A-PIE: An Algorithm for preserving Privacy, quality of Information, and Energy consumption in Participatory Sensing Systems

Idalides J. Vergara-Laurens^{a,*}, Diego Mendez^b, Luis G. Jaimes^c, Miguel Labrador^d

^a*Department of Electrical and Computer Engineering - Universidad del Turabo, Gurabo, Puerto Rico*

^b*Department of Electronics Engineering - Pontificia Universidad Javeriana, Bogota, Colombia*

^c*College of Science, Engineering, and Mathematics, Bethune-Cookman University, Daytona-Beach, FL*

^d*Department of Computer Science and Engineering - University of South Florida, Tampa, FL*

Abstract

We present A-PIE, a hybrid privacy-preserving mechanism for Participatory Sensing Systems that provides a high level of privacy protection as well as a high quality of information while minimizing the energy consumption. A-PIE takes into consideration the variability of the variable of interest to identify clusters, and divides the target area in cells of different sizes. A-PIE applies anonymization or double-encryption to balance privacy protection, quality of information and energy consumption based on the cell's size. Extensive experimentation, using a real air monitoring system, shows the superior performance of the proposed mechanism when compared with most important privacy-preserving mechanisms.

Keywords: Privacy-Preservation, Quality of Information, Energy Consumption

1. Introduction

The main characteristic of Participatory Sensing (PS) is the use of mobile devices to collect and report data in order to produce information [1]. As such, PS is limited only by the number of cellular phone users available in the region of interest. This characteristic becomes one advantage when PS systems are compared with traditional monitoring systems. Take a pollution monitoring system as an example: current systems are usually formed by few static and expensive stations located in strategic places in a city, county or state. These stations measure and report the measurements to a central infrastructure, which uses such data to estimate the value of the variables of interest for the entire region. However, the main restriction of this type of system is the quality of the estimation in regions far from the monitoring stations because those estimations could be too coarse, and in some cases imprecise [2]. For instance, Dutta *et al.* [3] presents the Common Sense project to allow people to measure their personal exposure to pollution. On the other hand, Mendez *et al.* [4] tackle this problem by taking advantage of the large number of participants of a PS system and their mobility. A similar system is SecondNose [5] which collects environmental data to monitor some air pollution indicators to foster participants' reflection on their overall exposure to pollutants. Finally, Devarakonda *et al.* [6] presents a vehicular-based mobile approach for measuring fine-grained air quality in real-time designing one device that can be deployed on public transportation, and the second a personal sensing device. These systems sense the level of pollution as people travel to build accurate pollution maps that can be used by governmental organizations to monitor and control the amount of gases placed in the atmosphere by some industries and by the community.

*Corresponding author

Email address: ivergara@suagm.edu (Miguel Labrador)

However, as with any other new technology, PS also brings new problems and challenges that need to be addressed if we want this new sensing paradigm to bring all the expected benefits. Among the most important challenges, we have the following:

- *Inference*: Since PS applications collect huge amounts of data, more precisely, 2-dimensional data series with gaps in time and space, PS systems need to use interpolation and inference techniques in order to effectively fill such gaps and present appropriate results to the final users.
- *Data accuracy and validity*: Since sensors are supposed to be cheap and they are in the hands of non-expert users, there is a lot of room for sensor failure and human mistakes. Mistakes in data collection, sensor's manipulation, data transmission, as well as the intervention of malicious users, may result in low accurate and/or invalid sensor data.
- *Privacy*: In order to provide a good quality of information to the final user, the actual sensing location and time are desirable for PS systems as well as PS systems require in somehow identify bonafide participants in order to avoid data from malicious participants. Therefore, the submission of sensing reports by the participants implies the disclosure of the participant's spatio-temporal information. Thereby, while they are submitting their sensing reports, they are making their private information available at the same time. The result is that many PS systems will not be able to be implemented in practice because users will not be willing to participate if their private information is not protected somehow [7, 8, 9, 10]. In this context, according to the work in [11], the problem of sharing participant's location information is analogous to a hospital publishing medical records to epidemiologist and other medical researchers: it can be beneficial to society but invades on privacy.

For the first challenge, inference techniques, such as Kriging interpolation, are useful when the correlation between variables, locations, and time (previous measurements). Kriging is a BLUE (Best Linear Unbiased Estimator) interpolator, i.e., it is a linear estimator that matches the correct expected value of the population (unbiased) while minimizing the variance of the observations (Best) [12, 13]. Mendez *et.al.* [14], tackle the second challenge proposing a technique for data verification to detect and remove outliers and invalid data using a local data aggregation process, the algorithm detects abnormal correlations among the measurements in the same cell (close-by participants). This approach is on that the correlation among all random variables in the same cell should follow the same behavior.

Finally, several privacy-preserving mechanisms have been proposed in order to address this privacy protection challenge. These mechanism can be classified as **Anonymization**, **Obfuscation** and **Encryption** based techniques. Anonymization-based mechanisms aim to protect the association between a participant and her private data (e.g., location) transforming the original data into a *generalized* data that represent a set of participants [15] (e.g. Tesselation [16], microaggregation [17], l-diversity [18], data aggregation [19]). Obfuscation-based techniques aim at protecting the same association but transforming the private information of the user (e.g., location) with no consideration of other participants [20, 21] (e.g., Random Perturbation [22, 23], Point-of-interest [24, 25], negative survey [26]). Finally, encryption-based mechanisms aim to protect the privacy of the participants using cryptographic techniques for the transmission and storage of the reported data [27, 28] (e.g., group signature [29], double encryption [30]). However, although several privacy-preserving and inference mechanisms are available for PS systems that address those challenges, one important problem is that these mechanisms have been devised and studied in an isolated or independent manner, as if they were the only mechanisms working in the system and being applied to the reported data. For instance, privacy mechanisms focus on participants' protection and do not consider the data validity and accuracy problem.

This paper presents **A-PIE**, an **A**lgorithm for preserving **P**rivacy, quality of **I**nformation, and **E**nergy consumption as an extension of our previous work published in [31]. The proposed is a hybrid mechanism that considers the data verification process and combines the advantages of anonymization, obfuscation and encryption. The proper combination of these privacy-preserving techniques allows the improvement of the quality of information provided to final users while keeping a high level of privacy protection for participants. The proposed mechanism includes an algorithm that dynamically changes the cell sizes of the grid of the area of interest according to the variability of the variable being measured and chooses different privacy-preserving mechanisms depending on the size of the cell. In the case of big cells, where the variability of the variable of interest is low and therefore it is more important to protect the user's real location, a *Point of Interest*-based mechanism [25] is used for location obfuscation and data

anonymization. On the other hand, in the case when the users are located inside of small cells (where users can be identified easier) a double encryption approach only used to protect their privacy and increase the quality of the information, as the reported location is the real location. This idea is based on the work in [32] where we proposed a mechanism to encourage the movement of the participants by creating dynamic *Density Maps* based on the variability of the variable under study. In addition, this paper presents an evaluation model that allows the comparison between different types of privacy-preserving mechanisms. The proposed model considers the *quality of information* provided to final users in terms of the coefficient of determination (R^2), the *energy consumption* (E) and privacy level provided to participants in terms of the *probability to be located* (P_{loc}) and the *probability to be followed* (P_{bf}).

In our previous work, we presented the *probability to be located* (P_{loc}) to measure the privacy protection when the adversary has access to only one record which is called the *snapshot case* [15]. In this paper, we present the mathematical model to measure the privacy protection level for the historical case (i.e., the adversary gains access to several records) which is more realistic situation in terms of privacy protection. In addition, this paper extends the energy consumption analysis for privacy-preserving mechanisms for PS. Using P-Sense [4], we evaluate A-PIE and other important privacy-preserving mechanisms following the proposed evaluation model. Our results indicate the superiority of A-PIE and show the trade-offs available in terms of these three performance metrics.

The rest of the paper is organized as follows: Section 2 presents the system model including the architecture, data processing model, and the threat model; Section 3 presents the actual algorithm for A-PIE; Section 4 presents the performance evaluation model, and the experimental settings for this paper; Section 5 contains the quality-of-information analysis including the evaluation model, and a comparison of A-PIE and the most important privacy-mechanisms available in the literature; Section 6 presents the energy consumption analysis including the energy evaluation model and experimental results; Section 7 presents the privacy analysis for the snapshot case including the mathematical model, results, and trade-off; Section 8 presents the privacy analysis when the historical case is taken into consideration followed by the conclusions and future work for this research in sections 9 and 10 respectively.

2. System Model

In this section we describe the system architecture, the data processing model, and the threat model used in the paper.

2.1. System architecture

Figure 1 shows the system architecture presented in [25] and used here to implement and evaluate the different privacy-preserving mechanisms studied in this work. The PS system consists of the following four independent entities:

Figure 1. System architecture [25].

- **Mobile nodes:** set of mobile devices with enough computing power and communication capabilities to perform the sensing task, run the privacy mechanism, and transmit the data to the application server.
- **Data broker:** responsible for sending the list of points of interest to mobile nodes, receiving the sensing reports, validating the mobiles' signature, performing the initial data aggregation, and forwarding the anonymized data to the application server.

Figure 2. The data processing model defines the four stages in our systems.

- *Application server*: responsible for performing the inference process and providing the information to the final users.
- *Points of interest generator server*: responsible for dividing the target area into cells and assigning locations to points of interest. This process is crucial in order to guarantee data privacy and a good quality of information.

