Motion Tracking using IMU Sensors

Hua Huang

In this lecture

- Introduce the basics of motion sensing using IMUs
- A3: an advanced orientation tracking method
- Armtrak: an advanced position tracking method

Position tracking using accelerometer

$$A_{measured} = g + a_{accl}$$

$$a_{accl} = A_{meas} - g$$

Need to subtract gravity to obtain acceleration due to motion

$$\frac{d^2x}{dt^2} = \mathrm{A_{meas}} - \mathrm{g}$$

$$\frac{d^2x}{dt^2} = \mathrm{A_{meas}} - \mathrm{g} + \boxed{\mathrm{n}}$$

Hardware noise

$$\frac{dx}{dt} = \int (A - g + n)\, dt = \int (A - g) dt + \int n\, dt$$

$$x = \iint_0^t (A - g) dt + \boxed{\iint_0^t n\, dt}$$

Error accumulates dramatically with time

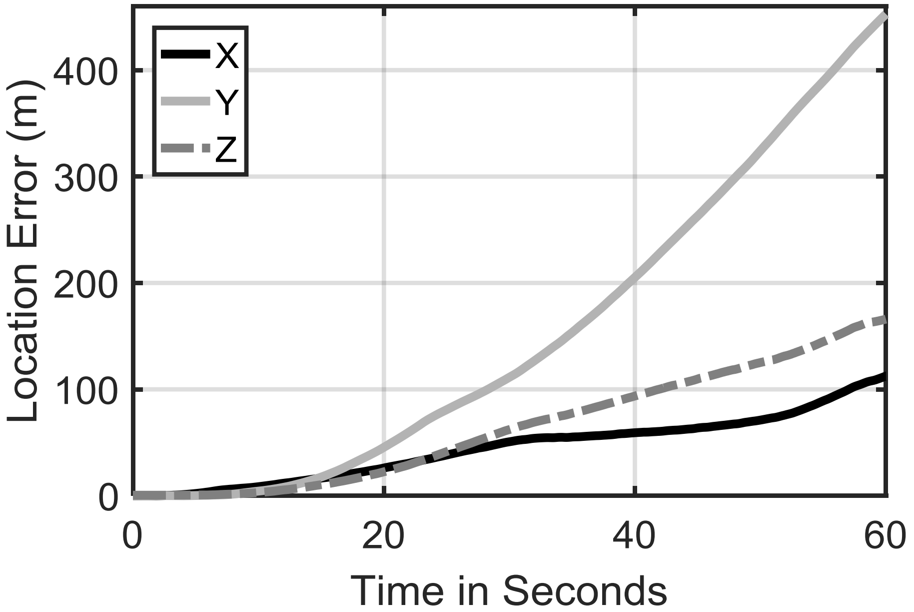
Drifting errors: a fundamental limitation

$$x = \iint_0^t (A - g)dt + \boxed{\iint_0^t n\, dt}$$

Error accumulates dramatically with time

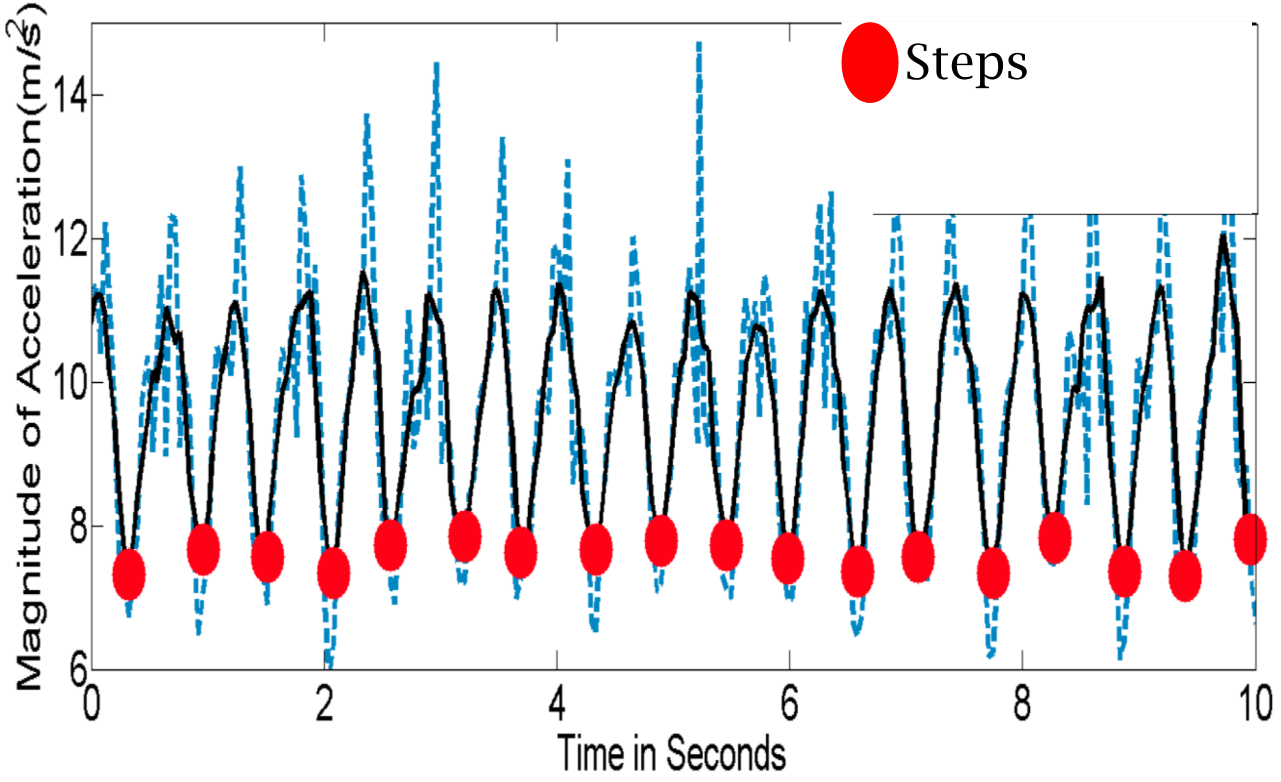
How large is this error?

Experiment: hold the phone in the hand, estimate its position



Dead Reckoning: Accelerometer to measure distance

- Double integration fails dramatically $\cancel{x} = \iint_0^t (A - g)dt + \iint_0^t n\,dt$
- However, accelerometer is good in tracking steps



Distance = step_count * step_size

Combining distance estimates with compass directions, we can dead reckon

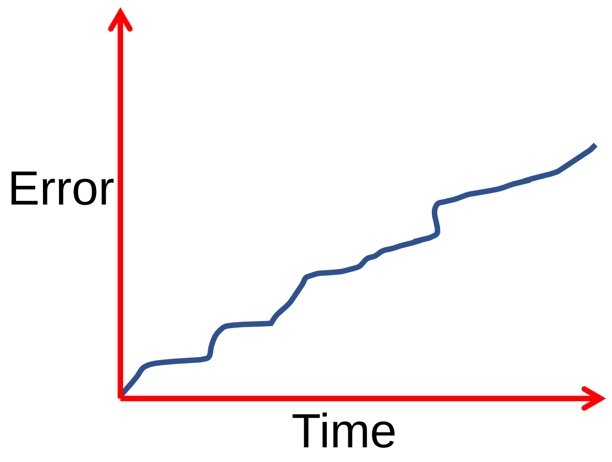
Orientation tracking using gyroscope

$$gyro = \omega$$

$$\frac{d\theta}{dt} = \omega$$

$$\frac{d\theta}{dt} = \omega + noise$$

$$\theta = \int_0^t \omega \,.dt + \boxed{\int_0^t noise\,.dt}$$



Error accumulates over time

What are the major factors influencing the performance of smartphone gyroscopes?

Parameter	**Sensor Settings/Conditions**	**Typ**	**Unit**
Output noise	$r = \pm 320°/\text{sec}$, $b = 330\text{Hz}$	0.9	°/sec (rms)
	$r = \pm 320°/\text{sec}$, $b = 50\text{Hz}$	0.4	°/sec (rms)
	$r = \pm 160°/\text{sec}$, $b = 50\text{Hz}$	0.2	°/sec (rms)
	$r = \pm 80°/\text{sec}$, $b = 50\text{Hz}$	0.1	°/sec (rms)
Temperature coefficient	ADIS 1626x	0.005	°/sec/°C
Linear acceleration	Any axis	0.2	°/sec/g

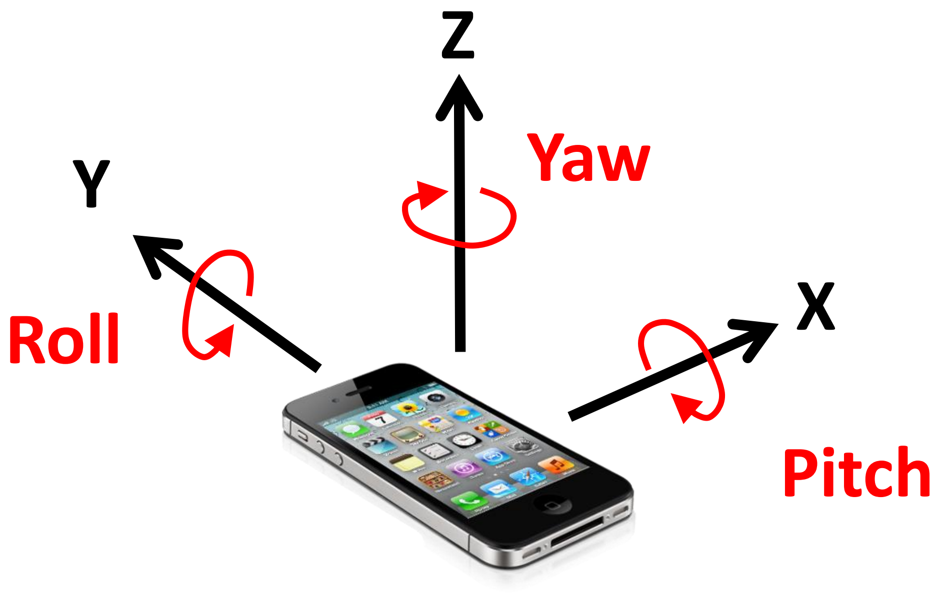
MEMS sensors: Widely blamed for its poor accuracy

Papers / Apps	Error / Statements
UnLoc @ MobiSys'12	"Error accumulates over time", 20 meters within 3 mins
Walkie-Markie @ NSDI'13	"Rapid error accumulation as distance increases"
Sensing Vehicle Dynamics@ MobiSys'13	"Gyro sensor readings can be noisy and unreliable"
Characterization study @ TCMS	Temperature and humidity affects MEMS gyroscope
Gyrophone @ USENIX Security'14	"Susceptible to ambient acoustic noises"
Sensor Box @ Android	65° within 3 mins
Seene @ iPhone	30° within 3 mins

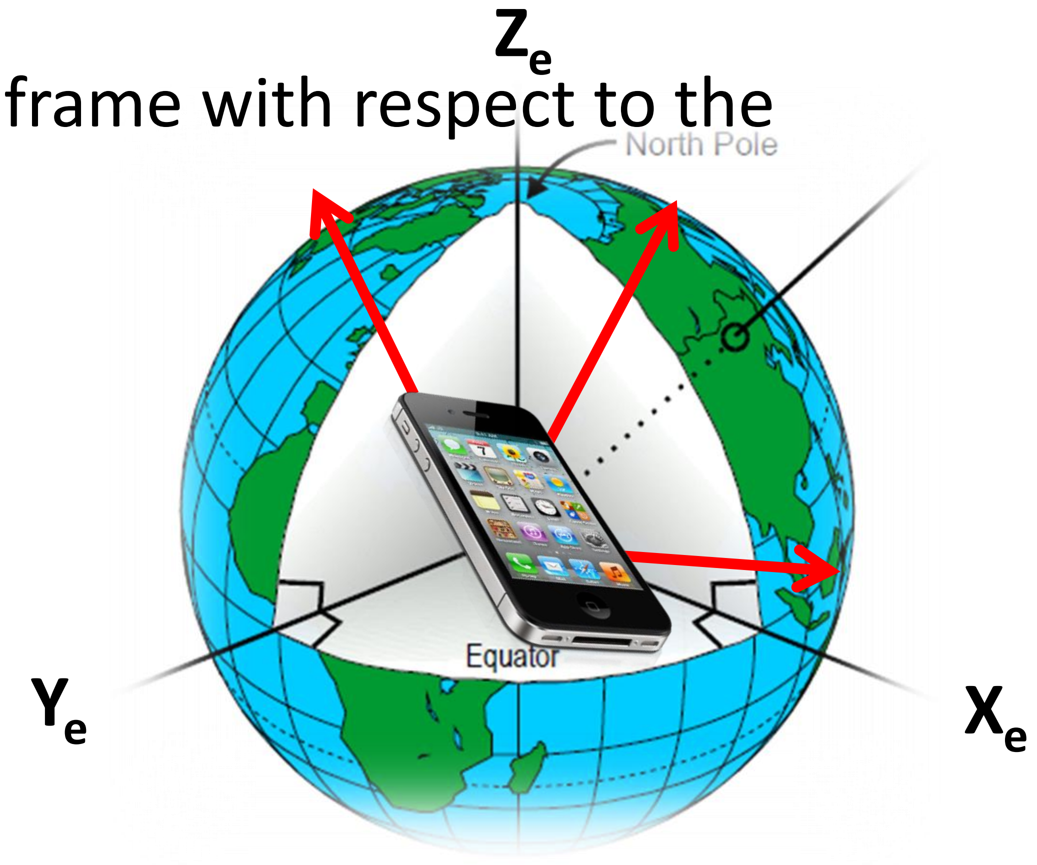
Use it Free: Instantly Knowing Your Phone Attitude (a.k.a: orientation)

What is phone attitude (orientation)?

