International Journal of
Geo-Information

Article

Achieving Differential Privacy Publishing of Location-Based Statistical Data Using Grid Clustering

Yan Yan [1,*], Zichao Sun [1], Adnan Mahmood [2], Fei Xu [1], Zhuoyue Dong [1] and Quan Z. Sheng [2]

[1] School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China; sunzc@lut.edu.cn (Z.S.); xufei@lut.edu.cn (F.X.); dongzy@lut.edu.cn (Z.D.)
[2] Faculty of Science and Engineering, School of Computing, Macquarie University, Sydney, NSW 2109, Australia; adnan.mahmood@mq.edu.au (A.M.); michael.sheng@mq.edu.au (Q.Z.S.)
* Correspondence: yanyan@lut.edu.cn

Abstract: Statistical partitioning and publishing is commonly used in location-based big data services to address queries such as the number of points of interest, available vehicles, traffic flows, infected patients, etc., within a certain range. Adding noise perturbation to the location-based statistical data according to the differential privacy model can reduce various risks caused by location privacy leakage while keeping the statistical characteristics of the published data. The traditional statistical partitioning and publishing methods realize the decomposition and indexing of 2D space from top to bottom. However, they can easily cause the over-partitioning or under-partitioning phenomenon, and therefore need multiple times of data scan. This paper proposes a grid clustering and differential privacy protection method for location-based statistical big data publishing scenarios. We implement location-based big data statistics in units of equal-sized grids and perform density classification on uniformly distributed grids by discrete wavelet transform. A bottom-up grid clustering algorithm is designed to perform on the blank and the uniform grids of the same density level based on neighborhood similarity. The Laplacian noise is incorporated into the clustering results according to the differential privacy model to form the published statistics. Experimental comparison of the real-world datasets manifests that the grid clustering and differential privacy publishing method proposed in this paper is superior to other existing partition publishing methods in terms of range querying accuracy and algorithm operating efficiency.

Keywords: statistical release of big data; location privacy; differential privacy; privacy spatial decomposition; grid clustering

check for updates

Citation: Yan, Y.; Sun, Z.; Mahmood, A.; Xu, F.; Dong, Z.; Sheng, Q.Z. Achieving Differential Privacy Publishing of Location-Based Statistical Data Using Grid Clustering. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 404. https://doi.org/10.3390/ijgi11070404

Academic Editor: Huayi Wu

Received: 28 March 2022
Accepted: 12 July 2022
Published: 15 July 2022

1. Introduction

With the rapid proliferation of emerging technologies such as Mobile Internet and the Internet of Things, numerous location-based big data services are becoming increasingly popular in various fields, including, but not limited to, population distribution statistics [1], urban planning and management [2], intelligent traffic scheduling [3], and disease epidemic control [4]. Particularly, in the current critical period of COVID-19 pandemic, a series of location-based big data applications (i.e., health codes, communication itinerary cards, and close contact self-examination procedures) provide great convenience for epidemic prevention and control in terms of epidemiological investigations, epidemic distribution and statistical analysis, personnel and material scheduling, etc.

Location information is highly correlated with personal privacy. Improper release or reverse reasoning analysis [5–7] of location-based big data statistics can easily lead to the disclosure of personal privacy of a user's specific location, movement trajectory, living habits, health status, hobbies, economic conditions, etc., and may even endanger a user's property and life [8]. Therefore, addressing the above-mentioned privacy protection issues during the publishing and utilization of location-based big data statistics remains an indispensable task and acts as a considerable bottleneck in the development of the big data industry.

The statistical partition and publishing method is widely used in location-based big data services. It can provide users with accurate and timely traffic statistics, facilitate people to plan reasonable travel times and routes, and obtain high-precision Location Based Services (LBS). When combined with the differential privacy model and once the noise disturbance has been incorporated in the statistics of location-based big data, it can further reduce the risk of users' privacy leakage while maintaining the statistical characteristics of the published data. During the COVID-19 pandemic, spatiotemporal location big data provided case statistics, distribution analysis, and trajectory estimation for pandemic prevention and control and which subsequently assisted in social resumption of work and production. Figure 1 is a typical example of statistical publishing of location-based big data, which depicts the spatial distribution of COVID-19 infectors in Hubei province on 31 March 2020 [9]. Figure 2 is the result of RAPPOR, an anonymous crowdsourced statistical data privacy-preserving technique based on differential privacy, wherein the true sample distribution is shown in black, and the light green depicts the estimated distribution based on the decoded Randomized Aggregatable Privacy-Preserving Ordinal Response (RAPPOR) reports [10]. As can be observed from Figure 2, when the number of report from users reaches one million, the statistical results reported according to RAPPOR closely traced the statistical distribution of the real data, which does not affect the availability of the published big data statistical results.

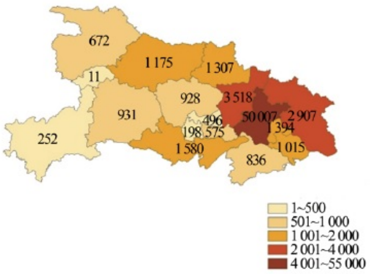


Figure 1. Spatial distribution of COVID-19.

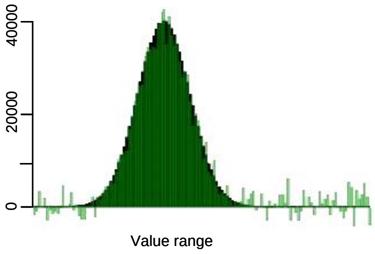


Figure 2. Normal distribution using RAPPOR.

The big data statistical publishing method based on the differential privacy model protects the privacy of each user under the premise of ensuring data availability by adding a certain random noise to the statistical publishing results. The partition structure of spatial location information and the introduction of differential privacy noise lead to errors between the published big data statistics' results and the real statistical results. This may reduce the availability of the released data. For example, a user queries the number of taxis available within 1 km of their own location and the noisy statistics' results returned by the LBS system is 11. However, the real situation is that there are only two taxis within a user's query range. Therefore, such statistically released results are likely to mislead users to wait for a long time, thereby resulting in an unsatisfactory LBS service experience. Similarly, let us suppose that, in accordance with the location information of confirmed COVID-19 patients in a certain area, the number of close contacts is about 12,000; nevertheless, the released statistical result after incorporating differential privacy noise turns out to be 8700. This can thus disrupt the supply of the medical aid equipment and other pandemic prevention materials as they would not be able to cater to the needs of this region, thereby delaying the timely treatment of patients. Therefore, improving the accuracy of location-based big data statistical results on the premise of ensuring location privacy is an important issue for big data services.

In order to reduce the impact of spatial partition structure and differential privacy noise on the availability of published data, this paper hereby proposes a bottom-up grid clustering algorithm and a corresponding differential privacy preserving data publishing algorithm. The range counting querying errors are reduced, and the availability of released statistical results is improved.

The rest of the paper is organized as follows: Section 2 reviews the state-of-the-art pertinent to private spatial decomposition methods. Section 3 defines the differential privacy model used for the publishing of location-based big data statistical information. Section 4 analyzes the problems of traditional privacy spatial decomposition and introduces the proposed grid clustering algorithm. Section 5 details the proposed statistical publishing and differential privacy protection algorithms. Finally, Section 6 reports a set of empirical studies, whereas Section 7 concludes the paper, laying out the limitations and future works of the research.

2. Related Work

Privacy spatial decomposition is one of the important means to realize the application of location-based big data statistical publishing. The statistical release and differential perturbation of location-based information within the partition and indexing area facilitate reducing the risk of users' location privacy. The traditional privacy spatial decomposition methods partition the 2D space from top to bottom and index the partition areas either according to location independent grid or tree structure or some location-based hierarchical structures.

The grid-based partition and indexing structure is the most common privacy decomposition methods, wherein the uniform grid (UG) method and the adaptive grid (AG) method are the typical representatives [11]. The former separates the 2D space into equal sized grids and calculates the statistical result in the units of grid. The latter performs a further grid partition on the dense areas based on the UG method. In [12], the authors employ contour maps to describe the distribution of location points in spatial crowdsourcing services and partition the spatial area into disjointed units to protect workers' location privacy. The partition method proposed in [13] firstly obtains the preliminary density-adaptive grid partition by judging the distribution of location-based information on the longitude and latitude directions, and, subsequently, the adaptive grid partition was performed on the dense grids to save unnecessary partition process and reduce publishing errors. In [14], the authors propose a three-layer adaptive grid partition method based on Bernoulli random sampling technology which employs the exponential mechanism and high-pass filtering technology to filter out grids smaller than the predefined threshold on the second layer of the partition structure but continues to partition the grid larger than the predefined threshold. The location privacy protection scheme proposed in [15] uses a three-layer adaptive grid structure and a fully pyramid grid algorithm based on differential privacy. In [16], the authors use the statistical information of particle distribution to extract the optimal spatial decomposition so that each unit contains particles with uniform spatial distribution. A common problem with the grid-based partition method method is difficult to balance the noise error with the uniform assumption error, which affects the accuracy of the published statistical results.

The tree structure has better hierarchical characteristics and can provide more convenient spatial range querying services. The Quad-opt algorithm [17] separates the 2D space using a complete quad-tree structure and designs a post-adjustment method which improves the range counting querying accuracy to a certain extent. However, the complete quadtree structure does not consider the distributions of data, thereby resulting in a large uniform assumption error. In [18], the complete quadtree partition results are adjusted and merged bottom-up in accordance with the uniformity conditions in order to reduce the uniformity assumption error. In [19], the authors separate the entire area into four sub-units of different sizes and recursively invoke the same partition process until a reasonable spatial structure is obtained. In [20], the authors propose an Unbalanced quadtree partition method based on regional uniformity and which traverses the subtree according to the depth-first strategy and adaptively carry out partition according to the uniformity. However, tree-based partition methods need more recursive operations, and, therefore, the execution efficiency of the partition and publishing algorithm is dragged down. Some privacy spatial decomposition methods organically combine grid-based and tree-based structures to form hybrid partition structures [21,22], thereby improving the accuracy of range counting querying services.

