APPLYING KALMAN FILTERING ON A QUADRUPED ROBOT

Lucian Cucu

Professor:
A. Ijspeert

Supervisors:
A. Sproewitz,
A.Tuleu

EPFL Biorobotics Laboratory

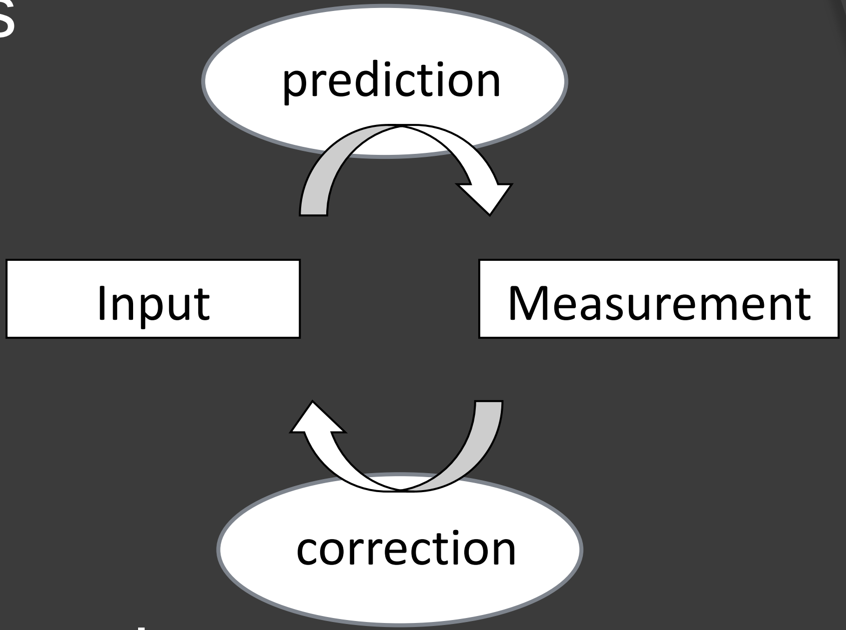
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Main objectives

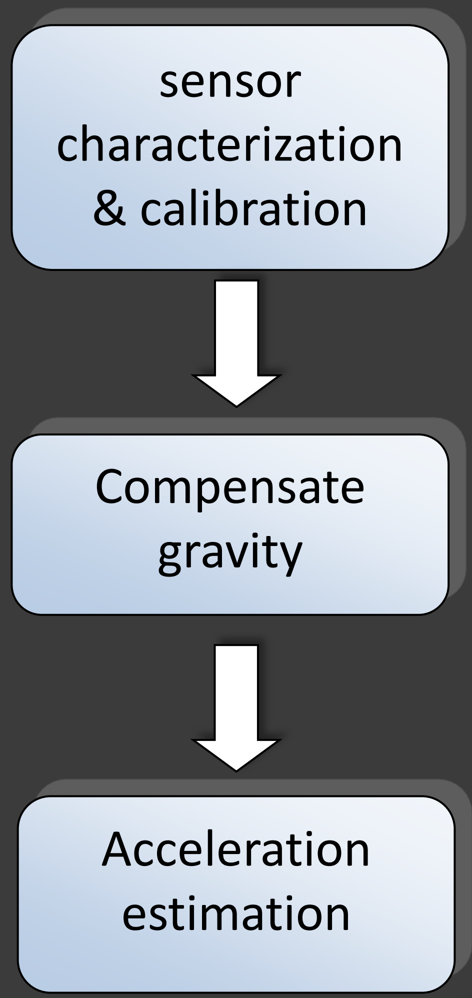
- Instantaneous pitch and roll angle estimate (of CoM)
- Instantaneous forward and upward acceleration (of CoM)

The Kalman filter: a reminder

- Multiple data sources
- Each data is noisy
- Final estimate more precise than any of the data

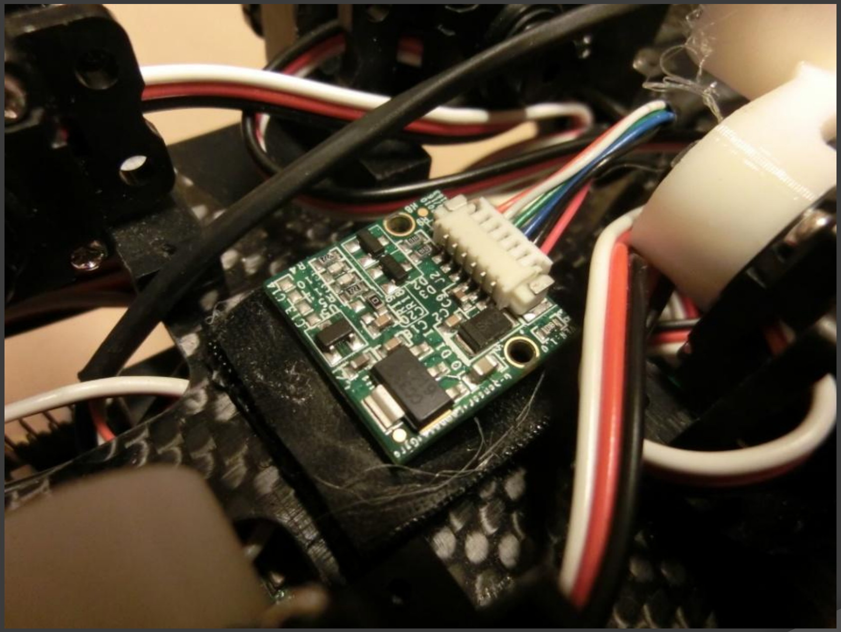


Project outline



Hardware

- 9 DOF IMU mounted near the CoM
  - 3 axis accelerometer
  - 3 axis gyro
  - 3 axis compass

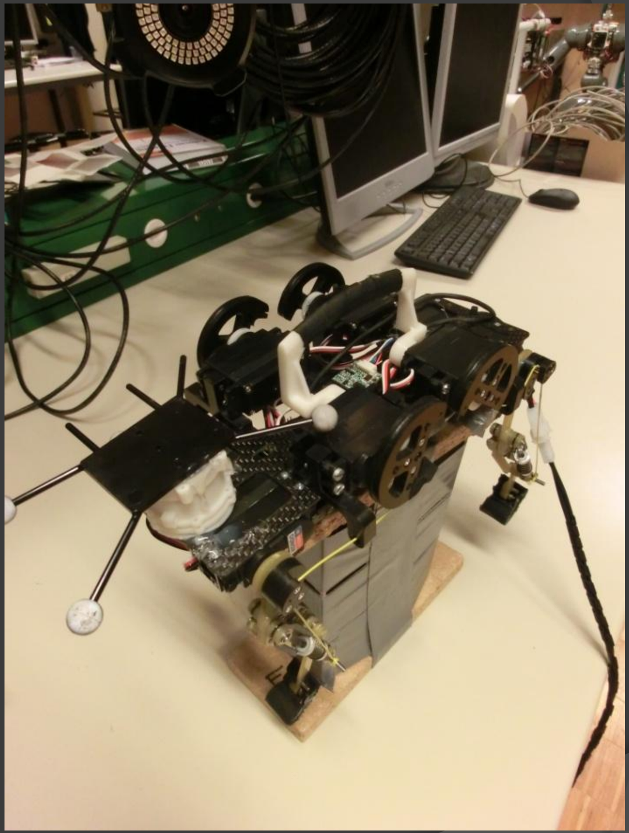


- MoCap
  - 12 camera system
  - "Absolute" reference

sensor calibration → Compensate gravity → Acceleration estimation

Hardware

- Quadruped robot: Cheetah- Cub
  - open loop control
  - no sensor input
  - IR markers on the back
  - trot gait

