論文
4-4-2

공간 데이터베이스 시스템에서 근사 k-최대근접질의의 처리방법
(The Method to Process Approximate k-Nearest Neighbor Queries in Spatial Database Systems)

선 휘 준(Hwi-Joon Seon)[1] 김 홍 기(Hong-Ki Kim)[2]

요 약

공간 데이터베이스 시스템에서는 주어진 위치에서 가장 근접한 k개의 객체를 찾는 근사 k-최대 근접 질의가 자주 발생한다. 근사 k-최대 근접 질의의 성능을 높이기 위해서는 색인에서 검색되는 노드의 수를 최소화할 수 있어야 한다.

본 논문에서는 기존의 알고리즘을 확장하여 동적인 공간 데이터베이스 환경에서 R-트리 유형의 색인 구조를 이용한 근사 k-최대 근접 질의 처리방법을 제안하고 그 성능을 평가 한다

실험결과에 의하면, 제안된 방법은 객체의 분포 형태, 질의 크기 그리고 근사율에 관계없이 항상 낮은 디스크 접근 횟수를 보였다.

ABSTRACT

Approximate k-nearest neighbor queries are frequently occurred for finding the k nearest neighbors to a given query point in spatial database systems. The number of searched nodes in an index must be minimized in order to increase the performance of approximate k nearest neighbor queries.

In this paper, we suggest the technique of approximate k nearest neighbor queries on R-tree family by improving the existing algorithm and evaluate the performance of the proposed method in dynamic spatial database environments.

The simulation results show that a proposed method always has a low number of disk access irrespective of object distribution, size of nearest neighbor queries and approximation rates as compared with an existing method.

1. 서론

최근의 정보 서비스들은 대용량의 공간 데이터베이스 시스템을 기반으로 하고 있다.

이러한 공간 데이터베이스 시스템에서는 주어진 위치에서 가장 근접한 k(≥1)개의 객체를 찾는 k-최대 근접 질의가 자주 발생한다[3,4,5,6,7,8].

1) 정회원 : 서남대학교 컴퓨터 정보통신학과 조교수
2) 정회원 : 동신대학교 컴퓨터학과 부교수

논문심사 : 2003. 3. 17
심사완료 : 2003. 4. 29

그러나 적중 에러 없이 정확한 k-최대 근접 객체를 검색하기 위해서는 연산 및 보조기억장치 접근을 위한 많은 처리 시간이 요구된다.

최대 근접 질의의 처리비용을 최적화하기 위해서는 색인에서 검색되는 노드의 수를 최소화할 수 있어야 한다. 이를 위해 최대 근접 질의의 처리시 색인에서 방문될 노드들이 정확히 선정되도록 위치 속성에 의한 검색거리 측도인 최적검색거리가 제안되었다[1]. 최적검색거리는 질의기준으로부터 객체 또는 부검색공간들이 반드시 존재하는 거리 중에서 최소의 거리이며, 최대 근접 질의의 처리시 질의기준의 유형에 관계없이 색인에서 검색될 노드들을 정확히 선택하기 위한 검색거리 측도이다.

본 논문에서는 동적인 공간 데이터베이스 환경에서 R-트리 유형의 색인구조를 이용한 근사 k-최대 근접 질의 처리방법을 제안하고 그 성능을 평가한다. 제안될 방법은 최적검색거리에 의한 최대 근접 질의의 처리방법을 확장한 것으로 분기와 한정(branch and bound)기법을 이용하여 색인에서 질의처리에 따른 검색비용을 최소화할 수 있다.

2. 관련연구

최적검색거리를 이용한 방법[1]에서는 최대 근접 질의 처리를 위한 기본개념과 기존의 측정방법이 가지고 있는 문제점을 해결한 새로운 검색거리 측도인 확장최소거리 및 최적검색거리를 정의하였다. N차원 검색공간에서 질의기준 Q와 최소경계사각형(minimum bounding rectangle: M)사이의 가장 가까운 거리인 확장최소거리(eXtended MINimum DISTance: XMINDIST)는 다음과 같다.