2.2. Data processing model

Our data processing model consists of four modules, as shown in Figure 2. The first module consists of the sensed data acquired and reported by the users. The set of collected data is used as input for the second module, which is in charge of applying the privacy mechanism and producing the protected data. The next module is the data verification module. Data verification, in the context of PS, consists on the process of detecting and removing spatial outliers to properly reconstruct the variables of interest. This module receives the protected data and verifies the validity of the data. Wrong measurements due to malfunctioning sensors and/or tampered data due to malicious users are removed by this module. Consider a pollution monitoring system where certain users would like to modify the sensed data in order to report a lower level of pollution in a specific area, and therefore, avoid penalty fees, and users reporting wrong data because of the low quality of the sensors or malfunctioning sensors. For this module, we rely on our previous work that implemented a hybrid neighborhood-aware algorithm for outlier detection that considers the uneven spatial density of the users, the number of malicious users, the level of conspiracy, and the lack of accuracy and malfunctioning sensors [33, 34]. In those publications, we showed that our algorithm performed as good as the best estimator, effectively removing the wrong and tampered data while considerably reducing the execution time. Figure 3 shows an example of how the data verification module reconstructs the spatial interpolation of the carbon monoxide variable after removing the tampered data. The final module is the inference engine, which receives all (good) measured data and uses Kriging to estimate the variable of interest throughout the entire area. By comparing real data from real locations, real data from modified locations, and estimated data using Kriging, we can determine the quality of information reported to the final users.

In addition, our model defines time slots for performing the sensing task called *Rounds*. Each round corresponds to a time interval in which participants sense and report data to the system. When the current round (R_t) is ended, the inference mechanism estimates the value of the variable of interest in each point of the area of interest, and a new round (R_{t+1}) starts.

Figure 3. The effect of the verification process in the final interpolation for CO_2 . Spatial interpolation of the original data (left), the tampered data (center), and the reconstructed data (right).

2.3. Threat model

The threat model defines the type of attacks that the mechanism is protecting from [15]. For this work, we consider an adversary trying to invade the participant's privacy, this means an adversary trying to establish the association between report(s) and participant identity in order to establish location, and/or the route followed by the participant through accessing reports sent by participants to the central infrastructure. In our case, in order to design a robust mechanism to protect the privacy of the participants, we consider internal and external adversaries; that means, our mechanism considers adversaries that are not part of the system as well as adversaries that are part of the system such as participants and administrators. Therefore, we assume that the adversary could compromise any component of the system, including eavesdropping the communication channel. Thus, we assume that the adversary may have access to the data broker, application server, or both.

In addition, our threat model considers the snapshot and historical cases. According to [15], if the adversary is limited to capture a single report transmitted by a participant, the system may be subject to *snapshot* attacks. On the other hand, if the adversary has access to the collected data (historical data from the repository server) and is able to establish historical correlations between different reports, the attack is called *historical*.

Finally, it is a common assumption in security research that adversaries know the algorithm used for protecting the information since these algorithms are often released to the public [35]. According to Bettini *et.al.* [15], if the privacy-preserving mechanism is known to everyone and the adversary makes use of that knowledge to compromise the system, then the attack is called *def-aware*. On the contrary, the attack is called *def-unaware*. Therefore, our threat model considers the *def-aware* attacks. There is worth in notice that Data verification module detects spatial outliers in such way that wrong measurements, due to malfunctioning sensors and/or tampered data due to malicious participants, are removed protecting the quality of information to the final user.

3. A-PIE: The Proposed Hybrid Mechanism

The proposed privacy mechanism is a combination of the *Points of Interest (POI)* mechanism and the *Double-encryption* technique described in [25] and [30], respectively. At first, the POI mechanism is designed to *obfuscate* the actual location of the participants using a set of pre-defined locations called points of interest. Each POI_i induces a partition of region of interest into sub-regions based on the distance to other POI_j . Thereby, the corresponding sub-region centering at POI_i or Voronoi cell [36, 37] consists of all points closer to POI_i than to any other POI_j . Thus, before reporting to the data broker, the mobile device changes the actual location on the record for the location of the closest POI. The main advantage of the POI mechanism is the fact that there is no need to have a centralized entity knowing the actual locations of the participants. Further, when the data broker receives the obfuscated data, it computes the average of the reported data from all the participants in the same cell and reports only one record per cell to the application server. Thus, this mechanism *anonymizes* the obfuscated data, adding an extra layer of privacy to the model. The main problem of this mechanism is the noise introduced in the data as a consequence of changing the real locations of the users.

The double encryption scheme is designed to prevent the association between the participants and their records in the system without modifying the actual data, therefore improving the quality of information provided to the final user. In this scheme, the mobile device encrypts the sensed data (D_i) using the public key of the application server producing $Enc_{AS}(D_i)$ (i.e., Encrypted data using the application server public key). Keep in mind that (D_i) contains the location reported by the user and the sensor measurements. In the next step, $Enc_{AS}(D_i)$ is encrypted using the public key of the data broker, and a double-encrypted record ($Enc_{DB}(Enc_{AS}(D_i))$) is transmitted to the data broker. Since the data broker and the application servers are independently managed, the double encryption mechanism makes it more difficult for an adversary to obtain the necessary data to identify the users. When the data broker receives the reported data, it de-encrypts those records obtaining $Enc_{AS}(D_i)$ only. Therefore, an adversary hacking the data broker knows who sent the data but has no information about the data itself. Finally, the data broker sends the encrypted record $Enc_{AS}(D_i)$ to the application server, where the record is de-encrypted to obtain the actual data D_i . In this case, an adversary compromising the application server only, will know the sense data but will not know who originated it.

The question that arises here is: *when should the mobile device use the POI or the double encryption mechanism?* In order to answer this question, we need to refer the reader to the *Inference stage* of the system model. In such stage, the application server applies the *Kriging* technique [12, 13], using the data reported by the participants. The result

of applying Kriging to the reported data is a *Map of estimated values* (M_{R_t}) for round R_t , for each point in the area of interest. This map presents to the final user the values of the variables of interest over the entire area -the pollution maps in our example. Then, based on the current map M_{R_t} , the *Points of interest generator server* (section 2.1) runs Algorithm 1 to define the the new set of points of interest and new set of Voronoi cell for the next round R_{t+1} .

Algorithm 1 Definition of Points of Interest for R_{t+1}

Require: M_{R_t}, k, n, s_{min}

```

1:  $c_{xy} \leftarrow Kmeans(M_{R_t}, k)$ 
2: for each  $c_{xy_i} \in c_{xy}$  do
3:    $V_i \leftarrow Voronoi(c_{xy_i})$ 
4:    $g_i \leftarrow \nabla M_{R_t}(V_i)$ 
5: end for
6:  $POI_{xy} \leftarrow c_{xy}$ 
7:  $g_{min} \leftarrow minimum(\vec{g})$ 
8:  $g_{max} \leftarrow maximum(\vec{g})$ 
9: divide  $[g_{min}; g_{max}]$  into  $n$  subinterval  $I_j$ 
10: for  $i = 1$  to  $k$  do
11:   classify  $g_i$  in the appropriate  $I_j$ 
12:    $\Phi_{xy} \leftarrow Kmeans(M_{R_t}(V_i), j)$ 
13:    $POI_{xy} \leftarrow POI_{xy} \cup \Phi_{xy}$ 
14: end for
15: for each  $POI_{xy_i} \in POI_{xy}$  do
16:   if  $Size(Voronoi(POI_{xy_i})) < s_{min}$  then
17:     Mark  $POI_{xy_i}$  for encryption
18:   end if
19: end for
20: return  $POI_{xy}$ 

```

Algorithm 1 is the most important piece of A-PIE. The main idea to calculate the variability of the variable of interest (e.g., pollution, temperature) after each round and change the size of the cells according to it. If the measurements present very low variability (measurements are within a small range of variation) inside a cell, the cell will very likely remain of the same size and the mechanism will choose to obfuscate the data, i.e., the users will send the reports using the location of the POI of their respective cells. On the other hand, if the variable of interest presents high variability (e.g., very different pollution or temperature values in different zones of the same cell), the algorithm will find those zones with different values and create new cells. If these new cells are smaller than a minimum cell size s_{min} , the mechanism will choose to encrypt the data because otherwise the user might be recognized considerably easier (the user will be confined to a smaller area). In this manner, when the mechanism obfuscates the data, it protects the privacy of the user more and saves energy, and when it encrypts the data, it provides more accurate information about the variable of interest but it spends more energy. These trade-offs will be explored later in the paper in more detail.

Algorithm 1 requires the *Map of estimated values* (M_{R_t}), the number k of initial POIs, the maximum number of subcells (n), and the minimum cell size (s_{min}) as input parameters. The first step seeks to find k cluster centroid locations (c_{xy}) in M_{R_t} . Next, for each centroid c_{xy_i} , the algorithm computes the *Voronoi cell* (V_i) in M_{R_t} and the *gradient of change* (g_i) for the variable of interest in the Voronoi cell. The minimum and the maximum gradient of change define an interval I , which is subdivided into n subintervals $I_1, I_2, \dots, I_n$. In the next step, each gradient of change g_i is classified in the appropriate subinterval I_j and the Voronoi cell is subdivided finding j clusters centroid locations (Φ_{xy}). Note that the new list of POI_{xy} for round R_{t+1} is the union between the set c_{xy} and each subset Φ_{xy} . Finally, the size of the cell centered on each POI_{xy_i} is compared to the minimum cell size (s_{min}) in order to determine which cells are too small to guarantee a good privacy level. Thus, when located in those cells, the mobile device selects the double encryption technique; otherwise, the mobile device selects the Points of Interest mechanism.

Figure 4 shows how A-PIE changes the shape and size of the grid cells over time according to the sensing conditions on each round to improve the quality of the information. The left figure shows the initial, completely squared grid that in the first round produces a quality of information of $R^2 = 0.719$. Five rounds later, the center figure shows

how the shape and number of cells are changed according to the variability of the variable being measured and now the R^2 improves to 0.831. Finally, the right figure shows a yet different grid and cell sizes that after 10 rounds improves the quality of information metric to $R^2 = 0.900$.

4. Performance Evaluation

This section presents the evaluation model and experimental setup utilized to assess the performance of the proposed privacy-preserving mechanism. In addition, the most important privacy-preserving mechanisms available in the literature are presented.

4.1. Performance Evaluation Model

The proposed evaluation model measures the *quality of information* (presented in section 5), in terms of the *Coefficient of Determination* (R^2), and the *energy consumption* (presented in section 6), in terms of *Joules*.