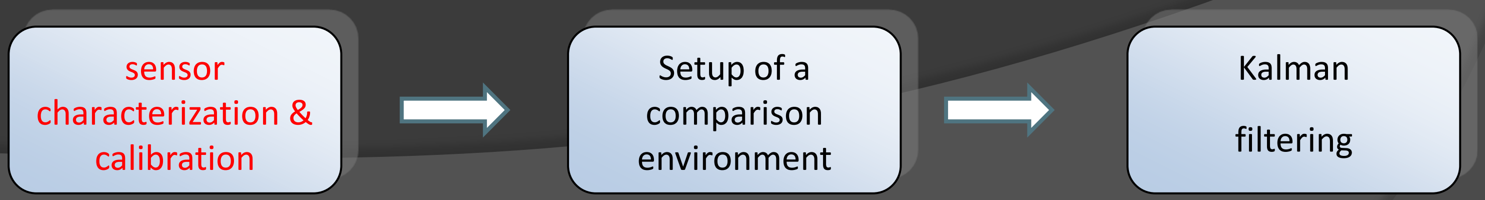


The Cheetah-cub

sensor calibration → Compensate gravity → Acceleration estimation

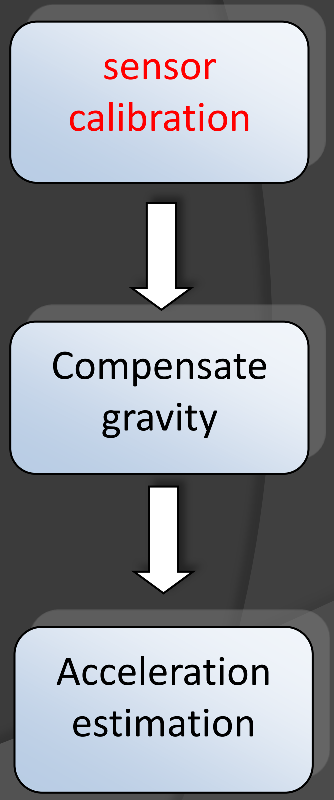
IMU Characterization

- Bias
- Scale factors
- Misalignment factors
- Temperature effect, Bias drift, etc.



IMU Characterization

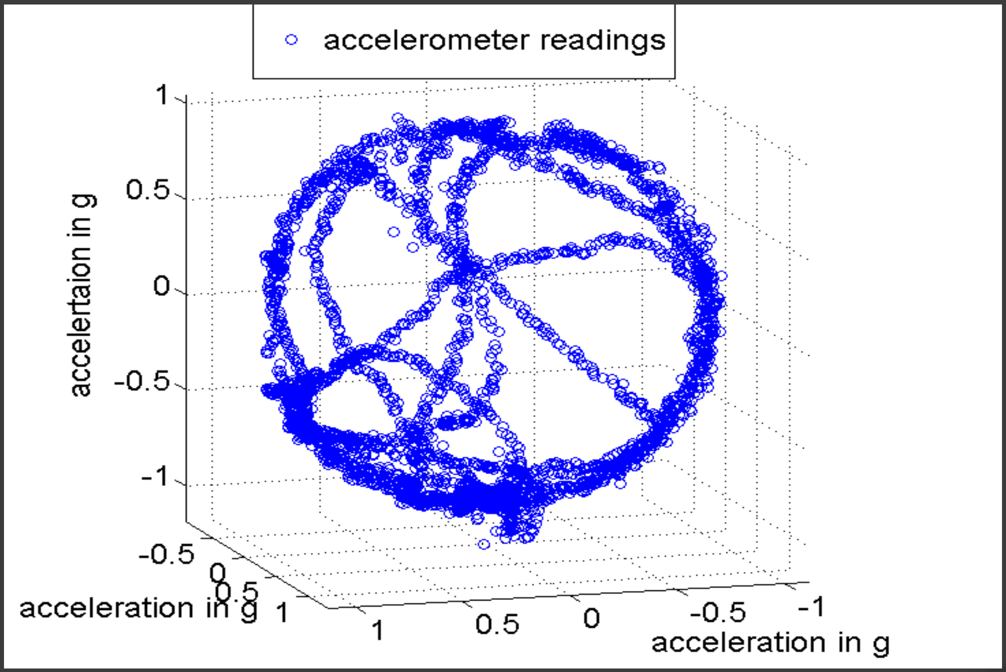
- 3 different “in-field” methods tested
- Ellipsoid method is preferred

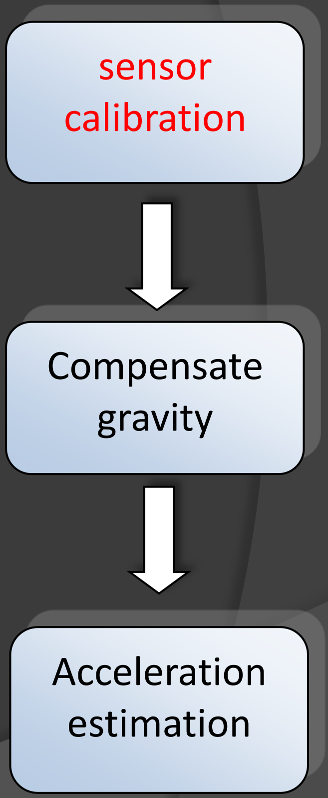


From : Timo Pylvänäinen, Automatic and adaptive calibration of 3D field sensors, Applied Mathematical Modelling, Volume 32, Issue 4, April 2008, Pages 575-587, ISSN 0307-904X, 10.1016/j.apm.2007.02.004
Implemention: Adrien Briod, EPFL

IMU Characterization

- 3 different methods tested
- Ellipsoid method is preferred

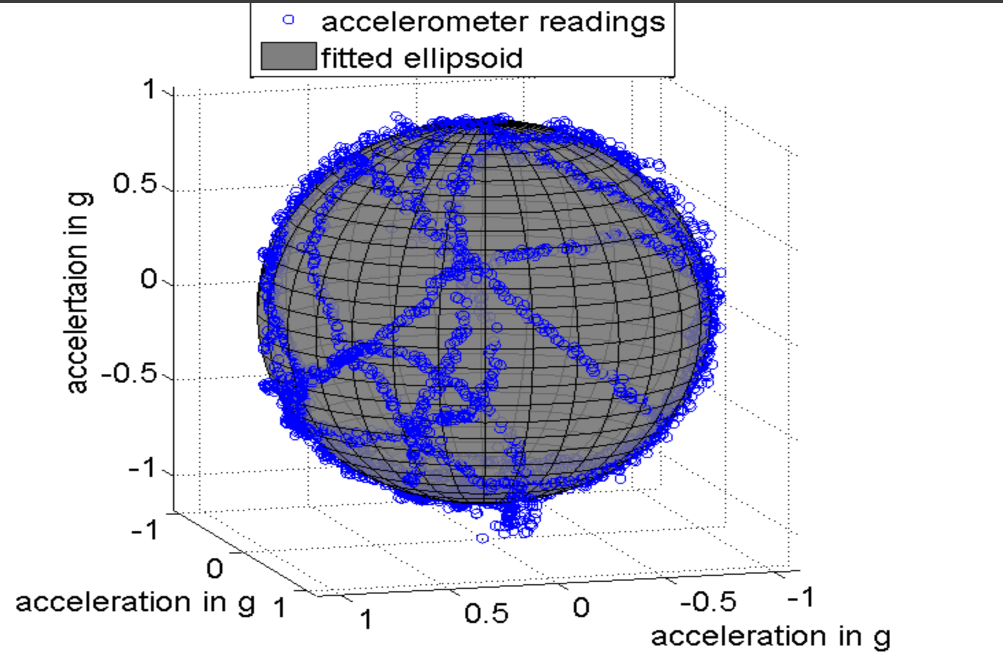




From : Timo Pylvänäinen, Automatic and adaptive calibration of 3D field sensors, Applied Mathematical Modelling, Volume 32, Issue 4, April 2008, Pages 575-587, ISSN 0307-904X, 10.1016/j.apm.2007.02.004
Implemention: Adrien Briod, EPFL

