Ear-Phone: An End-to-End Participatory Urban Noise Mapping System

Rajib Kumar Rana[†‡] Chun Tung Chou[†] Salil S. Kanhere [†] Nirupama Bulusu [II] Wen Hu[‡]
[†] School of Computer Science and Engineering, University of New South Wales, Sydney, Australia
[II]Department of Computer Science, Portland State University, USA
[‡] CSIRO ICT Centre Australia
{rajibr,ctchou,salilk}@cse.unsw.edu.au, nbulusu@cs.pdx.edu, wen.hu@csiro.au

ABSTRACT

A noise map facilitates monitoring of environmental noise pollution in urban areas. It can raise citizen awareness of noise pollution levels, and aid in the development of mitigation strategies to cope with the adverse effects. However, state-of-the-art techniques for rendering noise maps in urban areas are expensive and rarely updated (months or even years), as they rely on population and traffic models rather than on real data. Participatory urban sensing can be leveraged to create an open and inexpensive platform for rendering up-to-date noise maps.

In this paper, we present the design, implementation and performance evaluation of an *end-to-end* participatory urban noise mapping system called Ear-Phone. Ear-Phone, for the first time, leverages *Compressive Sensing* to address the fundamental problem of recovering the noise map from incomplete and random samples obtained by crowdsourcing data collection. Ear-Phone, implemented on Nokia N95 and HP iPAQ mobile devices, also addresses the challenge of collecting accurate noise pollution readings at a mobile device. Extensive simulations and outdoor experiments demonstrate that Ear-Phone is a feasible platform to assess noise pollution, incurring reasonable system resource consumption at mobile devices and providing high reconstruction accuracy of the noise map.

Categories and Subject Descriptors

C.m [**Computer Systems Organization**]: Miscellaneous–*Mobile Sensing Systems*

General Terms

Design, Experimentation, Performance

Keywords

Noise Pollution, Mobile Phones, Compressive Sensing, Participatory Sensing

1. INTRODUCTION

At present, a large number of people around the world are exposed to high levels of noise pollution, which can cause serious illnesses ranging from hearing impairment to negatively influencing productivity and social behavior [12]. As an abatement strategy, a number of countries, such as the United Kingdom [9] and Germany [10], have started monitoring noise pollution. They typically use a noise map (a visual representation of the noise level of an area) to assess noise pollution levels. The noise map is computed using simulations based on inputs such as traffic flow data, road or rail type, and vehicle type. Since the collection of such input data is very expensive, these maps can be updated only after a long period of time (e.g. 5 years for UK [9]). To alleviate this problem, a recent study [20] proposes the deployment of wireless sensor networks to monitor noise pollution. Wireless sensor networks can certainly eliminate the requirements of sending acoustic engineers for taking real measurements, but the deployment cost of a dedicated sensor network in a large urban space will also be prohibitively expensive.

In this paper, we instead propose an urban sensing approach (also known in the literature as participatory sensing [6], people-centric sensing [11] or community sensing [15]) for monitoring environmental noise, especially roadside ambient noise. The key idea in participatory sensing is to "crowdsource" the collection of environmental data in urban spaces to people, who carry smart phones equipped with sensors and location-providing Global Positioning System (GPS) receivers. The vision of participatory sensing is inspired by the success of other online participatory systems, such as Wikipedia, online reputation systems, and human computation systems such as the Google Image Labeler. Due to the ubiquity of mobile phones, the proposed approach can offer a large spatial-temporal sensing coverage at a small cost. Therefore, a noise map based on participatory data collection can be updated with a very small latency such as hours or days compared to months or years, making information provided by such a noise map significantly more current than that provided by traditional approaches.

It is non-trivial to build a noise pollution monitoring system based on mobile phones. Mobile phones are intended for communication, rather than for acoustic signal processing.[1] To be credible, noise pollution data collected on mobile phones should be comparable in accuracy to commercial sound level meters used to measure noise pollution. Since a participatory noise monitoring system relies on volunteers contributing noise pollution measurements, these measurements can only come from the place and time where the volunteers are present. Furthermore, volunteers may prioritize the use of the microphone on their mobile phones for conversation. Or they may choose to collect data only when the phone has sufficient energy. Consequently, samples collected from mobile phones are typically randomly distributed in space and time, and are incomplete. To develop a useful noise pollution monitoring application, we need to recover the noise map from *random* and *incomplete* samples obtained via crowdsourcing. In this paper, we address these challenges. Our main contributions are:

1. We present the design and implementation of an *end-to-end* noise mapping system, called Ear-Phone, to generate the noise map of an area using participatory urban sensing. EarPhone consists of mobile phones and a central server. It encompasses signal processing software to measure noise pollution at the mobile phone, as well as signal reconstruction software at the central server. This new noise mapping system is expected to cost significantly less than traditional noise monitoring systems.

2. We address the problem of incomplete samples that are obtained via crowdsourcing by using *compressive sensing*, focusing on roadside noise pollution.[2] To the best of our knowledge, this is the first application of compressive sensing to environmental noise data collection.

3. We evaluate Ear-Phone with extensive simulations and real-world outdoor experiments. The results show that Ear-Phone has reasonable accuracy, and resource requirements in terms of CPU load and energy consumption.

The rest of the paper is organized as follows. In the next section, we describe the Ear-Phone architecture followed by the system design in Section 3. Then, we evaluate Ear-Phone with both outdoor experiments (Section 4) and extensive simulations (Section 5). We present related work in Section 6 and conclude in Section 7.

[1]For example, devices such as the Nokia N95 or HP iPAQ do not support floating-point arithmetic, which must be emulated with fixed point operations.

[2]We focus on roads because typically noise pollution is most severe on busy roads.

2. EAR-PHONE ARCHITECTURE

In this section, we provide an overview of Ear-Phone. A detailed description of the system components is presented in Section 3.

The overall Ear-Phone architecture, depicted in Fig. 1 consists of a mobile phone component and a central server component. Noise levels are assessed on the mobile phones before being transmitted to the central server. The central server reconstructs the noise map based on the partial noise measurements. Note that reconstruction is required because the urban sensing framework cannot guarantee that noise measurements are available at all times and locations.

Let us begin with a mobile phone user who is walking along a street. We call a mobile phone with the Ear-Phone application a MobSLM, where SLM stands for "sound level meter" which is the instrument used by acoustic engineers to measure environmental noise level. When the mobile phone is not used for conversation the MobSLM on the phone is turned on. [3] When turned on, the signal processing module starts computing a loudness characteristic known as the equivalent noise level ($\mathrm{LA}_{\mathrm{eq,T}}$) over a time interval T from the raw acoustic samples collected by the microphone over the corresponding time interval. The computed noise level is further tagged with the GPS coordinates (which will be denoted by (lat,lon) and system time before being stored in the phone memory. The stored records $\langle$ time,lat,lon,$\mathrm{LA}_{\mathrm{eq,T}}$ $\rangle$ are uploaded to the central server when the mobile phone detects an open WiFi access point. Of course, 3G services on mobile phones can also be used to upload data.

