내용 기반 이미지 검색을 위한 복합 질의문 계획 생성 기법

(Generating Combined Query Plan for Content-Based Image Retrieval)

박 미 화 † 엄 기 현 ††
(Mee-Hwa Park) (Ky-Hyun Um)

요 약 이미지 데이타는 텍스트 데이타와는 달리 다양한 색상과 모양, 질감과 같은 비정형적인 특징을 가진다. 따라서 이미지 데이타베이스는 텍스트 기반의 전통 데이타베이스와는 다른 모델링 방법과 질의, 검색 방법을 사용한다. 특히, 내용 기반 이미지 검색에서의 검색 속도와 정확도를 향상시키기 위해서는 새로운 복합 질의문 계획 생성 기법이 필요하다. 본 논문에서는 이를 위해 먼저, 단일 조건을 갖는 시각 질의에 대한 처리 기법들을 토대로 여러 조건을 갖는 복합 질의를 처리하기 위한 복합 질의문 계획 생성 기법인 SSCC(Similarity Search for Conjunction Combination Query) 알고리즘을 제안한다. SSCC는 이미지 데이타베이스 검색 시스템에서 복합 질의를 처리하기 위한 질의 최적화 과정에서 질의 수행 시간과 투플 I/O를 최소화하는 질의문 계획을 생성하기 위해 사용된다. SSCC 알고리즘은 복합질의를 단일 질의들로 분해하고 퍼지 집합 이론을 도입하여 단일 질의의 결과들을 통합한다. 논문에서 연구된 내용 기반 복합 질의문 계획 생성 기법은 특정 이미지 영역에 국한되지 않으며 다양한 종류의 시각 질의를 수행하기 위한 효율적인 질의문 계획 생성 기법으로 사용될 수 있다.

Abstract Image database might deal with pictures that have a complicated coloring pattern and contain a number of shapes. So, there are essential differences between Image databases and traditional databases. These differences lead to interesting new issues, in querying, query processing, modeling, and in particular cause us to consider new type of query processing. In this paper, we consider methods of the evaluation of atomic multimedia queries and the evaluation of Boolean combinations of atomic queries. Unlike the situation in relational databases, where the semantics of a Boolean combination is quite clear, in multimedia databases it is not at all clear what the semantics is of even the conjunction of atomic queries. In order to make sure of this notion, we introduce "graded"(of "fuzzy") sets, in which scores are assigned to objects, depending on how well they satisfy atomic queries and complex queries. In this paper we propose a new production method of combination query plan, SSCC(Similarity Search for Conjunction Combination Query) and prove that it performs better than Fagin's A0 algorithm and Surya's Multi-step algorithm in all cases. Our algorithm provides fast retrieval and correct results and requires fewer database accesses.

1. 서 론

이미지 데이타베이스 검색 시스템을 개발하기 위한 요소 기술로는 대용량 비정형 정보를 데이타베이스에 저장하는 저장 관리 기법과, 이미지 정보를 구조적으로 표현하고 관리하는 데이타 모델링 기술, 질의 제시와 결과 표시를 위한 사용자 인터페이스 개발 기술, 그리고 사용자가 원하는 정보에 대한 빠른 접근과 정확한 검색을 지원하기 위한 검색 기술로 나눌 수 있다. 특히 이미지 정보 검색의 경우, 사용자가 원하는 정보를 신속하고 정확하게 추출하고 보여줄 수 있는 질의 처리 기법이 필수적이다[1].

· 본 연구는 동국대학교 논문게재연구비 지원으로 이루어졌음.
† 학생회원 : 동국대학교 컴퓨터공학과
mcehwap@dgu.ac.kr
†† 종신회원 : 동국대학교 컴퓨터공학과 교수
khum@dgu.ac.kr
논문접수 : 1999년 10월 27일
심사완료 : 2000년 9월 27일

이미지 데이타베이스 검색 시스템에서 사용하는 단순 질의에는 텍스트 키워드 질의와 색상 질의, 질감 질의, 모자익 질의, 모양 질의, 위치 질의, 공간 질의를 들 수 있다. 복합 질의의 예로는 예제 이미지 질의와 스케치 질의를 들 수 있다[2][3]. 즉 '사용자가 선택한 예제 그림과 색상과 모양이 비슷한 그림을 검색하라'는 질의는 색상 질의와 모양 질의를 AND 연산으로 결합한 것과 같은 의미를 가진다.

텍스트 키워드 질의를 제외한 단순질의와 복합 질의에 대해 유사한 이미지를 검색하는 내용 기반 검색에서는 기존의 텍스트 데이타베이스 시스템에서 사용하는 완전 일치(exact matching) 검색이 아닌 유사(similarity) 검색을 사용한다. 즉, 이미지 데이타베이스 검색 시스템의 질의 처리 기법은, 완전 일치 검색을 지원하는 관계형 데이타베이스 관리시스템(RDBMS)의 질의 처리 기법과는 다르게 된다. 대부분의 이미지 검색 시스템에서는 유사 검색을 위해 유사도 함수를 사용한다[4][5].

