PSDJS: A Privacy-preserving Spatial Dataset Joinable Search in Cloud

Zhengkai Zhang
1024041138
Nanjing, China
1024041138@njupt.edu.cn

Abstract

With the rapid development of cloud computing, cloud-based spatial dataset search has witnessed growing demand. As a critical database operation, joinable search has become increasingly vital. As cloud environments require data uploading, searching, and other operations, addressing data privacy concerns has become imperative, making privacy protection increasingly critical. This paper first proposes a privacy-preserving spatial dataset joinable search in cloud. We first design a grid-based joinable coverage distinction measurement model. Building upon this model, we present our baseline search scheme (PSDJS). To further enhance search efficiency and communication cost, we further introduce our optimized search scheme (PSDJS+), which employs a novel two-layer index (CG-HVIndex), a joinable coverage distinction check table (JCT), and a grouped Bloom filter. Finally, we conduct experimental evaluations on three real-world datasets, demonstrating that our solution achieves high search efficiency and small communication cost.

Keywords

Spatial Dataset, Joinable Search, Data Privacy, Cloud Computing

1 Introduction

In the era of data explosion, the volume and variety of data generated across industries have grown exponentially, making cloud-based storage and computation indispensable for efficient data management and analysis [5, 6, 8]. However, as organizations increasingly rely on cloud platforms to integrate and process multi-source datasets—particularly sensitive spatial data [12, 21] in domains like urban planning, healthcare, and logistics—the need for data privacy protection becomes paramount. Many datasets contain confidential or personally identifiable information, and their direct sharing or joining across untrusted domains raises significant privacy risks. Traditional data integration approaches often overlook these concerns, necessitating privacy-preserving techniques [16] that enable secure collaboration without exposing raw data.

Joinable search [1–4, 23] has emerged as a critical tool for enriching and analyzing distributed datasets, allowing users to discover and link related data across repositories. While existing research has focused on tabular data [17], spatial datasets—which underpin decision-making in transportation, disease control, and municipal planning—remain understudied in this context. Spatial data is often fragmented across independent platforms (e.g., OpenGeoMetadata, GeoBlacklight) or proprietary systems, creating barriers to interoperability. A privacy-preserving joinable search framework for spatial data would not only bridge these gaps but also ensure compliance with data protection regulations.

Related Work. Various schemes have been proposed to enable privacy-preserving spatial data search in the cloud [7, 10, 13, 15, 18, 22], but none of them address the search for spatial datasets. With the increasing application of spatial data, research on spatial dataset [20] has gradually attracted attention. However, this field is still in its infancy, with many key issues remaining to be addressed. Yang et al. [19] proposed a distributed tree-based index for efficient overlap and coverage joinable searches in multi-source spatial datasets, but their reliance on MBRs for joinability determination may lead to false positives. Li et al. [9] developed a privacy-preserving scheme using encrypted density distributions and order-preserving encryption for secure spatial dataset searches in cloud environments, but their approach focuses primarily on density computation rather than spatial coverage.

Motivation. In urban planning, the implementation of urban expansion projects requires data-driven decision making based on spatial datasets. A critical challenge lies in efficiently identifying adjacent areas surrounding existing urban zones that meet expansion criteria, which fundamentally constitutes a spatial joinable search problem. This process involves retrieving potential expansion areas through spatial relationship analysis, ensuring both joinability with target urban areas and compliance with specific spatial coverage constraints. Our research addresses this gap by investigating innovative applications of spatial dataset joinable search techniques in urban expansion planning. Our research aims to investigate spatial dataset joinable search technology to fill this gap. Specifically, we focus on designing effective models to reduce false positives in joinable searches and developing appropriate privacy-preserving strategies to ensure data security during cloud-based search processes.

Overall, the contributions of this paper are as follows:

2 Models, Priliminaries, and Problem Formulation

2.1 System Model and Threat Model

The system model and the thread model in this paper consist of three parts: Data Owner(DO), Data User(DU) and Cloud Server(CS). CS can provide users with superb computing power and reliable data search services, but it can steal and analyze sensitive data driven by illegal interests or financial gains. DO encrypts the spatial datasets and search indexes and shares the key with DU. DU uses the shared key to convert the query request into a trapdoor and sends it to CS.

