The Rise of People-Centric Sensing

Technological advances in sensing, computation, storage, and communications will turn the near-ubiquitous mobile phone into a global mobile sensing device. People-centric sensing will help drive this trend by enabling a different way to sense, learn, visualize, and share information about ourselves, friends, communities, the way we live, and the world we live in. It juxtaposes the traditional view of mesh sensor networks with one in which people, carrying mobile devices, enable opportunistic sensing coverage. In the MetroSense Project's vision of people-centric sensing, users are the key architectural system component, enabling a host of new application areas such as personal, public, and social sensing.

Andrew T. Campbell, Nicholas D. Lane, Emiliano Miluzzo, Ronald A. Peterson, Hong Lu, Xiao Zheng, Mirco Musolesi, and Kristóf Fodor
Dartmouth College

Shane B. Eisenman and Gahng-Seop Ahn
Columbia University

The evolution of sensing, computing, and communication technology over the past few years has brought us to a tipping point in the field of wireless sensor networking. A decade ago, research prototype hardware began to emerge, facilitating the genesis of wireless sensor networks as they exist today: small resource-limited embedded devices that communicate via low-power, low-bandwidth radio. A natural first application of these networks of custom devices was solving relatively small-scale specialized problems in the scientific and industrial domains, such as forest monitoring and preventative maintenance.

Although these problems and applications remain important, the recent miniaturization and subsequent introduction of sensors into popular consumer electronics like mobile phones (such as the Apple iPhone), PDAs (such as the Nokia N810), and MP3 players (such as the Nike + iPod) has opened the door to a new world of application possibilities. With wireless sensor platforms in the hands of the masses, and with the proper architectural support, we can leverage wireless sensor networks to address urban-scale problems or provide global information access (such as public-sensing applications). At the same time, people as individuals, or in social or special interest groups, can apply these new sensing networks to applications with a more personal focus. We see a continuing push in this direction and the advent of a new era of people-centric sensing.

In a people-centric sensing system, humans, rather than trees or machines,

1089-7801/08/$25.00 © 2008 IEEE

Figure 1. People-centric sensing applications can be thought of as having a personal, social, or public focus.

become the focal point of sensing and the visualization of sensor-based information is for the benefit of common citizens and their friends, rather than domain scientists or plant engineers. Additionally, the users' aggregate mobility both enables sensing coverage of large public spaces over time and lets individuals, as *sensing device custodians*, collect targeted information about their daily patterns and interactions. The sensing coverage of spaces, events, and human interactions is opportunistic in a people-centric system because the system architecture has no point of control over the human mobility patterns and actions that facilitate this coverage. Although this lack of control can translate into gaps in sensing coverage, the alternative of a world-wide web of static sensors is clearly untenable in terms of monetary cost, scalability, or management. Furthermore, by having people carry the sensing devices in their mobile phones, iPods, and so forth, a people-centric sensing system creates a symbiotic relationship between itself and the communities and individuals it serves.

In this article, we describe our vision of people-centric sensing and the architectural support we're developing in the MetroSense Project to realize this vision (http://metrosense.cs.dartmouth.edu).[1] People-centric sensing (Figure 1) gives rise to a host of new applications that we classify either as *personal sensing*, those focused on personal monitoring and archiving; *social sensing*, those in which information is shared within social and special interest groups; or *public sensing*, those in which data is shared with everyone for the greater public good (such as entertainment or community action). Each application focus comes with its own challenges in terms of how to best sample the data, understand it, visualize it, and share it with others. Several prototype applications we're developing in the MetroSense Project cover these sensing scenarios and help us look at how people can best learn from the raw data, meaningfully represent that information, and share it, as appropriate.

The MetroSense Vision

The MetroSense conception of a people-centric sensing system is based on a three stage Sense, Learn, Share framework. (See the "Related Work in People-Centric Sensing" sidebar for other approaches and applications.) In the sense stage, MetroSense leverages mobility-enabled interactions between human-carried mobile sensors (such as mobile phones and personal medical sensing devices), static sensors embedded in the civic infrastructure (such as vehicle-based sensing networks and home medical sensing networks), and edge wireless access nodes providing a gateway to the Internet. Together, these support the delivery of application requests to the mobile devices, the sampling of sensors

Related Work in People-Centric Sensing

People-centric sensing sits at the nexus of several research disciplines, including sensor networking, pervasive computing, mobile computing, machine learning, human-computer interfacing, and social networking. Significant research contributions made within each discipline have facilitated the rise of people-centric sensing, and research focusing on synthesizing these contributions is now emerging.[1]

