A Survey on Location Privacy-Preserving Mechanisms in Mobile Crowdsourcing

Arwa Bashanfar[1], Eman Al-Zahrani[2], Maram Alutebei[3],
Wejdan Aljagthami[4], Suhari Alshehri[5]
Information Technology Department
Faculty of Computing and Information Technology
King Abdulaziz University
Jeddah, Saudi Arabia

***Abstract*—Mobile Crowdsourcing (MCS) surfaced as a new affluent method for data collection and processing as a result of the boom of sensor-rich mobile devices popularity. MCS still has room for improvement, particularly in protecting workers' private information such as location. Therefore, the installation of privacy-preserving mechanisms that insulate sensitive information and prevent attackers from obtaining information is a necessity. In this paper, we discuss location privacy threats and analyze some recently proposed mechanisms that targeted location privacy in mobile crowdsourcing. Finally, we compare and evaluate these mechanisms according to specific criteria that we define in this paper.**

***Keywords*—Mobile crowdsourcing; privacy; security; location privacy-preserving**

I. Introduction

Crowdsourcing has been an evolutionary concept since 2006 when it was first introduced by Jeff Howe [1]. It can be viewed as an open call for problem-solving through the engagement of large groups of people. Those groups could be chosen according to specific criteria, for example, their job title, medical history, or their salary. Crowdsourcing enabled adequate and effective data collection solution [2].

With technology revolutions, smartphones are equipped with various kinds of sensors and thus became more powerful. This led to emerging smartphones into crowdsourcing and elevated participating level in performing various tasks to a new era. Specifically, in 2012 is when the concept of Mobile crowdsourcing (MCS) was introduced [3]. This concept is based on joining computers and humans to make crowdsourcing even more efficient with data uniquely generated and collected from multiple smartphones [4].

As a result of the growth of sensors and mobile devices popularity, Mobile Crowdsourcing (MCS) applications surfaced as a new affluent method for data collection. In MCS systems, exposure to security and privacy threats exists due to the human involvement and mobility characteristic. MCS systems still have room for improvement, particularly in protecting task and workers' private information such as identities and locations [5]. Therefore, a lot of researches proposed privacy-preserving mechanisms that insulate sensitive information and prevent attackers from obtaining access to private information.

With the growth of the MCS models, there have been other platforms extended from the MCS concept like Spatial Crowdsourcing (SC). The main characteristic of SC is that workers must be present in a specific location to accomplish the spatial tasks [6]. In this paper, we use both terms exchangeably.

This paper focuses on the issue of preserving the privacy of crowdworkers to increase their participation in fulfilling various tasks. The main goal of this paper is to review the state-of-the-art research on privacy-preserving techniques in spatial crowdsourcing.

This paper is organized as follows: Section II introduces the MCS model and its main entities. Section III presents location-based privacy threats. Section IV reviews recent approaches that have been proposed as solutions for location privacy-preserving. Section V discusses and compares the previously reviewed solutions. Section VI concludes the paper.

II. The MCS Model

MCS systems compose of four main entities: an end user or requester, a service provider (SP), crowdworkers, and a task. These entities interact with each other in real time. These four main components and the workflow among them are illustrated in Fig. 1. In the following paragraphs, we describe these four entities and the interaction among them.

- *End user (Requester)*
  An end user is the owner of the task that he/she wishes to be performed by certain people, and for that, he/she announces his/her tasks through a service provider and receives the response through the service provider.
- *Service Provider (SP)*
  SP acts as a trusted mediator between requesters and workers. It provides the platform for crowdsourcing services in which a SP receives a request from a requester and assigns it to the proper workers according to specific criteria, for example, workers' locations, task requirements, etc. An SP is responsible for incentivizing workers to participate in performing tasks through rewording system or any other mechanism that guarantees the willingness of workers to complete a task with correct responses.
- *Crowdworkers*
  Crowdworkers are the participants who perform requesters' tasks by providing the SP with responses to the corresponding published tasks. As a reward, workers sometimes get paid for successful participations.

