# Privacy Issues in Urban Computing using Mobile Crowdsensing

Prajakta Joglekar
Department of Computer Engineering
MAEER's Maharashtra Institute of Technology
Pune, India

Vrushali Kulkarni
Department of Computer Engineering
MAEER's Maharashtra Institute of Technology
Pune, India

## ABSTRACT
Urban computing is an exciting area of research with huge amount of urban data being generated every day. Citizens however put their privacy at stake, while generously trying to share data with the society. Nowadays, smart phone is the most common and convenient method of data capture, which has given rise to an emerging paradigm called Mobile Crowdsensing. This work discusses the features of mobile crowdsensing and focuses on the privacy issues in this method of data capture. The work studies privacy threats arising from different sensors and analyses the requirements for privacy. The main contribution of this paper is detailed analysis of how different sensors in the smartphone can unknowingly compromise the user's privacy and reveal his/her lifestyle and routine activities.

## General Terms
Privacy

## Keywords
privacy; mobile crowdsensing; urban computing; sensors; participatory sensing.

## 1. INTRODUCTION
Urban Computing has gained momentum with the advent of 'Smart Cities' concept. The main pillars of a smart city include infrastructure, technology, governance, environment and most importantly, the citizens. With the surge in the usage of smart phones, we, as citizens have become the generators as well as the consumers of massive amount of urban data. Be it health care apps, pollution monitoring apps or social networks, the citizens generate lot of digital footprints wherever they go. No wonder, these smart applications have become an integral part of our routine. Urban computing is indeed about building resilient framework to improve the quality of life of its citizens, and humans as sensors are contributing to it at a large scale. According to [1], urban computing involves data acquisition, integration and analysis to create smart environment. Smart applications like traffic planning, environment monitoring, social recommendation and public safety would be possible only with the involvement of the citizens. According to the latest statistics from Google, the number of mobile phone users in the world would expectedly cross the 5 billion mark by 2019. It is possible to leverage the capability of these phones to collect sensor data, thus eliminating the need to deploy static sensors [2]. "Human as a sensor" paradigm has thus created a near pervasive infrastructure.

The citizens, however, unknowingly put their privacy at stake while generously contributing to these smart applications. Our work is an effort to put forth various privacy issues while using a smart phone.

## 2. MOTIVATION
Our city - Pune (India), has been selected under Smart City initiative. We, the citizens of Pune, can contribute a lot to help build a sustainable framework. Instead of relying on municipal officers to collect and store data, we can use the power of mobile crowdsensing to collect ambient data and send it to a central place, where aggregation and analysis can be done for effective decision making. Citizens, however should be aware of privacy issues while sharing the data. Researchers also must design effective mechanisms to address these issues. Urban data is indeed voluminous and high velocity data and we believe that Big Data should address privacy by design and not as an add-on functionality later.

## 3. MOBILE CROWDSENSING
'Human as a sensor' is an evolving concept, where humans themselves generate data with the help of their devices. These mobile devices have a variety of sensing, computing and communication capability. There are various sensors present on smart phones. Some examples of smart phone sensors are GPS, microphone, barometer, accelerometer, gyroscope, light meter, pedometer, proximity sensor and camera. According to [3], the Mobile Crowdsensing paradigm (MCS) empowers ordinary citizens to contribute data sensed or generated from their mobile devices and facilitates aggregation of such data at a central place. MCS involves active as well as passive participation from users in data collection.

Note: Mobile Crowdsensing would be abbreviated as 'MCS' through the remainder of this work

### 3.1 Terms related to MCS
*3.1.1 Personal sensing*
In personal sensing applications, a participant collects data about herself (e.g. distance walked or sleep time).

*3.1.2 Social sensing*
In social sensing applications, the participant can record traffic patterns or locate parking spots.

*3.1.3 Environmental sensing*
In environmental sensing, the user monitors certain aspects of the ambient environment (e.g. air pollution, noise pollution, potholes)

*3.1.4 Community sensing*
This type of sensing aims to estimate a complex spatio-temporal phenomenon [4].

*3.1.5 Participatory sensing*
Users can choose when and for how long to monitor the events.

*3.1.6 Opportunistic sensing*
Users instruct their mobile device to capture the data unobtrusively and send it to the server.