The communication manager at the central server waits for user transmissions. When it receives user data, it converts the GPS coordinates of a record to a Military Grid Reference System (MGRS, see Section 3.2.2 for the detailed description) grid index and stores the information $\langle$ time, grid index, $\mathrm{LA}_{\mathrm{eq,T}}$ $\rangle$ in a data repository. Reconstruction is conducted at (predefined) periodic intervals[4]; when triggered, the reconstruction module is invoked to reconstruct the missing data. The reconstructed data is then stored in the data repository.

A query from an end user (e.g., what is the noise level on Oxford Street at 5pm on 28 October 2009?) is processed by a query manager at the central server. The location information (e.g., Oxford Street) of the query is first resolved into grid indices and the reconstructed data associated with those grid indices is fetched from the data repository. Then, the grid indices are converted back to GPS coordinates and the corresponding noise levels are overlaid on a geo-centric Internet map before being displayed to the end user.

[3]Note that in the current prototype deployment we have not implemented this feature. During our experiments we did not use the phone for conversation.

[4]Note that in this paper we primarily focus on the accuracy of the noise map obtained from participatory sensing. Determination of a suitable update interval is left for future work.

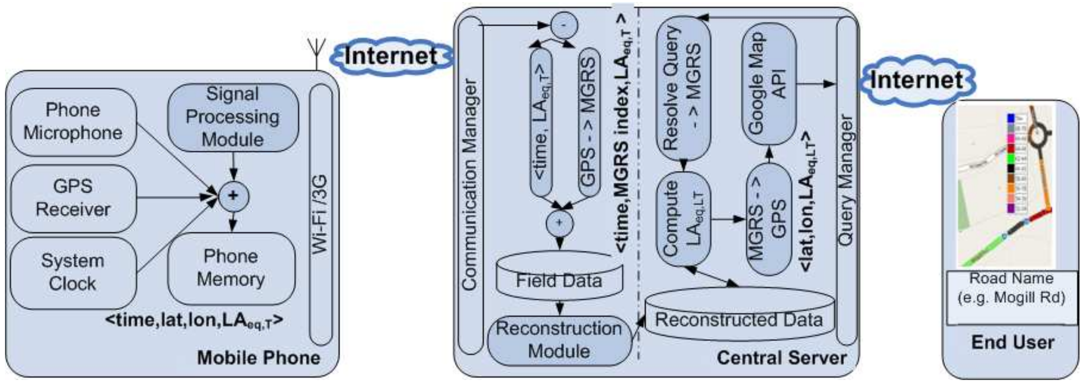


Figure 1: Ear-Phone Architecture

3. SYSTEM COMPONENTS

In this section, we describe the major components of Ear-Phone in detail.

3.1 Mobile Phone Components

3.1.1 *Signal Processing Module*

The aim of the signal processing module is to quantitatively assess the environmental noise. Noise level or loudness is typically measured as the A-weighted equivalent continuous sound level or $\mathrm{LA_{eq,T}}$. A-weighting is the commonly used frequency weighting that reflects the loudness perceived by human being [14]. Measured in decibel (dBA), $\mathrm{LA_{eq,T}}$ captures the A-weighted sound pressure level of a constant noise source over the time interval T, which has the same acoustic energy as the actual varying sound pressure level over the same interval. Note that sound pressure level is captured by a microphone as an induced voltage. The A-weighted equivalent sound level $\mathrm{LA_{eq,T}}$ in time interval T is thus given by

$$\mathrm{LA_{eq,T}} = 10\log_{10}(\underbrace{\frac{1}{T}\int_0^T (v_A(t))^2 dt}_{\bar{v}_A(T)}) + \Delta \quad (1)$$

where $v_A(t)$ is the result of passing the induced voltage $v(t)$ through an A-weighting filter and Δ is a constant offset determined by calibrating the microphone against a standard sound level meter.

In order to compute $\bar{v}_A(T)$, we design a tenth-order digital filter (whose coefficients are given in Table 1) whose frequency response matches with that of A weighting over the range 0–8kHz. This range is chosen because the acoustic standard, IEC651 Type 2 SLM [14], requires measurement of environmental noises between 0 and 8 kHz. Based on the coefficients of the digital filter (a_l, b_l where $l = 1..10$), we then calculate $\bar{v}_A(T)$ using the following algorithm.

Algorithm *Compute* $\bar{v}_A(T)$

1. Initialize: $Q = F_sT - 1$, F_s =Sampling Frequency, Sampling Period $T_s = \frac{1}{F_s}$;

Input: Voltage samples $v(kT_s)$ for $k = 0, 1, 2, \ldots, Q-1$ over duration $[0, T]$;
Output: $\bar{v}_A(T)$

2. Based on $\{a_l, b_l\}$ and initial condition, $v_A(kT_s) = 0$ for $k = 0, ..., 9$, recursively compute

$$\begin{aligned} v_A(kT_s) &= \sum_{\ell=1}^{10} a_\ell v_A((k-\ell)T_s) \\ &+ \sum_{\ell=0}^{10} b_\ell v((k-\ell)T_s) \text{ for } k \geq 10 \quad (2) \end{aligned}$$

3. Compute

$$\bar{v}_A(T) = \frac{1}{Q}\sum_{k=0}^{Q-1} v_A(kT_s)^2 \quad (3)$$

3.2 Central Server Components

3.2.1 *Computing Long-term Equivalent Noise Level, $LA_{\mathrm{eq,LT}}$*

In order to compute the long-term equivalent noise level $\mathrm{LA_{eq,LT}}$ over the duration NT (where $N > 1$ and N is an integer) from the equivalent noise levels $\mathrm{LA_{eq,T}}$ measured over shorter time durations T, we use the following standard formula:

$$\mathrm{LA_{eq,LT}} = 10\log_{10}[\frac{1}{N}\Sigma_{i=1}^{N} 10^{0.1\mathrm{LA_{eq,T_i}}}] \quad (4)$$

where N is the number of reference time intervals and $\mathrm{LA_{eq,T_i}}$ is the time average A-weighted sound pressure level in the i-th reference time interval. The above formula can be readily derived by noting that the equivalent noise level is defined as the logarithm of average noise power (see equation (1)).

3.2.2 *GPS, MGRS conversions*

The reasons for approximating GPS by square areas are two fold. First, computing the $\mathrm{LA_{eq,T}}$ for every possible GPS coordinate is impractical because there are

ℓ	0	1	2	3	4	5	6	7	8	9	10
b_ℓ	0.9299	-2.1889	0.7541	1.3229	-0.7728	0.1025	-0.2398	-0.0098	0.1154	-0.0103	-0.0033
a_ℓ		2.1856	-0.7403	-1.0831	0.6863	-0.2274	0.2507	-0.0058	-0.0821	0.0153	0.0004

Table 1: Coefficient of the digital filter that approximates A-weighting

infinite GPS coordinates. Secondly, the acoustic standards for monitoring noise pollution recommend measuring the pollution in square areas (Section 5.3.1(a) in [1]) assuming that the noise level is constant over that area. In order to approximate GPS by grids, we use MGRS, which can divide the earth surface into squares of 100 m $\times$ 100 m, 10 m $\times$ 10 m or 1 m $\times$ 1 m etc.