$$XMINDIST(Q,M) = \sum_{i=1}^{N} | Q_i - M_i |^2$$

여기에서

$$Q_i = \begin{cases} Q_{Li}, & Q_{Li} > M_{Ui} \\ Q_{Ui}, & Q_{Ui} < M_{Li} \\ 0, & otherwise \end{cases} \quad M_i = \begin{cases} M_{Li}, & Q_{Ui} < M_{Li} \\ M_{Ui}, & Q_{Li} > M_{Ui} \\ 0, & otherwise. \end{cases}$$

(Q_{Li}, Q_{Ui} : 질의범위의 시작과 끝,
M_{Li}, M_{Ui} : 최소경계사각형의 시작과 끝)

정의된 XMINDIST는 최소경계사각형 M에 포함되어 있는 부검색공간들 중에서 질의기준 Q에 가장 근접하고 있는 객체 또는 부검색공간을 결정하기 위한 거리이다.

최적검색거리(the Optimized MINimum value of all DISTances: OMINDIST)는 N차원 검색공간상에서 질의기준 Q로부터 최소경계사각형 M을 구성하는 임의의 N-1차원을 포함할 수 있는 거리들 중 최소거리이며 다음과 같다.

만약 Q와 M이 겹쳐있으면

$$OMINDIST(Q,M) =$$
$$\min_{1 \le i \le N} \Big\{ (| qr_i - mr_i |^2 + \sum_{\substack{i \neq k \\ 1 \le k \le N}} | Qr_k - Mr_k |^2),$$
$$(| qr_i - Mr_i |^2 + \sum_{\substack{i \neq k \\ 1 \le k \le N}} | Qr_k - mr_k |^2),$$
$$(| q_i - Mr_i |^2 + \sum_{\substack{i \neq k \\ 1 \le k \le N}} | Q_k - mr_k |^2) \Big\}$$

여기에서

$$qr_i = \begin{cases} Q_{Li}, & \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \ge \frac{(M_{Li}+M_{Ui})}{2} \\ Q_{Ui}, & otherwise \end{cases}$$

$$Qr_k = \begin{cases} Q_{Lk}, & \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \le \frac{(M_{Lk}+M_{Uk})}{2} \\ Q_{Uk}, & otherwise \end{cases}$$

$$q_i = \begin{cases} Q_{Li}, & \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \ge \frac{(M_{Li}+M_{Ui})}{2} \\ Q_{Ui}, & otherwise \end{cases}$$

$$Q_k = \begin{cases} Q_{Lk}, & \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \ge \frac{(M_{Lk}+M_{Uk})}{2} \\ Q_{Uk}, & otherwise \end{cases}$$

$$mr_i = \begin{cases} M_{Li}, & \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \le \frac{(M_{Li}+M_{Ui})}{2} \\ M_{Ui}, & otherwise \end{cases}$$

$$Mr_k = \begin{cases} M_{Lk}, & \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \ge \frac{(M_{Lk}+M_{Uk})}{2} \\ M_{Uk}, & otherwise. \end{cases}$$

그렇지 않으면

$$OMINDIST(Q,M) =$$
$$\min_{1 \le i \le N} \Big\{ (| qr_i - mr_i |^2 + \sum_{\substack{i \neq k \\ 1 \le k \le N}} | Qr_k - Mr_k |^2),$$
$$(| qr_i - Mr_i |^2 + \sum_{\substack{i \neq k \\ 1 \le k \le N}} | Qr_k - mr_k |^2) \Big\}$$

여기에서

$$qr_i = \begin{cases} Q_{Li}, & Q_{Li} \ge M_{Ui} \\ Q_{Ui}, & Q_{Ui} \le M_{Li} \\ Q_i, & otherwise \end{cases}$$

$$Q_i = \begin{cases} Q_{Li}, \\ Q_{Ui}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \le \frac{(M_{Li}+M_{Ui})}{2} \\ otherwise \end{matrix}$$

$$Qr_k = \begin{cases} Q_{Lk}, \\ Q_{Uk}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \ge \frac{(M_{Lk}+M_{Uk})}{2} \\ otherwise \end{matrix}$$

$$mr_i = \begin{cases} M_{Li}, \\ M_{Ui}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \le \frac{(M_{Li}+M_{Ui})}{2} \\ otherwise \end{matrix}$$

$$Mr_K = \begin{cases} M_{Lk}, \\ M_{Uk}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \ge \frac{(M_{Lk}+M_{Uk})}{2} \\ otherwise \end{matrix}$$

OMINDIST는 질의기준 Q로부터 최소경계사각형 M에 따른 부검색공간에 포함된 객체 또는 부검색공간들이 적어도 하나는 반드시 존재하는 최적의 거리이다.

