# 차원 축소 벡터들을 위한 인덱싱 및 검색
## (Indexing and Searching for Reduced-Dimensional Vectors)

정 승 도 † 김 상 욱 †† 최 병 욱 ††
(Seungdo Jeong) (Sang-Wook Kim) (Byung-Uk Choi)

**요 약** 본 논문에서는 각도 성분 근사와 차원 그룹화 기법을 이용한 차원 축소 기법에 의해 변환된 축소 데이터를 색인하고 검색하기 위해서 해결되어야 하는 문제들을 분석하고 이를 해결하기 위한 방법을 제안한다. 또한 다양한 실험에 의한 성능 평가를 통하여 제안하는 방법의 우수성을 규명한다.
**키워드** : 멀티미디어 정보검색, 다차원 색인, 질의 처리, 차원 축소

***Abstract*** In this paper, we first address the problems associated with indexing and searching for reduced-dimensional vectors, which are reduced by using a combination of angle approximation and dimension grouping. Then, we propose a novel method to solve the problems. We also show the superiority of the proposed method by performing extensive experiments with synthetic and real-life data sets.
**Key words** : Multimedia Information Retrieval, Multi-dimensional Indexing, Query Processing, Dimensionality Reduction

## 1. 서 론

멀티미디어 정보 검색의 효율을 높이기 위하여 특징 벡터들을 색인할 수 있는 많은 색인 구조(index structure)들이 제안되었다[1,2]. 그러나 고차원 특징 벡터를 사용해야 하는 멀티미디어 정보 검색에 적용할 경우, 특징 벡터의 차원이 증가할수록 색인 구조를 이용한 검색 기법들의 성능이 급격히 떨어지는 현상이 발생한다. 이러한 현상은 차원의 저주(dimensionality curse)라고 알려져 있다[3].

차원 축소(dimensionality reduction) 기법은 차원의 저주 문제를 해결하기 위한 대표적인 방법의 하나이다. 차원 축소 기법이 적용된 검색에서 검색 효율을 높이기 위해서는 필터링 단계에서 착오 채택(false alarm)을 최소화 하면서 동시에 착오 기각(false dismissal)이 발생하지 않도록 해야 한다. 착오 기각이 발생하는 것을 방지하기 위해서는 축소된 특징 벡터 공간에서의 두 벡터간의 거리는 원 특징 벡터 공간에서 벡터간의 거리 보다 항상 작거나 같아야 한다는 하한 조건을 만족해야만 한다[4].

참고문헌 [5]와 [6]에서는 각도 근사 기법과 차원 그룹화를 통한 차원 축소 기법을 제안한 바 있다. 검색에 있어서는 차원 축소 기법에 의해 축소된 저차원 벡터들과 질의 벡터를 순차적으로 비교하는 순차 검색 방식을 사용하였다. 그러나 빠른 검색을 위해서는 축소된 저차원 벡터 공간을 대상으로 하는 색인 기법이 요구된다. 본 연구에서는 기존의 다차원 색인 기법인 $R^*$-tree 색인 구조를 기반으로 차원 그룹화를 통해 축소된 저차원 특징 벡터들을 효율적으로 색인하고 검색하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 참고문헌 [5,6]의 내용을 간략히 요약한다. 제 3장에서는 기존 연구를 다차원 색인 구조와 결합하기 위한 문제를 살펴본다. 제 4장에서는 3장에서 살펴본 문제에 대한 해결 방안을 자세히 살펴보고 5장에서는 실험을 통하여 제안한 방법의 우수성을 검증한다. 끝으로 제 6장에서 본 논문을 요약하고, 결론을 내린다.

· 본 연구는 2009년도 한국학술진흥재단의 지원을 받아 수행된 연구임 (KRF-2009-013-D00104).

