

숨은 객체 식별을 위한 향상된 공간객체 탐색기법

김종완¹ · 조양현^{2*}

An Advanced Scheme for Searching Spatial Objects and Identifying Hidden Objects

Jongwan Kim¹ · Yang-Hyun Cho^{2*}

¹Professor, Department of Computer Engineering, Sungkyul University, Seoul 139-742, Korea

^{2*}Professor, Division of Computer Science, Sahmyook University, Seoul 139-742, Korea

요 약

본 논문은 주변탐색(Surround Search: SuSe)이라는 새로운 공간질의 방법을 제안한다. 이 기법은 현재 사용자의 위치를 중심으로 주변에서 가까운 관심영역의 공간객체를 탐색하는 것이다. 사용자 중심의 주변탐색은 증강현실과 같이 사용자가 관심 있어 하는 공간객체 중 가까운 것을 찾기 때문에 기존의 공간질의와 구별된다. 기존 기법은 질의 점과 객체 사이의 최단거리(MINDIST)를 기준으로 주변을 탐색하지만 제안 기법에서는 객체들 사이에 숨어있지만 관심의 대상인 숨은 객체를 식별하기 위해서 각도(Angle)를 함께 고려하여 탐색한다. 제안 기법의 특징은 기존기법이 거리만을 사용하여 가까운 객체를 탐색한 것과 달리 거리는 멀지만 숨은 객체까지도 찾아냄으로써 사용자의 선호도를 더 세밀하게 반영한다. 실험결과에서 제안기법인 SuSe는 최근접 이웃 탐색기법인 NN(Nearest Neighbor)과 비교하여 보다 정밀한 공간객체 탐색이 가능하며 향상된 탐색성능을 나타낸다.

ABSTRACT

In this paper, a new method of spatial query, which is called Surround Search (SuSe) is suggested. This method makes it possible to search for the closest spatial object of interest to the user from a query point. SuSe is differentiated from the existing spatial object query schemes, because it locates the closest spatial object of interest around the query point. While SuSe searches the surroundings, the spatial object is saved on an R-tree, and MINDIST, the distance between the query location and objects, is measured by considering an angle that the existing spatial object query methods have not previously considered. The angle between targeted-search objects is found from a query point that is hidden behind another object in order to distinguish hidden objects from them. The distinct feature of this proposed scheme is that it can search the faraway or hidden objects, in contrast to the existing method. SuSe is able to search for spatial objects more precisely, and users can be confident that this scheme will have superior performance to its predecessor.

키워드 : 객체 식별, 근접객체탐색, 증강현실, 공간분석, 공간질의

Key word : Object Identification, Near Object Search, Augmented Reality, Spatial Analysis, Spatial Query

접수일자 : 2014. 05. 18 심사완료일자 : 2014. 05. 30 게재확정일자 : 2014. 06. 09

* **Corresponding Author** Yang-Hyun Cho(E-mail:yhcho@syu.ac.kr, Tel:+82-2-3399-1787)

Division of Computer Science, Sahmyook University, Seoul 139-742, Korea

Open Access <http://dx.doi.org/10.6109/jkiice.2014.18.7.1518>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

모바일 환경 및 GPS의 발전으로 공간 데이터의 활용도가 증가하면서 공간객체의 탐색기법도 다양하게 연구되어 왔다. 특히, 위치기반서비스(LBS: Location Based Service)는 병원, 호텔 등의 위치를 제공하는 서비스로서 공간데이터의 활용이 증가 된다 [1, 2]. 예를 들어, 사용자는 현재 자신이 있는 위치를 중심으로 가장 가까운 병원, 호텔 등의 공간객체를 탐색 한다 [3, 4].

모바일 환경에서의 증강현실은 가까운 공간객체를 식별하여 해당 정보를 출력해주는 대표적인 사례이다. 이와 같은 공간 데이터의 활용은 여행지에서 관심 있는 주변 관광명소를 찾는 경우에도 사용되며 대표적으로 사용자로부터 가까운 위치를 탐색하는 최근접 이웃(NN: Nearest Neighbor)이 사용 된다 [5, 6].