We followed the Australian acoustic standard to determine an appropriate grid size. We assume that the volunteers walk along the pavement (or sidewalk), which is typically two meters wide, and measure ambient noise on the street level which is the aggregate of the noise generated by multiple moving vehicles. The Australian acoustic standard restricts the noise level difference between two adjacent grids to be no more than 5 dB (Section 5.3.2 in [1]). Therefore, we conducted a number of experiments where we put a MobSLM at a static position and put another MobSLM at difference distances from the first MobSLM and recorded the difference of $LA_{eq,1s}$ readings for each distance. For grid sizes of 10 $\times$ 10, 20 $\times$ 20, 30 $\times$ 30, 40 $\times$ 40 and 50 $\times$ 50 square meters, the corresponding noise level differences between adjacent grids were found to be 2.26 $\pm$ .06, 3.82 $\pm$ .05, 3.86 $\pm$ .03, 4.11 $\pm$ .02 and 4.97 $\pm$ .03 dB, respectively. We could therefore use square grids which are less than or equal to 50 meters in each dimension. We chose to use a grid size of 30m $\times$30m because it takes approximately 30 seconds for a Nokia N95 to acquire a GPS position. In that time, a person can travel 30 meters at normal walking speed (1 m/s). Furthermore, GPS has an accuracy of 10 meters in outdoor environments, therefore a 30 $\times$ 30 grid could help us to cope with the GPS accuracy. We use formulations in [17] to convert between GPS and MGRS.

3.2.3 Signal Reconstruction Module

To study the sampling requirements, communication overhead and reconstruction accuracy trade-offs within Ear-Phone,we developed two sensing strategies. In this section, we will describe the two sensing strategies, namely the *projection method* and the *raw-data method*, and also describe how the central server performs reconstruction using the information collected by these two different sensing strategies. For ease of explanation, we will explain the two sensing strategies with an example.

Consider the trajectory of two volunteers, A and B, along a section SG of a one dimensional street (see Fig. 2). Section SG contains three MGRS grid references: ℓ_1, ℓ_2 and ℓ_3. Suppose at times t_1 and t_2, volunteer A collects noise samples in grids ℓ_1 and ℓ_2, and B collects samples in grids ℓ_3 and ℓ_1 respectively. Note that the noise sample in a grid refers to the equivalent noise level $LA_{eq,1s}$ in that grid. The complete noise samples in section SG, during time t_1 and t_2 can be represented as a vector $x = [d(\ell_1, t_1), d(\ell_2, t_1), d(\ell_3, t_1), d(\ell_1, t_2), d(\ell_2, t_2), d(\ell_3, t_2)]^{\mathrm{T}}$, where $d(\ell, t)$ is the noise level at locations $\ell = \{\ell_1, \ell_2, \ell_3\}$ and time $t = \{t_1, t_2\}$. We refer to the vector x as a *noise profile*. Similarly, samples collected by A and B can be represented as vectors $x_A = [d(\ell_1, t_1), 0, 0, 0, d(\ell_2, t_2), 0]^{\mathrm{T}}$ and $x_B = [0, 0, d(\ell_3, t_1), d(\ell_1, t_2), 0, 0]^{\mathrm{T}}$ respectively.

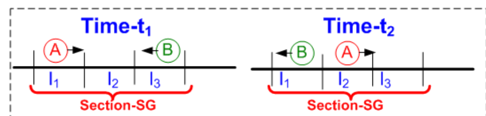


Figure 2: Illustration of urban sensing

In the projection method, A multiplies his measurement vector x_A with a projection vector $\phi_A = [\phi_A^1, 0, 0, 0, \phi_A^5, 0]^{\mathrm{T}}$, where ϕ_A^1, ϕ_A^5 are Gaussian distributed random numbers with mean zero and unit variance, and sends the projected value, $y_A = \phi_A^T * x_A$ to the central server. Note that the inner product $\phi_A^T x_A$ is known as a projection in compressive sensing.

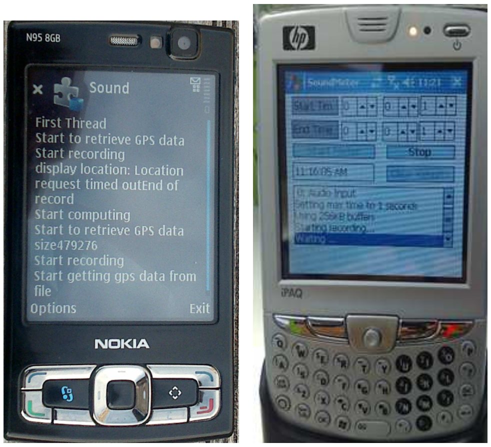
In the raw-data method, A directly sends his noise samples to the central server. Then, at the central server the projection vector for A's data is regenerated as $\phi_A = [\phi_A^1, 0, 0, 0, 0, 0; 0, 0, 0, 0, \phi_A^5, 0]^{\mathrm{T}}$, where $\phi_A^1 = \phi_A^5 = 1$. Note that the projected value is again given by $y_A = \phi_A^T x_A$. In fact, in this case, y_A is a vector consisting of A's measurements $d(\ell_1, t_1)$ and $d(\ell_2, t_2)$.

At the central server the reconstruction module accumulates the projected values from all volunteers in a vector $y = [y_A, y_B]^{\mathrm{T}}$ and forms the projection matrix, $\Phi = [\phi_A^{\mathrm{T}}, \phi_B^{\mathrm{T}}]$. The reconstruction proceeds in two steps. In the first step, the central server solves the following optimization problem:

$$\hat{g} \quad = \quad \arg \min_{g \in \mathbb{R}^N} \|g\|_1 \text{ such that } y = \Phi \Psi g \qquad (5)$$

where Ψ is a transform basis in which the noise profile x is compressible. In [19], we provide some evidence that the noise profile x is compressible in the Discrete Cosine Transform (DCT) basis. In the second step, an estimate of the noise profile x is given by $\Psi \hat{g}$. Note that the optimization problem (5) is a convex optimization and there exist efficient numerical routines for this class of problems.

In our current implementation we used a simplified "query to grid resolver", which is essentially a look up table, to store the grid indices of the road segments. In our prototype implementation, we only stored the grid indices of the road segments where our experiments were



(a) (b)

Figure 3: Screenshots of (a) Ear-Phone application running on Nokia N95 (b) Signal processing module running on HP iPAQ 6965.

conducted. We used widely available open-source software for query manager and communication manager, therefore we do not describe these components in further detail.

4. IMPLEMENTATION AND EVALUATION

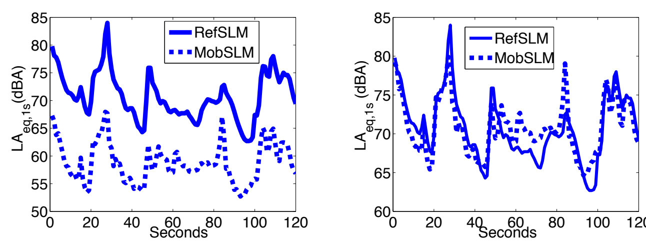
In this section, we first describe the Ear-Phone implementation. Then, we evaluate the system performance in terms of noise-level measurement accuracy, resource (CPU, RAM and energy) consumption and noise-map generation, which demonstrates that Ear-Phone is an effective end-to-end system for measuring noise pollution from incomplete and random samples inherent in participatory sensing.

