

---

# 태그 궤적 색인을 위한 인식공간 근접성 기법

김동현\* · 안성우\*\*

The Proximity Scheme of the Perceptual Space for Indexing The Trajectories of Tags

Dong Hyun Kim\* · Swng Woo Ahn\*\*

---

이 논문 또는 저서는 2006년 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (KRF-2006-331-D00460)

---

## 요 약

태그는 위치정보가 없기 때문에 리더의 식별자를 위치정보로 사용하지만 비연속적인 심볼형 정보이다. 따라서 태그 이동 궤적간의 근접성을 정의하는 것이 어려우며 사용자 질의 처리시 비효율적이다. 본 논문에서는 태그 위치를 표현하기 위하여 인식공간을 정의하고 인식공간 근접성을 제안한다. 인식공간 근접성은 정적 인식공간 근접성과 동적 인식공간 근접성으로 구성되며 이를 이용하여 태그 궤적간의 근접성을 측정하고 효율적인 색인을 구축할 수 있다. 제안한 태그 이동 궤적 근접성 함수를 평가하기 위해 태그 색인 기법인 IR-tree와 이동체 색인 기법인 R\*-tree를 대상으로 성능평가 실험을 수행하였다.

## ABSTRACT

Since tags do not have location informations, the identifiers of tags which are symbolic data are used as the location informations. Therefore, it is difficult to define the proximity between two trajectories of tags and inefficient to process the user queries for tags. In this paper, we define the perceptual space to model the location of a tag and propose the proximity of the perceptual spaces. The proximity of the perceptual spaces is composed of the static proximity and dynamic proximity. Using the proximity of the perceptual spaces, it is possible to measure the proximity between two trajectories of tags and build the efficient indexes for tag trajectories. We evaluated the performance of the proposed proximity function for tag trajectories on the IR-tree and the R\*-tree.

## 키워드

Proximity, Perceptual Spaces, Location Modeling, RFID, Tag

---

\* 동서대학교 컴퓨터정보공학부

\*\* 부산대학교 컴퓨터공학과

접수일자 : 2009. 08. 03

심사완료일자 : 2009. 08. 31

### I. 서론

유비쿼터스 IT 환경은 모든 사물에 센서와 네트워크 기능을 심는 것에서 출발하며 특히 센서 기반 기술 중에서 RFID 기술이 자동 데이터 수집 분야에서 가장 급속히 성장하고 있다. 그러나 다수의 RFID 태그가 리더를 통하여 이동할 때 다량의 RFID 데이터가 생성되며, 시간이 흐름에 따라 대용량의 시공간이력데이터로 데이터베이스에 누적된다. 따라서 태그에 대한 질의를 효율적으로 처리하기 위하여 태그 시공간이력데이터에 대한 색인 구조가 필수적으로 필요하다.

태그는 이동체와 달리 위치정보가 없기 때문에 리더의 식별자를 위치정보로 사용한다. 그러나 리더의 식별자는 비연속적인 심볼형 정보이기 때문에 태그 이동 궤적간의 거리나 속도를 측정하기 어렵다. 따라서 색인 구성 시 접근 가능성이 높은 궤적들을 인접한 노드에 저장하기 힘들기 때문에 태그나 리더에 대한 질의 수행 시 많은 수의 노드를 검색해서 질의 효율이 저하되는 문제점이 있다.

예를 들어 "t<sub>1</sub>~t<sub>6</sub>사이의 [빌딩A의 1층]에 있는 모든 tid를 검색하라"는 질의를 수행할 때 공간거리에 따라 색인을 구성하면 그림 1(a)처럼 2번의 탐색이 필요하나 포함관계에 따라 그림 1(b)와 같이 색인의 MBB를 구성하면 1번의 탐색으로 질의 처리가 가능하다.

