International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 06 , June - 2026

A Privacy-Preserving Framework for Crowdsourced Location-Based Services

Mohammed Tali Almalchy
University of Misan, Mechanical Department, Faculty of Engineering, Misan, Iraq

Imad Ali Hasoon
Faculty of Technical, University of Imam Ja'far Al-Sadiq, Misan, 1001, Iraq

***Abstract*- The increasing integration of mobile devices and location-based services (LBS) into everyday life requires the protection of users' location privacy, and this has become a critical challenge in recent times. Hence, this study proposes the integration of the crowdsourcing concept that leverages users' collective input to offer location-related data and services via a novel privacy-preserving framework that integrates crowdsourcing into LBS while ensuring anonymity of the user location. This work extends the existing work on location privacy protection by extending the model with the Double Cloak Area (DCL-Ar) technique, where two layers of anonymization (among users and fog nodes) protect against both passive and active attacks from untrusted service providers. The aim of the study is to demonstrate the models' capability to enable reliable task execution and data sharing in spatial crowdsourcing platforms without much impact on location integrity. Model evaluation was embarked using C# in a simulation environment that mimics real-world smart city applications. The results showed that the proposed DCL-Ar significantly enhanced the cache hit rates and minimized response latency while ensuring maximum location entropy. Therefore, this study has contributed significantly to the field by bridging a significant gap between secure location-aware systems and participatory sensing, towards ensuring both service accuracy and data reliability.** (*Abstract*)

Keywords: Privacy, crowdsourced, Location-Based Services (LBS), Cloak Area, anonymization

I. INTRODUCTION

In recent years, the proliferation of Location-Based Services (LBS) and crowdsourcing platforms has revolutionized the way users interact with their environment. LBS applications are now embedded in almost every aspect of daily life, from navigation and ride-sharing to fitness tracking and social networking. Simultaneously, crowdsourcing has emerged as a powerful model to harness collective intelligence, enabling users to contribute valuable data, insights, and services based on their geographic locations. This convergence of LBS and crowdsourcing, despite the significant benefits, comes with critical privacy concerns, especially related to sensitive location information disclosure; these privacy-related concerns are due to the continuous requirement of many LBS applications to access users' location in real-time, which, when merged with crowdsourcing (where users voluntarily submit data), significantly increases risks of location-based privacy breaches. The performance of the conventional anonymization methods in protecting users against advanced re-identification attacks, trajectory linking, and malicious service providers who may exploit location data for profiling or surveillance has remained unsatisfactory.

In spatial crowdsourcing where users' selection is proximity or mobility-based, dynamic task assignment further complicates the issue of privacy protection as users are demanded to reveal their exact or approximate locations, making them vulnerable to attacks or misuse.

This motivates the need for integrating trust-preserving mechanisms into participatory LBS systems. These mechanisms must ensure that:

- User participation is anonymous or pseudonymous.
- Task allocation and data aggregation do not require precise location disclosure.
- Participants feel safe and confident while contributing data, which in turn encourages greater engagement and improves system performance.

Therefore, this study proposes advanced architectures such as the Double Cloak Area and leverages decentralized models such as fog computing to ensure a trade-off between utility and privacy, thereby enabling scalable, secure, and privacy-aware spatial crowdsourcing platforms.

II. LITERATURE REVIEW

The convergence of location-based services (LBS) and crowdsourcing has attracted much interest in recent years, especially because of its capacity for real-time geographic data acquisition and participatory sensing. Estellés-Arolas et al. [1] established a fundamental concept and typology of crowdsourcing, whereas subsequent studies have concentrated on the privacy and trust issues arising from user-contributed data containing sensitive location information. The presence of significant vulnerabilities in spatial crowdsourcing systems during task allocation and result submission phases was detected [2] [3], in which user location may be made known to dangerous requesters or extracted through trajectory linkage. To address these issues, various privacy-preserving strategies