4.1 System Implementation

We have implemented the mobile phone components on two hardware platforms - the Nokia N95 and the HP iPAQ (Fig. 3). We choose Java as the programming language because it is platform independent. The various mobile components are implemented as separate application threads (e.g., GPS thread and signal processing thread) in Java. We used the raw-data method (see Section 3.2.3) as the sensing strategy for the current Ear-Phone prototype. The server component consists of a MySQL database and PHP server-side scripting. We used the MySQL database to store both the collected noise level data and the reconstructed noise level data. We used a PHP script to implement the server-side modules such as the communication manager, GPS MGRS converter, noise signal reconstruction module, and query manager (see Section 2 for the description of these modules).

4.2 Measurement Accuracy

Recall from Section 3 (Eq.(4)) that we need to know the calibration offset to measure $LA_{eq,T}$. We determine this offset by conducting a simple calibration experiment. We use the freely available Audacity tool [4] to produce a chain of one second wide pulses of varying amplitudes and compare the responses of our algorithm (when computing $LA_{eq,1s}$) on a Nokia N95 and a HP iPAQ with the responses of a commercial sound level meter, Center-322 SLM [7] (see Fig. 4(a)). We use the mean of the difference in readings between the commercial meter (we refer it by RefSLM) and our mobile based SLM, as the offset. After adding the computed offset, we repeat the experiment and plot the responses in Fig. 4(b). We observe that our mobile phone based SLMs have a precision of ±2.7 dB. Note that a difference of 3 dBA is *imperceptible* to the human ear. Note also that we found that phones from the same model could have different calibration offsets. This essentially means that a calibration technique needs to be developed to automatically calibrate the mobile phones of volunteers.



(a) Time series (Before calibration)

(b) Time series (After calibration)

Figure 4: Measurement Accuracy of Ear-Phone

In our current prototype deployment, we assume that phones are carried in the volunteer's palm or in a manner such that the microphone is not obstructed. However, we have also conducted experiments to investigate how the positions of the phone affect the measurement accuracy. In these experiments we kept a

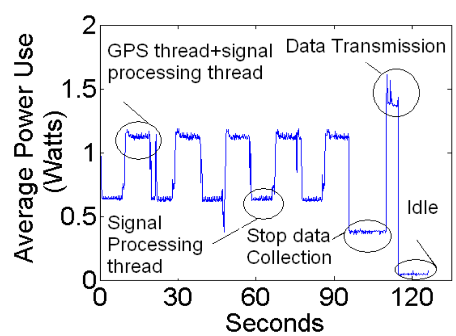
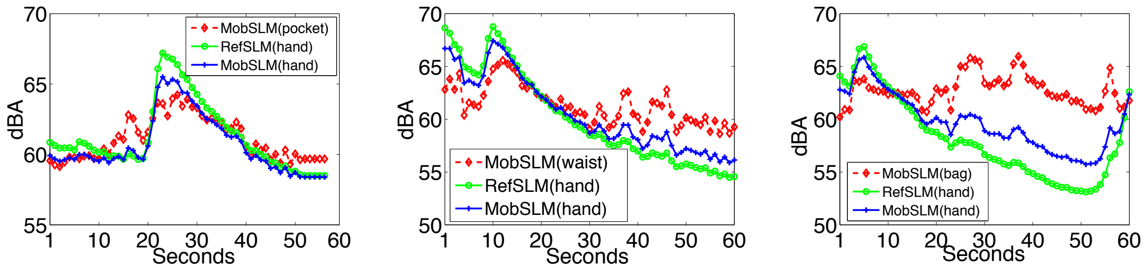


Figure 5: Power consumption of Ear-Phone on Nokia N95 for a two-minute period.



(a) (b) (c)

Figure 6: This figure shows the noise level (in dBA) when the MobSLM is carried (a) inside a shirt pocket (b) around the waist (c) inside a backpack. A RefSLM and a MobSLM are used as references.

MobSLM in three different positions: inside a volunteer's shirt pocket, on his waist belt, and in his backpack, and recorded roadside noise levels. In order to compare the noise measurement, the volunteer also carried a MobSLM in one hand and the RefSLM in the other hand. Fig. 6 summarizes the experimental results. If the MobSLM is being held in our palm, then its accuracy is within 2.7dB of the RefSLM. If the phone is in a shirt pocket or carried around the waist, the accuracy is within 3.4 dB of the RefSLM. The additional error of 0.7dB is small compared with the actual noise level of about 67dB. If the phone is carried in a backpack, then the accuracy is within 4.1dB of the RefSLM. An interesting observation of the results in Fig. 6 is that the effect of placing the MobSLM in the shirt pocket, around the waist and in a bag is not a lower noise level due to increased attenuation, rather the variance of the noise level becomes larger. A physical explanation of this observation is yet to be investigated but these results show that there is an opportunity to embed context-awareness to assist the MobSLM.

4.3 Resource Usage

4.3.1 *Power Benchmarks*

We measure the power consumption of Ear-Phone using the Nokia Energy Profiler, a standard software tool provided by Nokia specifically for measuring energy usage of applications running on Nokia hardware. The profiler measures battery voltage, current, and temperature approximately every fourth of a second and stores the results in the RAM. Fig. 5 shows the typical contribution of Ear-Phone to the overall energy budget during a two minute period with power consumption in Watts on the y axis. The highest spikes shown on the plot are due to the upload of data to the central server. However, a 30-second cycle is evident from this plot, wherein the high power consumption during the first half of this cycle is due to the concurrent execution of the GPS and signal processing threads, and in the second half, power consumption is due to the standalone execution of the signal processing thread. Note that due to resource limitations we can only get one GPS coordinate every 30 seconds on the Nokia N95 platform.

Table 2: CPU and RAM usage

	CPU Load (%)	RAM (MB)
Phone Idle	2±0.79	32.86
Ear-phone (Signal processing (thread only)	5.22±3.03	38.06
Ear-phone (Signal processing & GPS threads)	98.15±11.40	38.28

4.3.2 *Memory and CPU Benchmarks*

We also carried out benchmark experiments to quantify the RAM and CPU usage of Ear-Phone running on the N95 using the Nokia Energy Profiler tool. To precisely measure the resource consumption, we enable the screen saver to disassociate the resource occupation of the N95 LCD. We first measure the amount of RAM and CPU usage when the phone is idle. Then, we repeat the measurement to determine the power consumption of Ear-Phone with only the signal processing thread running. Finally, we repeat with both the signal processing and GPS threads running concurrently. The results in Table 2 show that Ear-Phone uses less than 40% of system RAM. Furthermore, Table 2 shows that the GPS thread dominates the CPU usage of Ear-Phone.

The current Ear-Phone implementation is not optimized for CPU utilization or power consumption since our main concern at this stage is the accuracy of the noise map. Proper techniques can be designed to minimize usage of these resources.

4.4 Performance Evaluation

To evaluate the performance of Ear-Phone as an end-to-end system, we conducted several outdoor experiments. Our primary goal is to investigate the impact of data availability on reconstruction performance. In the experiments, we reconstructed the noise map along a major road intersection in Brisbane, Australia. This intersection includes Mogill Road, a major artery that carries significant traffic and is thus noisy, and Bainbridge Drive, which is a branch road that leads to a residential neighborhood and is hence much quieter. We reconstructed the hourly noise map for time periods (off peak: 14:00 - 15:00 and peak: 8:00 - 9:00) along these road segments. To collect noise samples, we walked along these segments several times within the one hour period with Ear-Phone running on the Nokia N95. The path used is marked with arrows in Fig 7. The travel time was approximately 5 minutes for each walk (from start to end of the segment) and we traveled 8 times during a one hour period. Each walk represents a different person walking along the segment and contributing data.