2.2 Priliminaries

Enhanced Asymmetric Scalar-Product-Preserving Encryption (EASPE). EASPE [14] is a significant technology for searching encrypted data, which can obtain information by computing the inner product of two vectors without exposing data privacy. In EASPE, the secret key sk is defined as $\{s, M_1, M_2, \pi, r_1, r_2, r_3, r_4, r_5, r_6\}$. Given two vectors V_1 and V_2, encryption algorithms $EASPE.Enc(\cdot)$ and $EASPE.TrapGen(\cdot)$ process them to produce ciphertexts $\widetilde{V}_1 = \{M_1^T \widehat{V}_1', M_2^T \widehat{V}_1''\}$ and $\widetilde{V}_2 = \{M_1^{-1} \widehat{V}_2', M_2^{-1} \widehat{V}_2''\}$, respectively. In EASPE, the encryption algorithm preserves the inner product property such that for any encrypted vectors $\widetilde{V}_1$ and $\widetilde{V}_2$, $\widetilde{V}_1^T \cdot \widetilde{V}_2 = V_1^T \cdot V_2$ is satisfied.

Hierarchical Navigable Small World Graphs (HNSW). HNSW [11] is a highly efficient indexing structure designed for storing and retrieving vectors. It organizes vector data by constructing a multi-layer graph, where the upper layers are sparse representations of the lower layers, enabling rapid navigation and scope reduction during searches. By integrating the concepts of small-world networks and hierarchical navigation, HNSW significantly enhances the efficiency of vector searches while maintaining high accuracy, making it exceptionally well-suited for vector data storage and retrieval scenarios.

2.3 Problem Formulation

A spatial data repository is a set of spatial datasets, denoted as $\mathcal{D} = \{D_1, D_2, ..., D_n\}$, and D_i is a spatial dataset having a collection of spatial data points, $D_i = \{p_{i,1}, p_{i,2}, ..., p_{i,m_i}\}$, where $p_{i,x}$ is a location point that contains latitude and longitude.

Definition 1. Exemplar-based Spatial Dataset Joinable Search. Given an exemplar dataset D_e, a spatial data repository $\mathcal{D}$ and a joinable threshold γ, an exemplar-based spatial dataset joinable search, denoted as $Q = (\mathcal{D}, D_e, k)$, is to obtain k datasets from $\mathcal{D}$ as the search result R, such that each dataset in R and D_e is joinable and these k datasets have the highest coverage distinction with D_e. It means that R should satisfy the following conditions:

This paper aims to design exemplar-based privacy-preserving spatial dataset joinable search schemes that can effectively perform the joinable search while preserving the privacy of datasets and search requests.

3 Grid-based Joinable Coverage Distinction Measurement for Spatial Datasets

In this section, we propose a grid-based joinable coverage distinction measurement, which first determines the grid-based spatial dataset joinability and then calculates the grid-based joinable coverage distinction between spatial datasets.

Definition 2. Grid-based Spatial Dataset Representation. Given a two-dimensional spatial domain $\mathcal{P}$ and a grid partitioning granularity threshold θ, $\mathcal{P}$ is uniformly divided into $2^\theta \times 2^\theta$ grids, i.e., $\mathcal{P} = \{g_1, g_2, ..., g_r\}$, where $r = 2^\theta \times 2^\theta$ and $g_i \in \mathcal{P}$ is a grid in $\mathcal{P}$.

Definition 3. Grid Distribution of Spatial Dataset. Given a spatial dataset D_i, the grid distribution of D_i, denoted as G_i, is a set of grids where the location points are located, i.e.,

$$G_i = \{g_{i,j} \mid g_{i,j} \in \mathcal{P} \wedge \exists p_{i,t} \in D_i (p_{i,t} \prec g_{i,j})\}, \tag{1}$$

where $p_{i,t} \prec g_{i,j}$ represents that the location point $p_{i,t}$ is located in the grid $g_{i,j}$.

