

Location Based Authentication of kNN Queries To Reduce The CPU Time

P. Sathish^{1*} and S.Venkateswaran²

^{1*,2}*Department of Computer Science and Engineering,
Arasu Engineering College, kumbakonam (TN)- India*

www.ijcseonline.org

Received: Nov /5/2015

Revised: Nov/10/2015

Accepted: Nov/20/2015

Published: Nov/30/ 2015

Abstract— In outsourced spatial databases, the LBS provides query services to the clients on behalf of the data owner. However, the LBS provider is not always trustworthy and it may send incomplete or incorrect query results to the clients. Therefore ensuring spatial query integrity is critical. Efficient kNN query verification techniques which utilize the influence zone to check the integrity of query results. kNN query continuously reports the k results (restaurants) nearest to a moving query point. In order to minimize the communication cost between the service provider and the mobile client, a framework for authenticating both the query results and the safe regions of moving kNN queries is proposed. The proposed method can perform moving kNN query authentication with small communication costs and overhead.

Keywords— Query processing, security, integrity, nearest neighbor and protection

I. INTRODUCTION

Location-Based service providers (LBS) offer remote mobile clients with querying services on points-of-interest (e.g., restaurants, cafes, gas stations). A mobile client q issues a moving k nearest neighbor (kNN) query [14] in order to find k points-of-interest closest to q continuously while traveling. Such queries have numerous mobile applications. For example, a tourist may issue a moving kNN query to obtain k nearest restaurants continuously when walking in a city. A driver issues a moving kNN query to find k nearest gas stations continuously while driving. LBS that offer kNN querying services often return mobile clients a safe region [14] in addition to the query results. Given a moving client q , its safe region contains all possible query locations that have the same results as q . In other words, the client only issues a new query to the LBS (for the latest results) when she leaves the safe region. This optimization significantly reduces the communication frequency between the service provider and the clients. Unfortunately, the query results and safe regions returned by LBS may not always be accurate. For instance, a hacker may have infiltrated the LBS's servers so that results of kNN queries all include a particular location (e.g., the White House). Furthermore, it is possible that the LBS is self-compromised, and thus ranks sponsored facilities higher in its query results. The LBS may also return an overly large safe region to the clients for the sake of saving computing resources and communication bandwidth [17], [21]. On the other hand, the LBS may opt to return overly

small safe regions so that the clients have to request new safe regions more frequently, if the LBS charges fee for each request, or if the LBS wish to boost its request rate.

Recently, techniques for authenticating query results have received a lot of attentions [9], [10], [16], [17], [18], [19], [20]. Most authentication techniques are based on Merkle tree [13], which is an authenticated data structure (ADS) for ensuring the correctness of query results on a data set. Recently, Yang et al. developed an ADS called Merkle R-tree (MR-tree) for authenticating queries on a spatial data set, and also an improved tree called MR*-tree. Upon receiving a query issued by a mobile client, the LBS not only retrieve the query results but also compute a verification object from the tree. Specifically, the VO consists of certain tree entries that can be later utilized by the client to verify the correctness of results. The issue of authenticating moving kNN queries, however, has not been addressed yet. Existing authentication techniques for static spatial queries [5], [19] have their authentication target as the query results, being a subset of the data set. In contrast, the authentication target of moving queries includes the safe region, which is a geometric shape computed by the LBS at runtime but not part of the data set. Since a safe region is defined based on both query results as well as points not in the query results, the missing of a non-result point in the VO may also fail the authentication of the safe region. Thus, the above techniques cannot help in authenticating moving kNN queries. This paper is devoted to addressing this challenging issue of authenticating moving kNN queries. In this paper, we improve the best authentication method and prove that it achieves VO-optimality. This

Corresponding Author: *Sathish*

Department of CSE, Arasu Engineering College, Kumbakonam, India

optimality notion guarantees that the VO contains the minimum data points and tree entries (with respect to the given tree). We also present new optimization techniques for reducing the computation cost and the communication cost of our authentication method. It is especially important to minimize the mobile client's total communication cost as it translates to the client's money (paid to the mobile network provider).

