

NoiseSPY: A Real-Time Mobile Phone Platform for Urban Noise Monitoring and Mapping

Eiman Kanjo

© Springer Science + Business Media, LLC 2009

Abstract In this paper we present the design, implementation, evaluation, and user experiences of the NoiseSpy application, our sound sensing system that turns the mobile phone into a low-cost data logger for monitoring environmental noise. It allows users to explore a city area while collaboratively visualizing noise levels in real-time. The software combines the sound levels with GPS data in order to generate a map of sound levels that were encountered during a journey. We report early findings from the trials which have been carried out by cycling couriers who were given Nokia mobile phones equipped with the NoiseSpy software to collect noise data around Cambridge city. Indications are that, not only is the functionality of this personal environmental sensing tool engaging for users, but aspects such as personalization of data, contextual information, and reflection upon both the data and its collection, are important factors in obtaining and retaining their interest.

Keywords noise mapping · pervasive sensing · mobile computing · online mapping · pollution monitoring · ubiquitous computing.

1 Introduction

Sound is essential to our daily lives, but noise is not. Noise can be defined as sounds or noises that are loud, annoying and harmful to the ear. It is a source of irritation and stress for many people and can even damage our hearing if it is loud enough. Many of us are exposed to stressful levels of noise indoor and outdoor.

Environmental noise is unwanted sound from different sources, among road and rail traffic, construction work,

aircraft, sports events, leisure parks, and homes etc. Each has different characteristics and poses specific problems, for example, noise can annoy, disturb sleep and affect the health of people of different age groups [8].

Noise is on the increase in our society, in particular environmental noise cause by traffic. Road traffic is one of the most widespread sources of noise and there are various guidelines and laws regulating acceptable levels [11]. Research shows that over 40% of the population are bothered by noise from road traffic [9]. There is a demand for accurate data about noise exposure levels. In 2006, a position paper by the Commission [10] has stressed that every effort should be made to obtain accurate real data on noise sources.

Today's noise measurements in urban areas are mainly carried out by designated officers that collect data in a location of interest for successive analysis and storage, using a sound level meter or similar device. This manual collection method using expensive equipment does not scale as the demand for higher granularity of noise measurements in both time and space increases [15].

We claim it is not necessary to put thousands of fixed sensors out in to the environment; rather we can find out the nature of spatial and temporal variation by using a small and affordable tracked mobile sensors.

We believe that pervasive computing can ultimately engage people in mass participation environmental campaigns, raising awareness of environmental issues, supporting education, activism and democracy, and delivering environmental data on a scale never before possible. Mobile computing has the potential to allow both experts and the public to collect and understand environmental data such as air pollutants in urban areas with an accuracy and cost-efficiency that current noise assessment procedures cannot afford.

In this paper, we describe our noise monitoring system "NoiseSpy" that allows users to explore a city area while collaboratively collecting and visualising urban noise levels in real-time. We *gather information* about the environment

E. Kanjo (✉)
Computer Laboratory, University of Cambridge,
Cambridge, UK
e-mail: eiman_kanjo@hotmail.com

on a greater scale, we *inform the debate* by conveying environmental knowledge in new ways, and we demonstrate these ideas within an urban noise monitoring application.

NoiseSpy represents the first working environmental noise sensing system based on mobile phones as it uses the phone's microphone to assess sound levels in the surrounding environment. The sensor is tracked using the Global Positioning System (GPS). The aim of the NoiseSpy is to create an open platform to measure, annotate and localize noise pollution by actively involving individual citizens who use their mobile phones as noise sensors. We report on lessons learned and design issues arising from the implementation and use of our research prototype.

Our user study for this system was two-fold. First we wanted to demonstrate that mobile phone enabled sensing can be used at city level where it is possible to share data between users. This city based mode is facilitated by dynamic update of real-time noise map while maintaining a degree of anonymity for each individual journey. In the second one we looked at individual journeys to assess personal exposure level, in particular, for users who are more prone to the risk of serious noise pollution such as cycling couriers.

We think that the availability of personalized environmental information, together with the ability to compare it with other's exposure, can do more to raise awareness about environmental issues than global information provided by environmental agencies. NoiseSpy will form the basis of the action plans required by EU Noise Directive 2002/49/EC. Unlike current noise pollution data coming from static sensors installed in specific places, 'people-centric' data could have great value to scientists to better understand the noise pollution issue from a human point of view.

In the following sections, we discuss the system architecture, design, and implementation of NoiseSpy. We describe our experiences deploying an online mapping interface which is a community oriented web-based portal for visualization and sharing of urban noise information in real-time. We outline trials we have undertaken and describe an evaluation of noise data including sensing accuracy (calibration) and performance. Related work on mobile sensing is also discussed. Finally we conclude and present plans for future work both with Mobile Noise monitoring.

2 Noise health impact

Noise contributes greatly to diminishing city dwellers' quality of life. In particular exposure of people to noise levels above 65 dBA can cause severe health problems. Road traffic is a main source of noise in urban areas, accounting for about 80% of total noise pollution. In the European Union about 40% of the population is exposed to road traffic noise with an equivalent sound pressure level

exceeding 55 dB(A) daytime, and 20% are exposed to levels exceeding 65 dB(A).

Noise is subjective and different people react to it in different ways. What can cause annoyance to some people maybe barely noticeable by others. Noise can have the effect of causing people to feel annoyed simply because the noise is audible. As the noise level increases it can interrupt conversation, disturb sleep and, in extreme conditions, cause physical damage to those affected [2].

Subjected to 45 decibels of noise, the average person cannot sleep. At 120 decibels the ear registers pain, but hearing damage begins at a much lower level, about 85 decibels. The duration of the exposure is also important. There is evidence that among young children hearing sensitivity is decreasing year by year because of exposure to noise, including excessively amplified music [25]. Apart from hearing loss, such noise can cause lack of sleep, irritability, heartburn, indigestion, ulcers, high blood pressure, and possibly heart disease. One burst of noise, as from a passing truck, is known to alter endocrine, neurological, and cardiovascular functions in many individuals; prolonged or frequent exposure to such noise tends to make the physiological disturbances chronic. In addition, noise-induced stress creates severe tension in daily living and contributes to mental illness [9].

Noise is just one of the challenges facing urban birds. Until recently, concern has centred on other forms of pollution, including chemicals and light. Yet there is no doubt that noise has already contributed to a decline in the diversity of bird species around cities and major roads. Frank Rheindt at the University of Würzburg, Germany, measured the diversity of different bird species near a busy local highway and found a dearth of birds that sing at lower frequencies [6].

3 Background: assessment of environmental noise levels

A microphone converts pressure fluctuations into an equivalent electrical signal (sound pressure P) that can be post-processed to compute the loudness of the noise source that generated the acoustic wave. Average loudness levels over long periods of time are commonly used as noise indicators.