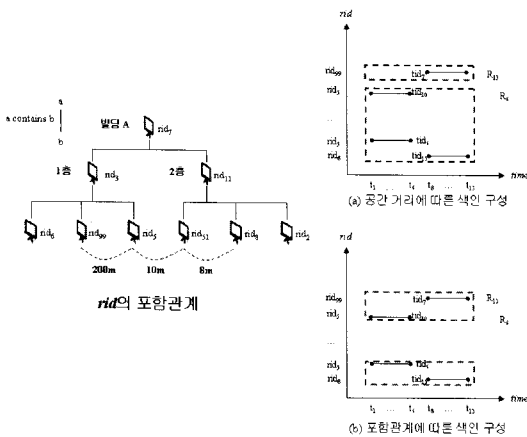


그림 1. 포함관계에 따른 색인 구성의 예.  
Fig 1. The Example of Index Construction by the Containment Relationship.

본 논문에서는 이러한 문제를 해결하기 위해 태그의 위치를 표현하기 위한 인식공간을 정의한다. 그리고 태그 이동 궤적들간의 근접성을 측정하기 위하여 인식공간에 기반한 인식공간 근접성 함수를 제안한다. 인식공간 근접성 함수는 포함속성, 간선속성 그리고 공간속성으로 표현된 정적 인식공간 근접성과 흐름속성으로 표현된 동적 인식공간 근접성으로 구성된다. 그리고 제안한 태그 이동 궤적 근접성 함수를 평가하기 위해 태그 색인 기법인 IR-tree와 이동체 색인 기법인 R\*-tree를 대상으로 성능평가 실험을 수행하였다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 기술하고 3장에서는 인식공간모델을 정의한다. 인식공간 근접성 함수를 4장에서 설명하고 5장에서 실험결과를 보인다. 그리고 6장에서 결론을 기술한다.

### II. 관련 연구

EPC Global에서는 태그 데이터를 위한 RFID 미들웨어 시스템의 표준안인 하나로 EPCIS(EPC Information Services)를 만들었다[1]. EPCIS는 다양한 비즈니스 응용들이 EPC Code를 이용하여 EPC와 연관된 데이터를 검색할 수 있도록 데이터 공유를 위한 표준 인터페이스를 응용 프로그램에게 제공한다. 이 중 EPCIS의 데이터 정의 레이어에서 위치 정보로 사용될 수 있는 네 가지 이벤트들을 정의하고 있다.

간격 데이터를 처리하기 위한 색인 구조는 다음 표 1과 같다.

표 1. 간격 데이터 색인.  
Table 1. Index for the Interval Data.

분 류		색인구조	
1 차원	해쉬 기반	메모리 베이스	. Containment Encoded Interval Index
		트리 기반	. Priority Search Tree . Interval Tree
	트리 기반	메모리 베이스	. External Priority Search Tree
		디스크 베이스	. External Interval Tree
다 차원	해쉬 기반	메모리 베이스	. Virtual Construct Rectangle Index . Cell-based Query Index-
		트리 기반	메모리 베이스
	트리 기반	메모리 베이스	. Segment R-tree
		디스크 베이스	. Segment R-tree

1차원 간격 데이터를 위한 색인인 Containment

**Encoded Interval** 색인[2]은 스트림 데이터에 대한 연속적인 간격 질의를 색인하는 기법으로 포함관계를 사용하여 간격을 분할한 뒤 해당 리스트에 삽입하는 방식을 취한다. **Priority Search Tree**[3]는 1차원의 간격을 2차원의 점으로 매핑 시키는 방법으로 아진트리 기반이다.

다차원 간격 데이터에 대한 색인 구조인 **Virtual Construct Rectangle** 색인[4]은 영역 질의를 색인하는 질의 색인 기법으로 영역 질의의 조건자를 미리 정의된 **VCR(Virtual Construct Rectangle)**이라는 사각형 구조의 가상 분할구조를 이용하여 분할하고 해당 구조와 연관되는 노드의 ID 리스트에 질의 ID를 삽입하는 하는 방식을 사용하며, **Segment R-tree**[5]는 **R-tree** 기반의 색인으로 다차원의 간격 데이터를 처리하기 위해 트리이다. 그러나 상기 기술한 색인 구조에서는 비연속적인 심볼 정보를 위한 근접성을 정의하지 않았기 때문에 태그 데이터를 효율적으로 색인하기 어렵다.