Definition 4. Grid-based Spatial Dataset Joinability Determination. Given a spatial dataset D_i, an exemplar dataset D_e, and a joinability threshold $\tau \in \mathbb{Z}^*$, D_i and D_e are joinable if and only if there are at least τ overlapping grids between D_i and D_e. The joinability between D_i and D_e is denoted as $J(D_i, D_e)$,

$$J(D_i, D_e) = \tag{2}$$

where G_i and G_e are the grid distributions of D_i and D_e, respectively.

Definition 5. Grid-based Joinable Coverage Distinction. Given a spatial dataset D_i and an exemplar dataset D_e, with their grid distributions G_i and G_e, the Joinable coverage distinction from D_i to D_e is the number of grids in G_i but not in G_e, denoted as $Dist(D_i, D_e)$,

$$Dist(D_i, D_e) = \tag{3}$$

We give an example to describe the above definition. As shown in Fig. 1,

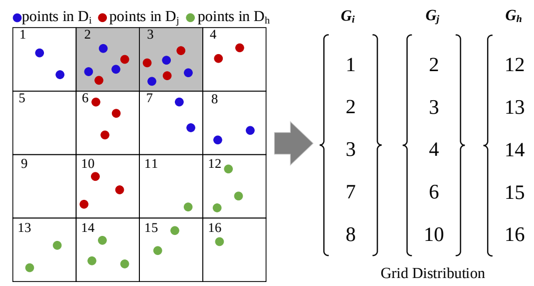


Figure 1: An example of grid-based joinable search model

In summary, the grid-based joinable coverage distinction measurement for spatial datasets can describe the spatial distribution of dataset by partitioning the space into grids and calculate the joinability and coverage distinction for spatial datasets. Compared to direct Euclidean distance calculations between points, this approach enables faster joinability determination while avoiding complex distance computations. Unlike MBRs, it provides finer-grained spatial distribution analysis. Adjusting grid granularity allows flexible search precision in practical applications.

4 The Baseline Search Scheme

In this section, we propose the baseline scheme PSDJS. First, we introduce the vectorization on grid distribution of spatial dataset.Then, we propose a secure joinable coverage distinction comparison. Finally, we provide the search processing of PSDJS.

4.1 Spatial Dataset Vectorization and Encryption

Definition 6. Vectorization on Spatial Dataset.

$$V_{x,i}[j] = \tag{4}$$

We give an example to show the vectorization on the spatial dataset.

Definition 7. G-VecSet and E-VecSet. The outputs of vectorization on a spatial dataset in $\mathcal{D}$ and an exemplar dataset are denoted as G-VecSet and E-VecSet.

- **G-VecSet**: for a spatial dataset $D_i \in \mathcal{D}$, the G-VecSet of D_i is $\mathcal{V}_i = \{V_{i,j} \mid g_{i,j} \in G_i\}$ according to Definition 6, where G_i the grid distribution of D_i.
- **E-VecSet**: for an exemplar dataset D_e, the E-VecSet of D_e is $\mathcal{V}_e = \{V_{e,y} \mid g_{e,y} \in G_e\}$ according to Definition 6, where G_e the grid distribution of D_e.

Definition 8. Encrypted G-VecSet and E-VecSet. Assuming that $\mathcal{V}_i$ and $\mathcal{V}_e$ are respectively the G-VecSet and E-VecSet of the spatial dataset $D_i \in \mathcal{D}$ and the exemplar dataset D_e, the encrypted G-VecSet and E-VecSet denoted as $\widetilde{\mathcal{V}_i}$ and $\widetilde{\mathcal{V}_e}$ are generated by applying the encryption algorithm $EASPE.Enc(\cdot)$ on each $V_{i,j}$ in $\mathcal{V}_i$ and each $V_{e,y}$ in $\mathcal{V}_e$, respecitvely. The encryption on $V_{i,j}$ and $V_{e,y}$ are shown in Eq.(5) and Eq.(6), respectively.

