An Enhanced Location Scattering Based Privacy Protection Scheme

NISHA NISHA, IYNKARAN NATGUNANATHAN, AND YONG XIANG, (Senior Member, IEEE)
School of Information Technology, Deakin University, Waurn Ponds, VIC 3216, Australia
Corresponding author: Nisha Nisha (nnisha@deakin.edu.au)

ABSTRACT Consumer privacy-preserving is a major concern in location-based services (LBSs). The paramount problem in LBSs is the disclosure of a user’s actual location while interacting with the location-based service provider (LSP). To address this issue, some privacy-preserving mechanisms introduce a trusted middle entity (TME) between a user and the LSP. However, a TME could be compromised, thus posing a serious privacy threat to user’s information. In this paper, we propose a novel dummy location scattering scheme (DLSS) to protect the location privacy of a user. Specifically, DLSS employs a dummy location generation technique to reduce the risk of location information exposure to untrusted entities. In addition, a pseudonym-based mechanism and a time delay technique are adopted to further improve the privacy of a user. We conducted extensive experiments with randomly generated users’ location to evaluate the robustness of our proposed scheme. When the number of users are increased to 300, the computation time of the proposed scheme is below 105 ms. Even for the larger number of points of interest (POIs) (>2000), the computation time is below 1500 ms in comparison with other existing schemes. Simulation results show that the proposed DLSS preserves privacy at low computation time and communication cost in comparison with the existing schemes.

INDEX TERMS Location privacy, adjusted location, minimum bounding area, scattered dummy locations.

I. INTRODUCTION

With the development of sensor technology, mobile devices, and wireless communication, location-based services (LBSs) have been gaining wide acceptance in recent years. The versatile nature of LBSs provides a variety of services to mobile users in several sectors including entertainment, health, work, and personal life. LBSs can be classified as continuous LBSs and snapshot LBSs. A user looking for nearby banks or restaurants are snapshot LBSs while tracking under any global positioning satellite (GPS) based applications are continuous LBSs. In LBSs, a mobile user prepares a request message that contains his/her geographical location along with query information and sends it to the location-based service provider (LSP). Based on the real-time geographical location, LSP provides the accurate point of interests (POIs) such as the nearest bank, restaurant, cinema, and hospital, to the mobile user [1], [2].

Although LBSs provide convenience and safety but they threaten the privacy of the user. If a user reports the exact location to the LSP, it can extract all the information about a user such as his/her personal habits, political affiliations, visits to bars, churches, banks, and even social relationships [3]. In this way, by collecting all the information, LSP can build the movement profile of a user and discloses the user’s sensitive information to the third party for advantage. This situation is even more worse in the case of continuous LBSs. Therefore, it is necessary to pay more attention to the location privacy of the user while using continuous LBSs. Generally, location privacy is the inability to guess people’s current or past locations. In other words, we can say that if location privacy is protected no smart attacker should be able to relate location information with the corresponding people. To address the location privacy concerns, several approaches have been developed in the literature to mitigate the risk of privacy loss in LBSs. They can be broadly categorized as a trusted middle entity (TME) based approaches or trusted middle entity free (TME-free) approaches.

TME-free approaches do not require a user to trust any entity and require no participation of middle entity between a user and LSP [4], [5]. To protect user’s sensitive information from untrustworthy LSP, a querying user adds some noise to his/her actual location information. These approaches provide privacy to a querying user at the expense of accuracy.

In TME-based approaches, a querying user requires to reveal his/her location-related information to a middle entity...
to get the services. The middle entity acts as an intermediate and is responsible for sending user queries to the untrusted-worthy LSP. Following Fig. 1 illustrates the architecture of TME-based approaches. Before sending the request to the LSP, a user reports the query-related information to the middle entity (sometimes known as anonymizer). This anonymizer provides privacy to a user by hiding the user’s exact location in the cloaking region that contains k users. The generalization of the cloaking region can be based on the user’s actual location [6], [7]. Given a cloaking region, the LSP process the query and returns a set of candidate POIs to the middle entity. However, the LSP cannot be able to identify any user with a probability greater than \(1/K\). The middle entity filters the results and returns the accurate POIs that are of the user’s interest. Compared to TME-free approaches, the TME-based approaches can prevent the untrusted LSP from knowing an actual location of the user. However, this middle entity has the knowledge of all user locations. Being a central point of a user information, it is always an attractive target for the adversaries to infer the user information. Therefore, it is very challenging to have an entity that can be completely trusted.

In this paper, we propose an enhanced dummy location scattering scheme (DLSS) for the LBS environment that protects the privacy of a querying user. The proposed scheme introduces a dummy location generation, a pseudonym mechanism, and combines with a time-delay technique to improve the user’s privacy. Before sending a query request, a user must first select an adjusted (or dummy) location and generates a predefined larger area in which a user’s actual location is encapsulated. A user then divides the predefined larger area into smaller sub-query areas in which each sub-query area has a unique dummy location, a pseudonym, and a user-defined query range. At a given time, some dummy locations are selected based on the proposed algorithm, and those selected locations are assigned in a group. The queries related to these groups of locations are requested with a time delay. Therefore, it is challenging for adversaries to get sensitive information about a querying user. The primary contributions of the paper can be summarized as follows:

- A novel framework is proposed for protecting the location privacy of a querying user by generating an adjusted location inside the query area. For additional privacy, a user’s adjusted location is encapsulated in his/her own generated larger area. We employed the scattering process that allows a user to divide the user-generated larger area into several sub-areas. With the help of these sub-areas, a user prepared a pool of sub-requests to be served by the LSP. The scattering of these sub-areas provides no information about a querying user to any entity.
- A time-delay requesting technique is adopted to let the querying user send the sub-requests to the LSP with a time delay. The LSP receives these sub-requests from different locations after a certain period of time. The goal is to confuse the LSP by sending several queries with a different query range and prevents from knowing the actual queried user.
- A pseudonym mechanism is used to prevent the tracking and identification of the querying user. A user always changes his/her pseudonym while on the move. Therefore, it will be impossible to link the current pseudonym with the adjusted location. This provides identity privacy and location tracking of a querying user.
- The security of the proposed DLSS is analyzed against the adversaries.
- The efficiency and effectiveness of the proposed scheme are demonstrated by the simulations. The simulation results show that the proposed DLSS preserves location privacy at low computation time and communication cost.

The rest of the paper is organized as follows: Existing location privacy protection approaches presented in section II. Section III provides an overview of the proposed scheme architecture and its detail. An elaborate discussion on the security part is presented in section IV, and simulation analysis is introduced in section V. Lastly, section VI concludes the paper.

II. RELATED WORK

Many promising privacy-preserving approaches have been proposed in LBSs in past years. Generally, they are categorized as TME-based approaches and TME-free approaches. In this section, we will describe the existing location privacy protection approaches used in LBSs and their drawbacks in detail.