Several ongoing projects are related to our people-centric sensing initiative in the MetroSense Project. SensorPlanet is a Nokia-initiated global research framework for mobile-device-centric wireless sensor networks.[2] It provides hardware platforms and a research environment that helps researchers collect sensor data on a large and heterogeneous scale and establishes a central repository for sharing the data. SenseWeb,[3] a Microsoft Research sponsored project, provides shared sensing resources and sensor querying and data-collection mechanisms to develop sensing applications. In both of these projects, participating universities develop their own applications and share the collected data to facilitate research on data analysis and mining, visualization, and machine learning.

The UCLA Urban Sensing initiative has a vision of equipping users to compose a sensor-based recording of their experiences and environment by leveraging sensors embedded in mobile devices and integrating existing public outlets of urban information (such as weather, traffic, and air quality).[4] Urban Sensing is exploring how to coordinate these individual stories of everyday life to document the urban environment, as well as how to fuse them with other sensed data about the city and feed that back into the physical, collective experience in urban public spaces.

The Intel-sponsored Urban Atmospheres project is also using sensors to explore the human condition.[5] The Massachusetts Institute of Technology Cartel project provides a mobile communications infrastructure based on car-mounted communication platforms exploiting open Wi-Fi access points in a city, and it provides urban-sensing information such as traffic conditions.[6] The CitySense project, developed by Harvard, the city of Cambridge, and BBN Technologies (www.bbn.com/technology/networking/citysense) provides a static sensor mesh offering similar types of urban sensing data feeds.[7]

References

1. T. Abdelzaher et al., "Mobiscopes for Human Spaces," *IEEE Pervasive Computing*, vol. 6, no. 2, 2007, pp. 20–29.
2. V. Tuulos, J. Scheible, and H. Nyholm, "Combining Web, Mobile Phones and Public Displays in Large-Scale: Manhattan Story Mashup," *Proc. 5th Int'l Conf. Pervasive Computing*, LNCS 4480, Springer, 2007, pp. 37–54.
3. A. Kansal et al., "SenseWeb: An Infrastructure for Shared Sensing," *IEEE MultiMedia*, vol. 14, no. 4, 2007, pp. 8–13.
4. A. Parker et al., "Network System Challenges in Selective Sharing and Verification for Personal, Social, and Urban-Scale Sensing Applications," *Proc. 5th Workshop Hot Topics in Networks* (HotNets-V), 2006.
5. E. Paulos and T. Jenkins, "Urban Probes: Encountering Our Emerging Urban Atmospheres," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 05), ACM Press, 2005, pp. 341–350.
6. B. Hull et al., "CarTel: A Distributed Mobile Sensor Computing System," *Proc. 4th Int'l Conf. Embedded Networked Sensor Systems* (SENSYS 06), 2006, ACM Press, pp. 125–138.
7. R. Murty et al., "CitySense: A Vision for an Urban-Scale Wireless Networking Testbed," *Proc. 2008 IEEE Int'l Conf. Technologies for Homeland Security*, IEEE Press, 2008, pp. 583–588.

specified by the request, and the delivery of sampled data back to the application. Application functions, including generating requests, data analysis, and visualization logic, might be installed directly on the mobile device, run on remote servers (such as a Web application) but communicate with the mobile device via wireless gateway nodes (such as a General Packet Radio Service [GPRS] gateway or Wi-Fi access point), or be split between the mobile device and these servers. An application sampling request specifies at least one required sensor type (such as an accelerometer) and the required *sampling context* — that is, the set of conditions required for the sampling to take place, including time of day, location, and sensor orientation.

In the learn stage, we analyze the sensed data using simple statistical measures and more involved machine-learning techniques to extract higher-level meaning. We choose the data-analysis techniques to apply and the data features to analyze that best match the availability and characteristics of the sensed data (such as noisiness or incompleteness) and the target application visualization. In addition, we leverage the system's people-centric nature by using social connections between system users when possible to improve the performance of learned models (such as activity classifiers) and decrease the time it takes to learn these models.

In the share stage, the individual visualizes the learned information and optionally shares it. For example, sharing is possible within social groups or within a global community.