$$\widetilde{V}_{i,j} = EASPE.Enc(V_{i,j}, sk) = \tag{5}$$

$$\widetilde{V}_{e,y} = EASPE.TrapGen(V_{e,y}, sk) = \tag{6}$$

Lemma 1. Given two encrypted vectors $\widetilde{V}_{i,j} \in \widetilde{\mathcal{V}_i}$ and $\widetilde{V}_{e,y} \in \widetilde{\mathcal{V}_e}$ where $\widetilde{\mathcal{V}_i}$ and $\widetilde{\mathcal{V}_e}$ are the encrypted G-VecSet and E-VecSet, respectively,

Proof: According to Definition 8 and EASPE, we have the following deduction.

$$\widetilde{V}_{i,j}^{T} \cdot \widetilde{V}_{e,y}$$

As a result, we have that Lemma 1 holds. ■

4.2 Secure Joinable Coverage Distinction Comparison

In this subsection, we propose the secure joinable coverage distinction comparison and calculate the secure joinable coverage distinction, and then we give a theoretical proof.

Definition 9. Secure Joinability Determinator.

Lemma 2.

$$\widetilde{V}_{i,j} \cdot \widetilde{V}_{e,y} =$$

where G_i and G_e are the grid distributions of the spatial dataset $D_i \in \mathcal{D}$ and the exemplar dataset D_e, respectively,

Proof: According to Lemma 1 and $\widetilde{V}_{i,j} \cdot \widetilde{V}_{e,y} =$, then $V_{i,j} \cdot V_{e,y} =$ is deduced. it is clear that Lemma 2 holds. ■

Theorem 1. Given a spatial dataset $D_i \in \mathcal{D}$ and an exemplar spatial dataset D_e,

$$EJ(D_i, D_e) \geq$$

Proof: According to Definition 9 and Lemma 2,

First, Lemma 2 demonstrates that

Definition 10. Secure Joinable Coverage Distinction. Given a spatial dataset $D_i \in \mathcal{D}$ and an exemplar dataset D_e,

$$EDist(D_i, D_e) = \tag{7}$$

where

Lemma 3. Given an encrypted G-VecSet $\widetilde{\mathcal{V}_i}$ and an encrypted all-ones vector $\widetilde{I}$,

$$\widetilde{V}_{i,j} \cdot \widetilde{I} =$$

where G_i

Proof: According to Lemme 1 and $\widetilde{V}_{i,j} \cdot \widetilde{I}$

Theorem 2. Given two

$$EDist(D_i, D_e) > \tag{8}$$

where $\widetilde{\mathcal{V}_i}$ and $\widetilde{\mathcal{V}_j}$

Proof: According to Definition 10, Lemma 3 and Theorem 1,

Lemma 3 indicates

4.3 Joinable Search Processing Algorithms

● ***Algorithms in the Setup Module***

$\{sk, K\} \leftarrow GenKey(1^{\xi})$.

$\mathcal{G} \leftarrow GenSD(\mathcal{D}, \theta)$.

$\{\widetilde{\mathcal{V}}, \widetilde{\mathcal{D}}\} \leftarrow EncSD(\mathcal{G}, \mathcal{D}, sk, K)$.

After encryption,

● ***Algorithms in the Search Module***

$Tr \leftarrow GenTrapdoor(D_e, I, \theta, sk, k, \tau)$.

$R \leftarrow JSearch(Tr, \widetilde{\mathcal{V}}, \widetilde{\mathcal{D}})$.

The search processing through two sequential stages: The details are shown in Algorithm 1.

Algorithm 1: JSearch($Tr, \widetilde{\mathcal{V}}, \widetilde{\mathcal{D}}$)

Input: the encrypted spatial data repository $\widetilde{\mathcal{D}}$, the encrypted G-VecSet $\widetilde{\mathcal{V}}$, and a trapdoor Tr

Output: the result set R

Initialize a priority $PQ = \emptyset$, a candidate set $C = \emptyset$, and $R = \emptyset$;

return R

Time Complexity Analysis: Our search scheme

5 The Optimized Search Scheme

In this section,

5.1 Bloom Filter-based Optimization

Definition 11. Bloom Filter-based Spatial Dataset Grid Vector (BG-vector).

