The Pervasive Sensor

Kristof Van Laerhoven

Lancaster University
United Kingdom
`http://www.comp.lancs.ac.uk/~kristof`

Abstract. Forget processing power, memory, or the size of computers for a moment: Sensors, and the data they provide, are as important as any of these factors in realising ubiquitous and pervasive computing. Sensors have already become influential components in newer applications, but their data needs to be used more intelligently if we want to unlock their true potential. This requires improved ways to design and integrate sensors in computer systems, and interpret their signals.

1 Introduction

Most dictionaries agree on the fact that sensors are devices that are capable of detecting and responding to physical stimuli such as movement, light or heat. The more perceptive reader will observe that this definition is not very specific: Identifying the sensors in a system is often fairly straightforward, but trying to track down examples of the first human-made sensors with this definition for instance is much harder. Similarly, the difficulty in exactly identifying the parts in a system that belong to one sensor, underlines the vagueness of what a sensor really constitutes.

This introduction will begin with challenging a few preconceptions one might have about sensors and the process of sensing and measuring, before moving on to stress the importance of sensors in the future of computing.

1.1 What Does a Sensor Look Like?

Show an engineer a photodiode, and chances are that he or she will identify it as a (light) sensor, or at least would not object to someone else calling it a sensor. Saying that a bird in a cage is a sensor might seem a lot more challenging, but given the definition above it would be possible to treat it as such: 19th century miners would carry along ‘cages’ with them to detect the presence of carbon monoxide or methane, deadly mine gases which humans cannot detect until it is too late. The canary in the cage, on the other hand, is more sensitive to these gases and drops dead minutes before humans will. Few will challenge the idea of a thermometer being a sensor, but convincing people that their pet canary is really a sophisticated toxic gases sensor is usually a bit harder.

Using a wide enough interpretation, almost anything fits in the definition of a sensor: In fact, even humans can be considered as sensors.

1.2 The Eye of the Beholder

The observation of, and response to, the state of the sensor plays a crucial role as well. The shape and colour of clouds, the way insects fly, or the arrangement of a Galileo's thermometer[1] may be interpreted as just pleasant things to look at by some people, but are tell-tale signs for weather forecasting to others. Instruments to measure the outside temperature during a cold winter, or devices to alert miners when dangerous gasses are present, are useless until they are observed properly by someone who monitors the sensor's output and takes actions accordingly.

1.3 One-way Bridge from Real to Digital

The sensor can also be regarded as a way to capture information from the real world and reflect it in the virtual or digital world. It is the ideal mechanism to integrate new information into an application, and sensing methods for many applications are evolving from a typical scenario where the human provides all input to the application, to one where this becomes more and more automated.

Healthcare monitoring applications, for example, used to rely on information that the doctors would manually enter in the system after doing the measurements themselves. Nowadays, this process has advanced to a stage where medical sensing devices are constantly observing the patient's state.

To focus on this automated sensing, the remainder of this extended abstract will restrict its use of the word sensor to the devices that detect physical phenomena without the help of a human.

1.4 Taking the Human Out of the Loop

Historically, sensors were primarily there to directly suit the user, and the user would be very involved in the process of sensing (e.g., tapping on a needle based barometer), interpreting the sensed (e.g., reading the amount of water in a rain meter), and taking action based on what was sensed (e.g., slowing down the car after noticing you are going too fast on the speedometer). Recent trends, however, -and ubiquitous computing is part of this as well- seem to minimise this human factor, and sometimes even remove it altogether from the process. Being able to build devices that interpret sensor data and act on it without disturbing the user, means that sensors would enable all these ubiquitous computing elements to live without strict user supervision, making them truly unnoticeable.

---

[1] Galileo's thermometer is made of suspended weights in a sealed glass cylinder containing a clear liquid. If this liquid changes temperature, it changes density and the suspended weights rise and fall. The weights' positions can thus be interpreted as a temperature measurement.

1.5 The Impact of Sensors

Sensors are increasingly being integrated and embedded in user interfaces to give the user a more intuitive, more sensitive, or more appropriate way to interact with computers. At the same time, sensors are also being used as replacements for user interactions altogether, in cases where the sensing is trivial, or where mistakes are limited and not critical. The sensors are therefore crucial components in the visions of ubiquitous and pervasive computing, and advancing research into sensors and how one can take advantage of their data is valuable even beyond these fields.

