OpenShoe

Past, current, and future work

KTH Elektro-
och systemteknik

By John-Olof Nilsson

Indo-Swedish navigation project workshop

120321 Bangalore

Outline

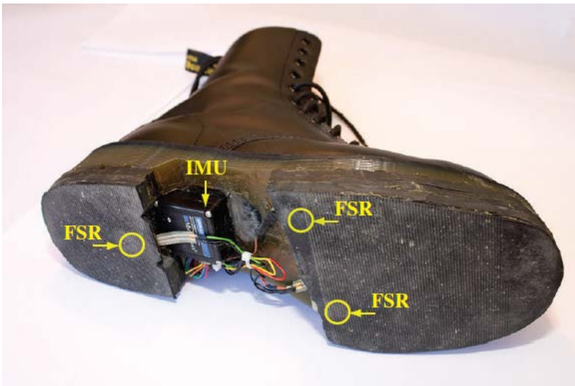
- Past work
- Current status
- Current work
- Future work

KTH Elektro-
och systemteknik

Past work

- Pre 2008-2011 – Buildup of experience
- Summer 2011 – OpenShoe feasibility study
- Fall/winter 2011 – OpenShoe implementation
- Spring 2012 – Evaluation/testing + publication

**KTH Elektro-
och systemteknik**



KTH Elektro-
och systemteknik

Current status

Current status

- First publication will be presented on april 23th

Foot-mounted INS for Everybody – An Open-source Embedded Implementation

John-Olof Nilsson[1], Isaac Skog[1], Peter Händel[1], and K.V.S. Hari[2]
[1]Signal Processing Lab, ACCESS Linnaeus Centre, KTH Royal Institute of Technology, Stockholm, Sweden
[2]Statistical Signal Processing Lab, Department of ECE, Indian Institute of Science (IISc), Bangalore, India

***Abstract*—We present an open-source, realtime, embedded implementation of a foot-mounted, zero-velocity-update-aided inertial navigation system. The implementation includes both hardware design and software, uses off-the-shelf components and assembly methods, and features a standard USB interface. The software is written in C and can easily be modified to run user implemented algorithms. The hardware design and the software are released under permissive open-source licenses and production files, source code, documentation, and further resources are available at www.openshoe.org. The reproduction cost for a single unit is below $800, with the inertial measurement unit making up the bulk ($700). The form factor of the implementation is small enough for it to be integrated in the sole of a shoe. Our performance evaluation of the system shows a position error for a 50 [m], straight-line trajectory of ±0.5[m] (1σ bound) perpendicular to and ±0.15 [m] along the trajectory. For a closed-loop, 80[m], symmetric, figure-of-eight trajectory the error is ±0.25 [m] in all directions.**

Fig. 1: Shoes with units of the presented foot-mounted INS implementation integrated in the heels and cabling with USB connectors at the shoe shafts.

I. Introduction

Foot-mounted inertial navigation is not rocket science and the path from a theoretical algorithm to a realtime embedded implementation might seem deceptively short. However, our experience is that the difficulties of developing a well-performing foot-mounted inertial navigation system (INS), yet surmountable, are easily underestimated. The number of software and hardware components that need to work together is large enough such that getting them to work in harmony is not trivial. Further, obtaining sufficient computational power and versatility on an embedded processor and sufficiently low computational cost of the filter algorithms require care in platform selection and algorithm implementation. Moreover, attaining a form factor and mechanical durability, such that the system can be integrated into the sole of a shoe and still be able to use off-the-shelf components and easily available construction methods, is challenging. Few solutions exist on the market or in the literature with even fewer solutions that provide the documentation and modifiability desired by researchers.

Therefore, to give researchers, teachers, and system designers a shortcut to a working foot-mounted INS implementation suitable for further research, education, and rapid prototyping, and usable as a component in larger pedestrian navigation systems, we present an open-source, embedded, foot-mounted INS implementation containing both hardware design and software/algorithm implementations. Our hope is that such an implementation will save time, sweat, and tears for navigation researchers as well as facilitate the use of the technology by researchers not specialized in aided INS, e.g. in fields such as biomedical engineering, behavioral science, and ubiquitous computing. The value of the embedded implementation also lies in its modularity and in its small weight, bulk, and price in comparison with the typical sensor-plus-laptop research systems. These properties alleviate the work of integrating the foot-mounted INS in larger realtime navigation systems, and make it feasible to equip a larger number of users with foot-mounted INS units for field performance tests and cooperative navigation studies.