최적검색거리를 이용한 방법[1]에서는 OMINDIST을 적용함으로써 질의 처리시 검색되어야 하는 노드를 정확히 선택할 수 있음을 정리하였으며, 이는 색인에서 방문되는 전체 노드의 수가 최소화됨을 나타내기 때문에 질의처리에 따른 비용을 최적화함을 보였다.

3. 근사 k-최대 근접 질의 처리방법

근사 질의 처리방법은 절대적인 해답이 요구되지 않는 환경에서 주어진 근사율(approximate rate) ε 만큼의 오류를 허용함으로써 질의처리 성능을 높일 수 있는 방법이다. 특히 데이타 차원이 높아짐에 따라 질의처리에 따른 성능 향상이 더욱 두드러지게 나타난다[8].

근사 질의처리는 다음과 같이 정의되는 주어진 근사율을 기반으로 하여 실행된다. k-최대 근접 질의 처리에서 근사율 ε 의 의미는 색인에서 절대 검색으로 찾을 수 있는 k 번째로 가까운 객체까지의 거리에 (1+ε)을 곱한 거리이내의 객체들 중에서 k개의 가장 가까운 객체를 질의처리 결과로 반환한다는 것이다.

[정의 1] k-최대근접 객체의 검색시 근사율ε (ε ≥ 0.0)은 다음과 같다.

$$\frac{distance(Q, O)}{distance(Q, O_k)} \le (1+\varepsilon)$$

여기에서

Q : 질의기준,

O : 근사 검색의 결과로 반환된 객체,

O_k: 절대 검색에 의한 k번째 최대근접객체

□

색인을 이용하여 주어진 질의 기준으로부터 가장 가까운 k개의 근사 객체를 검색하는 동안 XMINDIST와 OMINDIST의 차이가 아주 크거나 작은 경우가 발생할 수 있다. 이를 위해 질의기준과 최소경계사각형의 모든 꼭지점들 중에서 가장 먼 점까지의 거리인 최대거리(the MAXimum DISTance value of all distances: MAXDIST)를 사용한다. MAXDIST를 사용함으로써 근사 k-최대 근접 질의 처리시 색인에서 적어도 k개의 객체를 검색하고 방문할 노드의 수를 최소화할 수 있다.

[정의 2] N차원 검색공간에서 질의기준 Q와 최소경계사각형 M간의 최대거리 $MAXDIST$를 다음과 같이 정의한다.

$$MAXDIST(Q,M) = \max_{1 \le i \le N} \left\{ \left(|qr_i - Mr_i|^2 + \sum_{\substack{i \ne k \\ 1 \le k \le N}} |Qr_k - Mr_k|^2 \right), \left(|q_i - Mr_i|^2 + \sum_{\substack{i \ne k \\ 1 \le k \le N}} |Q_k - Mr_k|^2 \right) \right\}$$

여기에서

$$qr_i = \begin{cases} Q_{Li}, \\ Q_{Ui}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \ge \frac{(M_{Li}+M_{Ui})}{2} \\ otherwise \end{matrix}$$

$$Qr_k = \begin{cases} Q_{Lk}, \\ Q_{Uk}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \le \frac{(M_{Lk}+M_{Uk})}{2} \\ otherwise \end{matrix}$$