- 3D misalignment of the phone's local frame with respect to the Global-frame (earth)



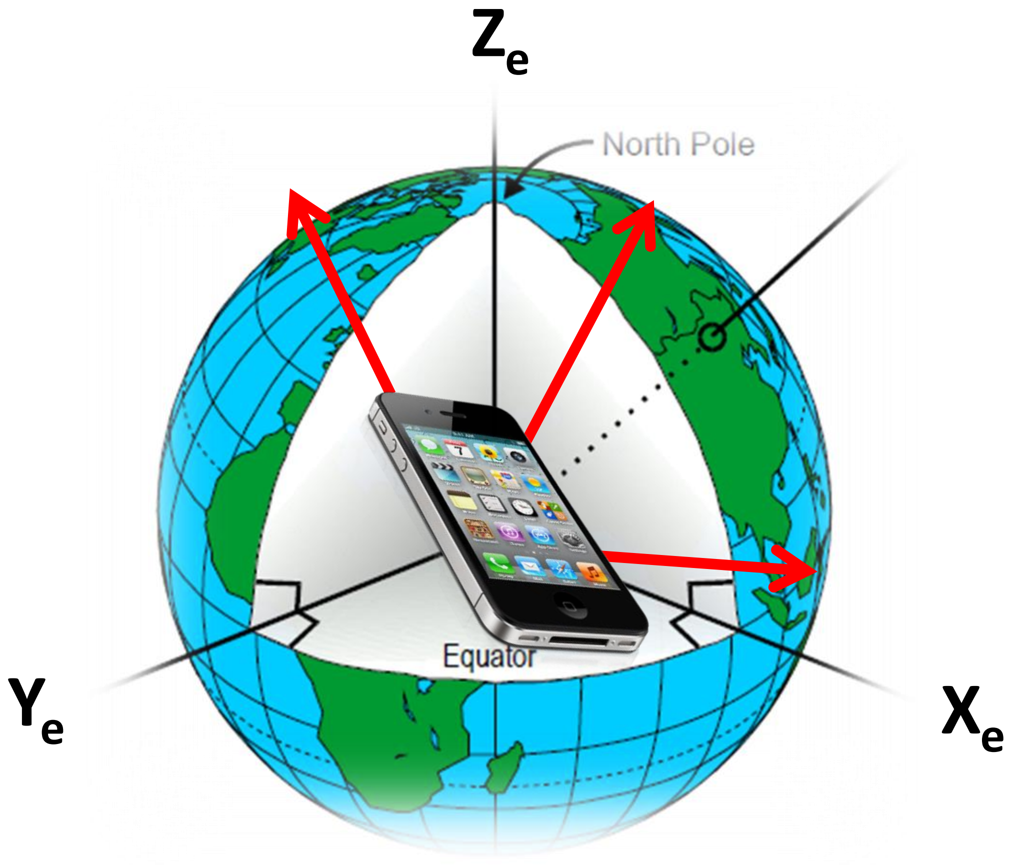
Local-frame



Global-frame

What is phone attitude (orientation)?

- Misalignment between two frames
  - Rotation Matrix **R**
  - Yaw, Pitch, Roll (Euler Angles)
- 3 degrees of freedom



Geo-frame

Why phone orientation is important?

Dead-reckoning based localization

3-D photography

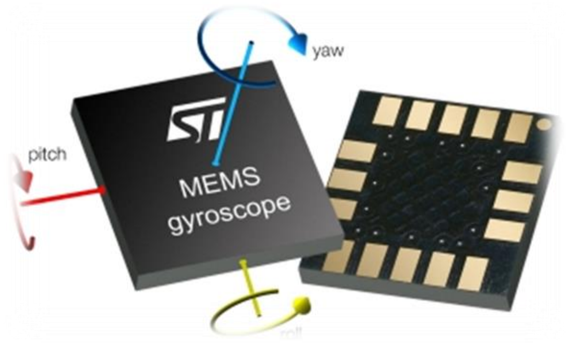
Fine-grained gesture recognition

Mobile gaming

Basic approaches for orientation tracking

How to estimate attitude?

- Inertial Measurement Unit (IMU) sensors



MEMS Gyroscope

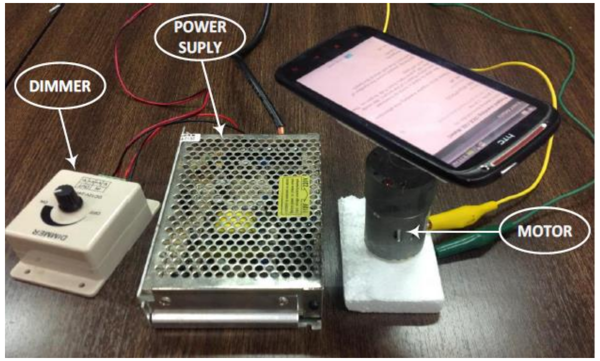
Accelerometer

Compass

Gyroscope experiment

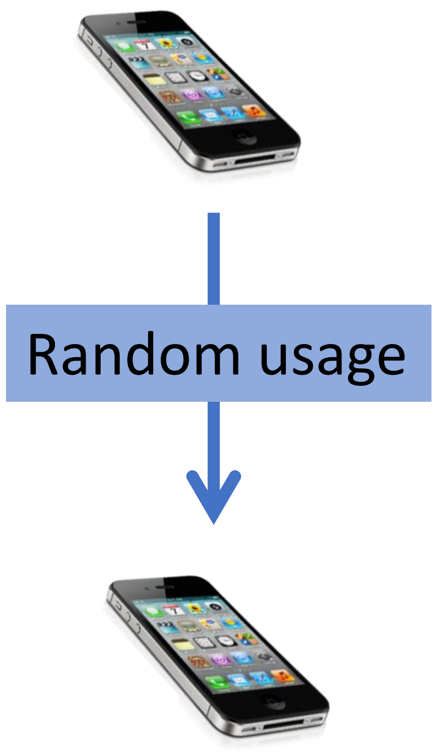
- Error Sources
  - Time integration
  - Temperature -- single point compensation has already been done
  - Phone motion

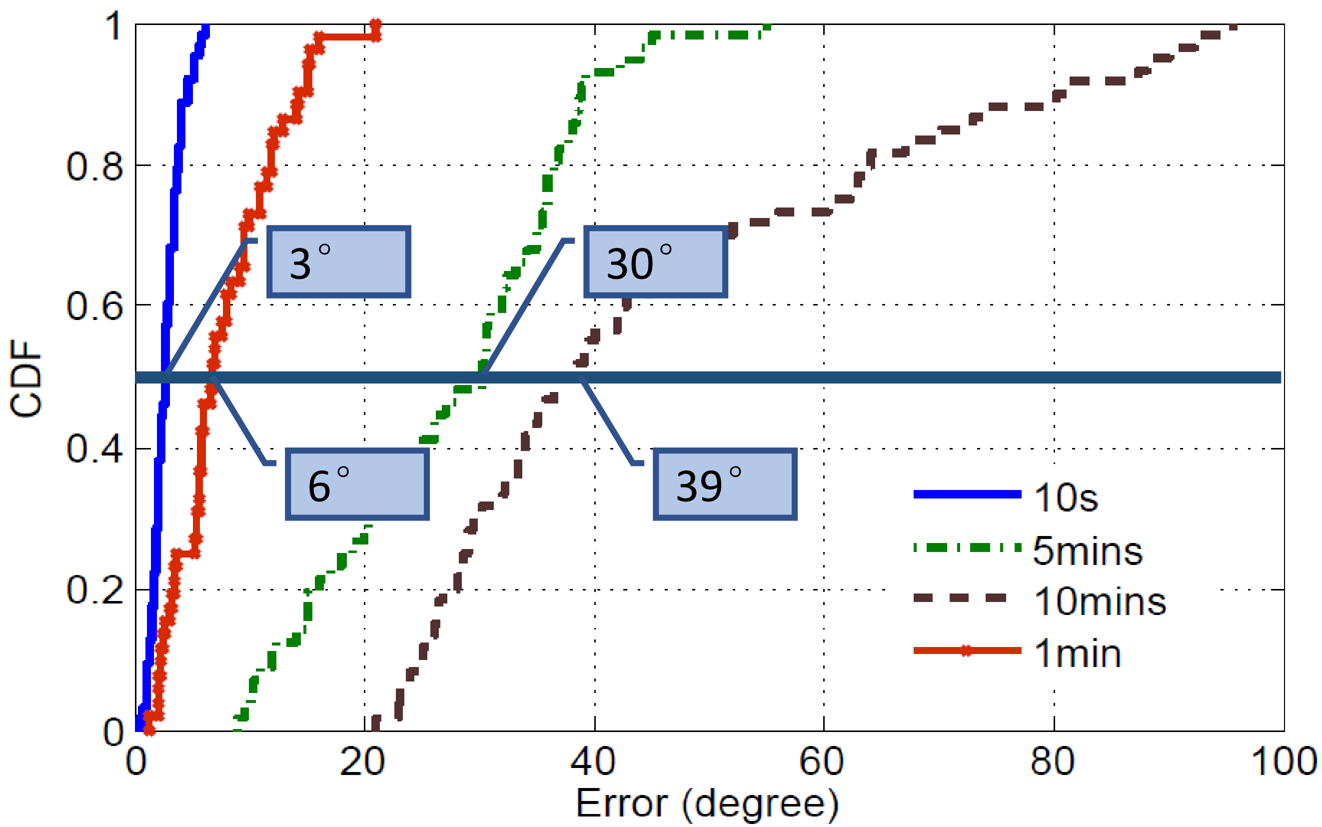
- Experiment devices
  - HTC Sensation XE mobile phone
  - Motor, dimmer, and a power supply



Impact of time integration

- Drifting error
  - the error of attitude estimation after a certain period of usage





Impact of phone motion

- Phone motion

Rotational

Angular velocities

Translational

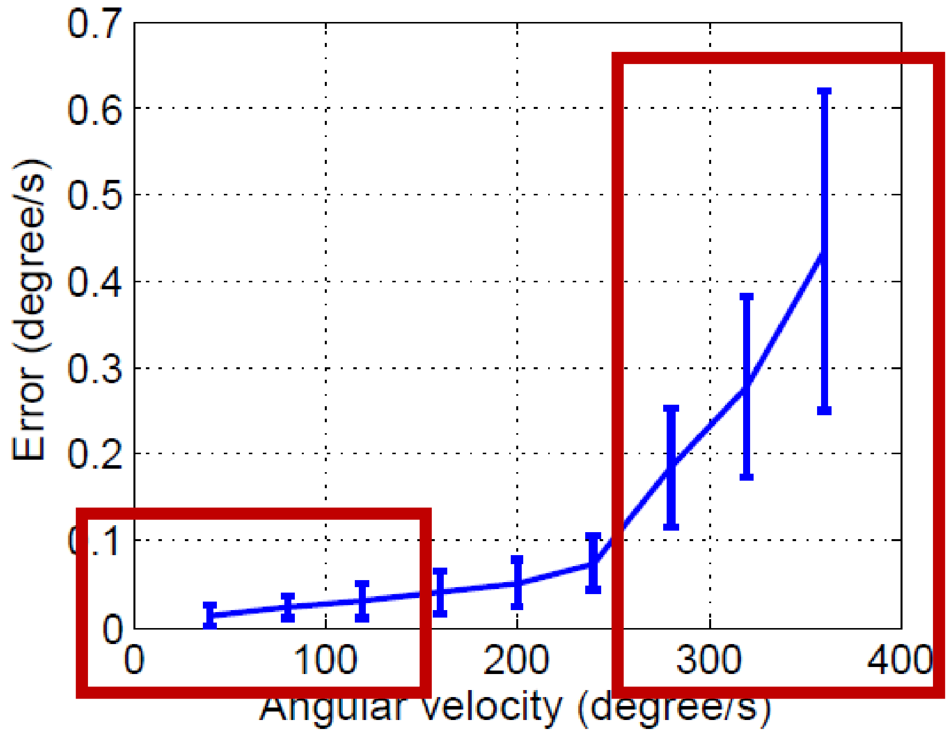
Linear accelerations

Impact of rotational motion

- Low-frequency motion v.s. out-of-range motion

Rotational

Angular velocities

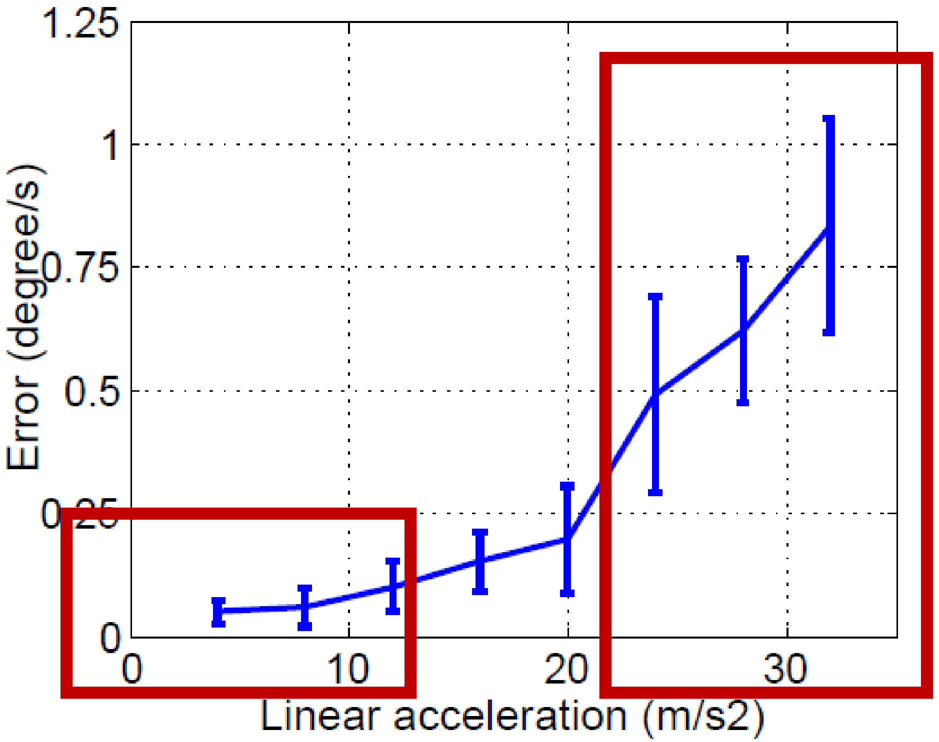


Impact of translational motion

- Low-frequency motion v.s. out-of-range motion

Translational

Linear accelerations



Gyroscope performance summary

- Error is almost **linearly** proportional to the tracking time and mobile phone motion (linear acceleration & angular velocity).