The instantaneous sound pressure level P_v of a sound is usually expressed in logarithmic units with respect to a given reference pressure level and is computed according to the following equation:

$$L_{SP} = 10 \text{Log} \left(\frac{P_v^2}{P_{ref}^2} \right) \quad (dB) \quad (1)$$

where $P_{ref}=0,00002 \text{ N/m}^2$, the reference value for airborne sound and P is the average sound pressure level.

In sound, decibels measure a scale from the threshold of human hearing, 0 dB, upward towards the threshold of

pain, about 120–140 dB. Because decibels are such a small measure, they are computed logarithmically and cannot be added arithmetically. An increase of 10 dB is perceived by human ears as a doubling of noise. Most sounds in our day-to-day environment have sound pressure levels on the order of 30 to 100 dB.

Even if there are several different standard procedures for the computation of the above defined noise indicators, a few important issues like, e.g., the spatial distribution of the measurement points or the necessary acoustic data processing, must always be carefully considered when measuring sound pressure levels. The average human ear senses tone resulting from sound oscillation in frequencies between 20 and 20,000 Hertz (Hz). Indeed, since the human ears are better equipped to hear mid and high frequencies than low frequencies, humans find mid and high frequency noise to be the most annoying. Because the human ear doesn't respond to all of these frequencies equally well, weightings are applied to the sound levels measurements (in the frequency domain) to take into account this selective behavior of the human hearing system.

Among the available standard weighting methods the European regulation requires the use of the A-weighting function, originally defined in [13] and then adopted in numerous international standards. The A-weighted sound level pressures and noise indicators are indicated in A-weighted decibels or dB (A).

A-Weighting plot has frequency on one axis and a measure of sound pressure on another in decibels (dB). The charts display the relationship between frequency and our perception of equal loudness (see Fig. 1). Using this filter, the sound level meter is thus less sensitive to very high and very low frequencies.

4 Measurement of environmental noise levels using NoiseSpy

To be used as noise pollution sensing on mobile devices, Mobile phones must be able to compute the noise indicators defined in the previous section. On Nokia NSeries phones it

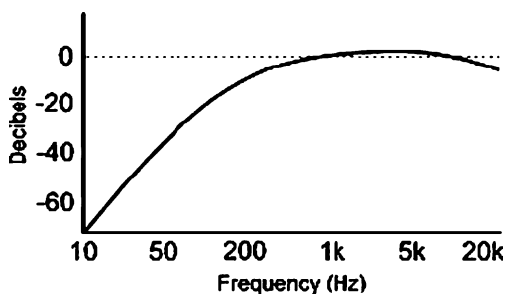


Fig. 1 A-weighting curve

is possible to relay audio packets into the audio stream¹ and have these processed as soon as they are received by the underlying subsystem. The overall logical architecture of the system is shown in Fig. 2.

First, the sound monitor component of NoiseSpy system receives $x=16$ -bit PCM (Pulse-Code Modulation is a digital representation of an analog signal) at speed 8,000 Hz. Samples are signed integers in the range $-32,768$ to $32,767$, zero means silence, the further from zero, the louder the sound. Figure 3, reports the voltage response of the mobile phone microphone to environmental sounds in an open street. These responses are raw signal samples, read from the microphone at a rate of 8 KHz.

Each block of samples is stored in a memory buffer; the mean average of these samples is calculated in a single sound value P_v , the mean value is shown in formula (1):

$$P_v = \sum_{i=1}^n x_i \tag{2}$$

where $n=8,000*\tau$ and τ is update rate of the sound level.

For example, if the buffer size is 3,200 byte, the buffer will be filled with 1,600 samples every $\tau=1,600/8,000=0.2$ s, providing an update rate of $l = 5$ sound level measurements per second.

We use decibel to scale the loudness level P_v , received by the phone microphone, as shown in Eq. 1.

As mentioned in the previous section, the computation of noise indicators requires the noise measurement algorithm to apply A-weighting filtering.

We have fully implemented the A-Weighting filter on the phone which can be described as a function of the frequency f by the following equations and can be found from the following formulas (3, 4, 5, 6), (dBA is the A-Weighted sound measurement):

$$W_{A1} = 10\text{Log} \left[\frac{1.562339f^4}{(f^2 + 107.65265^2)(f^2 + 737.86223^2)} \right] \tag{3}$$

$$W_{A2} = 10\text{Log} \left[\frac{2.242881 \times 10^{16}f^4}{(f^2 + 20.598997^2)^2(f^2 + 12194.22^2)^2} \right] \tag{4}$$

$$W_A = W_{A1} + W_{A2} \tag{5}$$

$$dBA = W_A + dB \tag{6}$$

¹ An audio stream is a continuous flow of audio data that has no clearly defined beginning or end.

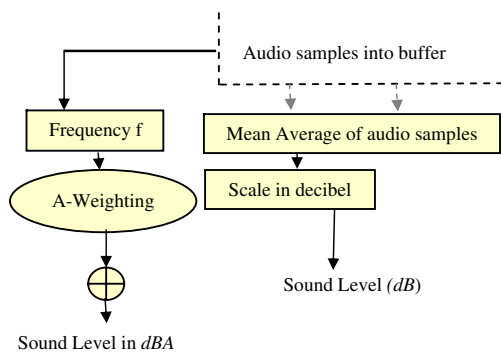


Fig. 2 Logical architecture for NoiseSpy Sound measurement

Initial calibrations of the environmental sensors was carried out by co-locating sensors with sound meters provided by Cambridge city council (IC). Figure 4, below shows the traces of the calibrated NoiseSpy sensor (in blue) and the Norsonic Noise meter (in pink). Two sounds in decibels dBA are plotted versus time which shows a high correlation between the two sound meters. The data collected in an office while two people engage in conversation intermittently. It is clearly visible that the deviation between the two graphs is very small.

5 Implementing NoiseSpy

Accessing the phone microphone: On Symbian OS v9.1s there are many APIs available to the developers, providing a rich Multimedia functionality set. Nokia provides an API to interface with the microphone and to manage the audio stream which is a continuous flow of audio data that has no clearly defined beginning or end. The user of an audio stream (in this case, the user is our audio processing algorithm) is able to relay audio packets into the stream and have these processed as soon as they are received by the underlying subsystem.

NoiseSpy employs standard client-server architecture (Fig. 2). The software application runs on a mobile phone, which is currently any of the Nokia Series 60 phones 3rd generation, it is written in Native Symbian C++. We have chosen Native Symbian C++ as it enables the phone software to access all the development APIs which are not available by JME. Also the JME audio API that exhibits a memory leak [24].

By default, noise data is sampled once every second however, data sampling rate can be adjusted by the user. Each noise data is combined with the last valid GPS location information such as: Latitude, Longitude, speed, bearing, UTC date and time. The phone IMEI (phone ID), user name and journey ID, phone battery level are also attached to the data.

Every time the software on the mobile phone is run, the noise data is displayed graphically on the phone screen along with the other location information as shown in Fig. 5. Also

the user is prompted to select a GPS device for location tracking the built-in GPS-chipset on the phone or you an external GPS-receiver that can be connected to the phone through Bluetooth. Once the GPS connection established, the data starts to feed into a log file on the phone memory. Every time the software on the mobile phone is run, a log file is created with a filename that includes the current timestamp. The software is made available on-line [21].

