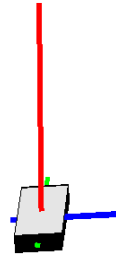
Accel



Kalman(Accel+Gyro)



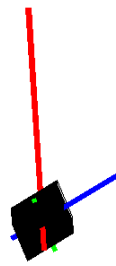
EKF



Accel



Kalman(Accel+Gyro)



EKF

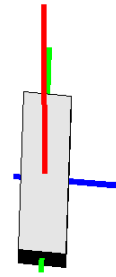
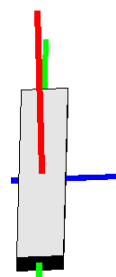
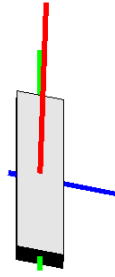


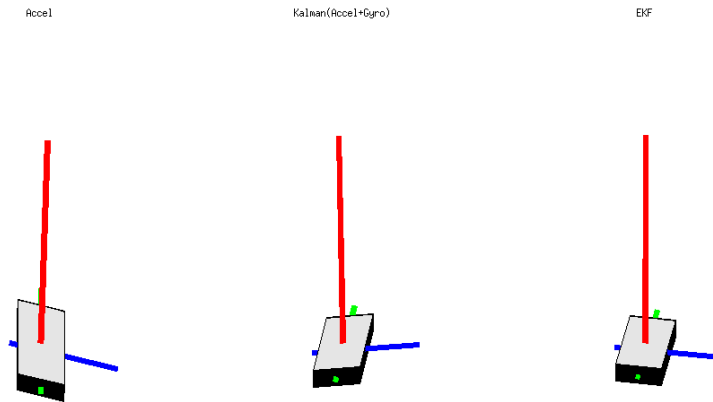


Accel

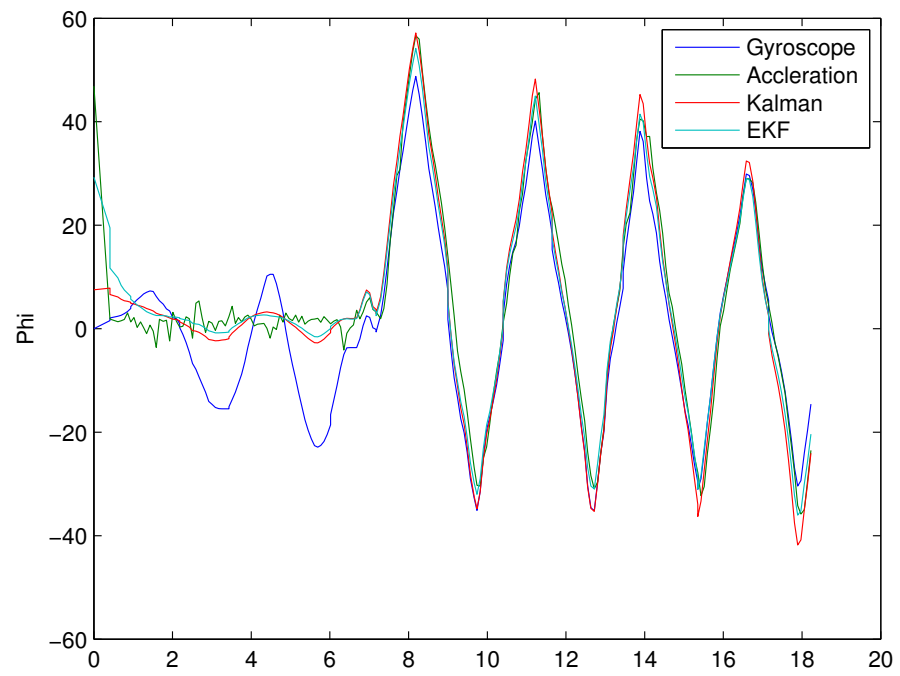
Kalman(Accel+Gyro)