### III. 인식 공간 모델

태그는 자체적으로 좌표를 표시할 수 있는 기능이 없기 때문에 리더의 위치 정보를 이용하여 태그의 위치를 추상화시키는 것이 유일한 수단이다. 따라서 본 논문에서는 태그가 항상 전체 시스템의 비즈니스 흐름을 따라 이동한다고 가정한다. 그리고 태그의 위치를 기록하는 비즈니스 환경의 물리적인 공간을 논리적 공간으로 표현하기 위하여 인식 공간을 다음과 같이 정의한다.

- 정의 1: 인식 공간(**Perceptual Space, PS**)은 시스템 사용자에 의해서 하나의 단위로 인식되어지는 공간이다. 인식 공간은 계층 구조를 형성하며 계층 구조는 격자형(**lattice**) 구조를 가진다. 계층 구조상의 최하위 인식 공간을 단말 인식공간라 하고 그 외의 인식 공간을 비단말 인식공간이라 한다.
- 정의 2: 단말 인식공간(**Terminal PS, TPS**)은 인식공간 계층구조에서 최하위 인식공간이다.
- 정의 3: 비단말 인식공간(**Non-Terminal PS, NTPS**)은 인식공간 계층구조에서 단말 인식공간을 제외한 나머지 인식 공간이다.

리더의 기본 인식위치정보를 기반으로 태그의 이동 방향정보를 획득하는 것은 불가능하다. 따라서 시스템이 적용된 비즈니스 흐름과 연동하여 태그 이벤트 발생점인 판독점에서 태그의 입력과 태그의 출력을 표현한 태그이동그래프를 이용하여 태그의 이동방향을 표현한다.

- 태그이동그래프(**Tag Flow Graph, TFG**)는 기 정의된 비즈니스 흐름에 따라 기술된 태그 이동 그래프로 0개 이상의 입력과 출력 간선을 가지는 방향성 그래프이다.

인식공간을 이용하여 태그 이동 궤적을 표현하는 방법은 크게 3가지가 있다. 첫 번째는 인식 공간으로의 입력 이벤트가 발생할 때 마다 이벤트의 발생 시간을 이용하여 점으로 표현하는 방법이다. 두 번째는 한 이벤트가 발생한 시점과 다음 이벤트가 발생한 시점을 중심으로 두 발생 시점을 연결하여 선분으로 표현하는 방법이다. 그러나 두 가지 방법은 태그 이벤트 발생 시점 사이 또는 인식공간을 떠난 시간에 대한 질의를 수행하기 어려운 단점이 있다.

마지막으로 세 번째는 인식공간으로의 입력 이벤트가 발생한 시점과 인식공간으로부터의 출력 이벤트가 발생한 시점인 두 개의 시간 값을 이용하여 이동 궤적을 간격으로 표현하는 방법[6]이다. 이 방법은 발생 시점 사이에 대한 질의도 수행할 수 있고 한 궤적에서 출력 시간에 대한 질의도 수행할 수 있다 따라서 태그의 이동 궤적은 다음과 같이 간격형 데이터로 표현한다.

- 정의 4: 태그 궤적(**tr**)은 시간  $t_{enter} \sim t_{leave}$  시간동안에 인식 공간  $pid_i$ 에서 위치한 태그  $tid_k$ 의 태그이동궤적이다.

### IV. 인식공간 근접성

#### 4.1 정적 인식공간 근접성

정적 인식공간 근접성은 3가지 속성으로 구성된다. 첫 번째는 인식공간간의 계층 구조를 이용한 포함속성이다. 인식공간은 사용자가 인식하는 공간이기 때문에 상위 비단말 인식공간은 하위 단말 또는 비단말 인식공간들로 구성되어지며 이로 인해 인식공간간에는 포함관계가 형성된다.

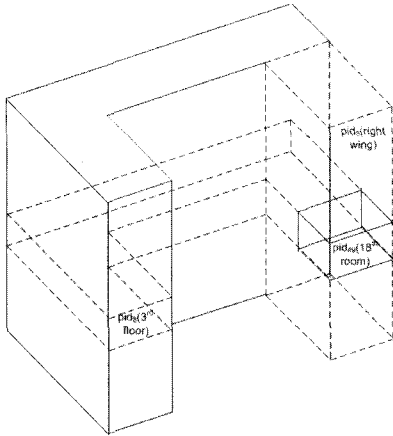


그림 2. 비즈니스 환경에서의 인식공간의 예.  
Fig 2. The Example of Perceptual Spaces in the Business Context.