The presented system is a zero-velocity-update (ZUPT)-aided INS built from off-the-shelf components and with a standard USB interface. The filtering is implemented on a microcontroller (μC) fitted with an OEM inertial measurement unit (IMU) in a casing, which, in turn, is integrated in the sole of a shoe. Figure 1 shows a pair of shoes with the system units integrated in the heel. The system can easily be configured to run any user implemented algorithm executable in $\sim 10^5$ clock cycles per time update. The software and hardware design are released under permissive open-source licenses. Available resources include ready-to-compile, as well as pre-compiled source code, hardware production files, a Matlab system interface, and documentation, with everything from electronic schematics and CAD-drawings of casing components to code documentation and installation instructions. All resources can be downloaded from **www.openshoe.org**.

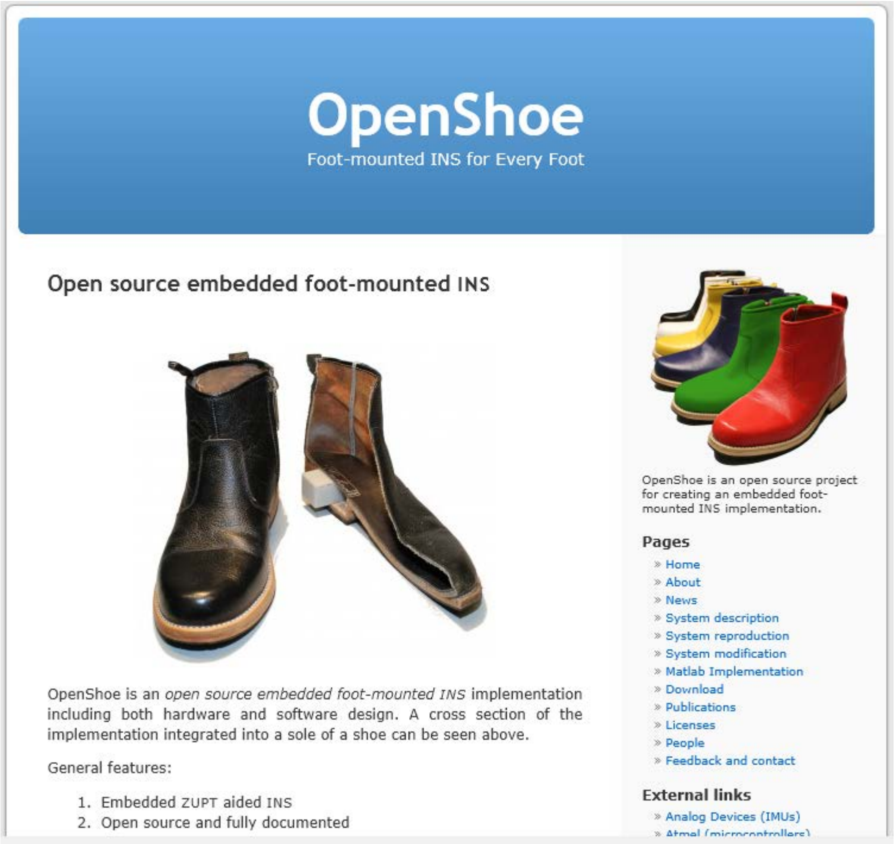
II. Foot-Mounted INS

A foot-mounted INS is simply an INS mounted on the foot combined with additional filtering that improves the accuracy