NN은 질의 점으로부터 최근접 이웃을 탐색하면서 최근접거리(MINDIST; Minimum Distance)를 사용하기 때문에 가까운 객체만 탐색한다. NN으로는 주변의 관심 대상을 찾는 경우 거리는 멀지만 두 대상 사이에 가려져있는 또 다른 관심 대상(숨은 객체)을 식별하기 어렵다.

사용자의 관심이란 단순히 거리만이 좌우한다고 할 수 없다. 예를 들어 거리가 멀지만 사용자가 평소에 꼭 방문해 보고 싶었던 명소나 전쟁에서 아군보다 멀리 있으면서 아군에 의해 가려져서 보이지 않는 적군의 식별에서처럼 숨은 객체가 중요한 대상이 되기도 한다. 그러나 최근까지 연구된 NN기법에서는 본 논문의 아이디어를 고려하고 있지 않다.

우리는 본 논문에서 숨은 객체까지 탐색할 수 있는 주변탐색(Surrounder Search: SuSe) 기법을 제안한다. 제안기법은 새롭게 시도되는 새로운 연구로서 객체를 탐색할 때 거리와 두 객체 사이에 숨은 객체를 찾고자 각도(Angle)를 사용한다. 즉, 사용자의 주변에 있지만 숨겨진 관심대상의 공간객체를 탐색하여 사용자의 관심에 밀접한 객체를 탐색하는 기법이다.

기존의 최근접 이웃은 [그림 1]과 같이 NN을 적용하여 탐색하면 NN={Obj2, Obj4, Obj5, Obj6, Obj8, Obj9}의 결과를 갖는다. 이때, 숨은 객체 Obj1, Obj3은 다른 객체보다 멀리 있기 때문에 탐색되지 않는다.

NN의 단점을 극복하고자 제안기법에서는 각 객체를 중심으로 각도를 적용하여 숨은 객체의 탐색이 가능하

도록 하였다. 즉, 제안기법의 한 객체는 $SeSu=\{Obj1, Obj2, Obj3, Obj4, Obj5, Obj6, Obj8, Obj9\}$ 로 탐색된다. 이는 거리가 멀다는 이유로 관심의 대상에서 제외되는 NN의 단점을 극복한 것으로 사용자 중심의 공간객체 탐색기법이다.

논문은 다음과 같이 구성된다. 2절에서는 기존에 연구된 공간 데이터 탐색기법을 살펴보고 기존기법의 적용한계를 설명한다. 3절에서는 제안기법인 SuSe에 대해서 설명한다. 4절은 실험데이터를 이용하여 기존의 탐색기법과 제안기법을 비교 실험한다. 논문의 제안 및 실험에 대한 종합적인 결과는 5절에서 논의한다.

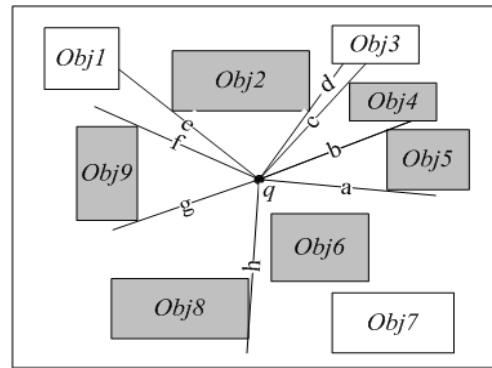


그림 1. 최근접 이웃 탐색
Fig. 1 Nearest Neighbour Search

II. 관련연구

현재 위치에서 가장 가까운 최근접 이웃을 탐색하는 기법은 기존에 NN [7]이 연구되었다. 이 기법은 공간 데이터를 공간색인인 R-tree에 저장하며 [8-10] 질의 점에서부터 각 노드의 MBR까지의 최소근접거리인 MINDIST를 계산하여 최근접 이웃을 탐색한다.