Clustering analysis plays an important role in data mining process for a long time. Cluster data records that are spatially close to each other in the same area not only facilitates improving the accuracy of statistically released data but also protects the location information corresponding to a single record. This kind of grid-based or density-based clustering algorithm [23–27] can be used to classify data of any shape. Since it only requires less calculations pertinent to the distance between a small number of grid nodes, the clustering algorithm has a high level of efficiency. In [28], the authors propose a location-based big data clustering algorithm based on density and grid partition. The proposed cell distance analysis model greatly reduced the distance calculation and the traversal of the location points. In [29], the authors propose an online data stream clustering method based on density grids. The grid-based method is employed to reduce the number of calls of the distance function, thereby improving the clustering quality. In [30], the authors propose a strongly connected grid clustering algorithm for the parallel processing of clustering based on MapReduce. Grid clusters with strong connectivity are merged by calculating the connectivity weight matrix so as to realize efficient clustering of location data. With the improvement of machine learning methods, high-performance grid clustering and density clustering algorithms are gradually increasing. Combining the above clustering algorithms with the publishing of location-based statistical information can improve the availability of published data on the basis of making full use of the distribution characteristics of location-based data. Research in this area has a very broad application space.

3. Differential Privacy

Definition 1 (ϵ-Differential Privacy [31,32])**.** *For a pair of neighboring datasets T_1 and T_2 with $\parallel T_1 - T_2 \parallel_1 \leq 1$, a randomized algorithm K with domain N is ϵ-differentially private if, for all $S \subseteq Range(K)$, there is:*

$$P_r[K(T_1) \in S] \leq e^{\epsilon} \times P_r[K(T_2) \in S] \tag{1}$$

The parameter ϵ in Formula (1) is referred to as the privacy budget. The smaller the value of ϵ, the higher is the degree of privacy protection provided by the algorithm K. Even if an attacker obtains all the data except for a specific target record, the attacker still cannot determine whether records corresponding to a target exist in the original dataset, thereby realizing privacy protection.

Definition 2 (Sensitivity [33])**.** *For a given query mapping function f, the sensitivity $\triangle f$ is defined as the maximum L_1-norm distance between the output of the query mapping function on neighboring datasets T_1 and T_2:*

$$\triangle f = \max_{T_1, T_2} \parallel f(T_1) - f(T_2) \parallel_1 \tag{2}$$

Definition 3 (The Laplace Mechanism [33])**.** *For numeric queries, the Laplace mechanism achieves differential privacy protection by incorporating a small amount of independent noise to* *the output of the query mapping function f. Let $f(T)$ represent the result obtained by the query mapping function f on the original dataset T; then, the query result returned by the Laplace mechanism can be expressed as $K(T) = f(T) + \eta$, wherein η is a continuous random variable satisfying the Laplace distribution (centered at 0) with scale b, and its probability density function can be expressed as:*

$$P_r[\eta = x] = \frac{1}{2b}e^{-\frac{|x|}{b}} \tag{3}$$

Combined with the definition of sensitivity, the incorporated independent noise satisfies a zero-mean Laplace distribution of magnitude $b = \frac{\triangle f}{\epsilon}$.

Theorem 1 (Serial Combination Characteristics [34])**.** *For a set of randomized algorithms $\{K_1, K_2, \ldots, K_n\}$ on the same dataset T, wherein K_i is an ϵ_i-differentially private algorithm $(1 \le i \le n)$. Then, their combination $\{K_1, K_2, \ldots, K_n\}$ is $\sum_{i=1}^{n} \epsilon_i$ differentially private.*

Theorem 2 (Parallel Combination Features [34])**.** *If the dataset T is composed of a set of independent and disjoint subsets $\{T_1, T_2, \ldots, T_n\}$, a set of randomized algorithms $\{K_1, K_2, \ldots, K_n\}$ are ϵ_i differentially private, respectively; then, their combination $\{K_1, K_2, \ldots, K_n\}$ is $max\{\epsilon_i\}$ differentially private.*

Definition 4 (Noise Error)**.** *For any querying range Q, the error between the released statistical results and the original statistical results after privacy protection processing is referred to as the noise error:*

$$Noise\_Error(Q) = |Count(Q) - Count^*(Q)| \tag{4}$$

wherein $Count(Q)$ represents the statistical result of the original location-based information within the querying range Q, and $Count^*(Q)$ manifests the statistical result of the published data within the same area.

Definition 5 (Uniform Assumption Error)**.** *Since the 2D space cannot be accurately partitioned into a single point, the partition and publishing algorithms often assume that the location-based information within the smallest partition area is uniformly distributed. The range counting querying error transpired as a result of the uniform assumption estimate is called the uniform assumption error:*

$$Uni\_Error(Q) = |\sum_{i=1}^{m} r_i \cdot Count(P_i) - Count(Q)| \tag{5}$$

wherein P_i $(i = 1, 2, \ldots, m)$ represents the partition area that intersects with the querying range Q and satisfies $\underset{1 \le i,j \le m \wedge i \ne j}{P_i \cap P_j} = \emptyset$. r_i $(i = 1, 2, \ldots, m)$ is the ratio of the intersection of the querying range Q and the partition area, and $Count(Q)$ represents the statistical result of the original location-based information within the querying range.

4. The Grid Clustering Algorithm for Location-Based Big Data Statistical Information

4.1. The Problems of Traditional Privacy Spatial Decomposition

Unreasonable partition structure is likely to increase the publishing errors for location-based big data statistical information. The errors mainly include differential privacy noise error and uniform assumption error. If the local distribution of location-based big data has the sparse characteristic as shown in Figure 3a, the accurate statistical value in the querying range Q_1, i.e., the red dotted box is 1. The returned query result after incorporating differential privacy noise according to Definition 3 is about $(3 + 2.6) \times \frac{4}{9} \approx 2.5$ and which is closer to the real statistical result. The partition and differential privacy protection result of the local distribution of location-based big data are portrayed in Figure 3b. The returned query result within the same range Q_1 is about $1 + 6.3 = 7.3$ (where 6.3 is the noise error). From this example, we can observe that, if a partition structure possesses a large number of empty nodes, i.e., grids with no location-based information, incorporating differential privacy noise introduces large noise error and reduces the querying accuracy of the published results.

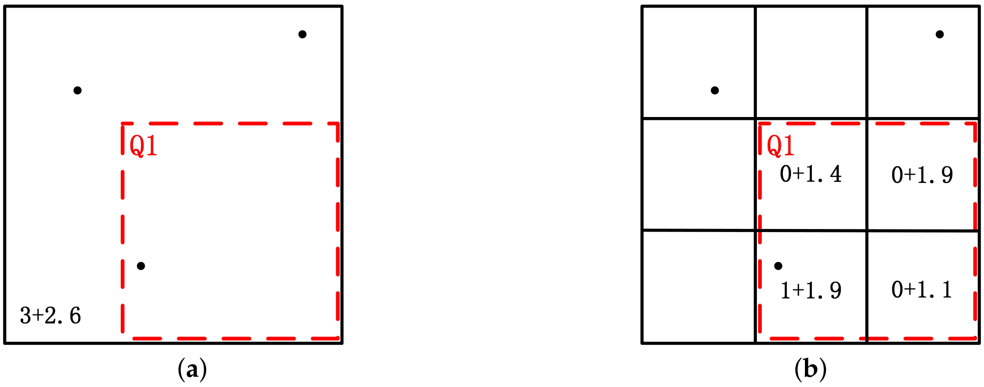


Figure 3. The relationship of sparse data distribution and partition structure. (**a**) example of sparse distribution; (**b**) corresponding partition and range query.

If the local distribution of location-based big data has the uniform characteristic as depicted in Figure 4a, the accurate statistical value in the querying range Q_2, i.e., the blue dotted box, is 16. The returned query result after incorporating differential privacy noise is about $(148 - 7.6) \times \frac{1}{9} \approx 15.6$ and which is also closer to the real statistical results. However, for the partition and differential privacy protection result of the corresponding distribution portrayed in Figure 4b, the returned query result within the same range Q_2 is $16 - 6.9 = 9.1$ (where 6.9 is the noise error). Therefore, the over-partitioning regions with relatively uniform spatial distribution also leads to a decrease in the precision of range counting queries.

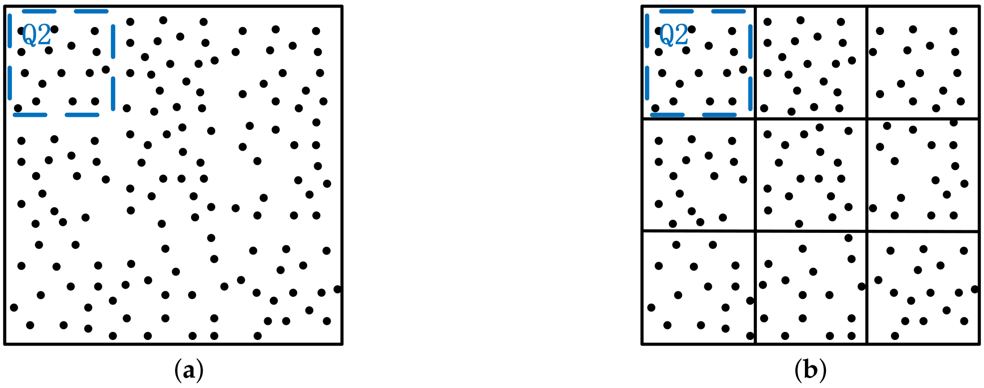


Figure 4. The relationship of uniform data distribution and partition structure. (**a**) example of uniform distribution; (**b**) corresponding partition and range query.

If the local distribution of location-based big data has non-uniform characteristics as portrayed in Figure 5a, the accurate statistical value in querying range Q_3, i.e., the green dotted box, is 3, and the returned query result after incorporating differential privacy noise is $(101 + 11.6) \times \frac{1}{9} \approx 12.5$. The publishing error between the released and the real statistical result is relatively high. However, for the partition and differential privacy protection result of the corresponding distribution depicted in Figure 5b, the returned query result within the same range Q_3 is $3 + 1.1 = 4.1$ (where 1.1 is the noise error). It can be observed that, for areas with non-uniform spatial distribution, under-partitioning causes larger uniform assumption error and affect the accuracy of range counting queries.