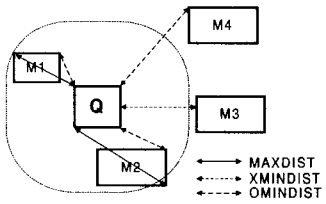
$$q_i = \begin{cases} Q_{Li}, \\ Q_{Ui}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Li}+Q_{Ui})}{2} \ge \frac{(M_{Li}+M_{Ui})}{2} \\ otherwise \end{matrix}$$

$$Q_k = \begin{cases} Q_{Lk}, \\ Q_{Uk}, \end{cases} \quad \begin{matrix} \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \ge \frac{(M_{Lk}+M_{Uk})}{2} \\ otherwise \end{matrix}$$

$$Mr_k = \begin{cases} M_{Lk}, & \text{if } \frac{(Q_{Lk}+Q_{Uk})}{2} \geq \frac{(M_{Lk}+M_{Uk})}{2} \\ M_{Uk}, & otherwise. \end{cases}$$

□

[그림 1]은 질의기준 Q로부터 XMINDIST가 가장 작은 최소경계사각형 M1에 대한 MAXDIST를 기준으로 다음 방문대상에서 노드들을 제외하는 예이다. k-최대 근접 객체의 검색시 MAXDIST의 사용은 하나의 최소경계사각형에 포함되는 모든 객체 또는 부검색공간들과 비교가 가능하므로 색인에서 방문되는 노드의 수를 최소화할 수 있다.



[그림 1] 최대거리 MAXDIST의 예
[Fig. 1] Example of MAXDIST

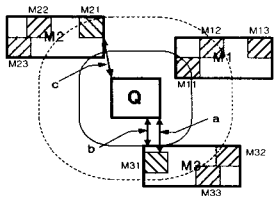
다음은 색인에서 검색하는 동안 방문할 필요가 없는 노드들을 방문대상에서 제외하기 위해서는 기존의 방법을 근사 k-최대 근접 질의의 처리에 적용할 수 있도록 확장한 것이다.

AP1: 질의기준 Q로부터 k번째로 작은OMINDIST(Q,M') 거리를 갖는 최소경계사각형 M'에 대하여 OMINDIST(Q,M')/(1+ε)보다 큰 XMINDIST(Q,M) 거리를 갖는 최소경계사각형 M에 해당하는 노드는 다음 검색대상에서 제외한다.

AP2 : 질의기준 Q로부터 k번째로 작은 OMINDIST(Q,M') 거리를 갖는 최소경계사각형 M'에 대하여 OMINDIST(Q,M')/(1+ε)보다 실제 거리 distance(Q,O)가 더 큰 객체 O를 최대근접 객체 대상에서 제외한다.

AP3 : 질의기준 Q로부터 현재까지 발견된 k번째로 가까운 객체 O의 실제 거리 distance(Q,O)에 대하여 distance(Q,O)/(1+ε)보다 큰 XMINDIST(Q,M) 거리를 갖는 최소경계사각형 M은 검색대상에서 제외한다.

AP4 : 임의의 최소경계사각형 M1,M2,...,Mi가 k개이상의 객체를 포함하고(i≤k) 질의기준 Q로부터 가장 큰 MAXDIST를 갖는 Mi에 대한 MAXDIST(Q,Mi) 거리보다 큰 XMINDIST(Q,M) 거리를 갖는 최소경계사각형 M은 다음 검색대상에서 제외된다.



[그림 2] 절대 검색과 근사 검색의 예(k=2, a<c)
[Fig. 2] Absolute searching and approximate searching

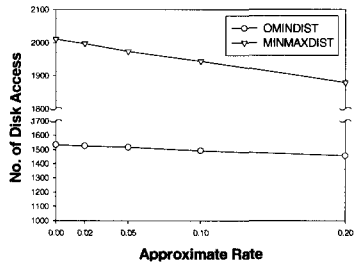
[그림 2]는 k=2인 경우 절대검색과 근사검색의 예를 나타낸 것이다. 절대검색의 경우 질의기준 Q로부터 XMINDIST가 k번째로 작은 M2의 OMINDIST를 기준으로 검색영역과 겹치는 최소경계사각형 M1, M2, M3을 검색하게 된다. 따라서 절대검색의 결과는 M31이 된다. 그러나 근사검색의 경우는 XMINDIST가 작은 순서로 최소경계사각형 M1과 M2를 먼저 검색하여 질의기준 Q로부터 현재까지 발견된 가장 가까운 k번째 최소경계사각형 M21까지의 거리 c와 M3까지의 XMINDIST 거리 b를 비교한다. 그리고 c/b ≤ (1+ε)를 만족하면 최소경계사각형 M3를 검색하지 않고 질의결과로 M31대신 M21 산출한다.