IMU Characterization

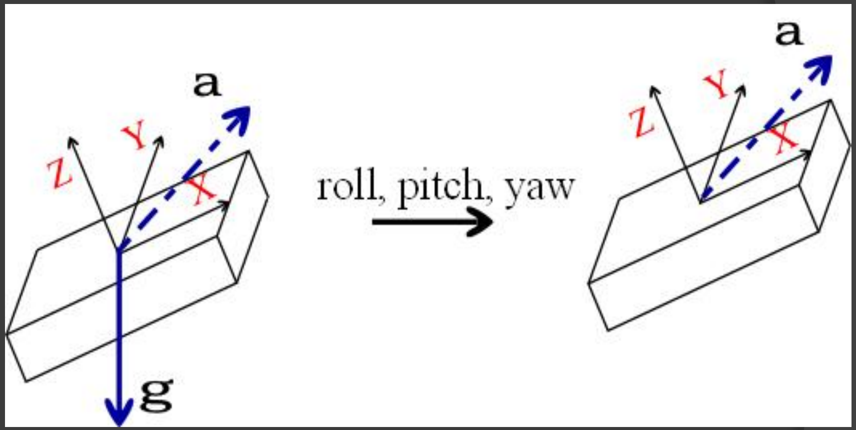
- 3 different methods tested
- Ellipsoid plot is preferred



sensor calibration → Compensate gravity → Acceleration estimation

Achieving gravity compensation

- 1st step: estimate orientation with a 1st Kalman filter



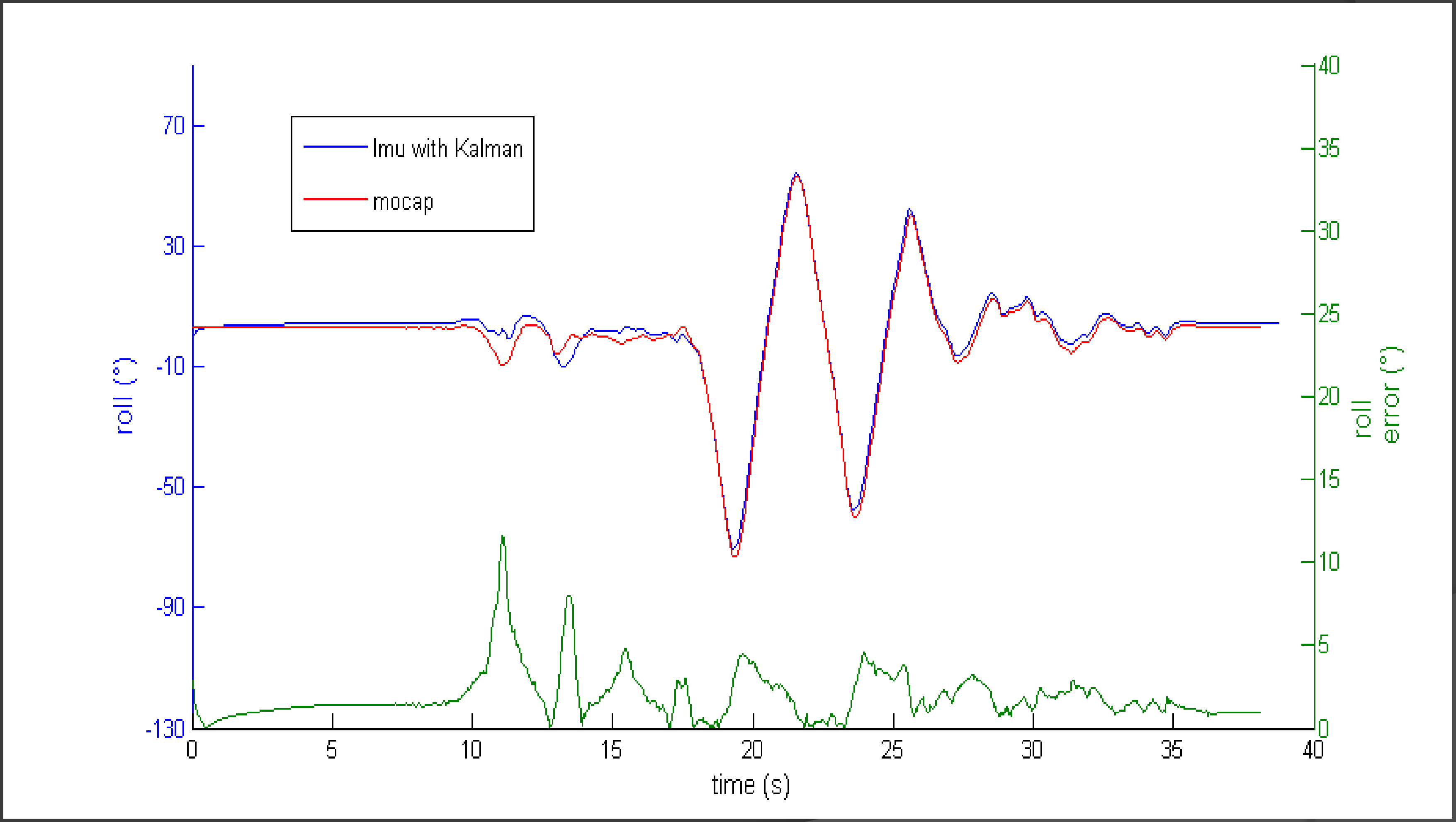
- 2d step: remove gravity
- 3rd step: compare validity of results with MoCap

Kalman filter for attitude estimation

- Kalman filter to combine sensor data
- Gyroscope for prediction
- Accelerometer for correction

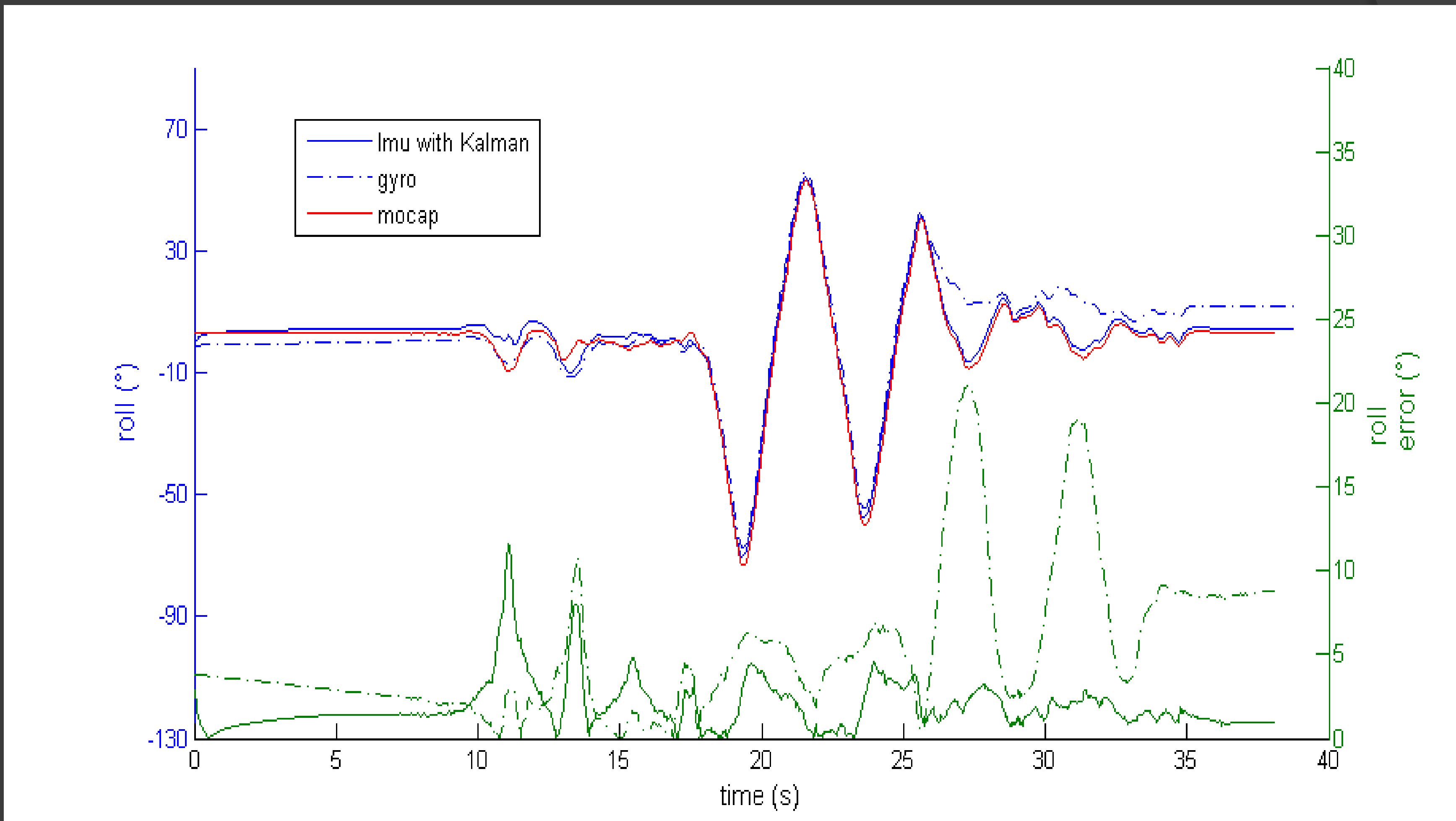
sensor calibration → Compensate gravity → Acceleration estimation