KTH Elektro- och systemteknik

Current status

- Homepage presentation on a reasonable level



KTH Elektro-
och systemteknik

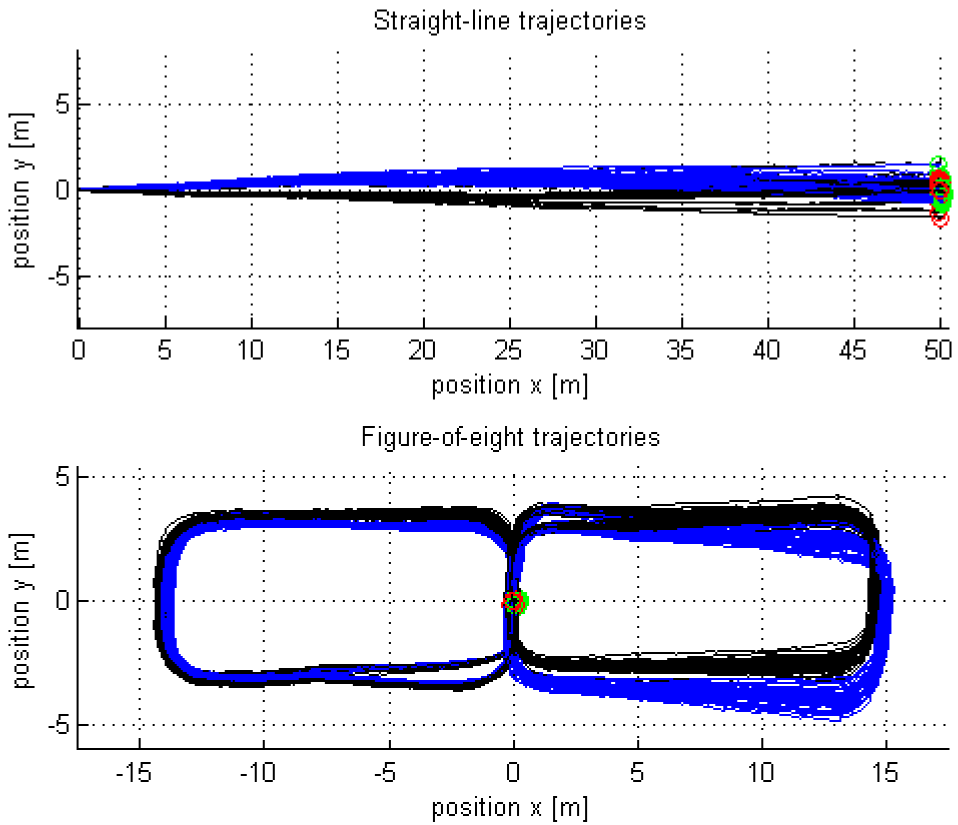
Current status

- 6 complete units (4+2)
- 20 PCBs manufactured
- 5 units (-IMU) + 5 PCB will be produced

