멀티미디어 데이타베이스에서 최근접 질의의 성능평가에 관한 분석 모델

(An Analytic Model for the Performance of Nearest Neighbor Queries in Multimedia Databases)

이 주 홍 † 차 광 호 †† 김 형 주 ††† 정 진 완 ††††

(Ju-Hong Lee) (Guang-Ho Cha) (Hyoung-Joo Kim) (Chin-Wan Chung)

요 약 다차원 색인 트리를 이용하는 최근접 질의는 멀티미디어 데이타베이스템에서 자주 사용되는 중요한 질의 형식으로서 그 성능 분석은 질의 성능 개선을 위해 중요하다. 지금까지 다차원 트리에서의 질의 성능 분석 모델에 관한 대부분의 연구는 R 트리와 같은 특정 트리에서의 범위 질의에 대한 분석만을 주로 다루고 있었으나 최근 다차원 색인 트리에서 최근접 질의에 대한 성능 모델[1]이 발표되었다. 그러나 이 모델은 1-최근접 질의만을 다루고 있다. 본 논문에서는 다차원 색인 트리에서 일반적인 k-최근접 질의의 성능 분석 모델을 제시한다. 이 모델은 다차원 색인 트리의 종류나 k-최근접 질의 처리 알고리즘의 종류에 관계 없이 적용되는 모델이다. 모델의 기본 개념으로서 지역 평균 볼륨과 가변 밀도 함수의 개념을 소개한다. 본 모델의 이점은 다음과 같다: 임의의 데이타 분포를 가진 데이타 집합에 대해서도 적용할 수 있고, 1-최근접 질의뿐 아니라 k-최근접 질의에서도 잘 적용되며, 색인 트리에 저장된 데이타로 곧바로 분석하므로 시간이 많이 소요되는 시뮬레이션 없이 빠르게 분석할 수 있다. 본 모델의 정확성을 평가하기 위해서 여러 가지 분포의 데이타 집합에 관하여 실험하였다. 실험 결과는 저차원 또는 중차원 데이타 집합에 대하여 데이타의 분포에 관계없이 정확한 결과를 보여주고 있다.

Abstract The k-nearest neighbor query in multidimensional index tree is one of the most frequently used query types in multimedia databases. It is important to analyze the performance of the *k-nearest neighbor query* for its performance improvement. Until now, most of the analytic models are restricted to a particular type of the index tree, for example, the R-Tree and they concentrate on the analysis of the range query. Recently, a cost model [1] was reported for nearest neighbor queries. However, the model considered only 1-nearest neighbor queries rather than k-nearest neighbor queries. In this paper, we present an analytic model for the performance of the k-nearest neighbor query in multidimensional index trees. This model is independent of kinds of multi-dimensional index trees and k-nearest neighbor algorithms. As a basis of the model, we introduce the concept of the regional average volume and the varying density function. The advantages of our model are in particular as follows: It is applicable to any type of datasets with arbitrary distributions (uniform and non-uniform ones), works for the k- as well as 1-nearest neighbor query, and is a dynamic analysis method which enables a rapid analysis without requiring a time-consuming simulation of data. To estimate the accuracy of our model, we conducted a various range of experiments on the datasets with various distributions. The results show that our analytic model is accurate for the data sets with non-uniform distributions as well as uniform distributions in low and mid dimensions.

1. 서 론

멀티미디어 데이타베이스나 지리 정보 시스템에서는 데이타를 주고 그것과 유사하거나 거리상으로 가까운 데이타를 찾는 질의를 많이 사용한다. 이러한 질의를 유사성 질의 또는 최근접 질의라고 하는데, 결과로서 가장 가까운 k개의 데이타를 찾아내는 질의를 k-최근접 질의라고 한다. 다차원 공간 데이타를 사용하는 대다수의 응용에서는 k개의 가장 근접한 데이타를 찾는 k-최근접 질의가 필요하다. 예를 들면, 의료정보 시스템에서는 환자의 MRI 이미지를 기존의 데이타 베이스에 저장된

· 이 연구는 과학기술부의 핵심 S/W기술개발 사업에서 지원받았음.

† 종신회원 : 한국과학기술원 정보및통신공학과
jhlee@islab.kaist.ac.kr
†† 종신회원 : 동명정보대학교 멀티미디어공학과 교수
ghcha@tmic.tit.ac.kr
††† 종신회원 : 서울대학교 컴퓨터공학과 교수
hjk@oopsla.snu.ac.kr
†††† 종신회원 : 한국과학기술원 전산학과 교수
chungcw@islab.kaist.ac.kr

논문접수 : 1998년 7월 2일
심사완료 : 1999년 3월 10일

이미지와 비교하여 유사한 것을 찾아내고 함께 저장된 병력과 치료 정보를 토대로 환자의 진단과 치료에 활용할 수 있다. 비디오 데이타 베이스에서는 사용자가 원하는 비디오를 찾기 위하여 질의 이미지를 주고 이와 유사한 대표 프레임 이미지를 갖는 샷(shot)을 찾을 수 있다.

이러한 최근접 질의를 효율적으로 처리하기 위해서는 근접한 데이타들을 찾을 수 있게 설계된 다차원 색인 트리가 필수적이다. 최근에 R 트리와 같이 노드 경계가 사각형인 형태를 갖는 다차원 색인 트리에서의 최근접 질의를 효율적으로 처리할 수 있는 알고리즘이 개발되었다[2, 3] 이들 알고리즘은 디스크 검색 회수로서 성능이 측정되고 색인 트리의 특성에 따라서 검색 성능이 영향을 받는다. 최근접 질의의 성능을 분석하고 비교하기 위해서는 트리에 독립적인 분석 모델이 필요한데 대부분의 분석 모델들은 [4,5,6,7] R 트리와 같은 특정한 트리에 한정되어 있으며 질의 영역이 고정된 값으로 주어지는 범위 질의에 한정되어 있다. 최근에 경계가 사각형인 트리의 최근접 질의 검색에 필요한 디스크의 검색 회수를 분석하는 모델이 발표되었는데[1] 이 모델은 색인 트리의 종류와 무관하게 성능을 비교적 잘 예측하였고 고차원에서도 잘 적용되었지만 $k-$최근접 질의에 대한 분석이 아니라 $1-$최근접 질의에 대한 분석으로서 일반화된 분석 기법이라기 보다는 특수한 경우에만 한정한 분석이라고 할 수 있고 비균일한 데이타 분포를 가진 데이타 집합에 대하여는 검증되지 않았다. 또한 확률과 그 미분 값를 계산하기 위해서 데이타 베이스에 현재 들어 있는 데이타를 확률 모형을 이용해 샘플링하여 분석하므로 매우 시간이 많이 걸리는 방법이다.