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 06 , June - 2026

have been proposed; for instance, a k-anonymity-based approach for location obfuscation has been proposed by Wang et al. [4], whereas Zhao et al. developed strategies for privacy-preserving distance computation to safeguard geographical queries in mobile sensing contexts. Most often, these centralized approaches suffer from latency, scalability, and singular points of vulnerability-related problems. Conversely, the Double Cloak Area (DCL-Ar) architecture proposed by Bahbouh et al. [5] implements a dual-layer anonymization system utilizing fog computing; DCL-Ar not only improves response time through local cache-assisted fog nodes, but also enhances location uncertainty by anonymizing both at the peer user level and across fog node layers, outperforming basic k-anonymity and differential privacy models in dynamic environments. The importance of trust in crowdsourced systems has also been emphasized; for instance, a trustability-based dynamic active learning system for crowdsourced emotional audio classification has been introduced by Hantke et al. [6] to illustrate the influence of annotator reliability on model quality; this principle is equally applicable to location data providers. The role of crowdsourced data in sensitive applications, such as COVID-19 detection from cough audio, has been investigated by Chang et al. [7]; the authors introduced a deep learning framework employing ensemble learning and uncertainty estimates to manage noisy, imbalanced crowdsourced datasets. This underscores the necessity for stringent privacy, trust, and data quality measures when handling publicly supplied geographic or biometric data. The results indicate that safeguarding user privacy, guaranteeing the reliability of data sources, and facilitating real-time responses are crucial for building reliable, scalable, and privacy-aware crowdsourcing location-based services systems.

III. METHODOLOGY

To achieve the ultimate goal of the application, the main structure of the system is built on several levels and identical algorithms that complement each other:

A. Algorithms

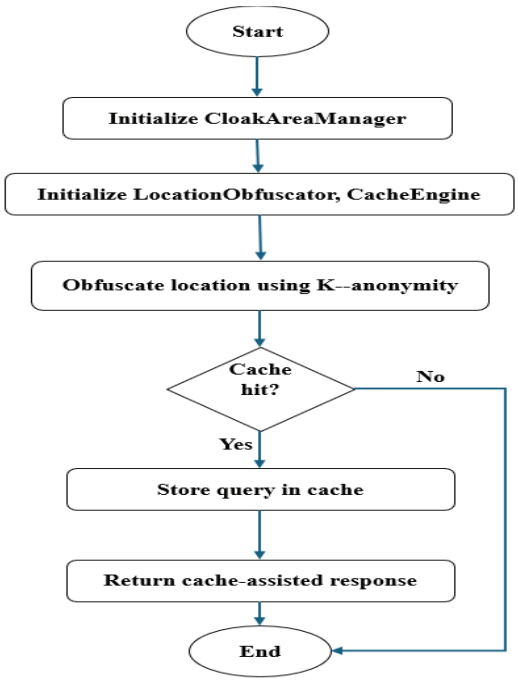
To implement the proposal in this work, the system was designed to be accomplished with a set of algorithms that allow the user and the entity providing the location service to exchange Crowdsourced data without harm to either party; this is seen in the flowchart of the system (Figure 1) and flow-steps of the algorithm in Table.1.

In table 1 outlines the primary algorithmic steps in our privacy-preserving location-sharing system. Each step adds a layer of security or performance improvement:

Table 1. STEPS OF THE ALGORITHM

Step	Coordinates	Privacy Mechanism	Benefit
1	(x, y)	Input from user	Starts privacy workflow
2	(x1, y1)…	K-anonymity (CloakArea1)	Hides user in a crowd
3	($\bar{X}$, $\bar{Y}$)	Obfuscation (mean coord)	Masks original location
4	($\bar{X}$, $\bar{Y}$)	Fog cache check	Reduces latency
5	($\bar{X}$, $\bar{Y}$)	Caching at fog layer	Efficient and local data reuse
6	—	Indirect response	Privacy preserved end-to-end

Fig. 1. Flowchart of the system



Step 1. User Input: The user submits location coordinates (x, y); this marks the beginning of the privacy workflow.

Step 2. Cloak Area 1 (User-Level Anonymity): Using the k-anonymity method, the system replaces the user's coordinates with a cloaked region containing at least k users (x1, y1)…. This prevents unique user identification.

Step 3. Obfuscation: The actual location is masked by blending it into a crowd; this is achieved by averaging the coordinates ($\bar{X}$, $\bar{Y}$).

Step 4. Fog Cache Check: The fog node checks if relevant data is already available to reduce redundant queries.

Step 5. Caching: The anonymized request is locally stored at the fog layer for reuse, thereby improving efficiency.

Step 6. Indirect Response: The user gets a response without disclosing exact geographical coordinates, hence maintaining end-to-end privacy.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 06 , June - 2026

B. Architecture

There are two levels of anonymization within the core of the system:

- Anonymizer1: Works at the user level to hide users within a group (Cloak Area 1).
- Anonymizer2: Operates at the fog node level, adding another layer of anonymity between fog nodes (Cloak Area 2).