KTH Elektro-
och systemteknik

Current status

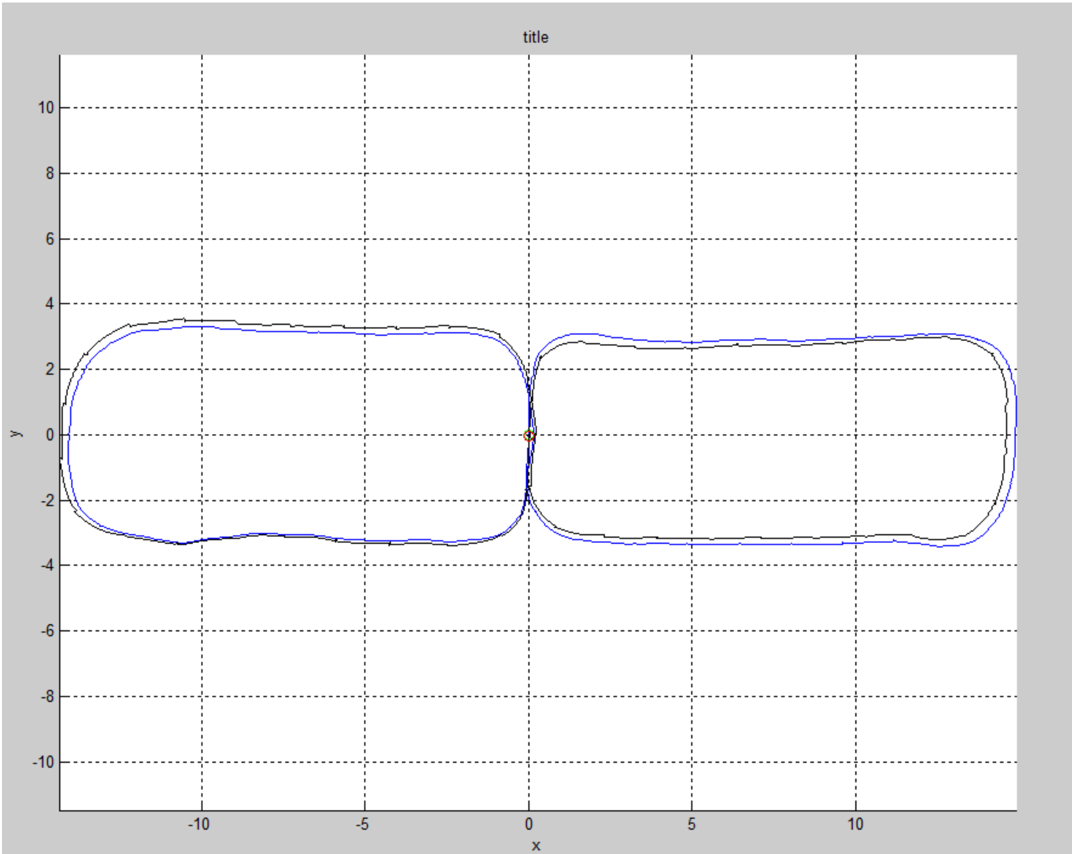
- Basic performance evaluation done



KTH Elektro-
och systemteknik

Current status

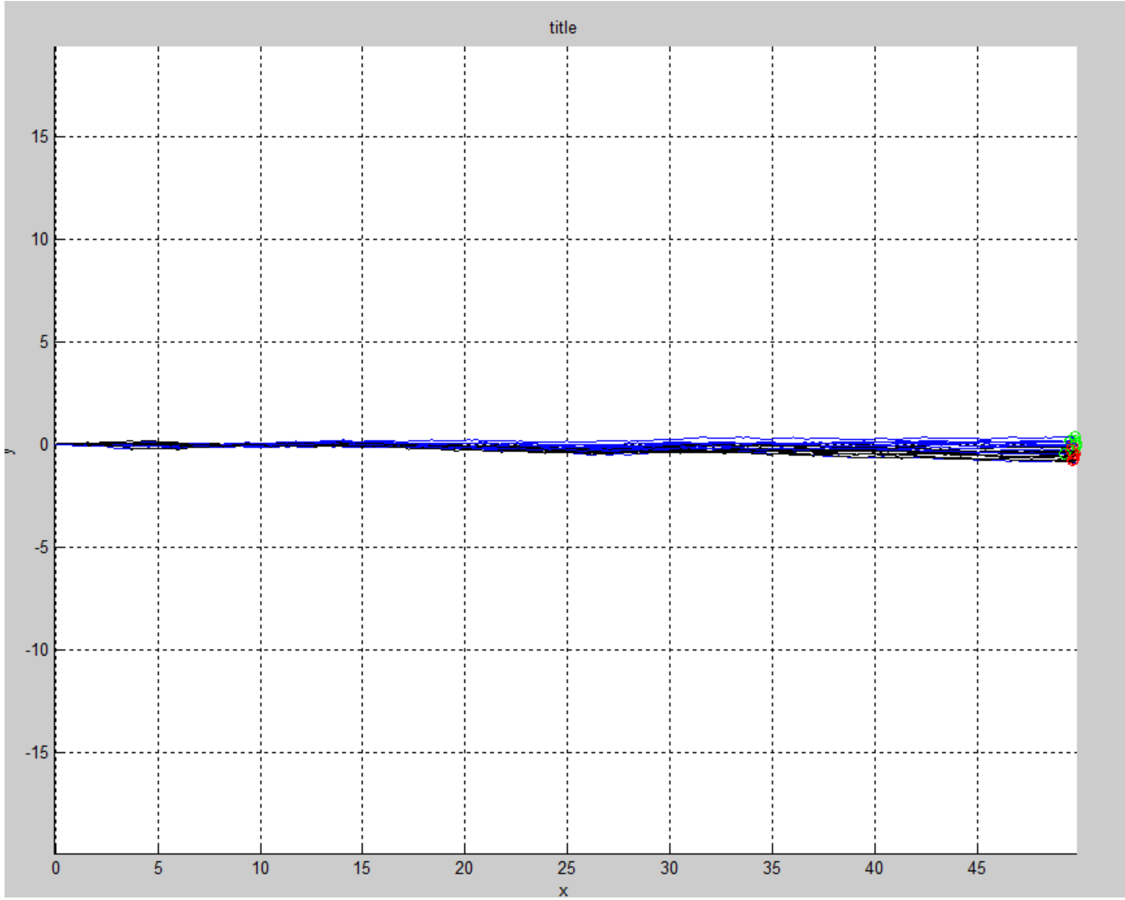
- Synced C/C++ interface several IMUs



KTH Elektro-
och systemteknik

Current status

- We can run!