이 논문에서는 노드의 경계가 사각형으로 이루어지는 다차원 색인 트리에서 $k-$최근접 질의의 성능 분석을 위한 모델을 제시한다. $k-$최근접 질의의 분석은 범위 질의의 경우와 비교해서 분석하기가 더 어려운 측면이 있다. 즉 질의 성능을 구하기 위해서 한 노드가 액세스되는 확률을 계산할 때, 범위 질의는 고정된 질의 영역이 주어지므로 그 확률을 곧바로 구할 수 있는데 반해서 $k-$최근접 질의는 k 값만 주어지므로 질의 위치에 따라서 질의 영역의 크기가 달라지므로 확률을 계산하기가 좀더 복잡하다. 따라서, 본 논문에서는 $k-$최근접 질의의 질의 영역을 어떻게 결정하는가에 초점을 맞추고 질의 성능을 분석한다.

본 논문에서는 지역 평균 볼륨과 밀도 함수에 대한 개념을 소개한다. 지역 평균 볼륨은 $k-$최근접 질의의 질의 구의 평균 반경을 계산할 때 사용되는데 $k-$최근접 질의 구의 평균 반경은 균일한 분포를 갖는 데이타 집합에 대한 성능 분석에 적합하다. 그러나 지역 평균 볼륨을 이용하여 구한 질의 구의 평균 반경으로 계산한 성능은 비균일 분포를 갖는 데이타 집합에 대해서는 잘 적용되지 않는다. 이 것은 비균일의 정도가 클수록 평균 반경과 실제 반경 간의 차이가 커지기 때문이다. 따라서 비균일 분포를 갖는 데이타 집합에서 질의 구의 반경에 대한 정확한 측정 방법이 필요하다. 이를 위해서 데이타 밀도 함수를 사용하는데 데이타 밀도에 대한 근사값으로 노드 밀도를 사용하여 질의 구의 반경을 계산하고 디스크 검색 회수를 계산한다.

본 논문에서 제시한 다차원 색인 트리의 성능 모델은 비균일 분포를 갖는 데이타 집합에 대해서도 잘 적용되고 2~8차원에서 정확한 결과를 보여 주고 있지만 그 이상으로 차원이 높아지면 오차가 커진다. 그러나 대표적인 다차원 트리인 R^*-트리[8]는 5차원 이상에서는 효율성이 크게 저하되며, $X-$트리[9]도 R^*-트리보다는 다차원에서 매우 효율성이 높으나 8차원 이상에서는 성능이 저하되는데, 이런 현상은 $R-$트리[10]계열의 다차원 색인 트리에서는 일반적인 현상이고 실제 응용에서도 중저차원 응용이 많이 있으므로[11, 12, 13], 본 논문의 결과가 실용적으로 가치가 있다고 볼 수 있다.

본 논문은 다음과 같이 구성되어 있다: 2절에서는 다차원 색인 트리에 기반한 $k-$최근접 질의의 해석적인 분석 기법을 보인다. 3절에서는 다양한 분포에서 수행한 실험 결과를 보이고 실험 결과에 대하여 해석하고 4절에서 결론을 맺는다.

2. $k-$최근접 질의 성능의 분석

이 절에서는 $k-$최근접 질의의 성능을 분석하기 위한 모델을 제시한다.

먼저, $k-$최근접 질의의 성능을 분석 하는 문제를 형식적으로 정의하자. n을 데이타 공간의 차원이라고 하자. 데이타 공간이 길이 1로써 정규화되어 있다고 가정한다. 즉 $W_i=[0,1), 1\le i\le n$ 일때, $W=W_1\times W_2\times\ldots\times W_n$ 는 모든 데이타 객체가 정의되는 n 차원 단위 데이타 공간이다. 색인 트리에는 N 개의 데이타 객체를 저장하는 m 개의 노드 $s_1, s_2, \ldots, s_m$ 가 있다고 가정하고, 각각의 노드는 한 개의 디스크 페이지에 대응한다고 가정한다. 노드 s_i의 디렉토리 영역 $DR(s_i)$ 는 s_i 안에 들어가는 모든 객체들을 포함하는 n 차원 최소 경계 상자이다. 즉, $DR(s_i)$는 $[l_1, r_1]\times\ldots\times[l_n, r_n], l_i, r_i\in$

$W_i, l_i \le r_i, 1 \le i \le n$ 로 표현된다. D 를 n-차원 데이타 공간에서 정의된 거리 함수라 하자. 어떤 한 점 $(x_1, x_2, \ldots, x_n)$과 양의 정수 k 가 주어졌다고 하면 k-최근접 질의라는 것은 점 $(x_1, x_2, \ldots, x_n)$에서 부터 거리 함수 D 로 계산했을 때 k 개의 가장 최근접한 데이타 객체를 찾는 질의이다. 대개는 두 점간의 거리를 계산하는 거리 함수로서 L_p 메트릭 함수를 사용한다. X와 Y를 각각 n차원에서의 두 점 $X=(x_1, x_2, \ldots, x_n)$, $Y=(y_1, y_2, y_n)$ 라고 하면, X 와 Y 사이의 거리, $D(X, Y)$ 는 다음과 같이 정의된다.

$$D(X, Y) = \left[\sum_{i=1}^{n} |x_i - y_i|^p \right]^{1/p}$$

여기서 p 는 Minkowski 메트릭 차원이다. $p=1$인 경우에는 맨해튼 거리 함수가 되며, $p=2$에 대해서는 유클리디언 거리 함수가 된다. 유클리디언 거리 함수가 일반적으로 사용되는 함수이기 때문에, 본 논문에서는 유클리디언 거리 함수를 사용한다.

본 논문에서의 성능 모델의 목적은 k-최근접 질의의 평균 성능을 예측하기 위한 정확한 해석적인 식을 구하는 것이다. 질의 처리의 성능을 측정하는 단위로써 평균 디스크 검색 회수를 사용하였는데 이는 평균 질의 성능에 디스크 검색 회수가 가장 큰 요인이기 때문이다.

2.1 k-최근접 질의의 성능 모델

본 논문에서는 k-최근접 질의에 필요한 디스크 검색 회수를 결정하는 성능 모델을 제시한다. 디스크 검색 회수는 질의 중심에서 k 개의 가장 가까운 데이타 객체를 둘러 싸는 하이퍼 구(球)와 교차하는 색인 노드의 개수를 계산하여 구한다. 그러면 k-최근접 질의를 수행하는 데 필요한 평균 디스크 검색 회수 DA 는 다음과 같이 주어진다.

$$\sum_{i=1}^{m} probability(R_k \cap DR(s_i) \neq \emptyset) \qquad (1)$$

여기서 R_k는 평균 크기를 가지는 k-최근접 질의의 영역이고, $DR(s_i)$는 노드 s_i의 디렉토리 영역이다. m 은 색인 트리에 있는 노드의 개수이다. 디렉토리 영역 $DR(s_i)$의 크기는 색인 트리로부터 직접 구할 수 있고, R_k의 평균 크기는 데이타 집합의 분포 특성으로부터 구할 수 있다. 표 1은 이 논문에서 사용되는 기호와 정의들을 요약한 표이다.