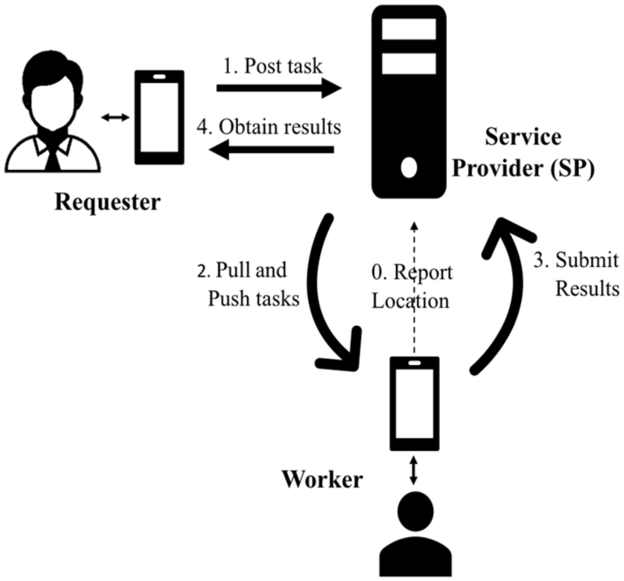

Fig. 1. The MCS Model.

- *Task*
  Task is the job posted by a requester to be accomplished by crowdworkers. This task may contain information about the requesters and their locations. Generally, task assignment has two modes. The first mode is called server-assigned tasks (SAT) and also known as push mode; it is when a task is assigned by a SP to workers. The second mode is called worker-selected tasks (WST) and also known as pull mode. In this mode, workers get to choose what task to perform.

III. Location-based Privacy Threats

There are three issues that are related to MCS: security, privacy, and trust. The MCS threats can be categorized based on these issues as follow:

1) Security threats such as eavesdropping, Sybil attacks, False Data Uploading, are caused by open wireless connections and the distribution and mobility of workers.
2) Privacy threats such as workers data disclosure, end users' personal information leakage and task privacy [7].
3) Trust threats such as worker trust which directly impact data trust.

In this paper, we mainly focus on location privacy-preserving. Before analyzing proposed mechanisms that targeted location privacy, we need to discuss the location-privacy attacks that threaten the crowdworkers locations.

Location-privacy attacks can be divided based on the task assignment mode into pull mode attacks and push mode attacks. The main difference between pull mode and push mode attacks is that in the pull mode, attackers try to identify the area range to find the exact worker's location. But in push mode, the attackers try to identify the worker's location from the location updates.

There are many attacks that can happen on pull mode such as task sampling attack, location homogeneity attack, and map matching attack [6]. The first attack is the task sampling attack in which the attackers link the location of participants to location-based tasks to know the location of a specific worker. The second attack is the location homogeneity attack, where the attacker may link some workers in the same region with the same sensitive attributes and disclose personal information like diseases, hobbies, etc. The last attack is map matching attack, in which attacker eliminates areas from the map such as lakes and rivers where it is impossible for the worker to exist in these areas, this will increases the attacker's chances of finding the worker's exact location.

On the other hand, there are many attacks that can happen on push mode such as task tracking attack, maximum movement boundary attack, and location inference attack. The first attack is the task tracking attack where the attacker tries to learn a pattern from the worker's location updates to disclose other information. The second attack is the maximum movement boundary attack in which the attacker computes maximum movement boundary to specify the worker's location at a specific time. The last attack is the location inference attack where the attacker uses background knowledge and workers' location to disclose workers' privacy.

IV. Location Privacy-Preserving Techniques

There are many research works and proposed techniques that have been conducted to address location privacy concerns. In some systems, workers must send their locations to the service provider to be used later in task assignments. As a result, workers become vulnerable to attacks such as eavesdropping. In contrast, other systems give the worker the authority to choose a task he/she wishes to perform. This means that the worker gets to explore information related to the published tasks. Therefore, proposed privacy-preserving techniques can be categorized into two categories based on the mode of task assignment: SAT and WST. These modes are described in Section II.

In the following paragraphs, we are going to review some recently proposed techniques and organize them based on their mode of task assignment.

A. SAT Mode

Alharthi et al. [8] proposed DCentroid, which is a novel framework for crowd workers location privacy in spatial crowdsourcing (SC). The DCentroid is designed to overcome the issues of the location privacy in SC by utilizing the dummy-based technique. The idea of the proposed mechanism is to hide the actual location of the crowd worker by creating dummy locations instead of the real location then sending these locations to the SC-server. As shown in Fig. 2, the structure of this framework consists of three components which are requester, crowd worker, and SC-server.

