Senior Thesis Proposal

Using Wireless Sensor Networks to Detect Falling

Brad Mager

ECE 3992
University of Utah
April 30, 2012

Thesis Advisor: Dr. Neal Patwari

Abstract

As the number of people over 65 increases, there is a growing demand for facilities that can provide care to the elderly, as well as for technologies that can improve the level of care offered. One area of concern is that of falling, which accounts for a large number of fatalities among the elderly, especially if medical assistance is delayed. My research involves using a wireless sensor network to detect falls, so that caregivers can be immediately alerted if one of their residents has fallen.

Introduction

The number of elderly people in the United States is growing rapidly. By 2050, the number of people 65 and older is projected to rise to 88.5 million, more than double the population of elderly today [1]. This rapid change in demographics is leading to an increased need for care facilities that can provide day-to-day assistance to people in their later years [2]. Because most people would rather live autonomously for as long as possible, assisted living facilities are becoming a popular option, since residents can live on their own while having care providers nearby to assist with daily activities. However, lack of constant monitoring means the residents may become ill or injured without a caregiver knowing right away.

One major concern for the elderly is that of falling, which affects nearly one-third of all adults over 65 each year and is a leading cause of death in the elderly [3]. It would therefore be useful if some kind of ambient technology could detect if a resident has fallen and alert a caregiver at the facility. Currently, detecting if someone has fallen requires the person to wear a device — either one that senses the fall using an accelerometer, or one that the wearer uses to call for help with a push-button. However, such a device offers no advantage if a person forgets to wear it or chooses not to because it is uncomfortable or seems undignified. Ideally, these devices will be superseded by smart environments that can provide real-time monitoring of residents who may need extra assistance. Some efforts in this direction include floor vibration sensors, microphone arrays, and Doppler radars [4].

In another new area of research, arrays of wireless RF (radio frequency) sensor nodes are being used to detect where people are in a building, based on how they affect the RF field [5, 6]. This kind of technology — called device-free localization (DFL) — can be used for intrusion detection and emergency situations such as earthquakes, and there is now interest in employing it in assisted living facilities [7].

The goal of this project is to determine how to use device-free localization to detect, in real time, if someone has fallen. The final result will be an algorithm that can process data from an array of RF sensor nodes and detect falling with a small probability of false alarms, and this will include determining an optimal arrangement of sensor nodes for detecting falling.

Device-Free Localization Research

Research in device-free localization is currently being conducted in the SPAN Lab (Sensing and Processing Across Networks) at the University of Utah, under the direction of my advisor, Dr. Neal Patwari. The proposed project will improve upon DFL research as described below, and it will use the same node hardware as currently used in other experiments, so no new hardware needs to be developed.

DFL research experiments are typically conducted as follows. An array of sensor nodes is set up in or around a room, typically in a perimeter arrangement (Fig. 1). These transmit and receive RF signals, with each node taking a turn to transmit. A person moving within the room affects the RF field, which causes variation in the received signal strength (RSS) the nodes measure from each other. By applying the appropriate algorithms to the data collected from the nodes, it is possible to determine the location of a person with radio tomographic imaging, in which the RSS data are processed to create an image that shows the position of one or more persons at each moment.