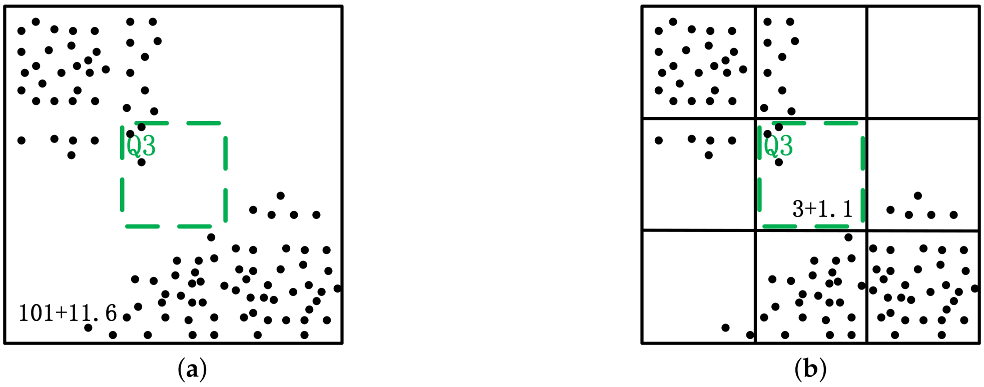


Figure 5. The relationship of non-uniform data distribution and partition structure. (**a**) example of non-uniform distribution; (**b**) corresponding partition and range query.

4.2. Proposed Method

In order to overcome the above-mentioned problems during privacy spatial decomposition, we propose a grid clustering algorithm for location-based big data statistical information. Firstly, the 2D space covered by the location information is partitioned into uniform grids in accordance with the minimum size of the range counting querying service of location-based big data. The number of location points in each grid is counted and used as the density of the grid. In many practical applications, including but not limited to, population statistics, traffic statistics, and location based services, statistics are published by means of range counting [35,36]. Therefore, our preliminary grid partition and statistical method is in line with practical application scenarios. Secondly, the uniform distribution judgment condition has been designed for the grid areas in a bid to determine the distribution characteristics of the non-empty grids. The grid density is graded via discrete wavelet transform which, therefore, facilitates in achieving coarse clustering of location-based big data statistical results. Finally, a bottom-up grid clustering algorithm is proposed. The uniform grid and empty grid belonging to the same level of density are clustered and merged according to the neighborhood similarity. This, therefore, reduces the noise error and uniform assumption error in the released statistical results.

Definition 6 (Grid Uniformity)**.** *For a grid G with a density of $Den(G)$, let $Den(D_1)$, $Den(D_2)$, $\ldots$, $Den(D_i)$ be the density of the sub-regions of grid G partitioned from horizontal, vertical, diagonal, and other directions. If a row vector $V = \{Den(D_1), Den(D_2), \ldots, Den(D_i)\}$ is constructed, then the distribution uniformity of the grid G can be represented by the variance $Var(V)$ of vector V. If the following Formula (6) is satisfied, the grid G is said to be a uniformly distributed grid:*

$$|LFC - \log_{10}(\frac{Den(G)}{i})^2| \leq \theta \tag{6}$$

wherein $LFC = log_{10}(Var(V))$, i is the number of sub-regions after multi-directional segmentation and θ is the threshold.

Definition 7 (Neighborhood Grid)**.** *Let C_i be the center of grid G_i and the distance between any two grids with shared adjacent edge be 1. If the distance between grids G_i and G_j satisfies the following Formula (7), the grid G_j is referred to as the neighborhood grid of the grid G_i (as shown in Figure 6).*

$$dis(C_i, C_j) \leq \sqrt{2} \tag{7}$$

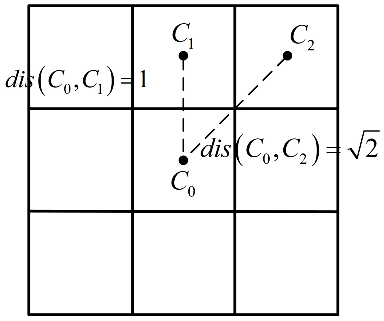


Figure 6. Neighborhood grids.

Regions with the same uniformity but having large differences in terms of their density are not suitable for direct merging. Therefore, we employ the discrete wavelet transform (DWT) so as to classify the matrix formed via the grids' density. Subsequently, the grids of different levels are merged according to the neighborhood similarity to form the final publishing structure. DWT can decompose a signal into different sub-bands with frequency and time information in a bid to realize localized and detailed analysis of time or space. The decomposition process of DWT is depicted in Figure 7. The input signal $X(n)$ is firstly computed by the low-pass (marked as $f\_l(n)$) and high-pass (marked as $f\_h(n)$) horizontal filters respectively, and is then down-sampled by a factor of 2. Subsequently, the outputs L_1 and H_1 are processed by the low-pass and high-pass vertical filters separately, and are again down-sampled. If necessary, the next-level DWT decomposition can be performed on the low-frequency coefficients (LL_1) of the previous layer DWT according to the same method. The low-frequency coefficients of the DWT transform contain most of the energy of a signal. Therefore, the low-frequency coefficients of the density matrix after DWT transform were selected to carry out the classification of grid density in this paper.

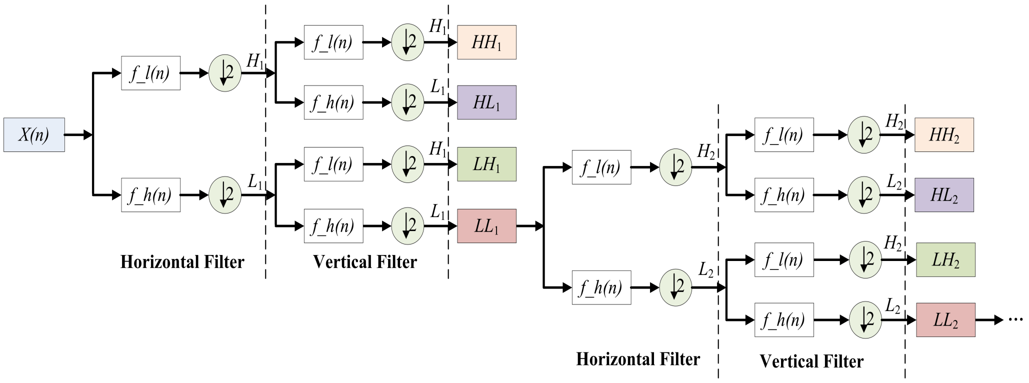


Figure 7. Decomposition process of DWT.

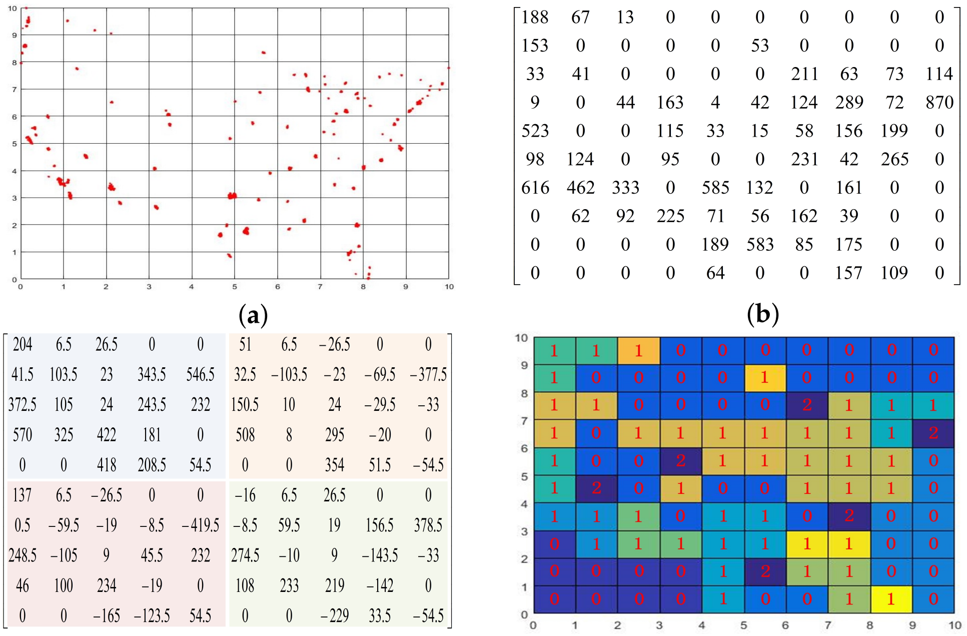
Definition 8 (Grid Density Grading)**.** *Let LL refer to the low-frequency coefficients of the original grid density matrix after the 2D DWT transform. Given two thresholds,* $T_1 = \frac{1}{3}\overline{LL}$ *and* $T_2 = \frac{2}{3}\overline{LL}$*, the original grid density matrix can be classified into three levels:*

$$Class(i,j) = \{ \begin{array}{lll} 1 & if & LL(i,j) \leq T_1 \\ 2 & if & T_1 < LL(i,j) \leq T_2 \\ 3 & if & T_2 \leq LL(i,j) \end{array} \tag{8}$$

Algorithm 1 delineates the detailed implementation mechanism of the grid clustering algorithm proposed in this paper. Lines 1–18 perform uniform grid partition and density statistics on the 2D space covered by the location dataset. The initial space is segregated into empty grids (flag = 0), uniform grids (flag = 1) and non-uniform grids (flag = 2) by uniformity judgement. The uniform grids are further graded via their density. Lines 19–34 complete various types of grid clustering: for uniform grids with flag = 1, the neighborhood grids with the same density classification are integrated to form a cluster; for empty grids with flag = 0, similar clustering and merging are done on adjacent empty grids. The above operation facilitates in reducing the incorporation of Laplace noise and avoiding excessive accumulation of noise errors. For non-uniform grids with flag = 2, as discussed in Section 4.1, a combination of this kind of area causes large uniform assumption errors. Therefore, we will not cluster these type of grids via our proposed algorithm. Lines 35–36 return the final cluster structure matrix. Figure 8 portrays the results of the proposed grid clustering algorithm by taking the location dataset, Storage, as an example (different colors are used to represent different clusters).