Sense: Exploit Mobility, Be Opportunistic

From an infrastructural viewpoint, sensing, computing, and communication resources are already widely deployed in the form of end-user electronics, enterprise and public radio access networks, and Internet backhaul. Therefore, there's no longer a need to deploy specialized custom-built hardware to collect and transport people-centric sensor data. Instead, we aim to symbiotically leverage this extant infrastructure to support people-centric applications. We also take advantage of the increasing integration of sensors into off-the-shelf consumer devices (such as the Nokia N95 mobile phone, Nokia N810 tablet PCs, and iPod Touch) to transparently sample the device's users and their environment.

Users' Roles

Using human-carried devices as a fundamental building block of the sensing system raises the question of what roles people, as sensing device custodians, should (or are willing to) play in the architecture. Deciding to what extent custodians should be conscious, active participants in meeting application sensing requirements has significant design implications, especially in terms of defining the fundamental research challenges to implementing a robust, scalable, and secure system. We label the two end points of the design spectrum of custodian awareness and involvement[2] as *opportunistic*[1] and *participatory*.[3]

With participatory sensing, the custodian consciously opts to meet an application request out of personal or financial interest. A participatory approach incorporates people into significant decision stages of the sensing system, actively deciding which application requests to accept, what data to share, and to what extent privacy mechanisms should be allowed to impact data fidelity. Because humans make most of the tough sensing and privacy decisions, a purely participatory system design focuses on tools that help people share, publish, search, interpret, and verify information collected using a device. Purely participatory sensing places many demands on involved device custodians (for example, prompting via their device GUI for authorization to take a sound sample or share a particular image sample), which restricts the pool of willing participants. People's tolerance in enduring interruptions on behalf of applications limits the number and request load of concurrent applications that we can likely support. Furthermore, under the participatory approach, an application must have a critical mass of community appeal. These factors might combine to limit both an application's scale and the diversity of applications that a purely participatory people-centric network could support.

In the MetroSense Project, we emphasize an opportunistic approach, which shifts the burden of supporting an application from the custodian to the sensing system by automatically determining when devices can be used to meet application requests. In this paradigm, custodians configure their devices to let applications run (subject to privacy and resource usage restrictions), but they might not be aware which applications are active at any given time. Instead, a custodian's device is used whenever its state (geographic location, body location, and so on) matches an application's context requirements. In this way, applications can leverage the sensing capabilities of all system users without requiring human intervention to actively and consciously participate in the application, lowering the bar for applications to run in people-centric networks. To support symbiosis between the custodian and the system, sensor sampling occurs only if the privacy and transparency needs of the custodian are met. The main privacy concern is the potential leak of personally sensitive information indirectly when providing sensor data such as the custodian's location. To maintain transparency, opportunistic use of a device shouldn't noticeably impact the device's normal user experience.

Challenges of Opportunistic Sensing

Along with the aforementioned benefits, the opportunistic paradigm introduces several challenges. The opportunistic use of sensor custodian devices means that sensing is a secondary, low-priority operation on the mobile device. Consequently, the device might only be able to meet the sensing requirements defined by an application request for short and intermittent periods. Opportunistic systems also take on much more of the decision-making responsibility and are thus more complex and might use more resources. Specific challenges that must be overcome for opportunistic sensing to be feasible include

- determining the device's sampling context,
- adapting to the devices' changing resource availability and sampling context,
- achieving sufficient sensing coverage in the face of sensing target mobility, and
- sustaining custodian privacy.

We're currently investigating methods to address these challenges.

Sensing context. Sensing context is the metadata that describes the conditions to which the sensing hardware is exposed and affects both the sensor data and its ability to perform the sensing operation. Knowledge of sensing context is required as an input to a number of operations of an opportunistic sensing system with sensor sharing. It helps evaluate potential candidate sensor devices in terms of a given application request and, during servicing of application requests, indicates when sampling should be started and stopped. More generally, the sensing context is important for understanding the sampled data, especially from consumer devices in which sensing is largely a second-class citizen and samples might be taken under suboptimal conditions — for example, when the device is in a pocket or purse.

Sensor sharing. For an opportunistic sensing system to collect samples that meet a general set of application requirements (such as sensor type, location, physical orientation, and time), it must be able to adapt to the sensing devices' changing resource availability and sampling context. For example, a mobile phone might run out of memory or power or be placed in the custodian's pocket before a required light level, sound, GPS, or image sample is taken.