A. TRUSTED MIDDLE ENTITY FREE APPROACHES

TME-free approaches require the user to communicate directly with the LSP. Thus, the user doesn’t need to trust any party while issuing a query request for the LSP [8], [9]. In collaboration based approaches, a user can collaborate with each other to satisfy their privacy requirements and through the mobile devices. Before sending any request to LSP, a querying user first collaborates with his/her nearby peers and request for the services [10], [11]. Shokri et al. proposed a novel scheme called Mobicrowd which allows the users to redeem the information from the nearby neighbors. This allows the querying user to get the services without contacting the LSP [12]. Another P2P communication called CAST is proposed to reduce the connection with LSP and request the queries from the peers first [13]. Jung et al. proposed a P2P architecture to protect the location information using personal data storage (PDS) [14]. Another decentralized...
privacy protection approach is introduced in [15] that allows the nodes to cache POIs data in which they are located in and serves the neighboring queries. A fully distributed architecture is proposed in [16] that allow the user to specify their privacy requirements and to find their cloaking regions without revealing their precise locations. The major limitation of collaboration-based approaches is that mobile devices must have powerful computational capabilities.

In addition to that, dummy based approaches allow the user to add noise to the location information, thereby protecting from the untrusted entities [17]–[20]. Niu et al. proposed an enhanced DLS scheme for protecting the location privacy of the user by considering a set of realistic dummy locations [21]. Another scheme in [22] presented a dummy selection algorithm that integrates caching to maximize privacy. However, the proposed scheme works well for snapshot queries. Chen et al. presented a privacy protection scheme to maximize location privacy by utilizing cache proxies and k-anonymity [23]. To enhance the existing dummy-based schemes, authors in [24] proposed an efficient solution by considering the spatial-temporal correlations between the neighboring location sets from various aspects such as direction similarity and time reachability. In [25] developed an efficient solution for generating the dummy locations while considering the semantic location information. The proposed algorithm allows the generated dummy locations to minimize the exposure of the user’s actual location and proved effective against the attackers. Another dummy location generation algorithm based on semantic quantification of location is presented in [26] that protects the privacy leakage of the user by considering various aspects such as historical query probability, location semantics and physical dispersion uniformity. The disadvantage of dummy-based approaches is that the addition of noise will affect the quality of service. Besides, the user privacy is sacrificed at the expense of the precision of the location information.

B. TRUSTED MIDDLE ENTITY BASED APPROACHES

The introduction of an intermediate into the system provides a protection barrier from an untrustworthy LSP. Pseudonym based methods and Anonymity based methods are some widely adopted methods in TME approaches. Pseudonym based methods are another popular TME-based approaches to interrupt the linkage between the user’s identity and his/her corresponding events [27]. Before forwarding the request to the LSP, the middle entity anonymizes their actual locations by pseudonyms. Beresford et al. proposed a mixzone model to enhance the privacy of the user with minimal computational complexity [28]. Gao et al. presented a privacy-preserving substructure to protect the trajectory of the querying user. The scheme enhances the mix zones model by considering time factor [29]. Palanisamy and Liu proposed another mixzone based framework proposed in [30] and [31] to minimize the location exposure by considering geometry of the zones, spatial and temporal resolution of location, movement patterns of users. The misbehaviour of the middle entity is one of the major drawback of pseudonym based approaches. Anonymity based approaches hides the user’s location information among other users. Reference [32] proposed a middleware privacy protection algorithm that introduced a centralized entity to process the user request. Gedik and Liu proposed location anonymization algorithm that runs by the trusted server to hide the user information such as identity and spatial temporal information from the LSP [33]. A transformation-based location anonymization technique presented in [34] to optimize the process of querying users’ request. However, this method provides an opportunity to track the querying user while using continuous LBSs. Hwang et al. proposed a new time obfuscated approach that splits the series of query issuing time of the querying user [35]. Liao et al. proposed a trajectory privacy protection algorithm that adopts the \((K-1)\) dummy locations using a sliding window technique [36]. To overcome all the defects, Zhang et al. proposed a caching and spatial k-anonymity scheme to improve the privacy of the user in continuous LBSs [37]. The proposed scheme adopts the concept of multilevel caching to minimize the interaction between a user and LSP. In this scheme, a middle entity called as anonymizer employs a markov model that predicts the user’s next location, number of the queries issued in the past, user’s mobility pattern. Another robust spatial cloaking technique presented in [38] for query and location privacy.

In addition to these approaches, semi-trusted middle entity based approaches are widely adopted in which the user will reveal partial information to the middle entity. Peng et al. presented a privacy-preserving scheme to enhance the privacy of the querying user by employing the a semi-trusted entity named as function generator [39]. The function of this entity is to distribute spatial parameters to the users’ and LSP so that they can perform mutual transformation between a real location and a fake location. Schlegal et al. proposed a two-server approach to protect the location privacy in continuous LBSs [40]. Their scheme requires a query server (or semi-TME) to hold the encrypted location details of the querying user and LSP to hold the decryption key. The major issue with this scheme is when the query server colludes with LSP or any malicious adversary, then user privacy can be breached. Similar work is presented by the authors in [41] to enhance the privacy of the user and reduces the overhead of the LSP. Zhang et al. proposed a uniform grid and caching scheme (UGC) in LBSs in which the user defined grid structure sent to an entity named as converter before sending it to the LSP. However, the scheme has some limitations: If a querying user requested information is not available in the anonymizer’s cache, then the user forms a cloaking region with the nearby neighbours. For that, a querying user needs to disclose his/her exact location, type of POIs, and query range. It is very hard to trust the nearby neighbours. In addition to that, semi-trusted anonymizer knows that a querying user must be somewhere at the center of the requested region. Since anonymizer knows the requested cells and LSP knows the uniform grid information of the user, this
might put users’ at great risk. Moreover, by analyzing the user’s request provides an opportunity to build the mobility pattern of the querying user. Failure of single anonymizer and performance bottleneck are the main drawback of these schemes. To solve this issue, Zhang et al. presented a multi-level scheme that combines Shamir threshold scheme with dynamic pseudonym technique, and k-anonymity approach to enhance the privacy of the user [42]. The fragmented request and responses are forwarded by the user’s selected multiple anonymizers which provides content privacy.

Performance congestion and central failure are the main limitations of using TME-based approaches. Once the middle entity colludes with the smart adversaries, it can pose a serious privacy threat.

In comparison with the existing approaches, we propose a dummy location scattering scheme (DLSS) that allows a querying user to request the services from the LSP without the aid of an intermediate entity. In order to securely access the services from the untrusted LSP, a user adopts a pseudonym mechanism with a dummy location technique. A user chooses an adjusted location within his/her own actual query range. Based on an adjusted location, a user generates a larger area in which his/her actual location is hidden. Following that, this user-generated larger area is divided into smaller sub-areas. Each of the fragmented sub-area has its own dummy location and a user-assigned pseudonym. The introduction of the time-delay technique adds another level of privacy to the proposed scheme in which dummy locations are randomly divided into groups and queries related to these groups of locations are requested with a time delay. In this way, the user gets the results from these grouped areas of the dummy locations without disclosing his/her information. Therefore, privacy can be achieved with the minimum exposure of a user information.

### III. PROPOSED SYSTEM ARCHITECTURE

This section describes the system architecture of the proposed work followed by the proposed DLSS in detail.