Definition 12. Grouped Bloom Filter-based Exemplar Dataset Grid Vector Set (BE-VecSet).

The BG-vector and BE-VecSet share the same

Optimization.

5.2 CG-HVIndex and JCT-based Optimization

Definition 13. Coarse-grained Grid Partition.

Definition 14. Coarse-grained Grid Vectorization. Given the

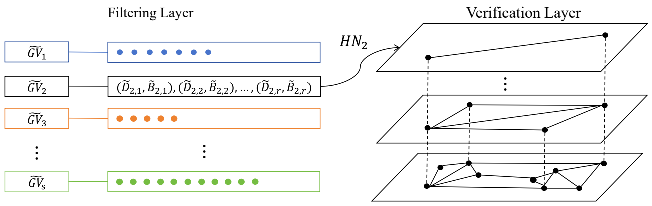
Definition 15. Coarse-grained Grid-based Exemplar Dataset Per-filtering Vector (EP-vector).

Definition 16. Coarse-grained Grid-based High-dimensional Vector Inverted Index (CG-HVIndex). The detailed structure of CG-HVIndex is described as follows:

(1) **Filtering Layer.**

(2) **Verification Layer.**

$$HN = \tag{9}$$



(a) CG-HVIndex

	D_1	D_2	D_3	$\cdots$	$\cdots$	D_r
1	0	σ	0	$\cdots$	$\cdots$	0
2	0	0	0	$\cdots$	$\cdots$	σ
$\vdots$	0	0	σ	$\cdots$	$\cdots$	0
u	σ	σ	0	$\cdots$	$\cdots$	σ
$u+1$	sum(1)	sum(2)	sum(3)	$\cdots$	$\cdots$	sum(r)

(b) JCT

Figure 2: Example of

Next, we present

Definition 17. Joinable Coverage Distinction Check Table (JCT).

$$JCT[r,c] = \tag{10}$$

where $\widetilde{B}_{e,r}$ is

The construction of JCL is

Algorithm 2: BuildIndex($\mathcal{P}^I, \mathcal{D}, \mathcal{G}, SK$)

Input: the coarse-grainded grid partition $\mathcal{P}^I$, the spatial data repository $\mathcal{D}$, the grid distributions $\mathcal{G}$ and the set of secret keys $SK = \{sk, sk', K\}$

Output: the encrypted CG-HVIndex $\widetilde{\mathcal{I}}$

return $\widetilde{\mathcal{I}}$

Optimization. During the search process,

5.3 Optimized Search Processing

We adopt the IHC-index and the above strategies to

$\{sk, sk', K\} \leftarrow GenKey(1^{\xi})$.

$\widetilde{\mathcal{I}} \leftarrow BuildIndex(\mathcal{P}^I, \mathcal{D}, \mathcal{G}, SK)$.

$Tr \leftarrow GenTrapdoor(D_e, \Theta, I, k, u, sk, sk')$.

$R \leftarrow OptJSearch(\widetilde{\mathcal{I}}, Tr)$.

Algorithm 3: OptJSearch($\widetilde{\mathcal{I}}, Tr, \tau$)

Input: the encrypted IHC-index $\widetilde{\mathcal{I}}$, a joinability threshold τ and the trapdoor $Tr = \{\widetilde{\mathcal{F}}, \widetilde{I}, \widetilde{GV}_e, u, k\}$

Output: the result set R

Initialize a priority $PQ = \emptyset$, a candidate set $C = \emptyset$, and a result set $R = \emptyset$;

return R

Time Complexity Analysis: Assuming s is the

6 Security Analysis and Performance Evaluation

6.1 Security Analysis

Our scheme adopts

6.2 Setting

In the paper, we evaluate our scheme using three real-world spatial data repositories

Our scheme with Python and conduct experiments on

For each experiment, we take The are shown in Table 2.