To help the sensing system be more robust to these changes, we're developing a sensor sharing mechanism. This approach allows application requests assigned to a particular device to borrow samples from the best-suited sensors (those matching the required sensing context and not already in use by another application on the device) of any available neighboring device. Devices exchange current context information, and data is selected from the device with a context that most closely matches the application's requirements. Given the potentially rapid dynamism of sampling context, a research challenge is determining a context-matching metric that, when used for this sharing mechanism, provides samples with high average-case fidelity with respect to the applications requirements.

Mobile target sensing. In people-centric sensing, we need to support the tracking and sensing of mobile targets (such as a noisy truck or a missing child's voice) with mobile sensing devices. There are two major challenges in building mobile event sensing systems using mobile sensors that people carry. First, mobile sensors need to be informed about the sensing target (that is, be "tasked") before sensing, but for efficiency, only those near the mobile target. Second, there's no guarantee that there will always be enough mobile sensors around the target to maintain sensing coverage.

To efficiently establish a sensing area around the target, a mobile sensor that detects the target using its sensors forwards the task to its neighbors. To recover a lost target, we estimate the area to which the target is predicted to move based on a distributed Kalman filter and then use a geocast scheme to forward the task to the sensors in the predicted area.

Privacy. Opportunistic sensing faces barriers to wide-scale adoption unless users trust the system to provide privacy guarantees on par with those provided by state-of-the-art systems. Sensing device custodians will fear that sensitive personal information will leak from both the collected data samples and the process by which the samples are collected. For example, during sensor sharing, the shared data might reveal information about a device's context. Also, sensor data such as images, sound, and accelerometer data might contain information that custodians don't wish to expose about themselves. Furthermore, even those who aren't custodians and might not be the primary sensing targets are vulnerable to an accidental compromise of privacy — a "second-hand smoke" of people-centric sensing systems. For example, an application measuring traffic noise might sample a mobile phone's microphone as the custodian stands at a busy city intersection, but the audio sample might also contain fragments of a passerby's private conversation.

Ongoing work in the MetroSense Project has begun to address these issues by providing sensing device custodians with a notion of anonymity through k-anonymous tasking.[4]

Learn: Understanding Opportunistic Data

Once the data has been sampled and delivered to the application, we must extract higher-level meaning from the raw samples. Two main challenges facing data analysis in the people-centric sensing domain[5] — and in constructing accurate inference models in general — are the lack of appropriate sensor data inputs and the time and effort that must be spent in training models that give sufficient classification accuracy. The commercial off-the-shelf (COTS) devices on which we must build our people-centric applications are heterogeneous in terms of sensing and other resources such as memory, battery capacity, and CPU power, which impacts model construction and usage. The data inputs most useful in generating high-accuracy models might not be available on all devices, requiring users of less capable devices to settle for less accurate models based on other available data features. For example, using the current state of technology, a common approach is to extract data features from a GPS sensor to generate an indoor/outdoor classifier. However, only a relatively small percentage of mobile phones in the US market today include GPS.

Hence, we must carefully consider where in the architecture classifiers should run. Two possible solutions for model creation are opportunistic feature vector merging and social-network-driven sharing of models and training data.

Opportunistic Feature Vector Merging

With this approach, we seek to push the performance of classification models possible with sensor-poor devices toward that possible with sensor-rich devices. When merging feature vectors — that is, multi-element numerical object or activity representations — data features available from more capable devices are borrowed and merged with data features natively available from a less capable device in the model-building stage, letting us build a higher accuracy model even for the less capable device. This borrowing is facilitated by opportunistic interaction, both direct and indirect, between a less capable device and a more capable device in situ.

In a direct interaction scenario, as two mobile phone users follow their daily routines, a mobile phone without GPS can borrow GPS data features from a mobile phone with GPS as an input to its indoor/outdoor location classifier. For indirect interaction, both devices collect data samples according to their respective capabilities. Subsequently, centralized matching between other features collected by both devices might provide for a binding between the feature vector collected by the phone without GPS and the GPS features collected by the GPS-equipped phone. The GPS features can then essentially be borrowed via this binding.

Social-Network-Driven Model and Data Sharing

Even when devices provide an appropriate set of data features to build accurate models, users might be required to gather a large set of training data (perhaps manually labeling it) before the applications using the models' outputs will perform at their best. The inconvenience in both the labeling of training data and the time required for model training to complete might act as disincentives to the broad-scale adoption of new people-centric applications. We propose sharing training data among users to reduce training time and labeling effort by amortizing the model training cost over all system users. However, this is likely to reduce the resulting model's accuracy because, for example, people do the same activity in many slightly different ways and might describe the same activity with slightly different labels.