The workflow of the DCentroid framework acts as follows: Firstly, the crowd worker creates three dummy locations around the real location by using the Direct Dummy Algorithm. This algorithm generates sixteen directions of the crowd worker position to realize all possible dummies then it chooses randomly three dummy locations. After that, the algorithm removes the elected point to avoid duplicate selection of that location. This algorithm assumes that the closest point to the crowd worker location must not be further than three units and not less than one unit, in order to ensure the travel distance metrics and to protect the crowd workers privacy. Secondly, the crowd worker issues the dummy locations to the SC-server. Thirdly, the SC-server computes the range of these locations then computes the estimated distance from the crowd worker location to the requester task location by using the Standard Euclidean Distance. Finally, The SC-server issues the task to the closest crowd worker depending on the computed estimated distance.

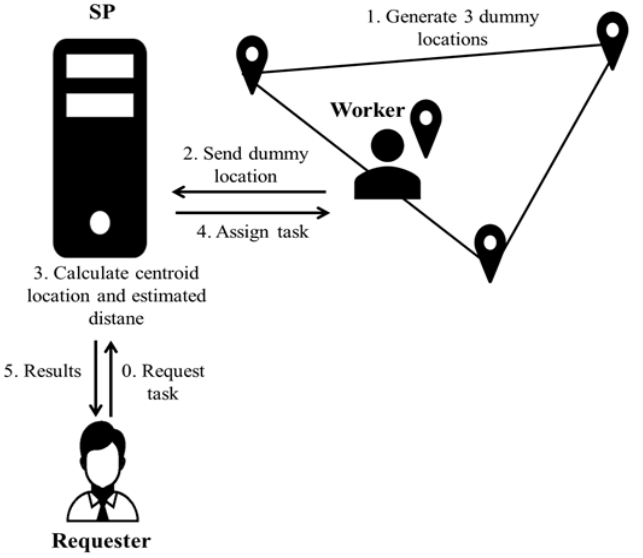
SP
1. Generate 3 dummy locations
Worker
2. Send dummy location
4. Assign task
3. Calculate centroid location and estimated distane
5. Results
0. Request task
Requester

Fig. 2. Dummy-based Location Privacy-preserving Technique [8].

Liu et al. proposed a location privacy framework in [9]. This framework uses homomorphic encryption scheme [10] to protect location privacy and secures index technique to save participants' locations. It contains six phases.

The first phase, Worker Registration, the crowd worker sings up to the registration server SR, encrypts its location using homomorphic encryption scheme and sends the encrypted location to SR. SR will index all workers' locations and store them in secure KD tree. The second phase, Task Submission, the requester submits task's location in the same encrypted way. The third phase, Distance Computation, the computation server SC computes the distance between the worker's location and the task's location without disclosing the real locations (by using homomorphic encryption concept). The fourth phase, Task Assignment, SC assigns the task to the closest worker according to the calculated distance. The fifth phase, Task Notification, SR notifies the worker about the task that he/she needs to perform. Finally in the last phase, in order to enable the worker to know the task location, the requester encrypts the task location with the worker public key and sends it to the worker. The overall framework is shown in Fig. 3. Therefore, This framework protects the location privacy in SC.

Zhu et al. proposed a location privacy scheme that applied the clustering algorithm in [11]. Thus, in this scheme, the distributed spatial clustering algorithm is executed by the workers in proactive and on-demand modes. The proactive mode has more rapid responses in comparison to the on-demand mode because the workers in proactive mode periodically run the clustering algorithm through peer-peer communication links within one hop. However, the proactive mode might result in the escalation of communication overhead. In on-demand mode, the server will broadcast the signal to the workers once it receives the spatial task to initialize the clustering algorithm. Afterward, the workers run the clustering algorithm. Clustering is initiated through every worker choosing a random number between 0 and 1. After that, the cluster head is selected if the random number is less than the threshold value. Therefore, the cluster head will broadcast an advertisement message to non-heads to join a cluster. If a non-head receives 2 or more advertisement messages from different cluster heads, he/she will choose the nearest cluster head and join that cluster. Finally, the head cluster knows all locations of the members in the cluster. After the clustering step, each cluster head sends the location of the virtual cluster center (VCC) which is calculated according to the cluster members' locations. The two-level task assignment algorithm is performed when a requester generates a spatial task as shown in Fig. 4. In the primary level assignment, the server assigns the task to the nearest cluster head to the spatial task's location using VCC location. After that, the secondary level assignment is performed by the chosen cluster head in the primary level assignment. Then, the head assigns the task to the nearest cluster member to the required location by task. Finally, the worker will move to the task's location and perform the task. Therefore, the workers' locations aren't disclosed to the server or to the requester.