MINDIST는 [그림 2]와 같이 판단하며 수식 (1)과 같이 계산한다. 즉, MINDIST는 R-tree에서 질의 점 q로부터 각 MBR까지의 최소거리를 의미한다. 이때, q가 MBR 안에 있다면 MINDIST=0이 된다. 즉, MINDIST란 다음의 수식과 같이 R-tree에서 질의 점 q와 사각형 R사이의 최소거리를 의미한다.

$$\forall o \in O, \text{MINDIST}(q, R) \leq \|(q, o)\| \quad (1)$$

수식(2)는 MINMAXDIST를 나타내며 이는 사각형 R까지의 가까운 거리 중 가장 먼 거리를 나타낸다. MINMAXDIST는 최근접 이웃(NN)을 이용하여 가장 가까운 k개의 공간객체를 구하는 k-NN에서 사용된다. MINMAXDIST는 다음의 식과 같이 질의 점 q와 사각형 R사이의 MINMAXDIST까지의 거리보다 가깝거나 같은 거리를 갖는 공간객체가 존재하는 것을 의미한다.

$$\exists o \in O, \|(q, o)\| \leq MINMAXDIST(q, R) \quad (2)$$

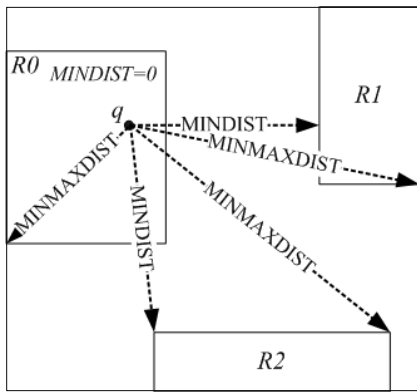


그림 2. 최소근접거리
Fig. 2 Minimum Nearest Distance

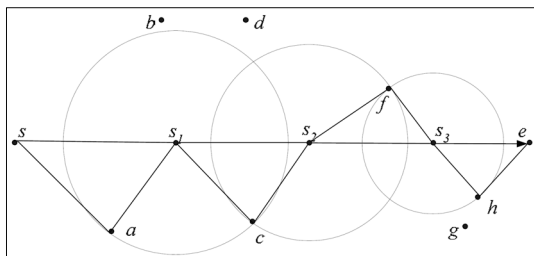


그림 3. 연속 최근접 이웃 탐색
Fig. 3 Continuous Nearest Neighbor Search

정적인 공간객체를 대상으로 탐색하는 NN 기법을 이동객체에 적용한 기법이 CNN(Continuous Nearest Neighbor search) [11]이다. CNN은 [그림 3]과 같이 질의 점이 s에서 e로 이동하는 상황에서 각 NN을 중심으로 최근접 이웃을 탐색한다. 이는 기존에 NN이 고정된 위치에서의 최근접 이웃을 탐색하는 것을 이동환경에 적용한 것이다. 그러나 이 기법은 질의 위치가 이동하

는 것이 NN과 다를 뿐 가장 가까운 거리를 기준으로 탐색한다.

CNN이 두 객체 사이에서 최근접 이웃을 탐색할 때 탐색순서에 따라 성능에 영향을 받는 것에 착안하여 탐색구간을 제한하는 연구로 Slabbed_CNN [12]이 있다. 이는 [그림 4]와 같이 슬랩(Slab)을 이용하여 탐색구간을 제한함으로써 질의 점이 이동하는 경우에도 최근접 이웃을 향상된 탐색성능으로 찾게 한다.

지금까지 살펴본 가장 가까운 객체를 탐색하기 위해 연구된 NN, CNN 그리고 slabbed_CNN은 모두 거리를 기반으로 하기 때문에 두 객체 사이에 가려진 객체를 찾는 주변탐색을 수행하기에는 적합하지 않다.