예를 들어 그림 2에서 보듯이 18번방은 사용자의 주제 설정에 따라 오른쪽 날개 빌딩의 공간으로 인식될 수도 있고 또는 3층의 공간으로 인식될 수도 있다. 따라서 단말 인식공간 pid49는 비단말 인식공간인 pid5의 하위 인식공간인 동시에 pid8의 하위 인식공간이 된다.

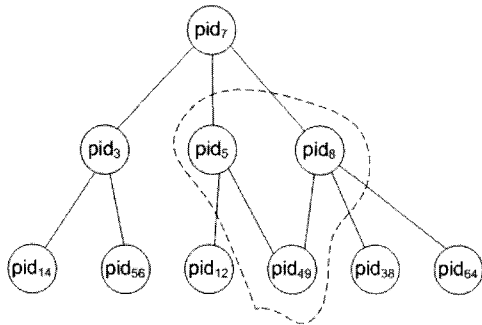


그림 3. 인식공간간의 포함관계를 위한 격자구조.  
Fig 3. The Lattice Structure for the Containment Relationships between Perceptual Spaces.

이러한 인식공간간의 포함 관계를 표현하기 위하여 하나의 하위 인식공간이 여러 상위 인식공간에 포함될 수 있는 격자 구조를 사용한다. 예를 들어 그림 2의 인식공간 pid49, pid8 그리고 pid5간의 관계를 그림 3의 격자구조로 표현된다.

사용자의 질의 때 상위 비단말 인식공간에 대한 질의는 해당 인식공간을 구성하는 하위 단말 인식공간에 대

한 질의로 변경되어 수행되기 때문에 같은 상위 인식공간을 가지는 단말 인식공간들은 다른 상위 인식공간을 가지는 단말 인식공간에 비하여 같이 접근될 가능성이 높다. 두 인식공간  $pid_i$ 와  $pid_j$ 간의 포함속성은 다음과 같이 정의한다.

- 정의 5: 포함속성  $C(pid_i, pid_j)$ 는  $pid_i$ 와  $pid_j$ 로부터 루트까지의 경로에서 같은 초상 인식공간을 만날 때까지의 경로 길이이다

두 번째 속성은 태그이동그래프의 간선 속성이다. 태그 이동 그래프의 간선은 인식공간에서 태그의 입출력 인식 이벤트가 발생하는 지점을 나타낸다. 질의의 조건자로 tid와 pid가 사용되기 때문에 두 인식공간간의 간선의 수가 많은 경우에 두 인식공간에서 발생한 태그 이동 궤적들을 같은 노드에 색인하면 색인의 성능을 향상시킬 수 있다. 간선 속성은 다음과 같이 정의한다.

- 정의 6: 간선 속성  $E(pid_i, pid_j) = \# \text{ of } RP^{<i,j>}$  where  $pid_i, pid_j, RP^{<i,j>} \in TFG$  and  $i \neq j$

마지막 세 번째 속성은 두 인식공간간의 공간속성이다. 인식공간간의 포함 관계를 이용하여 계층구조를 구성할 때 동일한 비단말 인식공간을 부모 노드로 하는 단말 인식공간간의 간선 속성치가 같은 경우에 인식공간간의 공간 속성을 사용한다.

인식공간을 조건자로한 질의 수행 시 공간 위치가 고려될 가능성이 높기 때문에 포함 속성치와 간선 속성치가 같은 비단말 노드간에 공간 거리가 같은 궤적끼리 같이 검색될 가능성이 높다. 두 인식공간간의 공간 속성은 다음과 같이 정의한다.