Comparing results with Mocap data



Overall RMSE: ~ 2.4°

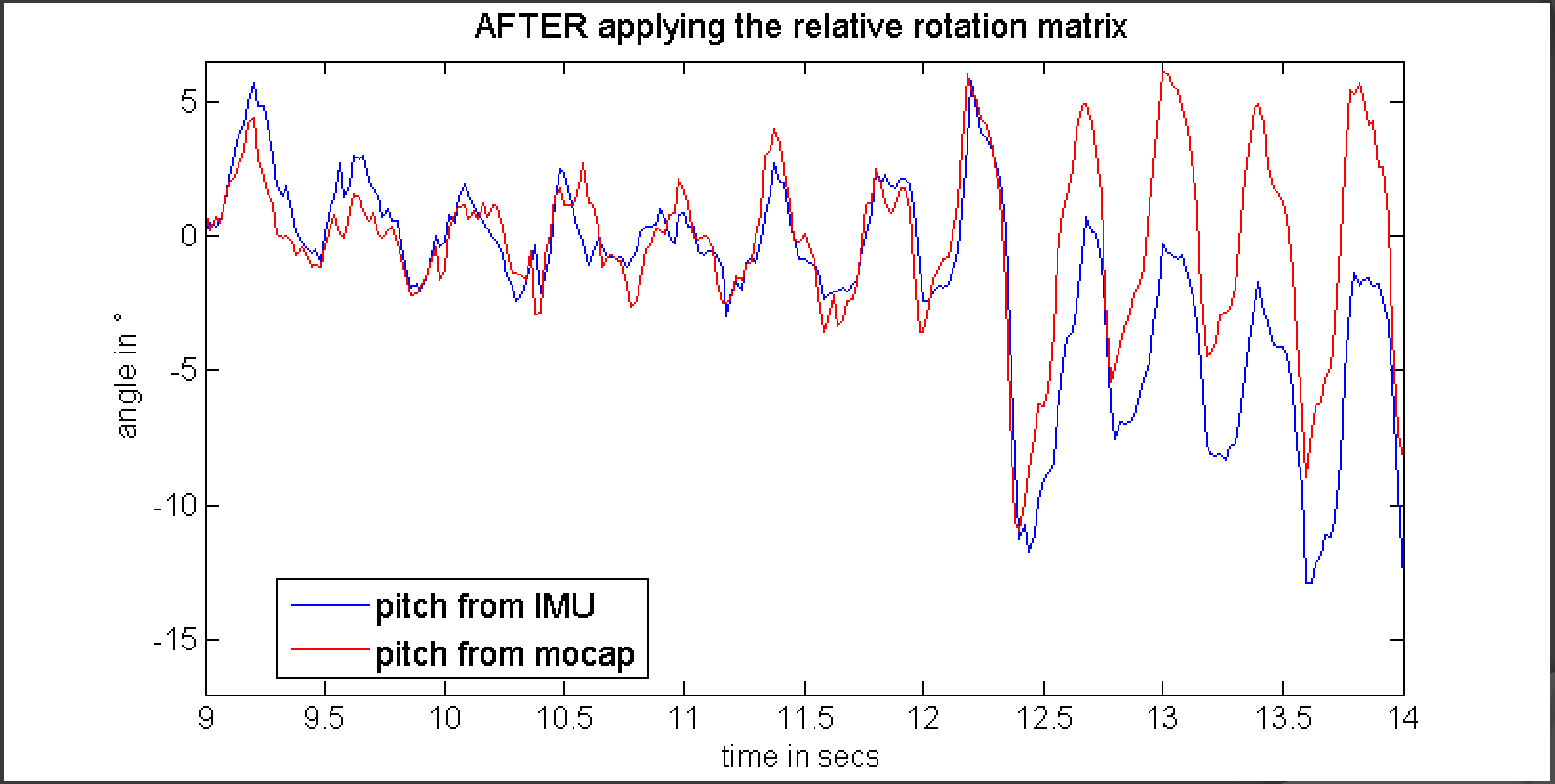
Comparing results with Mocap data



Overall RMSE: ~2.4° (IMU) and ~7.5° (only gyro)