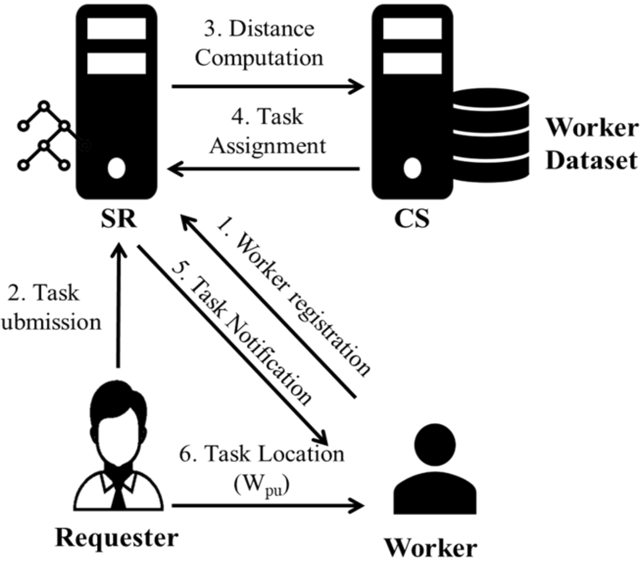

Fig. 3. Homomorphic Encryption Based Framework [9].

B. WST Mode:

Wang et al. [12] focused on two points, the first of which concerns about the auction algorithm while the second is about location privacy-preserving mechanism. Thus, the improved two-stage auction algorithm based on trust degree and privacy sensibility (TATP) is proposed to ensure the dynamics and fairness for the online incentive mechanism. Also, the $k-\varepsilon$ differential is proposed to protect the workers' location information. To encourage workers to participate in tasks and conduct truthfully, the first point is needed. Therefore, TATP is designed to determine winners in real time in contrast to the algorithms that existed previously.

In other words, TATP upgrades the traditional two-stage auction by increasing the truthfulness of auction and removing the injustice that occurred in the traditional two-stage algorithms, which rejected the first batch of workers. The first stage of TATP is the sample collection stage while the second stage is the contest stage that sets the bidding threshold in each transaction dynamically based on the first stage's result. On the other hand, to safeguard workers' location information, the combination between $k-anonymity$ and $\varepsilon-$ differential privacy preserving to produce $k-\varepsilon$ differential privacy-preserving is proposed.

In addition, Gaussian white noise is utilized to $\varepsilon-$ differential privacy-preserving. The $k-anonymity$ is characterized by the following properties: Spatial containment, Spatial resolution, Temporal containment, Temporal resolution, and Location k-anonymity. As a result of applying $k-anonymity$, attackers can't link location information to the correct worker. However, $k-anonymity$ doesn't cover the homogeneity attack dilemma. Therefore, the combination of $k-anonymity$ and $\varepsilon-$differential privacy preserving and the application of Gaussian white noise to $\varepsilon-$differential privacy-preserving is proposed. The relationship between ε and noise is an inverse relationship. So, if ε is smaller, the added noise increases which means more location privacy-preserving is achieved.

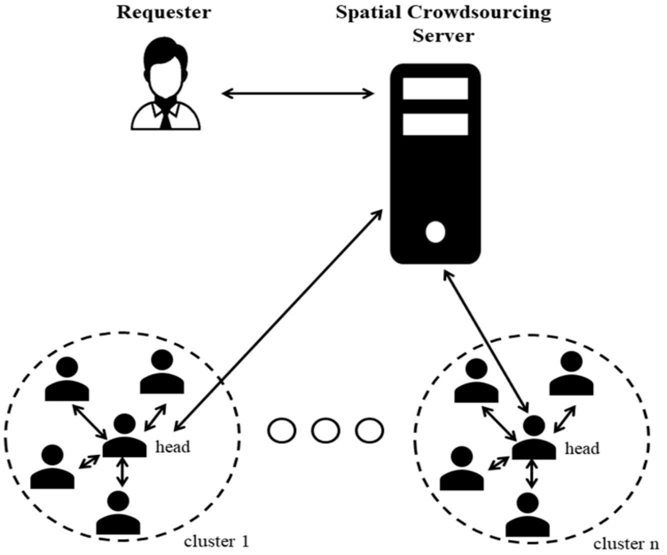

Fig. 4. Cluster-based Location Privacy-preserving Technique [11].

