Research Article

A Privacy-Preserving Continuous Location Monitoring System for Location-Based Services

Doohee Song, Jongwon Sim, Kwangjin Park, and Moonbae Song

1Information Communication Engineering, Wonkwang University, Iksan-shi 570-749, Republic of Korea
2Samsung Electronics, 416 Maetan 3-dong, Yeongtong-gu, Suwon, Republic of Korea

Correspondence should be addressed to Kwangjin Park; kwangjin.park@gmail.com and Moonbae Song; mbsong@gmail.com

Received 23 December 2014; Revised 23 March 2015; Accepted 2 April 2015

Academic Editor: Dingyi Fang

Copyright © 2015 Doohee Song et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To protect users' private locations in location-based services, various location anonymization techniques have been proposed. The most commonly used technique is spatial cloaking, which organizes users' exact locations into cloaked regions (CRs). This satisfies the $K$-anonymity requirement; that is, the querier is not distinguishable among $K$ users within the CR. However, the practicality of cloaking techniques is limited due to the lack of privacy-preserving query processing capacity, for example, providing answers to the user's spatial queries based on knowledge of the user's cloaked location rather than the exact location. This paper proposes a cloaking system model called anonymity of motion vectors (AMV) that provides anonymity for spatial queries. The proposed AMV minimizes the CR of a mobile user using motion vectors. In addition, the AMV creates a ranged search area that includes the nearest neighbor (NN) objects to the querier who issued a CR-based query. The effectiveness of the proposed AMV is demonstrated in simulated experiments.

1. Introduction

With advances in mobile communication technology and the widespread use of GPS-enabled mobile devices, location-based services (LBS) are growing rapidly. In LBS, mobile users issue spatial queries to LBS providers to obtain information based on their geographical locations [1–4]. Typical LBS applications include road navigation, area-specific weather forecasts, map information, automotive traffic monitoring, location-based SNS, and nearest point of interest (POI) queries [5–7].

Because LBS is provided for users based on their locations, users must report their location information to an LBS provider. Although LBS provides valuable services to users, revealing one's private locations to potentially untrustworthy LBS providers poses privacy concerns [8, 9]. Knowledge of a person's location can be used by an adversary to physically locate and identify the person. In fact, there are many real-life scenarios in which perpetrators abuse technologies to gain access to private location information about victims [10].

In an effort to preserve the privacy of LBS users, a number of location anonymization techniques have been studied. In particular, spatial cloaking that blurs a user's exact location into the cloaked regions (CRs) such that the CR satisfies the user-specified $K$-anonymity requirement has been widely used [11–23].

Several researchers have presented the concept of a location anonymizer, a trusted third party that acts as a middle layer between the user and the LBS server [6]. The anonymizer blurs the users' exact locations into CRs using existing location anonymization algorithms and sends the CRs, rather than the exact user location, to the LBS server. This mechanism enables the users to obtain the desired LBS without revealing their exact locations to the LBS server. The $K$-anonymity requirement given by the querier ensures that the CR is $K$-anonymous (e.g., the querier is indistinguishable among $K$ users within the CR), thus reducing the probability of the querier's location being exposed to untrustworthy parties to $1/K$. Because the LBS server does not know the querier's exact location, it can only return a set of candidate answers to the CR-based query.

This paper proposes a cloaking system model called anonymity of motion vectors (AMV), which provides anonymity for spatial queries. The proposed AMV can...
address user queries based on the user’s cloaked location information from the location anonymizer. The AMV minimizes the CR by predicting user movements based on the motion vector. The proposed method can be applicable in both outdoor and indoor environments. For example, a client in the car moves along the road and a cloaked region (CR) can be created based on the client’s moving speeds and directions. In indoor environments, the movement range of a client is generally restricted due to various obstacles like walls and doors. This might make the motion vectors produced by the proposed AMV invalid. Nevertheless, the AMV can achieve effective query processing by adjusting the update intervals of the vector according to the size of the indoor space (i.e., the movement range of the client). The main features of the AMV can be summarized as follows:

1. Client movements are considered when generating a CR that satisfies the user-specified $K$-anonymity level.
2. The user location update cost of the anonymizer can be reduced by allowing the anonymizer to not check the user location from time $t$ to $t + n$.
3. Compared with the previous cloaking model (distributed grid based continuous cloaking (DGCC) (refer to Figure 1) [23], minimum cycle region (MCR) [24], which produces circular CRs (refer to Figure 2)), the AMV generates a smaller CR, thus lowering the query processing time.