4. 실험 및 성능 평가

이 장에서는 실험을 통하여 제안된 근사 k-최대 근접 질의 처리방법의 성능을 디스크 접근 횟수에 따라 평가한다. 실험에서는 R*-트리[2]에 최적검색거리 OMINDIST와 최대검색거리 MINMAXDIST에 의한 k-최대 근접 질의 처리 알고리즘을 적용한 후 이에 따른 실험결과에 의해 그 성능을 평가한다.

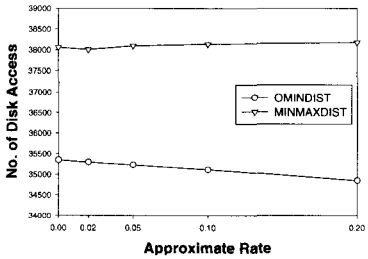
실험에서는 이차원 검색공간에서 중복되지 않고 균일하게 분포하는 30,000개의 사각형 객체를

사용하였으며, 객체들의 크기는 전체 검색공간의 1.0%, 0.001%인 경우를 고려하였다.



[그림 3] 근사율에 따른 디스크 접근 횟수 (질의크기 1.0%)
[Fig. 3] The number of disk accessed vs. an approximate rate(query size 1.0%)

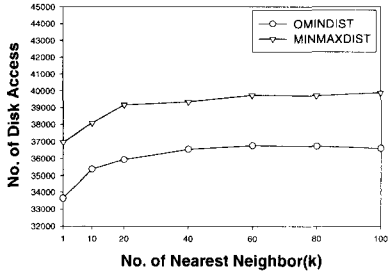
[그림 3]은 객체들의 크기가 0.001%이고 최대근접질의의 크기가 1.0%일 때 근사율에 따른 디스크 접근 횟수를 나타낸 것이다. 그림에서는 근사율이 증가함에 따라 OMINDIST에 의한 최대근접질의의 처리는 거의 일정한 디스크 접근 횟수를 보이며, MINMAXDIST에 의한 최대근접질의의 처리에 비해 낮은 디스크 접근 횟수를 나타낸다.



[그림 4]. 근사율에 따른 디스크 접근횟수 (질의크기 0.001%)
[Fig. 4] The number of disk access vs. an approximate rate(query size 0.001%)

[그림 4]는 객체들의 크기가 1.0%이고 최대근접질의의 크기가 0.001%일 때 OMINDIST와 MIN- MASDIST의 근사율에 따른 디스크 접근 횟수를 나타낸 것이다. 그림에서는 OMINDIST에 의한 질의처리 방법이 근사율이 증가함에 따라 MIN- MAXDIST에 의한 방법보다 디스크 접근 횟수가 점차적으로 낮아지는 경향을 보인다.

[그림 3]과 [그림 4]에서와 같이 OMINDIST에 의한 질의처리 방법이 MINMAXDIST에 의한 질의처리 방법보다 항상 낮은 디스크 접근 횟수를 보이고 특히 질의의 크기가 아주 작을 경우는 성능의 차이가 더욱 커짐을 알 수 있다.



[그림 5] 최대근접객체 수에 따른 디스크 접근 횟수
[Fig. 5] The number of disk access vs. the number of nearest neighbor object

[그림 5]는 객체들의 크기가 1.0% 이고 최대근접 질의의 크기가 0.001%, 근사율이 0.02일 때 최대 근접 객체의 수 k에 따른 디스크 접근 횟수를 나타낸 것이다. 그림에서는 k값이 증가함에 따라 디스크 접근 횟수가 선형적인 증가를 보이며, OMINDIST와 MINMAXDIST의 성능의 차이가 더 커짐을 보인다. 이는 색인에서 검색대상이 되는 노드의 선택시 OMINDIST에 의한 방법이 MIN-MAXDIST에 의한 방법보다 다음 검색대상에서 제외되는 노드들이 더 많기 때문이다.