With our social-network-driven sharing approach, training data is shared only within social circles, in which, we conjecture, group vocabularies and other commonalities lead to more consistent and understandable labeled training data and a higher model accuracy, while still reducing the quantity of per user training data required. Still, a careful consideration of the particular labeling problem is required in deciding within which social group sharing might be most effective. Initial results implementing these two techniques are promising.[6]

Additional Resource Considerations

In addition to model generation, resource limitations on mobile devices designed primarily for other purposes require that we carefully consider where data processing takes place. For example, due to CPU power limitations, we've noticed that running a full spectrum fast Fourier transform on a mobile phone can impact other ongoing operations and can run too slowly to keep up with the stream of sampled data. Such behavior violates our tenet of symbiosis with the device's primary user experience. Furthermore, due to local storage limitations, analysis that requires access to a large amount of historical data might not be possible without interaction with persistent storage on back-end

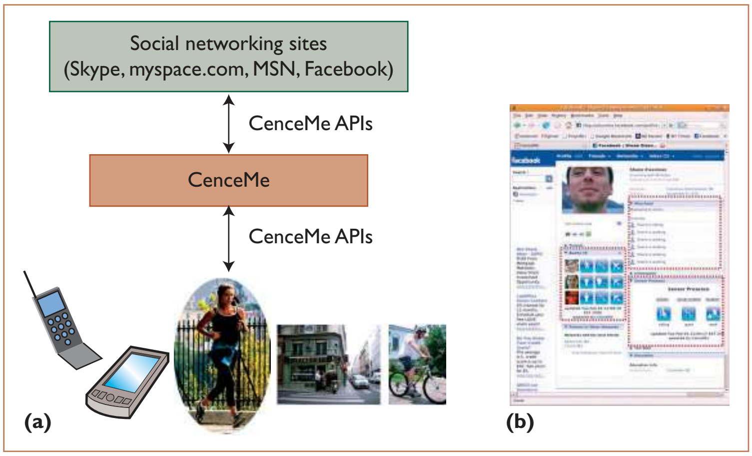

Figure 2. The CenceMe application. (a) CenceMe distills a user's sensing presence from samples taken from sensors embedded in personal mobile devices, sports equipment (such as running shoes or a bicycle), and the civic infrastructure. Users can share sensing presence with their friends through popular social networking applications. (b) We've built widgets for Facebook that allow expression of sensing presence through the friends list, the mini-feed, and a dedicated Sensor Presence display.

servers. Placing learning functionality requires a systemic view that considers mobile phone resource constraints, communication cost to the back-end servers, and the sampling rate required to detect and characterize the phenomena of interest.

Share: Enabling Social-Sensing Applications

People-centric sensor networking aims to support applications that engage the general public. This potential for interest across a broad segment of the population, in concert with our use of an opportunistic sensing design, facilitates the availability of a massive number of mobile sensing devices, in turn increasing the scope and scale of applications that we can support and improving the fidelity of sampled objects, events, and human activities. We've developed a number of applications in the MetroSense Project that incorporate personal, social, and public sensing.

CenceMe

The growing ubiquity of the Internet provides the opportunity for an unprecedented exchange of information on a global scale. Those with access to this communication substrate — especially the youth — increasingly incorporate personal information exchange into their daily routines via technologies such as email, blogs, instant messages, SMS, video sharing, social network software, and voice over IP (VoIP). Yet, the question of how to incorporate personal sensing information such as human activity inferencing into these applications has remained largely unexplored. Although existing communication forums let users exchange text, photos, and video clips, we believe a more richly textured user experience can be provided by integrating *automatically* harvested, processed, and shared sensor data into the mix.

With the CenceMe application,[7] we distill this sensed data (see Figure 2a) into what we call a user's *sensing presence*, a virtual representation of users' status in terms of their activities (sitting, walking, or meeting friends), disposition (happy, sad, or okay), habits (at the gym, coffee shop, or at work) and surroundings (noisy, hot, or bright).