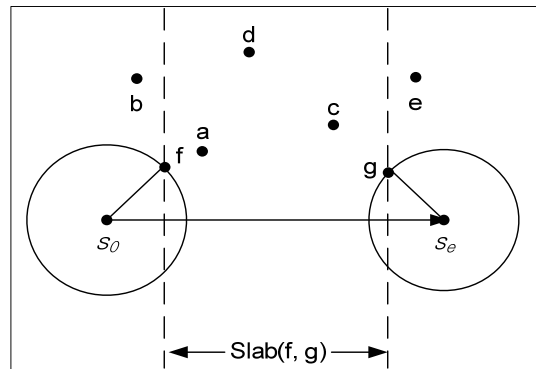


그림 4. 슬랩을 이용한 CNN 구간 제한
Fig. 4 Searching Scope in Slabbed_CNN

III. 주변탐색 기법

모바일 환경에서는 사용자가 모바일 장치와 함께 이동한다. 이동 환경에서 사용자의 위치를 중심으로 최근접 이웃 객체를 탐색하는 것은 위치기반서비스에서 가장 빈번하게 필요한 기능이다. 기존에는 거리를 기반으로 하여 가까운 택시, 호텔 등의 위치를 탐색했다[13-15]. 그러나 놀이 공원과 같이 사용자의 관심대상이 있는 곳에서는 거리만으로 사용자의 관심대상을 탐색한다는 것은 적합하지 않다. 왜냐하면 사용자는 거리가 조금 멀더라도 관심 있는 곳을 찾아갈 것이기 때문이다 [16, 17].

주변탐색기법의 일차적인 탐색방법은 NN과 같이 거리기반의 계산이 필요하다. 이때, 질의 점 q로부터 공간객체 r까지의 거리는 MINDIST로 계산한다. MINDIST

는 유클리드 거리(euclidean distance)로 계산되며 다음과 같이 정의된다.

정의1. MINDIST : 질의 점 q와 사각형 r이 주어졌을 때 q와 r 사이의 최소거리인 MINDIST(q, r)은 다음의 수식(3)과 같다.

$$MINDIST(q, r) = \sum_{i=1}^d |q_i - r_i|^2 \quad (3)$$

각도 기반의 공간객체 탐색은 질의 점으로 부터 각 MBR까지의 각도를 중심으로 한다. 1차적으로는 MINDIST로 주변객체를 탐색하고 각도로 2차적으로 탐색하여 모든 주변객체를 탐색한다[18].

[그림 5]는 세 개의 공간객체를 포함하는 MBR과 질의 점에서의 각도를 나타낸다. 논문에서 각도의 측정은 X축을 기준으로 양(+의 방향으로 진행된다. 정의1에 따라 거리기반으로 최근접 이웃을 탐색하면 Obj2, Obj4, Obj5가된다. 이때, Obj3은 관심대상이 될 수 있음에도 불구하고 거리가 멀다는 이유로 탐색대상이 못된다 [19]. 사용자가 q지점에서 주변의 관심 대상을 모두 탐색하고자 하는 경우 Obj3도 포함되어야 한다. 그러나 Obj3이 다른 객체에 의해 가려졌다면 탐색되지 않는다. 본 논문에서는 [그림 5]와 같이 한 MBR에 대해 rs~re까지 각을 구성한다. 그러나 x축이 0도 이므로 질의 점으로 부터 내부 MBR들의 각은 $\angle\Theta = \{(x, b), (b, c), (c, d), (d, re)\}$ 이다. re는 MBR Obj2와 겹치므로 각으로 인정될 수 없으며 d부터 Obj2의 왼쪽 모서리까지 새로운 각이 생성된다.

탐색알고리즘은 [그림 6]과 같이 거리기반의 탐색과 각도기반의 탐색을 수행하여 주변객체를 탐색한다. 질의 점 q와 R-tree로 색인된 데이터집합(dataset)을 받아들인다 [20]. 질의 점은 데이터집합의 모든 공간객체인 Objn과 비교하여 거리상 가장 가까운 최근접 이웃을 찾고 ObjList 리스트 배열에 저장한다(1-3줄).