† 정 회 원 : 한양사이버대학교 정보통신공학과 전임강사
sdjeong@hanyang.ac.kr
†† 종신회원 : 한양대학교 컴퓨터공학부 교수
wook@hanyang.ac.kr
buchoi@hanyang.ac.kr
논문접수 : 2009년 9월 2일
심사완료 : 2009년 11월 3일


정보과학회논문지: 데이타베이스 제37권 제1호(2010.2)

## 2. 관련 연구

### 2.1 유클리드 거리 근사

두 벡터 $X$, $Y$에 대한 유클리드 거리 함수 $D(X,Y)$는 식 (1)과 같이 정의된다.

$$D(X,Y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}=\sqrt{\|X\|^2+\|Y\|^2-2<X,Y>} \quad (1)$$

Cauchy-Schwartz 부등식은 식 (2)와 같이 두 벡터의 놈(norm)을 이용하여 벡터 내적의 상한(upper bound)을 정의한 것이다.

$$<X,Y>\le \|X\|\|Y\| \quad where\ \|X\|=\sqrt{\sum_{i=1}^{n}x_i^2} \quad (2)$$

따라서 Cauchy-Schwartz 부등식을 이용한 유클리드 거리 근사 함수 $D_{cs}(X,Y)$를 식 (3)과 같이 정의된다.

$$D_{cs}(X,Y)=\sqrt{\|X\|^2+\|Y\|^2-2\|X\|\|Y\|} \quad (3)$$

식 (2)의 Cauchy-Schwartz 부등식의 정의로부터 $D_{cs}(X,Y)$는 $D(X,Y)$의 하한(lower bound)이 된다.

그러나 $D_{cs}(X,Y)$에서는 두 벡터간의 각도 성분이 완전히 무시되므로 $D(X,Y)$와의 오차가 커지는 단점이 있다. 이를 해결하기 위해서 참고문헌 [5]에서는 기준 벡터(reference vector) 개념을 도입하여 각도 성분을 근사하는 각도 근사 기법을 제안하였다. 각도 근사 기법에서는 기준 벡터와 모든 데이터 벡터와의 각도 성분을 사전에 계산하여 저장해 두고, 질의 벡터가 주어질 때 질의 벡터와 각 데이터 벡터와의 각도 성분은 단순 뺄셈만으로 근사한다. 근사된 각도 성분과 놈을 이용한 거리 함수 $D_A(X,Y)$는 식 (4)와 같다[5]. 여기서 $\widetilde{\theta_{XY}}$는 각도 근사 기법에 의해 근사된 $X$와 $Y$ 사이의 각도 성분이다.

$$D_A(X,Y)=\sqrt{\|X\|^2+\|Y\|^2-2\|X\|\|Y\|\cos\widetilde{\theta_{XY}}} \quad (4)$$

**정리 1.** 각도 근사 기법을 이용한 거리 함수 $D_A(X,Y)$는 $D(Q,X)$의 하한 함수이다.

**증명.** 참고문헌 [5] 참조. ■

정리 1로부터 필터링 단계에서 $D_A(X,Y)$를 이용하는 2단계 검색은 정답을 보장함을 알 수 있다.

### 2.2 차원 그룹화를 이용한 차원 축소

$D_A(X,Y)$를 위하여 근사된 각도의 오차가 작아지기 위해서는 기준 벡터가 데이터 벡터와 질의 벡터에 의하여 형성되는 평면에 가까워야 한다. 그러나 고차원 공간에서는 이 세 벡터가 동일 평면상에 존재할 확률은 매우 낮다. 반면, 저차원 공간에서는 임의의 세 벡터가 하나의 평면에 가까워질 확률이 상대적으로 높다. 참고문헌 [6]에서는 이러한 특성을 반영하기 위하여 고차원 데이터 벡터를 낮은 차원의 데이터 벡터들의 집합으로 간주하는 차원 그룹화 기법을 제안하였다.

차원 그룹화 기법에서는 고차원 데이터 벡터의 속성값을 소수의 그룹으로 분리하여 하위벡터(subvector)들의 집합으로 표현하고, 각 하위벡터에 대하여 개별적으로 각도 근사 기법을 적용한다. 이를 위해 사전에 하위벡터별 놈과 기준 벡터와의 각도 성분을 저장한다. 이는 고차원 벡터로부터 축소된 저차원 벡터로, 차원 그룹화에 의해 차원 축소된 $k$차원 저차원 데이터 벡터는 식 (5)와 같다. $D(X,Y)$에 대한 거리 근사 함수 $D_{GA}(X,Y)$는 식 (6)과 같다. 여기서 $X_i$는 각 하위벡터이고 $\theta_{X_i}$는 해당 하위벡터와 기준 벡터와의 각도 성분이다.