KTH Elektro-
och systemteknik

Summary – Curren status

- Basic presentation of the system is available
- We have gained basic experience and confidence in the hardware – It works!
- Basic evaluation of the system shows similar performance to previous off-line implementations (which were 3-4 times as expensive)

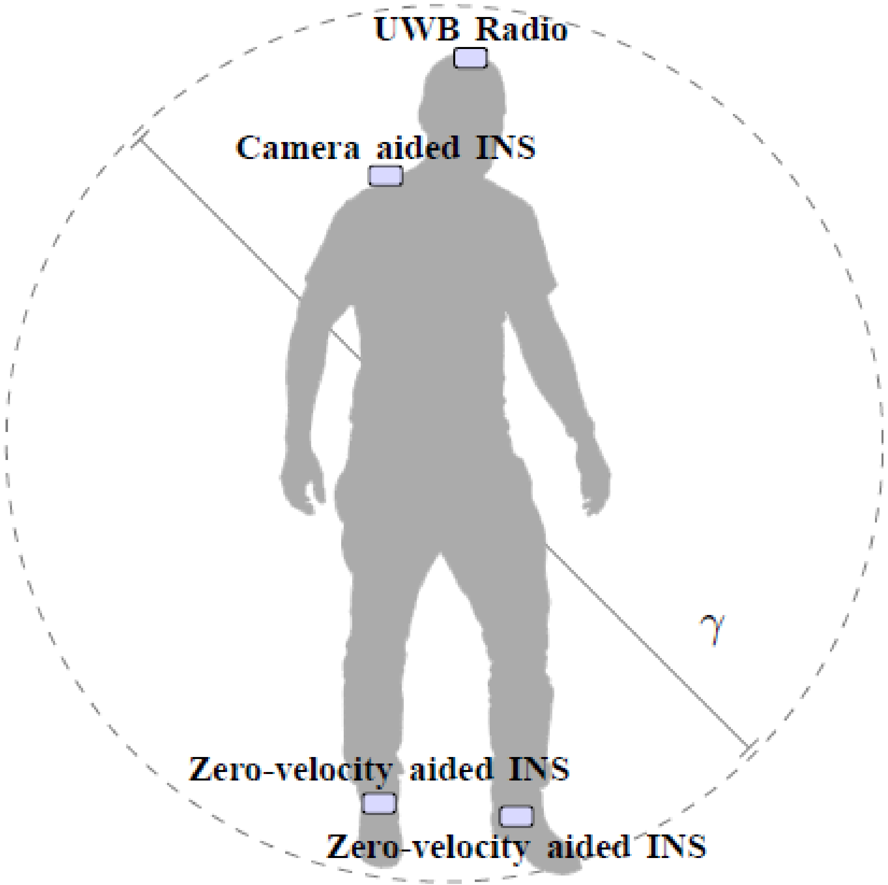
KTH Elektro-
och systemteknik

Current work

KTH Elektro-
och systemteknik

Current work

- Constrained filtering



KTH Elektro-
och systemteknik

Current work

- Upper/lower layer filtering separation

Instead of

$$\begin{bmatrix}\delta\mathbf{p}_k\\ \delta\mathbf{v}_k\\ \delta\boldsymbol{\theta}_k\end{bmatrix} = \begin{bmatrix}\delta\mathbf{p}_{k-1}\\ \delta\mathbf{v}_{k-1}\\ \delta\boldsymbol{\theta}_{k-1}\end{bmatrix} + \begin{bmatrix}\mathbf{0}_3\\ dt_k[\hat{\mathbf{R}}^n_{b,k}\tilde{\mathbf{f}}_k]_\times\delta\boldsymbol{\theta}_k + dt_k\hat{\mathbf{R}}^n_{b,k}\mathbf{w}_{\mathbf{f},k}\\ -dt_k\hat{\mathbf{R}}^n_{b,k}\mathbf{w}_{\boldsymbol{\omega},k}\end{bmatrix}$$