## 3.2 Architecture of MCS

Mobile Crowdsensing follows Service Oriented Architecture (SOA) in a distributed way. According to Christin[5], the main components of an MCS framework include Tasking, Sensing, Local Processing, Storage, Reporting, Server side processing and Presentation components. The application server centrally stores data of all the participants for mining and knowledge discovery. This makes it vulnerable to attacks and puts the participants' privacy at stake.

# 4. PRIVACY ISSUES IN MCS

What is privacy? In Cypherpunk's Manifesto [6], privacy is defined as "the power to selectively reveal oneself to the world". According to [7], privacy in mobile crowdsensing enables participants to maintain control over the release of sensitive information captured using their smart phone sensors.

## 4.1 How data can reveal your details?

Mobile crowdsensing applications empower citizens to use technology for effective sensing of ambient data, however they can put the privacy of the user at stake. The user here implies the participant who captures the data as well as the end user who queries such data. Details of the participant like his location (latitude and longitude) can be revealed to a finer granularity level, along with the timestamp at which the user was present there. In large scale deployments, where there are many contributors and an authority for coordination, trust relationships are weaker.

For example, in applications like NoiseTube [8], user can record noise levels using their smart phones, and readings are sent to the server in a tuple form: (measurement id, latitude, longitude, date and time, noise level). Thus, if no protection mechanism is applied, the adversary would know the location of the user at that time. Further, if the user records noise levels regularly at around same time and same path every day, the attacker can guess the route and daily routine of the user. If some apps record actual audio samples instead of only the decibel values, then privacy of the user would be compromised. GPS sensor measurements, when shared at a community level can help solve traffic problems, but individually can infer private information about the user, such as his daily commute or location of his home and workplace. Even though the personal details of the user are anonymized, a reverse look-up using location details can help identify the user.

Apart from the participants, end users' privacy can also be endangered. End users can send query to the application server to retrieve sensor readings and while doing so, they reveal information about themselves. Queries and subscriptions can reveal insights about their personal interests as well as disclose their location. For example, if Alice is trying to query the noise levels in certain area X, one may infer that she is interested to buy a house in that area.

One way to preserve participant's privacy is process the data on the device and upload such data instead of raw data. E.g. uploading only the average noise levels instead of raw audio files. The problem with this method is that phone based algorithms consume lot of energy. Another way is to have an end-to-end privacy preserving architecture. This involves use of efficient cryptographic tools, but would guarantee privacy for participants as well as end users, while balancing the computational load.

The next section describes how different sensors of a smart phone can compromise the user's privacy. To the best of our knowledge, these issues have not yet been presented in a consolidated fashion in any other literature work.

## 4.2 Sensors and information inferred

There are nearly ten different types of sensors in a smart phone. These sensors capture ambient data and associate a timestamp with it. The readings captured by sensors reflect the users' ambience in a true way as compared to data captured by WSN sensors. For example, as compared to a static sensor installed at a corner to capture traffic noise levels, a smartphone's microphone sensor can capture noise levels as perceived by its user, as the user always carries the mobile device with him. This also can compromise on his privacy as from the GPS data it is possible to infer the location of the user at the time of taking noise levels. Similarly, combination of GPS readings and accelerometer sensors can help estimate the mode of transportation the user must have taken to reach from source to destination

Table I lists all such sensors and the kind of information they can reveal about the user. The use of these sensors have been described in [9], however, we have also consolidated the type of information each sensor can reveal.

**Table 1. Information revealed by smartphone sensors**

| Sensor | Description | Information revealed |
|---|---|---|
| GPS | Captures current geo-coordinates | Current location of user, his home and office details, work timings |
| Microphone | Used to record sound and noise levels | Human voice and bystanders' talk can get recorded |
| Barometer | Senses changes in barometric pressure and altitude | Whether user is in an elevator |
| Light sensor | Measures brightness of the ambient light | User's environment and sleeping pattern |
| Proximity sensor | Used to turn off the screen while user is on call | Amount of time user spends on the phone |
| Fingerprint sensor | Captures and stores digital image of the fingerprint pattern | Prone to fingerprint spoofing attack |
| Camera | Used to take pictures | Sensitive images and bystander's privacy may be compromised |
| Accelerometer | Detect changes in orientation | Detect user's handwriting, Infer the place to the user walks to regularly |
| Gyroscope | Used to detect orientation of the phone | Whether user is aggressively driving, where the user taps on the screen |

# 5. THREAT MODEL FOR MCS

In a network security model, all parties can be protected from potential eavesdroppers using a Secure Socket Layer (SSL) or Transport Layer Security (TLS) protocol. However, there are internal adversaries who try to find out sensitive information related to the participants or end users, thereby endangering their privacy.