$$X_{GA}=\left[\|X_1\|,\theta_{X_1},\|X_2\|,\theta_{X_2},\cdots,\|X_k\|,\theta_{X_k}\right] \quad (5)$$

$$D_{GA}(X,Y)=\sqrt{\sum_{i=1}^{k}\left(\|X_i\|^2+\|Y_i\|^2-2\|X_i\|\|Y_i\|\cos\theta_{\widetilde{X_iY_i}}\right)}$$
$$where\quad \theta_{\widetilde{X_iY_i}}=\left|\theta_{X_i}-\theta_{Y_i}\right| \quad (6)$$

**정리 2.** 차원 그룹화에 의해 축소된 저차원 벡터를 이용하는 거리함수 $D_{GA}(X,Y)$는 $D(Q,X)$의 하한 함수이다.

**증명.** 참고문헌 [6] 참조. ■

정리 2로부터 필터링 단계에서 $D_{GA}(X,Y)$를 이용하는 2단계 검색 역시 정답을 보장함을 알 수 있다.

## 3. 연구 동기

R*-tree 기반의 검색에서, 필터링 단계에서는 방문한 R*-tree 노드 내에서 질의 MBR과 겹치는 데이터 MBR에 대해서만 재귀적으로 해당 MBR과 대응하는 하위 트리를 검색하여 후보 집합을 선별한다. 이는 질의 MBR과 겹치지 않은 데이터 MBR은 그 내부의 어떤 벡터도 질의 벡터와 유클리드 거리가 $\varepsilon$ 이내일 수 없으므로, 그 하위 단계는 더 이상 검색할 필요가 없기 때문이다. 후처리 단계에서는 후보 집합 전체에 대하여 실제 유클리드 거리를 계산함으로써 최종적으로 정답을 검증한다.

참고문헌 [6]에서는 고차원 벡터를 저차원 벡터로 축소하는 방법을 다루고 있다. 여기서, 축소된 저차원 벡터가 존재하는 공간은 벡터의 놈과 각도를 차원 성분으로 갖는다. 축소된 저차원 벡터를 기존의 R*-tree를 이용하여 색인하기 위해서는 다음 사항들을 고려해야 한다.

첫째, 데이터를 색인하기 위한 색인 공간에서의 데이터 MBR이 원 공간에서 가까운 벡터들의 집합을 표현하는가? 효율적인 색인 및 검색을 위해서는 원 공간에서 근접한 위치에 존재하는 데이터 벡터들이 변환된 색인 공간에서 동일한 MBR내에 포함되어야 한다.

둘째, 검색 시 색인 공간에서 질의 벡터와 MBR간에

어떠한 거리 함수를 적용할 것인가? 본 논문에서 다루는 색인 공간에서 벡터의 놈 성분을 다루는 $N$ 차원은 원 공간의 유클리드 거리와 직접적인 연관을 갖지만 기준 벡터와의 각도 성분을 다루는 $A$ 차원은 그렇지 못하다. 따라서 색인 공간 내에서의 질의 벡터와 MBR간의 유클리드 거리는 원 공간 내 그들 간의 거리를 반영하는데 직접적인 도움이 되지 않기 때문에 유클리드 거리 함수를 직접 사용할 수 없다. 각도 근사와 차원 그룹화를 이용한 차원 축소 기법에 의해 축소된 색인 공간에서는 이러한 문제를 해결하기 위한 새로운 전략이 요구된다.

## 4. 색인 및 검색 방법

### 4.1 *NA* 공간

차원 그룹화를 이용한 차원 축소 기법은 n차원 벡터를 유클리드 벡터 공간 $R^n$로부터 <*Norm, Angle*>로 구성되는 저차원의 $NA$ 공간으로 변환하여 차원을 축소한다. 색인은 $NA$ 공간상의 MBR $M$을 기반으로 이루어진다. 이때, $NA$ 공간에서 MBR 내의 벡터 중, 질의 벡터로부터 원 유클리드 거리가 가장 가까운 벡터를 거리 최소 벡터 $m$이라 정의한다.

R*-tree는 색인 공간에서 MBR 단위로 색인을 수행한다. 효율적인 색인을 위해서는 각 MBR이 원 공간에서 유클리드 거리가 가까운 벡터들을 포함해야 한다.