범위 질의는 응용 분야에서 질의 영역이 미리 주어지

표 1 기호와 정의

기 호	정 의
n	차원
N	색인 트리에 저장된 데이타의 개수
m	색인 트리의 노드 개수
s_i	색인 트리의 i번째 노드
$DR(s_i)$	노드 s_i의 디렉토리 영역
k	찾아야 될 가장 근접한 데이타 객체의 개수
R_k	모든 질의점에 대한 k-최근접 질의 영역의 평균 크기에 해당 하는 영역
$Vol(R_k)$	모든 질의점에서의 k-최근접 질의의 평균 볼륨
$R_k(X)$	중심점이 X인 k-최근접 질의 영역
$Vol(R_k(X))$	$R_k(X)$의 하이퍼 볼륨
r	n-차원 하이퍼 구의 반경
$S_n(r)$	반경 r인 n차원 하이퍼 구의 하이퍼 볼륨
DA	k-최근접 질의의 디스크 검색 회수

지만 k-최근접 질의에서는 k 값만 주어지므로 질의 영역의 하이퍼 볼륨 $Vol(R_k)$을 계산할 수 있어야 한다. 그런데 $Vol(R_k)$는 데이타 분포와 질의 위치에 따라서 크기가 다르다. 그러므로 k-최근접 질의의 성능 모델을 개발하는 첫 번째 단계는 $Vol(R_k)$또는 $Vol(R_k(X))$을 구하고 $Vol(R_k) = S_n(r)$ 또는 $Vol(R_k(X)) = S_n(r)$로부터 평균 반경 또는 X위치에 가변인 반경을 구한다. 마지막으로 식(1)에서 계산된 반경을 적용하여 디스크 검색 회수를 계산한다.

2.2 k-최근접 질의의 하이퍼 볼륨과 반경의 계산

다차원 색인 트리에 N 개의 균일하게 분포된 데이타 객체가 저장되어 있다고 가정하자. 데이타 공간을 n 차원의 단위 공간 $[0,1)^n$ 이라고 할 때, k-최근접 질의 영역의 하이퍼 볼륨 $Vol(R_k)$ 는, 전체 N 개의 데이타 객체 중에서 k 개의 데이타 객체를 포함하므로 근사적으로 k/N 이 된다. 그러나 데이타 객체의 분포가 균일하지 않은 경우에는 질의 위치 X에서의 데이타 분포에 따라서 하이퍼 볼륨 $Vol(R_k(X))$ 이 다르다.

그림 1 은 각각 +와 × 위치에 있는 두 개의 11-최근접 질의 Q_1과 Q_2를 보여 주고 있다. 질의 점 Q_1은 밀도가 적은 영역에 있고 질의 Q_2는 밀도가 높은 영역에 있다. 이 두 영역의 크기는 크게 다르다. 따라서 평균 볼륨이 k/N과 상당히 다를 수 있으므로 평균 볼륨 $Vol(R_k)$의 정확한 값을 구하기 위해서 지역 평균 밀도

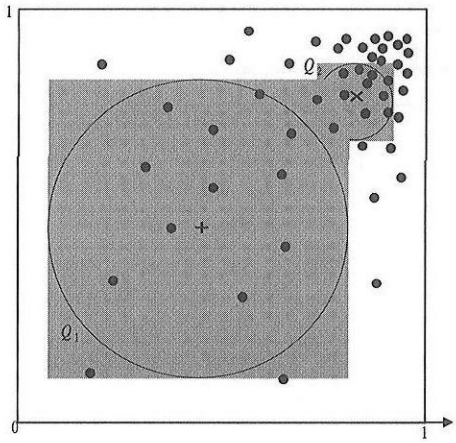


그림 1 각각 + 와 ×위치에 있는 두개의 11-최근접 질의

의 개념을 다음과 같이 소개한다.

정의1: (지역 평균 볼륨)

데이타 공간 DS 가 m 개의 영역 A_i , $i=1,..,m$ 으로 분할되어 있다고 하자. 영역 A_i안의 데이타 객체 당 평균 볼륨 ω_i는 $\omega_i = \frac{volume\ of\ A_i}{number\ of\ data\ objects\ in\ A_i}$ 이다. 그러면 m 개의 영역 전체의 지역 평균 Ω_m은 다음과 같이 정의된다.

$$\Omega_m = \sum_{i=1}^{m} \omega_i \frac{volume\ of\ A_i}{volume\ of\ DS} \tag{2}$$

데이타 객체 당 지역 평균 볼륨 Ω는 다음과 같이 정의된다.

$$\Omega = \Omega_\infty = \lim_{m\to\infty} \sum_{i=1}^{m} \omega_i \frac{volume\ of\ A_i}{volume\ of\ DS} \tag{3}$$

정의 2: (밀도 함수)

임의의 점 X 에서의 밀도함수 $D_{\triangle x}(X)$ 를 다음과 같이 정의한다.

$$D_{\triangle x}(X) = \frac{N(A,(X,\triangle X))}{Vol(A(X,\triangle X))} \tag{4}$$

여기서 $X=(x_1, x_2, ..., x_n)$, $\triangle X=(\triangle x_1, \triangle x_2, ..., \triangle x_n)$이고 $A(X,\triangle X)$는 i번째 차원의 범위가 $\left(x_i - \frac{\triangle x_i}{2}, x_i + \frac{\triangle x_i}{2}\right)$인 하이퍼 사각형이다.

$Vol(A(X,\triangle X))$는 $A(X,\triangle X)$의 하이퍼 볼륨이다. 그리고 $N(A(X,\triangle X))$는 $A(X,\triangle X)$ 영역 안에 있는 데이타 객체들의 개수이다. 데이타 공간에 있는 데이타 객체의 총 개수 N이 매우 크고 충분히 작은 영역 $\triangle X$에서 밀도 분포가 거의 균일하다고 가정하면, 불연속 함수로서의 밀도 함수의 정의 $D_{\triangle x}(X)$를 연속 함수의 미분으로 다음과 같이 표현할 수 있다.

$$D(X) = \lim_{\triangle X\to 0} D_{\triangle x}(X) = \lim_{\triangle X\to 0} \frac{N(A(X,\triangle X))}{Vol(A(X,\triangle X))} \tag{5}$$

전체 데이타 객체의 수 N은 최근접 질의 개수 k에 비해 매우 크다고 하자. 즉, $k \ll N$ 이다. 사용자는 보통 데이타베이스에 있는 많은 데이타 중에서 극히 일부분만을 검색하기 때문에 이러한 가정은 합당한 가정이라고 할 수 있다. 그러면 다음의 정리가 성립한다.