According to [10], following are the types of attacks for MCS system:

## 5.1 Task Tracking Attacks

These attacks are caused by semi-honest entities. In these attacks, the participant may reveal sensitive attributes like location, preferences or race. While the behavior of the user is captured to know whether he/she can perform a sensing task, it does compromise the privacy of that user.

## 5.2 Location Based Attacks

A location based or spatial attack can find out about the trajectory of the participant and her daily commuting locations, thus revealing her identity. This type of attack is common in MCS applications which collect GPS traces by capturing geo-coordinates. A location-based attack is also used to identify a query issuer by associating the query to the query location (i.e. location from which the query is issued)

## 5.3 Malicious Attacks

Narrow tasking is one of the malicious attacks in which the task entity imposes restrictions on certain attributes, which easily discloses the identity of the person taking up the task.

## 5.4 Collusion Attacks

In these type of attacks, several applications collude to link the information of a participant in order to de-anonymize him/her. These attacks are hard to detect and mostly occur due to pre-app permission models. The combined permissions of such colluding apps allow them to carry out attacks that any single app cannot carry out alone.

# 6. PRIVACY REQUIREMENTS OF MCS SYSTEM

From the above sections, it is clear that we need to design effective privacy mechanisms in a mobile crowsensing framework to protect against the threat. As described in [11], the requirements expected of privacy preserving techniques are as follows:

## 6.1 Node Privacy

The sensitive data reported by an individual node (a mobile device) should be protected from the network provider, the service provider (application server) as well as the end user querying the information.

## 6.2 Query Privacy

If the end user is trying to query the data, his personal details (like location and behavior) should not be revealed to anyone else.

## 6.3 Location Privacy

No entity, except the network operator, should be able to learn the location details of an individual mobile node.

## 6.4 Unlinkability

No user should be able to infer if two or more certain measurements were uploaded from the same mobile device.

Along with the above requirements, it is also essential that privacy preserving algorithms should consume minimum energy of mobile devices.

Many different privacy preserving frameworks have been proposed in the literature, which apply techniques like Data Perturbation [5], Cryptography [11], Differential privacy [12], k-anonymity [13], Tessellation [14], Data aggregation [15], Pseudonyms [16] and Access control mechanisms [17]. Table II does a comparison of few such papers on the basis of attacks and threats dealt with, using various privacy techniques.

We have focused on privacy issues in this scope of our work, and would be analyzing the effectiveness of the above techniques in future.

**Table 2. Privacy preserving frameworks for MCS**

| Papers | Attacks addressed / Privacy requirements met | Privacy mechanisms used |
|---|---|---|
| PEPSI [11] | Protection of data from internal adversaries, query and node privacy, report unlinkability, location privacy | Identity based encryption |
| Worker Location Privacy [12] | Location privacy, identity threat, threat from administrator | Differential privacy |
| NoiseTube Prime[15] | Location privacy, protection from internal adversaries, cloud provider and service provider | Cryptography – homomorphic encryption |
| SPPEAR [16] | Collusion attacks, anonymity abuse | Cryptography, Pseudonyms |
| PEPPER [18] | Task tracing attack, querier privacy preserved | Access control |
| PoolView[19] | Malicious attacks | Data Perturbation |

# 7. CONCLUSION

In this work we have proposed the use of mobile crowdsensing framework as a way to capture urban data for a smart city. We have described how different sensors of a smart phone can reveal sensitive details about the user, such as location, routine, lifestyle and activities along with the timestamp associated with those activities. We have also listed various types of attacks in an MCS system and summarized the privacy requirements to be taken care of.