The major notations used throughout this paper are as shown in Notations.

The rest of this paper is organized as follows. Section 2 highlights related works. Section 3 describes the proposed AMV model and depicts the creation of CRs using the previous MCR and proposed AMV. Section 4 describes the creation of $\mathcal{O}_{ER}$ for CR-based NN queries. Section 5 presents the equations of the AMV. The experimental results are given in Section 6, and Section 7 concludes the paper.

2. Related Work

To address the location privacy protection of LBS users, many location anonymization techniques have been proposed. Cloaking and $K$-anonymity are the most commonly used forms of location anonymization [10–22]. Figure 1 depicts an example of the distributed grid based continuous cloaking (DGCC) area creation when the required $K$-anonymity level is 4 (e.g., $K = 4$) [23]. The DGCC method provides the privacy protection of the query issuer by maintaining $K$-anonymous clients adjacent to the querier. In Figure 1(a), the CR is represented by a minimum bounding circle and contains four mobile clients ($q, C_2, C_3, C_5$) at time $t$. Figure 1(b) shows the 4-anonymous CR at $t + 1$. Note that some of the clients have moved and that the CR is enlarged to enclose the original four clients. In Figure 1(a), the radius of the CR is $2\sqrt{2}$, and the radius of the CR in Figure 1(b) is $3\sqrt{2}$. To avoid such an increase in the CR size, a cloaking algorithm that forms a CR with the $K − 1$ clients that are currently most adjacent to the querier has been proposed. However, the weakness of this approach is that an adversary can guess the actual querier (denoted here by $q$), who remains constant over time, while only $K − 1$ clients in the CR are continuously being updated (e.g., $q, C_2, C_3, C_5 → q, C_1, C_4, C_6$) [23].

The cloaking method in [12] considers both $K$-anonymity and $L$-diversity. It creates a minimum CR by first finding $L$ number of buildings. After satisfying the $L$-diversity requirement, it extends the CR to satisfy the $K$-anonymity requirement (e.g., the CR contains at least $K$ number of users). The New Casper proposed by Mokbel et al. [11] is a research prototype of the framework for anonymous LBS. It satisfies the four requirements of location anonymity: accuracy, quality, efficiency, and flexibility. The cloaking algorithm in [13] employs dummy generation in creating a CR.
To solve the building double counting problem, it adds the building grouping item to the index structure of the existing privacy grid approach and minimizes the CR by finding $K$ users (e.g., $K$-anonymity) in the grid cells adjacent to the buildings in the CR. However, these previous schemes ignore the client’s movements.

Location anonymization techniques for LBS that consider the movement information of mobile users have been suggested [24, 25]. The technique in [24] addresses the anonymization of snapshot and continuous LBS queries. It creates a minimum circular CR based on mobile user’s location trajectories. However, this requires tolerance of temporal errors, as the direction and speed of user movement are not taken into account. The technique in [25] considers the user’s speed and direction of movement in addition to location. It requires information about the exact location of all user nodes and calculates the querier’s movements based on his/her direction of movement and speed. There are some additional anonymization techniques that consider client movements using the Voronoi diagram, road networks, and historical location data [26–29]. However, none of these existing techniques support spatial queries with cloaked areas (e.g., CRs). Recently, a method for private $k$NN queries has been proposed. It creates a CR for the querier and performs query processing based on the type of objects (e.g., restaurants and cafes) in the CR [30]. This method does not consider $K$-anonymity while taking into account object types, so it cannot be directly compared with the proposed AMV.

### 3. Proposed AMV Model

The AMV proposed in this paper considers client movements to create a minimized $K$-anonymous CR and supports spatial queries with cloaked locations. The system model of the proposed AMV contains three components: the centralized LBS server, the location anonymizer, and the client. The location anonymizer is a trusted third party that is placed between the client and the LBS server [6]. To obtain LBS, the querier sends its location along with the spatial query to the location anonymizer. The location anonymizer that periodically receives location updates from mobile users blurs users’ exact locations into CRs and sends the query, along with the CR, to the LBS server. The anonymizer creates a session ID that is valid during the entire service period for the querier. The LBS server then processes the CR-based query from the anonymizer and returns a candidate list of answers to the anonymizer. Finally, the anonymizer computes the exact query answer from the candidate list and sends it to the querier.