#### A. SYSTEM ARCHITECTURE

One of the primary objectives of the proposed scheme is to preserve the privacy of the user when using continuous LBSs. The proposed architecture has three entities: certificate authority, mobile users, and LSP. Fig. 2 illustrates the interaction between the main entities in the proposed DLSS.

1) **Certificate Authority (CA):** The certificate authority is a trustworthy entity that is in charge of issuing pseudonyms to the new users. The CA will be contacted only during the time of registration in the proposed scheme.

2) **Mobile users:** We realistically assumed that all mobile users are equipped with GPS and internet-enabled smartphones. In the proposed DLSS, a mobile user can get the services by sending a query to the LSP. Each mobile user can store information in his/her respective cache and hence can reuse that information in future queries.

3) **Location-based service provider (LSP):** LSP is a service provider that has the capacity to provide a variety of services to mobile users. LSP stores all the necessary details of the POIs (e.g., supermarkets, car parks, bars, and restaurants) in its database and is capable of satisfying a user’s request.

#### 1) BASIC NOTATIONS

In this subsection, we define the actual locations and adjusted locations of a querying user.

**a: ACTUAL LOCATION**

Let \( A_u = (u_x, u_y) \) is a geographic coordinates of a querying user. We denote \( S \) as an area in which a user is moving i.e.,

\[
A_u = (u_x, u_y) \in S. \tag{1}
\]

**b: ACTUAL QUERY AREA**

Let us consider that the querying user is looking for any POIs within \( R \) distance from his/her actual location \( A_u \) and is denoted by

\[
Q_u = \text{Area}(R, A_u). \tag{2}
\]

where \( \text{Area}(\cdot) \) denotes a circular area that is centred at \( A_u \) with the radius of \( R \).

**c: ADJUSTED LOCATION**

The adjusted location \( M_u \) is randomly generated location that are selected by the querying user within his/her actual query range \( Q_u \). From the privacy perspective, a user don’t want to reveal his/her actual location to any entity. Therefore, an adjusted location (or dummy location) are generated to prevent the adversaries from obtaining any location information of the user. In the following equation, \( m_x \) and \( m_y \) are the geographic coordinates of an adjusted location \( M_u \),
The weak adversary has limited knowledge about the user in LBSs. They are the passive adversary that can observe the communication between the users and LSP but could not have access to LSP’s database or even user information.

a: WEAK ADVERSARY
The weak adversary has limited knowledge about the user in LBSs. They are the passive adversary that can observe the communication between the users and LSP but could not have access to LSP’s database or even user information.

\[ Q'_u = \text{Area}(R, M_u). \] (4)

Fig. 3 shows an example of the generation of an adjusted location. With the help of actual location \( A_u \) and radius \( R \), a user generates an actual query area \( Q_u \). Thereafter, a user another virtual circle is constructed at the same location \( A_u \) but with a different radius \( R/2 \), and an adjusted location can be chosen outside the virtual circle as shown in Fig. 3. Following that, a user prepares a dummy requesting query area \( Q'_u \) centered at \( M_u \) and same radius \( R \).

e: SCATTERED DUMMY SUB-QUERY AREAS
The scattered dummy sub-areas \( d_z \) can be defined as a set of smaller circular areas to be generated inside the user-generated larger area. Each of the generated dummy sub-area \( d_z \) has its own radius \( r_z \) and a dummy location \( l_z \). Fig. 5 shows the scattered dummy sub-query area.

\[ d_z = \text{Area}(r_z, l_z). \] (5)

where: \( z = 1, 2, \ldots n \).

2) THREAT MODELS
An adversary is an entity that always tries to gain unauthorized access to a user’s sensitive information. An adversary can be categorized as a weak or strong adversary.

b: STRONG ADVERSARY
The strong adversary has the ability to access all information related to the user stored in the LSP’s database. In our proposed scheme, we have assumed that the strong adversary knows the algorithm and tries to take advantage of user’s sensitive information by compromising the LSP. Therefore, it is likely to have a privacy-preserving scheme that can protect the privacy of the user against these adversaries.

B. PROPOSED DUMMY LOCATION SCATTERING SCHEME (DLSS)
Consider a scenario in which a user issues a range query to know a particular type of POIs within a user-specified distance to his/her current location. Generally, a user’s current location and query constitute personal information. However, this information may be adequate to violate the user’s privacy. In this situation, location privacy protection comes into play. Therefore, the purpose of our work is to protect the location privacy of the user when using the LBSs. In the proposed scheme, each user has a respective cache where he/she can store the results provided by the LSP. Cached results can be reused for future queries, hence reduces the number of requests sent to the untrusted LSP. Lack of desired information in the user’s cache lets a user proceed with the dummy location scattering scheme (DLSS) and obtain the desired POIs from the LSP without revealing any location-related information. In the proposed scheme, the LSP can’t obtain any useful information from the query request.

The proposed DLSS has two-step processes namely the generation phase and scattering phase. The generation phase allows the user to generate a larger area in which a user’s actual location is encapsulated. Following that, the scattering phase creates several dummy locations with their respective query areas to be scattered on the predefined larger area. Thereafter, these dummy locations are divided based on the proposed algorithm into groups, and queries related to these groups of locations are requested with a time delay. In this way, the user gets the results from these grouped areas of the dummy locations without disclosing any information. In DLSS, a time-delay technique is introduced to confuse the LSP from obtaining any information about the actual querying user. Even if any malicious adversaries collect all the information of the scattered sub-areas, it would be difficult to gain any location-related information about a querying user. The following describes the working steps of the proposed scheme in detail. Table 1 shows the notations used in the proposed scheme. Following are the steps that describe the working process of the proposed DLSS in detail.

1) REGISTRATION
One of the most prominent ways to maintain the privacy of the location data is to replace the exact identity with the pseudonym (sometimes known as virtual identity or fake identity). Therefore, the registration process starts by
contacting the CA for issuing a set of pseudonyms. The user first generates a pair of public and private keys \((PK_u, SK_u)\) using RSA public-key cryptography [43]. The public key \(PK_u\) is used to encrypt the message for secure transmission at both ends. While the private key \(SK_u\) is used to decrypt the message and retrieves the sensitive information. The registration process requires a querying user to send the identity \(ID_u\) and a public key \(PK_u\) to CA. The communication between the new user and CA is always encrypted. The new user encrypts the registration request \(R_{Req}\) with the public key of the CA and sends it to the CA as follows:

\[
R_{Req} = \{PK_{CA}(ID_u, PK_u)\}.
\]  

CA decrypts the registration request \(R_{Req}\) with the corresponding private key and acknowledges a user with a set of pseudonyms \(P_u\) and access identifier \(A_{ID}\). After receiving a \(R_{Resp}\) from the CA, the user decrypts the message to get the response as follows:

\[
R_{Resp} = \{PK_u(P_u, A_{ID})\}.
\]

After the registration process is completed, a user has a set of pseudonyms that act as a virtual identity when the request for services from the LSP. However, an access identifier \((A_{ID})\) is used to renew the pseudonym set after using all the pseudonyms.