The evaluation of the *privacy protection* problem requires a more detailed experimentation, hence in section 7 we first present the snapshot case (i.e. the adversary is limited to capture a single report transmitted by a participant) in terms of the *probability to be located* (P_{loc}). After that, in section 8 we present the historical case (i.e. the adversary has access to the historical data from the repository server and is able to establish historical correlations between different reports) in terms of the *probability to be followed* (P_{bf}).

4.2. The mechanisms and experimental setting

In order to collect real environmental data using a PS approach, we utilized the P-Sense system described in [34]. P-Sense is a PS system for air pollution monitoring that measures CO (ppm), CO₂ (ppm), combustible gases (ppm), air quality (4 discrete levels), temperature (F), and relative humidity (%). Figure 5 shows the measurements of four environmental variables after applying a spatial interpolation using a multivariate approach with the combination of the Kriging technique [33]. The figure shows the fine granularity of the measurements provided by P-Sense, definitively not attainable by static environmental stations located miles away from the campus.

The campus of the University of South Florida, in Tampa, Florida was the location for the data collection. The area of the campus is approximately 3.9 km^2 which was represented by a grid of 108×108 unit squares in this project. Figure 4 shows the grid used to represent the USF campus for this project. Thus, each unit square represents 334 m^2 . This mapping allows us to scale the system to cover larger areas, if needed. For instance, an area of interest of 100 Km^2 could be represented using the same grid (108×108) unit squares but now each unit square would represent 8574 m^2 .

Data was collected by a participant during three months, 3 times a day, by one hour each time, 3 to 4 days a week. Each time, the participant was instructed to walk around the campus during hour, sensing the variables of interest, in order to reach physical coverage of the campus. For this paper, we randomly selected 20 hours out of the 108 hours of data collection. For each selected hour, we used the Kriging interpolation technique to build a pollution map. That

Figure 4. Example of the estimation process using A-PIE for Relative Humidity (%) in USF campus, showing the first (left), the fifth (center) and the tenth (right) round.

Figure 5. A snapshot of the variables of interest over the USF campus.

means, we produced 20 pollution maps for the campus. Then, for each generated pollution map, we simulated 50 virtual participants that were deployed following an uniform distribution, and moved around the campus using the random walk algorithm for 10 sensing rounds. We assume that the variable of interest does not change fast from one round to the next in each location.

For comparison, in addition to A-PIE, our proposed hybrid mechanism, we implemented the most widely known privacy-preserving mechanisms for PS [38]. We used the parameters found in [38] for the best *Quality of Information* for each mechanism as follows:

- *Tessellation* This mechanism was implemented with $k = 3$. Additionally, tiles were constructed using x -axis order, meaning that locations are ordered based on x -axis location and the tiles are as big as necessary to contain k participants depending on its value.
- *Points of Interest* This mechanism was implemented dividing the area of interest using a grid of 5×5 cells.
- *Random Perturbation* This mechanism was implemented using a uniform distribution $[1, 10]$ for the displacement of the actual locations in each axis.
- *A-PIE* This mechanism was implemented dividing the area of interest using a grid of 5×5 cells for the first round. The parameters k and s_{min} for algorithm 1 were set to 25 clusters and 200 unit squares (66800 m^2) respectively.

5. Quality of Information Analysis

This section presents the *quality of information analysis* defining its evaluation model and the results of such evaluation.

5.1. Coefficient of Determination (R^2)

The quality of information is given by the coefficient of determination (R^2), which is defined as:

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \bar{y})^2} \quad (1)$$

where y_i are the known values, f_i are the interpolated values using the values modified by the privacy mechanism, and $\bar{y}$ is the mean of the known data. Therefore, the quality of information R^2 provides a measure of how different the real data and the estimations are. By calculating the coefficient of determination, the effect of the privacy mechanisms in the quality of the final interpolations can be measured [2]. The closer R^2 to 1, the better quality of information.

Figure 6. Quality of information reached by the Points of Interest (POI), A-PIE (POI + Encryption), Random Perturbation (Uniform distribution [1, 10]) and the Tessellation ($K = 3$) mechanisms.

5.2. Quality of Information Results

The main objective of the proposed mechanism is to improve R^2 without affecting the privacy of the participants. Figure 6 shows the quality of information achieved by the different mechanisms for the *Relative Humidity (RH)* variable measured by P-Sense at USF. In the case of Random Perturbation, the quality of information is better because of the lower average displacement compared to the POI mechanism. On the other hand, Tessellation has the worst quality of information because of the larger average displacement. Finally, in the case of A-PIE, a significant improvement for R^2 is obtained because the average displacement is decreased. In the first round A-PIE uses the 5x5 grid but for the following rounds, A-PIE is able to detect regions with high variability of the variable of interest and break those cells into smaller cells, thus reducing the average displacement in those regions. Not only that, in those cells smaller than S_{min} , the mechanism makes the mobile user to encrypt the data, which contains the real location of the user, which improves the quality of information even further.

6. Energy Consumption Analysis

This section presents the *energy consumption analysis* defining its evaluation model and the results of such evaluation.

6.1. Energy consumption (E)

The energy consumption of privacy-preserving mechanisms, measured in Joules, is the second performance metric. In order to calculate this metric, three processes have been defined to take place in the mobile device. Assuming that sensed data are stored in the mobile device, the first process creates a packet with one or multiple records (sensing samples) that may have been stored in memory. Once the packet is created, the mobile device applies the privacy-preserving mechanism and modifies the data to protect the privacy of the user. Then, the packet is transmitted to the application server. In order to measure the energy consumption of the privacy-preserving mechanisms, the battery of the cell phone was replaced by an external power supply that reported the voltage and actual current provided to the device every δt_i seconds. Then, Equation 2 was used to calculate the total energy used by the three processes described before.

$$E = \sum_i (V * A_i * \delta t_i) \quad (2)$$

where V corresponds to the voltage and A_i corresponds to the current in the interval δt_i .

Figure 7. Energy consumption comparison between the Points of Interest (POI), Random Perturbation (Rnd Pert), Tessellation (Tssltn), and the Double Encryption (Encrypt) mechanisms.

6.2. Energy-Consumption Results

The experiments to measure the energy consumption of the different mechanisms consisted of the transmission of 100 P-sense records (i.e., *location, timestamp, CO₂, CO, air quality, temperature, RH factor, and combustion gases concentration*) using Wi-Fi and 3G Networks. Each record is anonymized, obfuscated, or encrypted depending on the selected privacy-preserving mechanism. We used a Hewlett Packard E3631A Triple Output DC Power Supply to provide power to the cell phone, a Samsung SPH-D720 Nexus S with an Android OS Version 4.0.3, and a Motorola Droid Xtreme MB810 with an Android OS Version 2.2.1.

6.2.1. Energy Consumption of Each Privacy Mechanism

Figure 7 shows the comparison between the different privacy-preserving mechanisms using the Samsung SPH-D720 transmitting each P-Sense record as a packet through the Wi-Fi network. Obfuscation-based techniques, such as the POI and Random Perturbation, present the lowest energy consumption, as they do not require additional transmissions. Tessellation [29] has a higher energy consumption because it needs to establish a connection with the anonymization infrastructure in order to obtain the tile information before sending the report to the data broker. Finally, the double encryption mechanism is shown to be the worst in energy consumption because it needs to establish a *SSL channel*, which requires the transmission of the key, the double-encryption process, and the transmission of the data. Note that Figure 7 does not show A-PIE because the hybrid mechanism has different levels of energy consumption depending on the percentage of records encrypted.

6.2.2. Energy Consumption of A-PIE

As explained before, a record will be encrypted or obfuscated depending on the cell size. Figure 8 shows the energy consumption A-PIE, our hybrid mechanism, for different percentages of data encrypted using Wi-Fi and 3G

Figure 8. Energy consumption comparison using Wi-Fi and 3G for A-PIE.

Table 1. Processing time for double encryption (milliseconds).

Number of records	Packet creation	First encryption	SSL connection	Packet transmission	Packet time	Total time
10	54.89	6.76	899.83	5.80	967.27	9672.72
25	60.64	9.64	830.64	6.83	907.75	3630.99
50	101.38	19.79	823.34	18.07	962.59	1925.17
100	171.76	33.11	762.46	27.10	994.43	994.43

networks. A 0% means that all records were obfuscated while a 100% means that all records were encrypted. Notice that the larger the percentage of records encrypted, the larger is the difference in energy consumption between Wi-Fi and 3G networks. Comparing Figures 7 and 8, it can be seen how the energy consumption of A-PIE is lower than Tessellation when the number of records encrypted is equal to 0 and equal to the Encryption mechanism when the percentage of records encrypted is 100% (case Wi-Fi). For other values, the energy consumption of the hybrid mechanism increases linearly with the percentage of records encrypted.

6.2.3. Impact of the Establishment of a SSL Channel

An intuitive manner to decrease the energy consumption is to transmit several data records using the same SSL channel. In order to characterize the impact of the establishment of a SSL channel, the 100 P-sense records were reported in set of 10, 25, 50 and 100 records per packet. Therefore, the number of sent packets were 10, 4, 2 and 1 respectively. In the case of double encryption method, the privacy mechanism was divided into four stages: packet creation, first encryption, SSL-channel, and packet transmission through the SSL channel. Table 1 shows the processing time associated to the double encryption mechanism using the Motorola Droid Xtreme MB810 through the Wi-Fi network. Notice that the longest time is associated to the establishment of the SSL-channel due the certificate exchange process. Therefore, a good strategy to improve the energy consumption for encryption method is to decrease the number of times in which the establishment of a new SSL connection is needed meaning the maintenance as long as possible the SSL channel between the mobile node and the data broker.

7. Privacy Protection Analysis: The Snapshot Case

The question for the evaluation of the privacy protection in this case is: *What is the probability that an adversary could identify the location where the participant was located at when he/she performed the sensing?* In order to answer this question, this section presents the performance evaluation of the proposed mechanism from the privacy point of view. It includes a description of the performance metrics and the experimental setting utilized in the evaluation as well as the performance comparison with the most important privacy-preserving mechanisms available in the literature.