In the end, the experiments are done to assess and evaluate the proposed mechanism efficiency by verifying the effectiveness of TAPT over comparative to other auction algorithms and verifying the effectiveness of $k-\varepsilon-$differential privacy-preserving.

Jia et al. [13] focused on the mechanisms of user motivation to accomplish the tasks in MCS. The most important feature to users in MCS is the location privacy perspective, and for that reason, users may restrain participation and may input incorrect information to protect their privacy. In order to overcome those issues, the mixed incentive mechanism was proposed which consists of privacy protection and blockchain.

The network structure of blockchain is divided into three components which are intelligence crowd sensing networks, confusion mechanism, and blockchain. The crowd sensing network has two kinds of nodes, an ordinary user node which contains user information while the second is a miner node. The main role of the miner node is to produce a new block space. On another hand, the server in the crowd sensing network has two functions to do: issuing task information and get the sensing data from the blockchain. The confusion mechanism is developed to protect the crowd worker information by encoding the node information based on the Confusion Mechanism Encode Algorithm (CMA-E) and the Confusion Mechanism Decode Algorithm (CMA-D). The CMA-E encoded each part of user node information including longitude, latitude, age, gender, hobby, and Occupation.

The last component is the blockchain where the main purpose is to protect the user information from tampering. The structure of blockchain was changed by building the Merkle Tree and Currency Allocation using double-SHA256 hash algorithm. Fig. 5 illustrates the components and transactions of the proposed mechanism, which acts as follows: First, the server publishes a sensing task then the user in the crowd sensing network takes the sensing task. After that the sensing data enters the confusion mechanism as blocks where in each block there are nine user nodes and one mine node. second, the blockchain stores user information and provides the virtual coin to the user as a reward to motivate him to participate then the user can replace it with cash. Finally, the server retrieves the user information from the blockchain.

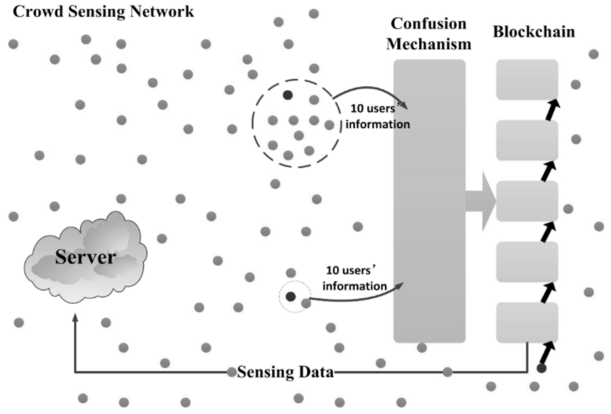

Fig. 5. Blockchain-based Incentive Framework [13].

Alamer et al. presented Location privacy-Aware Task rEcommendation framework (LATE) [14] and designed it to preserve workers' location during task recommendation in spatial crowdsourcing. The LATE framework can accomplish privacy-preserving location-matching between the workers' locations and the spatial task's geocast area by using Lagrange Interpolating Polynomials. Moreover, The LATE framework consists of four components: SC-server, trust management server (TMS), customers, and workers. TMS administers the trust level of workers to assist SC-server in its decision-making about the workers reliability to perform a task.

LATE is composed of four phases: Service Setup, Task Releasing, Task Recommendation, and Task Fulfillment. Firstly, the process initiates with the Service Setup phase, where the CS-server loads all services and establishes the public parameter. It also defines the region's hallmark, which assists in conducting the service geographic region for customers. Furthermore, worker generates a public-private key pair. Afterward, the certificate of the worker is issued from the certificate authority (CA). Moreover, TMS and customer choose a secret key, and calculate their public keys. Then, CA issues a certificate of TMS and customer. Additionally, TMS initializes its service for workers. Secondly, in the Task Releasing phase; when the customer covets using spatial crowdsourcing, the customer creates a task and defines some parameters of the task such as geocast area, expiration time and other attributes (e.g. reporting periods, benefits) illustrated in Cont. To prevent disclosure of geocast area, the customer produces a series of encrypted points of interest. Moreover, it prevents the task's disclosure by encrypting Cont using CA's public key and encrypting the secret key using TMS's public key. Finally, it sends an encrypted task to CS-server. Thirdly, in the Task Recommendation phase; when the worker desires to be involved in activities, the worker interacts with TMS by sending his certificate. The SC-server takes encrypted spatial task and worker's location, then utilizes the matching algorithm to check the worker's location. If the worker is located in the same geocast area of the task, it gets the recommended task. Fourthly, in Task Fulfillment phase; when the worker executes the task and issues the crowdsourcing report, the crowdsourcing report must be protected by the customer's public key.