To investigate the impact of data availability on the reconstruction, we reconstruct the noise profile by varying the number of contributing persons, and including the data contributed by the corresponding persons. For each person, we reconstructed the noise profile during his 5-minute travel. We reconstructed separately for Mogill Rd and Bainbbridge Drive. Using the reconstructed $LA_{eq,T}$, we computed $LA_{eq,LT=1hr}$ using Eq.(4). We repeated this process to compute $LA_{eq,1hr}$ using measurements from multiple people. Figs. 8 and 9 show the impact of measurements included from a varying number of persons on the reconstruction accuracy during off-peak and peak hours respectively.

When we use data from only one person, the reconstruction does not reveal any distinct patterns along the noisy and quiet streets. In fact, the reconstruction appears to be random (In our experiments, a single person collects only a small amount of information of the temporal-spatial noise profile, which is not sufficient for the Compressive sensing based reconstruction algorithm to succeed. This is why the reconstruction is random.). However, when we include data from multiple persons, the reconstruction gradually reveals the contrast between the noisy and quiet streets. Furthermore, after a certain threshold, increasing data contributors does not improve the reconstruction accuracy significantly. For example, comparing Fig. 8(c) and Fig. 8(d), it is evident that the reconstruction achieved by data from 4 people is similar to that from 6 people. A similar behavior can be seen in Fig. 9(c) and Fig. 9(d).

During these experiments, we simultaneously measured the $LA_{eq,LT}$ using our commercial sound level meters placed midway along Mogill Rd and Bainbridge drive. Comparing the reconstructed noise map with the commercial sound level meter readings, we find that we need measurements from at least 5 people during peak hour and from a minimum of 4 people during off-peak hour, for a reconstruction comparable to the commercial sound level meter. Note that data from 4/5 users was sufficient for the noise profile we considered in our experiments. It may change for different noise profiles. The amount of data needed depends on both the noise profile and the percentage of missing data. This topic is studied in the following section.

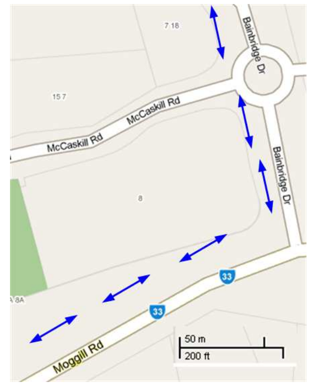


Figure 7: Data collection route

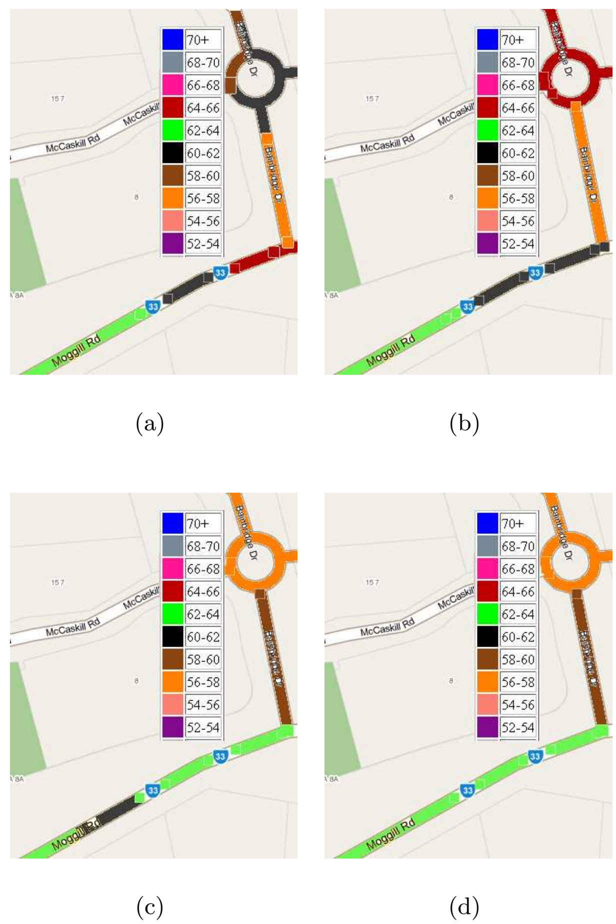
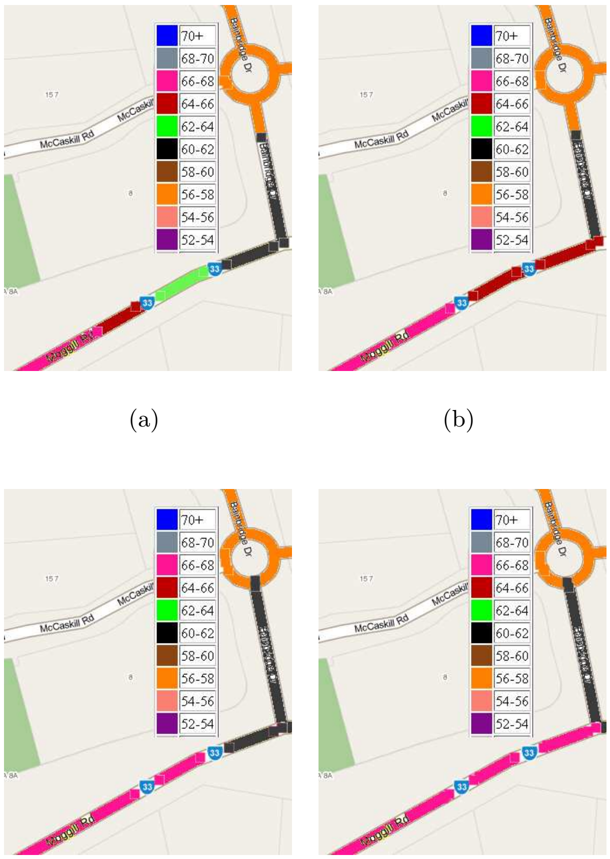


Figure 8: Noise map reconstruction during an off peak hour (2:00pm-3:00pm) using data from (a) 1 person, (b) 2 persons, (c) 4 persons and (d) 6 persons.



(a) (b)

(c) (d)

Figure 9: Noise map reconstruction during a peak hour (8:00am-9:00am) using data from (a) 1 person, (b) 3 persons, (c) 5 persons and (d) 7 persons.

5. SIMULATION

Real experiments certainly provide valuable information. However, real experiments are not repeatable. Conducting a real experiment on a large scale is expensive and time consuming. We therefore conduct simulation experiments where factors such as the number and mobility patterns of volunteers, sensing strategies (see Section 3.2.3) etc. can be varied easily. In this section, we will first describe how we perform measurement campaigns to collect noise profiles which will be fed into the simulation as ground truth. Next, we will describe the simulation itself and performance evaluation in terms of reconstruction accuracies.