7.1. Evaluation Model

The *Probability to be located* (P_{loc}) is proposed to measure the privacy protection level offered, by the available privacy-preserving mechanisms, to participants. In order to calculate P_{loc} in the snapshot case, the following definitions are introduced:

- *Round* (R_t): It is a time slot for performing the sensing task. Each round corresponds to a time interval in which participants sense and report data to the system. When the current round R_t is ended, the inference mechanism estimates the value of the variable of interest in each point of the area of interest, and a new round R_{t+1} starts.
- *Data record* (D_i): It is the i tuple of the form $(L_{xy}, R_t, \vec{B}_i)$ reported by a participant, where L_{xy} is the reported location, R_t is the current round, and $\vec{B}_i$ is a vector containing the sensed values. Thus, D_i does not contain any data that identifies the participant.
- *Map of estimated values* (M_{R_t}): It is a bi-dimensional matrix where M_{xy} corresponds to the values estimated by the inference mechanism for the variable of interest at location (x, y) in round R_t .

- *Cell location function* ($Cell(D_i)$): Given a reported record D_i , this function returns the cell (i.e., Voronoi cell V_j centered at L_{xy}) in which the record was anonymized/obfuscated from, and the size of the cell S_{V_j} .

$$Cell(D_i) \Rightarrow (V_j, S_{V_j}) \quad (3)$$

- *Location estimator function* ($Loc(M_{R_i}, D_i, Perc)$): Given a *map of estimated values* M_{R_i} , a percentile $Perc$, and the i data record D_i in round R_i , this function returns an area A of size S_A in which the adversary concludes, with certainty P_A , that the issuer was located at when she generated record D_i .

$$Loc(M_{R_i}, D_i, Perc) \Rightarrow (A, S_A, P_A) \quad (4)$$

A good quality of information means that the estimated value for the variable of interest is very close to the reported one. Therefore, in order to produce (A, S_A, P_A) , this function generates a distribution of the differences between the reported value for the variable of interest in D_i and each estimated value in the cell where D_i is located. Then, the function selects the $Perc$ percentile of such distribution.

- *Average area where the adversary concluded that the participant was located* ($\bar{S}_A$): The function $AVGLoc(M_{R_i}, D_i)$ is proposed in order to estimate the average area $\bar{S}_A$ where the data record D_i was generated. Given a *map of estimated values* M_{R_i} and the data record D_i in round R_i , this function returns the average size of the estimated area $\bar{S}_A$ in which the adversary concluded that the issuer of record D_i was located at when she generated D_i . In order to compute $\bar{S}_A$, Equation (4) is evaluated using D_i and different values for $Perc$ (e.g., 5%, 10%, ..., 100%) generating a set of estimated areas $\{(A_1, S_{A_1}, P_{A_1}), (A_2, S_{A_2}, P_{A_2}), \dots, (A_k, S_{A_k}, P_{A_k})\}$. Then, Equation 5 is used to calculate $\bar{S}_A$.

$$\bar{S}_A = AVGLoc(M_{R_i}, D_i) = \sum_{j=1}^k (S_{A_j} * P_{A_j}) \quad (5)$$

- *Probability to be located* (P_{loc}): Given an estimate of the average area where the issuer of record D_i was located at, this function computes the probability of finding the actual location of the issuer of the data record D_i in round R_i .

$$P_{loc}(AVGLoc(M_{R_i}, D_i)) \Rightarrow 1 - \frac{\bar{S}_A}{S_T} \quad (6)$$

where S_T is the size of the total area of interest. Notice that P_{loc} increases as $\bar{S}_A$ decreases because a larger $\bar{S}_A$ means a larger area in which D_i was located by the adversary (i.e., a lower probability of finding the actual location). Thus, a lower $\bar{S}_A$ increases the probability of finding the actual location in which D_i was generated.

7.2. The Snapshot Case Results

The experiments aim to measure P_{loc} when the adversary gains access to the data broker, the application server, or both.

7.2.1. Adversary Has Access to the Data Broker

This scenario is equivalent to an adversary compromising the anonymizing infrastructure in the case of k-anonymity techniques. Thus, an adversary compromising these components have access to the participant's actual data (e.g., actual location) in the case of Tessellation. On the other hand, in the case of Random Perturbation and the Points of Interest techniques, the adversary has access to the obfuscated data only. Finally, in the case of A-PIE, the adversary gains access to a portion of the reported data, which has been obfuscated in the mobile device because the other portion has been encrypted and it is not accessible to the adversary in this server.

The first row in Table 2 (P_{loc} at DB) shows the *Probability to be located* in this scenario based on Equation 6. In the case of Tessellation, P_{loc} is 1 because an adversary compromising the anonymization infrastructure has access to the actual data containing the actual location of the participants. In order to compute P_{loc} for Random perturbation, POI, and A-PIE, we use the mathematical model presented in section 7.1. Since the location is modified but the sensed data are not, the adversary, knowing the protection mechanisms, might use infer the area where the record was

Table 2. Probability to be located (P_{loc}) at the data broker (DB), at the application server (AS) and at both (DB + AS).

	Tessellation	Random Pert	Points of Interest	A-PIE
P_{loc} at DB	1	0.979	0.983	0.692
P_{loc} at AS	0.317	0.979	0.400	0.628
P_{loc} at DB + AS	1	0.979	0.983	0.989

generated. Using the equation 4, then we compute the probability that the actual location is located in the inferred area by the adversary for each percentile. Then, we compute the average area following equation 5. There is worth in notice that the adversary has access the collection of POI and the random perturbation function because they are publicly available. Therefore, the adversary knows the cell in the cases of POI and A-PIE. For the random perturbation scheme, we assume that the adversary define a circular area using the perturbation function. These cells and circle are used as S_T (i.e., the general area where the participant was located) in order to compute P_{loc} following the equation 6. In the case of Random Perturbation, P_{loc} is 0.979 because of the low average displacement needed in order to guarantee a good quality of information [38]. On the other hand, the Points of Interest mechanism has a P_{loc} equal to 0.983 because the data broker has the obfuscated records no the actual ones. Finally, A-PIE has a P_{loc} of 0.985 for the data records available at the data broker; however, these records only represent 70.2% of all records because the other ones were encrypted and therefore not available in the data broker. As a result, $P_{loc} = 0.692$ for our hybrid algorithm.

7.2.2. Adversary Has Access to the Application Server

In this scenario, the adversary compromises the application server, which is the component where the inference process is performed. In the case of Tessellation, the adversary does not have access to the actual data but the anonymized data. In the case of Random Perturbation, the adversary has access to the obfuscated data only. In the case of Points of Interest and A-PIE, the adversary has access to the anonymized data by the data broker as well as the actual data reported through the encryption path. Please recall that the amount of encrypted data in the application server depends on the size of the cells; mobile users who are located in cells smaller than S_{min} are the only ones encrypting their records.

The second row in Table 2 (P_{loc} at AS) presents P_{loc} for each privacy mechanism. In the case of Random Perturbation, P_{loc} is the same as the one presented on the first line (P_{loc} at DB) because the data broker and application server have both the same records. In the case of Tessellation with $k = 3$ and sixty users, as used for the experiments, the number of anonymized records is $\frac{1}{3}$ of the actual ones. Therefore, the average tile size is $\frac{1}{20}$ of the total area, meaning that $P_{loc} = 0.95$ for the anonymized records sent by the data broker. Since each anonymized record represents 3 participant records, P_{loc} for each actual record is $\frac{1}{3}$ of 0.95 which is 0.317.

In the case of the Points of Interest mechanism, the application server has access to the anonymized records sent by the data broker. With a 5x5 grid, P_{loc} for each anonymized record is $1 - \frac{1}{25} = 0.96$. Please recall, the Data Broker anonymizes the records located in the same cell (e.g., if the data broker receives three records located in the same cell, it computes the average and reports this average to the application server in one record instead of three records). Our experiments achieve an average of 2.4 records in each cell, producing a $P_{loc} = \frac{0.96}{2.4} = 0.4$ for each reported record from mobile devices. Finally, in the case the A-PIE mechanism, each records is double encrypted or obfuscated using POI depending on the cell size where the participant was located. Therefore, an adversary compromising the application server will have access to the encrypted records plus the anonymized records sent by the data broker. Thus, P_{loc} is the sum of P_{loc} for the encrypted records plus P_{loc} for the anonymized records sent by the data broker. In the case of the encrypted records, $P_{loc} = 1$ for each one of the records because they contain the exact location of the users; however, in our experiments these records only represent 29.8% of the total number of records meaning that 29% of the records where located in smaller cells than the *minimum cell size*. On the other hand, P_{loc} for the anonymized records is 0.961, which is applied to the remaining 70.2% of the records. However, since each anonymized record represents 2.04 of the original records (2.04 is the average number of obfuscated/anonymized users in each cell), $P_{loc} = 0.298 + 0.961 * 0.702/2.04 = 0.628$, as shown in Table 2.

7.2.3. Adversary Has Access to the Data Broker and Application Server

This is the worst case scenario because it assumes that the adversary has compromised both, the data broker and the application server. The third row in Table 2 (P_{loc} at DB + AS) presents the P_{loc} in this case. In the case of

Table 3. Performance comparison varying the minimum cell size (S_{min}).