거리를 기반으로 찾은 공간객체들은 각 객체들 사이에 각이 존재하는지 비교하기 위해 다시 반복 문으로 처리한다(5줄). 두 객체 사이에 각이 존재하면 해당 위치에서의 주변객체를 한 번 더 찾아 ObjList에 추가한다 (6-10줄). 거리와 각도에 따라 탐색된 공간객체는 최종적으로 ObjList에 저장된다.

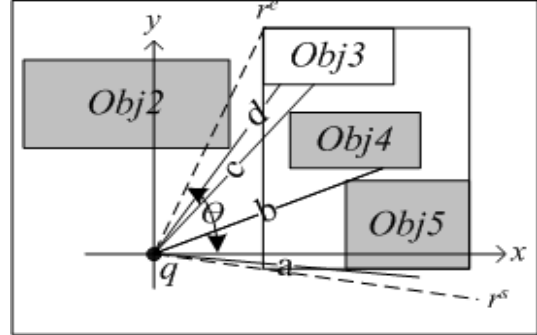


그림 5. MBR의 각도 계산
Fig. 5 Angle Computation of MBR

```

Algorithm ObjectSearch(q, dataset)
Input: q is a query point.
       dataset is a set of spatial Objects.
Output: a set of Objects
1: do
2: ObjList+=MINDIST(q, dataset);
3: loop
4: if (ObjList <> NULL) then
5: do
6: angle = start angle of Obji + ending
  angle of ∠Obji+1;
7: if(angle - (∠Obji+∠Obji+1) > 0) then
8: Search a new MBR between Obji
  and ∠Obji+1;
9: ObjList += a new MBR;
10: endif
11: loop
12: endif
13: return ObjList;
    
```

그림 6. 객체 탐색 알고리즘
Fig. 6 Object Search Algorithm

IV. 성능 평가

논문에서 제안한 기법은 기존의 최근접 이웃 탐색기법인 NN과 비교 하였다. 시뮬레이션을 위한 성능평가 항목 및 환경은 다음 표와 같다.

표 1. 성능평가 항목 및 환경

Table. 1 Evaluation lists and environment

구분	내용
노드접근 횟수	R-tree에서 주변객체를 찾기 위해 노드를 방문한 횟수
탐색 시간	주변객체를 탐색하는데 걸리는 시간
탐색된 주변객체 수	숨은 객체를 포함하여 질의 점으로부터 탐색된 주변객체의 수
시스템	Windows XP, 4MB 램
데이터 집합	실험용 데이터집합 10만개, 표준분포, 랜덤 데이터, 분포공간:3,000x3,000

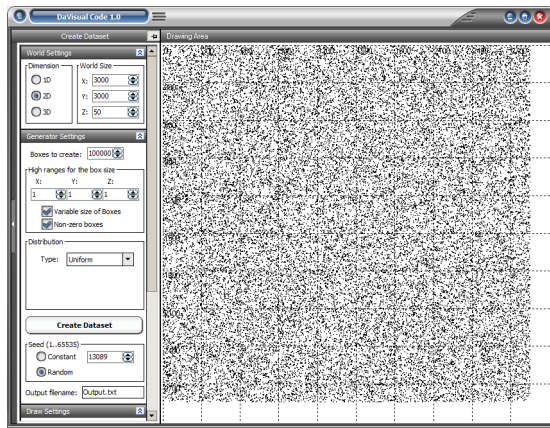


그림 7. 실험 데이터 집합
Fig. 7 Dataset for Simulation

공간객체는 [그림 7]과 같이 공간 데이터 생성 프로그램인 DaVisual Code1.0 [20]을 사용하여 10만개의 무작위 데이터를 생성하였다. 생성된 데이터는 R-tree로 색인을 구성한 후 탐색과정을 거쳐서 탐색횟수, 탐색시간 그리고 탐색되는 주변객체의 수를 비교하였다. 실험데이터에서 공간의 크기 3,000x3,000은 공간객체들이 분포될 범위이다.