We're evolving a prototype implementation of CenceMe that lets members of social networks access historical traces of their own data and, more powerfully, securely share their sensing presence among their buddies. For users on the go, we've implemented a client to show current buddy sensing presence on the GUI-based displays of most new mobile phones, using a set of simple and intuitive icons representing, for example, a user's activity and location. We include this sensing presence snapshot, along with a more complete, browseable representation of users' sensing presence and their buddies via the CenceMe Web portal. Examples include archived historical traces, comparisons of presence or social attributes with friends and extracted patterns and features of importance in a user's life routine.

The real power of automatically inferencing sensing presence, however, is the ability for a user to configure CenceMe to export this sensing presence, without direct intervention, across his or her online social networks. This has only recently been made possible through the release of developer APIs for Facebook, Skype, Pidgin, MSN Messenger, and the like. We've implemented several widgets users can add to their Facebook account (see Figure 2b) to share various representations of sensing presence with their Facebook buddies. CenceMe users share data according to CenceMe group membership policies set through the Web portal. CenceMe buddies are defined by the combination of buddy lists imported from the social networking application accounts a user registers with CenceMe.

Thus, CenceMe inherits group structures users have already created for other applications, but it also lets them specify more sophisticated group privacy policies within CenceMe.

Global Sharing in Virtual Worlds

Virtual world simulators such as Second Life represent one of the new frontiers in online entertainment and business services. People lead virtual lives in these alternative worlds using personal avatars. Bridging real life and these virtual worlds is challenging, but it enables new application scenarios for these systems via public sensing. We can use the sensors embedded in commercial mobile phones to infer real-world activities, which we can in turn reproduce in public virtual environments, sharing with the (virtual) world (see Figure 3).

Although previous research has brought static objects from the real to the virtual world, we are the first to bring people's real-world sensing presence to active subjects (that is, their avatars) in the virtual world.[8] People's sensor data can be rendered in the virtual world anywhere on the spectrum between reality and fantasy. That is, arbitrary mappings between sensed physical data and the avatar actions, appearance, and location are possible. Furthermore, the connection between the physical and virtual worlds need not be only one way, and we envision that users might receive communication (such as emails, instant messages, or SMS) or actuation triggers (such as a mobile phone vibration) to indicate the status or environment changes experienced by their avatar in the virtual world.

As an initial step, we've implemented activity recognition and voice-detection classifiers that run on a mobile phone, acting on data from local embedded sensors. We've also built a data bridge using available APIs to control a user's avatar in Second Life (see Figure 3).

Figure 3. Second Life integration with the physical world. Accelerometer data is collected from a person's mobile phone and classified into the activity states of sitting, standing, or running. These states are then injected into Second Life via the mobile phone object the avatar carries (inset). Second Life users define the profile for their avatar to interpret and render these incoming activity states. For example, in the figure, the user has mapped sitting, standing, and running to yoga-floating, standing, and flying, respectively.

BikeNet

BikeNet is a recreational application that contains elements of personal, social, and public sensing. There's substantial interest in the mainstream recreational cycling community in collecting data quantifying various aspects of the cycling experience, mirroring the broader interest in fitness metrics among exercise enthusiasts and other health-conscious individuals. Existing commercial bike-sensing systems targeting this demographic measure and display simple data such as wheel speed and provide simple inferences such as distance traveled and calories burned. These systems have become increasingly more sophisticated and miniaturized.

We've designed and implemented a system prototype reflecting a future in which wirelessly accessible sensors are commonly embedded in commercially manufactured bicycles, and the cyclist's mobile device (such as a mobile phone) interacts with these sensors during the ride to quantify aspects of cycling performance and environmental conditions. In terms of personal sensing, we view this system as akin to the Nike + iPod kit, a system for recreational runners that logs exercise history. The BikeNet application measures several metrics to give a holistic picture of the cyclist experience: current speed, average speed, distance traveled, calories burned, path incline, heart rate, CO_2 level, car density surrounding the cyclist, and galvanic skin response (a simple indicator of emotional excitement or stress level).[9] All data the system senses is stamped with time and location metadata. This data is provided to the cyclist immediately, for example, via the mobile phone's LCD, but it's also uploaded to a personal repository on remote BikeNet servers for long-term archiving and later trend analysis (such as cycling performance and personal health).