정리 1: 데이타 공간이 m개의 영역으로 분할되어 있다고 하면 R_k 의 평균 볼륨 $Vol(R_k)$은 근사적으로 $k\Omega_m$으로 주어진다. 즉, $Vol(R_k) \approx k\Omega_m$

(증명) $k \ll N$ 이므로 전체 공간에서 $R_k(X)$는 매우 작은 영역이다. 따라서, 영역 $R_k(X)$에서 데이타의 분포는 거의 균일하다고 간주할 수 있다. $R_k(X)$는 k개의 데이타를 가지고 있으므로, 볼륨과 밀도를 곱하면 k가 된다. 즉, $Vol(R_k(X)) \cdot D(X) = k$ 이다. 이렇게 하여 $R_k(X)$의 하이퍼 볼륨 $Vol(R_k(X))$을 다음과 같이 계산할 수 있다.

$$Vol(R_k(X)) = \frac{k}{D(X)} \tag{6}$$

$Vol(R_k) = E[\,Vol(R_k(X))\,]$ 이므로, 다음 공식을 얻을 수 있다:

$$Vol(R_k) = \int_{DS} \frac{k}{D(X)}\, dX \tag{7}$$

여기서 DS 는 데이타 공간, $[0,1)^n$이다.

데이타 공간 DS 가 m개의 영역 $A_i, i=1,...,m,$ 으로 분할되어 있다고 하면

$$Vol(R_k) = \int_{DS} \frac{k}{D(X)}\, dX =$$

$$\lim_{m\to\infty} \sum_{i=1}^{m} \frac{k}{D_{Ai}(X_i)} \cdot \frac{volume\ of\ A_i}{\sum_{i=1}^{m} volume\ of\ A_i} =$$

$$k \lim_{m\to\infty} \sum_{i=1}^{m} \frac{volume\ of\ A_i}{number\ of\ data\ objects\ in\ A_i} \cdot \frac{volume\ of\ A_i}{volume\ of\ DS}$$

$$= k \lim_{m\to\infty} \sum_{i=1}^{m} \omega_i \cdot \frac{volume\ of\ A_i}{volume\ of\ DS}$$

$= k\Omega \approx k\Omega_m$ 여기서 X_i 는 A_i의 중심점이다.

(8)

이제 n차원 하이퍼 구(球)의 하이퍼 볼륨 $S_n(r)$을 먼저 구한다. 2차원인 경우에는 $S_2(r)=\pi r^2$이다. $S_2(r)=\pi r^2=Vol(R_k)$이므로 반경 r은 다음과 같다.

$$r=\sqrt{\frac{Vol(R_k)}{\pi}}=\sqrt{\frac{k\Omega}{\pi}}\approx\sqrt{\frac{k\,\Omega_m}{\pi}} \tag{9}$$

다음 예는 식(8)과 식(9)의 적용 사례를 보여주는 예이다.

(예 1) 2-차원의 데이타 공간에서 생각한다. 즉 $X=(x,y)$이다. 밀도함수 $D(X)$를 $D(X)=\frac{4}{9}N(x+1)(y+1)$라고 할 때, 밀도 함수를 데이타 공간 전체로 적분하면 N이 된다.

즉, $\int_{ds}D(X)dX=\int_0^1\int_0^1\frac{4}{9}N(x+1)(y+1)dxdy=N$ 이다. 정리 1에 의해서, $Vol(R_x)=\int_{ds}\frac{k}{D(X)}dX=k\Omega$ 이므로 $\Omega=\int_{ds}\frac{1}{D(X)}dX=\int_0^1\int_0^1\frac{9}{4N(x+1)(y+1)}dxdy=\frac{9(\ln 2)^2}{4N}\approx\frac{1.05}{N}$ 이다. 따라서 $Vol(R_k)=k\Omega\approx\frac{1.05k}{N}$ 이다. $Vol(R_k)=\pi r^2$이고, $\frac{1.05k}{N}\approx\pi r^2$이기 때문에 R_k의 반경 r은 $r\approx\sqrt{\frac{1.05k}{\pi N}}$ 이다.

이 문제를 n차원으로 일반화 하려면 $S_n(r)$을 구해야 하는데, $S_n(r)$는 다음과 같이 재귀적으로 구할 수 있다:

$$S_n(r)=\int_r^{-r}S_{n-1}(f(x))dx,\quad n\geq 2 \tag{10}$$

여기서 $S_1(r)=2r$, $f(x)=\sqrt{r^2-x^2}$ 이다.

또는 $S_n(r)$을 단일 식으로 표현하면 다음과 같다 [1]:

$$S_n(r)=\frac{\sqrt{\pi}^n}{\Gamma(n/2+1)}r^n \tag{11}$$

여기서 $\Gamma(1/2)=\sqrt{\pi}$, $\Gamma(1)=1$, $\Gamma(x+1)=x\cdot\Gamma(x)$이다. $n=0,\ldots,3$에 대해서, $S_0(r)=1$, $S_1(r)=2r$, $S_2(r)=\pi r^2$, $S_3(r)=\frac{4}{3}\pi r^3$이다. 식(8)와 식(11)을 같게 놓으면 $Vol(R_k)=S_n(r)=k\Omega$이고, R_k의 반경 r에 관한 식은 다음과 같이 얻을 수 있다.

$$r=\sqrt[n]{k\Omega\cdot\frac{\Gamma(n/2+1)}{\sqrt{\pi}^n}}\approx\sqrt[n]{k\Omega_m\cdot\frac{\Gamma(n/2+1)}{\sqrt{\pi}^n}} \tag{12}$$

2.3 평균 디스크 검색 횟수

평균 디스크 검색 회수를 구하는 방법은 질의가 노드와 교차할 확률로 구해진다[6]. 따라서 반경 r인 n차원 하이퍼 구가 노드 S_i 의 디렉토리 영역 $DR(s_i)$와 교차하는 확률을 구하여 평균 디스크 검색 횟수를 계산한다. 먼저 2차원의 간단한 경우를 고려하고, n차원으로 일반화 시킨다. 예를 들어서 그림 2를 보자. $W=[0,1)^2$은 2차원의 단위 데이타 공간이다. 디렉토리 영역 $DR(s_i)$은 각 차원의 길이가 d_i인 사각형이다. 반경 r인 원 C 는 k-최근접 질의의 검색 영역 R_k이다. 디렉토리 영역 $DR(s_i)$가 질의 원 C와 교차할 확률은 디렉토리 노드가 검색될 확률과 같다. 따라서 다음 식을 얻는다.

$$Probabability(C\cap DR(s_i)\neq\emptyset)= area(DR(s_i))+perimeter(DR(s_i))\cdot r+area(C) \tag{13}$$