Algorithm 1 Grid clustering algorithm.

```
Require: Location-based dataset $D$; partition granularity $m$; uniformity threshold $\theta$; density grading threshold $T_1$ and $T_2$;
Ensure: Cluster structure matrix $CSM$; grid density matrix $Den$;
 1: Partition the 2D space of location-based dataset $D$ into $m \times m$ uniform grids
 2: $Den(i,j)$=0, $flag(i,j)$=0, $Class(i,j)$=0, $uni\_G(i,j)$=0, $null\_G(i,j)$=0, $(i,j = 1,2,...,m)$
 3: for each grid do
 4:     $Den(i,j) \longleftarrow$ calculate the density of current grid
 5:     if $Den(i,j) = 0$ then
 6:         $flag(i,j) = 0$
 7:     else
 8:         if uniformity of the current grid satisfies Formula (6) then
 9:             $flag(i,j) = 1$
10:         else
11:             $flag(i,j) = 2$
12:         end if
13:     end if
14: end for
15: $cA \longleftarrow$ complete DWT transform on $Den$ and get the low-frequency coefficients
16: for each grid with $flag$=1 do
17:     $Class(i,j) \longleftarrow$ grid density grading according to Formula (8)
18: end for
19: for each uniform grid with $flag$=1 do
20:     for each class within Formula (8) do
21:         if the current grid's neighborhood grid satisfies Definition (7) then
22:             $uni\_G \longleftarrow$ set the connected grid with the label of the current grid
23:         else
24:             $uni\_G \longleftarrow$ mark the current grid as a new sequence number
25:         end if
26:     end for
27: end for
28: for each empty grid with $flag$=0 do
29:     if the current grid's neighborhood grid satisfies Definition (7) then
30:         $null\_G \longleftarrow$ set the connected grid with the label of the current grid
31:     else
32:         $null\_G \longleftarrow$ mark the current grid as a new sequence number
33:     end if
34: end for
35: $CSM \longleftarrow$ merge the matrix $uni\_G$ and $null\_G$
36: return $CSM$, $Den$
```



(**a**)

$$\begin{bmatrix} 188 & 67 & 13 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 153 & 0 & 0 & 0 & 0 & 53 & 0 & 0 & 0 & 0 \\ 33 & 41 & 0 & 0 & 0 & 0 & 211 & 63 & 73 & 114 \\ 9 & 0 & 44 & 163 & 4 & 42 & 124 & 289 & 72 & 870 \\ 523 & 0 & 0 & 115 & 33 & 15 & 58 & 156 & 199 & 0 \\ 98 & 124 & 0 & 95 & 0 & 0 & 231 & 42 & 265 & 0 \\ 616 & 462 & 333 & 0 & 585 & 132 & 0 & 161 & 0 & 0 \\ 0 & 62 & 92 & 225 & 71 & 56 & 162 & 39 & 0 & 0 \\ 0 & 0 & 0 & 0 & 189 & 583 & 85 & 175 & 0 & 0 \\ 0 & 0 & 0 & 0 & 64 & 0 & 0 & 157 & 109 & 0 \end{bmatrix}$$

(**b**)

$$\begin{bmatrix} 204 & 6.5 & 26.5 & 0 & 0 & 51 & 6.5 & -26.5 & 0 & 0 \\ 41.5 & 103.5 & 23 & 343.5 & 546.5 & 32.5 & -103.5 & -23 & -69.5 & -377.5 \\ 372.5 & 105 & 24 & 243.5 & 232 & 150.5 & 10 & 24 & -29.5 & -33 \\ 570 & 325 & 422 & 181 & 0 & 508 & 8 & 295 & -20 & 0 \\ 0 & 0 & 418 & 208.5 & 54.5 & 0 & 0 & 354 & 51.5 & -54.5 \\ 137 & 6.5 & -26.5 & 0 & 0 & -16 & 6.5 & 26.5 & 0 & 0 \\ 0.5 & -59.5 & -19 & -8.5 & -419.5 & -8.5 & 59.5 & 19 & 156.5 & 378.5 \\ 248.5 & -105 & 9 & 45.5 & 232 & 274.5 & -10 & 9 & -143.5 & -33 \\ 46 & 100 & 234 & -19 & 0 & 108 & 233 & 219 & -142 & 0 \\ 0 & 0 & -165 & -123.5 & 54.5 & 0 & 0 & -229 & 33.5 & -54.5 \end{bmatrix}$$

(**c**)

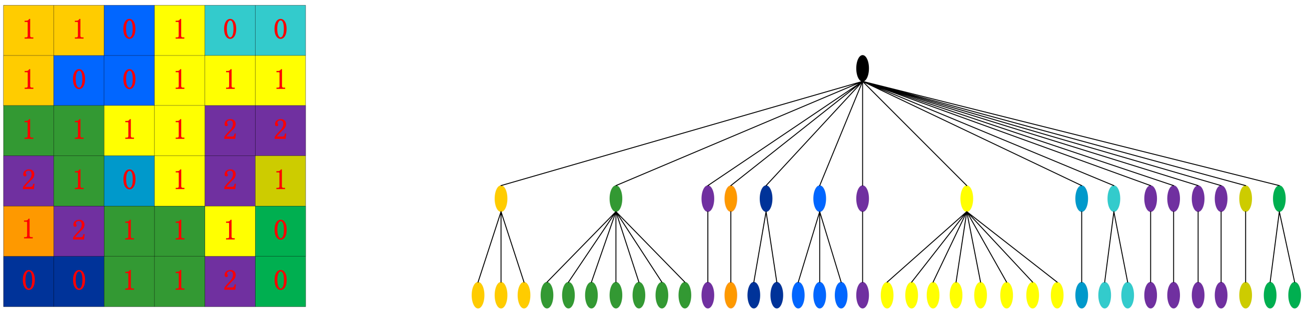
10										
9	1	1	1	0	0	0	0	0	0	0
8	1	0	0	0	0	1	0	0	0	0
7	1	1	0	0	0	0	2	1	1	1
6	1	0	1	1	1	1	1	1	1	2
5	1	0	0	2	1	1	1	1	1	0
4	1	2	0	1	0	0	1	1	1	0
3	1	1	1	0	1	1	0	2	0	0
2	0	1	1	1	1	1	1	1	0	0
1	0	0	0	0	1	2	1	1	0	0
0	0	0	0	0	1	0	0	1	1	0
0	1	2	3	4	5	6	7	8	9	10

(**d**)

Figure 8. Grid partition and clustering process of Storage dataset. (**a**) grid partition; (**b**) density matrix; (**c**) coefficient matrix after DWT; (**d**) grid clustering result.

5. Statistical Publishing and Differential Privacy Protection Algorithm

In order to realize the privacy protection of the published location-based big data, it is necessary to allocate the differential privacy budget according to the spatial indexing structure and incorporate disturbance noise into the statistical results of each spatial region. Algorithm 2 depicts the differential privacy preserving publishing process of location-based big data. Firstly, according to Algorithm 1, the distribution structure of the whole space is obtained. Then, the statistical results within each cluster are calculated, and the Laplacian noise with a privacy budget of ϵ is incorporated into the statistical results of each cluster. Finally, the noisy statistical results will be evenly allocated to each grid region according to the number of grids within the cluster in a bid to balance the noise error and the uniform assumption error and improve the usability of the published statistical results. Figure 9 portrays the schematic of the grid partition structure and the clustering process, wherein Figure 9a suggests the type of the underlying grid with red numbers (i.e., flag = 0 correspond to the empty grids, flag = 1 represent the uniform grids, and flag = 2 stand for the non-uniform grids). Only adjacent empty grids or uniform grids possessing the same density level can be integrated to form the grid clusters. Hence, neither the non-uniform grids nor the uniform grids with different density levels can be merged to reduce the uniformity assumption error.



1	1	0	1	0	0
1	0	0	1	1	1
1	1	1	1	2	2
2	1	0	1	2	1
1	2	1	1	1	0
0	0	1	1	2	0

(**a**) (**b**)

Figure 9. Illustration of the grid clustering result and the spatial indexing structure. (**a**) supposed grid clustering result; (**b**) correspondent spatial indexing structure.

Algorithm 2 Differential privacy preserving publishing algorithm.

```
Require: Cluster structure matrix $CSM$; grid density matrix $Den$; privacy budget $\epsilon$; Sensitivity $S$;
Ensure: Publish statistical result based on grid $Den^*$;
Den^*(i, j)=0
$n \longleftarrow$ calculate the number of clusters according to $CSM$
for each cluster do
    $u \longleftarrow$ calculate the number of grids within the current cluster
    $sum\_Den \longleftarrow$ sum the density of the $u$ grids according to $Den$
    $noisy\_sum = sum\_Den + Laplace(\frac{S}{\epsilon})$
    $Den^*(i, j) = noisy\_sum/u$
end for
return $Den^*$
```

Corollary 1. *Algorithm 2 can provide ϵ-differential privacy protection for the published location-based big data statistical results.*

Proof. For any querying range Q submitted by a user, the range counting query has the following use cases:

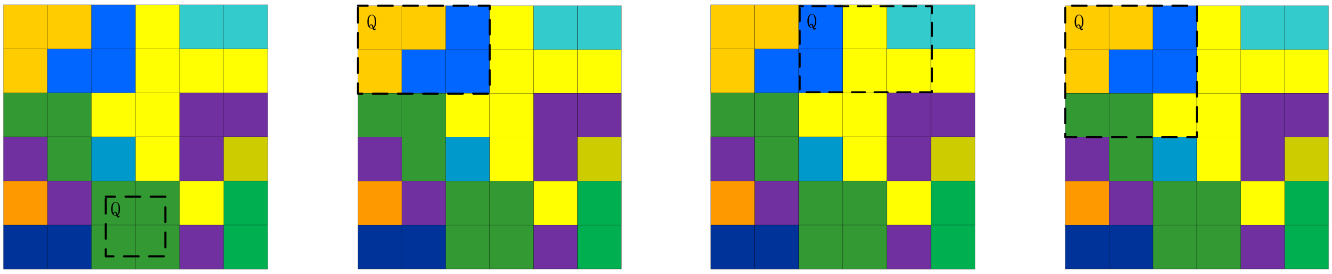
(1) The querying range Q located within the area covered by a certain cluster as depicted in Figure 10a. According to the grid clustering Algorithm 1 proposed in this paper, the querying range Q may be included in a single underlying grid or in a region merged by multiple underlying grids. In the former situation, it can be concluded that this single underlying grid is a non-uniform grid and forms a cluster alone.Therefore, Algorithm 2 incorporates Laplace noise in this grid with a differential privacy budget of ϵ. In the latter situation, Algorithm 2 incorporates Laplace noise to the merged region with a differential privacy budget of ϵ. According to the parallel combination characteristics of the differential privacy model described in Theorem 2, each grid that falls in the merged region satisfies the differential privacy protection of $max\{\epsilon_i\} = \epsilon$.

(2) The querying range Q consists of the area covered by A ($A \geq 1$) complete clusters as depicted in Figure 10b. In this case, each cluster conforms to the first use case described above. Therefore, according to the parallel combination characteristics of the differential privacy model described in Theorem 2, all the A complete clusters included in the querying range Q can provide differential privacy protection of $max\{\epsilon_i\} = \epsilon$.