5.1 Simulation Design

As in Section 4, we limit our consideration to noise measurements along a road, which can be modeled as a scalar field over a uniform 2-dimensional grid of cells with one spatial and one temporal dimension. We assume that each cell has a spatial width of D meters and a temporal width of T seconds. We use the ordered pair (i, j) to refer to the cell bounded by the spatial interval $[(i-1)D, iD]$ and temporal interval $[(j-1)T, jT]$. Assuming that $i \in N_s = \{1, 2, ..., n_s\}$ and $j \in N_t = \{1, 2, ..., n_t\}$, the reference grid covers a length of $n_s D$ meters and a duration of $n_t T$ seconds. We assume that the equivalent noise level $\mathrm{LA_{eq,T}}$ measured over each cell is almost constant. Now let $d(i, j)$ denote the equivalent noise level $\mathrm{LA_{eq,T}}$ measured in cell (i, j), then a *noise profile* S is defined as the set of all $\mathrm{LA_{eq,T}}$ measured over the defined grid, i.e. $S = \{d(i, j)\}_{(i,j)\in N_s\times N_t}$.

Our first task is to conduct a number of measurement campaigns to obtain *reference noise profiles* which we can feed into the simulation as ground truth. We conducted four experiments to collect $\mathrm{LA_{eq,1s}}$ under a variety of noise conditions and settings. The experimental conditions and parameters used are summarized in Table 3. During each of these experiments, we measured $\mathrm{LA_{eq,1s}}$ along Anzac Parade, which is a major artery road in Sydney. This road has two-way traffic with 3 lanes in each direction. The traffic flow was reasonably high as indicated by the mean noise level in Table 3. We used 6 MobSLMs (HP iPAQ) to capture the reference noise profile and placed them in 6 equidistant locations along the road with the microphone pointed towards the road. Different spatial separations are used in the experiments, see Table 3. The clocks on the phones were synchronized to ensure that all phones start and stop sampling at the same time. The MobSLMs measured $\mathrm{LA_{eq,1s}}$ during the experiment and stored the data in a text file which was downloaded to a computer at the end of the experiment. From each experiment, we created a reference noise profile, where $|N_s| = 6$ and $|N_t|$ is the experimental duration in seconds. We deliberately conducted one experiment (see Table 3) with a side road between the mobiles to create a reference profile with high noise variation (side road divides the traffic flow, therefore noise levels on either side of the road typically have high difference.).

Our simulation considers only discrete agent (we refer to simulated volunteers as agents) movements. Let $d_i \in [0, n_s D]$ denote the position of the agent at time iT seconds. The location of this agent at time $(i+1)T$ is given by $d_{i+1} = d_i + V_i T$ where V_i is the average speed (in ms^{-1}) of the agent in the time interval $[iT, (i+1)T]$. The value of V_i is assumed to be uniformly distributed in $[0, 1.11]$ where $1.11\ \mathrm{ms}^{-1} = 4$ km/hr is the typical walking speed [3]. The sign of V_i determines the direction of movement. In our setting, the agent is in cell $(\lceil \frac{d_i}{D} \rceil, i) \in N_s \times N_t$ at time iT, where $\lceil u \rceil$ denotes the smallest integer that is greater than or equal to u. We consider a particular agent and let $W \subset N_s \times N_t$ denote all the cells visited by this particular agent. To simulate urban sensing, we assume that an agent does not take samples at all visited cells (Due to privacy concerns, volunteers may not contribute samples near their home or office. The microphone may be in use for conversation).

Let $\tilde{W} \subset W$ denote the set of all cells whose data is contributed by this agent.

5.1.1 Simulating Sensing Strategies

In the projection method, an agent uses the $\mathrm{LA_{eq,1s}}$ samples collected in the cells in $\tilde{W}$ to form a projection. Recall from Section 3 that a projection is essentially a linear combination of the data. The agent computes

$$\tilde{y} = \sum_{(i,j)\in\tilde{W}} d(i,j)\eta(i,j) \qquad (6)$$

where $d(i,j)$ is the $\mathrm{LA_{eq,1s}}$ sample collected at cell (i,j) and $\eta(i,j)$'s (with $(i,j) \in \tilde{W}$) are $|\tilde{W}|$ random numbers drawn from the standard Gaussian distribution. The agent transmits the projected value $\tilde{y}$ to the central server, along with the seed used to generate the random coefficients of the projection vector. In the raw-data method, the agent sends $d(i,j)$ values and $(i,j) \in \tilde{W}$ (note that i and j represents location and time respectively) to the central server.

Let $\tilde{S} = \{d(i,j)\}_{(i,j)\in\tilde{W}} \subset S$ be the $\mathrm{LA_{eq,1s}}$ samples collected by volunteers. The reconstruction operation can be viewed as the estimation of the missing samples in the noise profile S from the information in $\tilde{S}$. Let $\hat{S} = \{\hat{d}(i,j)\}_{(i,j)\in N_s\times N_t}$ be a reconstruction of S. Then we compute root mean square (RMS) reconstruction error by:

$$S_{rms} = \sqrt{\frac{1}{n_s \times n_t} \sum_{1\le i<n_s, 1\le j<n_t} \left(d\,(i,j) - \hat{d}\,(i,j)\right)^2} \qquad (7)$$

5.2 Performance Evaluation

As discussed earlier, the key benefit of using compressive sensing is the ability to accurately reconstruct the spatio-temporal sensed field from incomplete and random samples. We now proceed to study the tradeoff between the reconstruction accuracy, communication overhead and the percentage of missing data for the two sensing strategies discussed in the paper namely: (i) the raw-data method and (ii) the projection method. We used the 4 different noise profiles as a reference and evaluated the reconstruction performance under varied mobility patterns and number of agents. In Figs. 10(a) to 10(d) we plot the reconstruction accuracy as a function of sampling requirements for our reference noise profiles. We observe that the raw-data method has better reconstruction accuracy for all 4 reference profiles, specifically when the amount of missing samples is large. We observe that due to the aggregation of data, reconstruction becomes difficult in the projection method (Note that the aggregation inevitably leads to loss of information, but the projection method can reduce the communication requirements, see the next paragraph.). Except for profile 4, Ear-Phone can reconstruct the profiles to within 3dBA error with 40% or fewer missing samples. The increase in sampling requirements from profile 1 to profile 4 can be explained in terms of the profile compressibility. One way to determine the compressibility of a profile is to study the percentage of transform coefficients needed to approximate a profile to a given level of accuracy. The last column of Table 3 shows that profile 1 is the most compressible while profile 4 is the least compressible.

To demonstrate the reconstruction quality, we plot a section of the reconstructed profile in Fig. 11. A total of 3 sections are shown in Fig. 11 for different percentages of missing samples for the raw-data method. Note that the reconstruction is pretty accurate at the cell level.

We now discuss the communication requirements of the raw-data and projection methods as a function of their reconstruction accuracy. Let C_{ref} denote the number of bytes returned, if $\mathrm{LA_{eq,1s}}$ samples from all the cells of our profile are returned and let C_{method} denote the corresponding number of bytes returned by either raw-data or projection method. Fig. 12 shows a typical plot of (we plot only the result from experiment 4 due to space restrictions) C_{method}/C_{ref} as a function of the reconstruction error. We observe that, to limit the reconstruction error within 3dBA (what humans cannot perceive), the projection method and the raw data method reduce the communication costs by 30% and 20% respectively compared to the state-of-the-art sampling technique. However, for a high reconstruction error (an increased amount of missing information), the raw-data method is more communication efficient than the projection method.