The BikeView portal (http://bikenet.cs.dartmouth.edu) provides a personal sensing re-

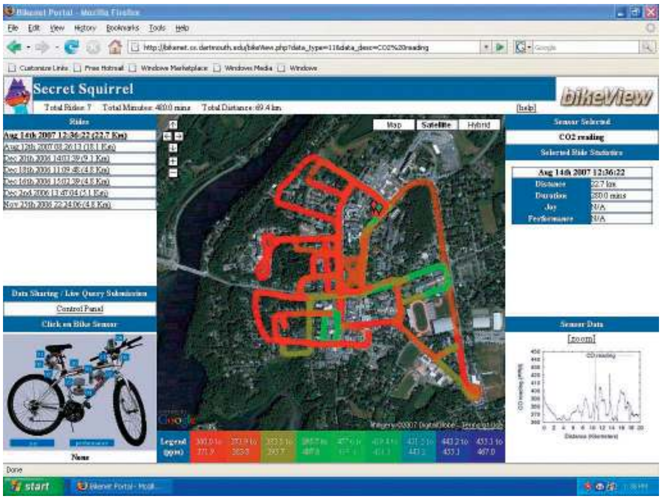

Figure 4. The BikeView application. The portal provides personal access to archived cycling data, which can be socially shared with cyclists or used to support a public sensing initiative. This CO_2 map is the result of multiple users' data merged to form a complete map of Hanover, New Hampshire.

pository for all ride statistics. Additionally, it lets users share information via real-time requests, a function most likely enabled within social groups such as for race-time management within a cycling team or for rough tracking of a family member to know when to order the pizza.

Sharing aggregate statistics and route rankings (optionally stripped of identifying information) is facilitated within the cycling community group. In addition, BikeNet facilitates public sensing and sharing by letting multiple users merge their individual data, for example, to create pollution, allergen, and noise maps of their city. Such a map not only provides a way to learn about the safest and healthiest ways to get around town, but it also might provide the basis for political action to improve the city. Figure 4 shows such a map, built through the BikeView portal, of CO_2 readings combined from users of our prototype BikeNet system mapping Hanover, New Hampshire.

Beyond the applications of people-centric sensing we've discussed thus far, we're working to push the MetroSense vision into other domains. Promising directions include the healthcare industry, where a people-centric sensing approach can facilitate grassroots monitoring and sharing of automatically collected health data. We believe this approach can be especially helpful in addressing the healthcare needs of populations currently underserved by the existing health infrastructure. We're also interested in understanding how people use people-centric sensing systems. What types of data are people least comfortable having automatically gathered, interpreted, and shared? Can the way people use such systems help researchers learn about (possibly hidden) social structures in the user community? Visit the MetroSense Project (http://metrosense.cs.dartmouth.edu) for the latest in people-centric sensing research.

Acknowledgments

This work is supported in part by Intel, Nokia, US National Science Foundation grant number NCS-0631289, the Institute for Security Technology Studies (ISTS) at Dartmouth College, and members of the MetroSense Project. ISTS support is provided by the US Department of Homeland Security under grant number 2006-CS-001-000001. The views and conclusions contained in this document are those of the authors and don't necessarily represent the official policies, either expressed or implied, of the US Department of Homeland Security.

References

1. A.T. Campbell et al., "People-Centric Urban Sensing," *Proc. 2nd ACM/IEEE Int'l Conf. Wireless Internet*, 2006.
2. N.D. Lane et al., "Urban Sensing Systems: Opportunistic or Participatory?" *Proc. 9th Workshop Mobile Computing Systems and Application*s (WMCSA 08), ACM Press, 2008.
3. A. Parker et al., "Network System Challenges in Selective Sharing and Verification for Personal, Social, and Urban-Scale Sensing Applications," *Proc. 5th Workshop Hot Topics in Networks* (HotNets-V), 2006; www.sigcomm.org/HotNets-V/parker06network.pdf.
4. A. Kapadia et al., "AnonySense: Opportunistic and Privacy-Preserving Context Collection," *Proc. 6th Int'l Conf. Pervasive Computing*, LNCS 5013, Springer, 2008, pp. 280–297.
5. J. Lester, T. Choudhury, and G. Borriello, "A Practical Approach to Recognizing Physical Activities," *Proc. 4th Int'l Conf. Pervasive Computing*, LNCS 3968, Springer, 2006, pp. 1–16.
6. N.D. Lane et al., "Cooperative Techniques Supporting Sensor-Based People-Centric Inferencing," *Proc. 6th Int'l Conf. Pervasive Computing*, LNCS 5013, Springer, 2008, pp. 75–92.
7. E. Miluzzo et al., "CenceMe — Injecting Sensing Presence into Social Networking Applications," *Proc. 2nd European Conf. Smart Sensing and Context* (EuroSSC 2007), LNCS 4793, Springer, 2007, pp. 1–28.