In the end, the security of LATE is proved by demonstrating that the attacker can't disclose the workers' locations and the spatial task's geocast area since they are encrypted. Also, the efficiency of LATE and practicality in computation and communication overhead are proved.

V. Discussion and Analysis

For the purpose of differentiating between the previously discussed techniques, we conducted a comparison using five main criteria: (1) Location privacy of worker and task, (2) the used technique in implementing each mechanism, (3) the overhead caused by applying every solution on a system, (4) the existence of third party, and (5) task assignment mode. For further illustration, we explain each one as follows:

- Used Technique: There are different methodologies used to hide or preserve the location privacy: Clustering, K-anonymity with $\varepsilon-$differential, Encryption, Blockchain, Dummy based technique, and Homomorphic encryption.
- Location Privacy (Worker | Task): some mechanisms are interested in protecting the worker location or task location, and others are interested in preserving both worker and task's location. From the table we can see that most techniques protect the worker's location, on the other hand, only two technique protect both worker and task's location from disclosure, both of these used encryption-based methodologies.
- Overhead (computational | Communicational): the impact an approach has by applying it to a mobile crowdsourcing system, if it's going to add computational overhead (where there is a lot of value computing required) or communicational overhead (where there is a continuous exchange of messages over the network that might cause high network traffic). We can distinguish three out of the six techniques that encountered overhead. The clustering had both computational and communicational overhead due to the constant calculating of distance on the Proactive Mode, these calculations require more of messages exchange. Similar to clustering, the homomorphic encryption causes both overhead types which come from the frequent update of SKD tree every time the worker changes location which is a time-consuming process when there is a lot of workers. The Blockchain techniques are known for being expensive in computations and communication due to the whole blocks and transactions being transmitted to and validated by all nodes. The Blockchain embeds the consensus algorithm which causes a high computational overhead that would raise with the number of nodes in the network [15] [16].
- Trust Third Party: There is only one solution that will depend on the existence of a trusted third party, which is the encryption technique. The third party usually exists with the assumption that it can be trusted. However, the trust third party does not exist in the real world [17].
- Assign task mode type: As previously mentioned in Section II, assignment mode for a task is either server-assigned task mode (SAT) or worker-selected task mode (WST). From the table, we can observe that half of the techniques are using the SAT mode, and the other half use the WST mode.

Table I presents the summary of the comparison among existing work according the previously specified criteria.

By observing the assessment of the six techniques, we can notice the best technique in terms of the previous criteria for each attack type:

1) In the SAT mode: we think that the Dummy-based technique, with the addition of encrypting the worker's location and the task using homomorphic cryptography, is an optimal solution for preserving location privacy. The dummy technique is a lightweight solution that depends on the idea of calculating centroid that is not the exact location of the worker

TABLE I. SUMMARY OF THE COMPARISON BETWEEN EXISTING WORK.

		Proposed Techniques					
		[11]	[12]	[14]	[13]	[8]	[9]
Used Technique		Clustering	$k-anonymity$ and $\varepsilon-$differential	Encryption	BlockChain	Dummy Based Technique	Homomorphic Encryption
Location Privacy	Worker	✓	✓	✓	✓	✓	✓
	Task			✓			✓
Overhead	Computational	✓	–	–	✓	–	✓
	Communicational	✓	–	–	✓	–	✓
Trusted Third Party		No	No	Yes	No	No	No
Assign task mode		SAT	WST	WST	WST	SAT	SAT

but relatively close to the real data [12]. Adding homomorphic cryptography will add privacy to the task location.

2) In the WST mode: The approach of utilizing both k-Anonymity and $\varepsilon-$differential is a suitable technique to be used for privacy protection in systems that use WST mode. This solution has no overhead impact on the system, and doesn't rely on a trusted third party to implement the mechanism, which makes it worthy of trust.