(3)The querying range Q consists of the area covered by B ($B \geq 1$) intersecting clusters as depicted in Figure 10c. In this case, each of the intersecting area conforms to the case (1) and can also provide differential privacy protection with a strength of ϵ.

(4) The querying range Q consists of area covered by A ($A \geq 1$) complete clusters and B ($B \geq 1$) intersecting clusters, as depicted in Figure 10d. For the complete clusters falling in the querying range Q, each cluster can provide ϵ differential privacy protection similar to use case (2). For the clusters intersected with the querying range Q, the intersected region conforms to use case (3) and provide ϵ differential privacy protection. Therefore, it can also provide differential privacy protection of ϵ according to the parallel combination characteristics of the differential privacy model.

Based on the above use cases, we can draw the conclusion that Algorithm 2 can provide differential privacy protection of ϵ for the published location-based big data statistical results. □



(**a**) (**b**) (**c**) (**d**)

Figure 10. Three use cases of the querying range Q. (**a**) Use case (1); (**b**) Use case (2); (**c**) Use case (3); (**d**) Use case (4).

6. Experiments and Analysis

In order to comprehensively evaluate the proposed grid clustering and differential privacy protection publishing algorithm (GCDPP) for location-based big data statistical information, we compare and analyze the proposed algorithm with a number of classical privacy spatial decomposition algorithms in terms of the availability of published location-based big data statistical results, efficiency of the privacy protection publishing algorithm, and the influences of partition granularity. The baseline methods include, but are not limited to, uniform grid partition method (UG) [11], adaptive grid partition method (AG) [11], standard deviation circle radius adaptive grid decomposition algorithm (SDCAG) [14], quadtree partition method (Quad-opt) [17], density-based quadtree partition method (DBP) [19], and unbalanced quadtree partition method (unbalanced quadtree) [20].

Experimental location-based datasets include the facility location information datasets, Storage http://www.infochimps.com/datasets/storage-facilities-by-neighborhood-2, accessed on 1 January 2022, and Landmark http://www.infochimps.com/datasets/storage-facilities-by-landmarks, accessed on 1 January 2022, provided by Infochimps; Checkin http://snap.stanford.edu/data/loc-gowalla.html, accessed on 1 January 2022, which provides location information from the social network site, Gowalla; and Taxi record dataset, Yellow_trip (2009–2016) https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page, accessed on 1 January 2022, provided by the New York City Taxi Management Committee. The experimental platform selects the Alibaba Cloud Server ECS (ecs.r6.2xlarge: 8-core CPU, 64 GB memory, 100 GB ESSD cloud disk, Windows Server 2019 Data Center Edition 64-bit), and all the algorithms are programmed in MATLAB.

6.1. Availability Analysis of the Published Location-Based Big Data Statistical Results

According to the application scenarios of location-based big data, the accuracy of the range counting query has been employed to ascertain the availability of location-based big data statistical publishing results. Range counting queries of different sizes are implemented on the published results and compared with the baseline methods.

For parameter settings of the baseline algorithms, we refer to their corresponding literature. Specifically, for the *Storage* dataset, the UG algorithm sets the constant parameter $c = 10$, the AG and SDCAG algorithm employ the privacy allocation ratio of $\alpha = 0.5$, the Quad-opt algorithm adopts the partition depth of $h = 6$, the DBP algorithm set maximum density difference $\beta = 5$, and Unbalanced quadtree algorithm use the uniformity judgment threshold of $\theta = 0.01$. The sensitivity of range counting query caused by incorporating differential privacy noise is $S = 1$. For the other three big datasets, the constant parameter was set to $c = 1000$ for the UG algorithm, division depth to $h = 8$ for the Quad-opt algorithm, and other parameters remained unchanged. The information pertinent to range counting queries of different sizes is depicted in Table 1. The differential privacy model incorporates Laplace noise with privacy budgets of $\epsilon = 0.01, \epsilon = 0.1$, and $\epsilon = 1$. Each type of query was randomly generated for 1000 times to ascertain the average relative error between the original and published statistical results. The definition of relative error is presented in Formula (9), wherein Q represents the querying range, $Count(Q)$ returns the statistical result obtained within the querying range on the original location-based dataset, and $Count^*(Q)$ is the noisy statistical result obtained on the published dataset within the same range Q. In order to prevent the denominator from being zero, we set $\rho = 0.001 \times |T|$ according to the UG algorithm [11], wherein $|T|$ represents the size of the location-based dataset.

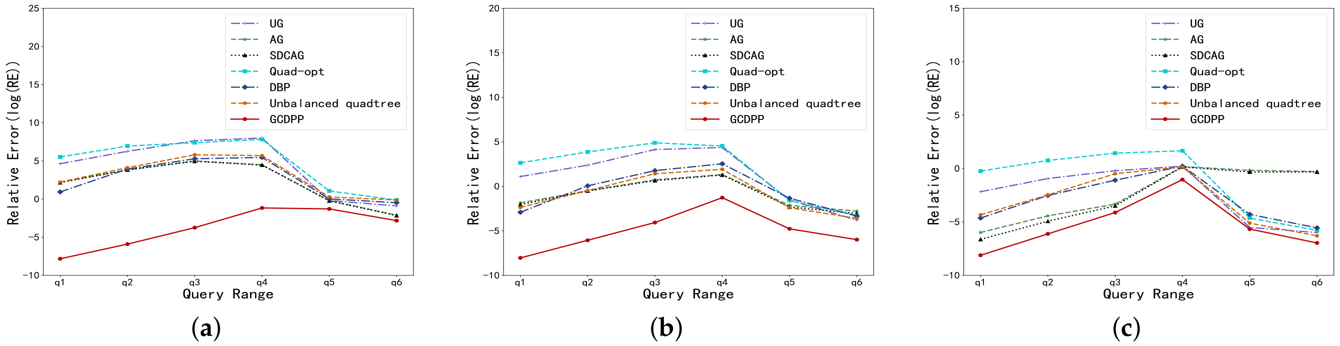
Table 1. Experimental datasets and querying range information.

Parameter		Storage	Landmark	Checkin	Yellow_trip
Data amount (points)		8938	870,052	1,000,000	10,996,214
Dataset coverage (longitude×latitude)		53° × 23°	57° × 25°	353° × 143°	1° × 1°
Querying range (longitude×latitude)	q_1	1.25° × 0.625°	1.25° × 0.625°	6° × 3°	0.02° × 0.02°
	q_2	2.5° × 1.25°	2.5° × 1.25°	12° × 6°	0.04° × 0.04°
	q_3	5° × 2.5°	5° × 2.5°	24° × 12°	0.08° × 0.08°
	q_4	10° × 5°	10° × 5°	48° × 24°	0.16° × 0.16°
	q_5	20° × 10°	20° × 10°	96° × 48°	0.32° × 0.32°
	q_6	40° × 20°	40° × 20°	192° × 96°	0.64° × 0.64°

$$RE(Q) = \frac{|Count^*(Q) - Count(Q)|}{max\{Count(Q), \rho\}} \tag{9}$$

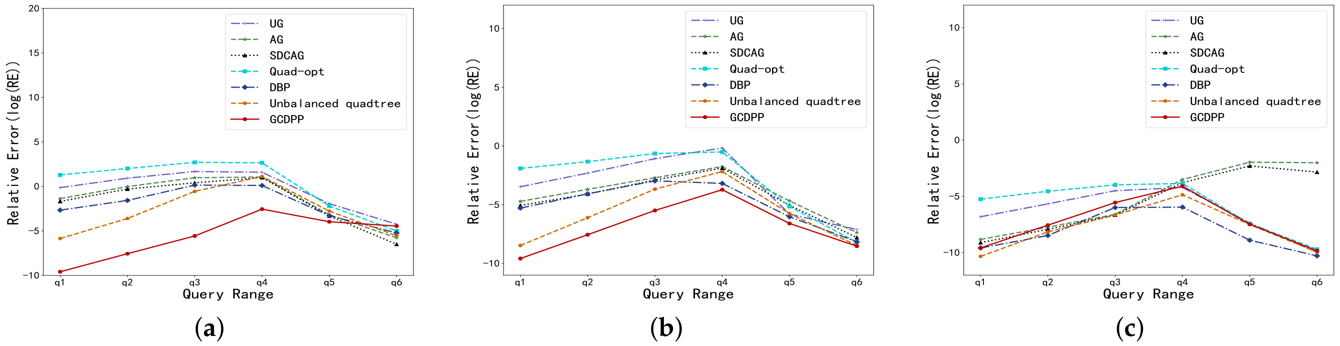
Figures 11–14 portray the comparison of querying accuracy, i.e., in terms of relative error, on different datasets and privacy budgets. On the same experimental dataset, the relative error of all the algorithms gradually decreases with an increase in the privacy budget ϵ. The primary reason is that the increase in the privacy budget ϵ reduces the incorporated Laplace noise, thereby shrinking the error between the published result and the real statistical value. For all the datasets, as the querying range increases from small to large, the relative error first increases and then decreases. This is mainly because, when the querying range is small (e.g., of size q_1 and q_2), the included areas only contain some underlying grids or merged cluster areas, and, therefore, the accumulated noise interference is small, thereby making the querying error relatively low. When the querying range is large (e.g., of size q_5 and q_6), the included areas contain more regions belonging to different clusters, and, therefore, the uniformity assumption error is small, thereby making the overall querying error relatively low.

When we compare the relative errors of various privacy preserving publishing algorithms on the same dataset, it can be observed that the UG and Quad-opt algorithm have higher relative errors in contrast to the others under various privacy budgets. The reason is that the spatial partition process of these two algorithms has nothing to do with the spatial distribution of location-based information, which not only produces more empty nodes and noise errors but also introduces more uniform assumption error owing to the non-uniform distribution of the nodes. The AG algorithm performs an additional fine-grained partition for the dense grids on the basis of the UG algorithm. However, it still cannot overcome the large number of noise errors caused by the empty nodes in areas with sparse location distribution. Therefore, the relative error of the AG algorithm is low when the querying range is small and the value deteriorates when the querying range is large. In the SDCAG algorithm, filtering and bucketing are used to reduce the noise error. Therefore, it provides better accuracy of range counting query than the AG algorithm. The DBP and Unbalanced quadtree algorithm implements a heuristic quadtree partition according to the uniformity of the location distribution, thereby compensating the uniformity assumption error and the noise error caused by the empty nodes to a certain extent. The proposed GCDPP algorithm adopts the strategy of bottom-up neighborhood clustering. The initial uniform grid partition takes into account the need for small-scale counting query services and retains a better accuracy of location services. The bottom-up neighborhood grid clustering reduces the excessive partition of sparse regions and suppresses excessive noise errors. In addition, it further facilitates merging the uniform regions of the same density level and apportions the differential privacy noise. Therefore, the proposed algorithm achieves better querying accuracy in contrast to the other algorithms under different datasets and different privacy budgets.