2) REQUEST GENERATION

In this paper, we mainly focus on the range queries i.e. a user looking for a particular type of POIs within a user-defined distance of actual location \(A_u\). For privacy concerns, a user doesn’t want to reveal \(A_u\). Therefore, the generation phase allows a user to generate an adjusted (or dummy) location and create a larger area (aka minimum bounding area \(MBA\)) in which a user’s actual location is encapsulated. To preserve the location privacy against the untrusted LSP, a querying user must first generate an adjusted location \(M_u\) using algorithm 1.

A querying user specifies a range with radius \(R\) centered at his/her actual location \(A_u = (u_x, u_y)\). Thereafter, a user sets a virtual circle with radius \(R/2\) centered at his/her actual location \(A_u\). Now, a user has two concentric circles centered at the same location but with a different radius.

Thereafter, a user chooses any random location \(M_u = (m_x, m_y)\) within areas of two concentric circles. This random location is entitled as an adjusted location of a user. The adjusted location \(M_u\) of a user should be selected in such a way that it must be greater than the virtual circle, but less than the actual query area \(Q_u\). Afterward, a querying user can send or receive any location-related information with his/her adjusted location \(M_u\). In the case of continuous requests, a querying user keeps switching an adjusted location, which also protects the user’s trajectory. In this way, it will be impossible for the adversaries to infer the user’s information while the user is on the move. The selection of an adjusted location is defined as follows:

\[
m_x = R' \cos \theta + u_x,
\]

\[
m_y = R' \sin \theta + u_y.
\]

where: \(R/2 \leq R' \leq R\).

Following that, a querying user generates another query area with the same radius \(R\) but centered at the user’s adjusted location \(M_u\). This query area is known as the dummy requesting query area \(Q_u'\). The idea of having \(Q_u'\) is to encapsulates the user’s actual location inside the user-generated dummy requesting query area \(Q_u'\). Moreover, one could have the information that the user’s actual query area is a subsection of the dummy requesting range area \(Q_u'\), and a user’s actual location can be anywhere in this area. Thereby, protecting the location privacy of the user.

In the next step, a querying user divides the dummy requesting query area \(Q_u'\) into quadrants and assigns an edge point \((a_i, b_i)\) intersecting the \(Q_u'\). These four edge points \((a_i, b_i)\) should be located on the \(Q_u'\) from the center \(M_u\) in the direction of north, south, east, and west respectively as shown in fig. 4. Thereafter, a user creates four points \((A_i', B_i')\) which is \(\beta\) distance away from the user-generated \((a_i, b_i)\) locations. A square is constructed where the middle point of each side is \((A_i', B_i')\). This dotted square is known as a minimum bounding area (or user-generated larger area). The detailed process is shown in algorithm 2. The scattering phase allows a user to divide the \(MBA\) into \(k \times k\) equal-sized user-specified grid \(g_k\), where \(k\) is the size of grid. It is important to note here that the value of the grid is chosen by a querying user depending upon his/her requirements. Now, assign the dummy locations intersecting the grid lines on the \(MBA\) as

| Table 1. Summary of notations. |
|--------------------------------|
| Notation | Description |
|---------|-------------|
| \(P_u\)  | Set of pseudonyms |
| \(R_{id}\)  | Request identifier |
| \(Q_{id}\)  | Query identifier |
| \(Q_u\)  | Actual query area |
| \(Q_u'\)  | Dummy requesting query area |
| \(Pub_u\)  | Public key of the LSP |
| MSG_{u,2LSP} | Messages from scattered dummy locations to LSP |
| MSG_{LSP,2u} | Message from LSP to the respective dummy locations |

![FIGURE 4. Example of minimum bounding area.](image)
Algorithm 1 Adjusted Location Generation

\textbf{Input:} Actual location \( A_u = (u_x, u_y) \), \( R \)
\textbf{Output:} Adjusted location \( M_u = (m_x, m_y) \)

1: A user sets a query radius centered at his/her actual location \((u_x, u_y)\) with radius \( R \).
2: With actual location \( A_u \), a virtual circle is constructed but with radius \( R/2 \).
3: A user chooses any random location \( M_u = (m_x, m_y) \).
4: Check distance \( \text{Dist} = \sqrt{(m_x - u_x)^2 + (m_y - u_y)^2} \)
5: if \( R/2 \leq \text{Dist} \leq R \) then
6: Adjusted location \( M_u \leftarrow (m_x, m_y) \)
7: else
8: Go to step 3
9: end if
10: Return \( M_u = (m_x, m_y) \).
11: Exit

\( l_1, l_2, \ldots, l_z \), where : \( z \) is intersecting points on MBA. To add more privacy, a user assigns a different query area \( r_z \) to each of the generated dummy locations \( l_z \). Also, a different pseudonym \( P_z \) is assigned to these dummy locations. In this way, a querying user fractionates a minimum bounding area into several small sub-areas. Each fractionated sub-area has its own unique pseudonym and a dummy location \( l_z \) centered with a different query area \( r_z \). Fig. 5 shows the scattering process of the dummy locations on MBA. In the figure, \( l_1, l_2, l_3 \) and \( l_4 \) are the dummy locations on the MBA.

The scattering process provides supplementary privacy to a querying user by creating a pool of dummy locations. Afterward, these dummy locations are divided based on the proposed algorithm into groups, and queries related to these groups of locations are requested with a time delay. In this way, a user obtains the results from these grouped areas of the dummy locations without disclosing any location-related information to the LSP. In other words, a user prepares a pool of scattered dummy locations and their respective sub-areas \( d_z \) that covers the dummy requesting query area of a querying user. Furthermore, at a given time, only one group (or a subset of \( d_z \)) is selected from the pool and sends to LSP for the services. The LSP when receives the requests and searches from its database and responds with the requested POIs by presuming that the request came from multiple users. After a certain period of time, a user then selects another group (or another subset of \( d_z \)) and sends it to the LSP. A user continues to process in this way until the pool becomes empty. The purpose of using the time-delay concept is to confuse the LSP from obtaining any location-related information about the actual querying user. Even if LSP collects all the information of these sub-areas, it would be challenging for the LSP or even adversaries to get any useful information about a querying user. In this way, a user’s actual query area \( Q_u \) and actual location \( A_u \) remains hidden from the LSP. The LSP searches and responds with the POIs information related to the received groups at different time intervals. A querying user when receiving POIs information from these groups can extract the accurate POIs information that requires in the actual query area \( Q_u \). At a given time \( t1 \), some dummy locations are selected and prepares a request message for the LSP as follows:

\[
MSG_{l_1LSP} = \{\text{Pub}_s(P_z, R_{id}, Q_{id}, \text{POI}_{type}, r_z)\}. \tag{9}
\]

Each dummy location has its own pseudonym \( P_z \), request ID \( R_{id} \), query area \( r_z \), query identifier \( Q_{id} \). For secure communication, all messages are encrypted with the public key of the server \( P_s \).