A few challenges summarize the key areas: 'Embedding and Interfacing Sensors' considers the practical issues of adding and networking a sensor component to an object or application, 'Learning from Sensed Data' points out the difficulty of replacing the human perception with algorithms, and 'Sensor Fusion' lists some prospects and problems when trying to combine sensors' output.

2 Embedding and Interfacing Sensors

Knowing the right sensor and algorithm that process the sensed data is not enough to guarantee the system will work; this section will begin with another requirement.

2.1 Considering the Physical Properties of Sensors

The location and orientation of a sensor can play a vital role. To show what the impact can be of *where and how* a sensor is attached, consider this scenario in the area of body-worn motion sensors: A student wants to build a wearable system that monitors his posture and activities throughout the day, and as an easy start he sets out to just detect at what times in the day he is sitting down, and at what times he is standing upright. Knowing a bit about sensors, he believes that an orientation sensor just above the knee would be sufficient: it would give a relatively horizontal reading when sitting down, and the sensor would give a vertical reading when standing upright. Figure 1 illustrates the reasoning behind this.

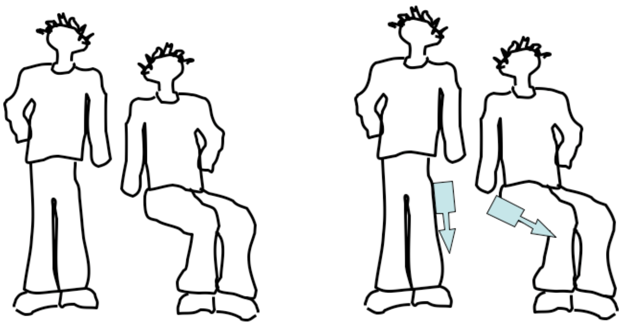

Fig. 1. Example of how to easily detect whether the user is sitting down or standing upright by attaching an orientation sensor to the upper leg. It seems to be a sound approach...

Figure 2 shows how the weaknesses of this approach become obvious when following the student trying his system out: To wind down after the efforts of building his wearable sensor, he visits his local pub. Standing at the bar, he habitually places his foot on a higher position, which results in the sensor being oriented almost horizontally and logging the student, mistakenly, as ‘sitting down’ (Figure 2a).

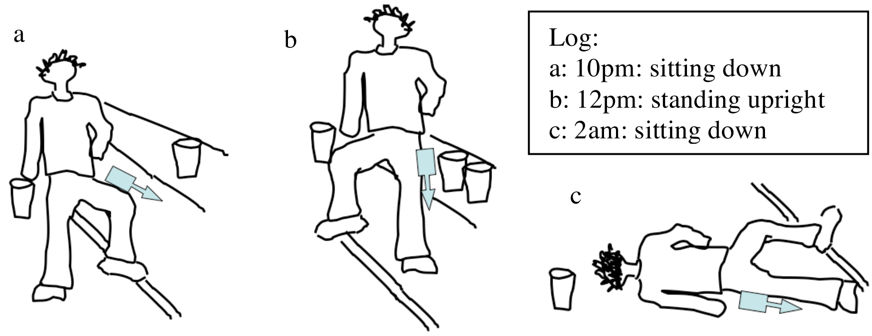

Fig. 2. A few examples of how the system from Figure 1 fails to correctly recognize ‘sitting down’ and ‘standing upright’ throughout the student’s evening.

Similar mistakes can also happen the other way around: the student might be sitting on an elevated surface, leaving his leg dangling downwards far enough for the system to register this as ‘standing upright’ (Figure 2b). Even worse, poses that are neither ‘sitting down’ nor ‘standing upright’ could also be wrongly detected as one or the other: lying down (Figure 2c) for instance has the upper leg in a horizontal position, making the system register ‘sitting down’.

It is important to stress here that these errors happen regardless of the quality of the sensor or the algorithms that treat the sensed data: increasing the sensor’s sensitivity or using better machine learning techniques will not help.

This scenario shows not only how crucial the location, the orientation, and the position of sensors are, it also points at the importance of *combining sensors*. Adding a sensor on the other leg would improve the system dramatically (though errors can still occur). This leads to another challenge: how to transport the data from multiple, distributed sensors.

2.2 Networking Sensors

The sensed data needs in many cases to be processed elsewhere or merged with other sensed data from different locations. The first practical issue becomes then to transport and perhaps send the data over a network. Also, having multiple sensors is often just the first problem; some applications require sensor nodes in a network to be insertable or removable in an ad-hoc fashion, or they require the nodes to do both sensing and routing of neighbouring nodes’ data.