II. RELATED WORK

Authentication techniques have been developed for a variety of queries, including relational queries, sliding window queries, spatial queries, text similarity queries, shortest path queries, moving kNN queries, moving range queries, and subgraph search. However, all existing Authenticated Data Structures (ADS) are either inapplicable or inefficient, since the authentication of kNN queries involves verifying both spatial proximity and text relevance. Moreover, authenticating an kNN query includes verifying both the top-k result and the accompanying safe zone. The safe zone is calculated based on both the objects in the top-k result and the objects not in the top-k result, so that missing a non-result object may cause a safe zone to fail in the authentication. Although authentication techniques for moving kNN queries and moving range queries involve safe zone verification, the safe zone of an kNN query is very different.

Authentication Framework

Authentication consists of two phases, i.e., initialization and query processing & authentication. In the initialization phase, the DO first gets a private key from a key distribution center. Next, it signs the ADS constructed on the data set using the private key and transfers the ADS and signatures to the SP. A client downloads a public key from the key distribution center and the signatures from the SP. In the query processing and authentication phase, the client first issues an kNN query. Upon receiving the query, the SP computes the top-k result, the safe zone, and a verification object (VO) that encodes the query result and its safe zone. The client gets the VO from the SP. The top-k result RS and its safe zone $_k(RS)$ are obtained from the VO. The correctness of the top-k result and the safe zone can be verified by the client using the VO, the signatures, and the public key. The client needs to send a new request to the SP only when it leaves the safe zone. When the query moves across the boundary of a safe zone, it requests an updated top-k result and corresponding safe zone. Therefore, authenticating an kNN query is equivalent to verifying the correctness of both the top-k result RS and the corresponding safe zone. Moreover The SP is the potential adversary. The SP is outside the administrative scope of the DO and thus cannot be trusted. With the exception of the DO's private key, adversaries are assumed to know all

information, including the public key for the secure-hash function, the ADS, the signatures, and the authentication algorithms. They may alter the data set or the ADS, and they may tamper with the search result

III. PROPOSED SYSTEM

This paper is devoted to addressing this challenging issue of authenticating moving kNN queries. This paper, improve the best authentication method and prove that it achieves VO-optimality. This optimality notion guarantees that the VO contains the minimum data points and tree entries (with respect to the given tree) .Also present new optimization techniques for reducing the computation cost and the communication cost of our authentication method. It is especially important to minimize the mobile client's total communication cost as it translates to the client's money (paid to the mobile network provider).

Advantages of Proposed System

- 1) Computation optimization that reduces the server and the client CPU time
- 2) VO compression that reduces the size of each VO
- 3) Authentication method achieves low communication cost and CPU overhead

SYSTEM DESIGN

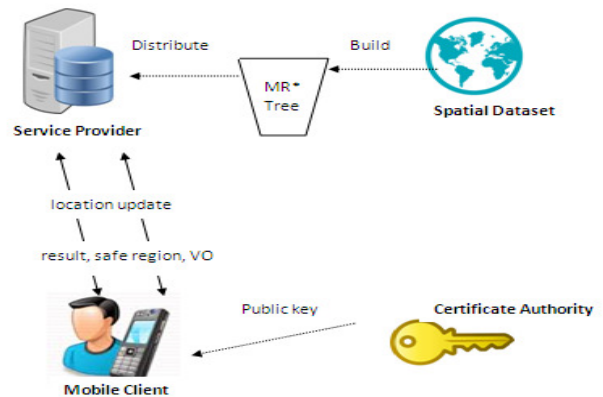


Fig:1 System Architecture

Optimization

Vertex-Based Method (Server)

Algorithm 1. Vertex-Based Method (Server)

- Receive from client: (Query point q , Integer k)
 Using MR-tree/MR*-tree T_D (on data set D)
- 1: $S :=$ compute the kNN of q from the tree T_D ;
 - 2: compute $V_k(S, D)$ from the tree T_D ;
 - 3: $\gamma := \max_{p \in S} \text{dist}(q, p)$; authenticate kNN
 - 4: $KVR := \odot(q, \gamma)$;

- 5: $\psi :=$ set of vertices of $V_K(S,D)$; authenticate safe region
- 6: $SRVR := \cup \psi \in \psi^*(\psi, \max p \in S \text{ dist}(\psi,p))$
- 7: $\Gamma := KVR \cup SRVR$;
- 8: $VO := \text{DepthFirstRangeSearch}(T_D, \text{root}, \Gamma)$;
- 9: send VO to the client;

Algorithm 1 is the pseudo-code of the server algorithm. Upon receiving the client location q and the number k of required NNs, it computes the kNN result S from an MRtree/MR*-tree. Then, it computes the safe region. Next, it defines verification region G so as to identify useful k points for verifying the kNN result and the safe region, and puts these points into the VO. Specifically, the verification region G is defined as the union of (i) the kNN query result verification region (KVR) and (ii) the kNN safe region verification region (SRVR).