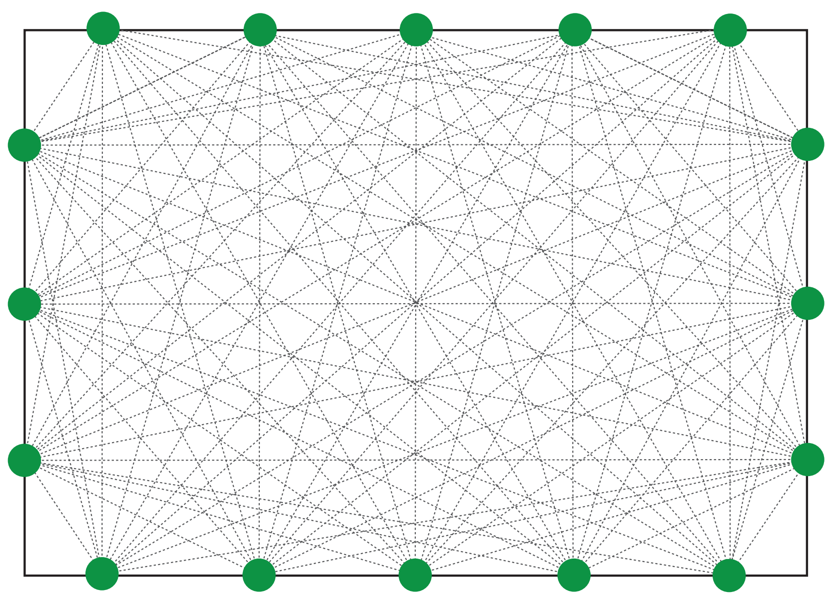

Fig. 1 — Example of the RF links between sensor nodes set up around a room, as viewed from above.

The RF sensor nodes are USB dongles based on a Texas Instruments design that uses their CC2531 System-on-Chip with an IEEE 802.15.4 compliant radio operating in the 2.4 GHz band [8]. The nodes are powered through the USB plug, either through a power adapter that plugs into a wall socket, or by attaching them to a battery pack with a USB attachment (Fig. 2). The nodes communicate their data through a base station node, which is connected to a computer gathering the data.

Fig. 2 — CC2531 USB dongle in a custom-built battery pack

Project Implementation

In my research, I will use a setup similar to the DFL setup described above. To date, most experiments and deployments of RF nodes used for device-free localization have been focused on determining the horizontal position of one of more people in a room or building. Detecting if someone falls will require determining a person's vertical position and motion. As such, a likely arrangement of sensors will be two levels of nodes set up around a room, one at about chest height and another near the floor.

To calibrate the system, I will first take data while standing and then lying in various locations around the room inside the array of sensors. This is simply to make sure it is possible to determine the difference between these two positions, which represent the extremes of the before and after positions of a fall. Processing this data will help to determine if it is necessary to adjust the arrangement of the nodes to optimally detect the difference in vertical positions.

In the following tests, I will mimic falling — or actually fall onto a pad, if I can do so without any chance of injury. This motion will be performed at different locations around the room, and in different ways, e.g., falling forward as if tripping, and simply crumpling to the floor as if passing out. In addition, data will be recorded from other types of vertical downward motion, such as sitting down, to make sure the system can detect the difference between that motion and falling, to avoid false positives.

All the data will be collected in a series of MySQL databases, then processed in Matlab to determine an algorithm that can detect falling in real time. It will likely take multiple rounds of experiments and work on the algorithm to successfully develop a system that performs as desired. The largest portion of the project will involve working with the collected data to understand what they show about someone's motion, then using that information to develop a specific algorithm for detecting falling. Typically, researchers in device-free localization can spend several weeks interpreting the data and refining the algorithms to accurately translate the sensor data into a representation of the actual motion.

The end goals of the project include determining the optimal placement of sensor nodes for DFL-based fall detection, and development of an algorithm that can detect falling in real time. When the system is up and running, several sensor nodes will be set up in a room monitoring movement and the data will be continually processed. If a person falls, or mimics falling for the sake of demonstration, the network will detect that and send an alert, possibly via a text message, to a caregiver.

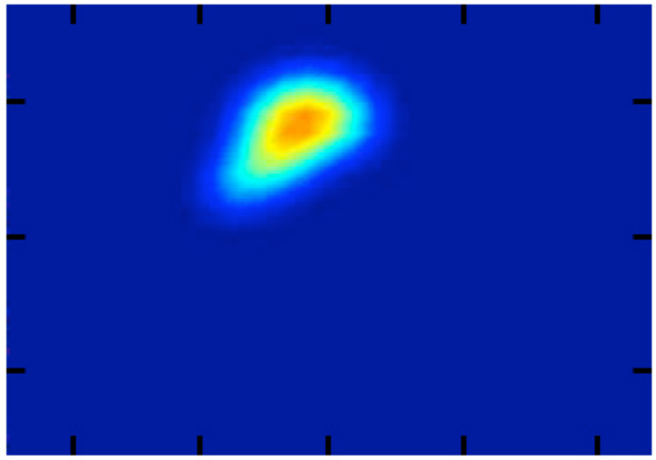

Fig. 3 — Radio tomographic image showing a person's location in a room, after processing the RF signals in Matlab
(*image courtesy of Dr. Maurizio Bocca*)