**정리 3.** $NA$ 공간의 임의의 MBR $M$은 원 공간에서 유클리드 거리가 가까운 벡터에 의해 형성된다.

**증명.** 본 논문에서의 색인 공간에서 임의의 MBR $M$의 영역은 $N_1 \le M_N \le N_2, \theta_1 \le M_\theta \le \theta_2$으로 표현된다. 원 공간에서 이 영역은 그림 2에서와 같이 부채꼴의 끝 부분이다. 기준 벡터는 모든 벡터에 대하여 동일하게 사용하기 때문에 각도 성분은 하나의 축을 기준으로 표현이 가능하다. 따라서 $NA$ 공간의 임의의 MBR에 대응하는 영역에 포함되는 원 유클리드 공간의 벡터는 $\{V | N_1 \le \| V \| \le N_2, \theta_1 \le V_\theta \le \theta_2\}$와 같고 이는 원 유클리드 공간에서 하나의 클러스터를 이루고 있으며, 유클리드 거리가 상대적으로 가까운 벡터들의 집합이다. ■

정리 3을 통하여 NA 공간의 MBR이 원 공간의 클러스터를 잘 반영함을 검증함으로써 효율적인 색인이 가능함을 보였다.

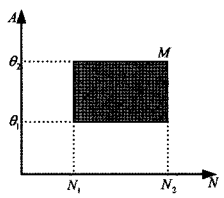


그림 1 $NA$ 공간에서의 MBR

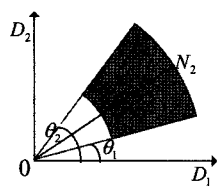


그림 2 원 공간의 대응 영역

### 4.2 MBR 확장

R*-tree 색인 기반 검색에서는 색인 공간에서 질의 벡터로부터 유클리드 거리가 ε이내인 MBR을 재귀적으로 찾아가면서 검색한다. 이 때, 착오 기각이 발생하지 않기 위한 조건은 질의 벡터로부터 특정 MBR까지의 색인 공간에서의 거리가 MBR 내부의 임의의 벡터까지의 거리보다 항상 작거나 같아야 한다는 것이다. 그러나 $NA$ 공간에서의 두 벡터간 유클리드 거리는 원 공간에서의 두 벡터의 유클리드 거리와는 다르다.

만약 $V \in M$인 데이터 벡터 중, $NA$ 공간에서 질의 벡터로부터 가장 가까운 데이터 벡터 $m$이 실제 유클리드 공간에서도 역시 가장 가까운 벡터이면 거리 최소 벡터 $m$을 기준으로 해당 MBR을 필터링 하더라도 착오 기각은 발생하지 않는다. 그러나, 특정 MBR내의 벡터 중 질의 벡터와의 유클리드 거리가 거리 최소 벡터보다 작은 경우 착오 기각이 발생하는 문제가 있다.

$V \in M$인 데이터 벡터 중, $NA$ 공간에서 질의 벡터 Q로부터 가장 가까운 데이터 벡터 $m$이 $\| Q \| \ge \| m \| \ge \| Q \| \cos(\theta_Q - \theta_m)$의 조건에 해당할 경우 이러한 현상이 발생한다. 그림과 같이 원 유클리드 공간에서 MBR의 일부 영역이 영역과 겹치는 것을 알 수 있다. 이 경우 $m$을 기준으로 해당 MBR을 필터링할 경우 질의 영역과 겹치는 영역에 존재하는 벡터는 착오 기각이 되는 문제가 발생한다.

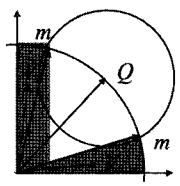
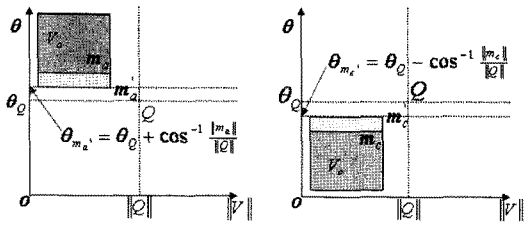


그림 3 $\| Q \| \ge \| m \| \ge \| Q \| \cos(\theta_Q - \theta_m)$의 경우

따라서 착오 기각 없이 검색하기 위해서는 그림 4와 같이 $m$이 질의 벡터에 가까워지도록 MBR을 확장하여야 한다. 이 때, 확장된 MBR에 대한 놈은 변화가 없으며, 각도 $\theta_{m'}$은 식 (7)과 같다.

$$\theta_{m'} = \begin{cases} \theta_Q + \cos^{-1}\dfrac{\| m \|}{\| Q \|}, \theta_m > \theta_Q \\ \theta_Q - \cos^{-1}\dfrac{\| m \|}{\| Q \|}, \theta_m \le \theta_Q \end{cases} \quad (7)$$

본 논문에서는 검색 과정에서 착오 기각 없이 검색하기 위하여 MBR을 확장해야 하는 MBR 확장 방법을 제안하였다. MBR 확장은 질의 벡터에 의존하기 때문에 검색 단계에서 이루어져야 한다. 이 과정은 약간의 CPU 연산 시간을 요구하지만 디스크 접근을 요구하지 않으므로 전체적인 검색 속도에 미치는 영향은 매우 적다.