- If working within short time period and slow motion range, gyroscope is **accurate**. A **high-speed** motion significantly pollutes the **consequent** estimation results!

Orientation estimation using gravity sensor and gyroscope

- Gravity extraction using low pass filters (e.g., Butterworth Filter in Android) **- pitch, roll**
  - The extraction is accurate when phone motion is low but complicated during high-frequency motion

- Magnetic north estimation from magnetometer **- yaw**
  - The estimation is accurate outdoors but complicated indoors (magnetic interference)

Gravity extraction using accelerometer

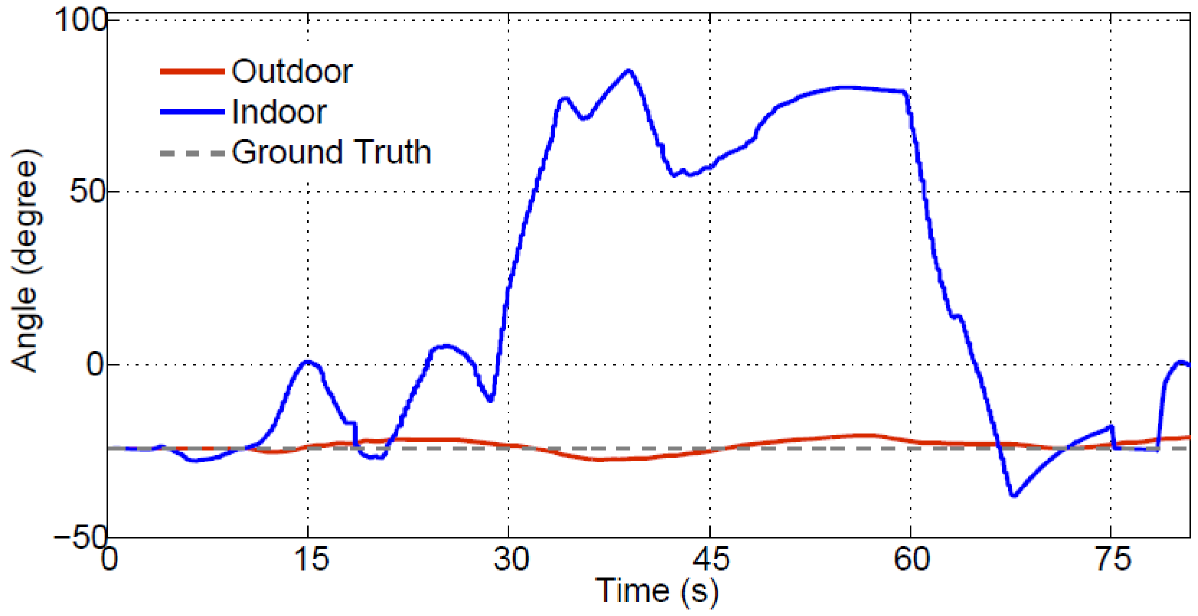
- Using low pass filters to extract gravity (e.g., Butterworth Filter in Android)

$$G(\omega_0) = \sqrt{\frac{1}{1 + {\omega_0}^{2n}}}$$

- Performance gain $G(w_0)$ depends on the phone motion

Earth north estimation using compass

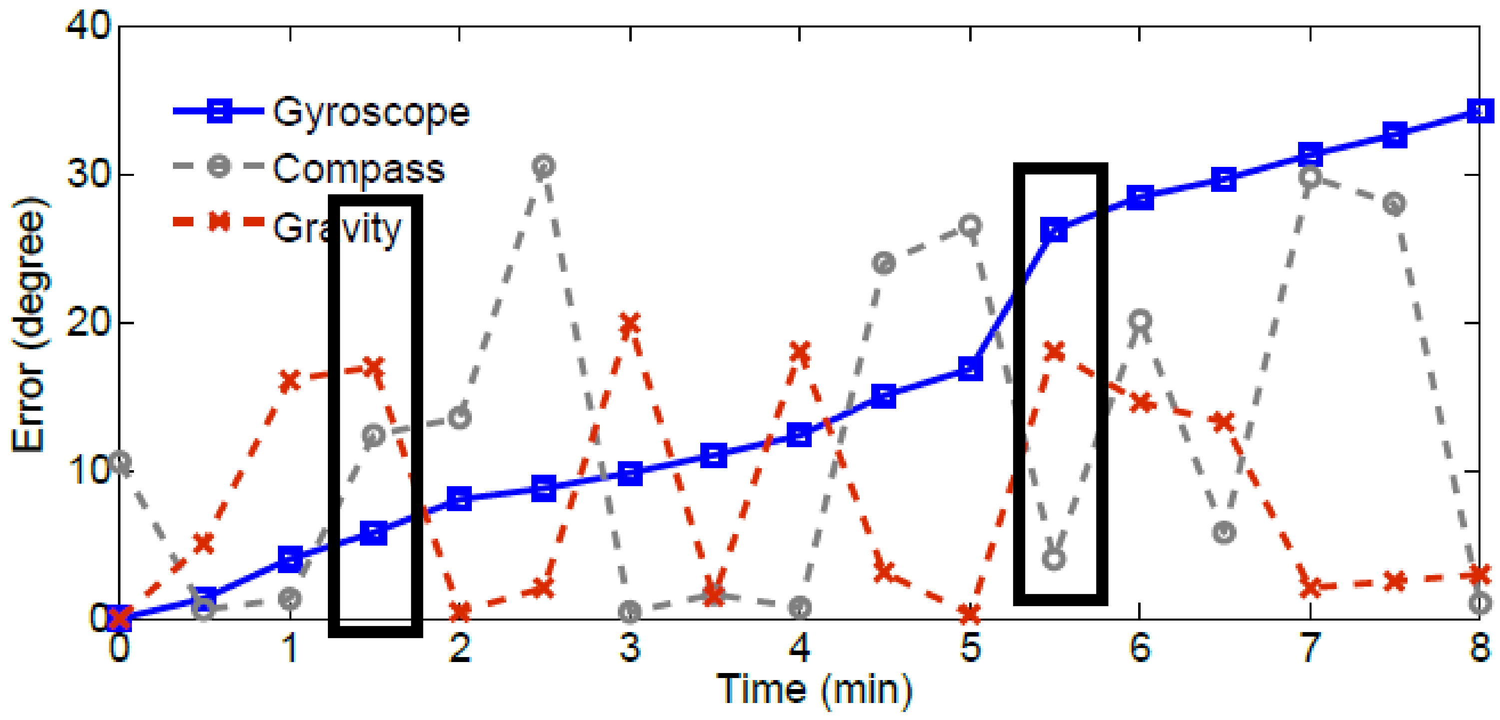
- Earth north estimation based on the earth magnetic field signal.



- Estimation is accurate outdoors but complicated indoors

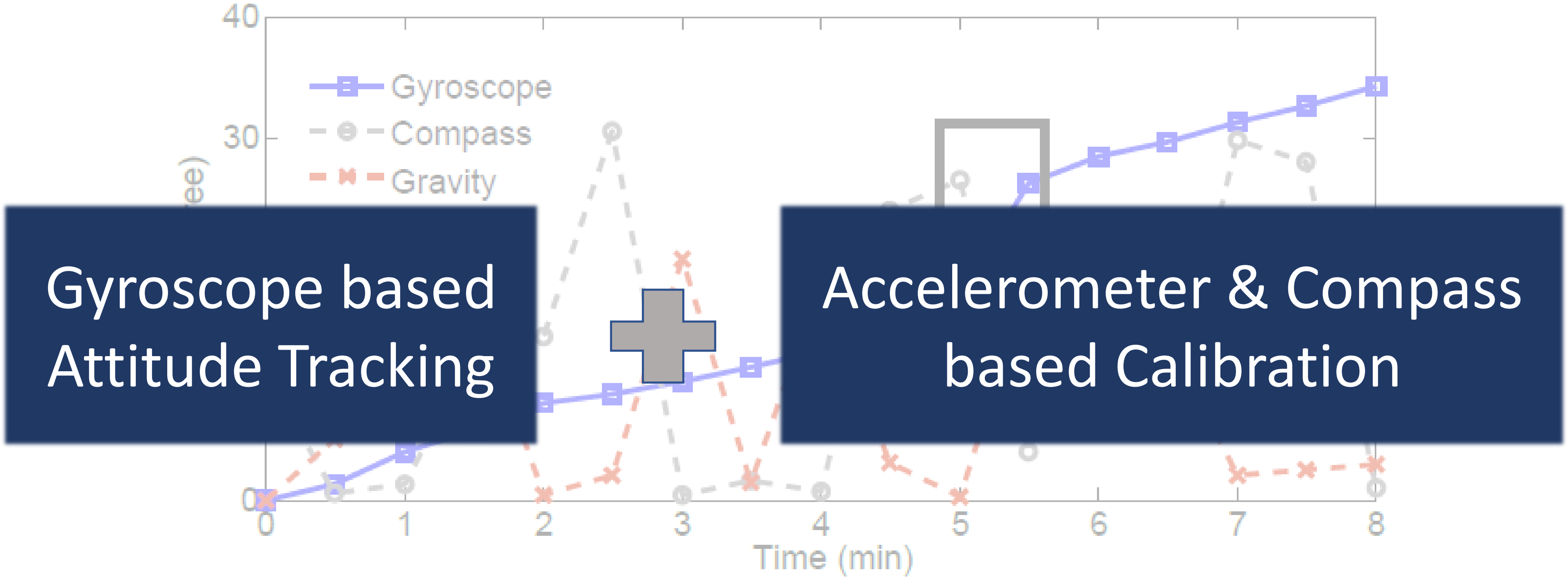
Different nature of the IMU sensors

- Sensing redundancy



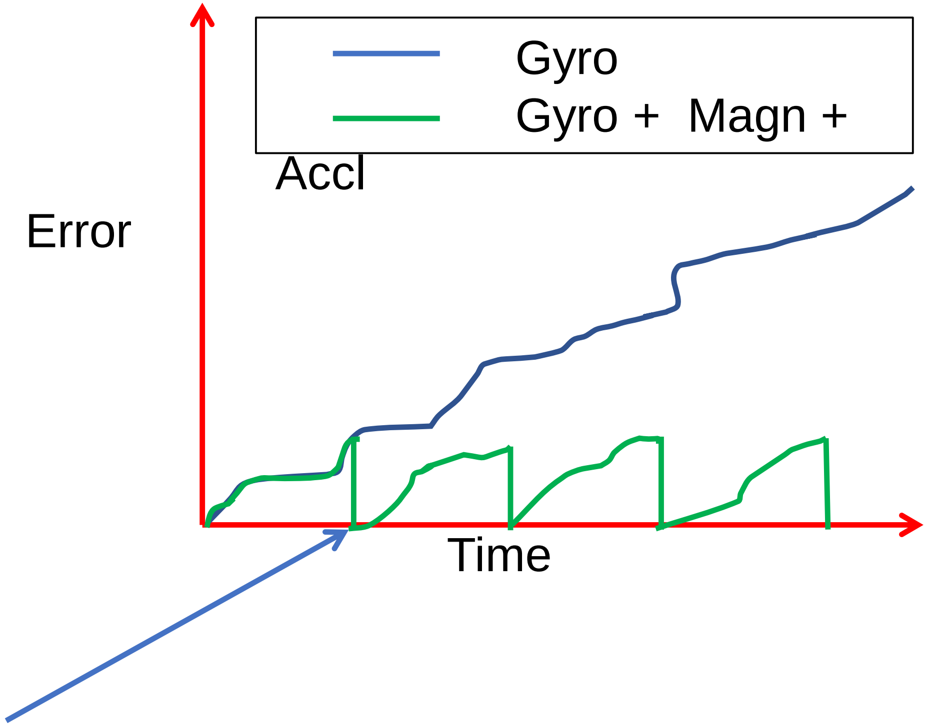
Different nature of the IMU sensors

- Sensing redundancy



High level idea to combine the two

- Use gyroscope to track orientation in general
- Errors will accumulate (drift)



- Reset errors with accelerometer/magnetometer (When the phone is static and no magnetic interference)

When to use accelerometer/compass resets

- When phone is static/slowly-moving
  - angular velocity less than 15 degrees per second (detected from gyroscope)
- When magnetic interference is low
  - phone is outdoors (detected from light sensor)

Attitude tracking

Is that all?

Problem: Reset opportunities could be too few

How to find good calibration opportunities?

- What are good calibration opportunities?
  - When the gravity extraction and earth north estimation are more accurate than the gyroscope estimation.

- Quality of the gyroscope estimation result.
- Quality of the combination of gravity and compass.