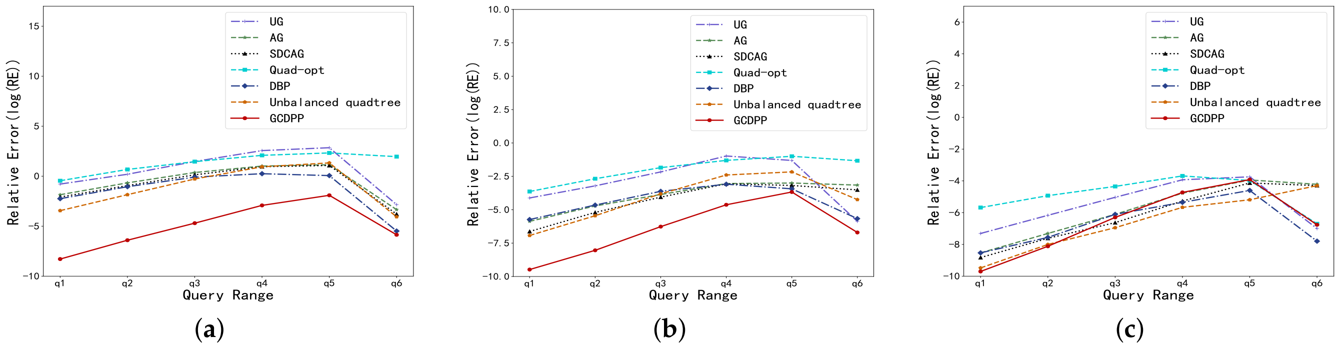
(**a**) (**b**) (**c**)

Figure 11. Comparison of querying accuracy on Storage dataset. (**a**) Storage dataset, $\epsilon = 0.01$; (**b**) Storage dataset, $\epsilon = 0.1$; (**c**) Storage dataset, $\epsilon = 1$.



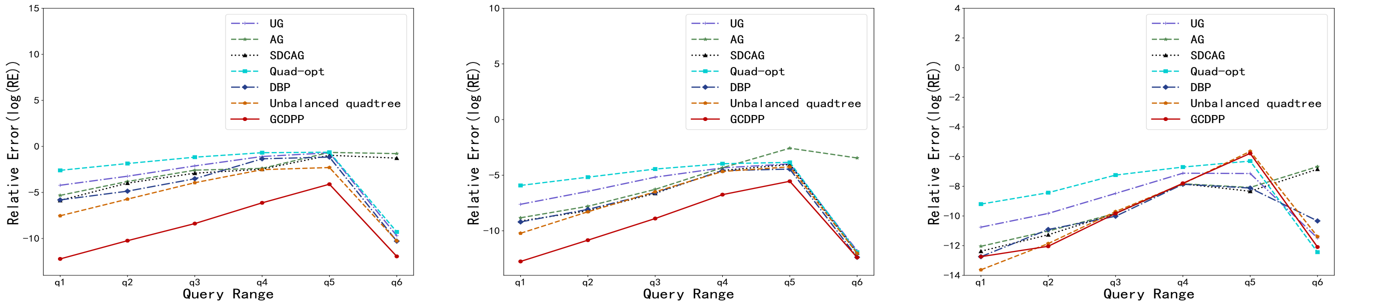
(**a**) (**b**) (**c**)

Figure 12. Comparison of querying accuracy on Landmark dataset. (**a**) Landmark dataset, $\epsilon = 0.01$; (**b**) Landmark dataset, $\epsilon = 0.1$; (**c**) Landmark dataset, $\epsilon = 1$.



(**a**) (**b**) (**c**)

Figure 13. Comparison of querying accuracy on Checkin dataset. (**a**) Checkin dataset, $\epsilon = 0.01$; (**b**) Checkin dataset, $\epsilon = 0.1$; (**c**) Checkin dataset, $\epsilon = 1$.



(**a**) (**b**) (**c**)

Figure 14. Comparison of querying accuracy on Yellow_trip dataset. (**a**) Yellow_trip dataset, $\epsilon = 0.01$; (**b**) Yellow_trip dataset, $\epsilon = 0.1$; (**c**) Yellow_trip dataset, $\epsilon = 1$.

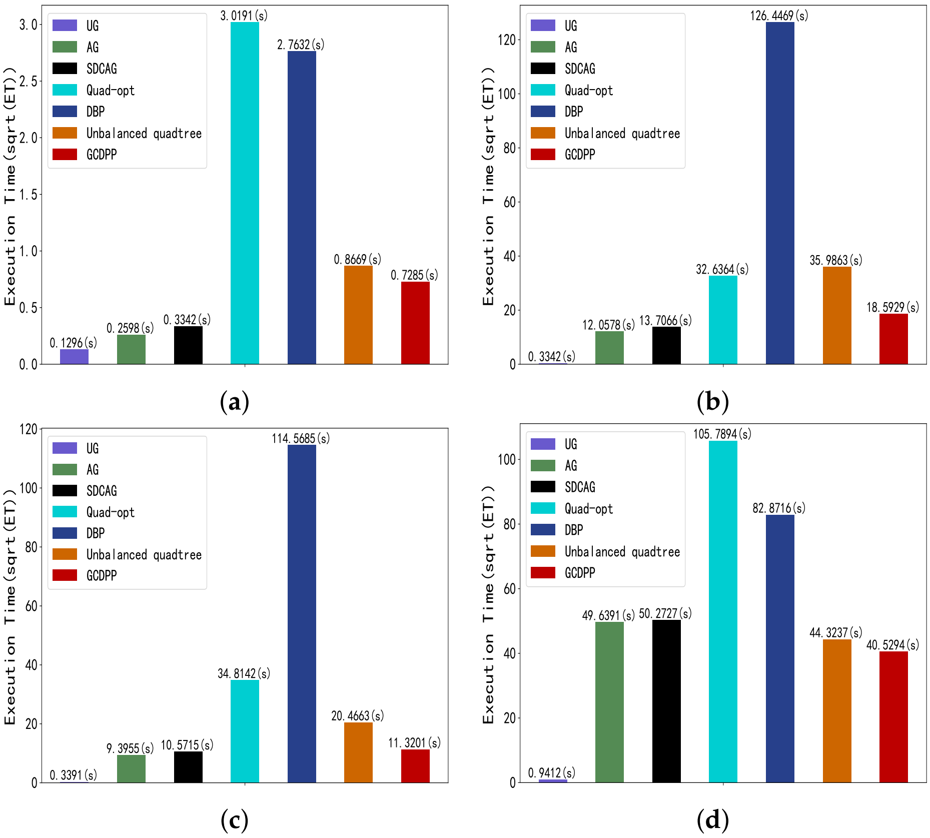
6.2. Efficiency Analysis of the Privacy Protection Publishing Algorithm

Table 2 compares the time complexity of the proposed GCDPP algorithm with the baseline methods. For a dataset encompassing n records, whilst the UG, AG, and SDCAG algorithm have the same time complexity of $O(n)$, the partition process of the UG algorithm only scans the input data for one time, whereas the AG and SDCAG algorithm scan the input data for at least two times to form a two-layer adaptive grid partition. The Quad-opt algorithm performs a recursive quadtree partition on the input dataset. The overall time complexity is about $O(n\log(n))$. The partition process of the DBP and Unbalenced quadtree algorithm is related with the specific distribution of location-based dataset and partition depth of quadtree. In the worst scenario, for a location-based dataset with n records and h partition depth, the time complexity of these two algorithms will be $O(hn)$. The proposed GCDPP algorithm first performs the same partition as the UG algorithm, and then merges the neighbourhood grids of the same type and density grade to form the clusters. Its time complexity is about $O(n + n) \approx O(n)$.

Table 2. Comparison of time complexity.

Algorithm	Time Complexity
UG	$O(n)$
AG	$O(n)$
SDCAG	$O(n)$
Quad-opt	$O(n\text{log}(n))$
DBP	$O(hn)$
Unbalanced quadtree	$O(hn)$
GCDPP	$O(n)$

Figure 15 depicts the comparison in the execution time of all the privacy protection algorithms on real location-based datasets. The parameter settings of these algorithms are the same as those in Section 6.1. Since the strength of differential privacy budget has no significant effect on the execution time of the statistical publishing algorithms, we take $\epsilon = 1$ as an example to conduct the experimental comparisons on the real location-based datasets of different scales. From a macro perspective, the overall execution time of all the algorithms increase with the size of the dataset. Specifically, the UG algorithm has the lowest execution time and is hardly affected by the size of the dataset. The AG and SDCAG algorithm perform an additional round of grid partition on the basis of the UG algorithm, and, therefore, they take a much longer time. In order to reduce the noise error, the SDCAG algorithm filters the grids with a raw count of 0 and allocates the similar grids into the same bucket. Thus, the overall execution time is slightly higher than the AG algorithm. Although the partition process of the Quad-opt algorithm does not consider the specific distribution state of the location information, the tree-based iterative process using depth-first traversal is primarily affected by the size of the dataset. Therefore, the overall execution time of the Quad-opt algorithm is significantly higher than other algorithms. The partition process of the DBP and Unbalanced quadtree algorithm need to be combined with the specific distribution of the location information. In the areas wherein the location distribution is particularly sparse or dense, they can avoid unnecessary partition. Therefore, they received better performance on the sparse dataset *Storage* and the dense dataset *Yellow_trip*. The proposed GCDPP algorithm firstly implements the uniform grid partition stage and then performs bottom-up neighborhood grid clustering. The grid-based clustering accelerates the merging of the spatial regions and saves much time compared to the process of adaptive grid partition and filtering. Thus, the execution time of the GCDPP algorithm is only secondary to the UG algorithm. It can be expected that, with an increasing in the size of dataset, the advantage of the proposed GCDPP algorithm will be more obvious.