(a) $\theta_m > \theta_Q$인 경우 (b) $\theta_m \le \theta_Q$인 경우

그림 4 *NA* 공간에서 확장된 MBR에 대한 거리 최소 벡터 m′의 위치 관계

### 4.3 검색 알고리즘

검색을 위하여 먼저, 질의 벡터의 놈과 질의 벡터와 기준 벡터간의 각도 $\theta_Q$를 계산한다. 구해진 놈과 각도 성분, 그리고 유사 허용치 $\varepsilon$를 이용하여 미리 작성된 R*-tree 검색을 수행한다. 이 과정에서는 질의 벡터로부터 특정 MBR까지의 거리를 형성하는 벡터 $m$을 구하고 $\|Q\| \ge \|m\| \ge \|Q\|\cos(\theta_Q - \theta_m)$의 조건에 해당할 경우 MBR의 각도 성분 확장을 통해 최종적인 거리 최소 벡터를 결정한다. 질의 벡터로부터 거리 최소 벡터까지의 거리는 $D_{GA}(X,Y)$에 의해 빠르게 계산한다. 계산된 거리가 유사 허용치 이내일 경우 해당 MBR과 대응되는 하위 단계 서브트리에 대하여 재귀적으로 탐색해 나간다. 계산된 거리가 유사 허용치 이상일 경우, 해당 MBR과 대응되는 하위단계 서브트리는 더 이상 탐색하지 않는다. R*-tree 검색이 끝나면 $D_{GA}(X,Y)$에 의해 근사된 유클리드 거리가 유사 허용치 $\varepsilon$보다 작은 정답 후보 집합이 결정된다. 후처리 단계에서 정답 후보 집합에 대해서 원 고차원 공간에서의 질의 벡터와의 유클리드 거리를 구하여 유사 허용치와 비교함으로써 최종 정답을 결정한다.

## 5. 성능 평가

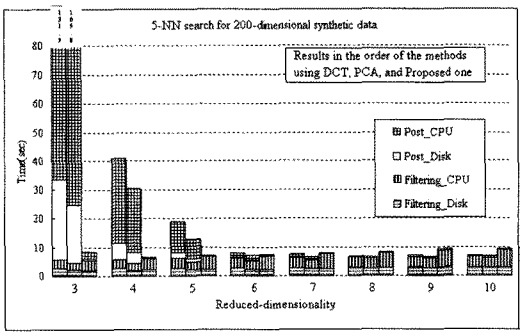
본 논문에서는 성능을 평가하기 위해 합성 데이터와 실제 데이터인 Corel 영상 데이터를 사용하였다. 합성 데이터는 25차원에서 200차원까지 다양한 차원 수를 갖는 벡터들의 집합이다. Corel 영상 데이터는 32차원의 특징 벡터로 표현되는 총 68,040장의 영상이다.

PCA를 이용한 차원 축소 기법과 DCT를 이용한 차원 축소 기법에 의해 축소된 저차원 데이터를 기존의 R*-tree에 색인한 후 검색 성능을 비교하였다. 성능 평가 지수는 검색 처리 시간으로 필터링과 후처리 단계에 대하여 각각 디스크 접근 시간과 CPU 사용 시간을 나누어 비교하였다. 각 실험마다 총 100개의 임의의 질의 벡터에 대한 결과의 평균을 구하였다.

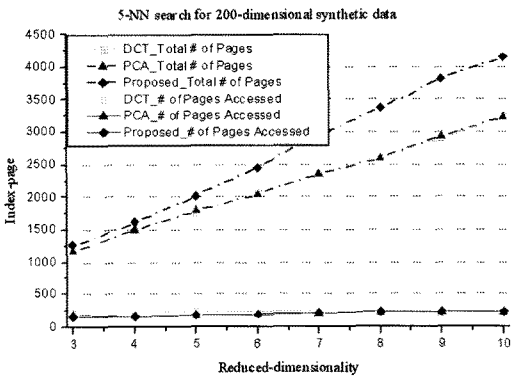
하드웨어 플랫폼은 2.8G Pentium Ⅳ와 512MB의 주기억장치가 장착된 PC이며, 소프트웨어 플랫폼은 MS Windows 2000 및 Visual C++6.0이다. R*-tree를 위한 페이지 크기는 4096 bytes로 설정하였고, 각 데이터 벡터는 독립된 페이지로 디스크에 저장하였다.