색인 트리에 있는 m개의 노드에 대해 전부 합하면, 식(1)을 이용하여 2차원 공간에서 평균 디스크 검색 횟수 $DA(2)$를 계산하면 다음과 같다:

$$\begin{aligned}DA(2)&=\sum_{i=1}^{m}Probability(C\cap DR(s_i)\neq\emptyset)\\&=\sum_{i=1}^{m}area(DR(s_i))+r\cdot\sum_{i=1}^{m}perimeter(DR(s_i))+\sum_{i=1}^{m}area(C)\\&=\sum_{i=1}^{m}d_i^2+2\cdot 2r\sum_{i=1}^{m}d_i+m\pi r^2\\&=\sum_{i=1}^{m}d_i^2\cdot S_0(r)+2\cdot\sum_{i=1}^{m}d_i^1\cdot S_1(r)+\sum_{i=1}^{m}d_i^0\cdot S_2(r)\end{aligned}$$

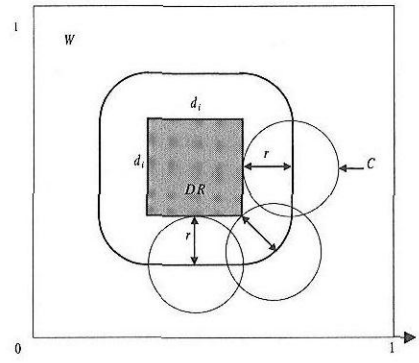


그림 2 디렉토리 영역 DR 과 질의 원 C가 교차할 확률 P

위의 식을 n-차원으로 일반화 하면, DA에 관하여 다음식을 얻는다:

$$DA(n) = \sum_{i=1}^{m_1} \sum_{j=0}^{n} \sum_{\{x_{i_1},\ldots,x_{i_j}\} \in \mathcal{S}(\{x_1,\ldots,x_n\})} hypervolume_{\{x_{i_1},\ldots,x_{i_j}\}}(DR(s_i)) \cdot S_{n-j}(r) \quad (14)$$

여기서 x_k 는 $DR(s_i)$의 k-차원의 길이이고, $\mathcal{S}(A)$ 는 집합 A의 멱 집합(power set)이고, $hypervolume$ 는 다음과 같이 정의된다:

$$hypervolume_{\{x_{i_1},\ldots,x_{i_j}\}}(DR(s_i)) = \prod_{k \in \{i_1,\ldots,i_j\}} x_k$$

여기서 $j=0$이면, $\{x_{i_1},\ldots,x_{i_j}\} = \emptyset$ 이다. 식(11)과 식(12)로부터 다음 식을 얻는다.

$$S_j(r) = \frac{\sqrt{\pi}^j}{\Gamma(j/2+1)} r^j = \frac{\sqrt{\pi}^j}{\Gamma(j/2+1)} \left(k\Omega \frac{\Gamma(n/2+1)}{\sqrt{\pi}^n} \right)^{\frac{j}{n}}$$
$$= \frac{1}{\Gamma(j/2+1)} (k\Omega\Gamma(n/2+1))^{\frac{j}{n}} \quad (15)$$

식(14)는 n-차원 데이타 공간에서 k-최근접 질의를 수행할 때 필요한 디스크 검색횟수를 계산하는 식이다. 그러나 데이타 분포가 균일하지 않은 경우에는 k-최근접 질의의 반경 r이 질의 부근의 밀도에 따라서 변화한다. 만약 노드의 크기가 충분히 작아서 노드 주의의 밀도가 노드 안의 밀도와 거의 같다고 가정하고 노드의 겹침 현상에 의한 효과가 매우 작다고 가정한다면 색인 노드 주의의 밀도 함수를 다음과 같이 정의하여 비균일 분포의 경우에 정확한 계산식을 얻을 수 있다:

$$D(s_i) = \frac{\text{노드 } s_i\text{안의 데이타 개수}}{\text{노드 } s_i\text{의 볼륨}} \quad (16)$$

만약 데이타베이스에 있는 전체 데이타 개수와 비교해서 색인 노드에 있는 데이타의 개수가 매우 적다고 가정하면, 색인 노드의 볼륨도 매우 작다. 따라서, 색인 노드와 교차하는 k-최근접 질의 영역의 밀도는 색인 노드의 밀도와 근사적으로 같다고 할 수 있다. 그러면 색인 노드 s_i 근처의 질의 $R_k(X)$의 볼륨은 다음과 같다:

$$Vol(R_k(X)) \approx \frac{k}{D(s_i)} \quad (17)$$

색인 노드 s_i 근처의 $R_k(X)$의 반경이 r_i이면 식(11)과 식(17)로부터, r_i는 다음과 같다:

$$r_i \approx \sqrt[n]{\frac{k}{D(s_i)} \cdot \frac{\Gamma(n/2+1)}{\sqrt{\pi}^n}} \quad (18)$$

식(14)과 식(15)에 있는 $S_j(r)$ 는 다음과 같이 $S_j(r_i)$로 교체할 수 있다:

$$DA(n) \approx \sum_{i=1}^{m_1} \sum_{j=0}^{n} \sum_{\{x_{i_1},\ldots,x_{i_j}\} \in \mathcal{S}(\{x_1,\ldots,x_n\})} hypervolume_{\{x_{i_1},\ldots,x_{i_j}\}}(DR(s_i)) \cdot S_{n-j}(r_i) \quad (19)$$

$$S_j(r_i) = \frac{\sqrt{\pi}^j}{\Gamma(j/2+1)} r_i^j$$
$$= \frac{\sqrt{\pi}^j}{\Gamma(j/2+1)} \left(\frac{k}{D(s_i)} \cdot \frac{\Gamma(n/2+1)}{\sqrt{\pi}^n} \right)^{\frac{j}{n}}$$
$$= \frac{1}{\Gamma(j/2+1)} \left(\frac{k\Gamma(n/2+1)}{D(s_i)} \right)^{\frac{j}{n}} \quad (20)$$

식(19)로부터 정확한 성능 값을 얻으려면 경계효과를 고려하여야 한다. 그림 3에서 설명되듯이 경계효과는 노드가 데이타 공간의 경계선 부근에 있을 때 발생한다. 그림 3에서 데이타 공간의 바깥으로 나온 영역 V_0은 노드를 검색할 확률과 무관한 부분이다. 그러나 식(19)에는 V_0 가 포함되어 있으므로 V_0 를 빼야 한다. 고차원 데이타 공간인 경우에는 경계효과가 자주 발생하므로 성능 계산에 큰 영향을 미치게 된다. 그러나 V_0의 볼륨을 해석적인 방법으로 계산하는 것은 매우 어렵다. 따라서, 확률적으로 볼륨을 계산하는 방법인 몬테카를로 적분방법을 사용하여 구한다. 몬테카를로 적분방법은 해석적으로 어떤 물체의 볼륨의 크기를 구하기가 어려울 때 사용하는 방법으로서, 물체를 포함하는 구간에서 균일한 분포의 난수를 발생시키고 그 중에서 물체 안으로 들어가는 난수의 개수를 전체 난수의 개수로 나누고 물체를 포함하는 구간의 전체 볼륨으로 곱하여 임의의 물체의 볼륨을 근사적으로 구하는 방법이다.