5.1 ServerSide

The database is also linked to real-time pollution, noise, traffic, weather and environmental information stored in MobSens database which is part of MESSAGE project [9]. MESSAGE project enables individuals to monitor their local environment and their private spaces (e.g. activities and health) by using mobile phones in their day to day life.

The server component of the system runs on a standard Apache server with PHP and PostGreSql database. PHP is used to script the logic on the server with PostGreSql being used for persistent storage. Also the PostGIS plugin-tool of PostGreSql is used for spatial queries.

As most mobile phone networks do not provide mobile phones with routable IP addresses, all communications requests must be initiated from the client side. These calls are sent from the client to the server over HTTP using POST requests, with the parameters being passed within the data of the POST request. The reply is then used to update the state of the client application. POST is used to send the noise data to the server. Parameters in the query string passed to a PHP script on the server takes the data and ingests it into the database. The PHP script also converts the data into XML and KML formats to be visualized on an online web interface as described in next section.

5.2 Online interface/mapping the noise

The online interface goals are introspection and awareness. We wanted to design an interface that allows users to explore their data and to make informed decisions about how they

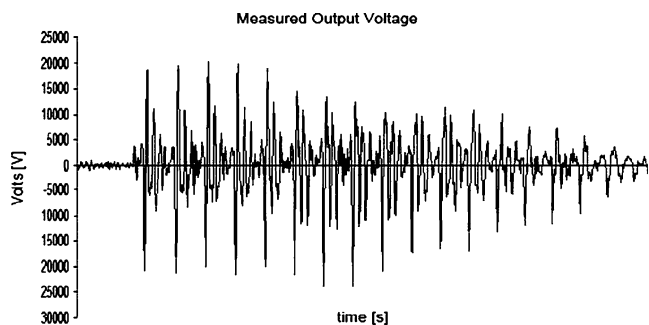
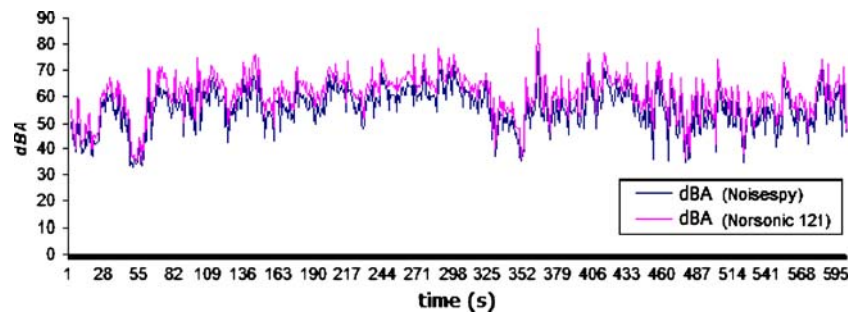


Fig. 3 Microphone voltage response to environmental sounds collected by Nokia N95 in an open street

Fig. 4 Comparison of graphs showing the discrepancy between the two noise measurements generated by the N95 phone microphone NoiseSpy and the Norsonic meter



interact with the environmental noise. The maps allow the determination of the number of people affected by different levels of ambient noise, the source of that noise (i.e. road, rail, air or industry) and the locations of the people affected.

Noise maps are created for the following purposes. Firstly, they provide an overview of the ambient noise environment in large urban areas and from major transport sources. Secondly map scan can be used to provide information on noise levels that can be linked to population data to estimate how many people are affected. This leads to the third use—and the main point of noise mapping—to help in the production of noise action plans to manage noise and reduce noise levels where appropriate.

A further benefit of having a noise model is that it can be used to assess the effects of transportation and other plans. Thus the effect of a proposed new road can be assessed and suitable noise mitigation can be designed to minimize its impact. This is particularly important in noise action planning, where a cost-benefit analysis of various options can be tested before a decision is made.

A pilot web interface with noise mapping will be made available online for public viewing as it will be described in the following sections.

5.3 Journey based visualization

Access to particular data is a matter of a policy that each user registers with a separate access control entity to the

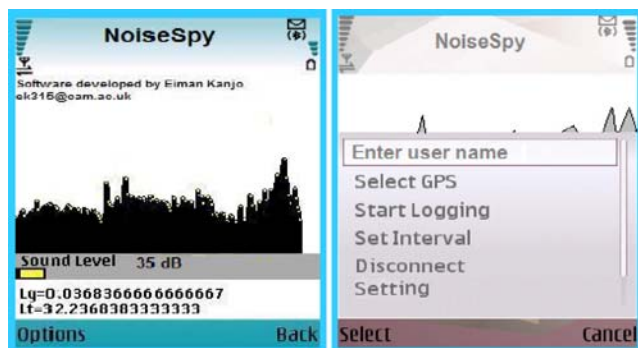


Fig. 5 The first form of on device visualization where the user sees a data/time graph while collecting data

database. Users have to use their user name and password to be able to view their own data. When the user scrolls over a trip on the pollution map, it highlights itself while all other trips are made less prominent and blend in with the background without completely disappearing. To be more specific, transparency of the trip of interest is decreased while the other trips are blurred by a constant factor that can be defined by system the user. When the user clicks on a trip on the map, the trip log automatically scrolls to the trip of interest. In the same time graph appears under the map to show noise level against time (see Fig. 6). The list of journeys is sortable, if the user has a specific date in mind, he or she can use the calendar to advance directly to that point in time. Once satisfied, the user selects trips which are displayed on a map.

5.4 City based visualisation in real-time

It is intuitive that providing better, more accurate data leads to less privacy concern for carriers, and that more privacy necessarily means less accurate data. The specifics of this tradeoff may be predicted for time and location data based on historical movement patterns, so we could allow the visualization interface to reason a certain granularity of

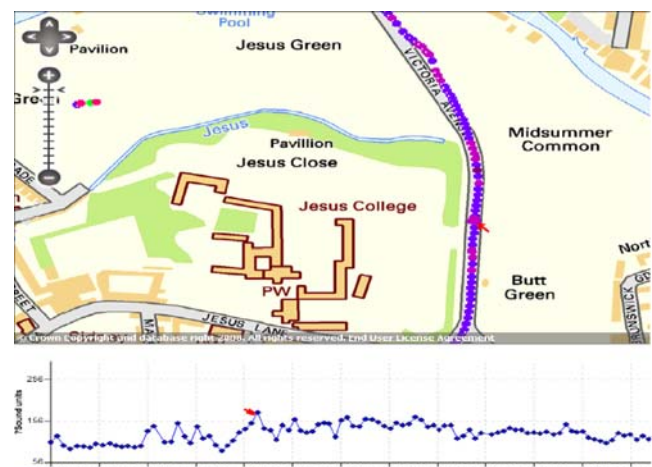


Fig. 6 Noise data collected by cyclists and overlaid on Ordnance Survey Map along with a time graph

either time or location dimensions and to blur the other dimension appropriately.