일반적으로 축소된 공간의 차원 수가 클수록 정보의 손실이 작으며, 이 결과 필터링 단계 후의 후보 개수가 작아지기 때문에 후처리 시간이 줄어들게 된다. 반면, 축소된 공간의 차원수가 커질수록 R*-tree의 크기가 커지기 때문에 필터링 단계를 위한 처리 시간은 늘어나게 된다. 실험 1에서는 저차원 공간의 차원에 따른 성능을 비교하기 위하여 축소 차원수를 3에서 10까지 변화시키면서 검색 시간과 접근 페이지 수를 비교하였다. 데이터 벡터의 원 차원은 200이고, 데이터 개수는 100,000개로 구성되어 있다.

그림 5는 축소 차원 수의 변화에 따른 전체 검색 시간과 접근 페이지 수를 보여주고 있다. 기존의 PCA나 DCT를 이용한 방법과 제안하는 방법 모두 축소 차원 수가 증가할수록 검색 시간이 현저히 줄어듬을 알 수 있다. 그러나 PCA나 DCT를 이용한 방법의 경우 필터링 단계 후 후보 개수가 많기 때문에, 고차원 데이터의 속성 값을 모두 사용해야 하는 후처리 시간이 상대적으로 길다.




(a) 검색 시간




(b) 접근 페이지 수

그림 5 축소 차원 수 변화에 따른 성능 비교

제안하는 기법의 경우, MBR을 확장하는 연산으로 인해 필터링을 위한 연산 시간은 상대적으로 길지만 필터링 효과가 크기 때문에 후처리 시간이 매우 작다. 최적의 경우는 제안하는 기법은 축소 차원수가 4, PCA나 DCT를 이용한 방법은 8인 경우이고 이때 검색 시간은 거의 같다. 그러나 색인에 사용된 전체 페이지 수를 비교해 보면, 기존의 방법은 제안하는 기법에 비해 1.6배의 부가적인 저장 공간을 요구한다. 전체 검색 시간을 비교해보면, 축소 차원 수가 3일 때 제안하는 기법은 DCT를 이용한 방법의 약 15배, PCA를 이용한 방법의 약 12배 빠르다. 축소 차원 수가 5인 경우에는 DCT를 이용한 차원 축소 방법의 약 3배, PCA를 이용한 차원 축소 방법의 약 2배 빨라짐을 알 수 있다.

그림 5(b)는 전체 페이지 수와 접근 페이지 수를 보여준다. 제안하는 기법의 경우 기존의 방법에 비하여 각도 성분을 저장해야 하기 때문에 페이지 수가 다소 증가하지만, 실제 검색 시 접근한 페이지 수는 PCA를 이용한 방법과 거의 같음을 알 수 있다. 이는 제안하는 기법에서 정확한 검색을 위해 MBR을 확장하더라도 검색해야 할 영역에 크게 영향을 주지 않으면서 효과적으로 검색이 가능함을 보여주는 것이다.

실험 2에서는 원 공간의 차원 수 변화에 따른 성능을 비교하였다. 사용된 데이터 벡터 수는 100,000개이고, 축소된 차원 수는 4로 고정하여 5-NN 검색을 수행하였다. 그림 6에서 본 실험에 대한 결과를 보였다.

기존의 DCT나 PCA를 이용한 방법의 경우 데이터 원 차원 수가 증가함에 따라 필터링 후 후보 개수가 증가하여 후처리 시간이 급격히 증가함을 알 수 있다. 따라서 전체 검색 성능이 떨어지는 현상을 보인다. 반면, 제안하는 방법의 경우 원본 차원 수가 증가하면 필터링을 위한 처리 시간은 기존 방법과 비슷한 수준으로 증가하지만 필터링 효과가 뛰어나기 때문에 후처리에 대한 부담이 거의 없다. 데이터 차원이 200일 경우, 제안하는 방법은 DCT를 이용한 방법에 비해 6.1배, PCA를 이용한 방법에 비해 4.6배 성능 향상이 있음을 확인하였다. 이는 제안하는 방법이 데이터 벡터가 고차원화 되어가는 멀티미디어 정보 검색에 매우 적합하다는 것을 보이는 것이다.