S_{min}	0	100	200	400
% Encrypted	0%	1.3%	29.8%	62.3%
% POI	100%	98.7%	70.2%	37.7%
P_{loc} at DB	0.987	0.973	0.692	0.369
P_{loc} at AS	0.480	0.485	0.628	0.752
P_{loc} at DB + AS	0.987	0.987	0.989	0.992
Wi-Fi (Joules)	2.43	3.08	15.83	30.47
3G (Joules)	8.35	10.35	53.60	103.08

Tessellation, $P_{loc} = 1$ because the adversary has access to the actual locations from the data broker. In the case of Random Perturbation, P_{loc} is the same presented above because the data in the data broker and application server are the same. Therefore, in this case, it does not make any difference if an adversary compromises the data broker, the application server, or both. In the case of Points of Interest, P_{loc} is the same as in the case where the data broker is compromised because it is the source of more detailed data compared to the application server. Finally, in the case of A-PIE, P_{loc} is the sum of P_{loc} at the data broker $P_{loc} = 0.692$ and P_{loc} of the encrypted records 0.298 at the application server because an adversary compromising both components will have access to all records in the system; therefore the final P_{loc} is 0.989.

From Table 2 and Figure 6, we can conclude that A-PIE provides the best balance between privacy protection and quality of information. The key for the better performance of our hybrid mechanism is the distribution of the records in the data broker and the application server since it avoids an adversary to get access to all the records when compromising one server. If the adversary compromises the data broker, he has access to the obfuscated records but not the encrypted ones, and viceversa with the application server. In addition, the obfuscation technique increases the difficulty for an adversary to identify the location where the records were generated because the records contain the modified location instead of the actual one.

7.2.4. Trade-offs

Table 3 presents the percentage of encrypted records (% Encrypted), the percentage of obfuscated records (% POI), the probability to be located at the data broker (P_{loc} at DB), application server (P_{loc} at AS) and both (P_{loc} at DB + AS), as well as the energy consumption for transmitting 100 P-sense records using Wi-Fi and 3G networks as the minimum cell size parameter S_{min} is varied. In addition, Figure 9 shows the trend of the quality of information depending on the minimum cell size S_{min} . Notice that lower values of S_{min} mean lower percentages of encrypted records, which is translated into lower energy consumption as well as lower quality of information and higher P_{loc} at the data broker. On the other hand, higher values of S_{min} mean higher percentages of encrypted records, which translate into higher energy consumption, higher quality of information, and a higher P_{loc} at the application server.

In conclusion, it is clear that S_{min} has an important effect on the privacy, quality of information, and energy consumption of the users. Therefore, this parameter can be changed by network administrators to better suit their needs and the needs of the users. The results show the better performance of A-PIE to protect the participants' privacy when compared with the other privacy-preserving approaches for the snapshot case.

Figure 9. Quality of information based on the minimum cell size (S_{min}).

8. Privacy Protection Analysis: The Historical Case

One application of PS systems is to support the urban and transportation planning process inferring the most popular travel routes in a city and understanding the human travel behavior [39, 40, 41]. Therefore, PS systems have generated significant interest in the collection of spatio-temporal data for moving objects [42, 43]. However, the availability of such data at a centralized location raises concerns regarding the privacy of mobile device clients since an adversary could use a trajectory-estimation method over such data in order to compromise the users' privacy [21, 44, 45].

The case of an adversary having access to the collected data (historical data from the repository server) allows him to identify the places visited by the participants and their mobility patterns through the reconstruction of their trajectories. For instance, trip destinations can reveal information about a person's health (e.g., frequent visits to a medical doctor or a hospital), lifestyle, net worth, or political associations [46]. Therefore, the analysis of the historical case is an important component in the evaluation of privacy-preserving mechanisms. The question in this case is *How much of the actual trajectory can be reconstructed by an adversary accessing a set of records?*

Suppose an adversary has access to a set of reported records, each record contains a location, a time-stamp and a value for the variable of interest. In order to reconstruct the trajectory of the participant, the adversary needs to determine which records have been issued by that particular participant. Recall that the adversary has access to privacy-protected data, then he may have access to modified or non-modified data based on the privacy-preserving mechanism applied in the system. In the case of a PS system using an non-attribute-modification mechanism, the adversary has access to the actual location, and he might use a traditional tracking algorithm in order to reconstruct the participant's trajectory [42]. In the case of an attribute-modification mechanism, the adversary does not have access to actual but a modified location, therefore, he needs to estimate the location where each record was generated as well as the records that belongs to the same trajectory. Since A-PIE, our hybrid privacy-preserving mechanism, utilizes anonymization, obfuscation and encryption, if the adversary compromises the data broker, he has access to modified data due the anonymization and obfuscation approach. On the other hand, if the adversary compromises the application server, he has access to the actual data that has been decrypted and stored in this server.

8.1. Evaluation Model

The *Probability to be followed* (P_{bf}) and the *Percentage of discovered segments* (P_{seg}) are proposed to measure the privacy protection level offered by the available privacy-preserving mechanisms when historical attacks are taken into consideration. In order to calculate P_{bf} the following definitions are introduced:

- *Roads on the area of interest (MRoads)*: It is a graph $G = \{V, E\}$ where V corresponds to the set of road intersection in the area of interest, and E corresponds to the set of actual roads between intersections.
- *Set of maps of estimated values ($ME_{[t,m]}$)*: Let $ME_{[t,m]}$ be the set of maps with estimated values (M_{R_i}) from round R_t to round R_m (i.e., $ME_{[t,m]} = \{M_{R_t}, M_{R_{t+1}}, \dots, M_{R_m}\}$).
- *Set of records in a given round (S_t)*: Let S_t be a set of data records in round R_t , i.e., $S_t = \{(L_{xy}, R_t, \vec{B}_i)^1, (L_{xy}, R_t, \vec{B}_i)^2, \dots, (L_{xy}, R_t, \vec{B}_i)^N\}$ where $(L_{xy}, R_t, \vec{B}_i)^k$ corresponds to the data record D_k at round R_t defined in Section 7.1.
- *Set of records during a time interval ($H_{[t,m]}$)*: Let H be a set of S_t 's, i.e., $H_{[t,m]} = \{S_t, \dots, S_m\}$, where S_t corresponds to a set of data records in round R_t and S_{t+1} corresponds to a set of data records at time R_{t+1} .
- *Road estimation function ($RoadEst(ME_{[t,m]}, H_{[t,m]}, MRoads, IS_k)$)*: Given a set of maps of estimated values $ME_{[t,m]}$, from round R_t to round R_m , a set of records $H_{[t,m]}$, the roads of the area of interest ($MRoads$) and a given participant (IS_k), this function estimates a set of roads ($RoadsTrack^{IS_k}$) where the trajectory of the participant IS_k is likely to be located.

$$RoadEst(ME_{[t,m]}, H_{[t,m]}, MRoads, IS_k) \rightarrow RoadsTrack^{IS_k} \quad (7)$$

As presented in [38], privacy-protection mechanisms for the reporting process are classified as *attribute modification* or *non-attribute modification* approaches. Therefore, the adversary might have access to different data

based on the type of privacy-preserving mechanism. In the case of *attribute modification* approaches, the system does not receive the actual location of the participant but her modified location. Therefore, it is necessary to estimate the location of the participant in each round R_i . Therefore, given a *map of estimated values* (M_{R_i}), a percentile $Perc$, the data record D_i in round R_i and the map of roads $MRoads$, an intermediate function called $SegRoadEst(M_{R_i}, Perc, D_i, MRoads)$ returns the segments $SegRoads_i$ on $MRoads$ in which the adversary concludes that the issuer was located at when she generated record D_i .

$$SegRoadEst(M_{R_i}, Perc, D_i, MRoads) \Rightarrow SegRoads_i \quad (8)$$

In order to produce $SegRoads_i$, this function generates a distribution of the differences between the reported value in D_i and each estimated value for the variable of interest (e.g., CO_2) in the cell in which D_i is located. Then the function selects the percentile $Perc$ in such distribution, estimates an area A which it is intercepted with $MRoads$ and estimates $SegRoads_i$. Finally, $RoadEst(ME_{[t,m]}, H_{[t,m]}, MRoads, IS_i)$ concatenates the segment roads $SegRoads_i$ in order to produce the complete set $RoadsTrack^{IS_k}$.

On the other hand, in the case of the *non-attribute modification* approaches, the system does have access to the actual location of the participant. Therefore, the adversary aims to reconstruct the trajectory followed by the participant. In general, the trajectory estimation problem is to find a set of N trajectories given a history H spanning P timestamps of exactly N objects [42]. Each trajectory T^k is defined as follows:

$$T^k = \{(x, y)_1^k, (x, y)_2^k, \dots, (x, y)_P^k\} \quad (9)$$

where $k = \{1, 2, \dots, N\}$ and $(x, y)_i^k$ corresponds to the location of the object k at round R_i .

Therefore, in the case of *non-attribute modification* approaches, the set $RoadsTrack^{IS_k}$ corresponds to the estimated trajectory T^k for the participant IS_k .

- *Percentage of discovered segments* (P_{seg}): Given the set of road segments $RoadsTrack_{IS_k}$ produced by the set of data records issued by the same participant IS_k , P_{seg} defines the percentage of the actual trajectory T^{IS_k} that is identified in $RoadsTrack^{IS_k}$.

$$P_{seg} = \frac{|seg(T^{IS_k}) \in RoadsTrack^{IS_k}|}{|T^{IS_k}|} \quad (10)$$

where $seg(T^{IS_k}) \in RoadsTrack^{IS_k}$ corresponds to the segments of the trajectory T^{IS_k} in the road segments $RoadsTrack^{IS_k}$.

- *Probability to be followed* (P_{bf}): Given the set of road segments $RoadsTrack_{IS_k}$ produced by the set of data records issued by the same issuer IS_k , P_{bf} defines the probability that the actual trajectory T^{IS_k} is identified in $RoadsTrack^{IS_k}$.

$$P_{bf} = \frac{|seg(T^{IS_k}) \in RoadsTrack^{IS_k}|}{|RoadsTrack^{IS_k}|} \quad (11)$$

where $seg(T^{IS_k}) \in RoadsTrack^{IS_k}$ corresponds to the segments of the trajectory T^{IS_k} in the road segments $RoadsTrack^{IS_k}$.

8.2. The Historical Case Results

This section presents the experiments and results on privacy protection for the historical case. Since A-PIE offers the best performance for the snapshot case, the experiments in this section aim to measure the probability to be followed (P_{bf}) and the percentage of discovered segments (P_{seg}) for the hybrid privacy-preserving mechanism when the adversary has access to a set of records and is able to establish historical correlations between them.