The proposed AMV assumes that the $K-1$ nearest clients that are most adjacent to the querier are bounded in a CR. This condition is assumed to lower the server overhead, which increases as the CR size increases.

Figure 2 shows the CR produced by the previous MCR method that considers only the user’s speed [24]. The dotted rectangle represents a minimum CR that satisfies the 4-anonymity requirement at $t+1$. The solid rectangle is a modified version of the CR that encompasses all of the grid cells that are fully or partially included in the initial CR (i.e., the grid cells that are intersected by the initial CR are also enclosed). The CR of the solid rectangle is confirmed as a final CR. Figure 1 shows the CR at $t$, and the area of this CR is 25.1. The CR at $t + 1$ shown in Figure 2 has a larger area of 88. In Figure 2, the movement radius of each client is represented by a circle because only the speed, not the direction, is known.

Figure 3 presents the CR created by the proposed AMV, which considers the speed and direction of movement using the motion vector. It is assumed that these clients are moving at the same speed as the clients in Figure 2. The 4-anonymous CR at $t+1$ shown in Figure 3 has an area of 42. In other words, the AMV produces a smaller CR than the MCR.

Compared to the existing technologies, the proposed AMV involves the extra cost of maintaining the speed and direction information of the moving users. However, the user
Algorithm 1: 1st $O_{ER}$ creation (filtering).

4. Finding $O_{ER}$ in NN Queries

This section illustrates how to create $O_{ER}$ for a CR-based NN query.

In Figure 4, it is assumed that there are no objects inside the CR. The CR of the AMV described in Section 3 is used to find $O_{ER}$. The solid rectangle in Figure 4 represents the CR of a mobile user who issued a query, and the dotted rectangle denotes a search area where the nearest object to the CR is found. To create an $O_{ER}$, one must first check the CR and then spread out to the neighboring grid cells around the CR one by one, until the nearest neighbor object $O_N$ is found. Here, the nearest object found is $O_3$. Then, the top-left vertex of the CR (denoted by “A”) that is farthest from $O_3$ is connected to $O_3$, which creates a line segment. Vertex $A$ becomes the center of a circle whose radius is the distance between $A$ and $O_3$; for example, $dist(CR, A) = \maxdist$. The grid cells within this circle and those intersected by this circle become $O_{EN}$ (the shaded area in Figure 4). If there is more than one object nearest to the CR, for example, $dist(CR, O_i) = dist(CR, O_{i+1})$, $O_i$ is prioritized over $O_{i+1}$.

Algorithm 1 describes the primary identification (filtering) of $O_{ER}$ depicted in Figure 4.

The initial $O_{ER}$ returned by Algorithm 1 is insufficient to answer the user’s exact NN query. In Figure 4, one can see there is an object other than $O_3$ that is nearer to the querier; for example, $dist(q, O_4) > dist(q, O_3)$. Algorithm 1 provides an inaccurate answer, in which the actual nearest object to the querier is $O_4$. To resolve this problem, the secondary (refined) $O_{ER}$ creation is suggested.

Figure 5 depicts the secondary creation of $O_{ER}$. In addition to the $O_{ER}$ found in the primary creation, other regions

can provide this motion information when submitting the location and $K$-anonymity parameter information to the LBS server. Thus, the overhead associated with this additional motion information is not significant. Assume that 1 byte of storage is required for the representation of client speed in the AMV and another 1 byte for the representation of client direction. As $x$- and $y$-axes exist in the coordinate information of object, it becomes $2 \times \text{coordinates (GPS coordinate information generally uses } x\text{-axis } 4\text{ bytes and } y\text{-axis } 4\text{ bytes; in case of detailed GPS measurement, the data size on } x\text{- and } y\text{-axes may increase). The size of object is typically assumed to be from } 128\text{ bytes to } 1,024\text{ bytes}\ [31].$ When the number of clients is 100, the size of data stored in the server is 1,000 bytes in the AMV, 900 bytes in the MCR, and 800 bytes in the DGCC. The DGCC yields the smallest data size, but it has a drawback that its data size increases by $t$ times whenever an update is made. Note that the proposed work is not intended to reduce the data size maintained in the clients. This work aims to minimize the data to be delivered by the server by discovering $O_{ER}$ and $O_{EN}$ in response to the client query. In other words, the proposed method intends to address the problems caused by the growth of CR sizes. For example, the CR of the MCR is larger than the CR of the AMV, which can lead to increased $O_{ER}$ and $O_{EN}$ (see Sections 3 and 4). As $O_{EN}$ of the MCR is greater than $O_{EN}$ of the AMV, the MCR gets heavier server overheads than the AMV. The client’s mobility pattern follows the random waypoint mobility model [32], which is widely used.
Algorithm 2: 2nd \( O_{ER} \) creation (refinement).