- 정의 7: 공간 속성  $D(pid_i, pid_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$  where  $x_i, y_i, x_j, y_j \in R^2$  and  $PS_i, PS_j$  and  $x_i, y_i$  is the center coordinates of  $PS_i$

#### 4.2 동적 인식공간 근접성

일반적으로 사용자의 관심도가 높은 인식공간에서 발생한 궤적끼리 같은 노드에 색인하면 색인의 성능이 향상될 수 있다. 그러나 사용자의 관심도는 정의하여 측정하기 어렵다. 공간 데이터의 경우에 공간 데이터의 밀

집도가 높은 영역일수록 해당 영역에 대하여 질의가 집중되기 때문에 이와 유사하게 태그 궤적의 수가 많은 인식영역에 대하여 사용자 질의가 많이 들어온다고 가정한다. 만약 태그 궤적이 많이 발생하는 인식공간들이 있다고 가정할 때 해당 인식공간들에서 발생한 궤적은 동시에 접근될 가능성이 높다.

예를 들어 그림 4에서 보듯이  $pid_i$ 와  $pid_j$ 간에 발생한 태그의 이동량이 1,000회이고  $pid_i$ 와  $pid_k$ 간에 발생한 태그의 이동량이 10회인 경우에  $pid_i$ 와  $pid_j$ 에서 발생한 태그 궤적들이  $pid_i$ 와  $pid_k$ 에서 발생한 태그 궤적들보다 같이 접근될 가능성이 높다.

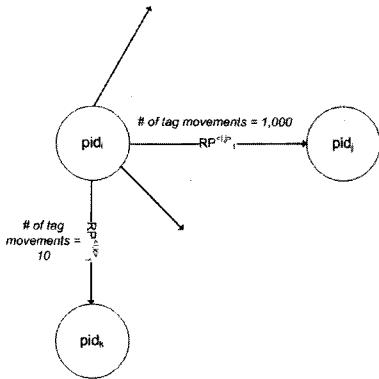


그림 4. 태그 이동량의 예.

Fig 4. The Example of the # of Tag Movements.

이러한 태그 이동량은 유동적이다. 즉, 태그의 이동량은 시간에 따라 변화한다. 따라서 태그의 이동량에 따른 근접성 수치는 시간의 값에 따라 달라지는 동적인 성질을 가지고 있다. 동적 속성으로 인하여 특정 시간대에 두 인식공간간의 근접성 수치가 높아서 같은 노드에 색인하였더라도 일정 시간이 흐른 후에는 오히려 다른 인식공간과의 근접성이 높아질 수 있다. 이를 위해 흐름속성을 다음과 같이 정의한다.

- 정의 8: 흐름 속성  $P(pid_i, pid_j) =$  where  $pid_i, pid_j, RP^{i,j} \in TFG$  and  $i \neq j$

4.3 인식공간 근접성 함수

태그 궤적을 시간(t), 태그 식별자(tid), 인식공간 식별자(pid) 인 3가지 도메인 상에서 표현하였기 때문에 근접성 값도 3가지 도메인을 이용하여 결정되어야 한다. 3

가지 도메인 중에서 태그 식별자인 tid는 동일 클래스 내에서 각각의 태그 식별자간에 연관성이 없기 때문에 태그 식별자간 근접성을 정의하기 어렵다.

따라서 인식 공간 도메인과 시간 도메인 상으로 태그 궤적을 투영시킨 상태에서 두 태그 궤적  $tr_1$ 과  $tr_2$ 간의 태그 근접성을 정의하여 측정한다. 태그 근접성을 정의하기에 먼저 인식 공간 식별자간의 근접성을 정적 인식공간 근접성과 동적 인식공간 근접성을 이용하여 다음과 같이 계산한다.

- 정의 9: 인식공간 근접성  $F(pid_i, pid_j) = w_0 \times (C(pid_i, pid_j) + E(pid_i, pid_j) + D(pid_i, pid_j)) + (1-w_0) \times P(pid_i, pid_j)$  where  $pid_i, pid_j, RP^{i,j} \in TFG$  and  $i \neq j$

두 태그 이동 궤적  $tr_1$ 과  $tr_2$ 간의 근접성 함수는 다음과 같이 정의하여 근접성을 측정한다.