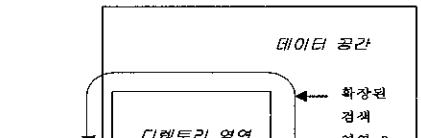


그림 3 2차원 데이타 공간에서의 경계 효과

3. 실험 결과와 분석

본 논문에서 제시된 해석적인 성능 모델을 평가하기 위해서 여러 가지 분포의 데이타에 대해서 실험을 수행

하였다. 실험에서는 다차원 색인 트리로서 X-TREE [9]를 사용하였다. X-Tree는 R-Tree와 같은 다차원 색인 트리로서 노드의 경계가 사각형으로 되어 있고 R-Tree나 R*-Tree 보다 높은 차원에서 좋은 성능을 보이므로 본 논문의 실험에 적용하였다. 그러나 본 논문에서 유도한 해석식은 X-tree나 R-Tree의 특성으로부터 유도하지 않고 노드의 모양이 직사각형이라는 사실로부터 유도하였으므로 노드의 경계가 사각형인 일반적인 인덱스 구조에 적용된다. 최근접 질의 알고리즘은 Rousopoulos 알고리즘[3]을 사용하였다. 트리의 노드에는 디렉토리 노드와 데이타 노드가 있는데 디렉토리 노드의 개수는 데이타 노드의 개수에 비해 매우 작으므로 디렉토리 노드의 일부를 메인 메모리에 미리 적재할 수 있다[9]. 따라서 본 논문에서는 데이타 노드의 검색회수를 고려하였다. 본 논문의 분석치는 질의 구의 반경을 곧바로 계산하는데 반해서 Rousopoulos 알고리즘은 질의 점으로부터 무한대의 반경을 가진 구로 검색을 시작한다. 그러므로 실험에서는 최종적으로 구해진 반경을 가진 질의 구와 겹치는 노드의 개수를 고려하였다. 따라서 다른 알고리즘을 적용하여도 같은 결과가 나올 것이다. 이 실험에서는 30만개의 데이타를 X-Tree에 저장한 후 실험하였고 데이타의 분포는 다음의 3가지 그룹으로 나누어 실험하였다.

1) 균일 분포(Uniform distributions) : 데이타는 균일한 random 분포를 따른다.
2) 정규 분포(Normal distributions) : $N(0,\sigma^2)$, 평균은 0이고 σ는 각 차원에서 $\sigma=2/3$ 이다.
3) 지수 분포(Exponential distributions): $1/\theta\times e^{x/\theta}$, θ는 각 차원에서 $\theta=0.5$ 이다.

정확성을 측정하기 위해서 상대 에러를 다음과 같이 정의하였다 :

$$\text{상대 에러} = \frac{|\text{디스크 검색계산치} - \text{실제 디스크검색회수}|}{\text{실제 디스크 검색 회수}}$$

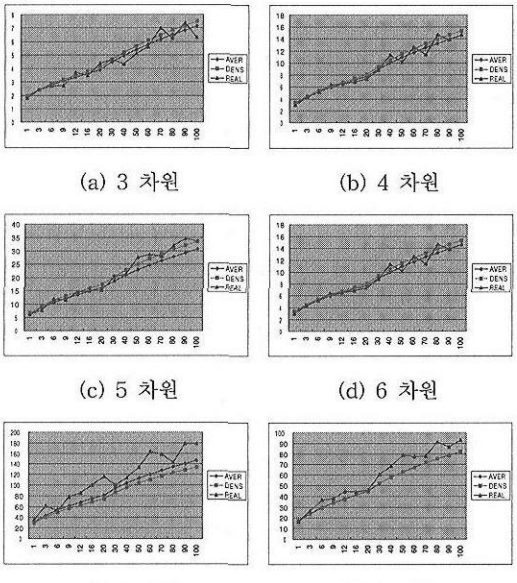
본 실험에서는 두 가지 분석 방법으로 실험하였다: 첫째는 지역 평균 볼륨을 사용한 분석 방법(AVER)이고 두 번째는 노드 밀도를 사용한 방법(DENS)이다. 이 두 가지 분석 방법으로 구한 데이타를 k-최근접 질의를 수행하여 얻은 실제 실험값(REAL)과 비교하였다. 실험 결과는 그림4,5,6에서 보여 주고 있다. AVER에 대해서는 다음과 같은 방법으로 값을 미리 계산하였다: 데이타 공간을 크기가 같은 많은 영역으로 나누고 각 영역에 대해서 범위 질의를 수행했다. 범위 질의의 결과로서 데이타 개수를 얻으면 ω_i를 구하고 식(2)를 적용하여 Ω의 근사치를 구한다.

각 실험에서 평균 디스크 검색 회수는 식(14)와 식(19)로 계산되는데, 몬테카를로 적분법을 사용하여 경계 효과를 제거하였다. 각 실험에서는 30개의 임의의 k-최근접 질의를 수행한 결과의 평균을 구하였다. k의 값은 1에서 100까지의 수로 하였고 3,4,5,6,7,8차원의 데이타 공간에 대해서 실험하였다. 그림 4에서는 균일한 분포를 갖는 데이타 집합에 대한 실험 결과를 보여주고 있다. AVER결과 값와 DENS 결과 값의 차이는 적음을 알 수 있다. 그리고 AVER 과 DENS 는 REAL과 7,8차원에서는 다소간 에러가 있으나 3차원에서 6차원까지는 매우 잘 일치한다. 그림 5에서는 정규분포를 갖는 데이타 집합에 대한 실험 결과이다. 데이타의 편중도(Skewness)가 적은 경우로서 AVER와 DENS의 차이가 적고 REAL과 잘 일치한다. 그림 6에서는 데이타의 편중도가 큰 경우로서 AVER과 DENS 간에 차이가 크게 나타나고 있고 차원이 커지면 그 차이도 커진다. 그러나 DENS와 REAL은 허용 가능한 오차 범위 내에서 잘 일치하고 있다. DENS와 REAL간의 오차는 차원이 높아지면서 커지는 경향이 있으나 에러율의 평균은 10% 미만이다. 그림 7에서는 3차원의 정규 분포를 갖고 각 차원의 분포가 서로 상관계수에 의해 종속되는 데이타에 대한 실험 결과를 보여 준다. 상관계수가 높아지면 AVER은 REAL과의 에러가 증가하는데 반해서 DENS는 상관계수에 관계없이 REAL과 잘 일치하여 AVER보다 정확하다.

