

Demonstration of a Wireless Sensor Network for Real-time Indoor Localisation and Motion Monitoring

Lasse Klingbeil
Autonomous Systems Laboratory
CSIRO ICT Centre
QCAT, Brisbane, Australia
Lasse.Klingbeil@csiro.au

Tim Wark
Autonomous Systems Laboratory
CSIRO ICT Centre
QCAT, Brisbane, Australia
Tim.Wark@csiro.au

Abstract

This paper describes our demonstration of a wireless sensor network for monitoring human motion and position in an indoor environment. Mobile sensor nodes comprising mote-type devices, along with inertial sensors are worn by persons moving inside buildings. Motion data is pre-processed onboard mobile nodes and transferred to a static network of seed nodes using a delay tolerant protocol with minimal radio packet overhead. A Monte Carlo based localisation algorithm is implemented, which uses a person's pedometry data, indoor map information and seed node positions to provide accurate, real-time indoor location information.

1 Introduction

The ability to track the position of nodes, e.g. vehicles, animals or people, opens up a wide range of applications in transport management, agriculture, military and health domains. Mobile nodes can also form an important role as data ferries in networks with disconnected nodes. As a result, localisation has been an important and growing research topic for wireless sensor networks [3, 1]. This paper describes a sensor network for real-time, mobile-node localisation, that uses a Monte-Carlo based approach to combine a local mobility model, indoor map information and proximity information from static seed nodes. The system has been designed as a low cost, effective solution for personal-tracking in indoor environments. The key advantages of the system we propose is that excellent localisation accuracy is achieved with low cost inertial sensors and radio transceivers, minimal system calibration and low network traffic.

2 System Description

The system is based on custom developed sensor network platform called Fleck[2]. Every Fleck contains a radio transmitter and a micro-controller as well as an interface to connect various sensor boards. The system

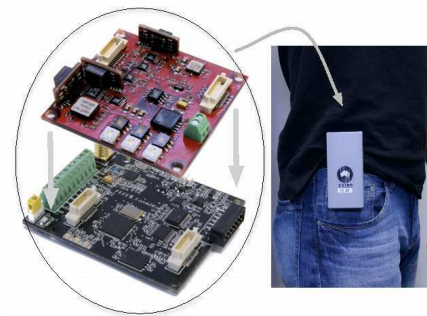


Figure 1. Fleck3 sensor board with IMU daughterboard attached

consists of two key parts, the static and the mobile networks. The static network comprises a number of nodes which are installed at fixed locations in the building. The base node is connected to a PC, which does the final processing and visualization (see Fig. 2). Each of the static nodes has a connection to the base node using a multi-hop routing protocol. The mobile network consists of nodes which are worn by persons moving through the building. As shown Fig. 1, the mobile nodes have an IMU daughterboard attached ('Inertial Measurement Unit', 3 axes accelerometers, gyroscopes and magnetometers) and are the data sources in the network. To enable implicit localisation of these nodes, their transmission power is reduced to limit the transmission range, r_t , to about 5m. A mobile node sending a packet to base via a forwarding

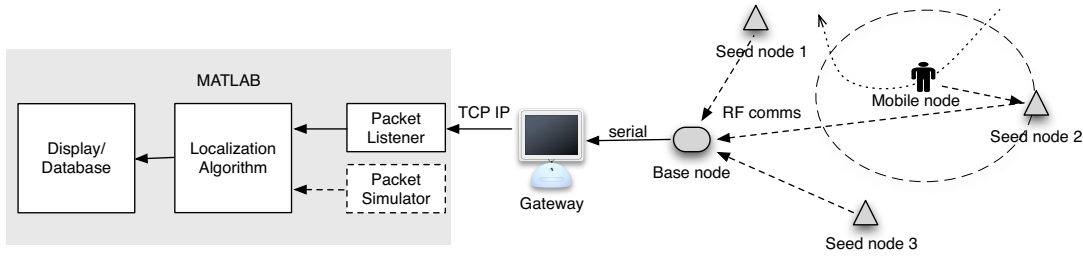


Figure 2. System architecture for indoor localisation system.

static node can be localised within a radius of roughly r_t around that static node. For transferring data from the mobile nodes to the seed nodes, a Delay Tolerant Network is used. Every motion event generated on a mobile node is locally stored in an event queue. At the same time, the mobile node beacons at a predefined interval, using the first event in the queue as a beacon message. If a nearby seed node receives this message, it forwards it back to the base node. The forwarded message is also received by the originating mobile node, which uses this as notification of packet delivery ('implicit acknowledge') and processes the next available event in the queue.

A crucial aspect of the system is preprocessing of motion data on mobile nodes. Preprocessing the raw data and extracting only relevant events reduces the number of packets that need to be sent over the network. The relevant information we extract for the proposed localisation method is (i) the occurrence of a walking step together, (ii) movement direction. To detect the occurrence of a walking steps, we use the data from the accelerometer sensor. These are corrected by the offset due to gravity and then filtered using a simple FIR averaging filter. Step events are then detected using an algorithm based on thresholds and heuristics to ensure robustness. For every detected step, the corresponding heading information is also calculated by combining the gyroscope and the magnetometer readings via a complementary filter.

The localisation algorithm uses Monte Carlo estimation techniques, which has already been used in previous localisation work [1, 3]. Our algorithm extends previous work however, by combining three key pieces of information in the filter: the proximity information from static seed nodes, mobility information derived from onboard inertial sensors and indoor map information. The algorithm runs on a PC and positions are updated every time a packet is received from one of the mobile nodes. Figure 3 shows the real and the reconstructed walking path using our system.

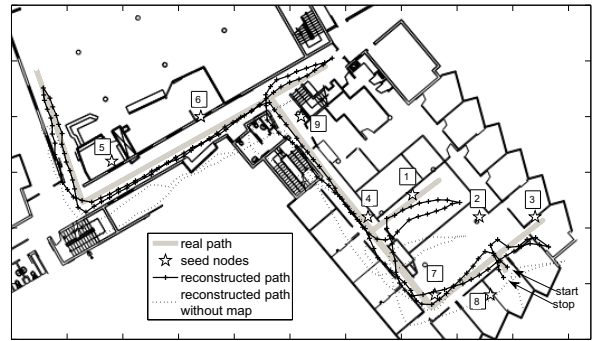


Figure 3. Real and reconstructed walking path using the heading based motion model

3 Demonstration Details

Our demonstration consists of a number of static seed nodes positioned in a confined area. A mobile node will be worn by a person, whose motion parameters like steps and directions are detected and sent to a base station. The localisation software will track the position of the person using the motion data, the seed node positions and map information.

References

- [1] L. Hu and D. Evans. Localisation for mobile sensor networks. In *Tenth International Conference on Mobile Computing and Networking (MobiCom)*, 2004.
- [2] L. Klingbeil and T. Wark. A wireless sensor network for real-time indoor localisation and motion monitoring. In *Information Processing in Sensor Networks (IPSN 08, to appear)*, 2008.
- [3] M. Rudafshani and S. Datta. Localization in wireless sensor networks. In *Information Processing in Sensor Networks (IPSN)*, pages 51–60, 2007.