Similarly, at time \( t2 \), another subset of dummy locations are selected and forwards a request to the LSP as follows:

\[
MSG_{l_2LSP} = \{\text{Pub}_s(P_{z1}, R_{id}, Q_{id}, \text{POI}_{type}, r_{z1})\}. \tag{10}
\]

3) REQUEST PROCESSING (BY LSP)

In the result processing phase, LSP receives the requests from these groups of dummy locations, decrypts, and processes them accordingly. It searches from its database about the queried POIs and prepares a response message to each of the dummy locations as follows:

\[
MSG_{LSP2d_1} = \{R_{id}, Q_{id}, \text{Set of POIs}\}, \tag{11}
\]

\[
MSG_{LSP2d_2} = \{R_{id}, Q_{id}, \text{Set of POIs}\}. \tag{12}
\]

Based on the information sent from the LSP, a user receive the responses and get the desired information in the same order. The positive side of the proposed DLSS is that no entity should guess any location-related information of a querying user because of the generation of an adjusted location, construction of user-defined MBA, and a time-delay concept used in the proposed scheme. Furthermore,
Algorithm 2 Scattered Dummy Location Sub-Areas

| Input: Adjusted location $M_u = (m_x, m_y), R$
| Output: Set of scattered dummy sub-areas $d_1 \ldots d_z$

1. A user with his/her adjusted location $M_u$ and radius $R$ prepares a query area.
2. Split the query area into quadrants and assign each of their edge point on this query area as $(a_i, b_i)$.
3. Choose a security parameter $\beta$ where: $2 \leq \beta \leq R/4$.
4. Generate four points $(A'_i, B'_i)$ which is at $\beta$ distance from $(a_i, b_i)$.
5. Construct a square (or MBA) such that the middle point of each side of the square is $(A'_i, B'_i)$.
6. Divide the MBA into equal sized user-defined grid $g_k$ where $k = size \ of \ the \ grid$.
7. Let $l_1, l_2, \ldots, l_2$ are dummy locations where $z = no. \ of \ intersecting \ points \ on \ MBA$.
8. The radius $r$ of these dummy locations are selected at random where: $R/2 \leq r \leq 3 \times R/4$.
9. Let a circle with radius $r_c$ centered at $l_z$ be a sub-area $d_z$.
10. Return $d_1, d_2 \ldots d_z$.
11. Exit

the adoption of the privacy-preserving-based pseudonym mechanism together with the scattered dummy locations and their sub-query areas concept makes it difficult for the malicious entity to acquire the identity or even an adjusted location of a querying user.

C. THEORETICAL ANALYSIS OF ALGORITHMS 1 AND 2

Algorithm 1 generates an adjusted location $M_u$ around the actual location $A_u$ by just setting the loci with $R/2$ and $R$ circles. So, the complexity is $O(1)$. Following that, Algorithm 2 consists of mainly three components. Firstly, defining the boundaries of the MBA based on the adjusted location and $\beta$ whose complexity is $O(1)$. Secondly, a user-generated area MBA is divided into $k$ equal sized grids, where $k$ is a function/multiplier of $(R + \beta)$. Its complexity is $O(ceil((R + \beta)/k) \times ceil((R + \beta)/k)))$. Lastly, selecting a random radius $r$ for all grid corners aka $l_z$ whose complexity is $O(z)$. Therefore, the overall complexity is $O(ceil((R + \beta)/k) \times ceil((R + \beta)/k)))$.

IV. SECURITY ANALYSIS

Because our paper is dealing with location privacy, we considered the attacks that mainly affect the privacy of the querying users. This section describes the security analysis of the proposed DLSS against various attacks.

A. PRIVACY

It is necessary to keep the user’s information private so that it will not be misused by any malicious entities in the system when using the services.

1) PRIVACY AGAINST WEAK ADVERSARY

As mentioned previously, a weak adversary is an occasional eavesdropper who has limited information about the querying user. In the proposed DLSS, we want to ensure that this eavesdropper cannot find the location information of a querying user except by random guess. We have considered the game played between the challenger $C$ and the adversary $A$. In our scenario, a user and LSP are considered as the challenger. An occasional eavesdropper generates two locations $L_1$ and $L_2$ for a given larger area $S$ and passes it to $C$. $C$ randomly chooses $L_c$ where $c \in (0, 1)$. Using $L_c$ and $\beta$, $C$ generates a larger area (MBA) and with the user-defined grid size, divides the area into smaller sub-query areas. $C$ generates a request message by encrypting the $R_{id}, Q_{id}, POI_{type}$, and forwards it to $A$. $A$ wins the game if he can extract the correct information.

**Theorem 1:** Our proposed DLSS achieves privacy against the eavesdroppers.

**Proof:** Because the request is encrypted with the public key of the $C$, an eavesdropper can’t get any location information about a querying user. However, to obtain the location details of the user, the weak adversary must have the following information:

1) Since all the communication between the new user and CA is encrypted, a weak adversary cannot access have access to a user’s information without knowing their corresponding private key. In addition to that, an eavesdropper cannot access to actual user location information because the user-generated scattered dummy locations send the requests by encrypting them with the public key of the LSP.

2) The process of generating an adjusted location $M_u$ of the querying user. This is because a user chooses a random location inside the query area by generating a virtual circle. Each time a request is made, a user chooses a different adjusted location.

3) The security parameter $\beta$ is randomly chosen at the time of generating the minimum bounding area MBA. This MBA is constructed by considering the user’s adjusted location $M_u$.

4) The nature of sending the request of dummy sub-query areas $d_z$ with some time-delay is completely random which makes it difficult for the weak adversary from knowing any information about the actual querying user. Even by analyzing these random requests, one could only guess that the user must be somewhere within this area.

2) PRIVACY AGAINST STRONG ADVERSARY

In this scenario, we have considered that the strong adversary is the untrustworthy LSP who can access the users’ information in its database. Since LSP manages all query-related information about a querying user, and hence, can be considered as a truthful but curious entity. Consider a game played between the challenger $C$ and the adversary $A$. In this scenario, the challenger is a user and the smart adversary is...
the LSP. An adversary $A$ creates two locations $L_1$ and $L_2$ passes them to the challenger $C$. $C$ randomly chooses $L_c$ where $c \in 0, 1$. Using $L_c$, $\beta$, and user-defined grid size, $C$ generates the set of dummy sub-query areas $(d_1, d_2, \ldots, d_c)$ and forwards it to $A$ and $A$ wins the game if it extracts the correct value of $c$.

**Theorem 1:** Our proposed DLSS achieves privacy against the strong adversary.

**Proof:** To obtain the information about a querying user, a strong adversary must know the process of generating the adjusted location $M_a$ of a querying user given only query radius $R$. This is because a user’s actual location can be anywhere within the virtual circle. It is important to note here that all dummy locations $(l_1, l_2, \ldots, l_c)$ have different query radius $(r_1, r_2, \ldots, r_c)$ that constitutes their sub-query areas $(d_1, d_2, \ldots, d_c)$. These scattered dummy locations are assigned with unique pseudonyms $(p_1, p_2, \ldots p_c)$ at the time of the request.