(**a**) (**b**)

(**c**) (**d**)

Figure 15. Comparison of operating efficiency of various algorithms under different datasets. (**a**) Storage dataset, $\epsilon = 1$; (**b**) Landmark dataset, $\epsilon = 1$; (**c**) Checkin dataset, $\epsilon = 1$; (**d**) Yellow_trip dataset, $\epsilon = 1$.

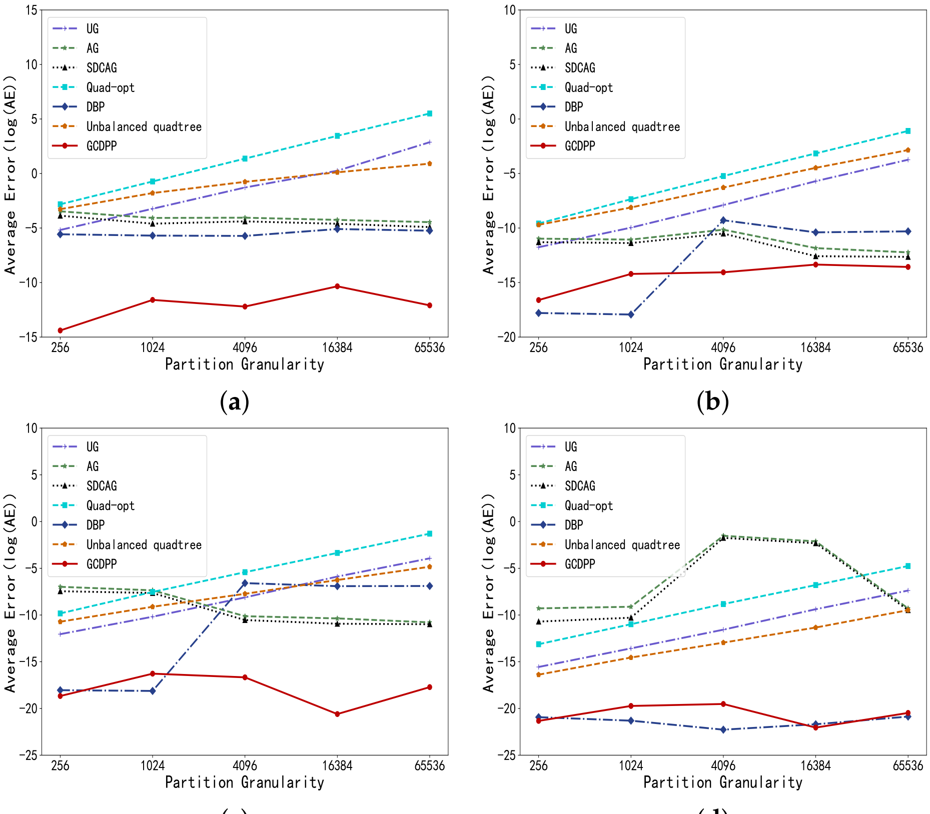
6.3. The Influences of Partition Granularity

As discussed in Section 4.1, traditional privacy spatial decomposition methods are easily affected by partition granularity, resulting in over-partitioning or under-partitioning problems that not only impair the availability of published data but may also drag down the efficiency of the entire data publishing algorithm. This section analyses the relationships between partition granularity setting, accuracy of published results, and publishing algorithm efficiency. Since the privacy budget strength has no obvious effect on the setting of partition granularity, we take $\epsilon = 1$ as an example to analyze the average error of the published statistical results with different partition granularities. For the sake of fairness, we keep the initial partition granularity of all the comparison algorithms basically the same. The average error of the published statistical results can be expressed as follows:

$$AE = \frac{\sum_{i=1}^{N} |Den_i^* - Den_i|}{|T|} \tag{10}$$

wherein N is the initial partition granularity of the location-based dataset, and $|T|$ represents the size of the dataset. Den_i stands for the original density of grid i, whereas Den_i^* returns the published density of the corresponding area.

Figure 16 portrays the average error of all the algorithms under different partition granularity. It can be observed that, with an increase in the partition granularity, the average error of the tree-based partition algorithms grows gradually, whereas the average error of the grid-based partition algorithms do not change obviously. The primary reason is that, when the partition granularity is small, the coverage area of a single grid will be large and the published statistical result may contain more non-uniform assumption errors due to the inhomogeneous distribution of data within the region. Nevertheless, the introduced noise error will be small because of the less number of empty grids. With an increase in partition granularity, the coverage area of each grid will be reduced. The non-uniform assumption error for the grid area will decrease, whereas the total noise error will increase because the number of empty grids may increase. The proposed algorithm achieves the lowest average error on almost all experimental datasets demonstrating the advantage of the proposed algorithm in the availability of the published statistical result.



(**a**)

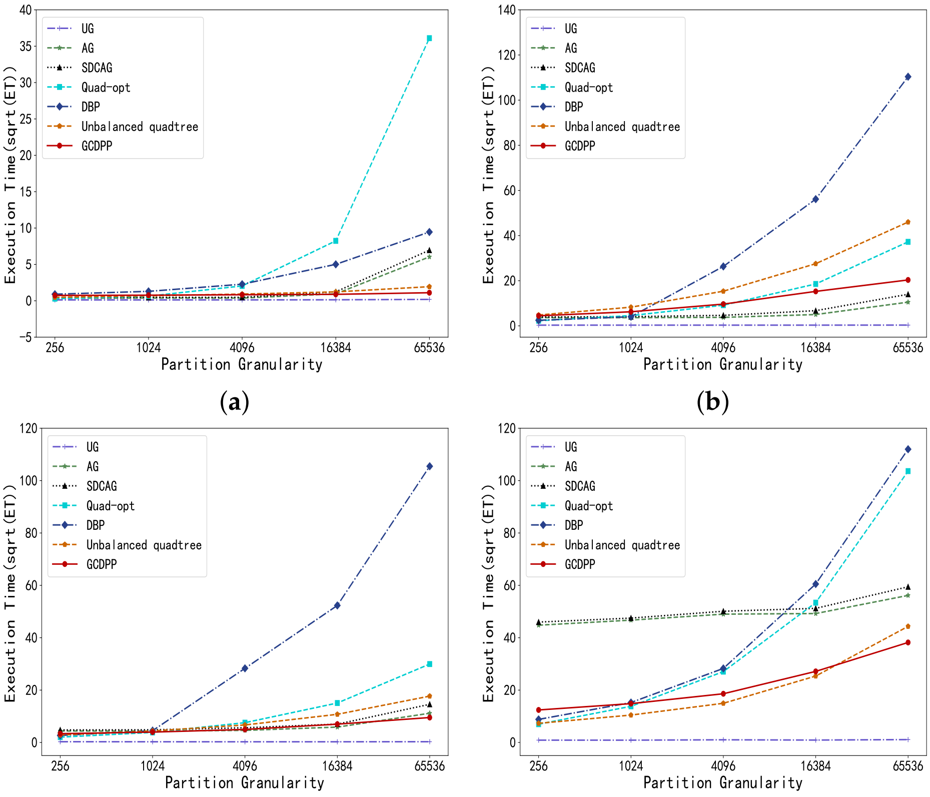
(**b**)

(**c**)

(**d**)

Figure 16. Partition granularity vs. average error. (**a**) Storage dataset, $\epsilon = 1$; (**b**) Landmark dataset, $\epsilon = 1$; (**c**) Checkin dataset, $\epsilon = 1$; (**d**) Yellow_trip dataset, $\epsilon = 1$.

Intuitively, the execution time of partition algorithms will increase with the size of dataset and partition granularity. Figure 17 depicts the execution time of all the algorithms under different partition granularity. The partition progress of UG algorithm is independent of the distribution of dataset, so it always maintains the lowest execution time and hardly changes with the change of the partition granularity. For the other algorithms, the execution time increases significantly with the partition granularity. The execution time of the AG and SDCAG algorithms is slightly longer than that of the UG algorithm. With an increase in a dataset's scale and distribution complexity, the efficiency of these two algorithms gradually deteriorates, especially for the largest and most complex dataset *Yellow_trip*. For the tree-based partition algorithms, the increase of partition gradually causes more recursive operations and uniformity judgments; therefore, the execution time of these algorithms suffers significantly. While the proposed GCDPP algorithm maintains the independent grid partition just as the UG algorithm, its bottom-up grid clustering process overcomes the excessive recursive operations. Therefore, with an increase in partition granularity, the proposed algorithm is still better than the other algorithms. It is foreseeable that the advantages of the proposed algorithm will be more obvious in the case of location-based dataset with larger scale and more complex distribution.



(**a**)

(**b**)

(**c**)

(**d**)

Figure 17. Partition granularity vs. execution time. (**a**) Storage dataset, $\epsilon = 1$; (**b**) Landmark dataset, $\epsilon = 1$; (**c**) Checkin dataset, $\epsilon = 1$; (**d**) Yellow_trip dataset, $\epsilon = 1$.

7. Conclusions

The rapid development of big data technology and the widespread popularization of mobile smart terminals has made various location-based services highly relevant to the work and life of users. Accordingly, the issues of location privacy leakage caused by the release of location-based big data statistics have started attracting widespread attention. Achieving the release of statistics of location-based information via the spatial decomposition method and the differential privacy protection model can effectively avoid various risks caused by location privacy leakage on the premise of ensuring data availability. In order to reduce the loss of data availability caused by over-partitioning and under-partitioning in the traditional top-down spatial decomposition method, this paper proposed a bottom-up grid clustering and differential privacy protection publishing method. The accuracy of location-based services is met through uniform grid partition and density statistics. Privacy protection of the published data is achieved by incorporating Laplace noise to location-based big data statistics according to the differential privacy model. The merging of the blank areas and the uniform distribution areas has been realized by the neighborhood grid clustering in order to reduce the noise error and the uniform assumption error. Experiments and analysis on the real location-based datasets prove that the proposed grid clustering and differential privacy publishing method have obvious advantages in improving the range querying accuracy and operating efficiency.