Vertex-Based Method (Client)

Algorithm 2. Vertex-Based Method (Client)

- Receive from server: (Verification object VO)
- 1: $h'_{\text{root}} :=$ reconstruct the root digest from VO;
 - 2: verify h'_{root} against the tree root signature;
 - 3: if h'_{root} is correct then
 - 4: $D' :=$ the set of data points extracted from VO;
 - 5: $R' :=$ the set of non-leaf entries extracted from VO;
 - 6: $S' :=$ compute the kNN of q from D' ;
 - 7: $\gamma' := \max p \in S' \text{ dist}(q,p)$;
 - 8: if $\forall e \in R', e.n^{\circ}(q, \gamma') = \theta$ then
 - 9: $V :=$ compute $V_K(S', D')$;
 - 10: $\psi :=$ set vertices of V ;
 - 11: $SRVR := \cup \psi \in \psi^*(\psi, \max p \in S' \text{ dist}(\psi,p))$;
 - 12: if $\forall e \in R', e.n \text{ SRVR} = \theta$ then
 - 13: return kNN result S' and safe region V ;
 - 14: return authentication failed;

Algorithm 2 is the pseudo-code of the client algorithm. Upon receiving the verification object VO from the server, it first reconstructs the root digest from the VO and verifies it against the tree root signature signed by the map provider. If the verification is successful, the VO is guaranteed to contain only entries from the original tree (i.e., no fake entries). Next, it proceeds to verify the correctness of the kNN result and the safe region provided by the VO. It extracts from the VO (i) a set D' of data points, and (ii) a set R' of non-leaf entries, and then computes the kNN result S' from D' .

IV. OUTPUT SCREENSHOTS

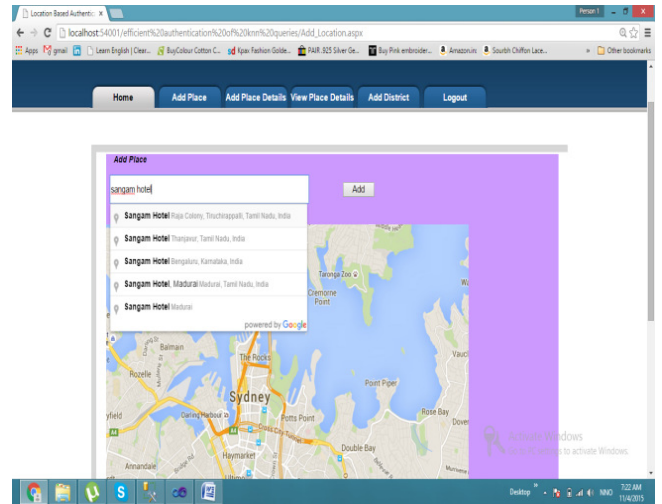


Fig:2 Adding Location Details



Fig:3 Location Based Search

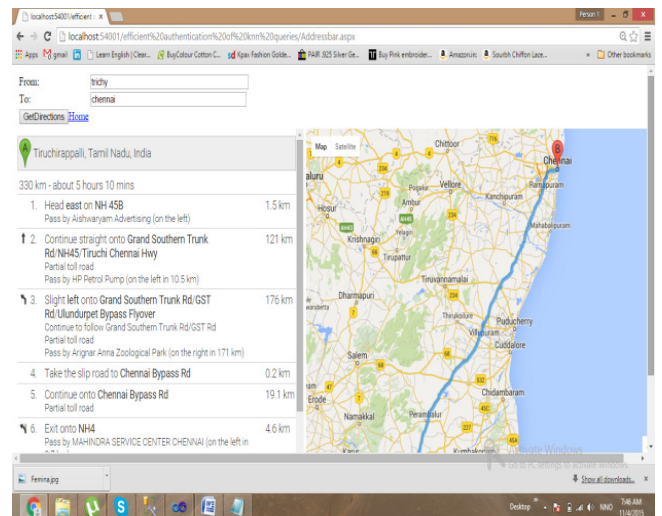


Fig:3 Google Map View

V. EXPERIMENTAL EVALUATION

Evaluate all methods on spatial real data. Each point-of-interest stores a lat-long coordinate (16 bytes) and a full geographic name (250 bytes). For dataset, MR*-tree for it with the page size as 4 Kbytes is build. Clients issue kNN queries with the default value of k as 10.