Moreover, these dummy locations are divided into groups, and queries related to these groups are requested with a time delay. It might possible that at a given time $t_1$, in a group only one dummy location sends a request to the LSP, and at time $t_2$ more than three dummy locations send the request. Therefore, this random pattern would be challenging for the strong adversary to guess any information about the querying user. Also, if the strong adversary collects all information of these sub-areas $d_c$, it is still impossible to guess the adjusted location $M_a$, and hence, actual location $A_u$ of the querying user.

**B. PSEUDONYMITY**

A querying user is not identified by any malicious adversary in our proposed DLSS. Instead of using a static identifier at all times, a querying user uses a different virtual identity to the malicious attacker to collect any information. Hence, eavesdroppers can’t get any information with their corresponding private keys.

**D. LOCATION TRACKING**

Location tracking can be used to track any targeted user over a period of time. However, the use of one pseudonym at all times can lead to the location tracking of the user. Our proposed DLSS is resistant to location tracking. Every time a request is made, a new adjusted location $M_a$ is chosen by a user inside the actual query area $Q_u$. Also, a unique pseudonym is adopted with an associated adjusted location. For additional privacy, a unique pseudonym is assigned to each of the user-generated dummy locations.

Even monitoring the sequence of the dummy locations’ requests after a certain time interval would not be effective for tracking the actual querying user. One could only make a random guess that a querying user is somewhere in this larger region. Therefore, the dynamic nature of the locations, as well as pseudonyms, makes it impossible to gain any sensitive information about the targeted user.

**E. OBSERVATION ATTACK**

An observation attack is a special kind of attack where an attacker can collect user information through observation. For example, if a querying user uses a pseudonym to protect privacy in continuous LBSs. Through continuous observation, an attacker can retrace all prior locations of the querying user with the same pseudonym by a single correlation. However, our proposed DLSS can withstand this attack as a querying user in the system keeps changing the pseudonyms when using the continuous LBSs. A user is moving continuously and requests the desired POIs with a different query radius. A user always selects a different adjusted location $M_a$ and a different value of $\beta$ for generating the MBA to hide his/her location inside the larger privacy area. Furthermore, the scattering of the dummy locations and their sub-areas together with time delay would not allow the malicious attacker to collect any information. Hence, protecting a user against the observation of an attacker.
V. SIMULATIONS

In this section, we evaluate the efficiency of the proposed DLSS experimentally with the UGC scheme [41] and DPP scheme [42] under various system settings. All the experiments were performed on an Intel-Core i5 3.4 GHz machine with 8 GB RAM. The simulations were carried out on the MATLAB R2017a platform. A user aligns the origin of the coordinate system (0, 0) with a particular GPS coordinate (e.g., Latitude: $-37^\circ 15' 0.5040''$ S, Longitude: $145^\circ 7' 8.5224''$ E as (0, 0)). In the experiments, we considered that the mobile users are uniformly distributed on an area of 1248 km$^2$, Geelong, Melbourne. Fig. 6 shows the geographic locations of different restaurants in Geelong city. The UGC scheme adopts order-preserving symmetric encryption and k-anonymity techniques to improve the privacy of the user. A user prepares a request containing the grid identifiers and sends it to the anonymizer. A successful match returns the results to the user. Otherwise, a user prepares a cloaking region that contains $(k-1)$ users. The anonymizer forwards the cloaking region to the LSP. Based on the received information, LSP responds with a set of POIs information. However, the scheme considered that the neighbors are trusted. A user sends his/her exact location, radius, and type of POIs to the nearby neighbors for generating a cloaking region. The anonymizer can deduce the approximate location of the user if the key is compromised. If anonymizer colludes with converter or LSP then the adversary can guess that the user could not be at the edges of the query area. Moreover, single-point failure and performance bottleneck are other major issues of using this scheme. To solve these issues, the DPP scheme proposed the concept of introducing multiple anonymizers between the user and the LSP. The scheme adopts Shamir threshold mechanism, dynamic pseudonym mechanism, and K-anonymity to improve the privacy of the user. A user divides the query content into multiple pieces of information using Shamir threshold mechanism. A request message containing the user’s location information and other query-related information sends to one of the anonymizers. Based on the user’s location, a cloaking region is generated and sends to the LSP. Although the scheme protects the privacy of the user but collusion of all anonymizers and LSP can pose serious privacy threats to the user and gains access to the user’s location and query information.

We analyzed the effectiveness of the proposed DLSS with the UGC scheme and DPP scheme by varying the query area, no. of the POIs, no. of the querying users, and no. of grids. We measured the average values of computation time and communication costs involved in the proposed DLSS.

A. IMPACT OF THE QUERY AREA

In this subsection, in order to analyze the impact of the query area, we calculate the computation time and communication cost by varying the querying radius $R$. Results are presented in Fig. 7. For the simulations we set $POIs = 2000$, and $Grid size = 3$. It can be seen from Fig. 7 (a) that there is a very small increase in the computation time while changing the $R$ as compared to the UGC and DPP schemes. The reason is due to the scattering of dummy locations on the user-defined larger area MBA and a user’s query area is covered by these dummy sub-query areas. However, in UGC and DPP schemes there is an increase in the computation time because a cloaking area is generated based on the user’s actual
FIGURE 8. Variation of computation time and communication cost against no. of POIs.

B. IMPACT OF THE POIs

In this subsection, we have analyzed the computation time and communication cost under varying POIs as shown in Fig. 8. We have considered $R = 1000$ m, Grid size = 3. It can be observed from Fig. 8 (a) that the computation time increase with the increase in the number of POIs. This is because when the POIs in the particular area increase, LSP needs more time to search the requested POIs. Our proposed DLSS has greater superiority over the UGC and DPP schemes because when POIs in the area increase, a large no. of POIs is covered by the user-generated dummy sub-query areas. A querying user filters the accurate POIs of an actual query area interest. On the other hand, UGC Scheme takes more time for transmission and matching the POIs information according to the spatial cloaking region generated by the user. Similarly, the computation time in the DPP scheme also increases with the increase in the no. of POIs. This is because the LSP and user take a longer time to search and process the request. In addition to that, the results are received according to the cloaking area and then refined according to the user’s interest. Therefore, contributes more computation time.

Similarly, Fig. 8 (b) shows that communication cost increase with the increase in no. of the POIs. The reason is
that as POIs increases in a particular query area, more data is returned from the LSP to the scattered dummy locations. But in UGC and DPP schemes, the query is processed by the anonymizer, and then by LSP. Therefore, more POIs in the area means more no. of POIs returned by the LSP based on the cloaking area. Following that, an anonymizer filters out the accurate POIs to a querying user. This contributes to longer communication costs.

C. IMPACT OF THE GRIDS

We have analyzed the performance of the proposed DLSS under varying grids when changing $R$ and POIs. It can be observed from Fig. 9 (a) and (b) that the computation time and data transfer cost of the proposed DLSS increases with the increase in the grid division while changing $R$ from 500 to 1000. This is because increasing the grid size means more dummy locations on the MBA are scattered and hence, more areas will be covered by these dummy locations. As a result, more time and cost are involved in processing the request from the user to the LSP, and then LSP to a user.