이 실험에서는 데이타의 편중도가 높은 비균일 분포의 데이타 집합에 대해서는 지역 평균 볼륨을 사용한 분석보다 밀도 함수를 사용한 분석이 더 정확함을 보여주고 있다. 전반적으로 볼 때 지역 평균 볼륨 방법은 균일한 분포를 갖거나 데이타의 편중도가 낮은 경우에는 잘 일치하지만 데이타의 편중도가 높은 비균일 분포의 데이타 집합이거나 상관계수에 의해 서로 종속된 정도가 높은 데이타에 대해서는 정확도가 낮아진다. 그러나 밀도 함수를 사용한 분석은 균일한 분포나 비균일한 분포 또는 상관계수에 의해 종속된 데이타에서 모두 실제 값과 잘 일치함을 알 수 있다.

표 2 와 표 3은 는 실제 실험 결과와 비교하여 성능 모델에서 계산된 값의 상대 에러를 나타낸다. 차원이 커지면서 에러도 커지는 경향이 있지만 3~5차원에서는 실제 값과 DENS의 분석 값이 매우 잘 일치하고 있고 평균 에러율도 10 % 미만이다. 6~8차원에서는 평균 15 % 이하로 일치하고 있다. 에러는 여러 가지 요인에 의해서 발생하는데 가장 중요한 요인은 차원이다. 차원

이 증가하면서 $Vol(R_k(X))$ 와 질의구의 반경에 대한 추정 값의 에러는 증가한다. 다른 에러의 원인으로는 질의 구의 반경 값이 노드 주위에서 일정하지 않기 때문에 발생하기도 한다. 또한 데이타 공간의 경계선에 인접해 있는 경우는 질의 구의 반경이 증가하는 경계 효과가 나타남으로써 에러가 발생한다. 그리고 차원이 커지면 노드주위의 밀도를 계산할 때 노드 겹침 현상에 의한 효과가 작다는 가정에서 에러가 발생한다. 즉 차원이 증가하면서 노드의 겹침 현상이 더 많이 발생하는데 노드의 겹침 비율이 증가하면 밀도의 계산이 부정확해지므로 반경의 추정이 부정확해지고 에러도 증가한다. 또한 노드와 겹치는 질의 구의 반경은 노드의 밀도로 계산하는데, 차원이 증가하면 노드의 크기가 충분히 작다는 가정이 잘 맞지 않게 된다. 즉 차원이 증가하면 노드의 변의 길이가 증가하고 노드와 겹치는 모든 질의구의 볼륨의 합이 노드의 볼륨보다 커져서, 노드 밖의 데이타 밀도가 노드의 밀도와 차이가 커지게 되므로 질의구의 반경 추정 값의 에러가 커지게 된다. 이러한 여러 가지 요인에 의한 에러는 차원이 증가하면 더 커지는 경향이 있다. 본 논문에서 제시한 성능 모델은 3~8차원 정도에서는 잘 일치하지만 그 이상의 차원에서는 차원이 증가함에 따라 발생하는 에러의 정도가 급격하게 증가한다.

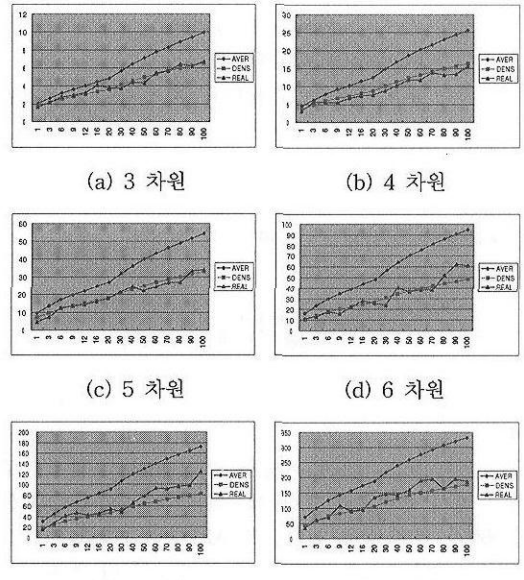


(a) 3 차원 (b) 4 차원

(c) 5 차원 (d) 6 차원

(e) 7 차원 (f) 8 차원

그림 4 균일한 분포를 갖는 데이타 집합에 대한 실험
(수직축 : 노드 검색회수 , 수평축 : k)

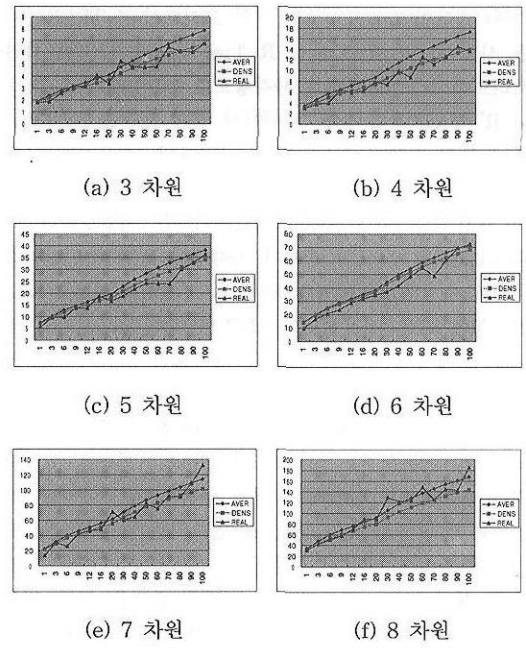


(a) 3 차원 (b) 4 차원

(c) 5 차원 (d) 6 차원

(e) 7 차원 (f) 8 차원

그림 6 지수 분포를 갖는 데이타 집합에 대한 실험
(수직축 : 노드 검색회수 , 수평축 : k)

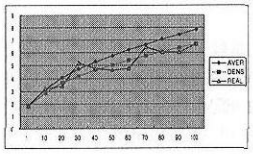


(a) 3 차원 (b) 4 차원

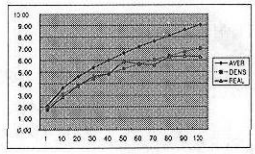
(c) 5 차원 (d) 6 차원

(e) 7 차원 (f) 8 차원

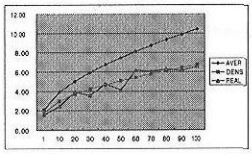
그림 5 정규 분포를 갖는 데이타 집합에 대한 실험
(수직축 : 노드 검색회수 , 수평축 : k)



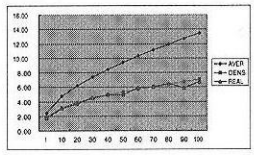
(a) 상관계수=0.0



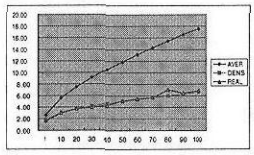
(b) 상관계수=0.1



(c) 상관계수=0.2



(d) 상관계수=0.3



(e) 상관계수=0.4

그림 7 각 차원의 데이타 분포가 서로 상관계수가 있는 3차원의 정규분포에 대한 실험 (수직축 : 노드 검색회수 , 수평축 : k)