The method of plotting the data was to divide the area A in view into grid squares and to place data points into the correct square based on GPS position. When the user selects a position on the map, it automatically becomes the centre of the map view. The mean and variance for each square are then calculated and plotted. Users can alter the size of the grid squares, the data can be plotted to within the accuracy limits imposed by the GPS positional error.

The total number of cells is a function of the proportions of N . The processing time is inversely proportional to the width of the cell w , increasing with square of the cell width reduction.

Users can query the data by defining a time and date frame. They can also view historical data overlaid with real-time data.

The goal here is to provide users with as much context as possible without confusing them or cluttering the screen (see Fig. 7). In addition this mode of visualization is made visible by all users and therefore it was important to provide a degree of anonymity for each individual journey.

In both visualization modes we color-coded traces based on impact and exposure values. The color scheme used ten shades of red to blue. Higher levels of say, noise impact, were darker shades of red. Similarly, trips that had lower noise impact were with lighter shades of blue.

Further more users can view the same journey in a variety of formats, including an overlay on Google Maps (see Fig. 8), Ordnance Survey OpenSpace (see Fig. 7) and Google Earth

KML which automatically opens GoogleEarth to view a 3D model of the measurements (see Fig. 10).

6 System evaluation

Throughout the pre-trial development process the system was developed in an iterative manner. Users were asked to use the system in a variety of locations. This was done in order to test the stability and the usability of the system. This pre-trial testing phase was important as it enabled the developers to identify both user and technical issues before the system was implemented.

6.1 City based experiment

We have collected a significant amount of data from many NoiseSpy trials starting in summer 2007 but here we only present data from a two week experiment with eight users.

For this experiment we have chosen to test the system on cyclists. Participant ages range from 23 to 52; 2 users of the 8 were female. The eight cyclists' routes and the times they started their rides are not-planned. Three of them were working for a local cycling courier company who start at their office and navigate around the city of Cambridge for few hours. The others were students with no particular environmental science knowledge. We have asked the users to collect data over two or three hours every day for the two weeks.

In use Nokia mobile phone was carried around the users' necks to make the phone microphone open to the air and

Fig. 7 GoogleMaps screenshot of city based visualization showing the noise data collected over two weeks period

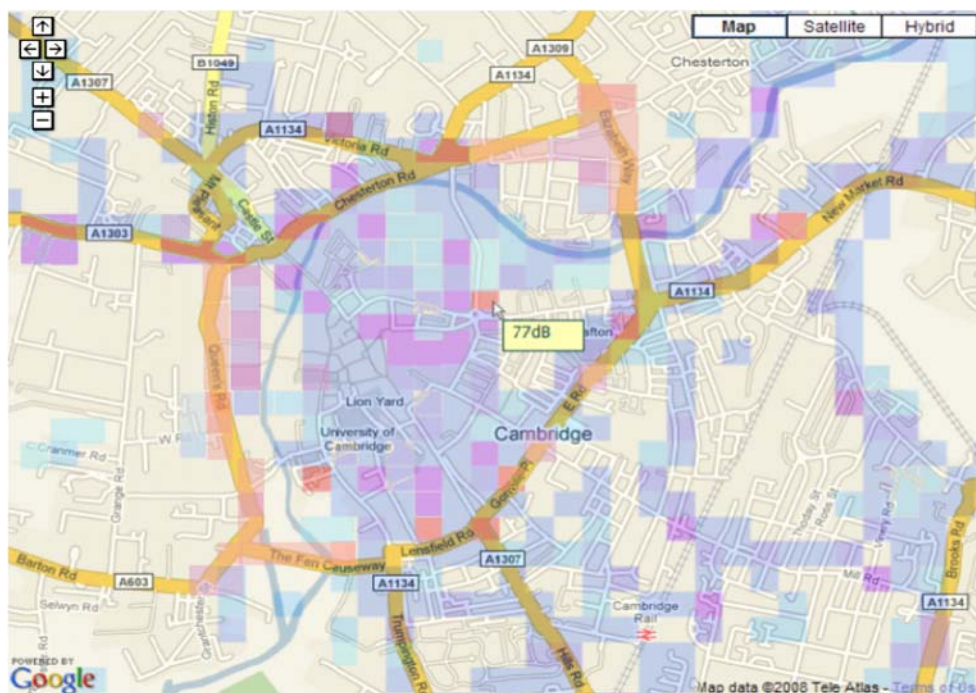


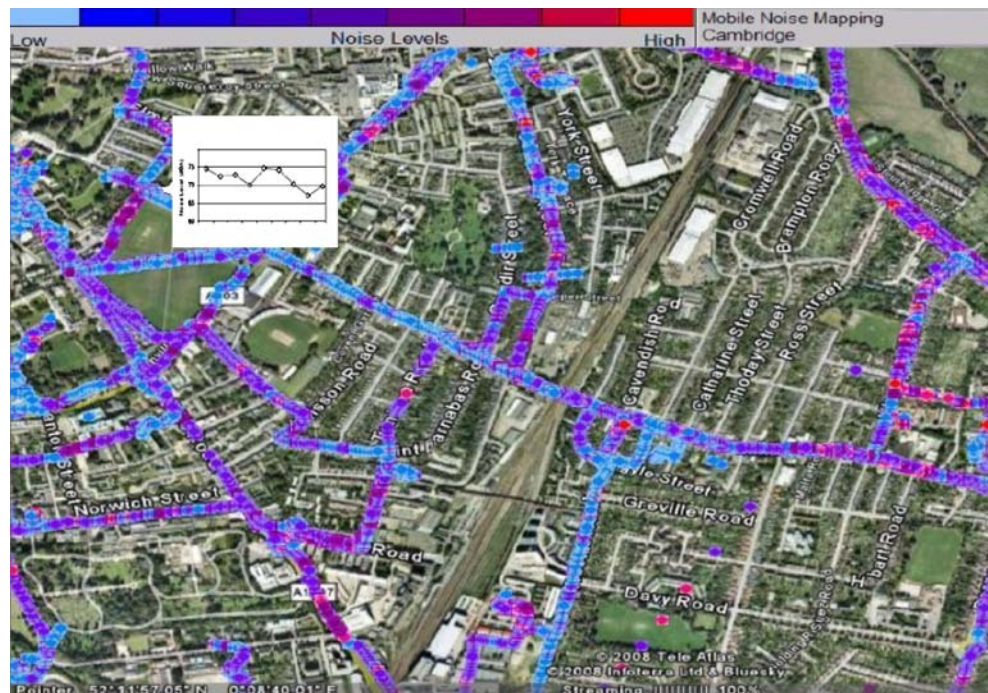


Fig. 8 A cycling courier user collecting data using NoiseSpy system

the small external GPS was placed in user’s pocket to allow hands free operation (see Fig. 8).

