ST Pedestrian Dead Reckoning Algorithm

enables autonomous indoor positioning

Pedestrian
Dead
Reckoning

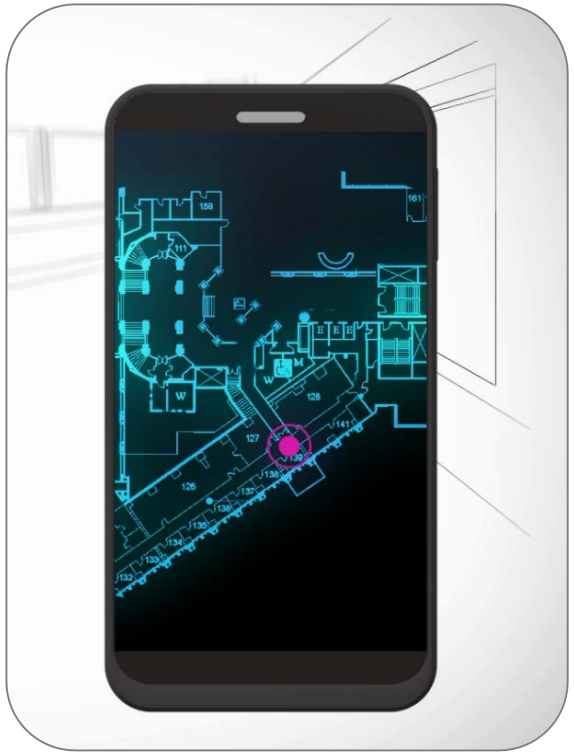
Components
of PDR

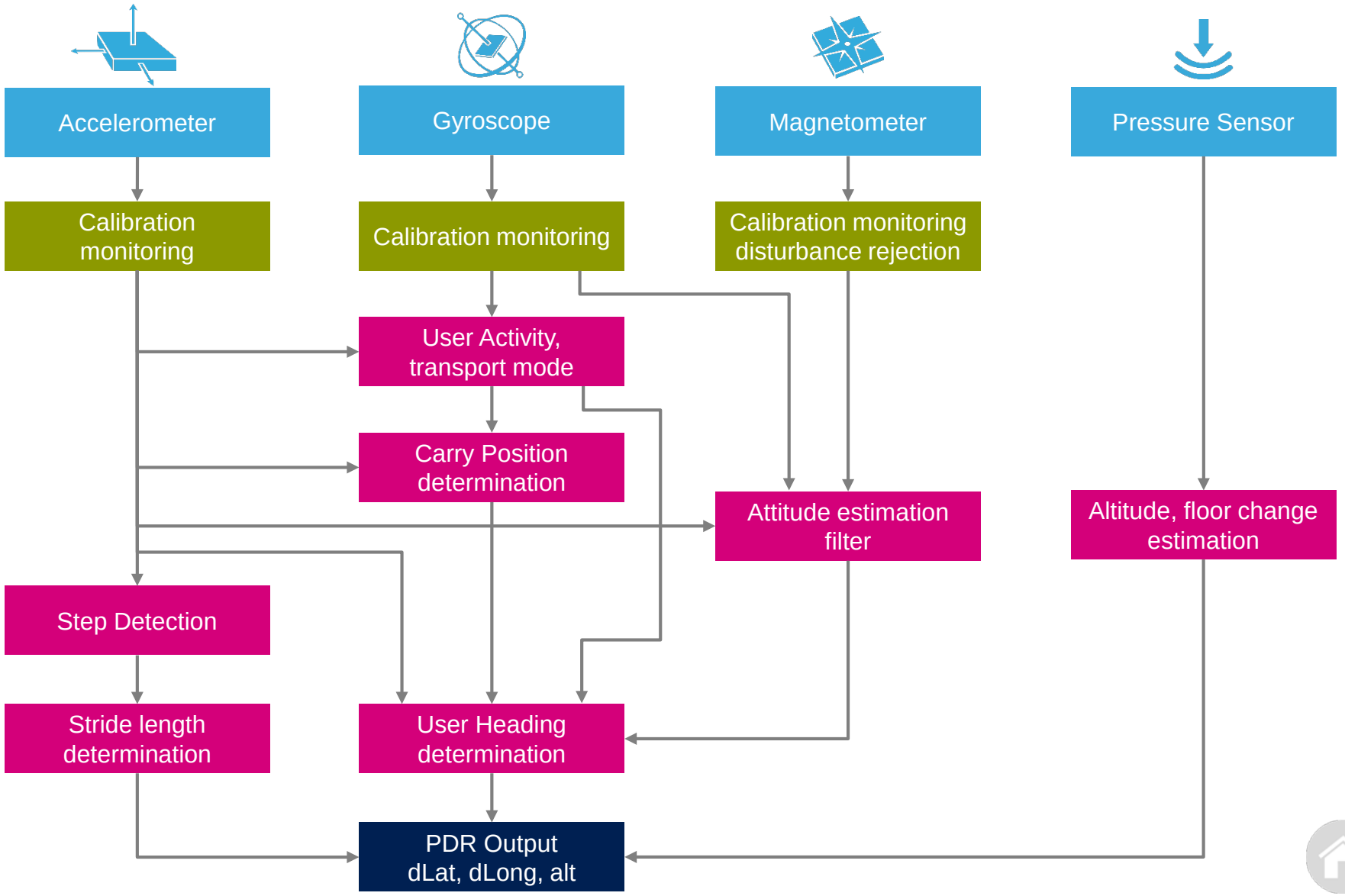
PDR
Phone Demo

PDR
Output

life.augmented

PDR Block Diagram





life.augmented

Components of PDR

- Sensor calibration:
  - Magnetometer, accelerometer
  - Continuous calibration monitoring
- Body placement (carry position) detection
- User activity mode detection
- Step detection, including false step rejection
- Variable stride length model and calibration
- Attitude Filter
- User walking direction determination
- Position update logic
- Error model

life.augmented

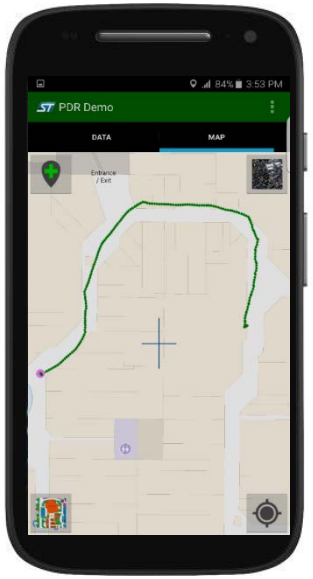
PDR Mobile Phone Demo

- Demo built using SensorTile and Android Smart-phone to display the real-time output
- PDR algorithms are implemented on STM32 microcontroller on SensorTile.
- PDR trajectory for the user path taken displayed using Android App
- Uncertainty estimates displayed in real-time

life.augmented

PDR Output

- Rich interface APIs are available to get additional information as per application requirements
  - sensor calibration parameters
  - step count
  - total distance
  - activity mode
  - carry position
  - attitude filter output



High Level Specifications

	Error (1 σ)
Step Detection	< 3%
Stride Length	~10% (without calibration), <5% (with calibration)
User Heading (including walk angle error)	10 deg