Comparing results with Mocap data

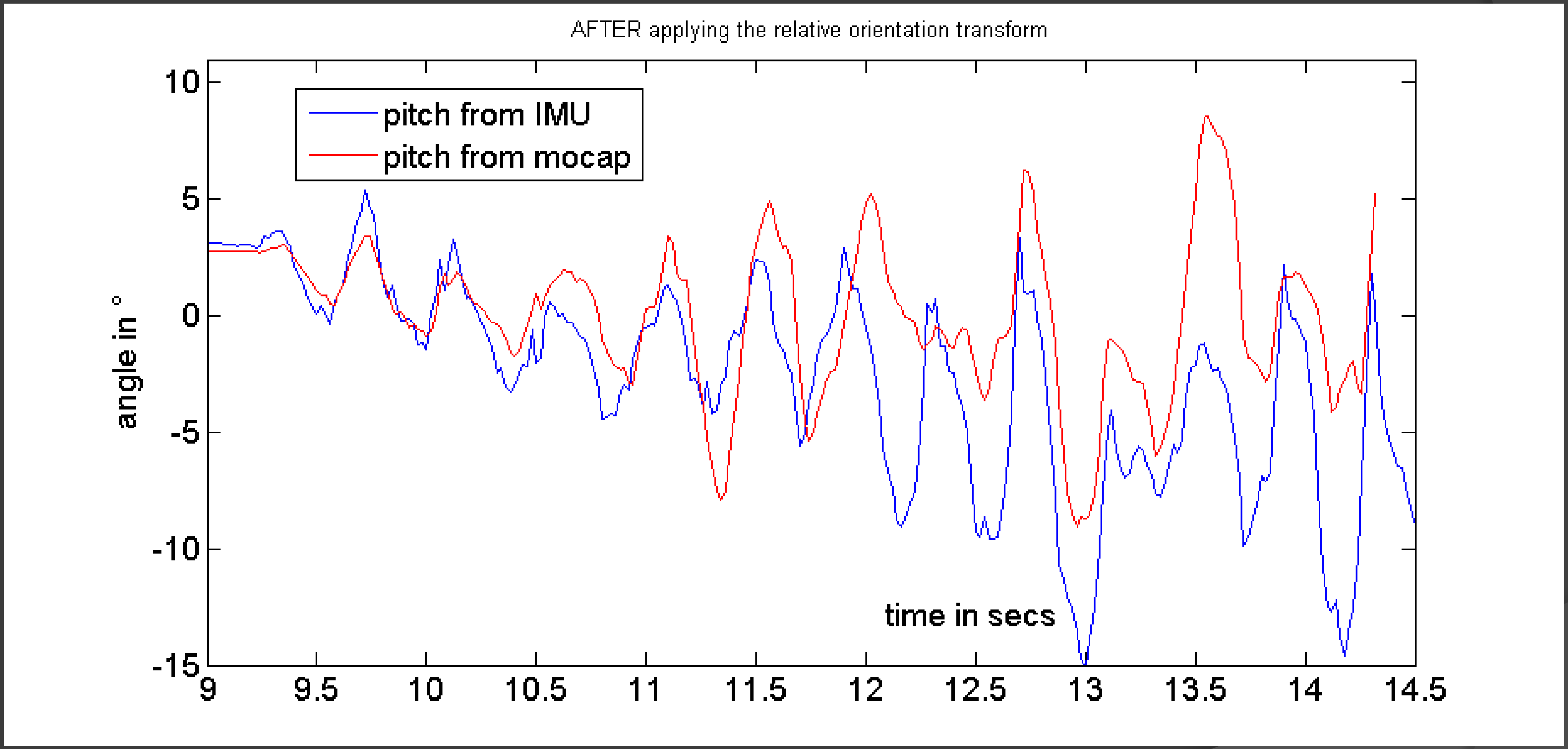
Pitch angle during a Cheeath run



sensor calibration → Compensate gravity → Acceleration estimation

Comparing results with Mocap data

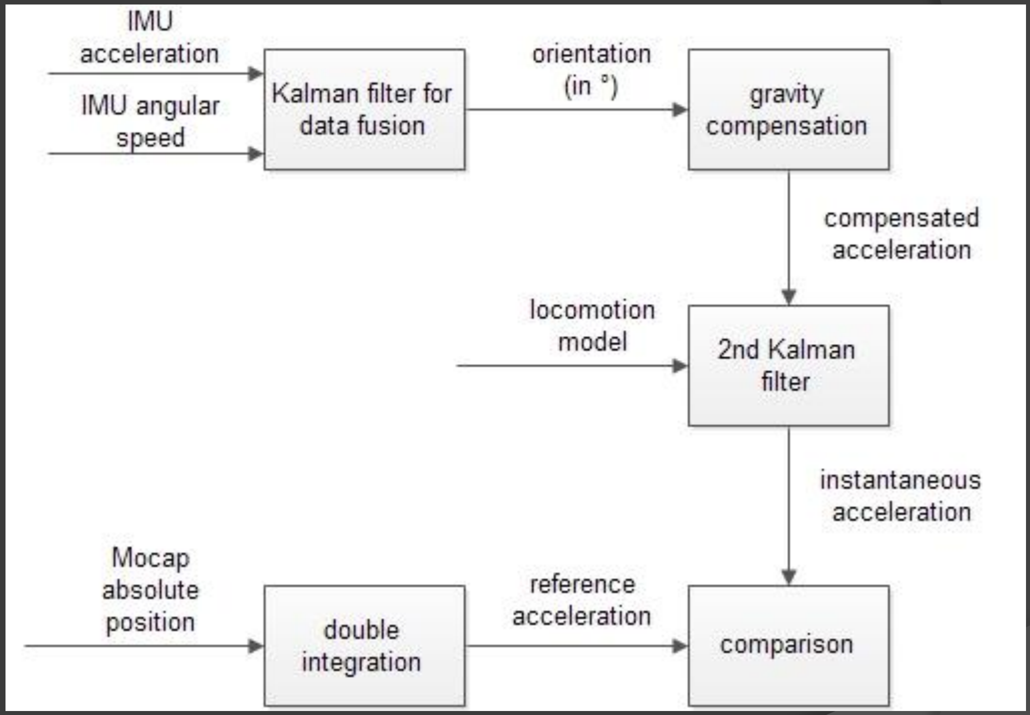
Pitch angle during a Cheeath run

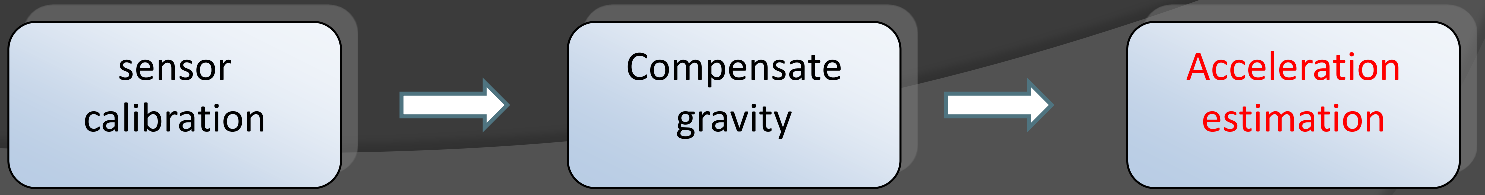


sensor calibration → Compensate gravity → Acceleration estimation

Acceleration estimation

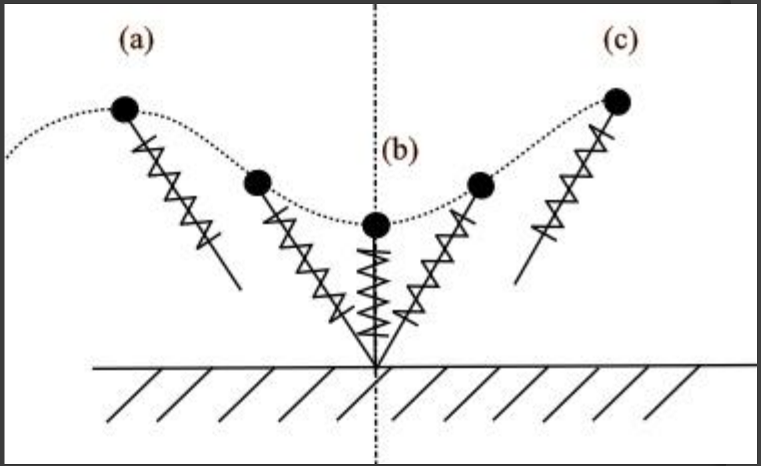
- 2nd Kalman filter
- Locomotion model for prediction
- Measured and compensated acceleration for correction





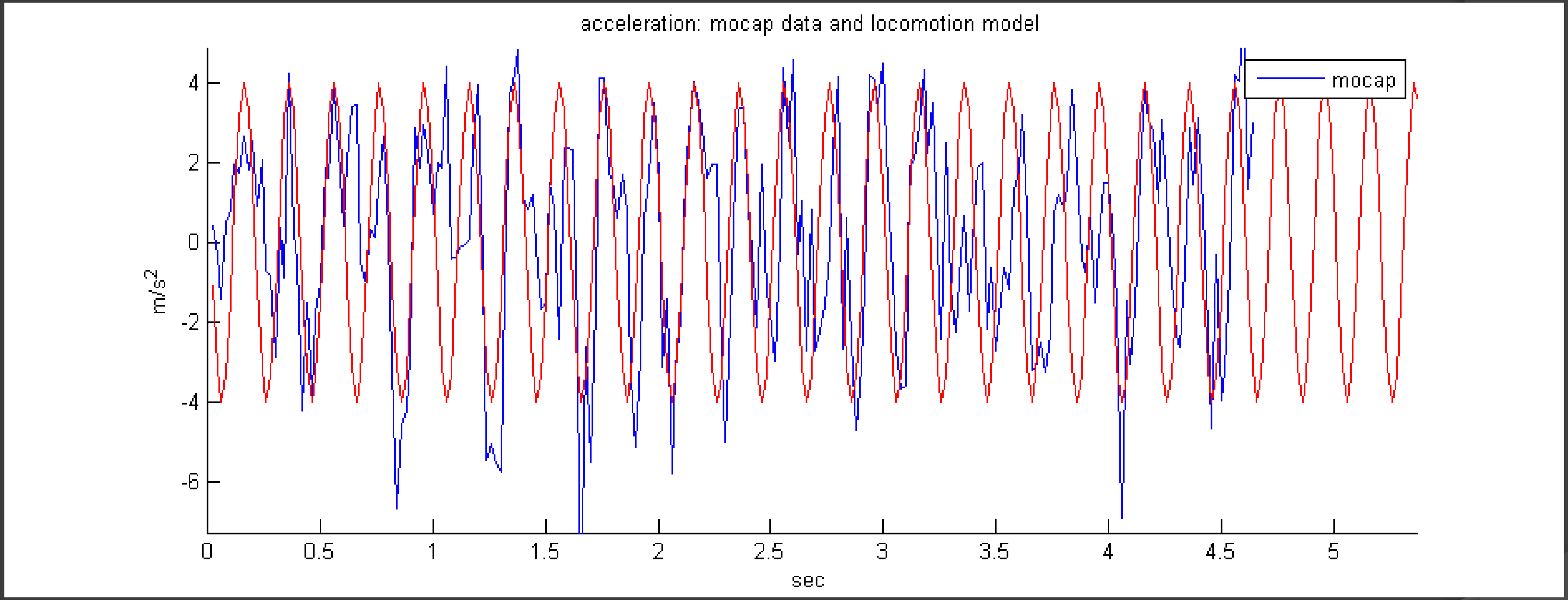
Locomotion model

- Imagine leg as simple spring
- Assume a symmetric and periodic trajectory
- Assume acceleration a sine wave with specific to the gait
- Amplitude and offset computed manually