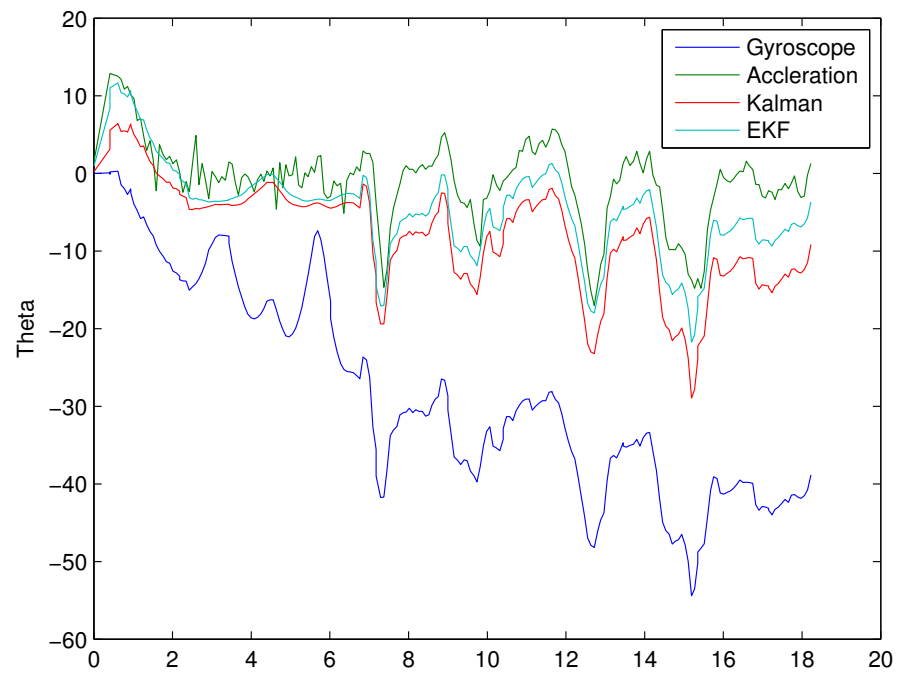
EKF

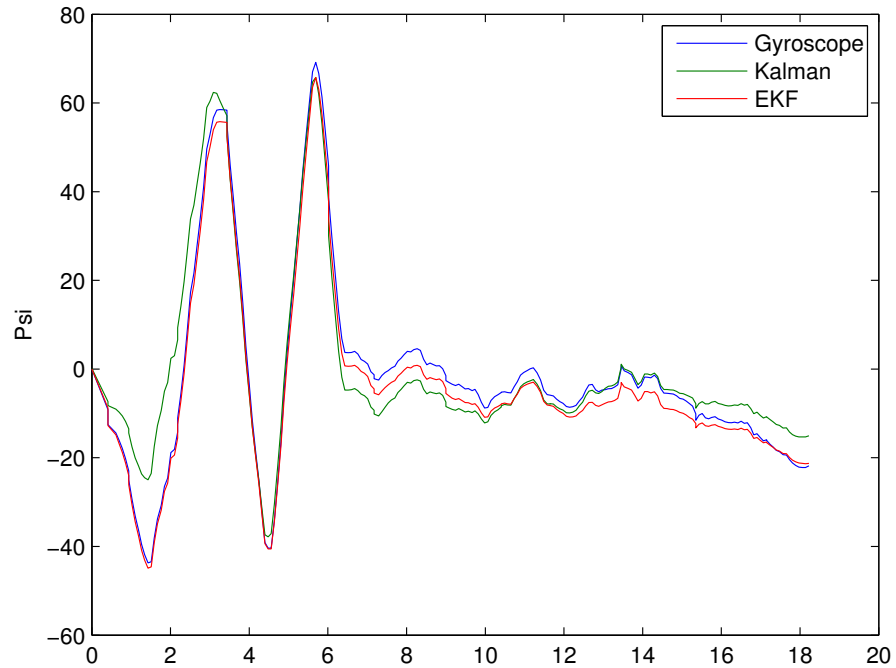




이에 대응하여 시간에 따라 측정한 오일러 값은 다음과 같다.







## 5 맺음말

오일러각이나 쿼터니언이나 그 값을 보고, 직관적으로 어떻게 회전하는지 알기가 어렵다. 그래서 지수행렬(exponential map)을 이용해 코드를 다시 작성해볼 예정이다. 또한 기존의 코드에는 많은 애러가 있다. 가령 한 방향으로 계속 회전 시키면 정확한 값을 내지 못한다.

지금 공부한 것을 바탕으로 한 단계 한 단계 배우면서 석사 때 계속 이어나갈 계획이다.

## 참고 문헌

- [1] Android developer site. <http://developer.android.com>.
- [2] Android Sensor Support from MATLAB. <http://kr.mathworks.com/hardware-support/android-sensor.html>.

- [3] Gustaf Hendeby, Fredrik Gustafsson, and Niklas Wahlström. Teaching sensor fusion and KalmanFiltering using a smartphone. In *Proceedings of the 19th World Congress of the International Federation of Automatic Control (IFAC)*, 2014.
- [4] 김성필. 칼만필터의 이해(*MATLAB*활용). 아진, 2010.
- [5] 박장선. Mems기술 및 정책동향. Technical report, 2006.