Nokia N95 and three N80 cell phones are used with (Holux GPSlim 236 SiRFStarIII) GPSs. The live sensor data collected by each of the bikes is directly uploaded to the backend system in real-time over GPRS. We have also run post-collection validation via manual checking and participant interviews to verify the integrity of the data set. Figure 9 shows map visualization of noise readings across Cambridge City rendered on top of Google Earth. Colors

Fig. 9 Sound measurements a trial in Cambridge, the coloured traces represent the loudness in dBA



represent individual intensity reading of noise during the trial.

6.2 Journey based experiment

The use of fixed routes for some trials was to ensure reliability and repeatability of results. Two routes along the length of Madingley Road and Huntingdon Road were used for most of the walking and cycling (see Fig. 10). Both roads have heavy traffic with many lorries, buses and bus stops. We thus expected to get high readings and high variation in readings on this stretch and GPSs are sought to perform well. Between three and five collectors went on up to four excursions each a day at around 10:30, 11:30, 14:00 for 5 days.

On other occasions, exploratory routes around the area were tried. The use of more exploratory walks was to investigate variation in the local area to see if there were anomalous highs or lows nearby.

For each trip we calculated the *Noise Exceedance Levels* which represents the percent of the run time that was spent at or above certain dB levels e.g. An *Exceedance Level 40* equal to 80 dB would mean that for 40% of the run time, the decibel level was equal to or higher than 80 dB.

Figure 10, the sound volume peaks near 80 dB when the route passes through the main intersections of the road where the automobile traffic is more prevalent.

Our experiments with the Cycling Couriers have been featured in the Science programme in the German Channel ARD, the main German Broadcaster [2] and the prestigious NewScientist magazine [23].

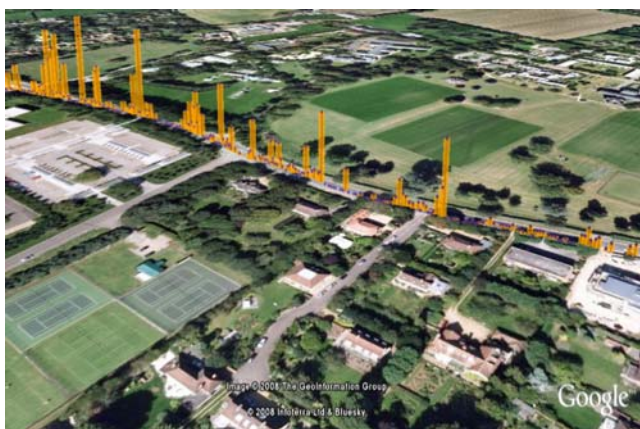


Fig. 10 Geo-logged noise (sound) measurements over Madingley Road, Cambridge, orange graph indicates wide variations in noise levels expressed in decibels (dBA)

7 Discussion

7.1 Overall results and quality of data

During the sampling period using the collection methodology outlined above, we were expecting up to 100,000 lines (since we asked the user to sample data at one second). However, 72,626 noise data entries in approximately 50,000 different positions around Cambridge city 72,626 lines of data hit the database. Some of the data was lost due to the following encountered problems:

- GPRS disconnection in some parts of the city,
- GPS not being switched on.
- NoiseSpy phone software being switched off accidentally while navigating other options on the phone through the mobile.

Almost all the participants find the idea of monitoring and viewing detailed information about noise levels in their local environment are close to compelling, useful, and fun. In particular, noise monitoring and the ability to view a graph of noise levels on the phone were their favorite beside online mapping tools and location tracking.

This pattern is confirmed in Fig. 11, where the participants favorite feature for different hours of the day derived from the analysis of system logs on the backend is shown.

By displaying noise levels as colours on the map at the locations they were generated, it was intended that the user could understand of noise in the physical area using the map representation. With the system described here, noise values become the main features, even though the map no longer fully or precisely represents that landscape. The reason for this is that the amount of traffic present at the time each location was visited is crucial to the perception and experience of that place. Thus, a location is relevant here not just as a place, but as a place and time with its

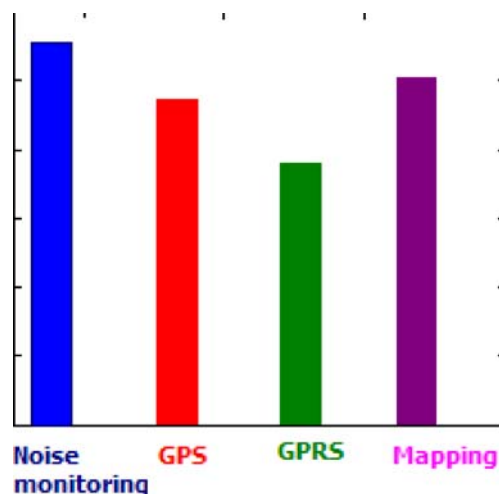


Fig. 11 A histogram showing participant's favorite features of the NoiseSpy system

associated traffic density, something record not directly represented by a map.

GPRS, like other wide-area wireless networks, exhibits many of the following characteristics: low and fluctuating bandwidth, high and variable latency and occasional loss of data. To gain clear insight into the characteristics of the real-time data upload in NoiseSpy system, we have conducted a series of experiments to evaluate GPRS performance in our system. For simplicity we have used O2 as the main network operator. We have asked 5 of our users to collect noise data over one week at different upload rate ranging from 1 to 10 s. As a result of this experiment, we observe that the signal quality increases when the data rate is reduced, as in Fig. 12.

With further attention to design of the equipment, physical and software reliability could be improved, but

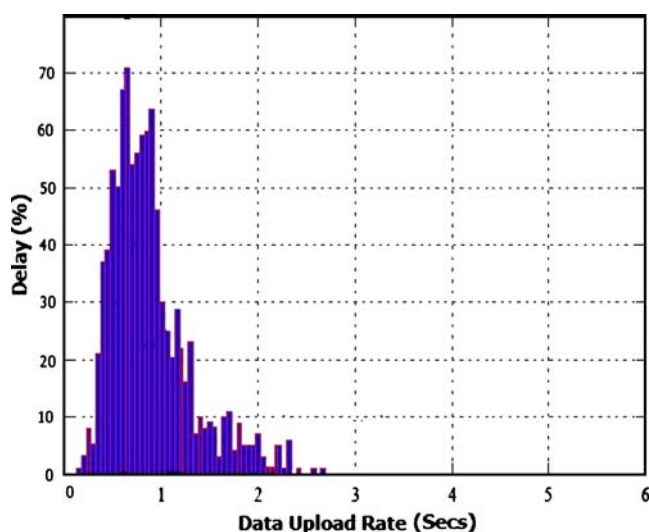


Fig. 12 A plot representing delay in real-time data upload at different data transfer rates

we also note that in longer trials, infrequent unreliability of the majority of device or persistent unreliability of a small number of devices is not a problem because the user is only passively collecting data, and the density of collection will mean that readings from other devices will compensate.

8 Challenges

8.1 Environmental noise collection

- Background noise, conversations and other activities may affect the intelligibility of the sound measurement when the background noise level is close to the level of the noise source of interest.