표 2 DENS와 REAL의 상대 에러

dimension	uniform	normal	Exponential
3	0.08	0.07	0.05
4	0.07	0.09	0.11
5	0.06	0.10	0.10
6	0.09	0.12	0.11
7	0.11	0.13	0.17
8	0.23	0.10	0.11

표 3 AVER과 REAL의 상대 에러

dimension	uniform	normal	Exponential
3	0.07	0.14	0.35
4	0.05	0.23	0.58
5	0.08	0.18	0.63
6	0.08	0.16	0.79
7	0.11	0.18	0.68
8	0.17	0.10	0.65

4. 결 론

본 논문은 다차원 색인 트리에서 k-최근접 질의의 성능을 해석적으로 계산할 수 있는 성능 모델을 제시하였다. 이 성능 모델을 위해서 본 논문은 두 가지 예측 방법으로 해석식을 유도하였다: (1) 데이타 객체 당 지역 평균 볼륨과 (2) 밀도 함수이다. 지역 평균 볼륨에 의한 방법은 계산이 빠르고 균일한 분포인 경우에 잘 적용되고 밀도 함수에 의한 방법은 데이타 분포가 균일하지 않은 경우에도 잘 적용되는 방법이다. 이 성능 모델은 여러 가지 데이타 분포와 차원에서 최근접 질의의 성능을 측정할 수 있고 평균 에러율이 15% 이내로서 허용 가능한 에러 범위이다. 이러한 결과는 k-최근접 질의의 하이퍼 볼륨을 계산할 때 밀도 함수를 적용하였기 때문이다. 이 모델은 차원이 증가하면 여러 가지 에러 요인에 의해서 에러율이 증가하기 때문에 고차원에서는 에러율이 크다. 그러나 실제로 다차원 색인 트리는 고차원에서 성능이 급격히 저하되는 현상이 발생하고[9] 실제 응용도 중저차원에 많이 있으므로[11,12,13] 본 논문에서 제시한 성능 모델은 실용적으로 가치가 있는 방법이라고 볼 수 있다. 또한 이 모델은 시간이 많이 드는 시뮬레이션 방법을 사용하지 않고 색인 트리를 가지고 계산하므로 시간이 적게 드는 빠른 방법이라고 할 수 있다. 향후의 연구과제로서 고차원에서도 적용이 가능한 k-최근접 질의의 성능 모델을 개발하는 연구가 있을 것이며, 본 모델을 질의의 최적화에 적용할 수 있는 방법에 관한 연구가 있다.

참 고 문 헌

[1] Berchtold S., Böhm C., Keim D., and Kriegel H.-P., "A Cost Model For Nearest Neighbor Search in High-Dimensional Data Space," *Proceedings of the 15th ACM Symposium on Principles of Database Systems*, pp.78-86, 1997.

[2] Hjaltason G.R. and Samet H., "Ranking in Spatial Databases," *Proceedings of the 4th International Symposium on Large Spatial Databases*, pp. 83-95. 1995.

[3] Roussopoulos N., Kelley S., and Vincent F., "Nearest Neighbor Queries," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 71-79, 1995.

[4] Faloutsos C. and Gaede V., "Analysis of n-dimensional Quadtrees Using the Hausdorf Fractal Dimension," *Proceedings of the 22st VLDB conference*, pp 40-50, 1996

[5] Faloutsos C. and Kamel I., "Beyond Uniformity and Independence : Analysis of R-trees Using the Concept of Fractal Dimension," *Proceedings of the 13th ACM Symposium on Principles of Database*

Systems, 1995.

[6] Pagel B., Six H., Toben H. and Widmayer P., "Towards an Analysis of Range Query Performance in Spatial Data Structures," *Proceedings of the 11th ACM Symposium on Principles of Database Systems (PODS)*, pp.214-221, 1993.

[7] Theodoridis Y. and Sellis T., "A Model for the Prediction of R-tree Performance," *Proceedings of the 14th ACM Symposium on Principles of Database Systems*, 1996.

[8] Beckmann N., Kriegel H-P., Schneider R., and Seeger B., "The R*-tree: an efficient and robust access method for points and rectangles," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 322-331, 1990.

[9] Berchtold S., Keim D., and Kriegel H.-P., "The X-Tree: An Index Structure for High-Dimensional Data," *Proceedings of the 22nd International Conference on VLDB*, 1996.

[10] Guttman A., "R-tree: a dynamic index structure for spatial searching," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 71-79, 1984.

[11] Faloutsos C., Ranganathan M., and Manolopoulos Y., "Fast subsequence matching in time series databases," *Proceedings of ACM SIGMOD*, pp 419-429, 1994

[12] Flickner M., et al, "Query by Image and Video Content: The QBIC System," *IEEE Computer*, vol. 28, No. 9, pp 23-32, September, 1995

[13] Wu D., Agrawal D., Abbadi A., Singh A., Smith T., "Efficient Retrieval for Browsing Large Image Databases," *Proceedings of ACM Conference on Information and Knowledge Management*, pp 11-18, 1996

[14] Brin S., "Near Neighbor Search in Large Metric Spaces," *Proceedings of the 21th International Conference on VLDB*, pp. 574-584, 1995.

[15] Cha G.-H. and Chung C.-W., "HG-tree: An Index Structure for Multimedia Databases," *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, pp. 449-452, June 1996.

[16] Cha G.-H., Park H.-H. and Chung C.-W., "Analysis of Nearest Neighbor Query Performance in Multidimensional Index Structures," *Proceeding of 8th International Conference on Database and Expert Systems Application*, pp.498-507, 1997.

이 주 홍

1983년 서울대학교 컴퓨터공학과 졸업(학사). 1985년 서울대학교 컴퓨터공학과 졸업(석사). 1985년 ~ 1989년 한국통신공사 사업지원단 전임연구원. 1989년 ~ 1996년 한국아이비엠 소프트웨어연구소 선임프로그래머. 1993년 ~ 현재 한국과학기술원 정보 및 통신공학과 박사과정. 관심분야는 멀티미디어 데이타베이스, Web 데이타베이스, 데이타 웨어하우스, 데이타 마이닝, 질의 최적화

차 광 호

1984년 부산대학교 계산통계학과(학사). 1989년 한국과학기술원 전산학과(석사). 1997년 한국과학기술원 정보및통신공학과(박사). 1986년 ~ 1987년 삼성반도체통신 시스템개발실 연구원. 1989년 ~ 1996년 데이콤 부가통신가업본부 연구원. 1997년 ~ 현재 동명정보대학교 멀티미디어공학과 전임강사. 1999년 ~ 현재 IBM Almaden Research Center 방문과학자. 관심분야는 객체지향 데이타베이스, 멀티미디어 데이타베이스, 멀티미디어 정보검색

김 형 주

제 26 권 제 1 호(B) 참조

정 진 완

제 26 권 제 2 호(B) 참조