All experiments were run on a 2.5 GHz Intel PC running Windows with 2 GB of RAM. In each experiment, the average performance measure (e.g. server and client CPU time) per client journey per timestamp is reported.

Server Cpu Time

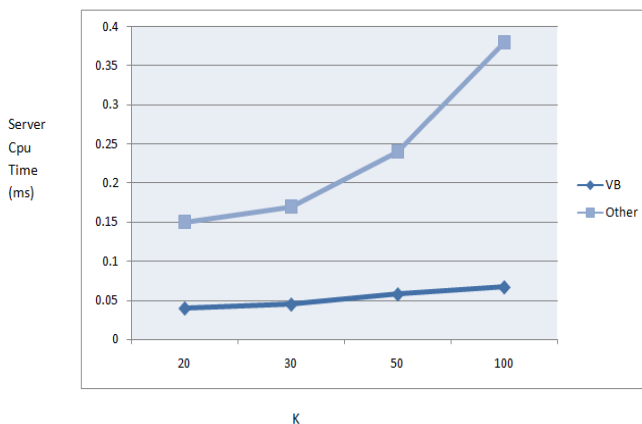


Fig:4 cpu time (server)

Client Cpu Time

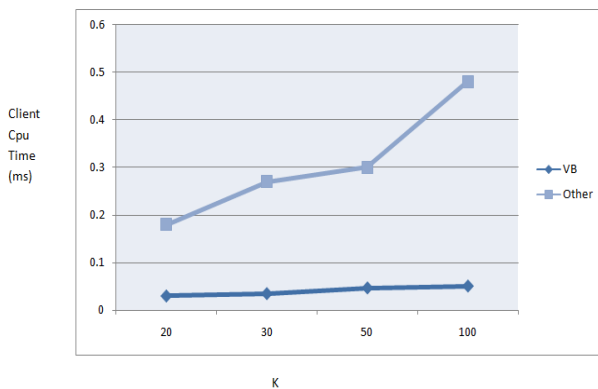


Fig:5 cpu time (client)

VI. CONCLUSION

Proposed an efficient method to authenticate moving kNN queries. Result proved that the method is VO-optimal, i.e., the verification object has the minimal size with respect to the given tree. Developed optimization techniques that can further reduce the computation cost, communication

frequency and cost between a moving client and the LBS. Furthermore, extended the solution to handle moving kNN queries that involve multiple data sets. Experimental results show that the authentication method achieves low communication cost and CPU overhead. An interesting future work is to avoid disclosing non-result points to the query client. It is challenging to authenticate a safe region as its vertices and edges may allow an adversary to infer some non-result points.

REFERENCES

- [1] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The R*-tree: An efficient and robust access method for points and rectangles," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 1990, pp. 322–331.
- [2] N. Bruno, S. Chaudhuri, and L. Gravano, "Stholes: A multidimensional workload-aware histogram," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2001, pp. 211–222.
- [3] M. A. Cheema, L. Brankovic, X. Lin, W. Zhang, and W. Wang, "Continuous monitoring of distance based range queries," IEEE Trans. Knowl. Data Eng., vol. 23, no. 8, pp. 1182–1199, Aug. 2010.
- [4] H. Hu, J. Xu, Q. Chen, and Z. Yang, "Authenticating locationbased services without compromising location privacy," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2012, pp. 301–312.
- [5] L. Hu, W.-S. Ku, S. Bakiras, and C. Shahabi, "Verifying spatial queries using voronoi neighbors," in Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inform. Syst., 2010, pp. 350–359.
- [6] G. S. Iwerks, H. Samet, and K. P. Smith, "Maintenance of K-nn and spatial join queries on continuously moving points," ACM Trans. Database Syst., vol. 31, no. 2, pp. 485–536, 2006.
- [7] A. Kundu and E. Bertino, "Structural signatures for tree data structures," Proc. VLDB Endowment, vol. 1, no. 1, pp. 138–150, 2008.
- [8] A. Kundu and E. Bertino, "How to authenticate graphs without leaking," in Proc. 13th Int. Conf. Extending Database Technol., 2010, pp. 609–620.
- [9] F. Li, M. Hadjieleftheriou, G. Kollios, and L. Reyzin, "Dynamic authenticated index structures for outsourced databases," in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2006, pp. 121–132.
- [10] F. Li, K. Yi, M. Hadjieleftheriou, and G. Kollios, "Proof-infused streams: Enabling authentication of sliding window queries on streams," in Proc. 33rd Int. Conf. Very Large Data Bases, 2007, pp. 147–158.
- [11] X. Lin, J. Xu, and H. Hu, "Authentication of location-based skyline queries," in Proc. 20th ACM Int. Conf. Inform. Knowl. Manage., 2011, pp. 1583–1588.
- [12] X. Luo, P. Zhou, E. W. W. Chan, W. Lee, R. K. C. Chang, and R. Perdisci, "Httpos: Sealing information leaks with browser-side obfuscation of encrypted