- 정의 10: 태그 이동 궤적 근접성 함수  $Prox(tr_i, tr_j) = F(pid_i, pid_j) \times |\max(t_{leave_i}, t_{enter_j}) - \min(t_{enter_i}, t_{enter_j})|$  where  $pid_i, pid_j, t_{leave_i}, t_{enter_j}, t_{enter_i}, t_{enter_j}$  is the values of  $tr_i$  and  $tr_j$  respectively

V. 성능 실험

본 논문에서 제안한 태그 이동궤적 근접성 함수의 성능을 평가하기 위하여 R-tree 계열의 색인인 IR-Tree[6]와 R\*-Tree[7]에 대하여 인식공간 근접성 함수를 이용하여 색인한 경우와 임의의 생성된 인식공간 식별자의 순서만을 이용하여 색인한 경우를 비교하여 질의 처리시 발생하는 노드의 접근 횟수와 디스크 접근 횟수를 측정하였다.

실험을 위하여 색인 및 데이터 생성기는 Java 2 Platform Standard Edition 5.0을 이용하여 구현하였으며 Windows XP Professional 운영 체제를 기반으로 2GB 메인 메모리와 Pentium IV 2.6GHz CPU를 장착한 PC상에서 수행되었다.

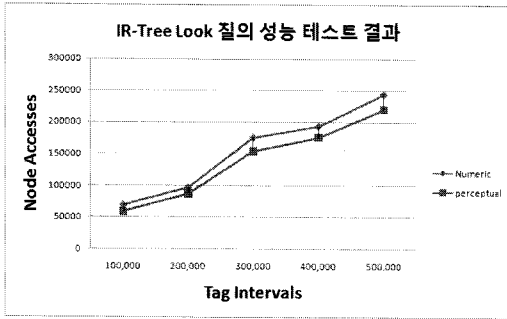


그림 5. IR-tree에서 노드 접근 수.  
Fig 5. # of node access in IR-Tree.

그림 5는 IR-tree에서 Look질의를 이용하여 질의 처리 시 노드의 접근 횟수 결과를 보여준다. 그림 5에서 보여 주듯이 제안한 인식공간 근접성 함수 기반 색인하였을 때 100K 데이터 집합보다 태그 쿼리의 수가 증가하는 500K 데이터 집합으로 갈수록 노드 접근 횟수의 차이가 늘어나며 임의로 배치하여 색인할 때보다 성능의 향상 비율이 늘어남을 알 수 있다.

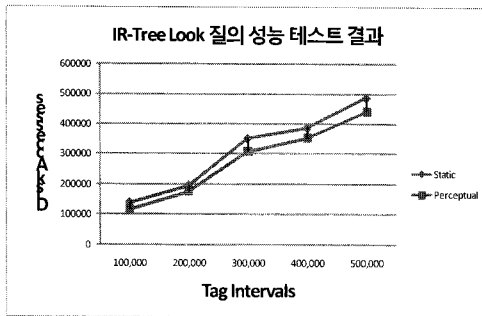


그림 6. IR-tree에서 디스크 접근 수.  
Fig 6. # of Disk Access in IR-Tree.

그림 6은 디스크 접근 횟수를 보여준다. 그림 6에서 보여주듯이 인식공간 근접성 함수에 따른 색인 기법이 접근 횟수가 적음을 알 수 있다. 특히 데이터 집합의 크기가 늘어날수록 두 기법간의 차이가 벌어지는 것을 보여주기 때문에 제안한 기법이 디스크 기반 색인 구조에서도 동일한 효과를 나타냄을 알 수 있다.

그림 7과 8은 R\*-Tree에서 동일한 실험을 수행한 결과이다. 그림 7과 8에서 보듯이 인식공간 근접성 함수를 기반으로 색인하였을 때 R\*-Tree에서 우수한 성능을 보이

는 것을 알 수 있다.

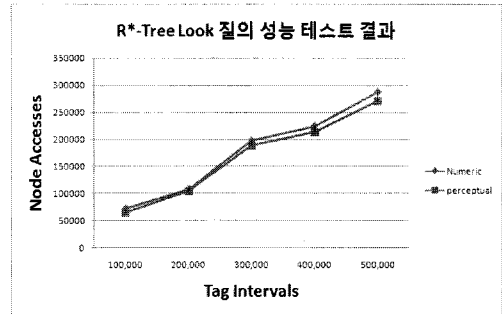


그림 7. R\*-tree에서 노드 접근 수.  
Fig 7. # of Node Access in R\*-Tree.