The two layers of Cloak Area enhance the initial results of stored and queried data and provide layers of protection with the benefits of strong two-tier protection for location data and reducing reliance on central cloud servers, which enables the system to improve real-time processing and bring computing closer to the user. Client Node Interaction via Secure Channels: the users (via smartphones or other devices) connect to the fog layer through encrypted and secure channels (e.g., TLS/SSL). The core of the system [8] communicates with the Mobile Client Node Interaction via Secure Channels, and the users (via smartphones or other devices) connect to the fog layer through encrypted and secure channels (e.g., TLS/SSL).

Also, Figure 2 illustrates the overall system architecture implemented using C# and .NET Core for privacy-preserving location-based crowdsourcing; the design encompasses online and mobile client applications that securely interface with a fog-based backend. The fundamental logic is divided into four primary components: CloakAreaManager manages k-anonymity grouping; FogNodeController orchestrates data flow and anonymization layers; LocationObfuscator executes coordinate distortion; and CacheEngine enhances response time via local caching. User-sourced data (latitude, longitude, and anonymity level) is processed through these modules. The system supports real-time simulation using multithreading and socket communication for 100,000 parallel user agents. This modular and scalable design ensures privacy protection at both the user and fog levels. The implementation allows flexible task assignment and secure query handling without revealing precise user locations. The diagram also highlights the interaction between users and fog nodes in a distributed architecture.

***c.** Implementation*

The system was created in the language C# and with interfaces using the technology of ASP.NET Core MVC.

Platform:

1. Web by Asp.NET Core MVC
2. Phone App by Xamarin .NET
3. Api by Asp.NET Core

C# and the .NET ecosystem exhibit strong support for modularity, security, and cross-platform development, and as such, were selected for the implementation of the proposed privacy-preserving crowdsourced LBS system. Specifically, ASP.NET Core MVC performs well and offers inherent support for RESTful APIs; it is also compatible with contemporary UI frameworks, and as such, was selected as the web interface. Xamarin .NET enabled the development of the mobile application; it also allows for native Android/iOS creation from a unified C# codebase, which minimizes redundancy and reduces maintenance-related cost.

ASP.NET Core Web API usage enhances scalability and reusability by providing a clear distinction between frontend and backend logic; C# provides the needed multithreading capabilities that aids appropriate mimicking of real-time fog collaboration and management of the 100,000 concurrent user agents; .NET platform contains numerous frameworks needed for implementation of the proposed fog-based anonymization paradigm, such as the for networking, encryption, and data caching framework.

Microsoft's ecosystem is an appropriate suit for research prototype deployment; it is a thoroughly documented suite that enables uniform implementation across desktop, online, and mobile platforms; it also provides the necessary tools for testing and integration of research prototypes. The presence of the ASP.NET Core, Xamarin, and API layers in the proposed design ensures a flexible, efficient, and privacy-aware

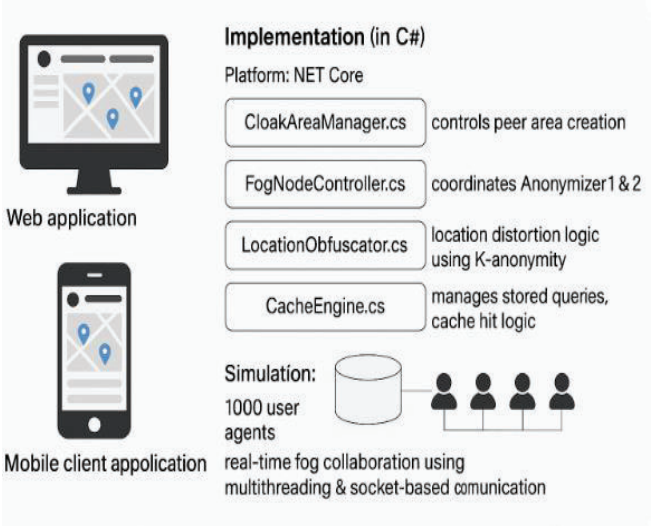


Fig. 2. Architecture of the system

framework.

Modules:

1. CloakAreaManager.cs: controls peer area creation.
2. FogNodeController.cs: coordinates Anonymizer1 & 2.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 06 , June - 2026

3. LocationObfuscator.cs: location distortion logic using K-anonymity.
4. CacheEngine.cs: manages stored queries, cache hit logic.