Currently, Mobile crowdsensing systems face major challenges with respect to the quality of data and privacy preservation. Proper incentive mechanisms can ensure good quality of data, however the participation of users can increase only if privacy is preserved. Also, increase in computational complexity due to encryption or other privacy preserving algorithms can drain the phone battery and consume energy and CPU time. Emerging big data technologies should take cognizance of this aspect and involve privacy by design in their architecture.

## 8. REFERENCES

[1] Zheng et. al., "Urban Computing: Concepts, Methodologies, and Applications", ACM Transactions on Intelligent Systems and Technology, Vol. 5, No. 3, Article 38, 2014

[2] Kapadia et. al., "VirtualWalls: Protecting Digital Privacy in Pervasive Environments", Pervasive Computing: 5th International Conference, PERVASIVE 2007, Toronto, Canada, May 13-16, 2007. Proceedings

[3] Guo et. al., "Mobile Crowd Sensing and Computing: The Review of an Emerging Human-Powered Sensing Paradigm", ACM Computing Surveys, Vol. 48, No. 1, Article 7, 2015

[4] Krause et. al., "Toward Community Sensing", Proceedings of the 7th International Conference on Information Processing in Sensor Networks, Pages 481-492, isbn 978-0-7695-3157-1

[5] Christin, "Privacy in mobile participatory sensing: Current trends and future challenges", Journal of Sytems and Software, volume 116, pages 57-68, 2016

[6] Eric Hughes, "Cypherpunk's Manifesto", website:http://www.activism.net/cypherpunk/manifesto.html

[7] Christin et. al., "A Survey on Privacy in Mobile Participatory Sensing Applications", The journal of systems and software, vol. 84, no. 11, pages 1928-1946, Elsevier, 2011

[8] I.N. Athanasiadis et al., "NoiseTube: Measuring and mapping noise pollution with mobile phones", Information Technologies in Environmental Engineering: Proceedings of the 4th International ICSC Symposium Thessaloniki, Greece, pages 215-228, 2009

[9] Lane et al., "A survey of Mobile Phone Sensing", Adhoc and Sensor Networks, IEEE Communication Magazine, September 2010

[10] Pournajaf et. al., "Participant Privacy in Mobile Crowd Sensing Task Management: A Survey of Methods and Challenges", SIGMOD, Vol. 44, No. 4, pages 23-34, 2015

[11] Cristofaro and Soriente, "Participatory Privacy: Enabling Privacy in Participatory Sensing", IEEE Network, vol. 27, no. 1, pages 32-36, 2013

[12] To, Ghinita and Shahabi, "A Framework for Protecting Worker Location Privacy in Spatial Crowdsourcing", Proceedings of VLDB Endowment, vol. 7, no. 10, pages 919-930, 2014

[13] Sweeney, "k-anonymity: a model for protecting privacy", Intl Journal of Uncertainty, Fuzziness and Knowledge-Based Systems Vol 10, Issue 5, 2002

[14] Shin et. al., "Location Privacy for Mobile Crowd Sensing through Population Mapping", Sensors 2015, no. 7: 15285-15310

[15] Drosatos et. al., "Privacy-preserving computation of participatory noise maps in the cloud", The journal of systems and software, Vol. 92, Pages 170-183, 2014

[16] Gisdakis, Giannetsos and Papadimitratos, "SPPEAR: Security \& Privacy-Preserving Architecture for Mobile Crowd-Sensing Applications", Proceedings of the 2014 ACM Conference on Security and Privacy in Wireless \& Mobile Networks, Oxford, UK, pages 39-50, 2014

[17] Shebaro et. al., "Context-Based Access Control Systems for Mobile Devices", IEEE Transactions on Dependable and Secure Computing, vol. 12, pages 150-163, 2014

[18] Dimitriou, Krontiris and Sabouri, "PEPPeR: A querier's Privacy Enhancing Protocol for PaRticipatory sensing", Security and Privacy in Mobile Information and Communication Systems: 4th International Conference, MobiSec 2012, Frankfurt Main, Germany, June 25-26, 2012

[19] Ganti et. al., "PoolView: Stream Privacy for Grassroots Participatory Sensing", Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, SenSys, pages 281-294, 2008.