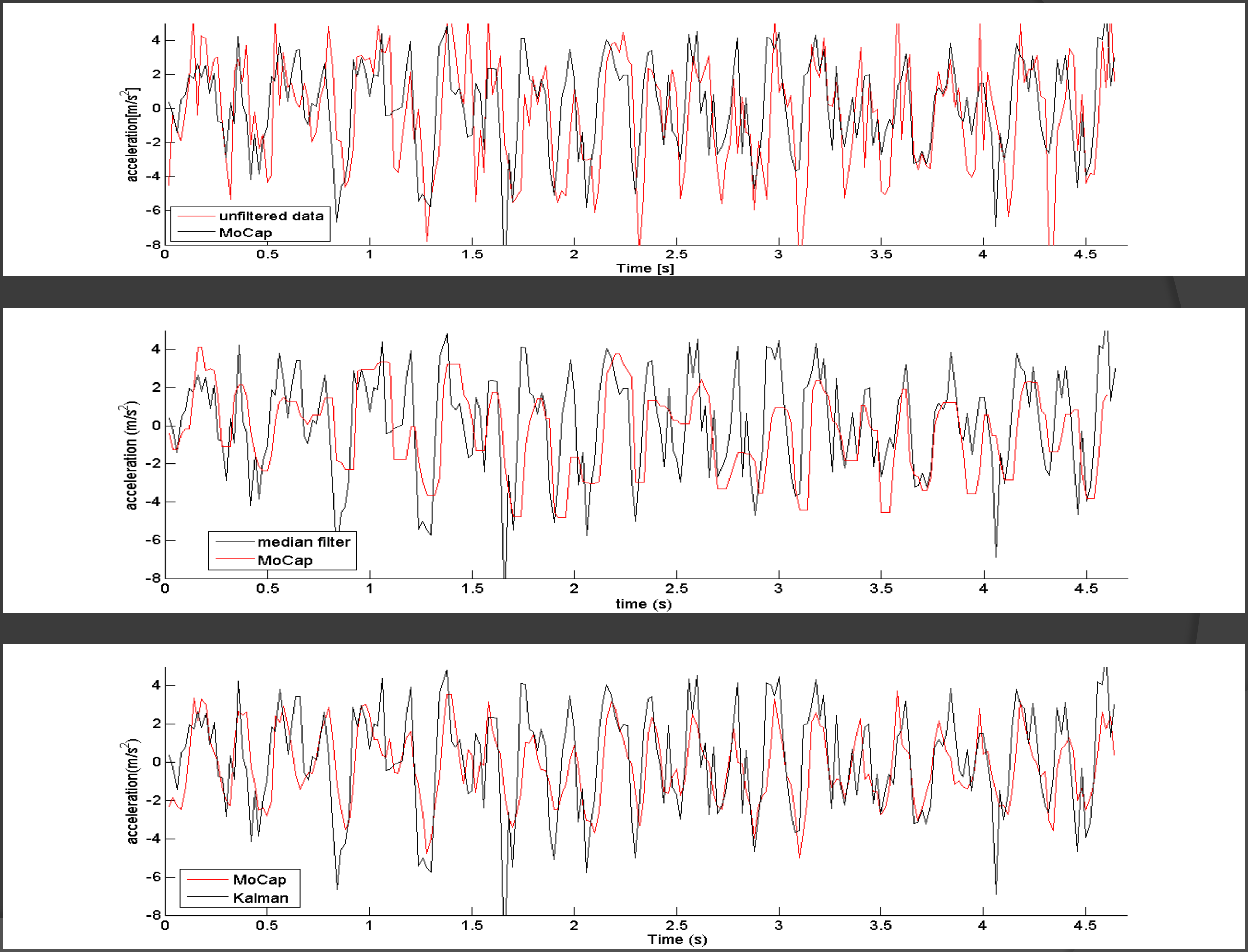


sensor calibration → Compensate gravity → Acceleration estimation

Locomotion model vs MoCap data



sensor calibration → Compensate gravity → Acceleration estimation


acceleration[m/s^2]
unfiltered data
MoCap
Time [s]
acceleration (m/s^2)
median filter
MoCap
time (s)
acceleration(m/s^2)
MoCap
Kalman
Time (s)

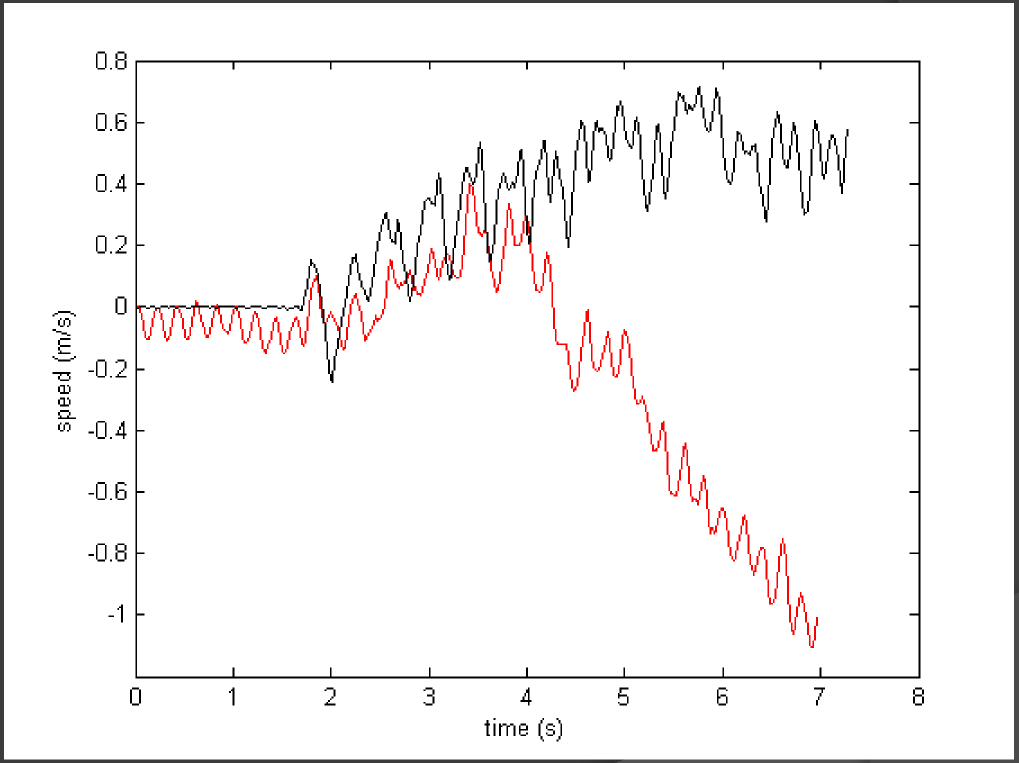
Results

Evaluation Method	Butterworth Low-pass	Median filter	Kalman filter
Correlation	0.28 ± 0.06	0.514 ± 0.04	**0.591 ± 0.04**
SNR(dB)	-1.21 ±0.4	0.71 ±0.4	**1.5 ± 0.5**
RMSE	2.93 ± 0.2	2.35 ± 0.2	**2.13 ± 0.02**