Algorithm 2

**input:** \( O_{ER} \) from the first creation, \( O_N \), four vertices of the CR (A, B, C, D)

**output:** refined \( O_{ER} \)

01: check the vertices of the CR;
02: exclude mindist and maxdist;
03: create the circles of vertices B and C using the first \( O_{ER} \) creation algorithm (e.g., create circles centering on B and C, respectively, with minmaxdist\(_B\) and minmaxdist\(_C\) as their radius)
04: Result = (the circle of B \( \cup \) the circle of C);
05: Result = Result \( \cup \) grid;
06: Result = Result \( \cup \) result of Algorithm 1;
07: return Result;

5. AMV Equations

This section defines the proposed AMV and the previous MCR using equations. \( \text{dist}(x, y) \) denotes the distance between \( x \) and \( y \). \( \vec{x} \) denotes the vector of \( x \). Equations (1) and (2) express how to calculate the CR of a moving user in the MCR and AMV, respectively. The DGCC does not consider (estimated) client movements so it does not create a CR of the user who issued a query. In the DGCC, the location of the querier is represented with coordinates. However, the DGCC provides \( K \)-anonymity, so the coordinates of the querier are protected. \( (X, x) \) and \( (Y, y) \) denote coordinates. The associated conditions are \( X \geq x \) and \( Y \geq y \):

\[
\text{MCR's CR} = \sqrt{\text{dist}(x_{\text{min}}, x_{\text{max}})^2 + \text{dist}(y_{\text{min}}, y_{\text{max}})^2} \times \pi,
\]

\[
\text{DGCC's CR} = (x, y),
\]

\[
\text{AMV's CR} = \vec{x} \times \vec{y}.
\]
The computation of the CR that satisfies the $K$-anonymity requirement in the AMV is represented in

$$K\text{-AMV} = \text{dist}(x_{\text{min}}, x_{\text{max}}) \times \text{dist}(y_{\text{min}}, y_{\text{max}}),$$

where $x_{\text{min}} \geq x_{2}, x_{\text{min}}$ denotes the smallest value in the $x$-axis of vector $x_{2}$ and $x_{\text{max}}$ denotes the largest value in the $x$-axis of $x_{1}$. In the $y$-axis, the same conditions as those in the $x$-axis are applied.

In comparison with the AMV, the circular CR of the MCR is transformed into a quadrate CR. The CR represented by a square is 1.17 times larger than the circular CR. If there are grid cells that intersect the CR, the CR is extended to enclose them. This decreases the error rate involved in CR-based queries. The equation below computes the CR of the MCR that satisfies the $K$-anonymity requirement:

$$K\text{-MCR} = \text{dist}(X_{\text{min}}^\text{KC}, X_{\text{max}}^\text{KC}) \times \text{dist}(Y_{\text{min}}^\text{KC}, Y_{\text{max}}^\text{KC}).$$

Similar to $K$-AMV, $X_{\text{min}}^\text{KC}$ denotes the smallest value along the $x$-axis of $X_{2}^\text{KC}$, whereas $X_{1} \geq X_{2}^\text{KC}$. $X_{\text{max}}^\text{KC}$ denotes the largest value along the $x$-axis of $X_{1}^\text{KC}$. These $x$-axis conditions are applied to the $y$-axis as well.

To represent the $K$-DGCC of the DGCC into an equation, the following definitions are made. When $C = \{C_{1}, C_{2}, \ldots, C_{k}\}$, $K$-anonymous clients satisfying the DGCC requirements are denoted as $KC = \{C_1, C_2, \ldots, C_K\}$:

$$K\text{-DGCC} = \text{dist}(X_{\text{min}}^\text{KC}, X_{\text{max}}^\text{KC}) \times \text{dist}(Y_{\text{min}}^\text{KC}, Y_{\text{max}}^\text{KC})$$

where $X_{\text{min}}^\text{KC}$ and $X_{\text{max}}^\text{KC}$ of $K$-DGCC denotes the clients that are located close to the querier, whereas $X_{\text{min}}^\text{KC}$ denotes those far from the querier. Only the clients that satisfy $KC > \{X_{\text{min}}^\text{KC}, X_{\text{max}}^\text{KC}, Y_{\text{min}}^\text{KC}, Y_{\text{max}}^\text{KC}\}$ can be included in the CR of $K$-DGCC.