반복된 실험결과에 의하면 객체들의 모든 분포에 있어서 질의의 크기가 작고 객체들의 크기가 클 때 OMINDIST를 이용한 질의처리 방법이 MINMAXDIST를 이용한 질의처리 방법보다 디스크 접근 횟수, 질의처리시간 면에서 더욱 높은 성능을 나타냈다.

5. 결론

대용량의 공간 데이터베이스 시스템에서 k-최대근접 객체를 찾는 질의의 처리는 많은 디스크 접근

과 질의처리시간을 요구한다. 또한 데이터의 차원이 증가함에 따라 검색비용이 크게 증가할 수 있다.

본 논문에서는 공간 데이터베이스 시스템에서 근사 k-최대 근접 질의를 효율적으로 처리하기 위한 알고리즘을 제안하였다. 제안된 알고리즘은 색인에서 적어도 k개의 객체를 검색하고 방문할 필요가 없는 노드를 정확히 제거하기 위해 최대거리 MAXDIST를 이용한다. 따라서 근사 k-최대 근접 질의 처리시 디스크 접근 횟수와 질의처리시간을 최소화할 수 있다.

실험에서는 근사 k-최대 근접 질의의 처리 성능을 근사율과 최대근접객체의 수에 따라 디스크 접근 횟수와 질의처리시간을 비교 평가하였다. 실험 데이타로는 균일분포를 이루는 사각형 객체를 이용하였다. 실험결과에 의하면 OMINDIST를 이용한 근사 k-최대 근접 질의의 처리는 객체의 분포형태, 객체의 크기에 관계없이 항상 낮은 디스크 접근 횟수를 보였다. 또한 질의의 크기가 작고 객체들의 크기가 클 때 MINMAXDIST를 이용한 질의처리 방법보다 더욱 높은 성능을 나타냈다.

※ 참고문헌

[1] 선휘준 외 1, "최적탐색거리를 이용한 최근접질의의 처리방법의 성능 평가," 한국정보처리학회 논문지, 6권 1호, pp. 32-41, 1999.1.

[2] N.Beckmann, H.Kriegel, R.Schneider and B.Seeger, "The R*-tree: an Efficient and Robust Access Method for Points and Rectangles," Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 322- 331, 1990

[3] C. Faloutsos. et a., "Efficient and Effective Querying by Image Content," Journal of Intelligent Information Systems, Vol. 3, No. 4, pp.231-262, 1994.

[4] D.A.White, R.Jain.,"Similarity Indexing with SS-tree," In Proc. Intl. Conf. on Data Engineering, pp.516-523, 1991.

[5] M.Flinker, et al.,"Query by Image and Video Content: The QBIC Sytem," IEEE Computer, Vol.28, No.9, pp.23-32, 1995.

[6] N.Roussopoulos, et al., "Nearest Neighbor Queries," In Proc. Intl. Conf. on Management of Data, ACM SIGMOD, pp.71-79, 1995.

[7] S.A.Nene, S.K.Nayar, "Closest Point Search in High Dimensions," In Proc. Intl. Conf. on Computer Vision and Pattern Recognition, 1996.

[8] S.Arya. et al., "An Optimal Algorithm for Approximate Nearest Neighbor Searching," In Proc. ACM-SIAM Sym. on Discrete Algorithms, pp.573-582, 1994.

선 휘 준

1988년 목포대학교 전산통계학과(이학사)

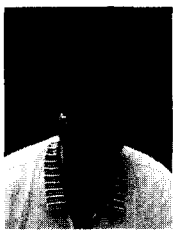
1990년 전남대학교 전산통계학과(이학석사)

1998년 전남대학교 전산통계학과(이학박사)

1997년 3월 ~ 현재 서남대학교 컴퓨터영상. 정보통신학과 조교수

관심분야 : 공간자료구조, 공간데이타베이스, 멀티미디어 시스템, 지리정보시스템,

김 홍기



1984년 전남대학교 계산통계학과(이학사)

1986년 전남대학교 계산통계학과(이학석사)

1996년 전남대학교 전산통계 학과(이학박사)

1991년 현재 동신대학교 컴퓨터학과 부교수

관심분야 : 공간자료구조, 공간데이타베이스, 컴퓨터그래픽스