Figure 13 shows two noise graphs recorded by two cyclists moving at the same speed. And although they were riding on the same side of the road (one after the other) it is clear that noise levels were different at some points. Both users and technical factors that can affect NoiseSpy performance such as:

- Performance was affected by the weather (e.g. wind and rain).
- Noise interference from user movements (e.g. while walking or cycling)
- When the user is engaged in a conversation while collecting environmental data

In the slightly longer term, noise levels may be higher in peak periods when the roads are busy, and lower in off-peak periods. Then again, there is a greater volume of activity from more people and traffic in the day-time than in the evening or at night.

Noise propagation through the air is complex and very difficult to model accurately. As stated earlier, traffic is the

main source of outdoor noise. The obvious feature of traffic is that, overall, it moves. Thus, not only is noise movement difficult to predict but the source is continually moving.

The noise levels over an area will be varying all the time. For example, noise levels may rise as a vehicle approaches, and reduce again after it has passed. This would cause a short-term variations in noise level. For example, one participant has walked past a quiet area and recorded high noise reading by a passing car at that point. Also high wind has caused noise levels to build up near junctions and open areas. As participants start to collect more data it becomes interesting to look for trends and longer patterns.

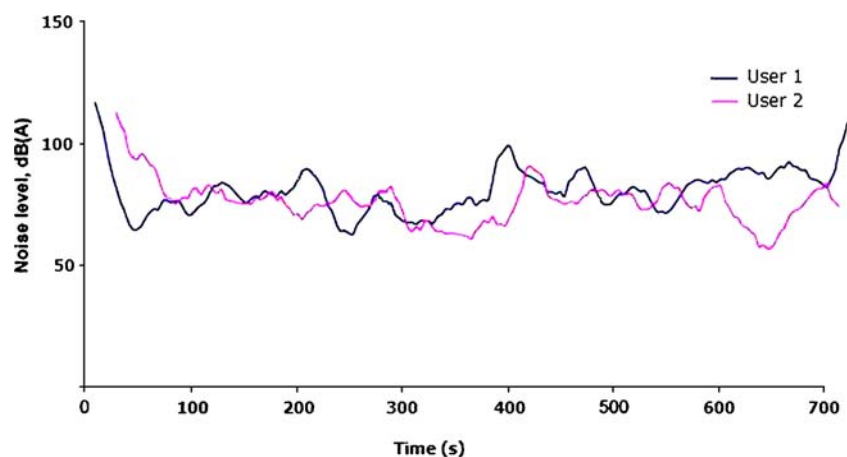
When it is not possible to eliminate or reduce the background noise, a correction factor may be applied to the overall measurement to find the level of the noise of interest.

In future trials, we hope to be able to answer questions such as how much variation is there in an area, and over what scale do changes occur; questions that are impossible to answer with a fixed sensor network without deploying many thousands of sensors.

8.1.1 Unconstrained participation

Even though the ubiquity of mobile phones makes mass participation feasible, as attempted in, it remains questionable how the general public can be motivated to voluntarily do so. How to involve the hidden majority of citizens who do not participate in local organizations and want to use such technology for personal purposes? In our noise pollution context, the goal is to not force people to make their measures public and thus to always contribute to a collective mapping process by allowing users to choose if they want to share or not all or a part of their measurements. And also simplicity, affordability and the benefit to the local environment will motivate the users to collect data using our system.

Fig. 13 Noise levels comparison curves recorded by two cyclists moving at the same speed on one side of a busy road in Cambridge



In addition, we are currently porting the phone software to other platforms such as Google Android and iPhone to make the software accessible by wider range of people.

8.1.2 GPS accuracy

With the GPS data, the situation is complicated by the fact that the positional accuracy is not just contaminated by noise. Due to multi-path reflections and the path to satellites being obscured by close proximity to tall buildings, positions can be consistently inaccurate but still contain a shape that matches a junction layout, even if that shape is displaced. Traces often drift, reaching 20 m or more from the true location before suddenly jumping back when a junction is reached and alternative satellites are visible for the position fix.

The positional accuracy of the GPS data is impossible to quantify unless the true position is known using another method. One of the traces contains 20 min of data from a known static location at a junction where the GPS position can be seen to drift around the centre of the road. By playing back the samples in real-time, obvious discrepancies in the GPS position become apparent at points where pedestrians wait to cross the road.

8.2 Power management

An important challenge when designing sensing application on the mobile phone platforms is power consumption, in particular, when various communication interfaces such as Bluetooth, GPS, WIFI and GPRS are used by the application.

We have used Nokia Energy Profiler for power management of the phone applications which is a standard software tool provided by Nokia specifically that allows developers to test and monitor their application's energy usage in real-time in the target device.

We found that Bluetooth communications, screen display, processing sensor data and GPRS radios are responsible for draining most of the battery power. The battery consumption relating to the use of the phone's internal microphone was trivial.

Figure 14, shows the energy consumption measured when the phone software is running in four different modes with the profiler for a period of sampling intervals ranging from 10 s with power usage percentage on the vertical axis.

On standby, the Nokia 95 phone runs for 200 h. However, this phone has a lot of other technologies attached which are a further drain on its power. The period of audio sampling and processing are inconsequential and use little power. Bluetooth proximity detection requires a 120 s scan period to capture neighboring MAC addresses due to the cache flushing limitations of the Bluetooth API in JME. GPS location detection is inherently power hungry

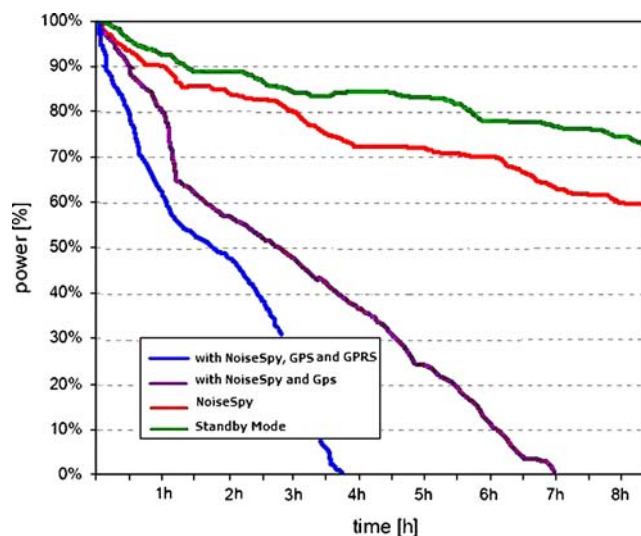


Fig. 14 The battery consumption over time, with four different modes settings

and takes time to acquire a lock when turned on. The highest spikes shown on the plot are due to the upload of data which uses the cellular radio. While this is a typical pattern of energy consumption there are other factors which can cause variations, including: distance to cell tower, environmental radio characteristics, the amount of data to upload, the number of Bluetooth neighbors, denial of resources due to the phone being in use for other purposes, network disconnections, sensor sample intervals, sample durations, upload interval, GPS lock time, and temperature.