6. RELATED WORK

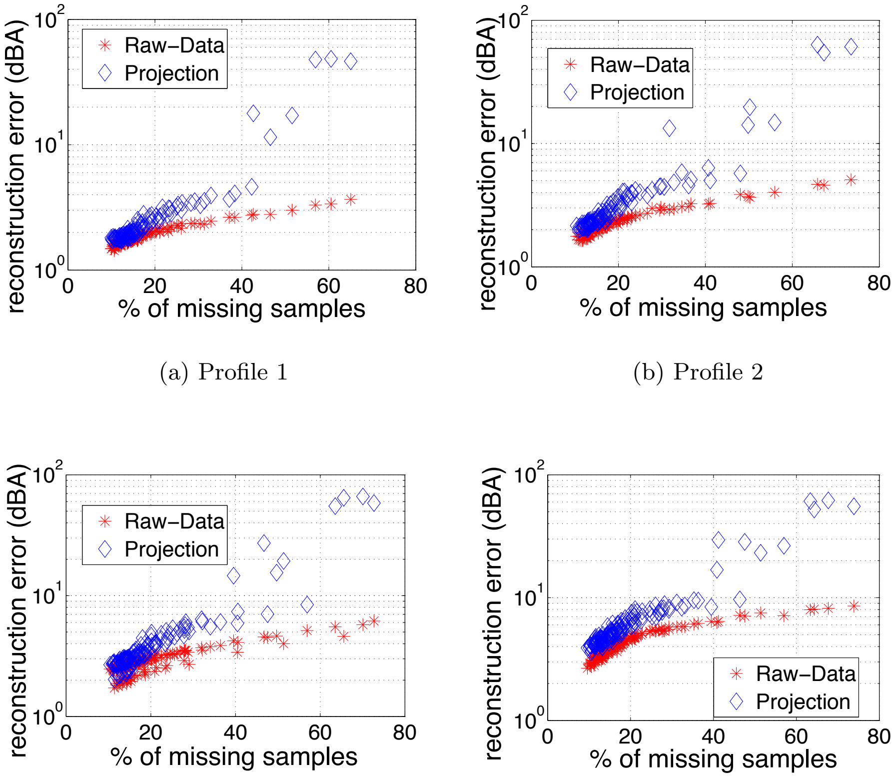
There are a number of efforts in the deployment of urban sensing applications, on the study of incentives to improve participation in human computation systems, and on improving the trustworthiness of participatory sensing. However, we focus our attention on the following.

In [21], the authors survey technical issues influencing the design and implementation of systems that use mobile phones to assess noise pollution. However, they do not provide an end-to-end system, and they do not study the problem of reconstructing the noise map from incomplete and random samples.

Noisetube [16] is a recently developed platform to generate a collective noise map by aggregating measurements collected by the public. As the authors do not provide any details on how they perform data aggregation, we cannot contrast EarPhone with this work.

Recent research in plenacoustic functions [2] studies the sampling requirement of an acoustic field. While the work in [2] deals with a continuous signal, our work considers a discrete signal over time and space. Specifically, we consider the equivalent noise level over a physical area and time duration.

Work presented in [13] studies the compressibility of acoustic signals in both spatial and temporal dimensions. A limitation of their work is that it is based on a single acoustic source in a laboratory setting. In ad-



(a) Profile 1 (b) Profile 2

(c) Profile 3 (d) Profile 4

Figure 10: Percentage of missing data (x-axis) and its impact on reconstruction accuracy expressed in RMS error (y-axis).

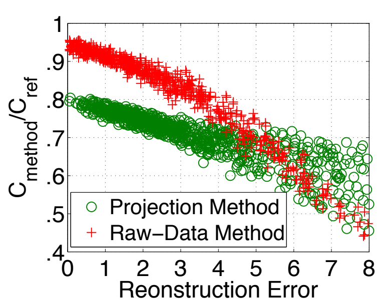


Figure 12: Reconstruction accuracy VS communication overhead

dition, they aim to reconstruct the pressure waveform. This is different from our focus on studying the compressibility of temporal-spatial field of noise levels in an outdoor environment, which are influenced by multiple acoustic sources.

Community Sensing [15] uses a traditional interpolation framework to estimate missing data, when data is obtained via crowdsourcing. In contrast, we apply compressive sensing to show that temporal-spatial noise profiles are in fact compressible and clarify the sampling-accuracy trade-off.

Compressive sensing has so far been applied in traditional low-power wireless sensor networks [18, 8, 5]. For example, Compressive Wireless Sensing (CWS) [5] derives a method to compute the projection using the wireless channel. However, CWS cannot be applied to urban sensing because CWS requires the entire data set to form the projection. In this paper, we have proposed sensing strategies that are suitable for urban sensing.

7. CONCLUSIONS AND DISCUSSION

In this paper, we presented the design, implementation and evaluation of Ear-Phone, an end-to-end noise pollution mapping system based on participatory urban sensing. Ear-Phone comprises signal processing software to measure noise pollution at the mobile phone, as well as signal reconstruction software and query processing software at the central server. To address the problem of noise map reconstruction from incomplete data samples, a key issue in crowdsourced sensor data collection, we exploit the compressibility of the spatial-