The case of having objects inside the CR ($O_N \in \text{CR}$) and the case of having no object inside the CR ($O_N \notin \text{CR}$) can be distinguished by developing the equations associated with CR-based NN query processing. Among the four vertices of the $K$-anonymous CR (e.g., $A$, $B$, $C$, and $D$), maxdist denotes the vertex farthest from $O_N$, and mindist denotes the shortest CR vertex from $O_N$. The distance from $O_N$ to the other two vertices is denoted by minmaxdist. Of these vertices, minmaxdist $L$ denotes the vertex farther from $O_N$, and minmaxdist $S$ denotes the other one. When $\text{dist}(O_N, A) > \text{dist}(O_N, B) > \text{dist}(O_N, C) > \text{dist}(O_N, D)$, the distance of each of the CR vertices to $O_N$ is expressed using the equations below:

$$\text{maxdist} = \text{dist}(O_N, A),$$
$$\text{mindist} = \text{dist}(O_N, D),$$
$$\text{minmaxdist}_L = \text{dist}(O_N, B),$$
$$\text{minmaxdist}_S = \text{dist}(O_N, C).$$

With the defined distances, the search area for the NN object that answers the CR-based query can be defined as follows:

$$\text{Search area of maxdist} = (\text{maxdist})^2 \times \pi.$$  

The search area of mindist is included in that of maxdist, so its calculation can be omitted.

The search areas of minmaxdist $L$ and minmaxdist $S$ should also be created:

$$\text{Search area of minmaxdist} = (\text{minmaxdist}_L)^2 \times \pi \cup (\text{minmaxdist}_S)^2 \times \pi.$$  

The final search area $O_{NN}$ for the NN query includes the search areas of (8) and (9): all of the grid cells enclosed by the computed search areas and those intersected by the search areas are included in the final search area.

In $K$NN queries, $k$ nearest objects satisfying the query can be both inside and outside the CR. In this case, the final search area for both the inside objects and the outside objects is calculated individually using the equations presented in this section. The results are then combined, for example, inside CR(8) ∪ (9) ∪ outside CR(8) ∪ (9).

### 6. Experimental Evaluation

Through simulated experiments, the performances of the proposed AMV and the previous MCR were measured for comparison. These experiments were performed on a computer with a 2.9 GHz processor and 4 GB memory. C++ was used to conduct the experiments. It was assumed that the querier was stationary and that $K = 1$ clients were moving. The distance of a single grid cell was set to 10 m. The MCR to which the AMV was compared created a minimum CR that satisfies the specified $K$-anonymity requirement (e.g., the stationary querier and $K = 1$ clients are bounded in a CR). The experiment results are the average of the values obtained by executing the experiments 100,000 times.

Table 1 summarizes the parameter settings of the experimental environment.

**Table 1: Experimental environment settings.**

| Parameters | Setting values |
|------------|----------------|
| Number of grids | 1000 × 1000 |
| Number of clients | 50,000, 70,000, (100,000), and 200,000 |
| Number of objects | 30,000, (50,000), 70,000, and 100,000 |
| Client’s movement radius (e.g., number of grids) | (10), 30, 50, and 100 |
| $K$-anonymity ($K$) | 5, (10), 20, and 30 |
| Update time (sec; $t$) | (3), 5, 7, and 10 |
| $k$NN ($k$) | 3, 5, (10), and 20 |

In the experiments, the movement radius of the client was set to 10 grid cells and the update interval of the anonymizer
Figure 7: CR sizes with respect to $K$-anonymity levels.