VI. CONCLUSIONS

Mobile crowdsourcing involves exchanging of sensitive and personal information as a consequence of task sharing and performing. Therefore, protecting both task and worker privacy is essential to encourage workers' participation. In this paper, we discussed attacks related to location disclosure. Moreover, we reviewed recently proposed mechanisms that aim to preserve location privacy, and we compared them and discussed the results of the comparison. Based on the comparison result, we can conclude that for crowdsourcing systems that uses SAT mode, it is better to merge between the Dummy-based technique and homomorphic cryptography to achieve location privacy-preserving. For WST mode, applying k-Anonymity and $\varepsilon-$differential will increase the location privacy-preserving and the trustworthiness of the system. As for the limitation, when deciding the overhead for each approach, we were relaying on the existence of heavy calculations, time consumption, and number communications required. Though we classified overhead into computational and communicational, more clear metrics are needed to compare between different types of overhead.

REFERENCES

[1] A. I. Chittilappilly, L. Chen, and S. Amer-Yahia, "A survey of general-purpose crowdsourcing techniques," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 9, pp. 2246–2266, Sep. 2016.

[2] E. Estellés-Arolas, R. Navarro-Giner, and F. G.-L. de Guevara, "Crowdsourcing fundamentals: Definition and typology," in *Advances in Crowdsourcing*. Springer International Publishing, 2015, pp. 33–48.

[3] J. Phuttharak and S. W. Loke, "A review of mobile crowdsourcing architectures and challenges: Toward crowd-empowered internet-of-things," *IEEE Access*, vol. 7, pp. 304–324, 2019.

[4] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, and D. Zeinalipour-Yazti, "Crowdsourcing with smartphones," *IEEE Internet Computing*, vol. 16, no. 5, pp. 36–44, 2012.

[5] Y. Wang, X. Jia, Q. Jin, and J. Ma, "Mobile crowdsourcing: framework, challenges, and solutions," *Concurrency and Computation: Practice and experience*, vol. 29, no. 3, p. e3789, 2017.

[6] R. Alharthi, A. Banihani, A. Alzahrani, A. Alshehri, H. Alshahrani, H. Fu, A. Liu, and Y. Zhu, "Location privacy challenges in spatial crowdsourcing," in *2018 IEEE International Conference on Electro/Information Technology (EIT)*. IEEE, 2018, pp. 0564–0569.

[7] K. Yang, K. Zhang, J. Ren, and X. Shen, "Security and privacy in mobile crowdsourcing networks: challenges and opportunities," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 75–81, August 2015.

[8] R. Alharthi, E. Aloufi, A. Alqazzaz, I. Alrashdi, and M. Zohdy, "Dcentroid: Location privacy-preserving scheme in spatial crowdsourcing," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2019, pp. 0715–0720.

[9] B. Liu, L. Chen, X. Zhu, Y. Zhang, C. Zhang, and W. Qiu, "Protecting location privacy in spatial crowdsourcing using encrypted data," *Advances in Database Technology-EDBT*, 2017.

[10] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 1999, pp. 223–238.

[11] B. Zhu, S. Zhu, X. Liu, Y. Zhong, and H. Wu, "A novel location privacy preserving scheme for spatial crowdsourcing," in *2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE, 2016, pp. 34–37.

[12] Y. Wang, Z. Cai, X. Tong, Y. Gao, and G. Yin, "Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems," *Computer Networks*, vol. 135, pp. 32–43, 2018.

[13] B. Jia, T. Zhou, W. Li, Z. Liu, and J. Zhang, "A blockchain-based location privacy protection incentive mechanism in crowd sensing networks," *Sensors*, vol. 18, no. 11, p. 3894, 2018.

[14] A. Alamer, J. Ni, X. Lin, and X. Shen, "Location privacy-aware task recommendation for spatial crowdsourcing," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2017, pp. 1–6.

[15] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Lsb: A lightweight scalable blockchain for iot security and privacy," *CoRR*, vol. abs/1712.02969, 2017.

[16] A. Dorri, M. Steger, S. S. Kanhere, and R. Jurdak, "Blockchain: A distributed solution to automotive security and privacy," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 119–125, Dec 2017.

[17] Y. Shen, L. Huang, L. Li, X. Lu, S. Wang, and W. Yang, "Towards preserving worker location privacy in spatial crowdsourcing," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.