life.augmented

ST MEMS Sensors Enable

Contextual Awareness

Activity Recognition

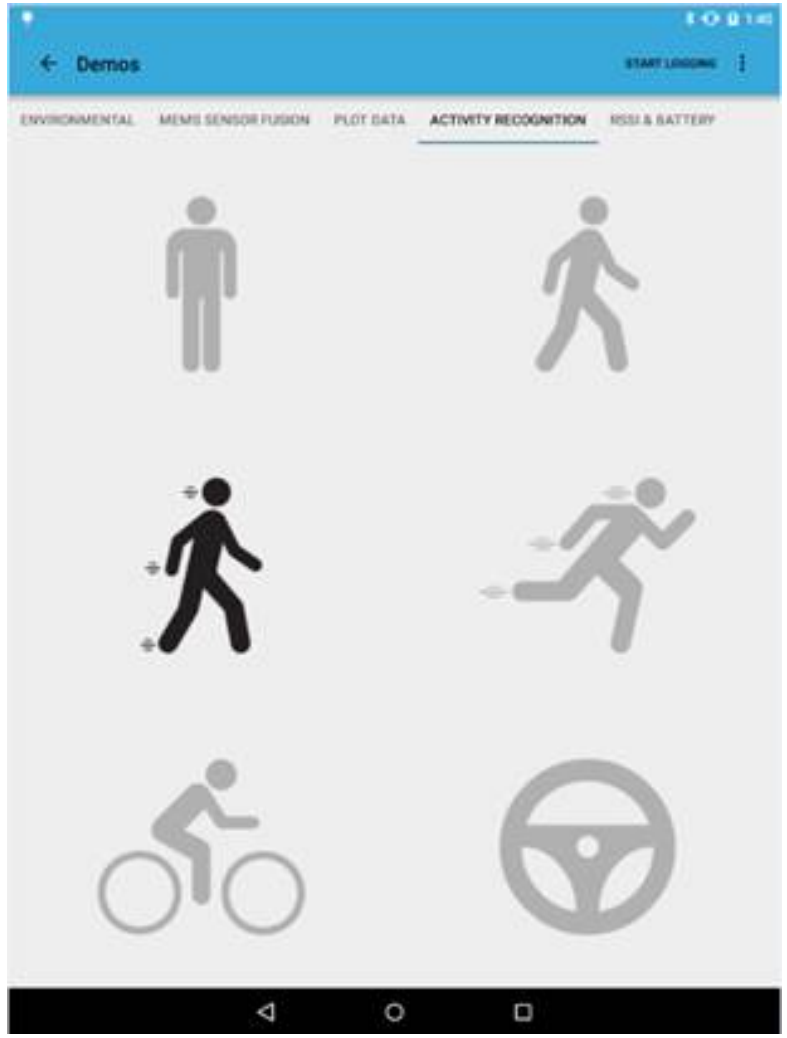
Carry Position Determination

Gesture Recognition

life.augmented

Activity Recognition Algorithm

- Detects Walking, Fast Walking, Running, Biking, Driving, Stationary
- Optimized for low power and always on experience
- Driven by 3-axis accelerometer.
- Uses Machine Learning models to achieve higher accuracy
- Available for download from Open.MEMS website



life.augmented

Activity Recognition Test results

- 682 data sets - 71 unique individuals, 48 hours of activity data
- Activities included (stationary, walking, fast walking, jogging, vehicle, bicycle) for different carry positions (body placement)
  - Pedestrian: Trouser pocket, in-hand, shirt pocket, in back pocket, near-the-head, ..
  - Vehicle: in cup-holder, in-shirt pocket, in-trouser pocket, ..
  - Bicycle: in-shirt pocket, in-trouser pocket

Actual Activity	Classified As						Detection Probability
	Stationary	Walking	Fast Walking	Jogging	Biking	Driving	
Stationary	16279	1	0	0	98	1431	91.41%
Walking	3	49030	51	9	483	25	98.85%
Fast Walking	0	116	3143	6	10	3	95.88%
Jogging	0	14	11	2781	8	2	98.76%
Biking	63	132	4	0	5292	633	86.41%
Driving	1113	6	1	0	436	7912	83.57%