8. M. Musolesi et al., "The Second Life of a Sensor Integrating Real-World Experience in Virtual Worlds using Mobile Phones," *Proc. 5th Workshop on Embedded Sensor Networks* (HotEmNets 08), ACM Press, 2008; www.cs.dartmouth.edu/~sensorlab/pubs/SL-HotEmnets08.pdf.
9. S.B. Eisenman et al., "The BikeNet Mobile Sensing System for Cyclist Experience Mapping," *Proc. 5th ACM Conf. Embedded Networked Sensor Systems*, (SENSYS 07), 2007, ACM Press, pp. 87–101.

Andrew T. Campbell is an associate professor of computer science at Dartmouth College, where he leads the Sensor Lab and the MetroSense Project and is a member of the Institute for Security Technology Studies (ISTS). His research interests include people-centric sensing and open spectrum wireless networks. Campbell has a PhD in computer science from Lancaster University, England. He received the US National Science Foundation Career Award for his research in programmable mobile networking. Contact him at campbell@cs.dartmouth.edu.

Shane B. Eisenman is a PhD candidate in the electrical engineering department at Columbia University as well as a member of the Control, Management and Telemedia Group and the Armstrong Project at Columbia University and a member of the Sensor Lab and the MetroSense Project at Dartmouth College. His research interests include wireless networking and mobile computing, with a particular interest in human-centric sensing, computation, and communication. Eisenman has an MSc in electrical engineering from Columbia University. Contact him at shane@ee.columbia.edu.

Nicholas D. Lane is a PhD candidate in Dartmouth College's Computer Science Department and a member of the MetroSense project and the Sensor Networks group at Dartmouth. His research interests include wireless sensor networks, particularly urban, and people-centric sensing systems. Lane has an MEng in computer science from Cornell University. Contact him at niclane@cs.dartmouth.edu.

Emiliano Miluzzo is a PhD candidate in the computer science department at Dartmouth College, a member of the Sensor Networks Group at Dartmouth, and involved in the MetroSense Project. His research interests are in wireless sensor networks, in particular distributed mobile sensing systems as well as architecture and application design for emerging people-centric sensing systems. Miluzzo has an MSc in electrical engineering from University of Rome La Sapienza. Contact him at miluzzo@cs.dartmouth.edu.

Ronald A. Peterson is a senior programmer in the Dartmouth College Computer Science Department and owner of Peterson Enterprises, a small software company. Peterson has a BA in physics from Lawrence University. Contact him at rapjr@cs.dartmouth.edu.

Hong Lu is a PhD student at Dartmouth College. His research interests include networked systems, mobile and ubiquitous computing, and wireless sensor networks. Lu has an MS in computer science from Tianjin University, China. Contact him at hong@cs.dartmouth.edu.

Xiao Zheng is a PhD student in computer science at Dartmouth College. His research interests include mobile networks and wireless sensor networks. Zheng has an MSc in computer science from University of Northern British Columbia, Canada. Contact him at zhengx@cs.dartmouth.edu.

Mirco Musolesi is a postdoctoral research fellow in the Department of Computer Science at Dartmouth College and a fellow of the Institute of Security Technology Studies (ISTS) at Dartmouth. His research interests include delay tolerant networking, mobile networking and systems, wireless sensor systems, mobility modeling, and social-network-based systems. Musolesi has a PhD in computer science from University College London. Contact him at musolesi@cs.dartmouth.edu.

Kristóf Fodor is a visiting Fulbright PhD student at Dartmouth College, cosupported by the Institute for Security Technology Studies (ISTS), a PhD candidate in computer science at the Budapest University of Technology and Economics in Hungary, and a graduate student in engineering economics at the University of West Hungary. His research interests include the design of architectures, protocols, and algorithms for self-organizing networks and mobile sensor networks. Fodor has an MSc in computer science from the Budapest University of Technology and Economics. Contact him at fodor@cs.dartmouth.edu.

Gahng-Seop Ahn is a PhD candidate in the electrical engineering department at Columbia University and involved in the MetroSense Project at Dartmouth College. His research interests are in wireless mobile networking and mobile sensing including service differentiation, medium access control, and mobile sensing and tracking systems. Ahn has an MSc in electrical engineering from Columbia University. Contact him at ahngang@ee.columbia.edu.