Opportunistic resetting

- What does the MEMS gyroscope measure?
  - Angular velocity: the attitude change of the mobile phone

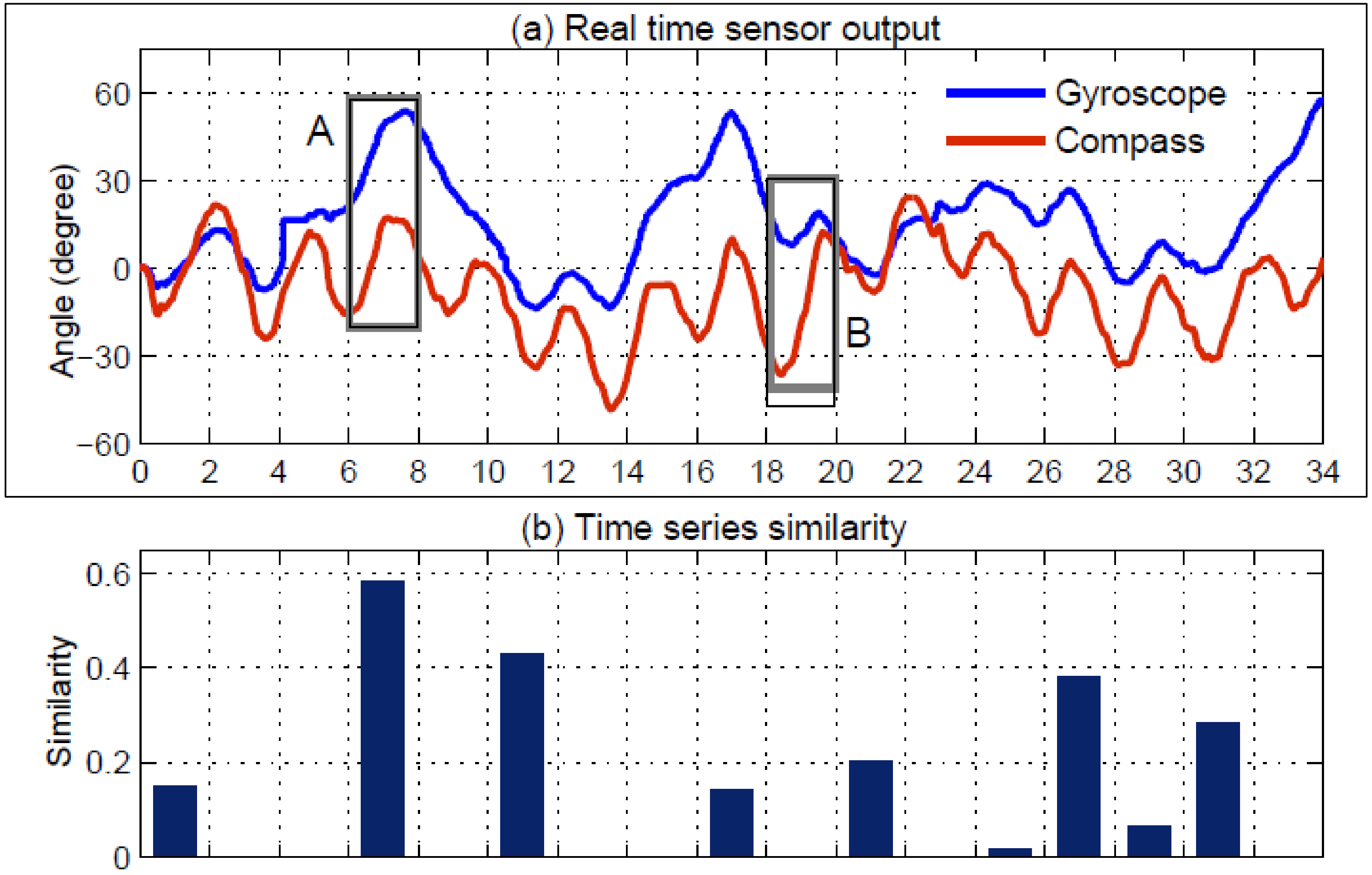
- Gyroscope is accurate within a short time period (e.g., 2 secs)

➔The measure of the attitude **change** is accurate

➔We can compare the **change** of gravity estimation and earth north estimation with that of gyroscope

➔Similar **trend** indicates a positive resetting opportunity

Opportunistic resetting

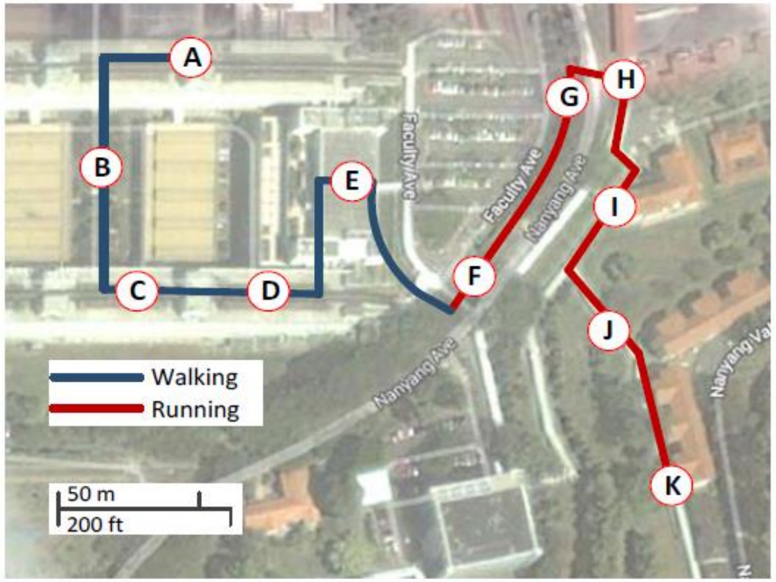


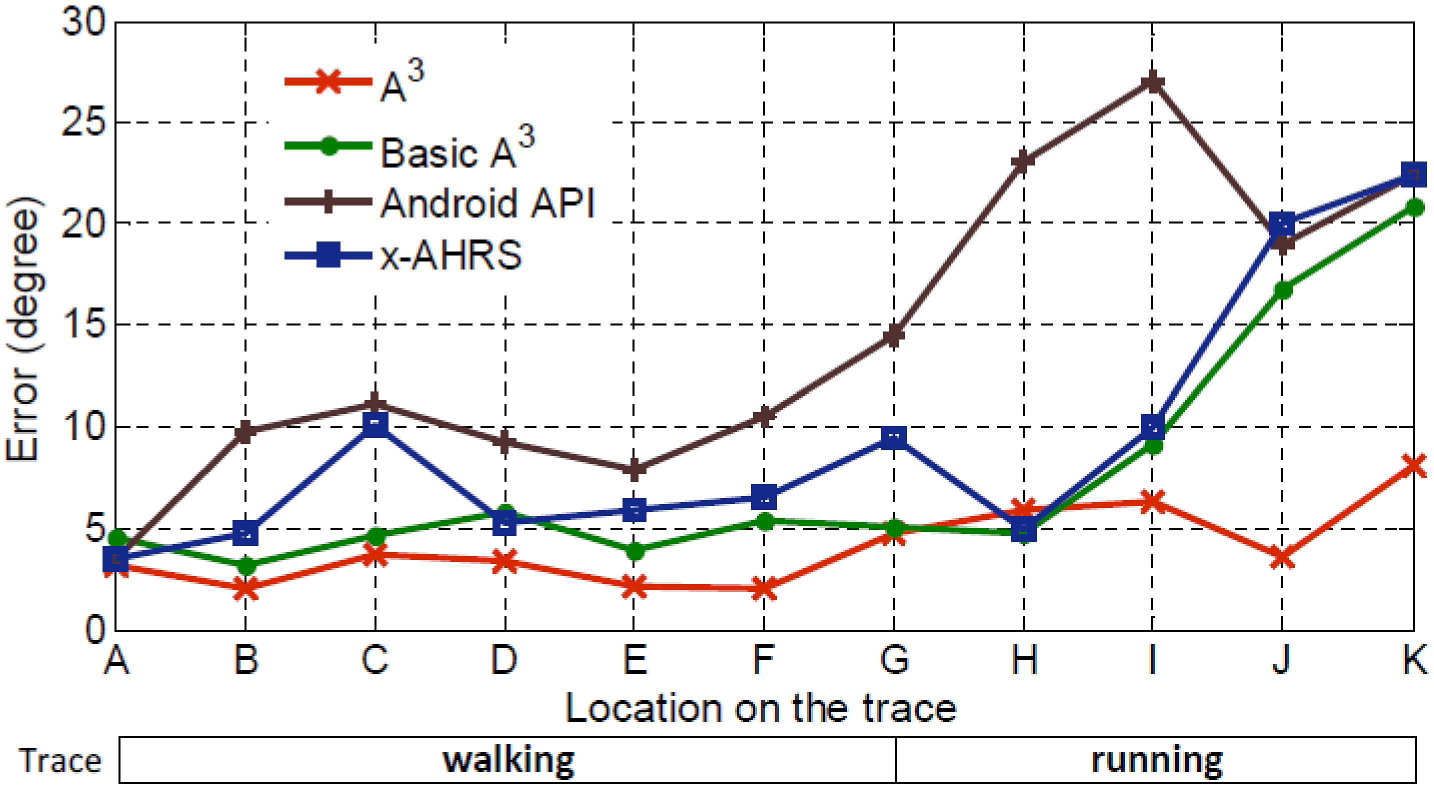
Evaluation

Evaluation settings

- Mobile Phones
  - HTC Sensation XE, Samsung Galaxy S2 i9100, and LG Google Nexus 4
- Scenarios: walking in hand & in pocket
- Comparison
  - Basic A^3
  - A^3
  - Android API
  - x-AHRS
- Popular apps investigation

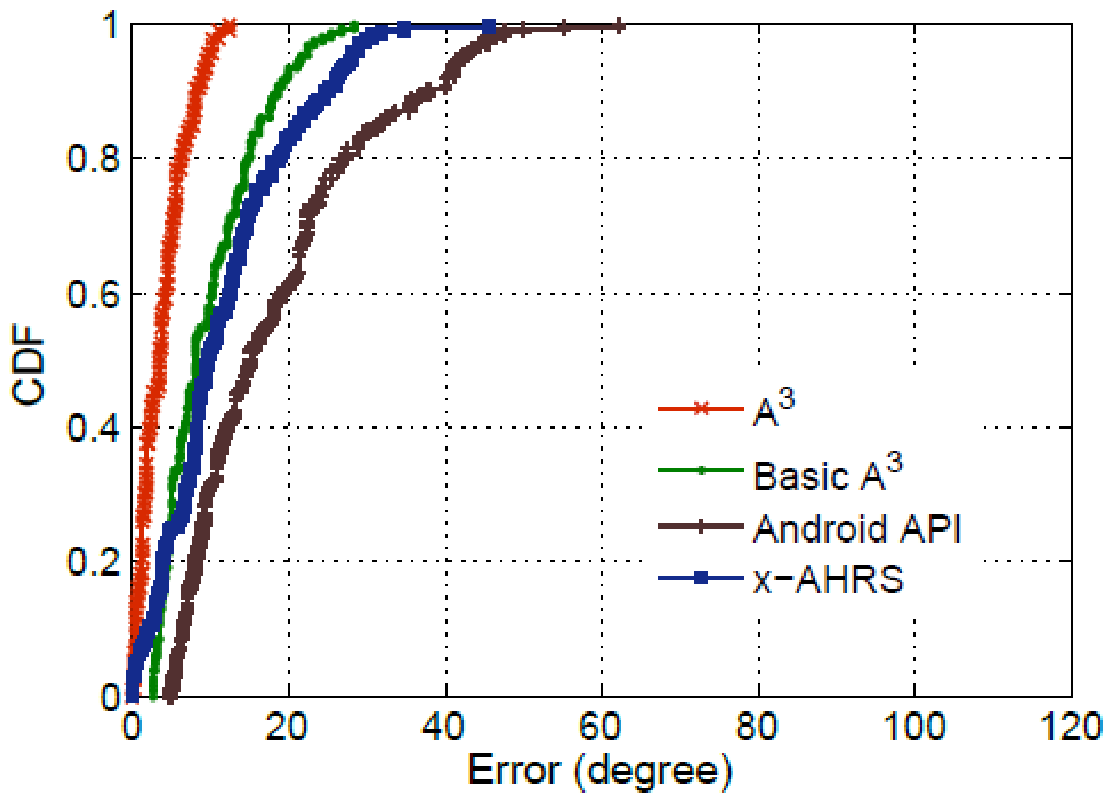
An instant trace



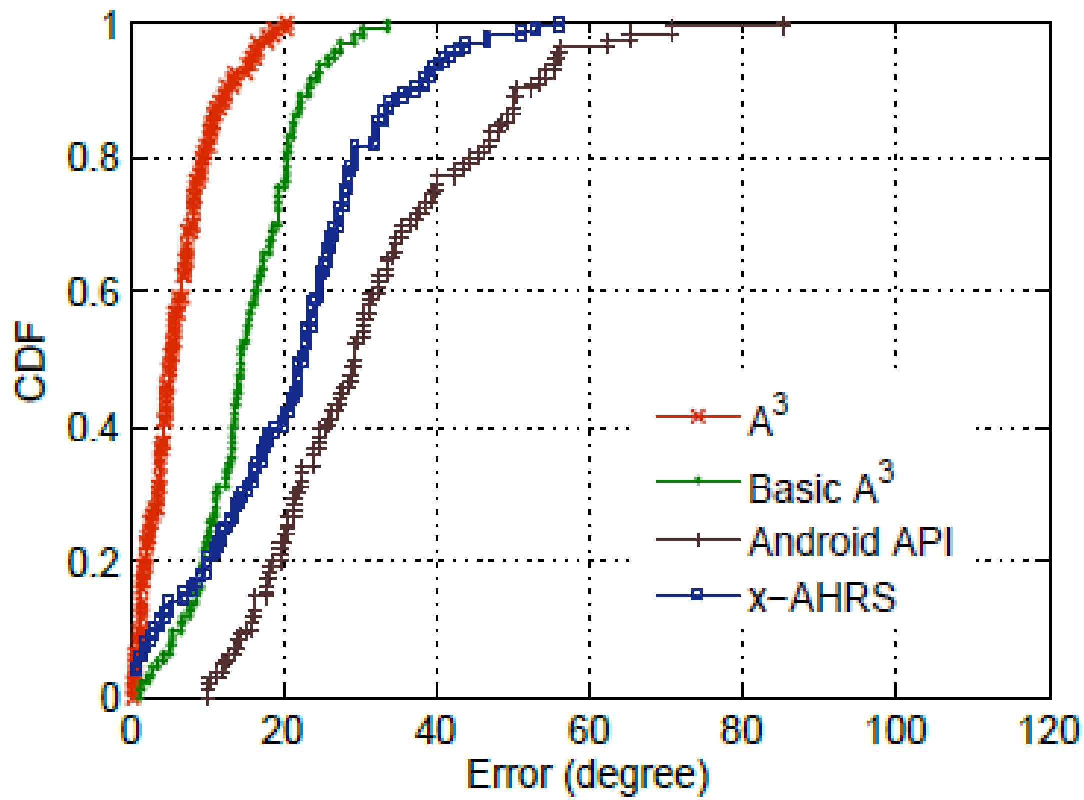


Performance in different scenarios

- Walking in hand



- Walking in pocket



In popular apps

App	Type	Time	App Error	A^3 Error
Sensor Box	Sensor app	1 min	40°	3.5°
		5 mins	50°	5°
		10 mins	75°	6.4°
Show Down	Game	1 min	9°	5°
		5 mins	30°	7°
		10 mins	35°	6°
Gyroscope Rotate	Sensor app	1 min	10°	5°
		5 mins	28°	8°
		10 mins	45°	4.5°

Conclusion

- Detailed studies to understand the basic performance of mobile phone IMU sensors and their sensitivity to environments
- A novel phone attitude estimation method which fully exploits the sensing redundancy of gyro, accelerometer and compass
- A novel opportunistic calibration technique which looks at the trend of the estimation instead of the absolute value

ArmTrak: Tracking the user’s arm movements using a wearable sensor

Understanding human arm motion

How is the arm moving?