Algorithm 2 Tracking algorithm using cells ($Tracking_{cells}$)**Require:** $ME_{[t,m]}$, $MRoads$, $H_{[t,m]}^k$, $Perc$, $maxS$ speed

```

1: for  $i = t$  to  $m$  do
2:    $C_i \leftarrow$  cell where  $H_i^k$  was generated
3:    $RC_i \leftarrow C_i \cap MRoads$ 
4:    $FP_i \leftarrow$  the frontier points of  $RC_i$ 
5: end for
6: Let  $Rtrack$  be the list identified points
7: for  $i = t + 1$  to  $m$  do
8:    $cR \leftarrow$  the roads that connect cells  $C_{i-1}$  to  $C_i$ 
9:    $nIP \leftarrow$  points on  $cR$  reachable from  $BP_i$  using  $maxS$  speed
10:   $bnIP \leftarrow$  points on  $nIP$  in  $Perc$ 
11:   $Rtrack \leftarrow Rtrack \cup bnIP$ 
12: end for
13:  $fnIP \leftarrow$  points on  $RC_m$  reachable from  $FP_{m-1}$ 
14:  $fnIP \leftarrow$  points on  $nIP$  in  $Perc$ 
15:  $Rtrack \leftarrow Rtrack \cup fnIP$ 
16: return  $Rtrack$ 

```

8.2.1. Adversary Has Access to the Data Broker

An adversary that compromises the data broker gains access to the reported data that have been obfuscated in the mobile device because the other portion has been encrypted and it is available in the application server. Since the reported location in each message corresponds to a cell, the adversary does not have access to the actual location and therefore the traditional tracking algorithms cannot be used to perform the privacy protection analysis, as they depend on the actual locations.

A new tracking algorithm is proposed for this case. Algorithm 2 is designed to estimate the roads where the participant was located when she generated each report. The main assumption for this analysis is that the adversary has eavesdropped the communication channel between the participant and the data broker. This is the worst case scenario for the participant's privacy because the adversary has the ability to identify the participant that generated each report. Thus, the adversary can make historical correlations between reports.

Recall that the adversary has access to a set of records where each one has an obfuscated location, a time-stamp (i.e., round) and a sensed value. First, Algorithm 2 identifies the cell where the report was generated. Then, the algorithm avoids the unreachable points due time and speed constraints for each cell (i.e., there is a maximum distance that the participant can travel due to the time and maximum speed in the area of interest). In order to find the reachable points, Algorithm 2 selects only the reachable points in the previous round R_{t-1} from each *border point*, defined as a location at the border and at one road in the cell, in the current round R_t . The idea is to identify the location where the participant should be located in the previous round in order to reach the current cell. Figure 10 shows an example for two rounds (R_{t-1} and R_t) including the border points for R_t as well as the reachable and unreachable points in R_{t-1} from R_t . In the example, the participant should be located in the reachable points in round R_{t-1} in order to reach the next cell in round R_t . Finally, given the fact that a high R^2 means a close estimated value to the actual one, the algorithm selects the locations, on each cell, that have a estimated value very similar to the actual one based on the percentile condition.

Figure 10. Example of point identification through time.

Figure 11. Tracking example using Algorithm 2.

Algorithm 2 requires the set of maps of estimated values ($ME_{[t,m]}$) which is publicly available, the map of roads ($MRoads$), the set of reports by the k participant ($H_{[t,m]}^k$), the percentile (i.e., defined in Section 8.1) and the maximum speed in the area of interest ($maxSpeed$). The algorithm estimates the roads (RC_i) in each cell (C_i) where the participant was located when reporting the measurements to the system. Then, the algorithm finds the border points on each road RC_i . For each report, the algorithm finds the reachable points from the border point (BP_i) in the previous cell (C_{i-1}) that meet the percentile classification ($Perc$). Finally, the algorithm finds the reachable points in the last cell (C_m) from the previous cell (C_{m-1}). The time complexity for Algorithm 2 is given by $O(m \times n)$ where n is the size of the $ME_{[t,m]}$ and m is the number of rounds.

The experiment conditions define two scenarios for Algorithm 2, as shown in Figure 11. In the figure, the dotted lines correspond to the estimated roads in which the participant could have been located when reporting to the system, while the x lines corresponds to the actual trajectory for three rounds (R_t, R_{t+1} and R_{t+2}). The first scenario defines cells with low overlapping between rounds. In this case, Algorithm 2 is very accurate because it accurately discovers segments connecting the consecutive cells. The second scenario defines cells with considerable overlapping. In this case, the Algorithm 2 is not very accurate because most of the cell is reachable from the cell of the previous round.

Figure 12 shows the behavior of the P_{bf} and P_{seg} as a function of the percentile used in Algorithm 2. As shown, the greater the percentile, the greater P_{seg} and the lower P_{bf} . This behavior is given by the fact that a greater percentile relaxes the condition of closeness to the actual value, therefore, more points are discovered increasing the value of P_{seg} . However, such relaxation means points that are not in the actual trajectory, decreasing the value of P_{bf} .

8.2.2. Adversary Has Access to the Application Server

In this scenario, the adversary has access to the actual points once decrypted on the application server. Therefore, traditional tracking algorithms can be used in order perform the privacy protection analysis. According to [47], the global nearest neighbor (GNN) approach has the best performance for trajectory reconstruction when the actual

Figure 12. P_{bf} and P_{seg} for A-PIE with $S_{min} = 200$.

points are available. Therefore, the GNN algorithm is used here to perform the privacy protection analysis when the application server is compromised by the adversary. However, given the double encryption scheme, the adversary does not have access to the participant's identity.

In order to calculate P_{bf} and P_{seg} , the GNN algorithm implemented in [48] is used in this case. Such implementation uses the *Munkres algorithm* to obtain the global observation-to-track assignments that minimizes the Σd_{ij}^2 [49]. The experiments shows that it has a $P_{bf} = 0.811$ and a $P_{seg} = 0.803$ when all data are encrypted (i.e., using the actual locations). In the case of A-PIE with $S_{min} = 200$, the algorithm achieves a $P_{bf} = 0.783$ and $P_{seg} = 0.326$ meaning that, on average, 32.6% of each trajectory is identified with an accuracy equal to 78.3%.

8.2.3. Adversary Has Access to the Data Broker and Application Server

In the case of A-PIE, an adversary that compromises the data broker and the application server gains access to the portion of the obfuscated data as well as the portion of the encrypted data. The main difference between those sets of records is given by the fact that if the adversary eavesdrops the communication channel between the participant and the data broker, he has access to the participant's identity; while the records in the application server do not have data about the participant's identity. Therefore, the adversary might use the GNN algorithm in order to reconstruct the participants' trajectories based on the encrypted data in the application server and use Algorithm 2 in order to reconstruct trajectories based on records available in the data broker. Then, using those sets of estimated trajectories, the adversary could conciliate them in order to improve the estimation of all trajectories, as presented in Algorithm 3.

Algorithm 3 is proposed to improve the trajectory reconstruction process. It requires the set of maps of estimated values ($ME_{[t,m]}$) which is publicly available, the map of roads ($MRoads$), the set of reports available in the data broker ($H_{[t,m]}$), the percentile (i.e., defined in Section 8.1), the maximum speed in the area of interest ($maxSpeed$) and the set of records $ENCH_{[t,m]}$ available in the application server. First, the Algorithm runs the *GNN algorithm* in order to estimate the participants' trajectories ($Tracks_{AS}$) using records available in the application server. Then, Algorithm 2 is used to estimate trajectories ($Tracks_{cells}$) for each $H_{[t,m]}^k$ (i.e., historical records reported by user k) in the record set $H_{[t,m]}$. The set of trajectories $Tracks_{cells}$ has two types of trajectories: trajectories with reports in all rounds and trajectories with missing records in some rounds. The trajectories with records in all rounds are collected into the set $Tracks_{DB}$ (i.e., trajectories with only obfuscated points using the point-of-interest mechanism). Then, for each trajectory T^k in the set $\{Tracks_{cells} - Tracks_{DB}\}$, Algorithm 3 finds those trajectories in the set $Tracks_{AS}$ that fill the time gaps and have reachable points from T^k . In order to complete T^k , Algorithm 3 requires that only one track, in $Tracks_{AS}$, meets the T^k requirements. In addition, every trajectory in $Tracks_{AS}$ used to complete a T^k trajectory is removed from the $Tracks_{AS}$ set. Finally, the algorithm returns a set of trajectories $Tracks_{AS}$ estimated

Algorithm 3 Tracking algorithm using cells and encrypted points

Require: $ME_{[t,m]}$, $MRoads$, $H_{[t,m]}$, $Perc$, $maxSpeed$, $ENCH_{[t,m]}$

```

1:  $Tracks_{AS} \leftarrow GNN(ME_{[t,m]}, ENCH_{[t,m]})$ 
2:  $Tracks_{cells} \leftarrow \emptyset$ 
3: for  $k = 1$  to  $m$  do
4:    $H_{[t,m]}^k \leftarrow find(k, H_{[t,m]})$ 
5:    $Tracks_{cells} \leftarrow Track_{cells} \cup Tracking_{cells}(ME_{[t,m]}, MRoads, H_{[t,m]}^k, Perc)$ 
6:    $TH^k \leftarrow time\ holes\ in\ H_{[t,m]}^k$ 
7: end for
8:  $Tracks_{DB} \leftarrow every\ Tracks_{cells}\ with\ TH^k = 0$ 
9:  $Tracks_{DB+AS} \leftarrow \emptyset$ 
10: for each  $T^k \in \{Tracks_{cells} - Tracks_{DB}\}$  do
11:    $T_{enc}^k \leftarrow find\ track\ in\ Tracks_{GNN}\ that\ match\ with\ T^k$ 
12:   if  $|T_{enc}^k| == 1$  then
13:      $CT^k \leftarrow T^k \cup T_{enc}^k$ 
14:      $Tracks_{AS} \leftarrow Tracks_{AS} - T_{enc}^k$ 
15:      $Tracks_{DB+AS} \leftarrow Tracks_{DB+AS} \cup CT^k$ 
16:   end if
17: end for
18: return  $Tracks_{AS} \cup Tracks_{DB} \cup Tracks_{DB+AS}$ 

```

Figure 13. P_{seg} and P_{bf} when the data broker and application server are compromised ($S_{min} = 200$).

using the encrypted records, the set of trajectories $Tracks_{DB}$ estimated using only data available in the data broker, and a set of trajectories $Track_{DB+AS}$ estimated using data available in the data broker and the application server. The time complexity for Algorithm 3 is ruled by the Munkres algorithm, used to estimate trajectories using the encrypted points, and it is $O(n^3)$ where n is the number of points to be followed [50].