It is important to mention here that taking images with the camera phone can alter the power consumption level sporadically depending how often it is used.

As application developers, we want to build applications that offer good fidelity and user experience without significantly altering the operational lifetime of the standard mobile phone; these are our future strategies to reduce the phone power consumption:

- Data upload from the phone via GPRS can draw a large amount of power, particularly when the phone is far from a cell base station. A challenge is therefore to reduce the use of these radios without significantly impacting the application experience. We will offer the users the option to change the data upload rate to GPRS, for example, users can choose to upload the data to the server frequently (every second) or less frequently (every minute). Also they could push the data to the server once a day.
- Similarly, we have added a timer to allow users to change the sampling rate from the sensors according to their task which might not require a frequent use of the radios for communications or acquisition of satellite signals for location coordinates.

- Processing sensor data on mobile phones is computationally intensive and could consume a considerable amount of energy, potentially limiting the usability of the system. With this in mind we will push the complex data processing tasks to the backend servers.
- Allowing users to switch off screen display and graphs.

8.3 Scalability

NoiseSpy framework has the potential to open up a large scale real-time environmental monitoring network around a city. However, this development presents new and formidable concerns arising from the need to transmit, integrate, model and interpret vast quantities of highly diverse spatially and temporally varying sensor data. For example maintaining sensor reliability and software consistency across more than 70 MobSens nodes raises challenges in terms of dealing with failed mobile nodes and network links; potentially frequent updates from many users; and the impact of software updates on node reliability.

We are looking at ways to enable updating the phone application on users' handsets automatically. This can be done by adopting client-server mechanism similar to MUPE [22] in which users do a single install only and the client can update itself once new versions are available.

8.4 Handling disconnections

NoiseSpy offers real-time networking paradigm which is frequently in contact with a cellular access point, whereby data is uploaded continuously to the backend server over GPRS. While connectivity is generally good in an urban environment, the GPRS connection is still not connected 100% of the time. This intermittent connectivity has led to delay-tolerant sensing; where sensed data is cached for a time before it is uploaded when the connection is restored.

GPS signals also subject to disconnection. While there have been some advances in improving the sensitivity of GPS receivers and developing techniques such as assisted GPS that permit a GPS receiver to use attenuated signals, the antenna of a conventional receiver must have a direct line of sight to the GPS satellites. Our users have reported some problems with GPS coverage in Cambridge city Centre due to the high urban canyons.

We are looking at ways to enhance our location tracking techniques, for example we are working with HWCommunications² to use their GSM-based Cellular positioning along side GPSs.

² <http://www.HWComm.co.uk/>

8.5 Privacy

A top concern of our users was the security and confidentiality of the data collected about them, once the data is uploaded it can be shared with others. For example, through the location system an adversary could obtain the current position of every individual. Continuous access to this information would allow him/her to track movements of an unknown user. An adversary might also attempt to tamper with the data received by NoiseSpy application, or to insert false data. We consider this to constitute a location privacy threat, even if the user is not able to link identities to the reported user locations. We seek to provide NoiseSpy system with some confidence in the integrity of the collected data.

Ongoing work in the NoiseSpy has begun to address these by providing privacy policies, which inform the user about our data handling practices and serve as the basis for the user's decision to release data. Also we are hoping to develop an efficient verification protocols that ensure secure data management and guarantee system integrity.

8.6 Memory and data logging

The system needs enough storage for data logging taking in account that devices like mobile phones must accommodate other storage needs as well (for example, leaving room for saving videos and photos).

The amount of data collected over an extended period of time is dependent upon the maximum data storage capacity of the mobile and the battery lifetime. The Nokia N95 mobile phone used in the system has up to 160 MB internal dynamic memory and memory card slot supporting up to 2 GB microSD memory cards.

To test the size of memory available for our applications we have collected sensor data using the N95 every second for five days. We found that only consumed 30.25 Mb of the internal phone memory.

Currently, our users have the following logging options which allow them to control how much data they want to store on the phone:

- Interval: logging multiple times per day.
- Intensive logging very frequently (e.g., every second).
- Regular intervals: loggings scheduled at regular intervals.
- Real-time logging over GPRS only.
- Real-time logging over GPRS and on-the-phone logging.
- On-the-phone-logging only.
- Upload log files to the server over once a day.
- Logging only when a person's location changes, by checking his/her speed which is measured by the GPS.

8.7 Challenges related to noise mapping

Maps are graphically appealing and intuitively easy to grasp. However, they're actually quite difficult to interpret rigorously, and they provide little insight into local-area dynamics without additional processing. Since some locations were heavily sampled, whereas some areas left completely un-sampled, and still others will be sparsely sampled, we need a mechanism for interpolating data at any given location.

We found also that defining the relationship between individual colours and Noise levels was problematic as the colour scheme for noise readings was chosen that would not show small changes in Noise. We are looking at various data interpolation and weighting techniques to show a realistic representation of the noise data on the map. For example, we are planning to employ HSV as an alternative colour space to RGB. Using HSV, every cell in the map will be assigned a hue which will correspond to the impact of noise intensity.

It is evident that the quality of the final results depends on the quality of the data and the methods used in every step of the noise mapping process. In order to improve the efficiency, the level of detail of the complete noise mapping process should be adapted to the purpose and desired level of detail of the entire noise effect study. This also means, finding a balance in the level of detail and the exactness between the different steps in the noise mapping process. It makes no sense to put a lot of effort in the accuracy of one step while another step in the process is done with less detail, losing the earlier obtained details. In conclusion, the accuracy of the methods and input data used should be in balance and adapted to the desired level of detail of the final results.

9 Related work

NoiseSpy builds upon a large body of related projects which utilize mobile phones as sensing devices. MetroSense project outlines an exciting opportunistic "people-centric" approach to mobile phone sensing including several deployments with bicycles [1].

Center for Embedded Network Sensing has setup a research initiative called "Participatory Sensing" that is developing infrastructure and tools to enable individuals and groups to initiate their own public "campaigns" for others to participate in by using networked mobile devices [14]. The MyExperience [12] tool is a mobile software application for *in-situ* data collection to support the study of human behavior and the evaluation of mobile computing technologies. MyExperience has been designed to record a wide range of data including sensors, images, video, audio and user surveys.

SensorPlanet is a Nokia-initiated global research framework for mobile-device centric wireless sensor networks. It helps researchers to perform innovative research on wireless sensor networks, where the mobile devices can be seen both as gateways to the mesh sensor networks and also as sensor nodes themselves [18–20].

Furthermore, Nokia has introduced the Echo Phone concept Nokia Eco Sensor concept which includes a wearable sensor unit that will house a number of optional sensors to monitor environment, health and local weather conditions, a dedicated mobile phone [5].

Research in urban noise pollution is relatively recent and still in an incipient stage. Some of the work in this area go back to Nokia 3250 which has a built-in sound meter [16]. Sound Monitor application is also built by SysOpen Digia to use the phone microphone to monitor sound [4]. A sound threshold value can be set to trigger on alarm. The alarm type can be a pre-recorded sound clip a plain text SMS, a phone call or an MMS message.