What is the meaning of this motion?

Gesture Recognition

What is the meaning of this motion?

Gesture Recognition

Running

Smoking

Drinking

Driving

Posture Tracking

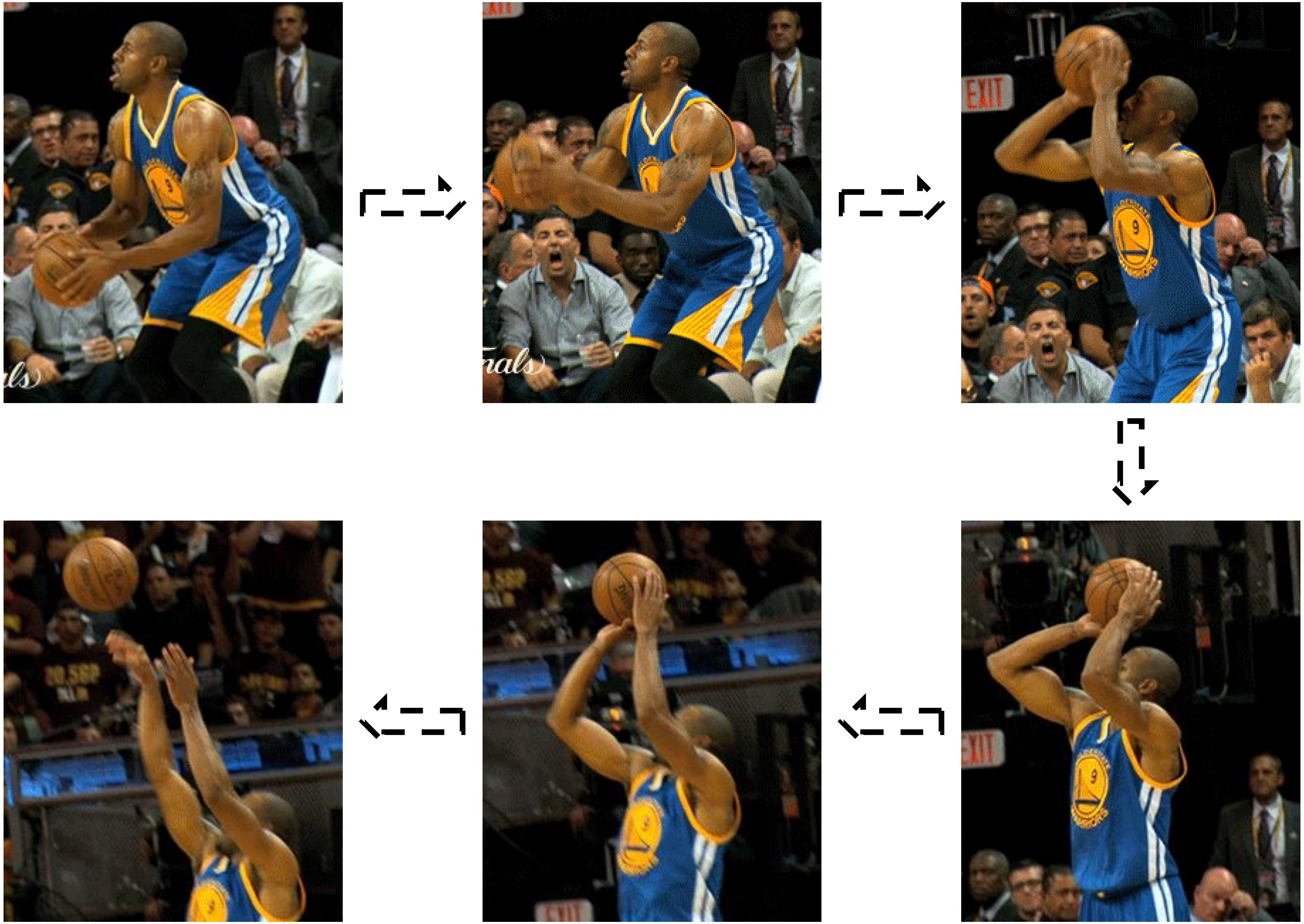
↑

How is the arm moving?

Gesture Recognition

↑

What is the meaning of this motion?


EXIT

Arm Posture Tracking - Applications

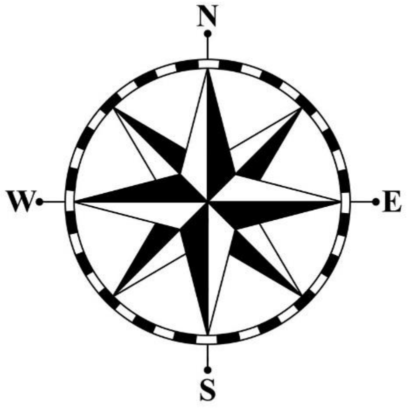
Natural User Interface

Sports Analytics

Can we track arm postures with a smartwatch alone?

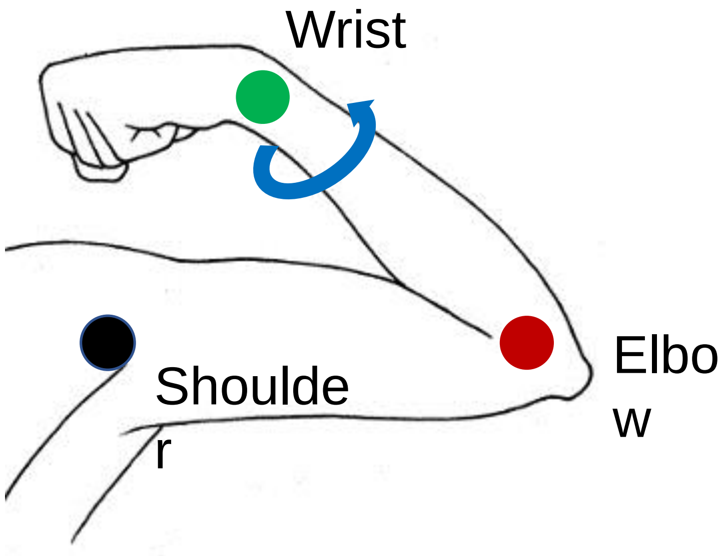
Can we track arm postures with a **smartwatch** alone?

- What is inside a smartwatch?

Accelerometer	Gyroscope	Compass
		
Acceleration along 3 axes	Rotation speed around 3 axes	North vector projected to 3 axes

Can we track **arm postures** with a smartwatch alone?

- What do we need to track?



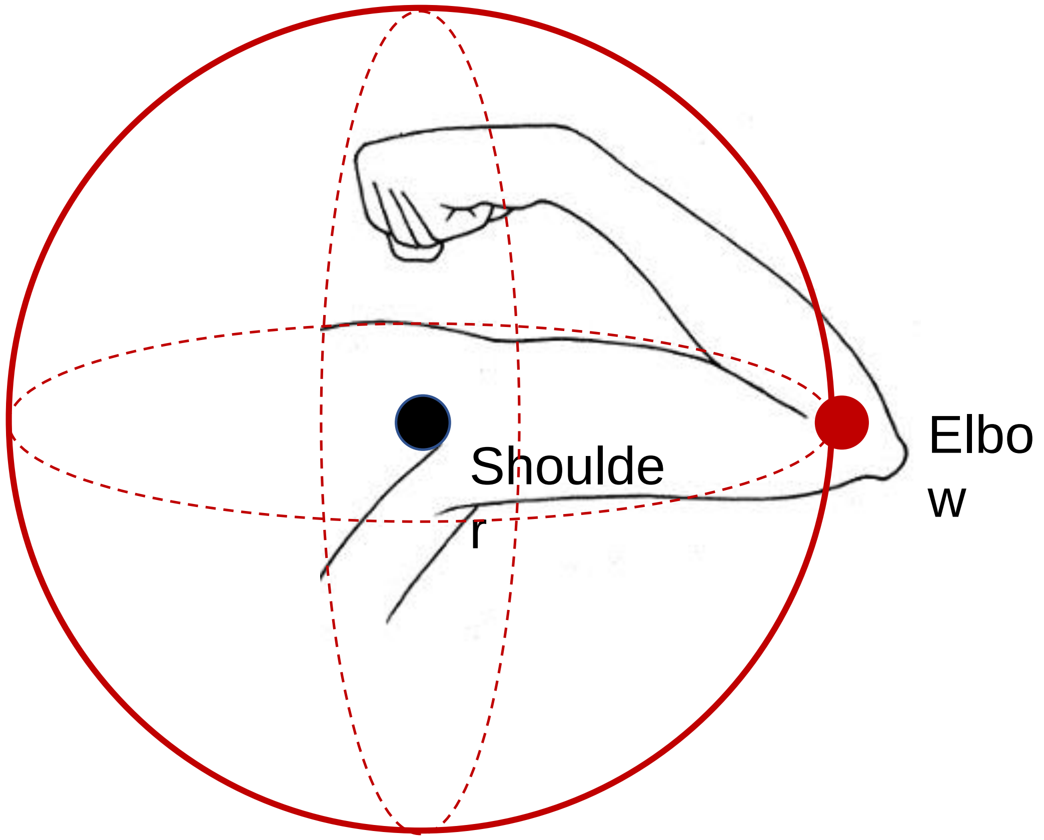
Posture = < Elbow Location, Wrist Location, Wrist Rotation >

Can we track **arm postures** with a smartwatch alone?

- What do we need to track?

Elbow
Location

3D Sphere
(DoF: 2)



Can we track **arm postures** with a smartwatch alone?

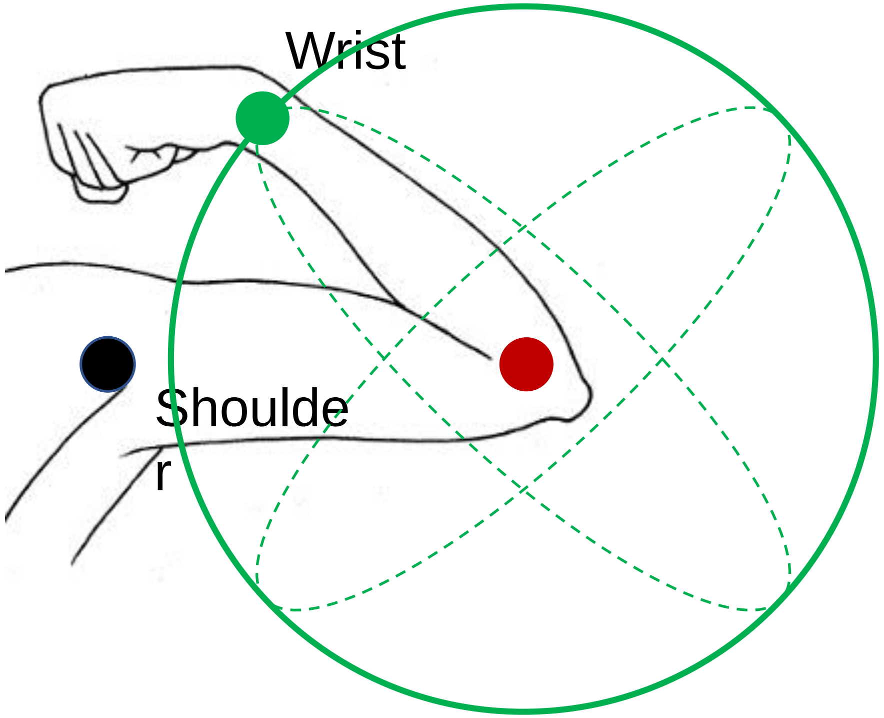
- What do we need to track?

Elbow Location

3D Sphere (DoF: 2)

Wrist Location

3D Sphere (DoF: 2)



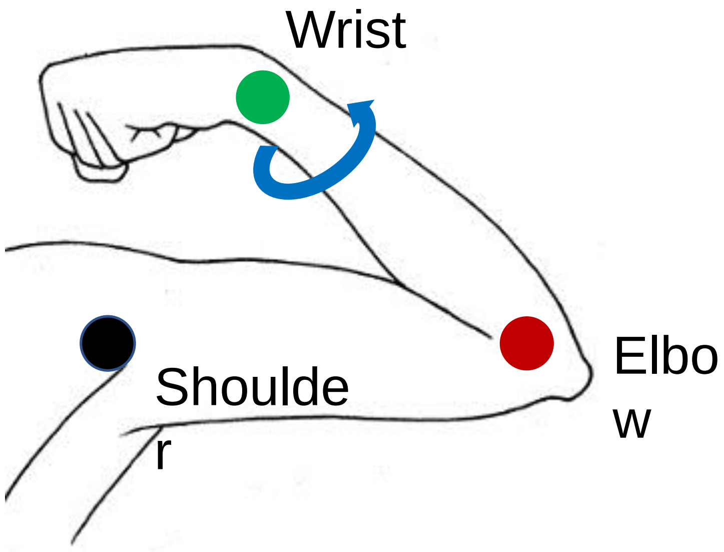
Can we track **arm postures** with a smartwatch alone?

- What do we need to track?