Interfacing the Project Components

Unlike a project in which a device is built, this research project will employ a system of experiments that have demonstrated success in producing valid data. The difference will be in the type of motion that is to be detected. As such, the hardware components and most of the software components have already been shown to work together. Further work will be done on refining the software on the nodes and determining the optimal arrangement of the nodes.

Below are the components of the project:

<u>Hardware</u>
Node deployment — array of RF sensor nodes set up in a room
Base station node — the node that gathers data from all the other nodes
Database server — receives data over the network and puts it in a MySQL database

<u>Software</u>
C program on nodes — determines how data is transmitted
Python code on GuruPlug & server — moves data across the network into database
Matlab — used for data processing

The nodes are programmed in C, and the Guru Plug and database server run Python code to get the data into a MySQL database. The largest portion of the project will involve working with the data in Matlab to understand what it shows about someone's motion, and how to use that information to develop a specific algorithm for detecting falling.

Bill of Materials & Vendor List

For the experiments, I will be using equipment already owned by the SPAN Lab, so there will likely be no need to purchase anything. However, for the sake of completeness in this proposal, the approximate costs of the items that will be used in my project, along with the manufacturers or distributors are shown below.

Note that the battery packs are made up of two parts: a 2-AA battery holder and one end of a USB extension cable. These items are purchased separately, then we build the USB battery pack by attaching the power leads (red and black wires) of the female end of the extension cable to the leads of the battery pack, then glue the USB plug to the battery holder. See Fig. 2 above.

Vendor and Parts List

	Part number	Lead time	Unit cost	Quantity	Total cost
CC2531 USB dongle Manufacturer: Newonics	N/A	4–6 weeks	$18.00	40	$720.00
2-AA Battery Holder, Enclosed with Switch Distributor: Pololu Robotics & Electronics	1160	in stock	$1.16	40	$46.40
1.5 ft USB Male to Female Extension Cable Distributor: Monoprice	5431	in stock	$0.87	40	$34.80
TI CC Debugger Manufacturer: Texas Instruments	CC-DEBUGGER	in stock	$49.00	1	$49.00
GuruPlug Server Manufacturer: Globalscale Technologies	003-GP00A100	in stock	$99.00	1	$99.00
				Total	**$949.20**

Risks

With a research project such as this, and because this is a relatively new area of research, the biggest risk is that it will not be possible to detect falling in real time with this kind of device-free localization. Should this happen, some useful results may still be salvaged from the data, or the nature of the experiments could be adjusted to determine some other application of this technology.

Another risk, which could delay the experiments, is the lack of a good location to perform the tests. More to the point, I may want to do further tests in a location that is no longer available. This would require re-doing previous tests in a new location so that different tests could be conducted in the same environment.

Finally, though less likely, the nodes could malfunction, in which case it may be necessary to have more nodes manufactured or borrow nodes being used in other experiments in the SPAN Lab. This could set the project back by several weeks.

What can help mitigate these risks is the fact that I have been working in the SPAN Lab over the past semester and have assisted other researchers in setting up and conducting tests. This has given me familiarity with the equipment as well as experience in conducting experiments of this nature. As such, I have been working with people who have expertise in this area and who can provide guidance in my research, increasing the likelihood of success.