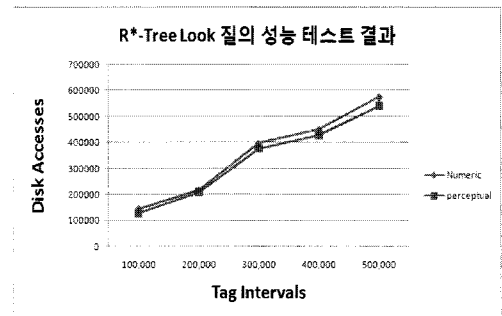


그림 8. R\*-tree에서 디스크 접근 수.  
Fig 8. # of Disk Access in R\*-Tree.

## VI. 결론

리더의 식별자는 비연속적인 심볼형 정보이기 때문에 색인 구성 시 접근 가능성이 높은 쿼리들을 인접한 노드에 저장하기 어렵다. 따라서 태그나 리더에 대한 질의 수행 시 많은 수의 노드를 검색해서 질의 효율이 저하되는 문제점이 있다.

본 논문에서는 태그 위치 데이터를 표현하기 위한 인식공간 모델링 기법을 제안하였다. 그리고 인식공간을 기반으로 한 인식공간 근접성을 정의하였다. 인식공간 근접성은 정적 인식공간 근접성과 동적 인식공간 근접성으로 이루어진다. 정적 속성은 포함 속성과 간선 속성으로 구성되며 동적 속성은 태그의 이동량인 흐름속성으로 구성된다. 그리고 이를 기반으로 한 태그 이동쿼리

근접성 함수를 제시하였다.

또한 IR-tree와 R\*-tree를 대상으로 제안한 이동계적 근접성 함수를 적용하여 태그 이동계적을 색인하고 성능평가 실험을 수행하였다. 실험 결과 Look 질의에 대하여 제안한 인식공간 근접성 함수를 이용한 색인 기법이 임의대로 인식공간을 배정하여 색인하는 기법보다 우수함을 보였다. 향후 연구로는 태그 식별자간의 근접성도 고려한 근접성 정의에 관한 연구가 필요하다.

### 참고문헌

- [1] EPCglobal, "EPC Information Services (EPCIS) Version 1.0 Specification," 2005
- [2] Kun-Lung Wu, Shyh-Kwei Chen, and S. Yu, "Query Indexing with cContainment-Encoded Intervals for Edfficient Stream Processing," Knowledge and Information System, Vol 9, Issue 1, pp.62-90, 2006.
- [3] E. M. McCreight, "Priority Search Trees," SIAM Journal of Computing, pp. 257-276, 1985.
- [4] Kun-Lung Wu, Shyh-Kwei Chen, Philip S. Yu, "VCR indexing for fast event matching for highly-overlapping range predicates.," SAC 2004, pp.740-747, 2004.
- [5] C. Kolovson and M. Stonebraker, "Segment Indexes: Dynamic Indexing Techniques for Multi-Dimensional Interval Data," Proc. ACM SIGMOD, pp. 138-147, 1991.
- [6] ChaeHoon Ban, BongHee Hong, DongHyun Kim, Time parameterized Interval R-tree for Tracing Tags in RFID systems, Proc of 16th International Conf on DEXA, 2005, pp.503-513
- [7] N. Beckmann and H. P. Kriegel, "The R\*-tree: An Efficient and Robust Access Method for Points and Rectangles," Proc. of the ACM SIGMOD Int'l Conf. on Management of Data, pp. 322-331, 1990.

### 저자소개



김동현(Dong Hyun Kim)

2003년 부산대학교 컴퓨터공학과 박사  
2004년 ~ 현재 동서대학교 컴퓨터정보공학부 조교수

※관심분야 : RFID, 센서 데이터베이스, 시공간 데이터베이스



안성우(Swng Woo Ahn)

2009년 부산대학교 컴퓨터공학과 박사  
2009년 ~ 현재 부산대학교 컴퓨터공학과 박사 후 과정

※관심분야 : RFID, 이동체 데이터베이스, 공간 데이터베이스