논문에서 제안한 SuSe기법을 다양한 최근접 이웃 탐색기법들 중 NN과 비교한 이유는 서두에서 논의한 CNN과 Slabbed_CNN이 NN을 응용하여 새로운 속성을 추가하였으나 여전히 주변탐색에는 도움을 주지 못하기 때문에 순수한 NN을 대상으로 한 것이다. NN을 고려한 자세한 이유는 다음과 같다. SuSe기법이 각도를 사용하여 주변의 객체를 탐색하기 때문에 탐색횟수와

탐색시간에 있어서 NN 보다 더 느릴 수 있다. 그러나 본 실험에서 NN과 비교한 첫 번째 이유는 현재까지 주변을 탐색하는 기법에 가장 가까운 것이 NN이기 때문이며, 둘째, 각도를 추가적으로 사용하여 비교하지만 NN보다 성능이 많이 나쁘지 않다는 것을 보여주기 위한 것이다.

다음의 [그림 8], [그림 9] 그리고 [그림 10]은 공간객체를 각각 10K~100K까지 일정한 묶음을 구분해서 질의 점을 중심으로 주변의 가까운 1000개의 객체를 찾는 질의에 대한 세 가지 실험의 결과이다.

[그림 8]에서 공간객체를 탐색할 때 데이터 수가 많을수록 SuSe의 노드접근횟수가 작아진다. 이는 한번 접근한 노드에서 주변객체를 탐색하게 되는 경우가 많아지므로 NN보다 적은 노드접근횟수를 나타내는 것이다. 그러나 주변객체를 탐색하는 데 걸리는 시간은 SuSe가 조금 많다. 이는 숨어 있는 객체들을 찾기 위해 각도를 계산하기 때문이다.

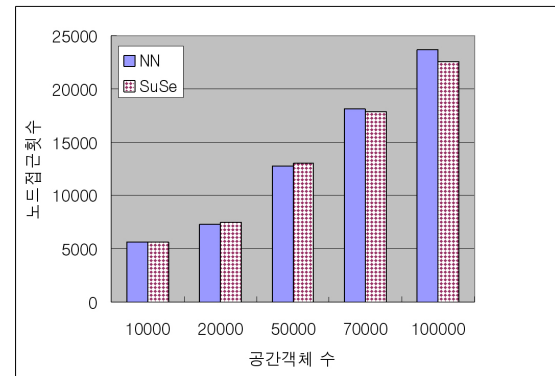


그림 8. 노드 접근 횟수
Fig. 8 Number of Node Accesses

[그림 9]는 NN보다 약간의 시간 지연이 발생하는 그림이다. 10만개의 데이터집합에서 NN과 SuSe는 탐색시간에 있어서 근소한 차이를 나타낸다. 논문에서 제안한 주변탐색기법의 객체탐색은 NN보다 더 많이 찾는다. 이 기법은 단순히 가장 가까운 거리의 대상을 찾는 것이 아니라 질의 점 주변에 있으면서 접근이 가능한 모든 주변객체를 찾는 것이다.

[그림 10]과 같이 실제 탐색되는 주변객체는 NN보다 많지만 탐색성능(노드접근횟수 및 시간)에서는 NN보다 많이 나쁘지는 않다. 즉, 관심대상이 되는 주변객

체를 충분히 탐색하면서도 NN과 비슷한 성능을 나타낸다.