- Results are computed from 6 runs
- Kalman filter outperforms standard filters, but results are not outstanding

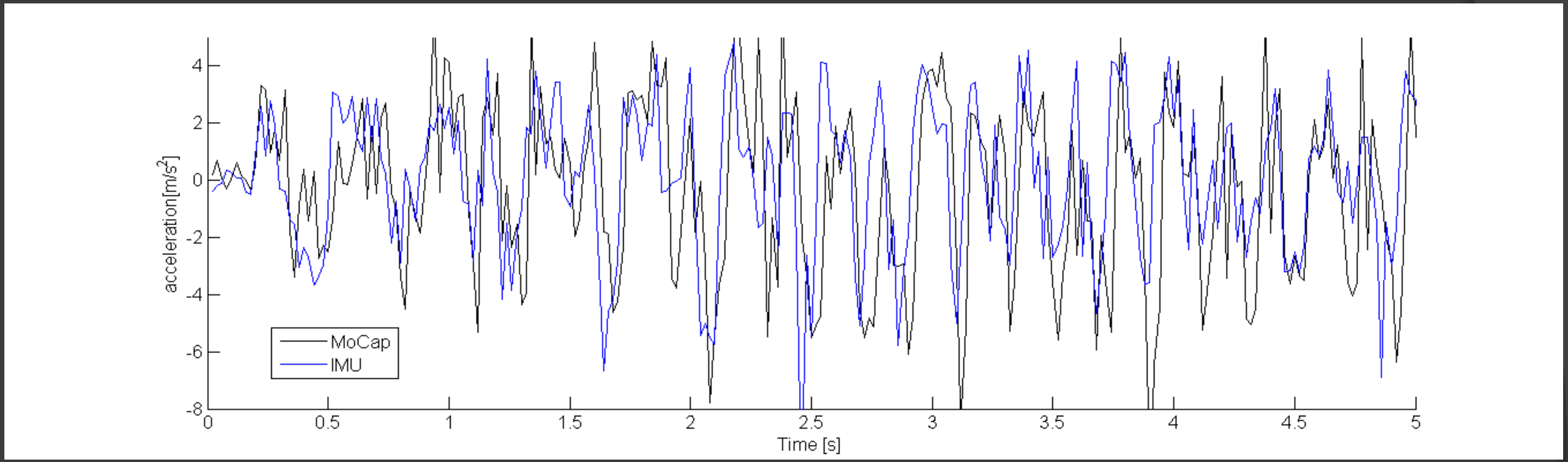
Speed estimation ?

- Mocap speed average ~ 0.4 m/s
- Bias, noise and other errors make a speed estimation from sole integration difficult

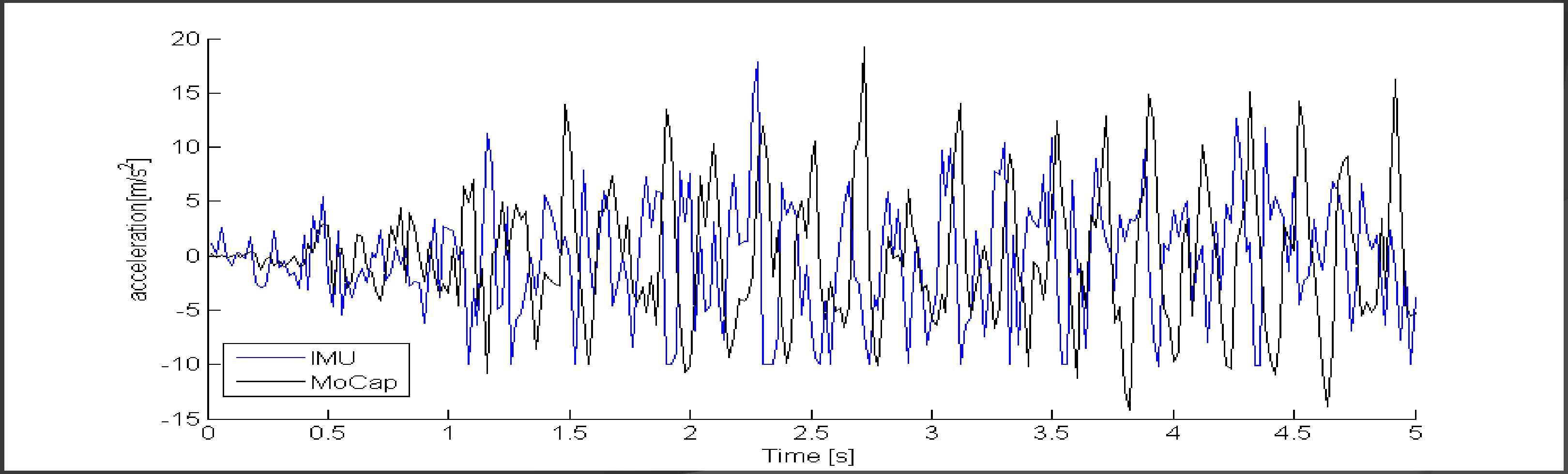


sensor calibration → Compensate gravity → Acceleration estimation

Acceleration estimation on Z axis ?