Elbow Location — 3D Sphere (DoF: 2)

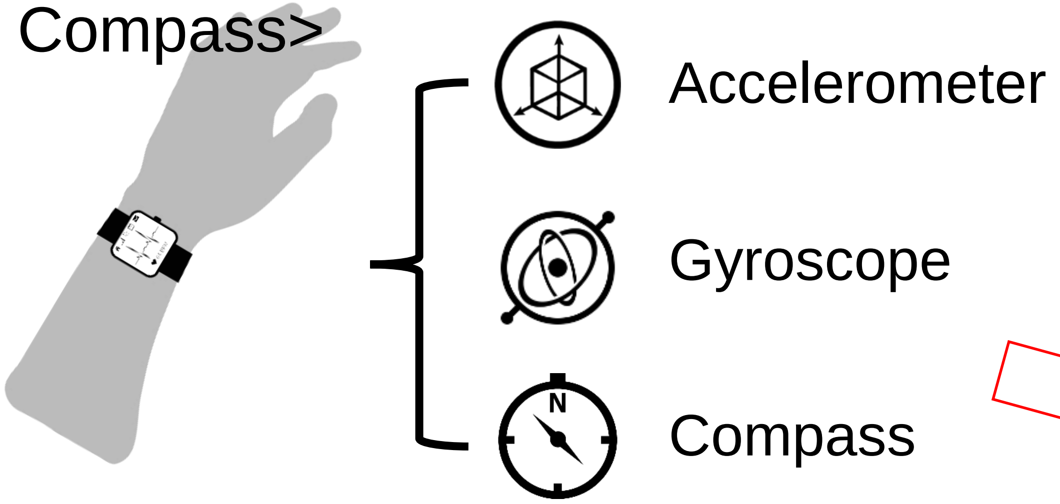
Wrist Location — 3D Sphere (DoF: 2)

Wrist Rotation — 1D Angle (DoF: 1)

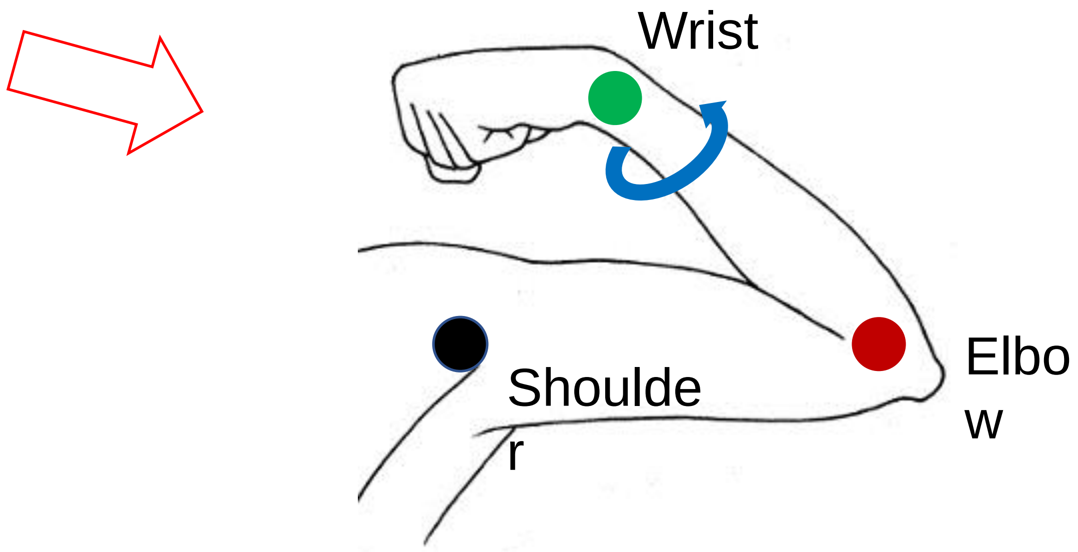


Can we track **arm postures** with a smartwatch alone?

Smartwatch = < Accelerometer, Gyroscope, Compass>



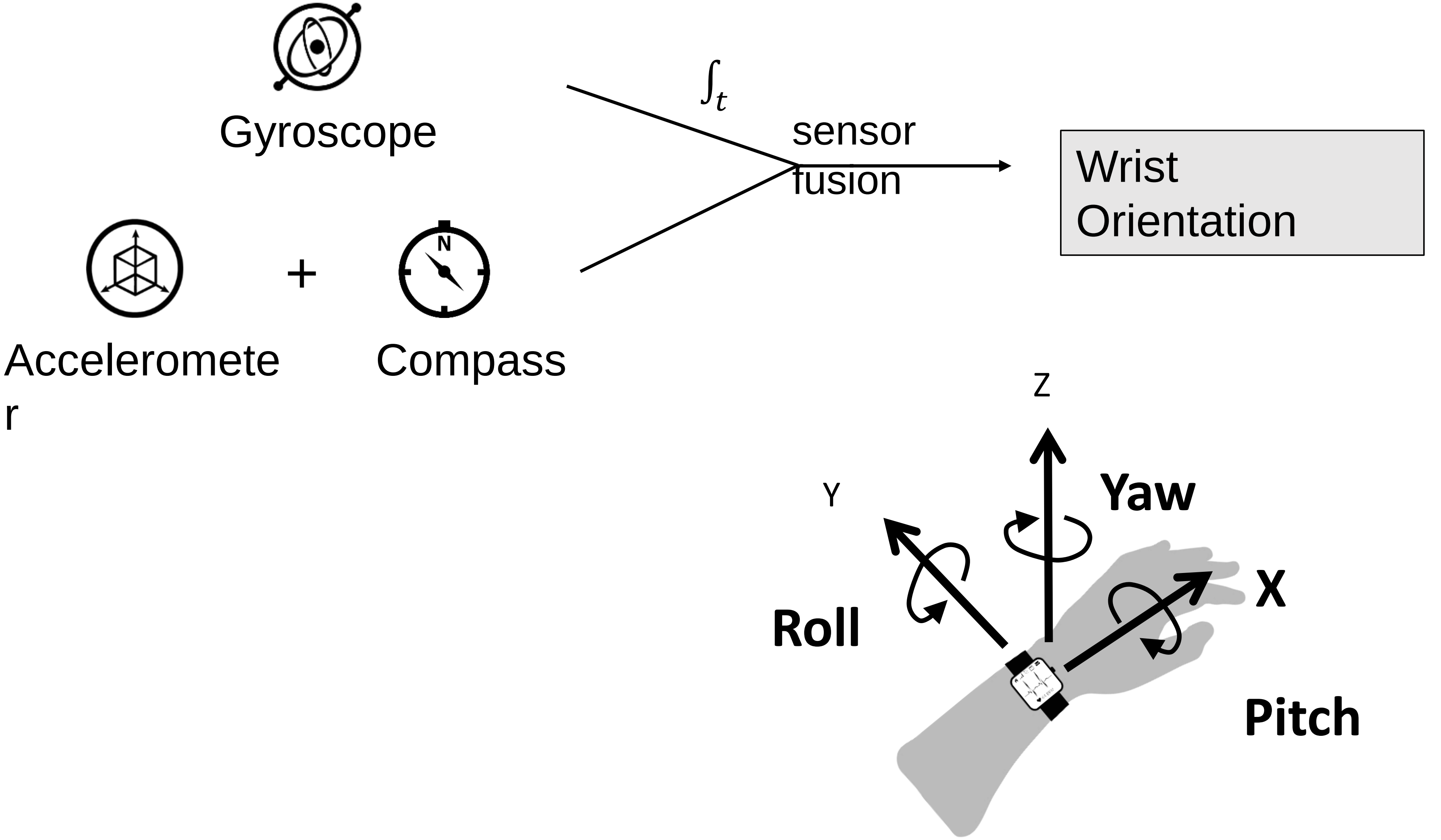
?



Posture = < Elbow Location, Wrist Location, Wrist Rotation >

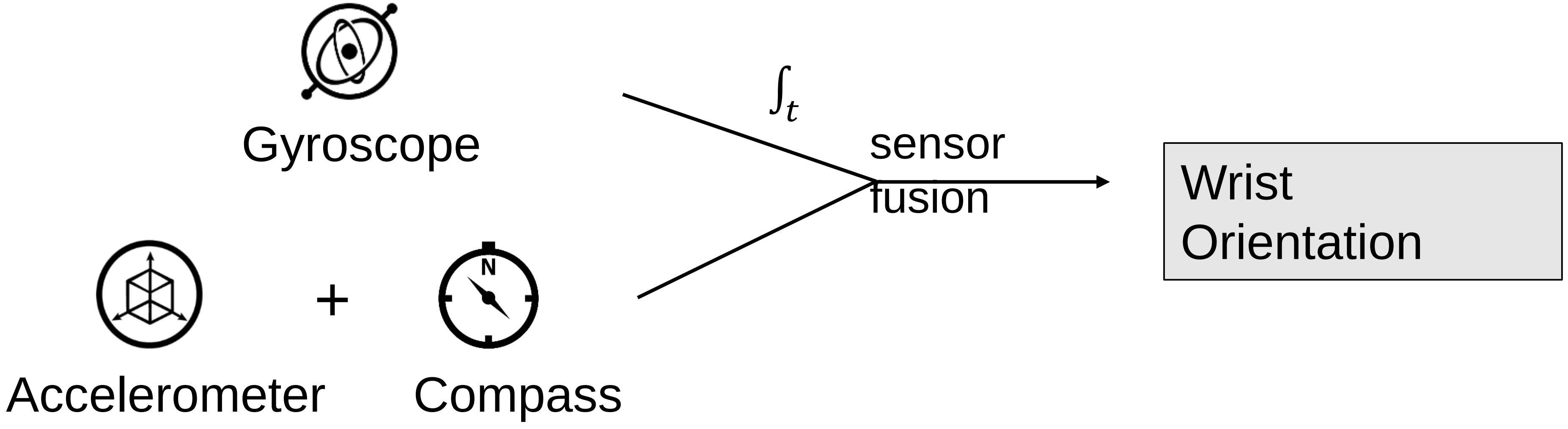
Experiment 1: Double Integration

Step 1

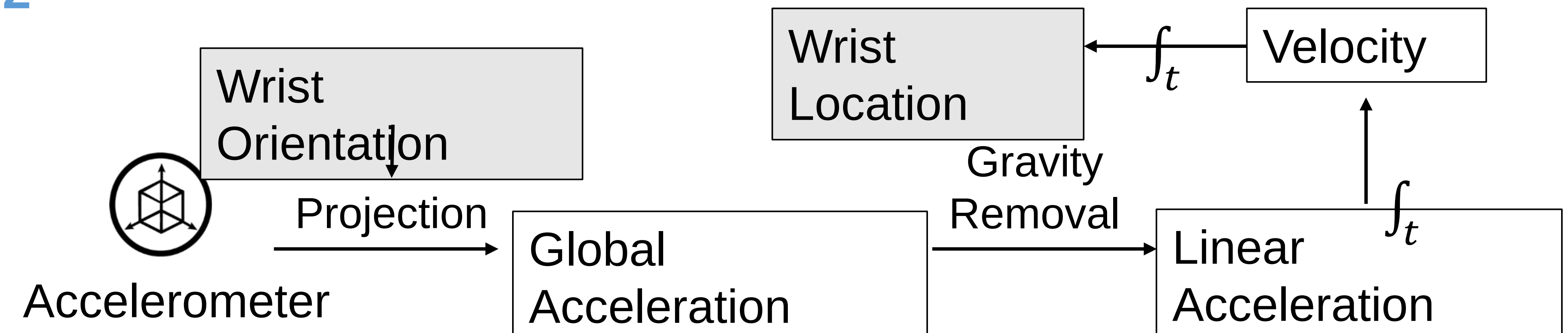


Experiment 1: Double Integration

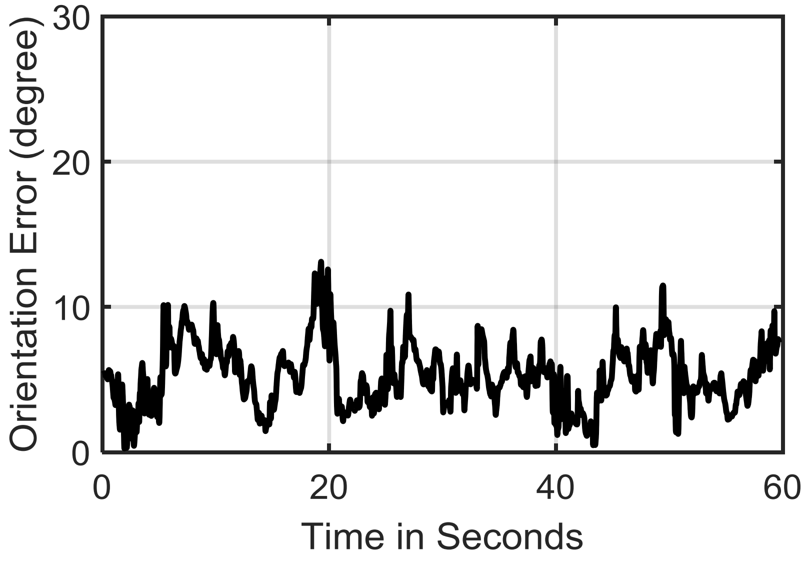
Step 1

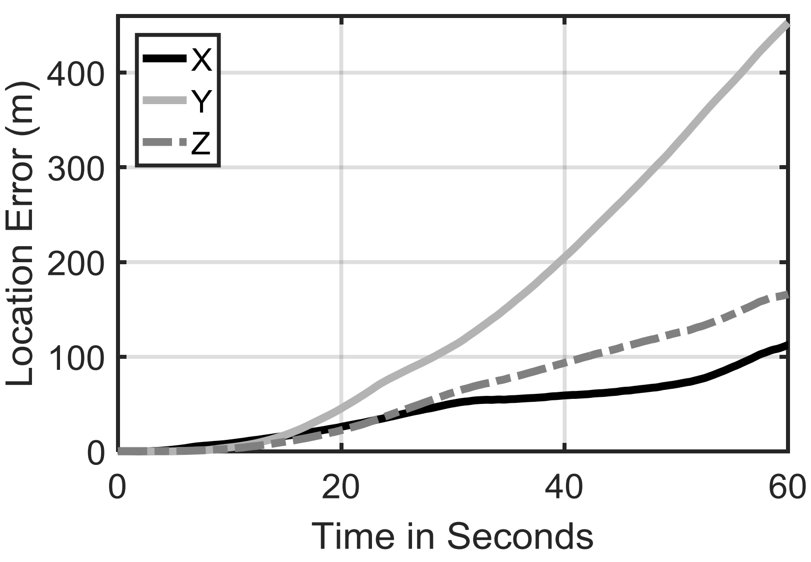


Step 2



Experiment: Double Integration





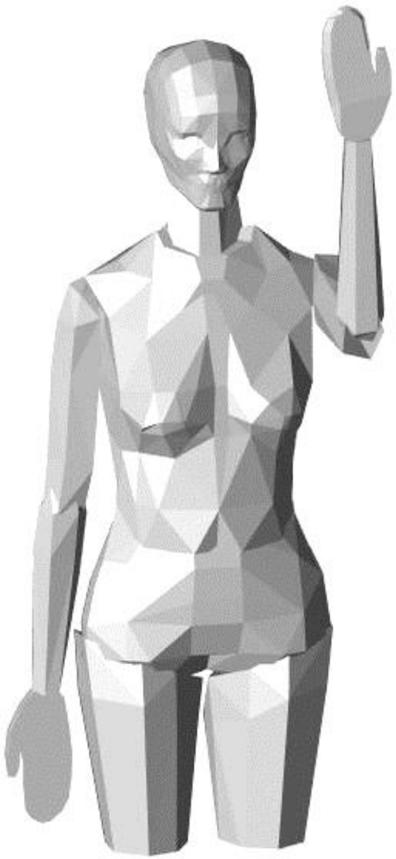
Wrist Orientation

Wrist Location

- Wrist orientation error is okay…
- Wrist location error goes unbounded!