Table 1: Details of spatial data repositories

Data repository	Storage(GB)	Number of datasets
I......	......	......
P......	......	......
T......	......	......

Table 2: Parameter settings

Notations	Meanings	Default values
k	the	20
n	the	120,000
θ	the	9
α	the	1200
σ	the	4
Θ	the	4
u	the	5

6.3 Search Accuracy and Recall rates Evaluation

We use the search result R_{real} under plaintext as the criterion.

Search accuracy and recall versus α.

Search accuracy and recall rates versus σ.

Search efficiency and accuracy versus u.

6.4 Search Efficiency Evaluation

Search time cost versus n and α.

Search time cost versus k.

Search time cost versus θ and Θ.

6.5 Communication Cost Evaluation

Index size and trapdoor size versus n.

Index size and trapdoor size versus θ.

Index size and trapdoor size versus α.

Trapdoor generation time versus α and θ.

7 Conclusion

References

[1] Yuhao Deng, Chengliang Chai, Lei Cao, Qin Yuan, Siyuan Chen, Yanrui Yu, Zhaoze Sun, Junyi Wang, Jiajun Li, Ziqi Cao, Kaisen Jin, Chi Zhang, Yuqing Jiang, Yuanfang Zhang, Yuping Wang, Ye Yuan, Guoren Wang, and Nan Tang. 2024. LakeBench: A Benchmark for Discovering Joinable and Unionable Tables in Data Lakes. *Proc. VLDB Endow.* 17, 8 (2024), 1925–1938.

[2] Yuyang Dong, Kunihiro Takeoka, Chuan Xiao, and Masafumi Oyamada. 2021. Efficient Joinable Table Discovery in Data Lakes: A High-Dimensional Similarity-Based Approach. In *37th IEEE International Conference on Data Engineering, ICDE 2021, Chania, Greece, April 19-22, 2021*. 456–467.

[3] Yuyang Dong, Chuan Xiao, Takuma Nozawa, Masafumi Enomoto, and Masafumi Oyamada. 2023. DeepJoin: Joinable Table Discovery with Pre-trained Language Models. *Proc. VLDB Endow.* 16, 10 (2023), 2458–2470.

[4] Raul Castro Fernandez, Ziawasch Abedjan, Famien Koko, Gina Yuan, Samuel Madden, and Michael Stonebraker. 2018. Aurum: A Data Discovery System. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*. 1001–1012.

[5] Qingqing Gan, Xiaoming Wang, Daxin Huang, Jianwei Li, Dehua Zhou, and Chao Wang. 2022. Towards Multi-Client Forward Private Searchable Symmetric Encryption in Cloud Computing. *IEEE Trans. Serv. Comput.* 15, 6 (2022), 3566–3576.

[6] Qinlong Huang, Guanyu Yan, and Qinglin Wei. 2023. Attribute-Based Expressive and Ranked Keyword Search Over Encrypted Documents in Cloud Computing. *IEEE Trans. Serv. Comput.* 16, 2 (2023), 957–968.

[7] Xinyu Lei, Alex X Liu, Rui Li, and Guan-Hua Tu. 2019. SecEQP: A secure and efficient scheme for SkNN query problem over encrypted geodata on cloud. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. 662–673.

[8] Cong Li, Xinyu Feng, Qingni Shen, and Zhonghai Wu. 2024. On the Security of Secure Keyword Search and Data Sharing Mechanism for Cloud Computing. *IEEE Trans. Dependable Secur. Comput.* 21, 4 (2024), 4306–4308.

[9] Pengyue Li, Hua Dai, Sheng Wang, Wenzhe Yang, and Geng Yang. 2024. Privacy-preserving Spatial Dataset Search in Cloud. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*, Edoardo Serra and Francesca Spezzano (Eds.). 1245–1254.

[10] Xiaoguo Li, Tao Xiang, Shangwei Guo, Hongwei Li, and Yi Mu. 2021. Privacy-preserving reverse nearest neighbor query over encrypted spatial data. *IEEE Transactions on Services Computing* 15, 5 (2021), 2954–2968.