다음으로, 실험 3에서는 데이터 벡터의 개수 증가에 따른 검색 성능을 비교하였다. 데이터 벡터의 차원은 200으로, 축소된 차원 수는 4로 고정하여 5-NN 검색 실험을 진행하였다. 그림 7에 본 실험에 대한 결과를 보였다.

기존의 DCT나 PCA를 이용한 방법의 경우, 데이터 벡터의 개수가 증가함에 따라 검색 성능이 급격히 저하됨을 알 수 있다. 이는 데이터 벡터의 개수가 증가하면서 데이터가 넓은 영역에 분포하게 되고, 따라서 후보 개수가 급격히 증가하는 이유에 기인한다. 반면, 제안하는 방법은 차원이 증가 할수록 필터링 단계에서의 처리 시간은 증가하지만, 필터링 효과가 뛰어나기 때문에 후처리 시간에 대한 부담이 크게 줄어들게 된다. 따라서 전체 검색 성능이 기존 방법에 비해 향상됨을 알 수 있다.

제안하는 방법은 DCT를 이용한 방법에 비해 6배에서 최대 6.6배까지 성능 향상이 있음을 확인할 수 있다. PCA를 이용한 방법에 비해서는 4.5배에서 최대 5.4배까지 성능 향상이 있었다. 실험 결과를 통해, 기존 방법들에 비해 제안하는 방법이 대용량화 되어가는 멀티미디어 정보 검색에 더 적합함을 알 수 있다.

마지막으로, Corel 영상 데이터를 이용하여 기존 방법들과의 성능을 비교하였다.

Corel 데이터는 합성 데이터에 비해 매우 밀집되어 있어 상대적으로 필터링 효과가 크지 않다. 따라서 후처리 단계가 길어지는 특징을 갖는다. 축소된 차원 수가 3인 경우, 제안하는 방법의 경우에도 후보 개수가 많기 때문에 후처리 시간의 비율이 상대적으로 높다는 것을 알 수 있다. 그러나 필터링 후의 후보 개수가 기존의 방법들에 비해 훨씬 적기 때문에 가장 우수한 검색 성능을 보였고, DCT를 이용한 방법에 비해 8배까지 성능 향상이 있음을 알 수 있다. 축소된 차원 수가 5인 경우

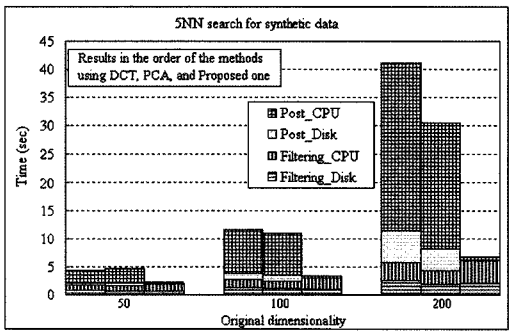



그림 6 차원 수 변화에 따른 성능 비교

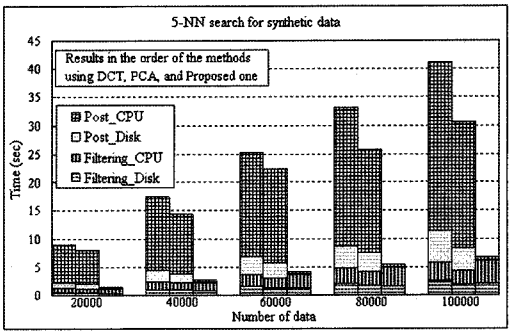



그림 7 데이터 벡터의 개수에 따른 성능 비교

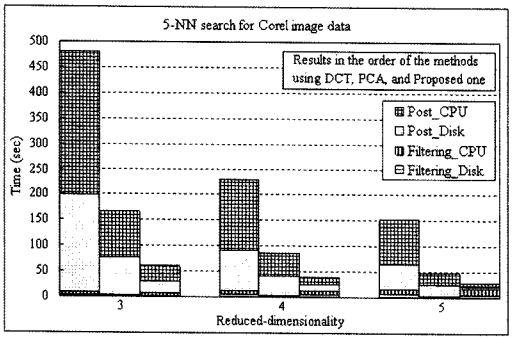



그림 8 Corel 영상 데이터에 대한 축소 차원 수의 변화에 따른 성능 비교

에 대해서도 5.4배의 성능 향상을 보였다. PCA를 이용한 방법에 비해서는 1.8배에서 2.8배까지 성능 향상이 있음을 확인하였다.