Figure 8: The size of CRs with respect to the client’s movement radius.

was 3 seconds. CR of AMV generates 30.9% and 55.6% less CR for each of the MCR and the DGCC. $X_{\text{min}}$ ($X_{\text{max}}$) of $K$-DGCC denotes the clients that are located close to the querier, whereas $X_{\text{max}}$ denotes those far from the querier. Only the clients that satisfy $KC \ni \{X_{\text{min}}, X_{\text{max}}, Y_{\text{min}}, Y_{\text{max}}\}$ can be included in the CR of $K$-DGCC. Therefore $K$-DGCC’s slope is increased significantly when $K$ was increased in the graph. Though the MCR considers only the speed of user movements, the AMV considers both the direction and speed using the motion vector, thus producing a smaller CR.

Figure 8 shows the size of CRs with respect to the movement radius of the client. In the experiments, the $K$-anonymity level was set to 10 ($K = 10$), and the anonymizer update interval was 3 and 5 seconds. On average, CR of AMV creates 49.2% and 67.6% less CR for each of the MCR and the DGCC. In Figure 8, for AMV, DGCC, and MCR, the CR size increases as the client’s movement radius increases, but the rate of increase of the AMV is less steep than that of the MCR and DGCC. When $t = 5$, the DGCC has the largest CR. The reason is that the distance of the querier from the clients satisfying $KC$ increases as the movement radius of client and update time increase.

Figure 9 depicts the size of CRs with regard to the location anonymizer’s update interval, denoted by $t$. The location of the moving client at $t$ is predicted based on the speed and direction of the client’s motion vectors, and the CR is created by considering the predicted client location. In the experiments, the $K$-anonymity level was 10, and the client’s movement radius was 10 and 30 grid cells. On average, CR of AMV makes 42.7% and 65.2% less CR for each of the MCR and the DGCC.

Figures 10 through 14 examine the search area $O_{\text{ER}}$ for the NN objects satisfying the CR-based query and the number of objects $O_{\text{EN}}$ in $O_{\text{ER}}$. In the experiments, the number of clients was set to 100,000, and the number of objects was 50,000. The $k$ of the $k$NN query was 10, and the anonymizer update interval was 3 seconds. The client’s movement radius was 10 grid cells. The experiment results of a 10NN query (e.g., $k = 10$) are presented in Figures 10 through 13, and Figure 14 shows the result of a $k$NN query (e.g., $k$ is a variable).

Figure 10 shows $O_{\text{ER}}$ for the NN objects that satisfy the CR-based 10NN query. The size of $O_{\text{ER}}$ decreases as the number of objects increases, which occurs at the following intervals: 30,000, 50,000, 70,000, and finally 100,000. This occurs because the objects satisfying the 10NN query are located nearer to the CR when the objects are more densely populated. $O_{\text{ER}}$ of AMV generates 46.7% and 53.7% less $O_{\text{ER}}$ for each of the MCR and the DGCC.

Figure 11 presents $O_{\text{EN}}$ of the 10NN query that varies according to $K$-anonymity levels. $O_{\text{EN}}$ of AMV creates 24.2%
and 41.3% less $O_{EN}$ for each of the MCR and the DGCC. As demonstrated above, the AMV yields a smaller CR than the MCR and DGCC, which decreases the size of $O_{ER}$. Naturally, the number of objects enclosed in a smaller $O_{ER}$ is smaller.

Figure 12 shows $O_{EN}$ of the 10NN query with respect to the number of clients, varying from 50,000 to 200,000. $O_{EN}$ decreases as the number of clients increases because the CR size decreases when the clients are more densely populated. A smaller CR leads to a smaller $O_{ER}$, which, in turn, decreases $O_{EN}$. On average, $O_{EN}$ of AMV creates 42.0% and 51.7% less $O_{EN}$ for each of the MCR and the DGCC.

Figure 13 shows $O_{EN}$ in connection with the client’s movement radius. $O_{EN}$ of AMV generates 55.4% and 71.6% less $O_{EN}$ for each of the MCR and the DGCC. The gap between the AMV, DGCC, and MCR graphs grows as the client’s movement radius increases. Compared with the AMV, the MCR’s and DGCC’s CR size increases more rapidly as the client’s movement radius increases, and a larger CR leads to a larger $O_{ER}$ (and a larger $O_{EN}$).