Many common standards exist for these cases, with especially body-wide wireless networks gaining a lot of popularity over the last years (think for instance of the IEEE 802.15 standards, Bluetooth and ZigBee). There exist many good overviews of these types of networks (see for instance [13] or [14]), but a different, non-wireless method of networking will be discussed next, to widen the spectrum of sensor networking possibilities.

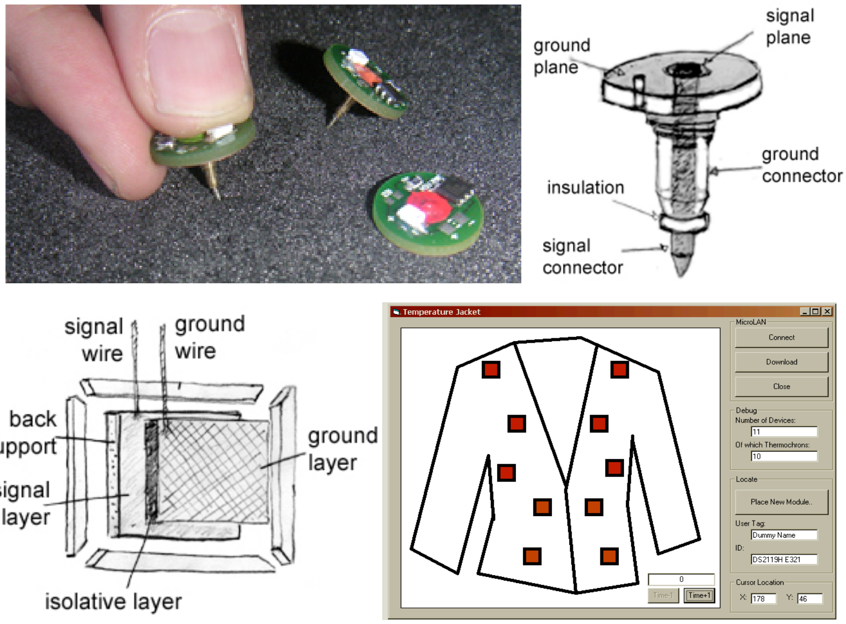

Fig. 3. The components of Pin&Play: top: the node with pin connector, bottom-left: the surface structure (using conductive textile), and bottom-right: an application using clothing as the surface and temperature measuring pins that have been attached ad hoc to the clothing [12] [9].

Pin&Play is based on the vision that layered surfaces can be used as a network for objects, such as sensor nodes, that become attached to these surfaces. It is very similar to the Pushpin initiative from Lifton and Paradiso [6], but has a simpler, bus-type structure with a master-slave type of architecture. Even tiny devices can be attached by means of pin adaptors and a surface with layers of conductive sheets (see Figure 3), to gain power and networking capabilities with the freedom of being plugged in any place or in any orientation.

This network is far more appealing from an engineering point of view, since it doesn't require the network nodes to have batteries or wireless communication modules. Everything is handled via a 'master' that provides the power to the entire network and regulates network traffic, the nodes just need to be pinned in the surface to switch them on and introduce them into the network. The use of off-the-shelf components and a well-supported network protocol resulted in robust and small prototypes that are cheap, easy to (re)produce and yet more than powerful enough for many applications (see Figure 3). The network can handle hundreds of devices in a small, though two dimensional, space, which is especially attractive in the augmentation of small and mobile sensing applications. The main disadvantage for use of this networking technology in sensor networks, however, is the low bandwidth (at best 16300 bits per second). The network has in its current state also no way of finding out where a node is in the network, unlike many wireless solutions.

Pin&Play is nevertheless a great alternative to traditional wired and wireless networks, especially for dense topologies and applications where nodes rest on a common surface and where sensors' simplicity and size are more important than their transmitting speed (e.g., [11]).

3 Learning from Sensed Data

Many applications in ubiquitous and pervasive computing do not use the sensors' output directly. Instead, they classify the sensed information into concepts that are more useful to the application. These mappings between the raw sensor data and classes of interest can be straightforward (e.g., thermometer values that get classified in 'warm' and 'cold', or a passive infrared detector's values that are transformed to 'motion' and 'no motion'). These mappings are for many applications a lot more complicated, however; think for instance of a microphone for which its audio signal can be transformed into syllables or words in speech recognition, or a camera for which its images can be mapped to objects in image analysis. This more complex mapping of sensor data to high-level concepts has notably in the field of human-computer interaction frequently been marked as *context awareness*, where information from sensors is used and classified to give a description of the user's context.