The modularization of the proposed system was aimed at ensuring scalability, maintainability, and clear separation of concerns; peer groups are generated by the CloakAreaManager.cs module based on spatial proximity; this ensures k-anonymity through dynamic selection of the neighboring users. FogNodeController.cs serves as the core coordinator, managing interactions between anonymization layers (Anonymizer1 and Anonymizer2) and directing workflow across modules. LocationObfuscator.cs distorts locations by employing statistical aggregation to conceal actual coordinates and enhance entropy; CacheEngine.cs enhances system performance by caching and reusing query responses, thereby minimizing redundant processing and response duration.

Together, these modules represent a tiered, privacy-focused approach that includes user-level anonymization and edge-level query optimization; this stratification allows for independent testing, troubleshooting, and improvements for each functionality. This also complies with established guidelines for distributed systems, where modularity increases reliability. The design is modularized to facilitate the future incorporation of encryption, access control, or trust management components. This framework provides robust privacy assurances and better system performance.

Simulation: A total of 100,000 user agents with real-time fog collaboration using multi-threading and socket-based communication were involved in the simulation study.

The performance and scalability of the architecture were assessed using simulations with 100,000 user agents; this simulation allowed the testing of the fog-based anonymization and caching strategies under real workload conditions. With multithreading, it was possible to parallelize user interactions and replicate concurrent access patterns mostly encountered in highly populated settlements.

To mimic real-time, network-oriented messaging between mobile users and fog nodes in this study, socket-based communication was selected during the simulation to allow evaluation of the latency, cache efficiency, and system performance on a large scale. The simulation also evaluated the efficacy of the adopted modular approach under conditions of high inquiry density. The outcome of the simulation study showed a significant reduction in the need for backend communication due to the adopted fog collaboration and local

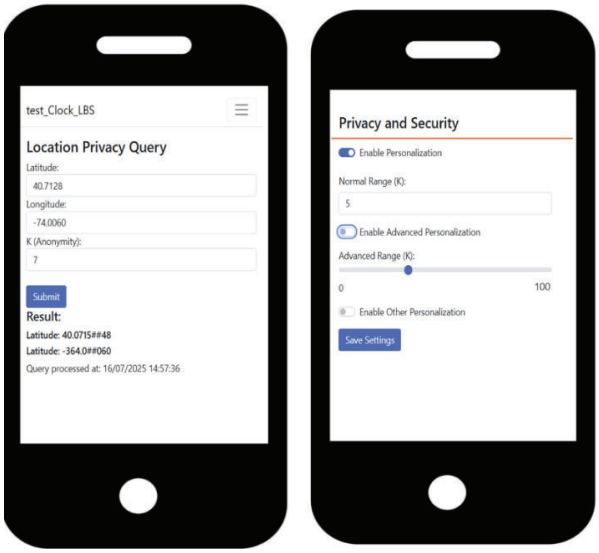


Fig. 3. Implementation test of the system

caching approach. The simulation further helped in identifying the potential bottlenecks, as well as enhancing the query routing and anonymization logic. By simulating such a high number of agents, the study empirically demonstrated that the proposed system met the demands of privacy-aware LBS in large-scale deployments.

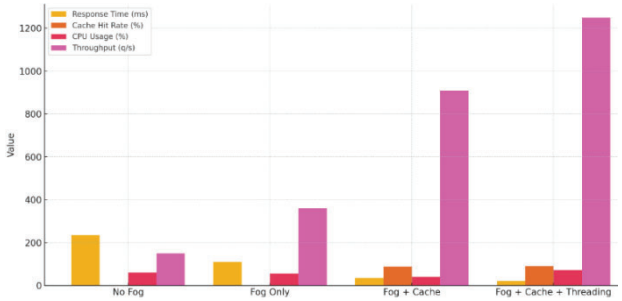


Fig. 4. Chart illustration of the implementation test and results

TABLE 2: IMPLEMENTATION TEST AND RESULTS

Scenario	Avg Response Time (ms)	Cache Hit Rate (%)	CPU Usage (%)	Through-put (queries/sec)
Without Fog Collaboration	235	0%	60%	150
With Fog Collaboration (no cache)	110	0%	55%	360
With Fog + Cache Enabled	35	88%	40%	910
With Fog + Cache + Threading	22	91%	72%	1250

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 06 , June - 2026

IV. CONCLUSION AND FUTURE WORK

In conclusion, the proposed Double Cloak Area (DCL-Ar) architecture offers a novel and effective solution for protecting user location privacy in spatial crowdsourcing environments. By combining two layers of anonymization at both peer and fog node levels, it significantly enhances user anonymity without compromising system responsiveness; the architecture effectively balances the trade-offs between privacy, accuracy, and latency, rendering it suitable for real-time applications. The viability, scalability, and appropriateness of the proposed method for deployment in smart city infrastructure were demonstrated via implementation with C# using ASP.NET Core and a multithreaded fog node simulation. Furthermore, the incorporation of a cache engine enhances performance by minimizing redundant searches and facilitating localized response management; the system's modular structure allows easy extension and maintenance.