Algorithm 3 produces a set of estimated trajectories divided as: $Tracks_{AS} = 31.56\%$, $Tracks_{DB} = 39.55\%$ and $Tracks_{DB+AS} = 28.89\%$. The set of $Tracks_{AS}$ has a $P_{seg} = 0.298$ and $P_{bf} = 0.789$. Figure 13 shows the behavior of the P_{bf} and P_{seg} as a function of the *percentile* used in Algorithm 3. It is worth in noticing that Algorithm 3 improves the P_{seg} of estimated trajectories when compared with the case in which only the application server is compromised as well as improve the P_{bf} of estimated trajectories when compared with the case in which the data broker is compromised.

These results confirm the intuitive assumption that compromising both servers results in a better estimation of trajectories for the adversary. However, these results show the robustness of the proposed algorithm because in the worst case scenario, in which the data broker and application server are compromised, the adversary has a lower probability to estimate the actual participants' trajectory when compared with the double encryption scheme.

8.3. Trade-offs

Figure 14 shows the behavior of P_{seg} and P_{bf} when the data broker is compromised by the adversary. First, the lower S_{min} , the higher P_{seg} because more points have been obfuscated using POI and Algorithm 2 is able to discover more points of the actual trajectory. On the other hand, P_{bf} has a stable behavior regardless of the S_{min} value.

Figure 14. Tradeoff P_{bf} and P_{seg} for $S_{min} = \{50, 200, 400\}$ when the data broker is compromised.

Table 4. Comparison for P_{seg} and P_{bf} varying the minimum cell size (S_{min}).

S_{min}	50	200	400
% Encrypted	0%	40.7%	69.7%
% POI	100%	59.3%	30.3%
P_{seg} at AS	0	0.326	0.478
P_{bf} at AS	0	0.783	0.791

In addition, Table 4 presents the trade-offs for P_{seg} and P_{bf} when the application server is compromised by the adversary. The higher S_{min} , the higher P_{seg} because there are more encrypted locations and the GNN algorithm has more points to estimate the actual trajectories. On the other hand, P_{bf} has a similar value for $S_{min} = \{200, 400\}$ meaning a good accuracy for the algorithm. In the case of $S_{min} = 50$, the percentage of encrypted points is 0, therefore the P_{seg} and P_{bf} are equal to 0. However, in this latter case, the system produces a lower quality of information as shown in Figure 9.

Figure 15 presents P_{seg} and P_{bf} as a function of the S_{min} value and the percentile. As shown in the figure, the higher S_{min} , the higher P_{bf} because more actual points are available and a higher accuracy is achieved. On the other hand, the higher S_{min} , the lower P_{seg} because there are more encrypted points and less obfuscated points in the data broker.

In conclusion, the results show the good level of privacy protection provided by A-PIE, our hybrid algorithm, when the historical case is taken into consideration. The S_{min} parameter determines the level of P_{seg} and P_{bf} which are inversely proportional given that the scenario with a high P_{seg} (i.e., the adversary has compromised the data broker) produces a low P_{bf} . On the other hand, the scenario with a high P_{bf} (i.e., the adversary has compromised the application server) produces a low P_{seg} . The worst case is reached in the scenario in which the data broker and the application server have been compromised. In such scenario, a high S_{min} means that the adversary reaches more accuracy to discover the participant's trajectory. However, the average P_{seg} is around 55% of the actual trajectory with an accuracy equal to 65%.

9. Conclusions

This paper presents A-PIE, a new privacy-preserving mechanism designed to improve the quality of the information provided to final users while guaranteeing a good protection level and low-energy consumption. A-PIE combines anonymization, data obfuscation, and encryption techniques to increase the quality of information and privacy protection without increasing the energy consumption in a significant manner. The proposed hybrid algorithm dynamically changes the cell sizes of the grid of the area of interest according to the variability of the variable of interest being measured and chooses different privacy-preserving mechanisms depending on the size of the cell. In small cells,

Figure 15. Tradeoff P_{bf} and P_{seg} for $S_{min} = \{200, 400\}$ when the data broker and application server are compromised.

where users can be identified easier, the algorithm uses encryption techniques to protect the privacy of the users and increase the quality of the information, as the reported location is the real location. On the other hand, anonymization and data obfuscation techniques are used in bigger cells where the variability of the variable of interest is low and therefore it is more important to protect the real location (privacy) of the user.

In addition, this paper presents the evaluation of the privacy protection problem on A-PIE for the snapshot and the historical case. An evaluation model is presented for each case as well as set of experiments when an adversary compromises the data broker, application server or both. The results shows the better performance A-PIE when compared to other algorithms for the snapshot case. In addition, the hybrid algorithm achieves a good protection level when the historical case is taken into consideration because the records division in the data broker and application server decrease the effectiveness of tracking algorithms to estimate the actual participants' trajectories.

In conclusion, the S_{min} parameter governs the performance of the proposed privacy-preserving mechanism in terms of quality of information to the final user, the energy consumption and privacy protection level. The lower S_{min} , the lower the quality of information, the lower the energy consumption and the lower privacy protection in the data broker for both the snapshot and historical case. The higher S_{min} , the higher the quality of information and the higher the privacy protection in the data broker, but the lower privacy protection in the application server. Therefore, the balance in the quality of information, privacy protection and energy consumption is achieved when the S_{min} reaches a medium value.

10. Future Work

Since people tends to follow similar mobility patterns (e.g., during the morning most people travel towards down town), an adversary might use this external information in order to improve the estimation of the participant's trajectories. Usually, data suppression is applied in order to protect the participant's privacy but, as explained before, data suppression decreases the quality of information [51]. Although A-PIE protects participants for the historical case, the research challenge is to protect the participant's privacy against an adversary with this external information.

On the other hand, we would like to integrate A-PIE as core component of an incentive mechanism for participatory sensing. For instance, since the proposed hybrid mechanism divides the area of interest into cells in order to improve the quality of information, then the challenge is how to buy samples in the each of the cells without revealing the actual location of the participant. For bigger cells the participant may reveal the location of the cell where she is located, but she should not reveal her location when located in smaller cells. We invite the reader to review [52, 53, 54, 55], this set of references corresponds to an excellent prior work on reputation and privacy-preserving schemes for PS applications

Acknowledgements

The authors would like to acknowledge the work of Yarelis Ares and Omar Aviles, REU students from Universidad del Turabo, Puerto Rico, for implementing the different privacy mechanisms and running the experiments to determine their energy consumption. Their work was supported by the National Science Foundation under grant No. 1062160, an REU Site on Ubiquitous Sensing.

References

- [1] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, M. B. Srivastava, Participatory sensing, in: World Sensor Web Workshop - ACM Sensys, Boulder, CO, USA, 2006.
- [2] D. Mendez, A Framework For Participatory Sensing Systems, University of South Florida Libraries Catalog, 2012.
- [3] P. Dutta, P. M. Aoki, N. Kumar, A. Mainwaring, C. Myers, W. Willett, A. Woodruff, Common sense: participatory urban sensing using a network of handheld air quality monitors, in: Proceedings of the 7th ACM conference on embedded networked sensor systems, ACM, 2009, pp. 349–350.
- [4] D. Mendez, A. J. Perez, M. A. Labrador, J. J. Marron, P-sense: A participatory sensing system for air pollution monitoring and control, in: Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom) Workshops, Seattle, Washington, USA, 2011, pp. 344–347.