Schedule and Milestones

May 2012: Set up a test experiment, perhaps with just 8 nodes, to minimize the amount of data to process. Use this experiment to gather a few sets of data, gain more experience in using Matlab to process this kind of data, and determine the kind of visualization tools that need to be developed. The visualization tools are important for turning massive amounts of numerical data into something the human eye can more easily interpret and sort through.

<u>Milestone 1</u>: Conduct Experiment 1 and develop visualization tools in Matlab.

June 2012: Use the visualization tools to look at the data gathered in Experiment 1 to decide on which feature sets are likely to be part of an algorithm to detect falling. This process involves isolating different parts of the data and determining if they might offer any evidence that someone has fallen. For example, one can select any individual link — the data from just two nodes — or sets of links, such as data from all the nodes at ground level, and examine those. The groups of node data most likely to indicate falling become part of the feature set.

<u>Milestone 2</u>: Look at the data with visualization tools and decide on an initial feature set.

July – August 2012: Conduct a full experiment, Experiment 2, with about 40 nodes, as described earlier. This will involve standing and lying in different positions to help calibrate the system, and trying out various types of falling in different parts of the room. Once the system is set up, tests can be conducted fairly quickly, perhaps in a day or two. However, some adjustment in the setup may be necessary, and it is desirable to get as much data as possible once the nodes have been set up, so I have allowed at least a week for Experiment 2. The rest of the time for this milestone will be going through all the data and refining the feature set. Because there will be much more data than in the first experiment, this will take more time.

Milestone 3: Conduct Experiment 2, process the data, and refine the feature set.

September – October 2012: Conduct Experiment 3, which will likely have a slightly different arrangement of nodes from the previous experiment, depending on the results of processing the earlier data. This experiment may also include testing different types of motion that were not tried before, such as sitting down, to see if the system can discriminate between false positives and the real thing.

Milestone 4: Conduct Experiment 3 and process the data.

November – December 2012: By this point, there should be enough information on which feature sets indicate falling to develop an algorithm, which is the task for this milestone. Other work involves optimizing the processing of the data so that falling can be detected in real time.

Milestone 5: Develop the formal algorithm.

January – February 2013: Test the system to confirm that it works as advertised. Ideally, this will be a set of experiments conducted in different environments, to more closely mimic the real world. In fact, some of these test may be conducted in a house or apartment, where furniture can alter the signals that the nodes receive from each other.

Milestone 6: Validate the system.

March – April 2013: Write up the results in a formal paper and present at the ECE Department Technical Open House.

Milestone 7: Write a formal paper and give at least one presentation of the results.

References

[1] G. Vincent and V. Velkoff, The Next Four Decades, The Older Population in the United States: 2010 to 2050, U.S. Census Bureau, May 2010.

[2] M. Pieper, M. Antona and U. Cortés, Ambient Assisted Living, ERCIM News, Number 87, Oct. 2011.

[3] J. Stevens, G. Ryan and M. Kresnow. Fatalities and injuries from falls among older adults – United States, 1993–2003 and 2001–2005, Morbidity & Mortality Weekly Report, 2006;55 (45):1221-1224

[4] L. Liu, M. Popescu, M. Skubic, M. Rantz, Automatic Fall Detection Based on Doppler Radar Motion Signature, 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops.

[5] J. Wilson and N. Patwari, Radio Tomographic Imaging with Wireless Networks, IEEE Transactions on Mobile Computing, vol. 9, no. 5, May 2010, pp. 621-632.

[6] N. Patwari and J. Wilson, "RF sensor networks for device-free localization: Measurement, models and algorithms", *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1961-1971, Nov. 2010.

[7] O. Kaltiokallio and M. Bocca, "Real-time intrusion detection and tracking in indoor environment through distributed RSSI processing", in Proc. of the 17th IEEE Int. Conf. on Embedded and Real-Time Computing Systems and Applications (RTCSA 2011), Toyama, Japan, August 2011.

[8] Texas Instruments, "A USB-enabled system-on-chip solution for 2.4 GHz IEEE 802.15.4 and ZigBee applications", available at www.ti.com/lit/ds/symlink/cc2531.pdf