[11] Yury A. Malkov and Dmitry A. Yashunin. 2020. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. *IEEE Trans. Pattern Anal. Mach. Intell.* 42, 4 (2020), 824–836.

[12] Yinbin Miao, Guijuan Wang, Xinghua Li, Hongwei Li, Kim-Kwang Raymond Choo, and Robert H. Deng. 2025. Efficient and Secure Geometric Range Search Over Encrypted Spatial Data in Mobile Cloud. *IEEE Trans. Mob. Comput.* 24, 3 (2025), 1621–1635.

[13] Yinbin Miao, Yutao Yang, Xinghua Li, Zhiquan Liu, Hongwei Li, Kim-Kwang Raymond Choo, and Robert H Deng. 2023. Efficient privacy-preserving spatial range query over outsourced encrypted data. *IEEE Transactions on Information Forensics and Security* 18 (2023), 3921–3933.

[14] Yinbin Miao, Yutao Yang, Xinghua Li, Linfeng Wei, Zhiquan Liu, and Robert H. Deng. 2024. Efficient Privacy-Preserving Spatial Data Query in Cloud Computing. *IEEE Trans. Knowl. Data Eng.* 36, 1 (2024), 122–136.

[15] Fuyuan Song, Zheng Qin, Liang Xue, Jixin Zhang, Xiaodong Lin, and Xuemin Shen. 2021. Privacy-preserving keyword similarity search over encrypted spatial data in cloud computing. *IEEE Internet of Things Journal* 9, 8 (2021), 6184–6198.

[16] Xiangyu Wang, Jianfeng Ma, Ximeng Liu, Robert H. Deng, Yinbin Miao, Dan Zhu, and Zhuoran Ma. 2020. Search Me in the Dark: Privacy-preserving Boolean Range Query over Encrypted Spatial Data. In *39th IEEE Conference on Computer Communications, INFOCOM 2020, Toronto, ON, Canada, July 6-9, 2020*. 2253–2262.

[17] Yuexiang Xie, Zhen Wang, Yaliang Li, Bolin Ding, Nezihe Merve Gürel, Ce Zhang, Minlie Huang, Wei Lin, and Jingren Zhou. 2021. FIVES: Feature Interaction Via Edge Search for Large-Scale Tabular Data. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. 3795–3805.

[18] Guowen Xu, Hongwei Li, Yuanshun Dai, Kan Yang, and Xiaodong Lin. 2018. Enabling efficient and geometric range query with access control over encrypted spatial data. *IEEE Transactions on Information Forensics and Security* 14, 4 (2018), 870–885.

[19] Wenzhe Yang, Sheng Wang, Zhiyu Chen, Yuan Sun, and Zhiyong Peng. 2025. Joinable Search over Multi-Source Spatial Datasets: Overlap, Coverage, and Efficiency. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*. 585–598.

[20] Wenzhe Yang, Sheng Wang, Yuan Sun, Zhiyu Chen, and Zhiyong Peng. 2023. Efficient Spatial Dataset Search over Multiple Data Sources. *arXiv preprint arXiv:2311.13383* (2023).

[21] Chengyuan Zhang, Ying Zhang, Wenjie Zhang, and Xuemin Lin. 2016. Inverted Linear Quadtree: Efficient Top K Spatial Keyword Search. *IEEE Trans. Knowl. Data Eng.* 28, 7 (2016), 1706–1721.

[22] Songnian Zhang, Suprio Ray, Rongxing Lu, Yunguo Guan, Yandong Zheng, and Jun Shao. 2022. Efficient and privacy-preserving spatial keyword similarity query over encrypted data. *IEEE Transactions on Dependable and Secure Computing* 20, 5 (2022), 3770–3786.

[23] Erkang Zhu, Dong Deng, Fatemeh Nargesian, and Renée J. Miller. 2019. JOSIE: Overlap Set Similarity Search for Finding Joinable Tables in Data Lakes. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*. 847–864.