at 820[Hz], we can use

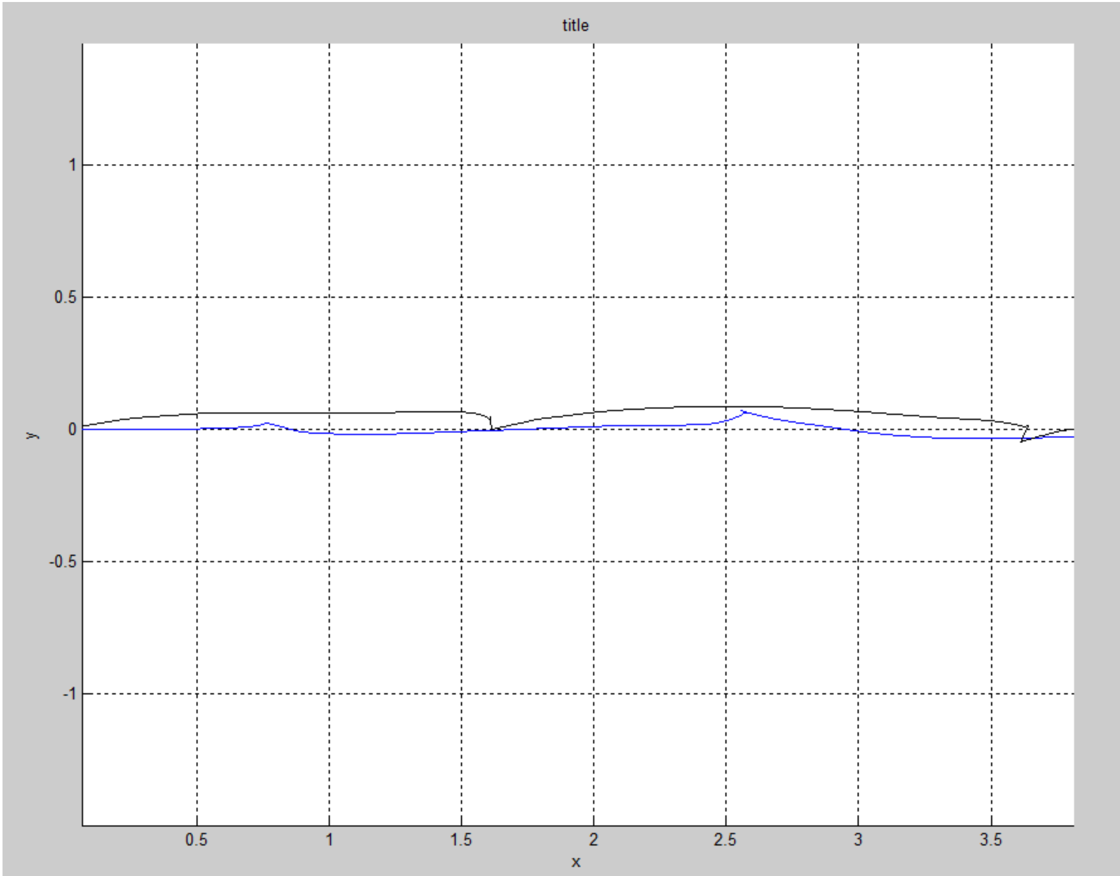
$$\begin{bmatrix}\mathbf{p}_k\\ \psi_k\end{bmatrix} = \begin{bmatrix}\mathbf{p}_{k-1}\\ \psi_{k-1}\end{bmatrix} + \begin{bmatrix}\mathbf{R}_{k-1}d\mathbf{p}_k\\ d\psi_k\end{bmatrix} + \mathbf{w}_k$$

at ~1[Hz]

KTH Elektro-
och systemteknik

Current work

- RTS smoothing



KTH Elektro-
och systemteknik

Current work

- Fixed-point algorithm implementation

  Super-cheep implementation!

KTH Elektro-
och systemteknik

Summary – Current work

- Constrained filtering
- Upper/lower lever filtering division
- Smoothing
- Fixed-point implementation

KTH Elektro-
och systemteknik

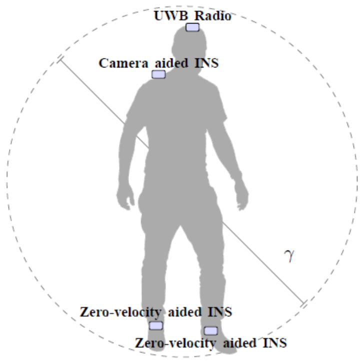
Future work

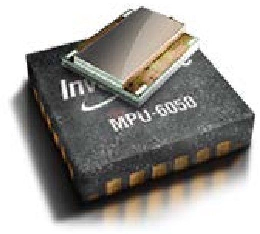
KTH Elektro-
och systemteknik

Needed future work

In order of importance

1. Integration
2. Integration
3. Integration
4. Integration
5. Magnetometer
6. InvenSense IMU
7. (Wireless)





KTH Elektro-
och systemteknik

- + alot of small fixes and features

The end

KTH Elektro-
och systemteknik