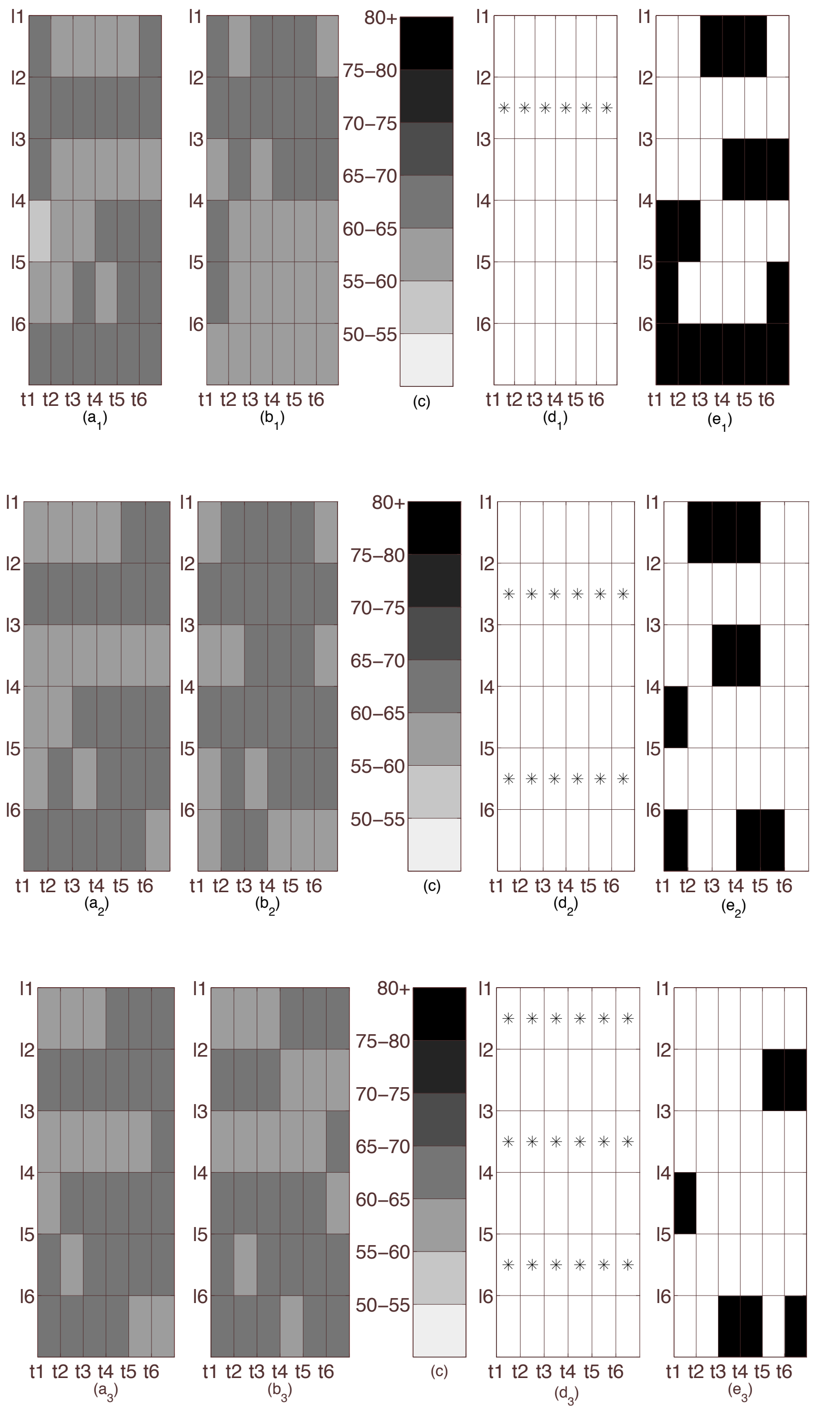


Figure 11: This figure shows the reconstruction performance at the cell level. Each row of this figure consists of 5 sub-figures (a_i), (b_i), ..., (e_i) where $i = 1, .., 3$. Each row ($i = 1, 2, 3$) shows the reconstruction of a section of the profile for a given percentage of missing data. The percentage of data used for rows 1, 2, 3 are, respectively, 18.42%,34.73% and 45.03%. Sub-figure (a_i) shows a section of the reference profile. Note that each section consists of 6 locations (l1,...,l6) over a duration of 6 seconds (t1,..,t6). The same reference profile is used for all 3 rows. (c) The scale of noise levels (d_i) * in a cell means the $\mathbf{LA}_{\mathrm{eq,1s}}$ sample from that cell is used in the reconstruction. (e_i) Reconstruction error. A black-filled cell indicates that the error for that cell is more than 3 dBA. The more white cells the better reconstruction.

Exp No.	Date and time	Mean, Standard Deviation of sound level (dBA)	Spatial separation (meters)	Duration (min)	Continuous road segment without side roads	% of DCT coefficients needed to approximate the profile to within 1 dBA RMS error
1	21/08/08 3:00 pm	73.05,2.95	10	20	yes	27.83
2	21/08/08 4:30 pm	70.09,4.43	10	15	yes	35.15
3	29/08/08 5:14 pm	70.43,5.16	50	15	yes	39.94
4	01/09/08 6:24 pm	71.22,5.55	50	10	no	44.14

Table 3: Experimental settings for collecting the reference noise profiles

temporal noise profile and apply recently developed reconstruction methods from compressive sensing. We study the sensing and communication requirements of Ear-Phone. Using simulation experiments, we show that Ear-Phone can recover a noise map with high accuracy, allowing nearly 40% missing samples while reducing communication costs by 30%. Two different noise mapping experiments report that Ear-Phone can accurately characterize the noise levels along roads using incomplete samples.

Mobile phones are often carried inside bags or pockets, while our experiments were conducted with a phones held in a volunteer's palm. In future work, we plan to incorporate context-awareness in our system such that Ear-Phone only samples ambient noise when the phone is in the right environment.

8. ACKNOWLEDGEMENTS

We would like to thank the anonymous reviewers for their helpful comments, Prof. Prabal Dutta for shepherding this paper and Dr. Brano Kusy (CSIRO) for being an internal reviewer. This work is supported by ARC Discovery grant DP0770523. N. Bulusu is also supported by NSF grant 0747442.

9. REFERENCES

[1] Australia/New Zealand Standards Committee AV/5. Australian Standard: acoustics description and measurement of environmental noise. AS 1055.3 1997,Part 3–Acquisition of data pertinent to land use.

[2] T. Ajdler and M. Vetterli. The Plenacoustic function, sampling and reconstruction. In *WASPAA*, page 147, 2003.

[3] Alberta Center for Active Living. Watch your steps:pedometers and physical activity. *WellSpring*, 14(2):489–509, 2003.

[4] Audacity. Free, cross-platform sound editor and recorder. `http://audacity.sourceforge.net`.

[5] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak. Compressive wireless sensing. In *IPSN*, pages 134–142, 2006.

[6] J. Burke et al. Participatory sensing. In *WSW06: Mobile Device Centric Sensor Networks and Applications*, 2006.

[7] Center Technology Corp. Center c322.

[8] C. T. Chou, R. Rana, and W. Hu. Energy efficient information collection in wireless sensor networks using adaptive compressive sensing. In *In Proc. LCN 2009*, pages 443–450, 2009.

[9] DEFRA. Noise mapping england. `http://www.noisemapping.org/`.

[10] Department for Health and Environment of the City of Munich (Germany). Noise maps 2007. `http://tinyurl.com/`.

[11] S. Eisenman et al. Metrosense project:people-centric sensing at scale. In *Workshop on World-Sensor-Web (WSW06): Mobile Device Centric Sensor Networks and Applications*, 2006.

[12] European Union. Future noice policy, com (96) 540 final. European Commission Green Paper, Nov 1996.

[13] A. Griffin and P. Tsakalides. Compressed sensing of audio signals using multiple sensors. In *EUSIPCO 2008*, Aug. 2008.

[14] International Electrotechnical Commission. Electroacoustics - sound level meters - part 2: Pattern evaluation tests, April 2003.

[15] A. Krause, E. Horvitz, A. Kansal, and F. Zhao. Toward community sensing. In *IPSN*, pages 481–492, 2008.

[16] N. Maisonneuve et al. Noisetube: Measuring and mapping noise pollution with mobile phones. In *ITEE 2009*, pages 215–228, May 2009.

[17] National Geospatial-Intelligence Agency (NGA). Datums, ellipsoids, grids, and grid reference systems. *DMA TECHNICAL MANUAL*.

[18] R. Rana, C. T. Chou, and W. Hu. Energy-aware sparse approximation technique (east) for rechargeable wireless sensor networks. In *In Proc. EWSN 2010*, 2010.

[19] R. Rana, C. T. Chou, S. Kanhere, N. Bulusu, and W. Hu. Ear-phone: An end-to-end participatory urban noise mapping system. Technical Report UNSW-CSE-TR-0920, University of New South Wales, Sydney, Australia, October 2009.

[20] S. Santini et al. First experiences using wireless sensor networks for noise pollution monitoring. In *Proc. of the REALWSN'08*, Apr. 2008.

[21] S. Santini, B. Ostermaier, and R. Adelmann. On the use of sensor nodes and mobile phones for the assessment of noise pollution levels in urban environments. In *Proc. of the INSS 2009*.