Figure 14 presents $O_{EN}$ of the CR-based $k$NN query with respect to varying $k$, from 3 to 5 to 10 to 20. The MCR graph increases more rapidly than the AMV graph. When $k = 3$, there is a high likelihood that the requested object can be found inside the CR, thus minimizing $O_{ER}$. When $k = 20$, the objects outside the CR need to be searched, which increases both $O_{ER}$ and $O_{EN}$. On average, $O_{EN}$ of AMV makes 45.8% and 62.1% less $O_{EN}$ for each of the MCR and the DGCC.
7. Conclusion

This paper presents a cloaking system model that enables mobile users to make spatial queries without revealing their exact location information. The proposed AMV generates a minimum $K$-anonymous CR by predicting the future location of the moving client based on the motion vector. This enables the AMV to generate a smaller CR than the previous cloaking method. In addition, the AMV creates a search area $O_{ER}$ for the nearest neighbor objects that satisfies the NN query with the $K$-anonymized region. The $O_{ER}$ creation algorithms are performed in two phases: filtering that finds the initial $O_{ER}$ and refinement that supplements the initial $O_{ER}$. By minimizing $O_{ER}$ for the CR-based NN query, the AMV allows the location anonymizer to return the exact query answers (e.g., the NN objects) to the querier.

In the future, the performance of the AMV can be further examined through experiments in which the querier is moving rather than remaining stationary.

**Notations**

- **CR:** A cloaked region
- **$O_N$:** The nearest object
- **$O_{ER}$:** A range search area for the nearest objects satisfying the spatial query
- **$O_{EN}$:** The number of objects in $O_{ER}$
- **$\text{dist}(a, b)$:** Distance between $a$ and $b$
- **MBR:** A minimum bounding rectangle
- **$C_i$:** $i$th client
- **$O_i$:** $i$th object
- **$K$-anonymity:** $K$ neighboring users to the querier
- **$k\text{NN}$:** $k$ nearest objects to the querier.

**Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

**Authors’ Contribution**

Kwangjin Park and Moonbae Song are equal contributors to this paper.

**Acknowledgments**

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2013R1A1A004593 and 2014R1A1A2054315).

**References**

[1] K. Park, “Efficient data access for location-dependent spatial queries,” Journal of Computer Science and Technology, vol. 29, no. 3, pp. 449–469, 2014.

[2] D. Song and K. Park, “An efficient adaptive bitmap-based selective tuning scheme for spatial queries in broadcast environments,” KSII Transactions on Internet and Information Systems, vol. 5, no. 10, pp. 1862–1878, 2011.

[3] K. Park and P. Valduriez, “A hierarchical grid index (HGI), spatial queries in wireless data broadcasting,” Distributed and Parallel Databases, vol. 31, no. 3, pp. 413–446, 2013.

[4] K. Park, “Location-based grid-index for spatial query processing,” Expert Systems with Applications, vol. 41, no. 4, pp. 1294–1300, 2014.

[5] C.-G. Park, J.-J. Kim, and K.-J. Han, “Efficient POI search for location-based services in road networks,” Journal of Computing Science and Engineering, vol. 40, no. 2, pp. 141–151, 2013.

[6] H. Sim, “LBS description element and research trend,” Journal of Information Processing Systems, vol. 20, no. 6, pp. 3–12, 2013.

[7] Y. Sun, T. F. La Porta, and P. Kermani, “A flexible privacy-enhanced location-based services system framework and practice,” IEEE Transactions on Mobile Computing, vol. 8, no. 3, pp. 304–321, 2009.

[8] J. Voelcker, “Stalked by satellite—an alarming rise in GPS-enabled harassment,” IEEE Spectrum, vol. 43, no. 7, pp. 15–16, 2006.

[9] J. Wrrior, E. McHenry, and K. McGree, “They know where you are,” IEEE Spectrum, vol. 40, no. 7, pp. 20–25, 2003.

[10] W. R. Muntz, T. Barclay, J. Dozier et al., IT Roadmap to a Geospatial Future, The National Academics Press, Washington, DC, USA, 2003.

[11] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, “The new Casper: query processing for location services without compromising privacy,” in Proceedings of the 32nd International Conference on Very Large Data Bases, pp. 763–774, September 2006.

[12] J. Kim, A.-R. Lee, Y.-K. Kim, J.-H. Um, and J.-W. Chang, “Cloaking method supporting K-anonymity and L-diversity for privacy protection in location-based services,” Journal of Korea Spatial Information Society, vol. 10, no. 4, pp. 1–10, 2008.

[13] J.-Y. Kim, E.-H. Jeong, and B.-K. Lee, “A design of cloaking region using dummy for privacy information protection on location-based services,” The Journal of Korea Information and Communications Society, vol. 36, no. 8, pp. 929–938, 2011.