## 6. 결 론

본 연구에서는 각도 근사와 차원 그룹화를 이용하여 축소된 저차원 벡터를 효과적으로 색인하고 검색하는 방법에 관하여 다루었다.

R*-tree는 색인 공간내의 데이터 벡터를 MBR 단위로 색인한다. 따라서 효율적인 색인을 위해서는 원 유클리드 공간에서 밀집된 데이터 벡터가 색인 공간에서 하나의 MBR을 형성하여야 한다. 본 논문에서는 유클리드 공간과 NA 공간의 벡터들 간의 위치 관계를 분석함으로써, NA 공간의 MBR이 원 공간에서 유클리드 거리가 가까운 벡터 군집에 의해 형성됨을 확인하였고, 이를 통해 효과적인 색인이 가능함을 보였다. 또한, 착오 기각 없이 검색하기 위해 MBR 확장 기법을 제안하였다.

실험 결과, 제안된 기법은 기존의 DCT를 이용한 방법과 비교하여 3배에서 15배까지, PCA를 이용한 방법과 비교해서는 2배에서 12배까지의 성능 개선 효과를 가지는 것으로 나타났다. 특히, 차원이 높아질수록 성능 개선 효과가 더욱 커지는 경향을 보였고, 원본 데이터 수가 증가해도 성능 변화가 적음을 확인할 수 있었다. 이는 고차원화, 대용량화 되어가는 멀티미디어 정보 검색 분야의 특성을 고려할 때, 매우 바람직한 결과라 할 수 있다.

## 참 고 문 헌

[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. Int'l. Conf. on Management of Data, ACM SIGMOD*, pp.322-331, 1990.

[2] P. Ciaccia, M. Patella, and P. Zezula, "M-tree: An Efficient Access Method for Similarity Search in Metric Spaces," In *Proc Int'l. Conf. on Very Large Data Bases, VLDB*, pp.426-435, 1997.

[3] R. Weber, H. J. Schek, and S. Blott, "A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces," In *Proc. Int'l. Conf. on Very Large Data Bases, VLDB*, pp.194-205, 1998.

[4] T. Seidl and H.-P. Kriegel, "Efficient User-adaptable Similarity Search in Large Multimedia Databases," In *Proc. Int'l. Conf. on Very Large Data Bases, VLDB*, pp.506-515, Aug. 1997.

[5] S. Jeong, S.-W. Kim, K. Kim, and B.-U. Choi, "An Effective Method for Approximating the Euclidean Distance in High-Dimensional Space," In *Proc. Int'l. Conf. on Database and Expert Systems Applications*, pp.863-872, 2006.

[6] S. Jeong, S.-W. Kim, and B.-U. Choi, "Dimensionality Reduction for Similarity Search with the Euclidean Distance in High-Dimensional Application," In *Multimedia Tools and Applications*, vol.42, no. 2, pp.251-271, 2009.

정 승 도

1999년 한양대학교 전자·전자통신·전파공학과 학사. 2001년 한양대학교 전자통신전파공학과 석사. 2007년 한양대학교 전자통신컴퓨터공학과 박사. 2009년~현재 한양사이버대학교 정보통신공학과 전임강사. 관심분야는 멀티미디어정보검색, 증강현실, 텐서 기반 응용 등

김 상 욱

1989년 서울대학교 컴퓨터공학과 학사. 1991년 한국과학기술원 전산학과 석사. 1994년 한국과학기술원 전산학과 박사. 현재 한양대학교 컴퓨터공학부 부교수. 관심분야는 데이타베이스 시스템, 저장 시스템, 데이타 마이닝, 멀티미디어 정보 검색, 공간 데이터베이스/GIS, 주기억장치 데이타베이스, 트랜잭션 관리 등

최 병 욱

1973년 한양대학교 전자공학과 학사. 1978년 일본 게이오대학교 전기공학과 석사 1981년 일본 게이오대학교 전기공학과 박사. 1981년~현재 한양대학교 컴퓨터공학부 교수. 관심분야는 멀티미디어정보검색, 컴퓨터 비전, e-Learning, Intelligent Tutoring System(ITS) 등