Carry Position Determination

- Detects Carry Positions: In Hand, Near Head, Shirt Pocket, Trouser Pocket, On Desk, Arm Swing
- Optimized for low power and always on experience
- Driven primarily by accelerometer data
- Uses Machine Learning based models to achieve higher accuracy
- Available for download from Open.MEMS website

life.augmented

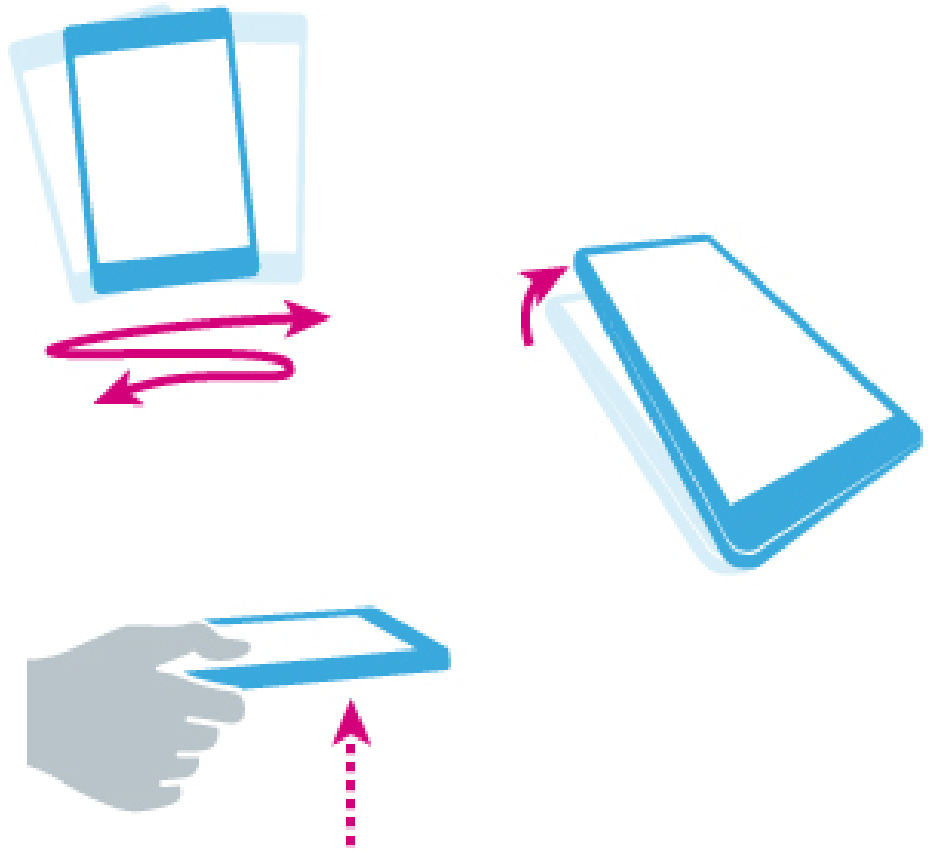
Carry Position Recognition Test Results

- Carry Positions included (in-hand, shirt pocket, trouser front pocket, trouser back pocket, near-the-head, on side arm swinging)
- For different activities like stationary, walking, fast walking

Carry Position	Detected as								Detection Probability
	Duration [sec]	Unknown	On Desk	In Hand	Near Head	Shirt Pocket	Trouser Pocket	Arm Swing	
On Desk	192	0.00	100.00	0.00	0.00	0.00	0.00	0.00	100.00%
In Hand	3694	0.00	7.23	92.66	0.00	0.00	0.08	0.03	92.66%
Near Head	975	0.41	14.77	0.00	84.62	0.00	0.10	0.10	84.62%
Shirt Pocket	1107	3.88	0.00	0.36	0.27	94.94	0.36	0.18	94.94%
Trouser Pocket	5287	0.00	0.00	0.00	0.11	0.08	99.79	0.02	99.79%
Arm Swing	1444	0.14	0.00	0.00	0.00	0.00	0.48	99.38	99.38%

Gesture Recognition

- Detects gestures such as Glance, Pick-up, Wake-up
- Uses Only 3 Axis Accelerometer
- Machine learning algorithms used for the feature
- Designed for always-on operation
- Available for download from Open.MEMS website



life.augmented