The more complex classification algorithms usually work by building an internal world model that is shaped by typical examples in a so-called training phase. This is similar to finding typical properties in a few representative pictures of the number seven for a character recognition algorithm, to make it afterwards able to recognize all new instances of seven.

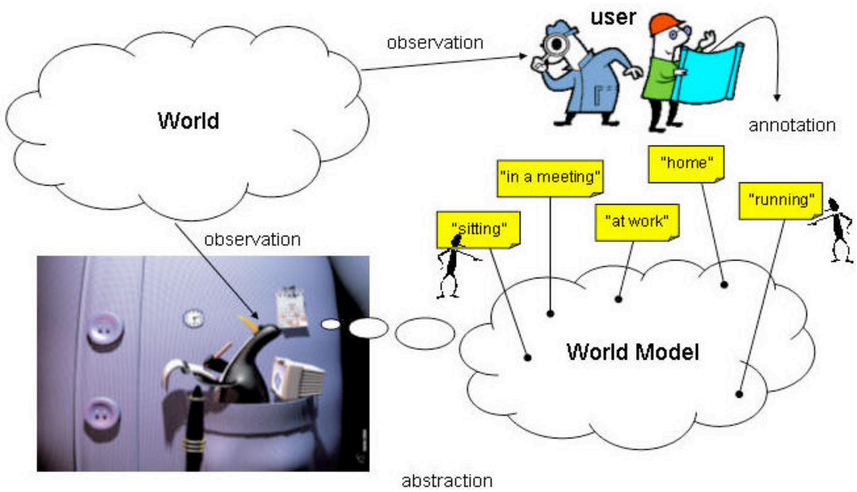

Fig. 4. Diagram of incremental learning: both user and system observe the same world, which is modeled and abstracted by the system and annotated by the user.

An interesting and highly flexible type of classification of sensor data goes a step further: *incremental learning* keeps the system's world model flexible, so that new contexts can be taught to the system at any time. It works by 'showing' the new context to the system repeatedly, similar to the way speech recognition can be optimised by letting the user repeat a few phrases. Figure 4 shows a diagram of such a system where both user and system perceive the world, using respectively senses and sensors, and where the system maintains a flexible internal world model that the user is then able to annotate with her concepts.

4 Sensor Fusion

The traditional sensing system usually has one or a few sensors, and is combined with an algorithm that is specific to that architecture. There exists an alternative architecture, however, which is more distributed in nature: it is based on a large number of small and simple sensors. This direction was taken in research at MERL [5], Philips [4], the TEA project [3], or more recently MIT [1] and [7], but is still rather new and unexplored. The combination of many simple sensors that individually give information on just a small aspect of the context, results in a total picture that might be richer than the one-smart-sensor approach. The distinction between both approaches is depicted in Figure 5.

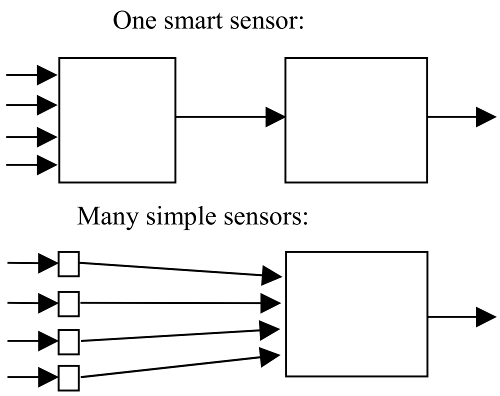

Fig. 5. Two sensor architecture approaches.

4.1 Many Sensors

Most of the benefits of using multiple sensors were mentioned in previous research (see for instance [5]):

- **Cheap**. The small simple sensors require generally less resources and cost less than for instance cameras and GPS systems. An extremely large amount of simple sensors could of course invalidate this.
- **Robust**. Since the sensors we use are small, they can smoothly be distributed over a larger area, which makes the sensing system less prone to errors. In case a sensor gets blocked or damaged, other sensors will still capture context-relevant information due to the redundancy in the sensors.
- **Distributed**. The size also allows the sensors to be integrated into clothing much easier, and the high number allows them to increase the sensed area.

- **Flexible**. The richness and complexity of the identifiable contexts is directly linked to the amount, position and kind of sensors. Adding, moving, or improving sensors hence increases the performance of the system.

4.2 Simpler Sensors

The real bottleneck in this method is the software algorithm that has to combine and analyze all the data. Research described in [8] defines the choices one has to make in finding a suitable algorithm and argues for using neural networks to first cluster the sensed data.