Double integration won’t work in unconstrained space

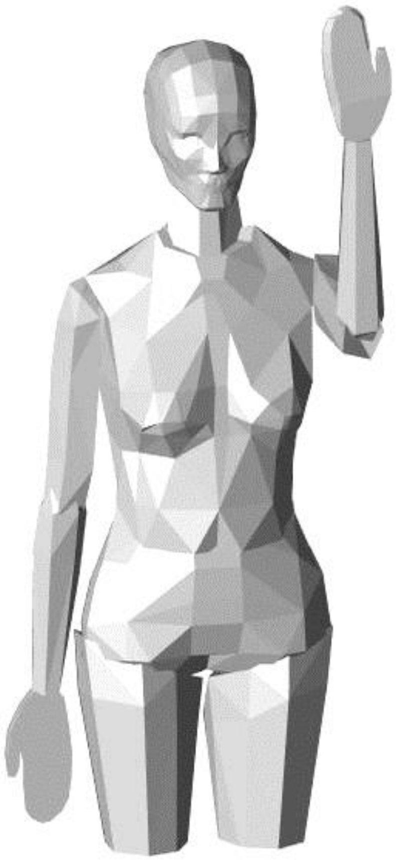
What happens if wrist orientation is fixed…



Forearm pointing upward
Palm facing towards yourself

What happens if wrist orientation is fixed…

Elbow

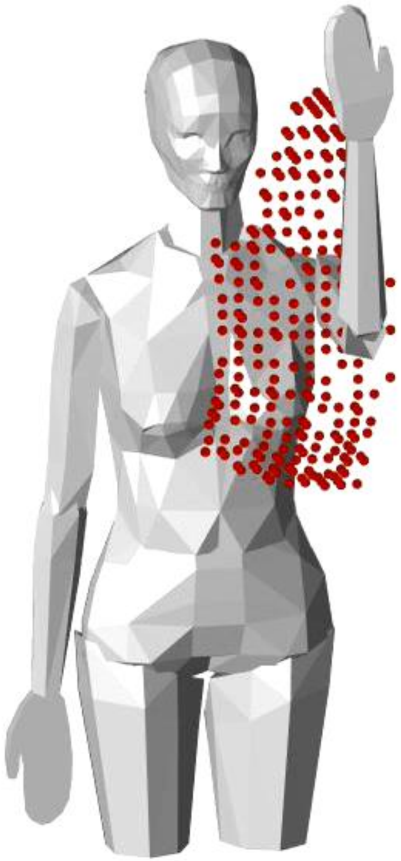


Forearm pointing upward
Palm facing towards yourself

Elbow Point Cloud:
A subset of elbow sphere

What happens if wrist orientation is fixed…

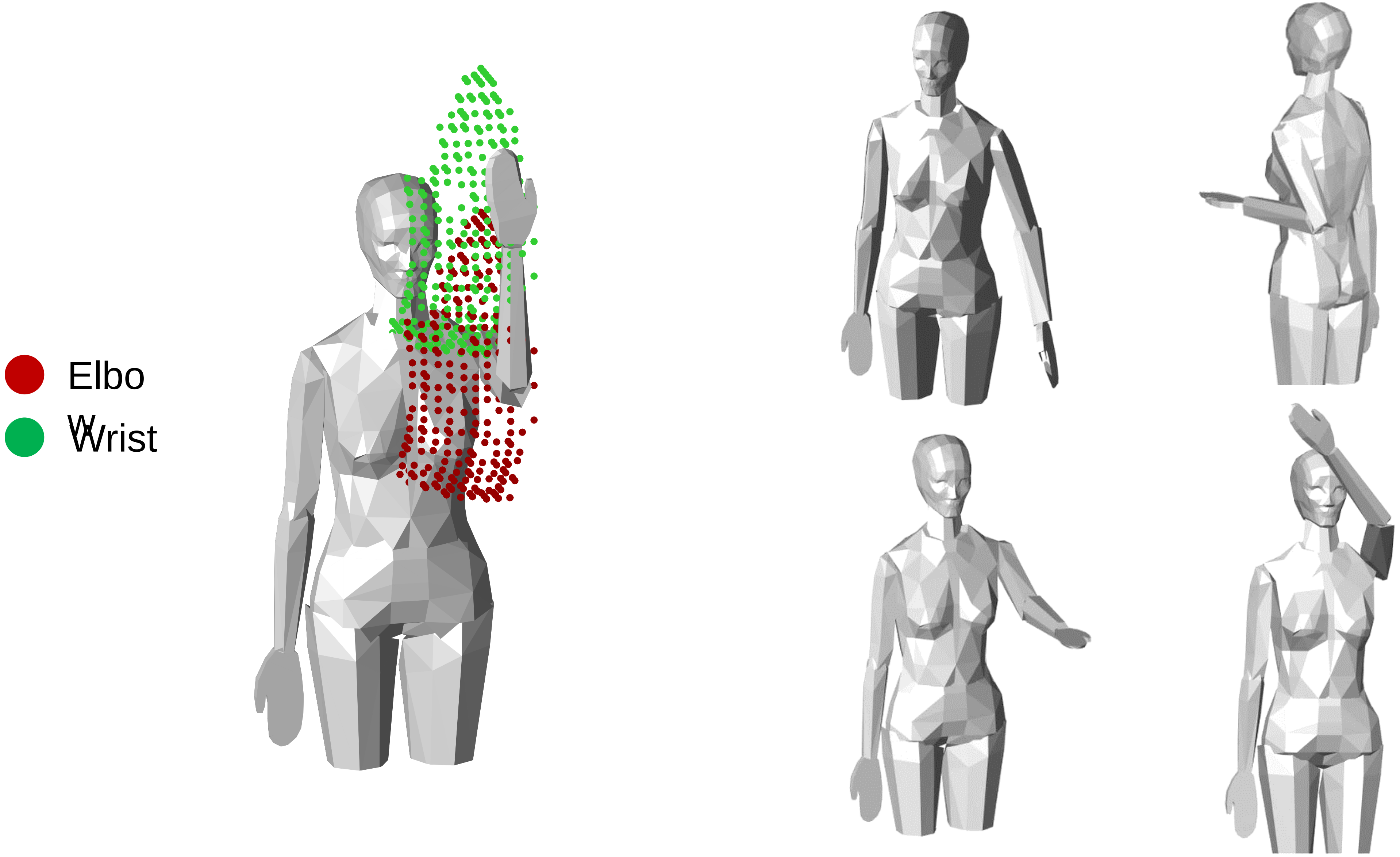
Elbo
w
Wrist



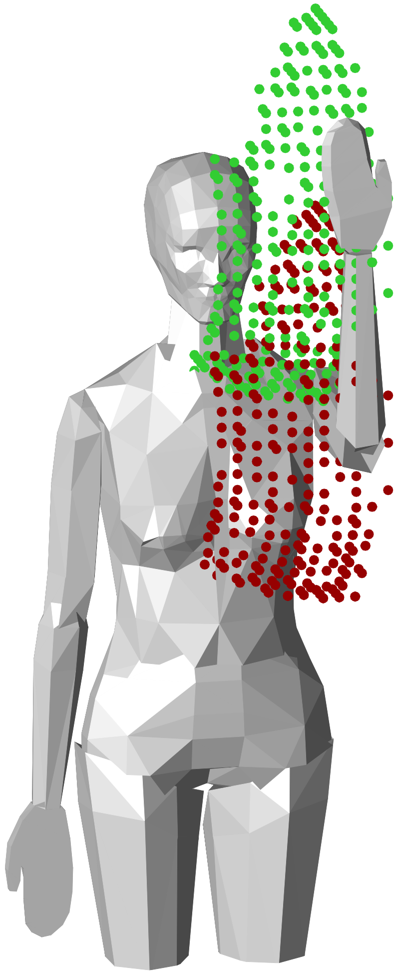
Forearm pointing upward
Palm facing towards yourself

Elbow Point Cloud:
A subset of elbow sphere

Wrist Point Cloud:
A shift of elbow point cloud, along forearm direction


Elbo
w
Wrist

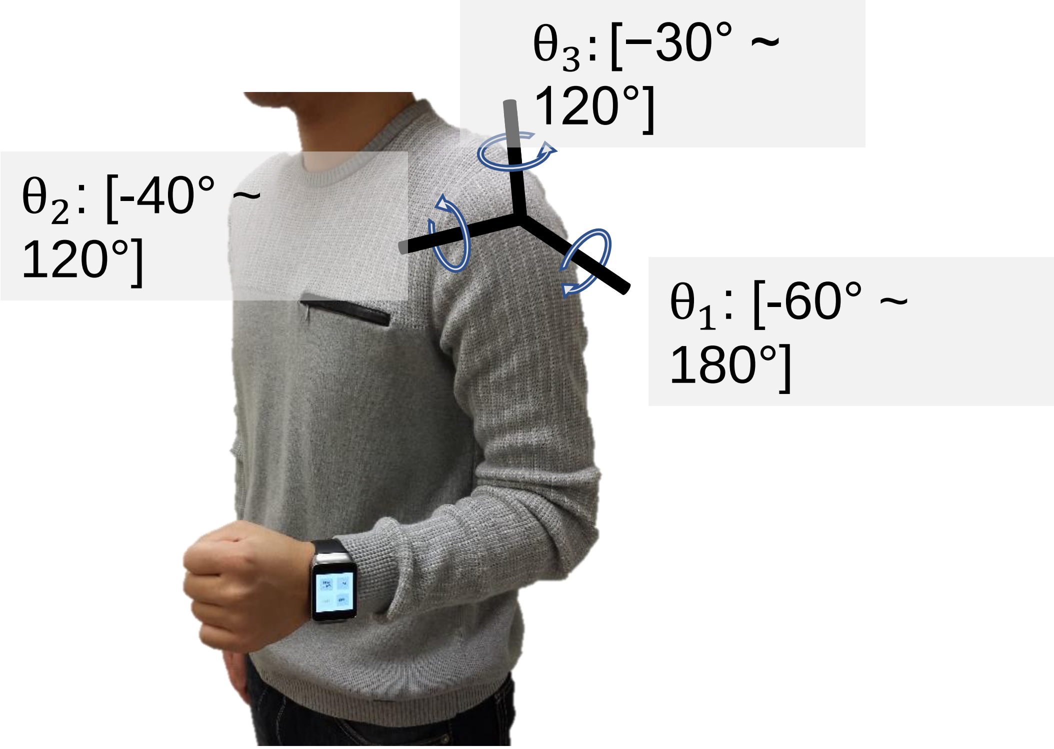
What happens if wrist orientation is fixed…



- For a fixed <u>wrist orientation</u>, arm posture space is small!
- This is promising, as we already estimate <u>wrist orientation</u> reasonably well…
- But how can we derive this point cloud for each <u>wrist orientation</u>?

Human Arm Model

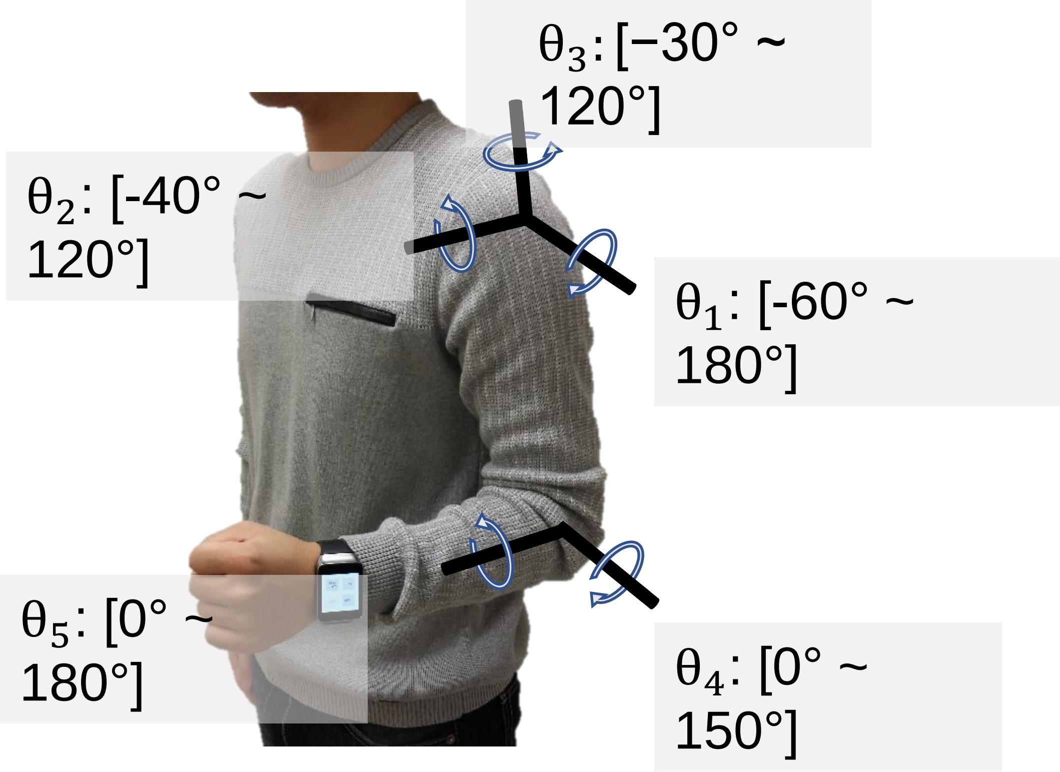
Shoulder: $\theta_1, \theta_2, \theta_3$