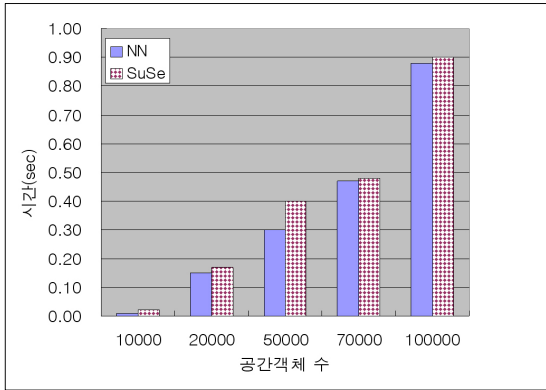


그림 9. NN과 SuSe의 탐색 시간
Fig. 9 Search Time of NN and SuSe

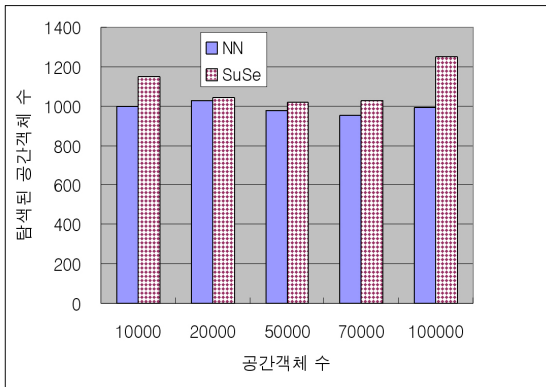


그림 10. 탐색된 주변객체 수
Fig. 10 Number of Surround Objects

V. 결론

모바일 사용자가 자신의 위치에서 주변을 탐색하는 것은 일반적으로 가장 가까운 곳에 있는 한 장소를 찾는 것이다. 그러나 거리가 가까운 곳만 사용자의 관심 대상이라고 할 수는 없다. 즉, 거리상으로는 다른 객체들 보다 멀지만 사용자의 관심 대상이 있다면 이 객체도 주변탐색결과에 포함되어야 한다. 예를 들어 2차원 공간에서 게임을 수행할 때 이 기법을 적용하면 숨어 있는 상대 팀을 보다 많이 탐색할 수 있다.

실험결과와 같이 제안기법은 NN 보다 노드 접근횟수나 탐색시간에서 약간의 성능저하를 보이지만 이는 매우 근소한 차이로 나타났다. 마지막 실험에서는 NN 보다 더 많은 주변객체를 탐색하였으며 이는 사용자의 선호도를 반영한 숨은 객체가 함께 탐색됨을 의미한다. 결과적으로 SuSe기법은 NN과 비슷한 성능을 보이면서도 더 많은 주변객체를 탐색할 수 있으며 전쟁터와 같은 상황에서 주변의 적들을 보다 정확히 탐색하게 된다. 향후 본 제안기법은 공간게임이나 관광명소 탐색 등의 시스템에서 활용되면 보다 풍부한 정보를 제공하는 알고리즘으로 사용될 수 있을 것이다.

REFERENCES

- [1] J. Kim, S. J. Im, S. W. Kang, C. S. Hwang, S. K. Lee, "SQR-tree : A Spatial Index Using Semi-quantized MBR Compression Scheme in R-tree", *Journal of Information Science and Engineering(JISE)*, Vol. 23, No. 5, pp. 154-156, 2007.
- [2] H. S. Joo, J. W. Kim, "Spatial Data Compression for Location-Based Game", *ICCC2010 International Conference on Convergence Contents*, pp. 365-366, Dec. 2010.
- [3] O. S. Park, Jung, K. R., Kim, S. H., "Location Sensing Tech. and System for Ubiquitous Computing," *Weekly Technical Trend*, Vol. 1098, pp. 11-21, 2003.
- [4] K. Y. Lee, D. O. Kim, "Design of a Location Management System in the Ubiquitous Computing Environments," *Journal of KIISE*, Vol.12, No.6, pp.115-121, Dec. 2007.
- [5] M. Weiser, "Ubiquitous Computing," *ACM Conference on Computer Science*, Vol. 26. No. 10, pp. 418-438, 1993.
- [6] H. H. Kim, "Techniques on Multi-Marker for the Implementation of Augmented Reality," *Journal of KIISE*, Vol.15, No.11, pp.109-116, Nov. 2010.
- [7] N. Roussopoulos, S. Kelly, and F. Vincent. "Nearest Neighbor Queries," *ACM SIGMOD*, 1995.
- [8] A. Guttman, "R-tree : A Dynamic Index Structure for Spatial Searching," *ACM SIGMOD*, 1984.
- [9] T. Sellis, N. Roussopouls, C. Faloutsos, "The R+-tree : A Dynamic Index for Multi-Dimensional Objects," *VLDB*, 1987.
- [10] D. S. Kwon, "Protection of Location Privacy for Spatio-Temporal Query Processing Using R-Trees," *Journal of Society for E-Business Studies*, Vol.15, No.3, pp.85-98,