Noise mapping such as the one generated by Defra have been made using computer modelling techniques, based on information such as traffic flow data, road/rail type, and vehicle type data. No actual noise measurements have been made in the production of these strategic maps [3].

Oliveira et al. [17] discuss a GIS-based noise planning tool created by public contractors for the city of Belo Horizonte in Brazil. And [15] have reported their experiences in using the Tmote invent prototyping platform for collecting noise pollution data in both indoor and outdoor settings and pointed out the potentials and limits of our prototype of using wireless sensor nodes for environmental sensing.

10 Future work and conclusion

Overall, the user experience is positive. Because many of the users enjoyed using NoiseSPY, they kept the NoiseSPY after the initial trial phase for further testing. We are currently working on revising some of the components and improving a few architectural elements in order to reflect some of the valuable feedback from the participants. Specifically, future revisions of the NoiseSPY system will include:

- An improved NoiseSPY software module on the phone that prolongs the battery life. Our goal is to enable users to run the software as long as possible without recharging the device.
- Investigating in-network calibration methods to ensure long-term correct network operation between the phones.
- An enhanced version of the web portal, ideally we are looking for an improved version that would first reduce the wait time, but second reduce the amount of data shown to the user, as it is mostly similar data

consolidating and filtering the data before returning it to the user is essential.

- An improved finer grained privacy policy settings as well as an enhanced user interface to provide the user with more powerful ways to browse their traces as well as others.
- We are currently researching on the data fusion and aggregation technique to improve the system performance when large amount of data are collected and transferred, especially when the third part data (including the air pollution and weather data) are imported [7].

In this paper, we have presented NoiseSpy a transformation of the mobile phone from solely communication tool to that of a sound sensor rich personal measurement instrument and addressed the possibilities and practicality of using mobile sound monitor to scrutinize the ambient noise in the environment. It shows how modern persuasive technologies such as mobile devices can promote awareness and reflection to motivate discussions and lifestyle changes within individuals and groups.

Such systems can elevate individuals to have a powerful new voice in society, to act as citizen scientists, and collectively learn and lobby for change within their neighborhood, city, and nation. NoiseSpy system can be seen as a new urban planning tool that enables the determination of exposure to environmental noise, through noise mapping; This requires an adoption of action plans, based upon noise mapping results, which should be designed to manage noise issues and effects, including noise reduction if necessary.

References

1. Campbell A, Eisenman SB, Lane N, Miluzzo E (2006) People-Centric urban sensing. Proceedings of the 2nd ACM/IEEE Annual International Wireless Internet Conference (WICON), Boston, MA, 2–5 August 2006
2. Cycling Courier as Mobile Sensors. A video appeared on Wie Wissen TV programme on DasErste.de German TV channel 25/May/2008, http://www.daserste.de/wwiewissen/beitrag_dyn-uid_a39gj50phkd5w8pf-cm.asp
3. Defra Noise Mapping England, <http://noisemapping.defra.gov.uk/wps/portal/noise>
4. Digia SoundMonitor, <http://mysymbian.com/s60/software/applications.php?faq=4&fldAuto=1203>.
5. Eco Nokia Sensor Concept, <http://www.nokia.com/A4707477>, retrieved 2008-09-17
6. Yong E (2008) City songbirds are changing their tune. In *NewScientist*, issue 2649
7. Kanjo E, Bacon J, Roberts D, Landshoff P (2009) MobSens: making smart phones smarter. *IEEE Pervasive Computing*, vol. 8, no. 4, pp 50–57, Oct–Dec 2009, doi:10.1109/MPRV.2009.79
8. Environment Agency, www.environment-agency.gov.uk/
9. Environmental Protection, <http://www.environmental-protection.org.uk/noise/environmental-noise/noise-pollution/>
10. Paulos R, Honicky R, Goodman E (2007) Sensing atmosphere, workshop on sensing on everyday mobile phones. In *ACM Conference on Embedded Networked Sensor Systems (SenSys 2007)*
11. European Commission Working Group Assessment of Exposure to Noise (WG-AEN). Good Practice Guide for Strategic Noise Mapping and the Production of Associated Data on Noise Exposure, January 2006
12. Froehlich J, Chen M, Consolvo S, Harrison B, Landay J (2007) MyExperience: a system for in situ tracing and capturing of user feedback on mobile phones. Proceedings of *MobiSys 2007*, San Juan, Puerto Rico, June 11–14, 2007
13. Fletcher H, Munson WA (1933) Loudness, its definition, measurement and calculation. *Journal of the Acoustic Society of America* 4(2):82–108
14. Burke J, Estrin D, Hansen M, Parker A, Ramanathan N, Reddy S, Srivastava MB (2006) Participatory sensing. In *ACM Sensys, World Sensor Web Workshop*, Boulder, Colorado, USA, October 2006
15. Filippini L, Santini S, Vitaletti A (2008) Data collection in wireless sensor networks for noise pollution monitoring. *DCOSS*, pp 492–497
16. Nokia 3250, Noise meter, <http://www.nokia.co.uk/A4221048>
17. Oliveira MPG, Medeiros EB, Davis Jr CA (1999) Planning the acoustic urban environment: a GIS-centered approach. *ACM GIS '99*, Kansas City, MO, pp 128–133, November 1999
18. SensorPlanet, <http://www.sensorplanet.com>
19. Abdelzaher T, Anokwa Y, Boda P, Burke J, Estrin D, Guibas L, Kansal A, Madden S, Reich J (2007) Mobiscopes for human spaces. *IEEE Pervasive Computing*, vol. 6, no. 2, pp 20–29, Apr–Jun, 2007
20. Tuulos VH, Scheible J, Nyholm H (2007) Combining Web, mobile phones and public displays in large-scale: Manhattan story mashup. *Proc. 5th Int'l Conf. Pervasive Computing, LNCS 4480*. Springer, pp 37–54
21. www.escience.cam.ac.uk/mobile_sensing
22. Suomela R, Räsänen E, Koivisto A, Mattila J (2004) Open-source game development with the multi-user publishing environment (MUPE) application platform. In: Rauterberg M (ed) *Proceedings of the Third International Conference on Entertainment Computing 2004*. Lecture Notes in Computer Science 3166 Springer 2004, c Springer, pp 308–320
23. New Scientist article, Cyclists' cellphones help monitor air pollution. <http://technology.newscientist.com/article/dn13130-cyclists-cellphones-help-monitor-air-pollution.html>
24. Miluzzo E, Lane ND, Fodor K, Peterson RA, Lu H, Musolesi M, Eisenman SB, Zheng X, Campbell AT (2008) Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application. In *Proc. of 6th ACM Conference on Embedded Networked Sensor Systems (SenSys '08)*. Raleigh, NC, USA, Nov. 5–7, 2008
25. Noise Pollution, *The Columbia Electronic Encyclopedia*, 6th ed. Copyright © 2007, <http://www.infoplease.com/ce6/sci/A0835810.html>