Fig. 6. Experiments of [10] with networks containing a large amount of motion sensors: the left two using 40 accelerometers, the right two using 90 tilt switches.

There is unfortunately a theoretical limit to the number of sensors one can fuse together: the adaptive algorithms become slower and less effective as sensors get added, due to the Curse of Dimensionality [2]. Figure 6 shows snapshots of experiments with wearable sensor suits using motion sensors that are distributed on the legs, while the wearer performs certain activities of interest (such as walking, sitting down, running, climbing stairs, etc.). One suit uses 40 accelerometers attached (giving a 10 bit value each), whereas the other uses 90 much simpler tilt switches (1 bit each). These experiments have shown that even simple binary sensors are able to detect complex postures, as long as a sufficient amount of them is distributed. This is important because these simpler sensors require far less resources.

5 Summary

This extended abstract argues that sensors are key components in the vision of ubiquitous computing. The traditional sensor is defined vaguely already, but the new breed of sensors and sensing applications will require even more investigation. Placing and designing sensors optimally and organising them in networks, creating algorithms that are able to learn what is being sensed, and combining sensed data, are issues that will require more attention.

Acknowledgements

Thanks go out to the organisers and all participants of UCS 2004, Tokyo, for providing a most stimulating environment with plenty of questions, remarks, and quality research that shaped this extended abstract.

References

1. Bao, L., and Intille, S. S. “Activity Recognition From User-Annotated Acceleration Data” In Proceedings of Pervasive 2004, Lecture Notes in Computer Science. Springer Verlag, Vienna, Austria. 2004.
2. R. Bellman, Adaptive Control Processes. Princeton University Press. 1961.
3. Esprit Project 26900. Technology for Enabling Awareness (TEA). http://www.teco.edu/tea, 1999.
4. Farringdon, J., Moore, A., N. Tilbury, J. Church and P. Biemond “Wearable Sensor Badge & Sensor Jacket for Context Awareness”. In Proceedings of the Third International Symposium on Wearable Computers (ISWC’99), San Francisco, pp.107-113.
5. Golding, A. R. , and N. Lesh, “Indoor navigation using a diverse set of cheap, wearable sensors”. In Proceedings of the third International Symposium on Wearable Computers, 1999, pp. 29-36. 1999.
6. Lifton, J., Paradiso, J. “Pushpin Computing System Overview: A Platform for Distributed, Embedded, Ubiquitous Sensor Networks” In Proc. Of Pervasive Computing, LNCS 2414, Springer Verlag. 2002. pp. 139-151.
7. Tapia, E. M., Intille, S. S. and Larson K. “Activity Recognition in the Home Using Simple and Ubiquitous Sensors”. In Proceedings of Pervasive 2004, Lecture Notes in Computer Science. Springer Verlag, Vienna, Austria. 2004.
8. Van Laerhoven, K., Aidoo, K. and S. Lowette. "Real-time analysis of Data from Many Sensors with Neural Networks". In Proc. of the fifth International Symposium on Wearable Computers, ISWC 2001, Zurich, Switzerland, IEEE Press, 2001, pp. 115-123.
9. Van Laerhoven, K., A. Schmidt and H.-W. Gellersen. "Pin&Play: Networking Objects through Pins". In Proceedings of Ubicomp 2002, G. Boriello and L.E. Holmquist (Eds). Lecture Notes in Computer Science; Vol. 2498, Goteborg, Sweden. Springer Verlag, September 2002, pp.219 - 229.
10. Van Laerhoven, K., and H.-W. Gellersen. "Spine versus Porcupine: a Study in Distributed Wearable Activity Recognition". In Proceedings of the eighth International Symposium on Wearable Computers, ISWC 2004, Arlington, VA. IEEE Press, 2004, pp. 142-150.
11. Van Laerhoven, K., Villar, N., and H.-W. Gellersen. "A Layered Approach to Wearable Textile Networks". In Proceedings of the IEE Eurowearable 2003; IEE Press. Birmingham, UK, pp. 61-67.
12. Van Laerhoven, K., Villar, N., Schmidt, A., H.-W. Gellersen, M. Hakansson, and L. E. Holmquist. "Pin&Play: The Surface as Network Medium" In IEEE Communications, April 2003, Vol.41 No.4., ISSN:0163-6804, IEEE Press, pp. 90-96.
13. ZigBee Alliance: http://www.zigbee.org
14. Bluetooth SIG: http://www.bluetooth.com/