- Aug. 2010.
- [11] Y. Tao, D. Papadias, and Q. M. Shen. "Continuous Nearest Neighbor Search," *Proceeding of VLDB '02*, 2002.
- [12] S. Han, J. Kim, "A Search Interval Limitation Technique for Improved Search Performance of CNN," *journal of Korean Society for Internet Information*, Vol.9, No.3, pp.1-8, June 2008.
- [13] S. R. Na, K. Y. Hye, "User Location Anonymization Scheme Supporting Continuous Query Processing in Road Network," *Journal of KIISE*, Vol.17, No.1, pp.41-45, Jan. 2011.
- [14] J. Lim, Y. Park, D. Seo, J. Yoo, "An Efficient Continuous Reverse Skyline Query Processing Method in Metric Spaces for Location-based Services," *Journal of KIISE: Database*, Vol. 37. No.5, pp.250-257, Oct. 2010.
- [15] M. S. Kim, H. J. Yoo, J. Chae, W. Choi, "A Vehicles Location Inquiry Technique for Performance Improvement of LBS System," *Database*, Vol.26, No.3, pp.67-83, Dec. 2010.
- [16] J. H. O, J. S. Bae, D. W. Park, Y. H. Sohn, "Implementation of Location Based Service(LBS) using GPS for Various Sizes of Maps," Vol.8, No.4, pp.19-24, Apr. 2010.
- [17] K. S. Bok, M.S. Lee, Y. H. Park, J. S. Yoo, "An Effective Location Acquisition Method Based on RFID for Location Based Services," Vol.37, No.1, pp.33-44, Feb. 2010.
- [18] Y. Tao, D. Papadias, "Time Parameterized Queries in Spatio-Temporal Databases," *ACM SIGMOD*, 2002.
- [19] S. Ahn, B. Hong, C. Ban, K. Lee, "Design and Implementation of an Index Structure Using Fixed Intervals for Tracing of RFID Tags," *ICCSA2006, LNCS3981*, pp. 175-185, 2006.
- [20] Spatial Data Generator, DaVisual Code1.0. <http://isl.cs.unipi.gr>.



김종완(Jongwan Kim)

2007년 고려대학교 컴퓨터학과 이학박사
 2007년 건국대학교 정보통신대학 연구교수
 2010년 삼육대학교 연구교수
 2011년 ~ 2012년 University of New South Wales, Sydney Visiting Fellow
 2013년 ~ 현재 성결대학교 컴퓨터공학부 교수
 ※ 관심분야 : Mobile Distributed System, Spatial Data Indexing, Big Data Processing, Skyline Query, Mobile Game, CDN Security



조양현(Yang-Hyun Cho)

1982년 광운대학교 전자통신공학과(공학사)
 1985년 광운대학교 전자통신공학과(공학석사)
 2012년 광운대학교 전자통신공학과(공학박사)
 1987년 ~ 1997년 LG정보통신 전송기술개발실 과장
 1997년 ~ 현재 삼육대학교 컴퓨터학부 교수
 ※ 관심분야 : Computer Network, BcN, GMPLS