However, the research still has some limitations. Firstly, most of the existing spatial partition methods of location big data are based on grid or tree structures. However, the areas where human actually work and live are primarily closely related to the distribution of infrastructure and cannot be well represented by grids or tree structures. Geographic segmentation methods such as road network structures did not have effective index methods and cannot achieve efficient querying services for location-based big data. Therefore, how to design more detailed and flexible partition method combined with efficient index structure will facilitate improving the availability of published results and is one of our future works. Secondly, the paper only focuses on the statistical partitioning and publishing of static location-based big data. However, a user's location continuously and randomly changes over time. The dynamic changes of the locations of large number of users make the spatial partition structure of the previous statistical release time not applicable to the next release time. Therefore, how to design a statistical partition and publishing method combined with the dynamic change of location big data, and enhance the timeliness and practicability of published statistical results under the premise of ensuring users' privacy, will be a further research direction. Finally, the paper discusses the statistical publishing method of location-based big data based on the centralized differential privacy model. The premise is that users' locations are collected and statistically released via a trusted third-party. However, there is no completely reliable platform in practical applications. Problems such as technical failures, superuser leaks, and hacker attacks make the users' original locations in danger. Therefore, transferring the privacy protection process of users' locations to the their terminals and realizing local differential privacy location protection have become the research hotspot in recent years.

Author Contributions: Methodology and validation, Yan Yan; investigation, Zichao Sun, Fei Xu and Zhuoyue Dong; formal analysis, Yan Yan and Zichao Sun; resources, Quan Z. Sheng and Adnan Mahmood; data curation, Zichao Sun, Fei Xu and Zhuoyue Dong; writing—original draft preparation, Adnan Mahmood and Yan Yan; writing—review and editing, Quan Z. Sheng, Adnan Mahmood and Yan Yan; visualization, Adnan Mahmood and Zichao Sun; supervision and project administration, Quan Z. Sheng All authors read and agreed to the published version of the manuscript.

Funding: The research at hand was supported by the National Nature Science Foundation of China, No. 61762059.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: [http://www.infochimps.com/datasets/storage-facilities-by-neighborhood-2, http://www.infochimps.com/datasets/storage-facilities-by-landmarks, http://snap.stanford.edu/data/loc-gowalla.html, and https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page].

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, Z.; Qian, J.L.; Du, Y.Y.; Wang, N.; Yi, J.W.; Sun, Y.R.; Ma, T.; Pei, T.; Zhou, C.H. Multi-level spatial distribution estimation model of the inter-regional migrant population using multi-source spatio-temporal big data: A case study of migrants from Wuhan during the spread of COVID-19. *J.-Geo-Inf. Sci.* **2020**, *22*, 147–160.
2. Wu, H.; Gui, Z.; Yang, Z. Geospatial big data for urban planning and urban management. *Geo-Spat. Inf. Sci.* **2020**, *23*, 273–274. [CrossRef]
3. Mohammed, S.; Arabnia, H.R.; Qu, X.; Zhang, D.; Kim, T.H.; Zhao, J. IEEE access special section editorial: Big data technology and applications in intelligent transportation. *IEEE Access* **2020**, *8*, 201331–201344. [CrossRef]
4. Zhou, C.; Su, F.; Pei, T.; Zhang, A.; Du, Y.; Luo, B.; Cao, Z.D.; Wang, J.L.; Yuan, W.; Zhu, Y.Q.; et al. COVID-19: Challenges to GIS with big data. *Geogr. Sustain.* **2020**, *1*, 77–87. [CrossRef]

5. Gruschka, N.; Mavroeidis, V.; Vishi, K.; Jensen, M. Privacy issues and data protection in big data: A case study analysis under GDPR. In Proceedings of the 2018 IEEE International Conference on Big Data, Seattle, WA, USA, 10–13 December 2018; pp. 5027–5033.
6. Takbiri, N.; Houmansadr, A.; Goeckel, D.L.; Pishro-Nik, H. Privacy against statistical matching: Inter-user correlation. In Proceedings of the 2018 IEEE International Symposium on Information Theory, Vail, CO, USA, 17–22 June 2018; pp. 1036–1040.
7. Primault, V.; Boutet, A.; Mokhtar, S.B.; Brunie, L. The long road to computational location privacy: A survey. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 2772–2793. [CrossRef]
8. Yan, Y.; Eyeleko, A.H.; Mahmood, A.; Li, J.; Dong, Z.; Xu, F. Privacy preserving dynamic data release against synonymous linkage based on microaggregation. *Sci. Rep.* **2022**, *12*, 1–22. [CrossRef]
9. Li, D.; Shao, Z.; YU, W.; Zhu, X.; Zhou, S. Public epidemic prevention and control services based on big data of spatiotemporal location make cities more smart. *Geomat. Inf. Sci. Wuhan Univ.* **2020**, *45*, 475–487.
10. Erlingsson, Ú.; Pihur, V.; Korolova, A. Rappor: Randomized aggregatable privacy-preserving ordinal response. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, 3–7 November 2014; pp. 1054–1067.
11. Qardaji, W.; Yang, W.; Li, N. Differentially private grids for geospatial data. In Proceedings of the 2013 IEEE 29th International Conference on Data Engineering, Brisbane, QLD, Australia, 8–12 April 2013; pp. 757–768.
12. Xiong, P.; Zhang, L.; Zhu, T. Reward-based spatial crowdsourcing with differential privacy preservation. *Enterp. Inf. Syst.* **2017**, *11*, 1500–1517. [CrossRef]
13. Yan, Y.; Hao, X.H. Differential privacy partitioning algorithm based on adaptive density grids. *J. Shandong Univ. (Nat. Sci.)* **2018**, *53*, 12–22.
14. Zhou, G.; Tang, X.; Qin, S. Adaptive Grid Decomposition Algorithm based on Standard Deviation Circle Radius. *Int. J. Perform. Eng.* **2019**, *15*, 2145–2152.
15. Wei, J.; Lin, Y.; Yao, X.; Zhang, J. Differential privacy-based location protection in spatial crowdsourcing. *IEEE Trans. Serv. Comput.* **2022**, *15*, 45–58. [CrossRef]
16. Rodríguez, K.M.; Bossy, M.; Maftei, R.; Shekarforush, S.; Henry, C. New spatial decomposition method for accurate, mesh-independent agglomeration predictions in particle-laden flows. *Appl. Math. Model.* **2021**, *90*, 582–614. [CrossRef]
17. Cormode, G.; Procopiuc, C.; Srivastava, D.; Shen, E.; Yu, T. Differentially private spatial decompositions. In Proceedings of the 2012 IEEE 28th International Conference on Data Engineering, Arlington, VA, USA, 1–5 April 2012; pp. 20–31.
18. Wu, Y.; Lu, Q.; Cai, J.; Wang, X. Differential privacy two-dimensional data partitioning publication algorithm based on quad-tree. *J. Huazhong Univ. Sci. Technol. (Nat. Sci. Ed.)* **2016**, *44*, 99–104.
19. Yang, M.; Zhu, T.; Xiang, Y.; Zhou, W. Density-based location preservation for mobile crowdsensing with differential privacy. *IEEE Access* **2018**, *6*, 14779–14789. [CrossRef]
20. Yan, Y.; Gao, X.; Mahmood, A.; Feng, T.; Xie, P. Differential private spatial decomposition and location publishing based on unbalanced quadtree partition algorithm. *IEEE Access* **2020**, *8*, 104775–104787. [CrossRef]
21. Huang, S.; Chen, T.; Lu, Q.; Wu, Y.; Ye, S. Differentially privacy two-dimensional dataset partitioning publication algorithm based on kd-tree. *J. Shandong Univ. (Eng. Sci.)* **2015**, *45*, 24–29.
22. Yan, Y.; Hao, X.; Zhang, L. Hierarchical differential privacy hybrid decomposition algorithm for location big data. *Clust. Comput.* **2019**, *22*, 9269–9280. [CrossRef]
23. Ohadi, N.; Kamandi, A.; Shabankhah, M.; Fatemi, S.M.; Hosseini, S.M.; Mahmoudi, A. Sw-dbscan: A grid-based dbscan algorithm for large datasets. In Proceedings of the 2020 6th International Conference on Web Research, Tehran, Iran, 22–23 April 2020; pp. 139–145.
24. Suo, M.L.; Zhou, D.; An, R.M.; Li, S.L. Neighborhood density grid clustering and its applications. *J. Tsinghua Univ. (Sci. Technol.)* **2018**, *58*, 732–739.
25. Wu, B.; Wilamowski, B.M. A fast density and grid based clustering method for data with arbitrary shapes and noise. *IEEE Trans. Ind. Inform.* **2017**, *13*, 1620–1628. [CrossRef]
26. Xu, H.; Yao, S.; Li, Q.; Ye, Z. An improved k-means clustering algorithm. In Proceedings of the 2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), Dortmund, Germany, 17–18 September 2020; pp. 1–5.
27. Zhu, Q.; Tang, X.; Liu, Z. Revised dbscan clustering algorithm based on dual grid. In Proceedings of the 2020 Chinese Control In addition, Decision Conference (CCDC), Hefei, China, 22–24 August 2020; pp. 3461–3466.
28. Yu, Y.; Jia, Z.; Cao, L.; Zhao, J.D.; Liu, Z.W.; Liu, J.L. Fast density-based clustering algorithm for location big data. *J. Softw.* **2018**, *29*, 2470–2484.
29. Tareq, M.; Sundararajan, E.A.; Mohd, M.; Sani, N.S. Online clustering of evolving data streams using a density grid-based method. *IEEE Access* **2020**, *8*, 166472–166490. [CrossRef]
30. Hu, Y.S.; Lu, Y.H. Cell Clustering Algorithm Based on MapReduce and Strongly Connected Fusion. *Comput. Sci.* **2019**, *46*, 204–207+215.
31. Dwork, C. Differential privacy. In Proceedings of the 33rd International Colloquium on Automata, Venice, Italy, 10–14 July 2006; pp. 1–12.

32. Dwork, C. Differential privacy: A survey of results. In Proceedings of the International Conference on Theory and Applications of Models of Computation, Xi'an, China, 25–29 April 2008; pp. 1–19.
33. Dwork, C. Calibrating noise to sensitivity in private data analysis. *Lect. Notes Comput. Sci.* **2012**, *3876*, 265–284.
34. Dwork, C.; Roth, A. The Algorithmic Foundations of Differential Privacy. *Found. Trends Theor. Comput. Sci.* **2014**, *9*, 211–407. [CrossRef]
35. Losada-Rojas, L.L.; Gkritza, K. Individual and location-based characteristics associated with Autonomous Vehicle adoption in the Chicago metropolitan area: Implications for public health. *J. Transp. Health* **2021**, *22*, 101232. [CrossRef]
36. Hara, Y.; Yamaguchi, H. Japanese travel behavior trends and change under COVID-19 state-of-emergency declaration: Nationwide observation by mobile phone location data. *Transp. Res. Interdiscip. Perspect.* **2021**, *9*, 100288. [CrossRef]