Some of the future works in this area might include the following:

i. The consideration of adaptive privacy restrictions using machine learning to allow for the modification of the anonymity levels per context, risk factors, or user behaviour.
ii. Blockchain-based trust rating could also be integrated into the system to guarantee integrity and transparency in crowdsourced contributions.
iii. Testing the applicability of the proposed method in other privacy-sensitive sectors, such as emergency services, mobile health monitoring, and disaster response systems, to improve its generalizability.
iv. Integrating DCL-Ar with crowdsourced audio sensing can provide privacy-preserving solutions in public health while maintaining user confidence and data integrity.

v. LIMITATIONS AND FUTURE PRIVACY RISKS

Some of the identified limitations of this work that must be considered in future studies include the following:

i. It is currently assumed that fog nodes function with semi-honest motives and do not consider situations that involve collusion between these nodes and malevolent service providers; future research should aim to investigate more resilient adversarial threat models to address these deficiencies [9], [10].
ii. The study currently relies on fixed k-anonymity, which might bring about inconsistency in privacy levels in sparse or dynamic contexts; it is therefore important to implement adaptive anonymization techniques that consider variables such as context, population density, or data sensitivity to boost real-world applicability [11], [12].
iii. The current state of the proposed system does not allow users to dynamically modify their privacy settings; it is therefore crucial to incorporate individualized privacy settings, such as context-aware customization, that will improve user-centric data protection [13].
iv. Local caching arguably improved efficiency, but if poorly encrypted or access-controlled, it may jeopardize the integrity of important information; attackers may gain access to fog node caches to decode prior user locations [14], [15].
v. While simulations with 100,000 agents exhibit scalability, the lack of real-world implementation in settings like smart cities or healthcare constrains the validation of the model's practical applicability and efficacy in actual scenarios.

ACKNOWLEDGEMENTS

The authors would like to thank the University of Misan, Imam Al-Kadhum College, University of Al-Qadisiyah, and Imam Ja'far Al-Sadiq University for their academic support in this research.

REFERENCES

[1] Estellés-Arolas, E., & González-Ladrón-de-Guevara, F. (2012). Towards an integrated crowdsourcing definition. Springer.
[2] Ye, H., et al. (2019). Toward location privacy protection in spatial crowdsourcing. International Journal of Distributed Sensor Networks.
[3] Alharthi, R., et al. (2018). Location Privacy Challenges in Spatial Crowdsourcing. IEEE EIT.
[4] Wang, S., et al. (2020). Location Privacy-Preserving Distance Computation. IEEE Internet of Things Journal.
[5] Bahbouh, M., et al. (2023). Double Cloak Area Approach for Preserving Privacy and Reliability of Crowdsourcing Data. IEEE Access.
[6] Hantke, S., et al. (2018). Trustability-Based Dynamic Active Learning for Crowdsourced Labeling of Emotional Audio Data. IEEE Access.
[7] Chang, J., et al. (2022). UFRC: Unified Framework for Reliable COVID-19 Detection on Crowdsourced Cough Audio. IEEE EMBC.
[8] A. C. Jasim, I. A. Hassoon and N. Tapus, "Cloud: privacy For Locations Based-services' through Access Control with dynamic multi-level policy," 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT), Paris, France, 2019.
[9] M. Tang, S. Yu, Y. Yang, and K. Ren, "Privacy-preserving fog computing: A state-of-the-art survey," *Sensors*, vol. 21, no. 24, pp. 1–25, 2021.
[10] S. Ullah, M. Asim, M. F. Zafar, and H. Song, "Security and privacy in fog computing: Challenges, solutions and future directions," *Electronics*, vol. 10, no. 10, p. 1171, 2021.
[11] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.
[12] C. Dwork, "Differential privacy," in Automata, Languages and Programming, ICALP 2006, Springer, pp. 1–12.
[13] C. Hoepman, "Privacy design strategies," in IFIP International Information Security Conference, Springer, 2014, pp. 446–459.
[14] A. Sabouri and E. Saboori, "Privacy-preserving cache management in mobile edge computing," arXiv preprint arXiv:2012.03165, 2020.
[15] N. Zhang et al., "Smart cities and privacy: A review," IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2482–2506, 2019.