- [5] C. Leonardi, A. Cappellotto, M. Caraviello, B. Lepri, F. Antonelli, Secondnose: an air quality mobile crowdsensing system, in: *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational*, Helsinki, Finland, October 26–30, 2014, 2014, pp. 1051–1054. doi:10.1145/2639189.2670273. URL <http://doi.acm.org/10.1145/2639189.2670273>
- [6] S. Devarakonda, P. Sevusu, H. Liu, R. Liu, L. Iftode, B. Nath, Real-time air quality monitoring through mobile sensing in metropolitan areas, in: *Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing*, ACM, 2013, p. 15.
- [7] A. Kapadia, D. Kotz, N. Triandopoulos, Opportunistic sensing: Security challenges for the new paradigm, in: *Proceedings of the 1st International Workshop on Communication Systems and Networks. COMSNETS*, Bangalore, India, 2009, pp. 1–10.
- [8] D. Christin, Privacy in mobile participatory sensing: current trends and future challenges, *Journal of Systems and Software*.
- [9] A. R. Beresford, F. Stajano, Location privacy in pervasive computing, *IEEE Pervasive computing* (1) (2003) 46–55.
- [10] I. Boutsis, V. Kalogeraki, Privacy preservation for participatory sensing data, in: *Pervasive Computing and Communications (PerCom)*, 2013 IEEE International Conference on, IEEE, 2013, pp. 103–113.
- [11] M. Gruteser, B. Hoh, On the anonymity of periodic location samples, in: *Security in Pervasive Computing*, Vol. 3450 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2005, pp. 179–192.
- [12] P. Goovaerts, *Geostatistics for Natural Resources Evaluation*, 4th Edition, Vol. 4th printing, Oxford University Press, USA, 1997.
- [13] R. Moore, *Geostatistics in hydrology: Kriging interpolation*, Tech. rep., Mathematics Department, Macquarie University, Sydney. (1999).
- [14] D. Mendez, M. A. Labrador, K. Ramachandran, Data interpolation for participatory sensing systems, *Pervasive and Mobile Computing* 9 (2013) 132 – 148.
- [15] C. Bettini, S. Mascetti, X. Sean-Wang, D. Freni, S. Jajodial, Anonymity and historical-anonymity in location-based services, in: *Privacy in Location-Based Applications*, Vol. 5599 of *Lecture Notes in Computer Science*, Springer-Verlag, 2009, pp. 1–30.
- [16] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, N. Triandopoulos., Anonymsense: Privacy-aware people-centric sensing, in: *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services - MobiSys*, Breckenridge, Colorado, USA, 2008, pp. 211–224.
- [17] J. Domingo-Ferrer, Microaggregation for database and location privacy, in: *Next Generation Information Technologies and Systems*, Vol. 4032 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2006, pp. 106–116.
- [18] A. Machanavajjhala, D. Kifer, J. Gehrke, M. Venkitasubramaniam, L-diversity: Privacy beyond k-anonymity, *ACM Transactions on Knowledge Discovery from Data - TKDD* 1 (1).
- [19] W. He, X. Liu, H. Nguyen, K. Nahrstedt, T. T. Abdelzaher, Pda: Privacy-preserving data aggregation in wireless sensor networks, in: *Proceeding of the 26th IEEE International Conference on Computer Communications - INFOCOM*, Anchorage, Alaska, USA, 2007, pp. 2045–2053.
- [20] C. A. Ardagna, M. Cremonini, E. Damiani, S. D. C. Di Vimercati, P. Samarati, Location privacy protection through obfuscation-based techniques, in: *Data and Applications Security XXI*, Springer, 2007, pp. 47–60.
- [21] J. Krumm, A survey of computational location privacy, *Personal and Ubiquitous Computing* 13 (6) (2009) 391–399.
- [22] R. Agrawal, R. Srikant, Privacy-preserving data mining, in: *Proceedings of the 2000 ACM International Conference on Management of Data - SIGMOD*, Vol. 29, Dallas, Texas, USA, 2000, pp. 439–450.
- [23] R. Ganti, N. Pham, Y. Tsai, T. Abdelzaher, Poolview: stream privacy for grassroots participatory sensing, in: *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, Raleigh, North Carolina, USA, 2008, pp. 281–294.
- [24] M. Murshed, T. Sabrina, A. Iqbal, K. M. Alam, A novel anonymization technique to trade off location privacy and data integrity in participatory sensing systems, in: *Proceedings of the 4th International Conference on Network and System Security - NSS*, Melbourne, Australia, 2010, pp. 345–350.
- [25] I. Vergara-Laurens, M. A. Labrador, Preserving privacy while reducing power consumption and information loss in lbs and participatory sensing applications, in: *Proceedings of the IEEE Global Communication Conference - GLOBECOM Workshops*, Houston, Texas, USA, 2011, pp. 1247–1252.
- [26] J. Horey, M. M. Groat, S. Forrest, F. Esponda, Anonymous data collection in sensor networks, in: *Proceeding of the 4th Annual International Conference on Mobile and Ubiquitous Systems: Networking Services - MobiQuitous*, Philadelphia, Pennsylvania, USA, 2007, pp. 1 –8.
- [27] P. Kalnis, G. Ghinita, K. Mouratidis, D. Papadias., Preventing location-based identity inference in anonymous spatial queries, *IEEE Transactions on Knowledge and Data Engineering* 19 (12) (2007) 1719 –1733.
- [28] K. Yang, K. Zhang, J. Ren, X. Shen, Security and privacy in mobile crowdsourcing networks: challenges and opportunities, *Communications Magazine*, IEEE 53 (8) (2015) 75–81.
- [29] A. Kapadia, N. Tri, C. Cornelius, D. Peebles, D. Kotz, Anonymsense: Opportunistic and privacy-preserving context collection, in: *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services - MobiSys*, Breckenridge, Colorado, USA, 2008, pp. 280–297.
- [30] B. Hoh, M. Gruteser, R. Herring, J. Ban, D. Work, J. C. Herrera, A. M. Bayen, M. Annavaram, Q. Jacobson, Virtual trip lines for distributed privacy-preserving traffic monitoring, in: *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services - MobiSys*, Breckenridge, Colorado, USA, 2008, pp. 15–28.
- [31] I. Vergara-Laurens, D. Mendez, M. Labrador, Privacy, quality of information, and energy consumption in participatory sensing systems, in: *Proceeding of the 2014 IEEE International Conference on Pervasive Computing and Communications - PERCOM*, Budapest, Hungary, 2014, pp. 199–207.
- [32] D. Mendez, M. A. Labrador, Density maps: Determining where to sample in participatory sensing systems, in: *Proceedings of the Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing (MUSIC)*, Vancouver, Canada, 2012, pp. 35–40.
- [33] D. Mendez, M. A. Labrador, On sensor data verification for participatory sensing systems, *Journal of Networks* 8 (3).
- [34] D. Mendez, M. A. Labrador, Removing spatial outliers in ps applications, in: *Proceedings of the International Conference on Selected Topics in Mobile and Wireless Networking - iCOST*, Avignon, France, 2012, pp. 66–71.
- [35] C. Bettini, S. Mascetti, X. Sean-Wang, D. Freni, Spatial generalization algorithms for lbs privacy preservation, *Journal of Location Based Services* 2 (1) (2008) 179–207.

- [36] F. Aurenhammer, Voronoi diagrams a survey of a fundamental geometric data structure, *ACM Computing Surveys (CSUR)* 23 (3) (1991) 345–405.
- [37] A. Okabe, B. Boots, K. Sugihara, S. N. Chiu, *Spatial tessellations: concepts and applications of Voronoi diagrams*, Vol. 501, John Wiley & Sons, 2009.
- [38] I. Vergara-Laurens, D. Mendez, M. A. Labrador, On the interactions between privacy-preserving, incentive, and inference mechanisms in participatory sensing systems, in: *Network and System Security*, Vol. 7873 of *Lecture Notes in Computer Science*, Springer, 2013, pp. 614–620.
- [39] S. Hoteit, S. Secci, S. Sobolevsky, G. Pujolle, Ratti, Estimating real human trajectories through mobile phone data, in: *Proceedings of the 14th IEEE International Conference on Mobile Data Management - MDM*, Vol. 2, 2013, pp. 148–153.
- [40] M. Gonzalez, C. Hidalgo, A. Barabasi, Understanding individual human mobility patterns, *Nature* 453 (2008) 779–782.
- [41] H. Wang, F. Calabrese, G. D. Lorenzo, C. Ratti, Transportation mode inference from anonymized and aggregated mobile phone call detail records, in: *Proceedings of the 13th IEEE International Conference on Intelligent Transportation Systems - ITSC*, 2010, pp. 318–323.
- [42] N. Vyahhi, S. Bakiras, P. Kalnis, G. Ghinita, Tracking moving objects in anonymized trajectories, in: S. Bhowmick, J. K?ng, R. Wagner (Eds.), *Database and Expert Systems Applications*, Vol. 5181 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2008, pp. 158–171.
- [43] J. Hightower, G. Borriello, Location systems for ubiquitous computing, *Computer* 34 (8) (2001) 57–66.
- [44] M. E. Nergiz, C. Clifton, Thoughts on k-anonymization, *Data and Knowledge Engineering* 63 (3) (2007) 622 – 645.
- [45] M. E. Nergiz, M. Atzori, Y. Saygin, Towards trajectory anonymization: A generalization-based approach, in: *Proceedings of the 2008 ACM International Workshop on Security and Privacy in GIS and LBS*, Irvine, California, USA, 2008, pp. 52–61.
- [46] K. Shilton, Four billion little brothers?: Privacy, mobile phones, and ubiquitous data collection, *Communications of the ACM* 52 (11) (2009) 48–53.
- [47] S. Hoteit, S. Secci, S. Sobolevsky, C. Ratti, G. Pujolle, Estimating human trajectories and hotspots through mobile phone data, *Computer Networks* 64 (2014) 296–307.
- [48] J. Tinevez, Simple tracker, 2011.
URL <http://www.mathworks.com/matlabcentral/fileexchange/34040-simple-tracker>
- [49] F. Bourgeois, J. Lassalle, An extension of the munkres algorithm for the assignment problem to rectangular matrices, *Communications of the ACM* 14 (12) (1971) 802–804.
- [50] C. H. Papadimitriou, K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Dover Publications, 1998.
- [51] B. N. Schilit, A. Lamarca, G. Borriello, W. G. Griswold, D. McDonald, E. Lazowska, J. Hong, V. Iverson, Challenge: Ubiquitous location-aware computing and the place lab initiative, in: *Proceedings of the 1st ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots - WMASH*, San Diego, California, USA, 2003, pp. 29–35.
- [52] Q. Li, G. Cao, Providing privacy-aware incentives for mobile sensing, in: *Pervasive Computing and Communications (PerCom)*, 2013 IEEE International Conference on, IEEE, 2013, pp. 76–84.
- [53] D. Christin, C. Roßkopf, M. Hollick, L. A. Martucci, S. S. Kanhere, Incognisense: An anonymity-preserving reputation framework for participatory sensing applications, *Pervasive and mobile Computing* 9 (3) (2013) 353–371.
- [54] J. Zhang, J. Ma, W. Wang, Y. Liu, A novel privacy protection scheme for participatory sensing with incentives, in: *Cloud Computing and Intelligent Systems (CCIS)*, 2012 IEEE 2nd International Conference on, Vol. 3, IEEE, 2012, pp. 1017–1021.
- [55] A. Michalas, N. Komninos, The lord of the sense: A privacy preserving reputation system for participatory sensing applications, in: *Computers and Communication (ISCC)*, 2014 IEEE Symposium on, IEEE, 2014, pp. 1–6.