Human Arm Model

Shoulder: $\theta_1, \theta_2, \theta_3$

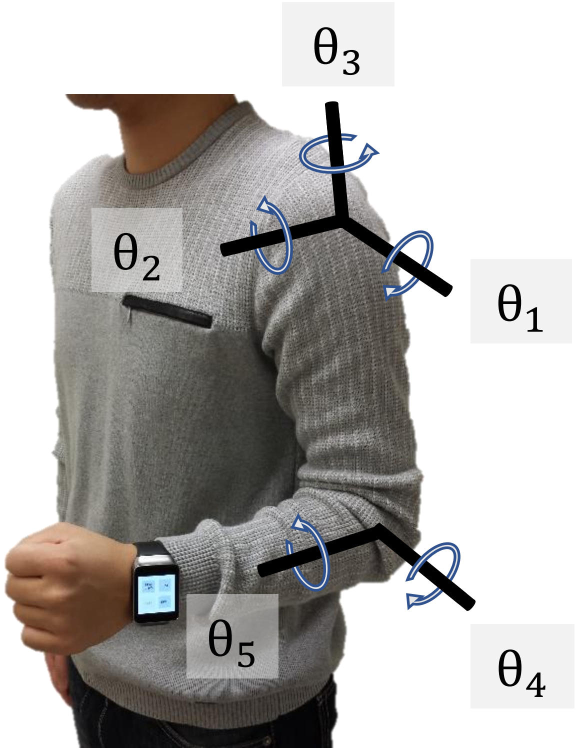
Elbow: θ_4, θ_5



Human Arm Model

Shoulder: $\theta_1, \theta_2, \theta_3$

Elbow: θ_4, θ_5



Elbow Location $= f(\theta_1, \theta_2)$

$$= \mathrm{l_u}\begin{pmatrix} \cos(\theta_2)\sin(\theta_1) \\ \sin(\theta_2) \\ -\cos(\theta_1)\cos(\theta_2) \end{pmatrix}$$

Wrist Location $= g(\theta_1, \theta_2, \theta_3, \theta_4)$

Wrist Orientation $= h(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$

Orientation – Location Mapping

1-N Mapping for each orientation

Wrist Orientation

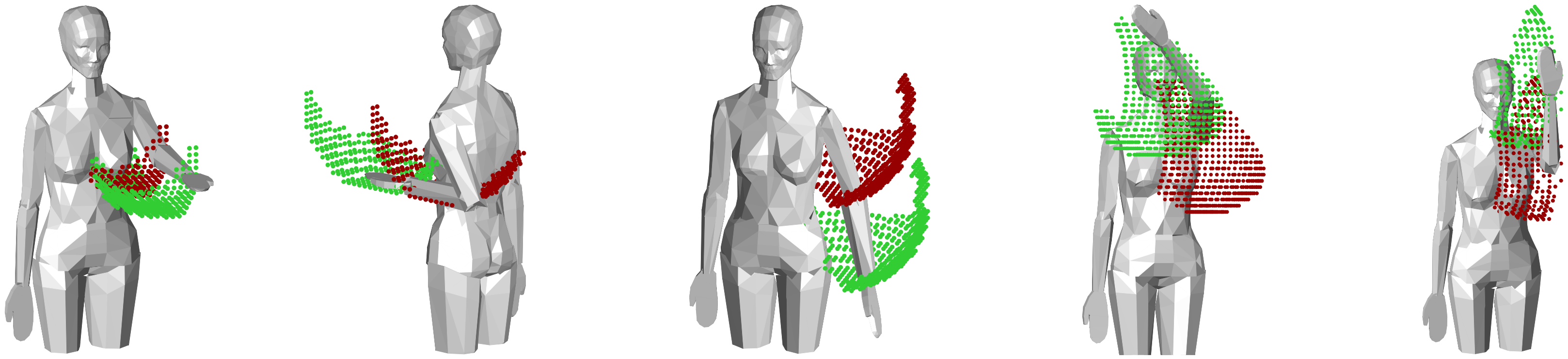
$h(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$

Elbow Location

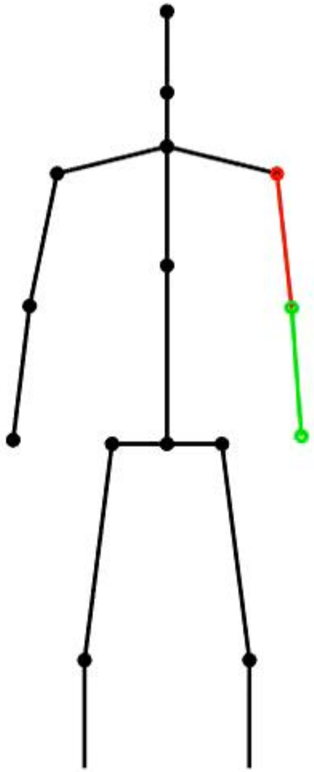
$f(\theta_1, \theta_2)$

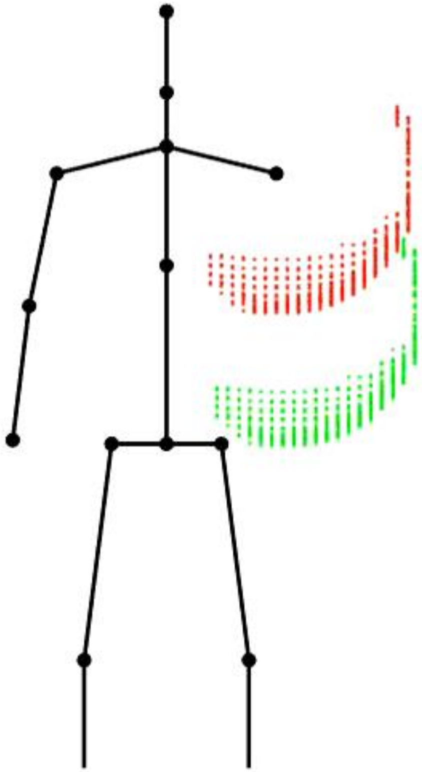
Wrist Location

$g(\theta_1, \theta_2, \theta_3, \theta_4)$



Video: Point Cloud Tracking



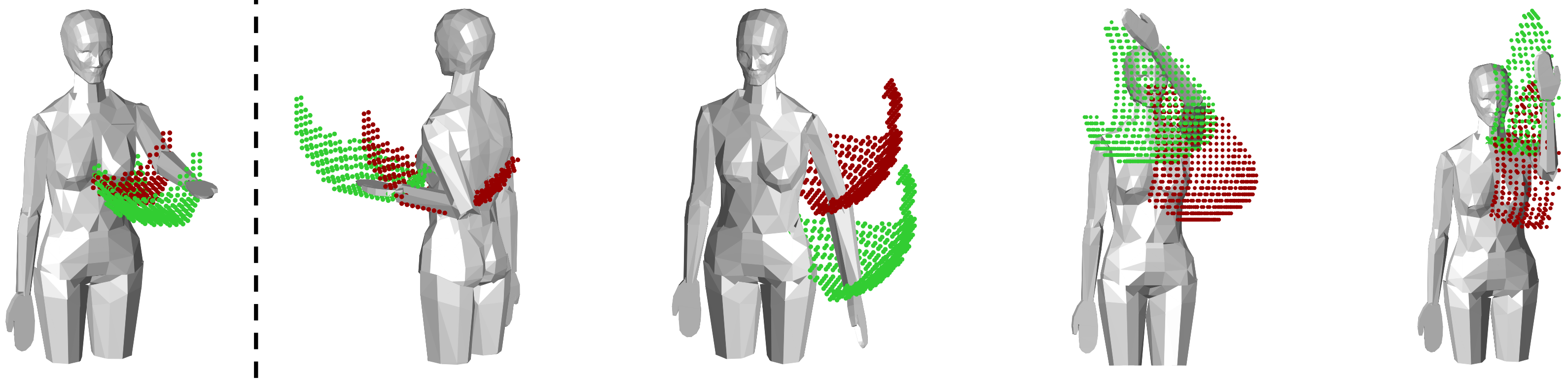


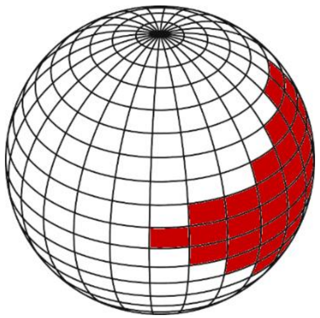
RGB
Video

Kinect
Groundtruth

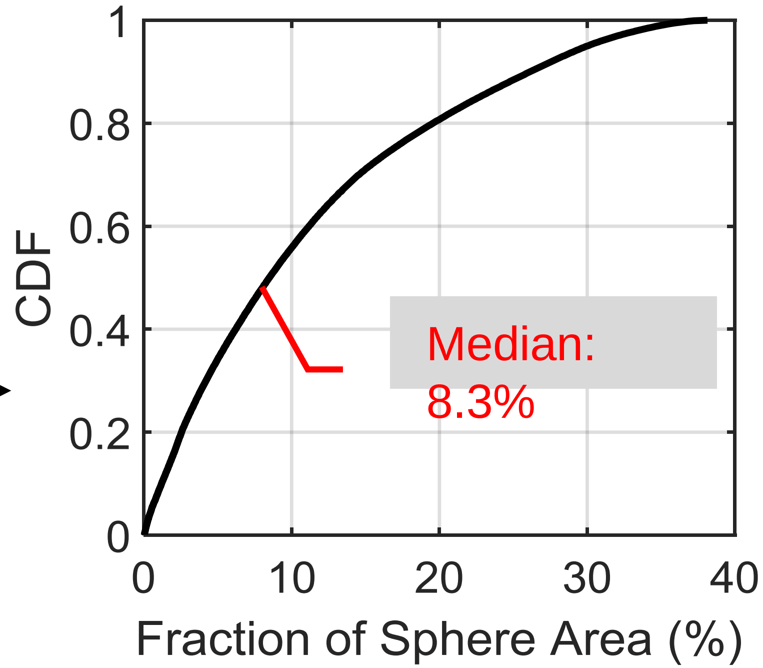
Elbow/Wrist Point
Clouds

How large are the point clouds?

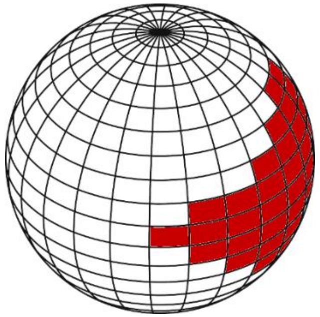




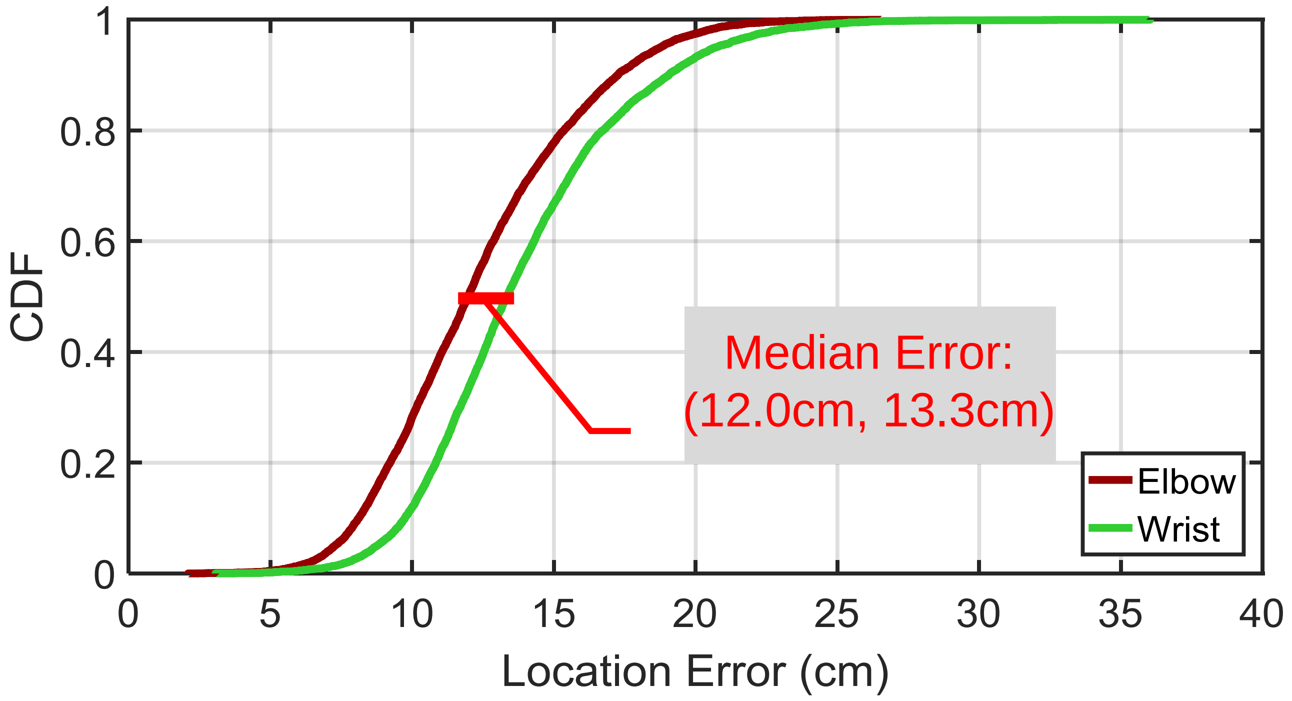
Fraction = $\frac{\text{Area of Red}}{\text{Area of Sphere}}$



How large are the point clouds?



Since they are small, what if we simply take an average?



Video: Write in the Air

Limitation

- Facing direction
  - Need to express arm posture in torso coordinate system

- Tracking on the move
  - Body motion will pollute accelerometer signal

Conclusion

- Tracking arm postures using motion sensors on a smartwatch alone

- <12cm, 13cm> tracking error for <elbow, wrist>