Acceleration on Y axis



Acceleration on Z axis

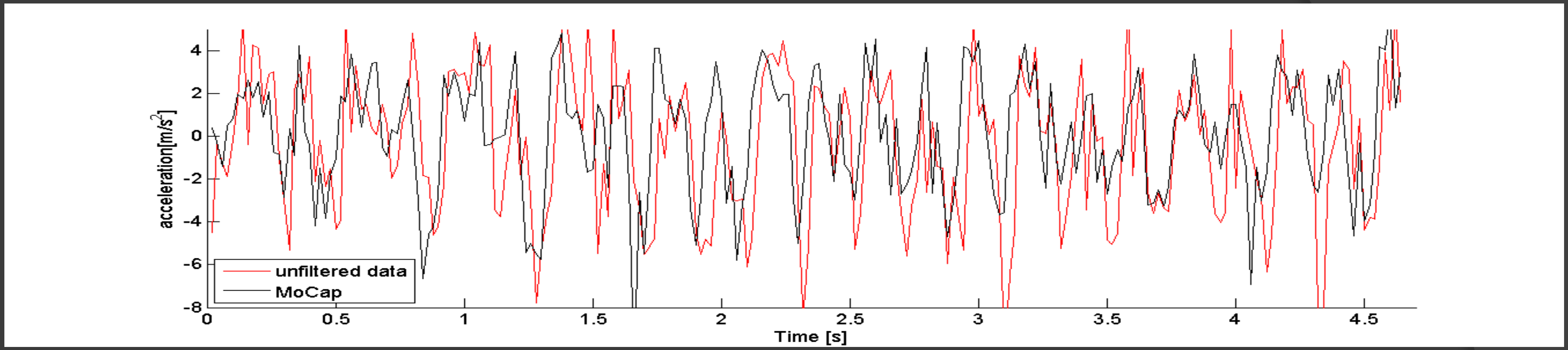
Conclusion

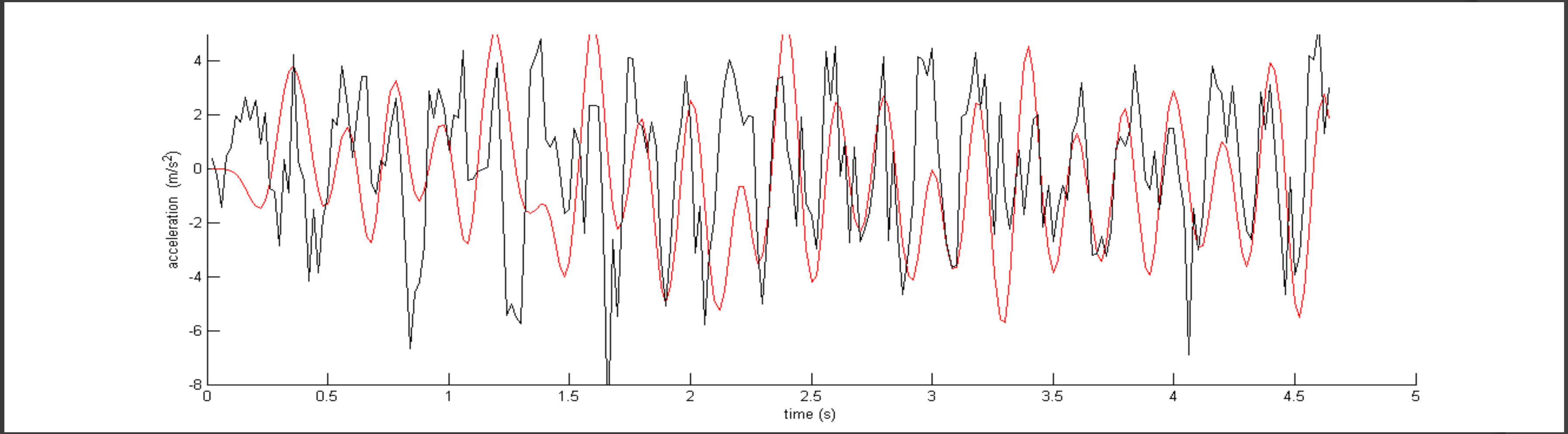
- Considerable time is required for IMU calibration
- Pitch and roll estimation OK with big amplitudes but bias too important at low amplitude movements
- Kalman filter for acceleration estimation less successful but outperforms standard filters
- A speed estimation is still not feasible
- More and better sensors and more elaborate Kalman design is required

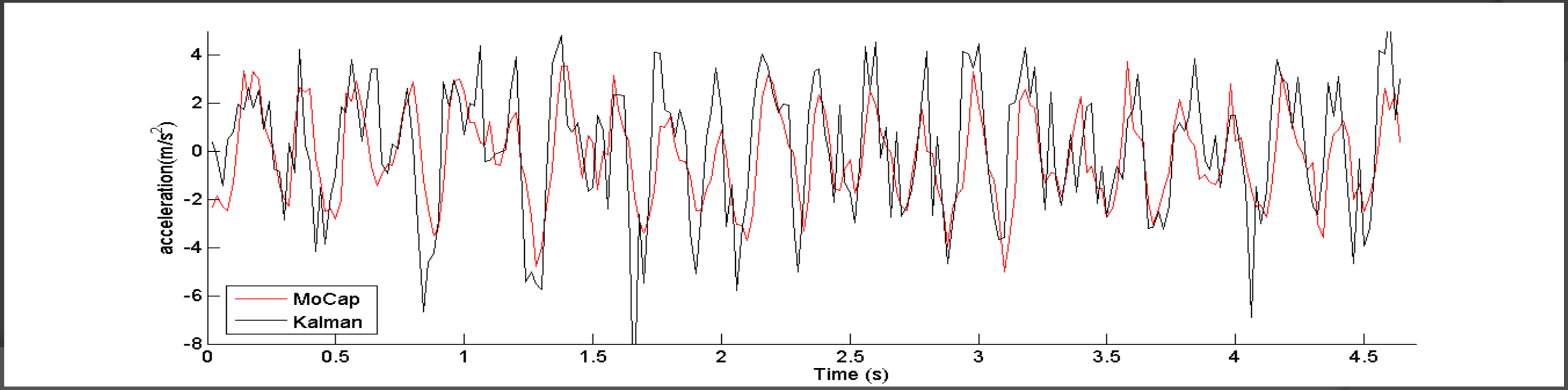
Thank you !

Annexe

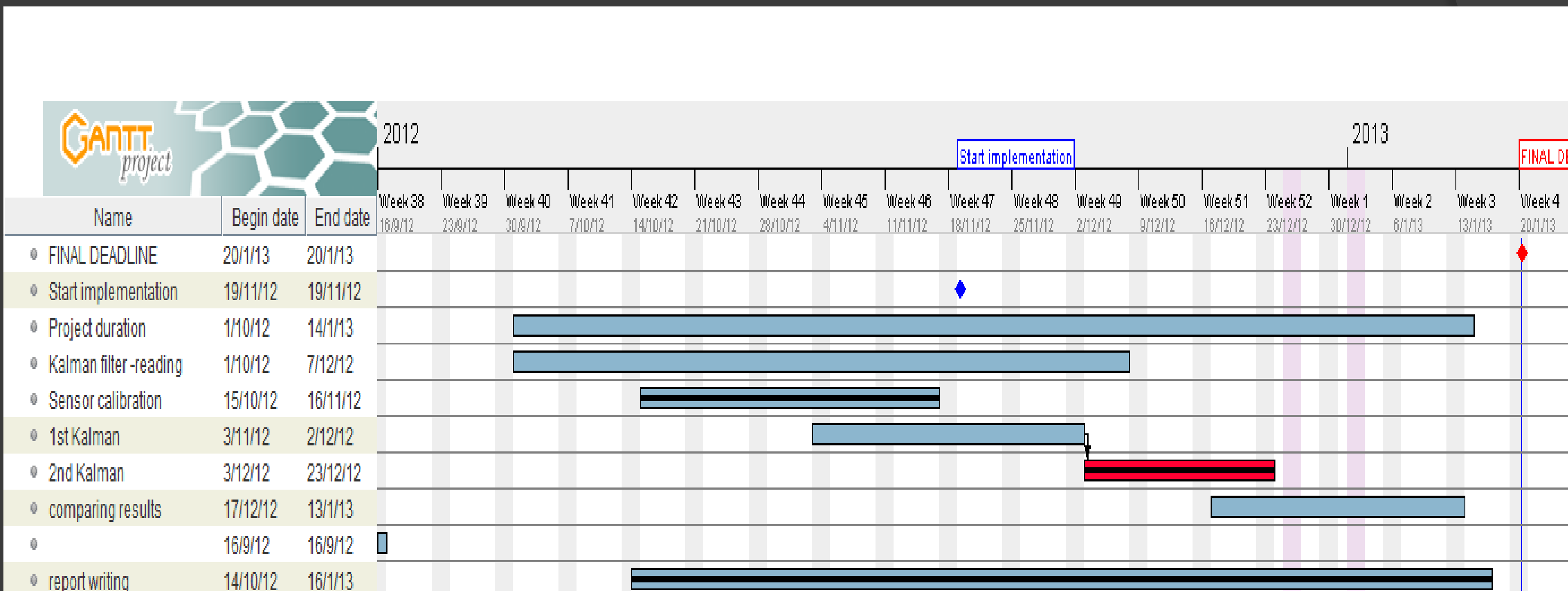
- Future Work:
  - Find better Kalman models, better hardware, calibrate better
  - Implemet in real time


acceleration[m/s^2]
Time [s]
unfiltered data
MoCap


acceleration (m/s^2)
time (s)


acceleration(m/s^2)
Time (s)
MoCap
Kalman

Gantt chart


GANTT project
2012
2013
Start implementation
FINAL DEA
Name
Begin date
End date
Week 38 16/9/12
Week 39 23/9/12
Week 40 30/9/12
Week 41 7/10/12
Week 42 14/10/12
Week 43 21/10/12
Week 44 28/10/12
Week 45 4/11/12
Week 46 11/11/12
Week 47 18/11/12
Week 48 25/11/12
Week 49 2/12/12
Week 50 9/12/12
Week 51 16/12/12
Week 52 23/12/12
Week 1 30/12/12
Week 2 6/1/13
Week 3 13/1/13
Week 4 20/1/13
FINAL DEADLINE 20/1/13 20/1/13
Start implementation 19/11/12 19/11/12
Project duration 1/10/12 14/1/13
Kalman filter -reading 1/10/12 7/12/12
Sensor calibration 15/10/12 16/11/12
1st Kalman 3/11/12 2/12/12
2nd Kalman 3/12/12 23/12/12
comparing results 17/12/12 13/1/13
16/9/12 16/9/12
report writing 14/10/12 16/1/13

SNR (signal/noise ratio)

- snr= 20*log10(norm(Y)/norm(Y-X))
- With Y = MoCap signal and X IMU signal