Similarly, Fig. 10 (a) and (b) display the performance of our proposed DLSS when POIs increase from 1000 to 2000 while varying the grid size. It can be seen from the figure that the computation time and communication cost increase with the increase in grid size while varying the no. of POIs. This is because an increase in grid size requires more areas to be covered by the user-generated dummy locations on the MBA. It is clear from the figure that the more POIs in the region, the more time the LSP takes to search and refine the results. This also contributes to more communication overhead.

D. IMPACT OF THE QUERYING USERS

In this subsection, we have analyzed the outcome of the computation time and communication cost under varying querying users while $R = 1000$ m, $POIs = 2000$, $Grid\,\,size\,\,=\,3$. Results are shown in Fig. 11. Let us consider a scenario where we have multiple querying users who issue an LBS query. It is clear from Fig. 11 (a) that the time required to process an LBS query from the multiple querying users increases. However, the UGC scheme outperforms our proposed DLSS in terms of computation time. This is because the proposed DLSS requires more time to process the same request due to the addition of the time-delay technique in the sub-areas $d_z$. To protect against the adversaries, the user
divides the larger privacy area MBA into smaller sub-query areas and assigns them in a group with a time delay. At a given time, only one group is allowed to send the request to the LSP. The LSP serves the requests of several sub-query areas that are the portions of a larger privacy area. Therefore, LSP takes more time to process the request from the multiple querying users. Additionally, the computation time of the DPP scheme is higher than the proposed DLSS and UGC scheme. This is because there are multiple anonymizers between the user and LSP. Each user has to coordinate the selected anonymizers to forward the requested information to the LSP. Hence, it might take a longer time for processing multiple users’ requests. On the other hand, Fig. 11 (b) shows that the communication costs are higher in the UGC and DPP schemes than in the proposed DLSS. The is because the generated cloaking area contributes more data transmission from LSP to anonymizer, and then from anonymizer to the LSP. Thus, resulting in more communication overhead.

VI. CONCLUSION
In this paper, a novel dummy location scattering scheme called DLSS, is presented to protect the location privacy of the user in continuous LBSs. Since the DLSS divides the user-generated larger area into subareas, no malicious adversaries can guess the actual location of the user. Another important benefit of using the proposed DLSS is the time-delay requesting technique that introduces some time gap between the scattered dummy locations’ requests to ensure that a user requests the services with a different query area. In addition to that, the user adopts a new pseudonym to prevent identification so that a malicious adversary cannot track the user. The simulation results show that the proposed DLSS is effective in terms of computation time and communication cost.