- flows,” in Proc. Netw. Distrib. Syst. Security Symp., 2011.
- [12] X. Luo, P. Zhou, E. W. W. Chan, W. Lee, R. K. C. Chang, and R. Perdisci, “Httpos: Sealing information leaks with browser-side obfuscation of encrypted flows,” in Proc. Netw. Distrib. Syst. Security Symp., 2011.
- [13] R. C. Merkle, “A certified digital signature,” in Proc. Adv. Cryptology, 1989, pp. 218–238.
- [14] S. Nutanong, R. Zhang, E. Tanin, and L. Kulik, “The V*-Diagram:
A query-dependent approach to moving knn queries,” Proc. VLDB Endowment, vol. 1, no. 1, pp. 1095–1106, 2008.
- [15] A. Okabe, B. Boots, K. Sugihara, and S. Chiu, Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, 2nd ed. Hoboken, NJ, USA: Wiley, 2000.
- [16] H. Pang, A. Jain, K. Ramamritham, and K.-L. Tan, “Verifying completeness of relational query results in data publishing,” in Proc. ACM SIGMOD Int. Conf. Manage. Data, 2005, pp. 407–418.
- [17] H. Pang and K. Mouratidis, “Authenticating the query results of text search engines,” Proc. VLDB Endowment, vol. 1, no. 1, pp. 126–137, 2008.
- [18] H. Pang, J. Zhang, and K. Mouratidis, “Scalable verification for outsourced dynamic databases,” Proc. VLDB Endowment, vol. 2, no. 1, pp. 802–813, 2009.
- [19] S. Papadopoulos, Y. Yang, S. Bakiras, and D. Papadias, “Continuous spatial authentication,” in Proc. 11th Int. Symp. Adv. Spatial Temporal Databases, 2009, pp. 62–79.
- [20] S. Papadopoulos, Y. Yang, and D. Papadias, “CADS: Continuous authentication on data streams,” in Proc. 33rd Int. Conf. Very Large Data Bases, 2007, pp. 135–146.
- [21] R. Sion, “Query execution assurance for outsourced databases,” in Proc. 31st Int. Conf. Very Large Data Bases, 2005, pp. 601–612.
- [22] Z. Song and N. Roussopoulos, “K-nearest neighbor search for moving query point,” in Proc. 7th Symp. Spatio-Temporal Database, 2001, pp. 79–96.
- [23] Y. Tao, D. Papadias, and Q. Shen, “Continuous nearest neighbor search,” in Proc. 28th Int. Conf. Very Large Data Bases, 2002, pp. 287–298.

AUTHORS PROFILE

P.Sathish has received M.SC(SOFTWARE ENGG) degree in the department of Computer Science and Engineering from Annamalai university in the year of 2012. Now he is pursuing M.E degree in the department of Computer Science and Engineering in Arasu Engineering College. His area of interest is Mobile Computing , Cloud Computing, Operating System, Image Processing and Network Security.

S.Venkateswaran has received BE in the department of Computer Science and Engineering from Arasu Engineering College in the year 2005 and M.Tech in the department of Computer Science and Engineering from Veltech University in the year of 2012. And now he is working as an assistant professor in Arasu Engineering College and also have 6 years of teaching experience in the same Institution. His area of Interest is Mobile Computing, Network security, Wireless Network and Image Processing.