[14] S. Park, J. Bai, and S. Park, “An anonymization technique of continuous query and query log for privacy in location-based services,” Journal of Computing Science and Engineering, vol. 38, no. 2, pp. 65–71, 2011.

**Figure 14:** $O_{EN}$ with respect to the $k$ value of the $k\text{NN}$ query.
[15] J. Lee, “Hierarchical clustering-based cloaking algorithm for location-based services,” *Journal of Korea Information and Communications Society*, vol. 8, no. 8, pp. 1155–1160, 2013.

[16] M. Gruteser and D. Grunwald, “Anonymous usage of location-based services through spatial and temporal cloaking,” in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, pp. 31–42, ACM, San Francisco, Calif., USA, May 2003.

[17] B. Gedik and L. Liu, “A customizable k-anonymity model for protecting location privacy,” in *Proceedings of the IEEE International Conference on Distributed Computing Systems*, pp. 620–629, June 2005.

[18] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, “Preventing location-based identity inference in anonymous spatial queries,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 12, pp. 1719–1733, 2007.

[19] B. Gedik and L. Liu, “Protecting location privacy with personalized k-anonymity: architecture and algorithms,” *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, 2008.

[20] C. Y. Chow, M. F. Mokbel, J. Naps, and S. Nath, “Approximate evaluation of range nearest neighbor queries with quality guarantee,” in *Advances in Spatial and Temporal Databases*, vol. 5644 of *Lecture Notes in Computer Science*, pp. 283–301, Springer, Berlin, Germany, 2009.

[21] L. Yao, C. Lin, X. Kong, F. Xia, and G. Wu, “A clustering-based location privacy protection scheme for pervasive computing,” in *Proceedings of the IEEE/ACM International Conference on Green Computing and Communications & International Conference on Cyber, Physical and Social Computing (GreenCom-CPSCom’10)*, pp. 719–726, IEEE, Hangzhou, China, December 2010.

[22] S. Park and S. J. Jeong, “An anonymization scheme protecting querist identification in location-based service with user profile,” *Journal of Computing Science and Engineering*, vol. 38, no. 4, pp. 201–207, 2011.

[23] H. Kim, Y. Kim, and J. Chang, “A grid-based cloaking area creation scheme for continuous LBS queries in distributed systems,” *Journal of Convergence*, vol. 4, no. 1, pp. 23–30, 2013.

[24] T. Xu and Y. Cai, “Location anonymity in continuous location-based services,” in *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, pp. 1–8, November 2007.

[25] Y. Wang, L. Wang, and B. C. M. Fung, “Preserving privacy for location-based services with continuous queries,” in *Proceedings of the IEEE International Conference on Communications (ICC ’09)*, pp. 1–5, IEEE, Dresden, Germany, June 2009.

[26] K. Vu, R. Zheng, and J. Gao, “Efficient algorithms for k-anonymous location privacy in participatory sensing,” in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM ’12)*, pp. 2399–2407, March 2012.

[27] B. Palanisamy and L. Liu, “MobiMix: protecting location privacy with mix-zones over road networks,” in *Proceedings of the IEEE 27th International Conference on Data Engineering (ICDE ’11)*, pp. 494–505, April 2011.

[28] T. Xu and Y. Cai, “Exploring historical location data for anonymity preservation in location-based services,” in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM ’08)*, pp. 547–555, IEEE, April 2008.

[29] C. Ardagna, G. Livraga, and P. Samarati, “Protecting privacy of user information in continuous location-based services,” in *Proceedings of the IEEE 15th International Conference on Computational Science and Engineering (CSE ’12)*, pp. 162–169, Nicosia, Cyprus, December 2012.

[30] X. Yi, R. Paulet, E. Bertino, and V. Varadharajan, “Practical k nearest neighbor queries with location privacy,” in *Proceedings of the 30th IEEE International Conference on Data Engineering (ICDE ’14)*, pp. 640–651, IEEE, Chicago, Ill, USA, April 2014.

[31] B. Zheng, W.-C. Lee, K. C. K. Lee, D. L. Lee, and M. Shao, “A distributed spatial index for error-prone wireless data broadcast,” *VLDB Journal*, vol. 18, no. 4, pp. 959–986, 2009.

[32] T. Camp, J. Boleng, and V. Davies, “A survey of mobility models for ad hoc network research,” *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.