REFERENCES
[1] R. Schlegel, C.-Y. Chow, Q. Huang, and D. S. Wong, “Privacy-preserving location sharing services for social networks,” IEEE Trans. Services Comput., vol. 10, no. 5, pp. 811–825, Sep. 2017.
[2] L. Li, R. Lu, and C. Huang, “EPLQ: Efficient privacy-preserving location-based query over outsourced encrypted data,” IEEE Internet Things J., vol. 3, no. 2, pp. 206–218, Apr. 2016.
[3] J. Son, D. Kim, M. Z. A. Bhuiyan, R. Tashakkori, J. Seo, and D. H. Lee, “Privacy enhanced location sharing for mobile online social networks,” IEEE Trans. Sustain. Comput., vol. 5, no. 2, pp. 279–290, Apr. 2020.
[4] H. Kido, Y. Yanagisawa, and T. Sato, “An anonymous communication technique using dummies for location-based services,” in Proc. Int. Conf. Pervasive Services (ICPS), 2005, pp. 88–97.
[5] H. Liu, C. S. Jensen, and M. L. Yu, “PAD: Privacy-aware dummy-based location privacy in mobile services,” in Proc. 7th ACM Int. Workshop Data Eng. Wireless Mobile Access, 2008, pp. 16–23.
[6] S. Zhang, Q. Liu, and Y. Lin, “Anonymizing popularity in online social networks with full utility,” Future Gener. Comput. Syst., vol. 72, pp. 227–238, Jul. 2017.
[7] H. Zhu, F. Liu, and H. Li, “Efficient and privacy-preserving polygons spatial query framework for location-based services,” IEEE Internet Things J., vol. 4, no. 2, pp. 536–545, Apr. 2017.
[8] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, “The new casper: Query processing for location services without compromising privacy,” in Proc. 32nd Int. Conf. Large Data Bases, 2006, pp. 763–774.
[9] A. Pingley, N. Zhang, X. Fu, H.-A. Choi, S. Subramaniam, and W. Zhao, “Protection of query privacy for continuous location based services,” in Proc. IEEE INFOCOM, Apr. 2011, pp. 1710–1718.
[10] J. Domingo-Ferrer, “Microaggregation for database and location privacy,” in Proc. Int. Workshop Next Gener. Inf. Technol. Syst. Berlin, Germany: Springer, 2006, pp. 106–116.
[11] T. Peng, Q. Liu, D. Meng, and G. Wang, “Collaborative trajectory privacy preserving scheme in location-based services,” Inf. Sci., vol. 387, pp. 165–179, May 2017.
[12] R. Shokri, G. Theodorakopoulos, P. Papadimitratos, E. Kazemi, and J. P. Hubaux, “Hiding in the mobile crowd: Locationprivacy through collaboration,” IEEE Trans. Dependable Secure Comput., vol. 11, no. 3, pp. 266–279, Jun. 2014, pp. 106–116.
[13] R. Gupta and U. P. Rao, “Achieving location privacy through CAST in location based services,” J. Commun. Netw., vol. 19, no. 3, pp. 239–249, 2017.
[14] K. Jung and S. Park, “Collaborative caching techniques for privacy-preserving location-based services in peer-to-peer environments,” in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2017, pp. 4497–4506.
[15] H. Jin and P. Papadimitratos, “Resilient privacy protection for location-based services through decentralization,” ACM Trans. Privacy Secur., vol. 22, no. 4, pp. 1–36, Nov. 2019.
[16] S. Liu, J. H. Wang, J. Wang, and Q. Zhang, “Achieving user-defined location privacy preservation using a P2P system,” IEEE Access, vol. 8, pp. 45895–45912, 2020.
[17] C. A. Ardagna, M. Cremonini, S. D. C. D. Vimercati, and P. Samarati, “An obfuscation-based approach for protecting location privacy,” IEEE Trans. Dependable Secure Comput., vol. 8, no. 1, pp. 8–27, Jan. 2011.
[18] T. Haru, A. Suzuki, M. Iwata, Y. Arase, and X. Xie, “Dummy-based user location anonymization under real-world constraints,” IEEE Access, vol. 4, pp. 673–687, 2016.
[19] S. Hayashid, D. Amagata, T. Haru, and X. Xie, “Dummy generation based on user-movement estimation for location privacy protection,” IEEE Access, vol. 6, pp. 22958–22969, 2018.
[20] S. Zhang, G. Wang, Q. Liu, and J. H. Abawajy, “A trajectory privacy-preserving scheme based on query exchange in mobile social networks,” Soft Comput., vol. 22, no. 18, pp. 6121–6133, Sep. 2018.
[21] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, “Achieving k-anonymity in privacy-aware location-based services,” in Proc. IEEE Conf. Comput. Commun. (INFOCOM), Apr. 2014, pp. 754–762.
[22] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, “Enhancing privacy through caching in location-based services,” in Proc. IEEE Conf. Comput. Commun. (INFOCOM), Apr. 2015, pp. 1017–1025.
[23] M. Chen, W. Li, X. Chen, Z. Li, S. Lu, and D. Chen, “LPPS: A distributed cache pushing based k-anonymity location privacy preserving scheme,” Mobile Inf. Syst., vol. 2016, Aug. 2016, Art. no. 7164126.
[24] H. Liu, X. Li, H. Li, J. Ma, and X. Ma, “Spatiotemporal correlation-aware dummy-based privacy protection scheme for location-based services,” in Proc. IEEE Conf. Comput. Commun. (INFOCOM), May 2017, pp. 1–9.
[25] G. Sun, S. Cai, H. Yu, S. Maharanj, V. Chang, X. Du, and M. Guizani, “Location privacy preservation for mobile users in location-based services,” IEEE Access, vol. 7, pp. 87425–87438, 2019.
[26] X. Shi, J. Zhang, and Y. Gong, “A dummy location generation algorithm based on the semantic quantification of location,” in Proc. IEEE Int. Conf. Artif. Intell. Comput. Appl. (ICAICA), Jun. 2021, pp. 172–176.
[27] M. Babaghayou, N. Labraoui, A. Aba Ari, N. Lagraa, and M. A. Ferrag, “Pseudonym change-based privacy-preserving schemes in vehicular ad-hoc networks: A survey,” J. Inf. Secur. Appl., vol. 55, Dec. 2020, Art. no. 102618.
[28] A. R. Beresford and F. Stajano, “Location privacy in pervasive computing,” IEEE Pervasive Comput., vol. 2, no. 1, pp. 46–55, Jan. 2003.
[29] S. Gao, J. Ma, W. Shi, G. Zhan, and C. Sun, “TPPF: A trajectory privacy-preserving framework for participatory sensing,” IEEE Trans. Inf. Forensics Security, vol. 8, no. 6, pp. 874–887, Jun. 2013.
[30] B. Palanisamy and L. Liu, “MohibMix: Protecting location privacy with mix-zones over road networks,” in Proc. IEEE 27th Int. Conf. Data Eng., Apr. 2011, pp. 494–505.
[31] B. Palanisamy, L. Liu, K. Lee, A. Singh, and Y. Tang, “Location privacy with road network mix-zones,” in Proc. 8th Int. Conf. Mobile Ad-Hoc Sensor Netw. (MSN), Dec. 2012, pp. 124–131.
[32] M. Gruteser and D. Grunwald, “Anonymous usage of location-based services through spatial and temporal cloaking,” in Proc. 1st Int. Conf. Mobile Syst., Appl. Services (MobiSys), 2003, pp. 31–42.
[33] B. Gedik and L. Liu, “Protecting location privacy with personalized k-anonymity: Architecture and algorithms,” IEEE Trans. Mobile Comput., vol. 7, no. 1, pp. 1–18, Jan. 2008.
[34] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, “Preventing location-based identity inference in anonymous spatial queries,” IEEE Trans. Knowl. Data Eng., vol. 19, no. 12, pp. 1719–1733, Dec. 2007.
[35] R.-H. Hwang, Y.-L. Hsieh, and H.-W. Chung, “A novel time-obfuscated algorithm for trajectory privacy protection,” IEEE Trans. Serv. Comput., vol. 7, no. 2, pp. 126–139, Apr./Jun. 2014.
[36] D. Liao, H. Li, G. Sun, and V. Anand, “Protecting user trajectory in location-based services,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2014, pp. 1–6.
[37] S. Zhang, X. Li, Z. Tan, T. Peng, and G. Wang, “A caching and spatial K-anonymity driven privacy enhancement scheme in continuous location-based services,” Future Gener. Comput. Syst., vol. 94, pp. 40–50, May 2019.
[38] C.-Y. Chow and M. F. Mokbel, “Enabling private continuous queries for revealed user locations,” in Proc. Int. Symp. Spatial Temporal Databases. Berlin, Germany: Springer, 2007, pp. 258–275.
[39] T. Peng, Q. Liu, and G. Wang, “Enhanced location privacy preserving scheme in location-based services,” IEEE Syst. J., vol. 11, no. 1, pp. 219–230, Mar. 2017.
[40] R. Schlegel, C. Y. Chow, Q. Huang, and D. S. Wong, “User-defined privacy grid system for continuous location-based services,” IEEE Trans. Mobile Comput., vol. 14, no. 10, pp. 2158–2172, Oct. 2015.
[41] S. Zhang, K.-K. R. Cho, Q. Liu, and G. Wang, “Enhancing privacy through uniform grid and caching in location-based services,” Future Gener. Comput. Syst., vol. 86, pp. 881–892, Sep. 2018.
[42] S. Zhang, G. Wang, and Q. Liu, “A dual privacy preserving scheme in continuous location-based services,” IEEE Internet Things J., vol. 5, no. 5, pp. 4191–4200, Oct. 2018.
[43] X. Yi, R. Paulet, E. Bertino, and V. Varadharajan, “Practical approximate k nearest neighbor queries with location and query privacy,” IEEE Trans. Knowl. Data Eng., vol. 28, no. 6, pp. 1546–1559, Jun. 2016.

NISHA NISHA received the bachelor’s degree in information technology and the master’s degree in computer science from the Kurukshetra University, India, in 2012 and 2014, respectively. She is currently pursuing the Ph.D. degree with Deakin University, Australia.

N. Nisha et al.: Enhanced Location Scattering Based Privacy Protection Scheme

* * *

IYNKARAN NATGUNANATHAN received the B.Sc. Eng. degree (Hons.) in electronics and telecommunication engineering from the University of Moratuwa, Katubedda, Sri Lanka, in 2007, and the Ph.D. degree from Deakin University, Australia, in 2012. From 2006 to 2008, he was a Software Engineer with Millennium Information Technology (Pvt) Ltd., Malabe, Sri Lanka. He is a Research Fellow with the School of Information Technology, Deakin University. His research interests include the IoT privacy, digital watermarking, audio, image processing, and telecommunication.

YONG XIANG (Senior Member, IEEE) received the Ph.D. degree in electrical and electronic engineering from the University of Melbourne, Australia. He is a Professor with the School of Information Technology, Deakin University, Australia. He has published six monographs, over 190 refereed journal articles, and numerous conference papers in his research areas. His research interests include information security and privacy, signal and image processing, data analytics and machine intelligence, the Internet of Things, and blockchain. He has served as the honorary chair, the general chair, the program chair, the TPC chair, the symposium chair, and the track chair for many conferences, and was invited to give keynotes at a number of international conferences. He was the Associate Editor of IEEE SIGNAL PROCESSING LETTERS and IEEE ACCESS, and the Guest Editor of IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS and IEEE MULTIMEDIA. He is the Senior Area Editor of IEEE SIGNAL PROCESSING LETTERS and the Associate Editor of IEEE COMMUNICATIONS SURVEYS AND TUTORIALS.