이미지 데이타베이스에서의 질의 처리 문제는 이러한 유사 검색 기법으로부터 파생된다. 완전 일치 검색은 질의 조건에 대해 100% 만족하는 데이타를 검색함으로 질의 결과에 대한 모호성이 존재하지 않는다. 따라서, 여러 개의 질의 조건을 갖는 복합 질의의 처리 결과도 명확성을 갖게 된다. 반면, 질의 조건에 대해 가장 근접한 데이타를 검색하는 유사 검색은 '근접하다'는 의미를 수학적으로 해석함에 있어 모호성이 존재한다. 또한 하나 이상의 질의 조건이 결합된 경우에는 '근접하다'는 단어에 대한 해석 기준이 더 모호해짐을 알 수 있다. 따라서 내용 기반 검색을 위한 질의 처리를 위해서는 먼저, '유사하다'는 단어에 대한 정의를 내린 다음 각각의 단순 질의와 복합 질의를 처리하는 기법에 대한 연구가 수행되어야 한다. 현재 내용 기반 이미지 검색을 위한 단순 질의 처리에 대한 많은 연구가 진행되고 있으며 복합 질의를 처리하기 위한 결과 통합 기법과 복합 인덱싱 기법들이 연구되고 있지만 도자기나 상표, 미술 이미지 등과 같이 특정 영역에 한정된 질의 처리 연구가 주를 이루며 검색 효율도 낮은 실정이다[6][7][8][9].

본 논문에서는 내용 기반 이미지 검색을 위한 최적화된 복합 질의문 계획 생성 기법을 제안한다. 복합 질의문 계획 생성 기법은 이미지 데이타베이스 검색 시스템에서 사용자가 원하는 그림을 찾기 위해 한번에 여러 개의 유사 검색을 수행해야 하는 복합 질의가 주어졌을 때, 이를 처리하기 위한 질의 최적화 과정에서 질의 수행 시간과 투플 I/O를 최소화시킨 질의문 계획을 생성하기 위해 사용된다. 본 논문에서 연구된 내용 기반 복합 질의문 생성 기법은 특정 이미지 영역에 국한되지 않으며 다양한 종류의 시각 질의를 수행하기 위한 효율적인 질의문 계획 생성 기법으로 사용될 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 소개하고 3장에서는 단일 조건을 갖는 단순 질의 처리 절차에 대해 설명한다. 3에서는 복수 조건을 가지는 복합 질의문 계획 생성 알고리즘을 설명한다. 4장에서는 제안한 알고리즘과 기존 방법에 대한 성능 비교 실험과 결과를 분석한다. 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

내용 기반 이미지 검색에서는 이미지의 기본 특징만을 사용하여 사용자의 요구를 정확하게 표현할 수 없으며, 대용량의 이미지 데이타베이스에 적용시킬 경우 검색 효율이 현저하게 감소한다. 이를 개선하기 위해서는 텍스트 기반의 검색 방법을 병행하여 수행해야 하며 검색 성능 향상을 위해 여러 종류의 특징들을 복합적으로 사용할 필요가 있다. 복합 질의를 처리하기 위해서는 단일 질의 조건에 대한 유사도 측정 알고리즘과 여러 조건이 결합된 복합 질의에 대한 유사성 정도를 측정하는 새로운 알고리즘이 필요하다. 복합 질의를 처리하기 위해 대부분의 시스템들은 결합된 여러 특징 조건들을 분리해서 개별적으로 처리한 후 그 결과를 통합하는 방법을 사용하고 있다[6][7][8][9]. 대표적인 예로 Garlic 시스템에서 사용되고 있는 Ronald Fagin이 제안한 A_0 알고리즘을 들 수 있다[6].

A0의 알고리즘은 데이타베이스에 저장되어 있는 객체들의 단순 질의에 대한 유사도를 단순 질의 처리 시스템에서 계산하고 임의(random)의 방법으로 다른 단순 질의에 대한 유사도를 참조할 수 있다고 가정한다. Garlic 시스템에서 사용하는 단순 질의 처리 시스템은 QBIC[10] 시스템이 담당한다. 각 단순 질의 처리 시스템에서 계산된 모든 객체에 대한 유사도는 유사도가 높은 순서대로 정렬되고 퍼지 집합 이론의 Intersection 함수와 Union 함수를 이용하여 각 단순 질의의 결과를 통합한다. A_0 알고리즘은 다음과 같이 정렬접근단계, 임의접근단계, 계산단계의 세 단계로 구성된다 :

1. 정렬 접근 단계(sorted access phase) : 서브시스템에서 각 단순 질의에 대한 유사도를 계산한 후 가장 유사한 T개의 객체를 추출한다. 모든 서브시스템에서 추출된 객체들의 집합을 L이라 할 때, T는 $L=\bigcap_{i=1}^{m} X_i$ 이 적어도 k개의 객체를 갖도록 정의된 상수이다. 여기서 X_i는 단순 질의 i에 대해 가장 유사한 T개의 객체를 추출한 결과 집합을 의미하고 m은 단순

질의의 수를 의미한다. k는 사용자에게 보여줄 결과 이미지의 개수를 의미한다.

2. 임의 접근 단계(random access phase) : L에 있는 모든 객체 x에 대해 다른 질의 시스템에 대한 유사도를 계산한다. 이 때 각 서브시스템에서 x에 대한 유사도를 임의적으로 가져올 수 있다고 가정한다.

3. 계산 단계(computation phase) : L에 포함된 모든 객체 x에 대해 질의 Q에 대한 유사도를 계산하고 가장 유사도가 높은 k개의 객체를 추출한 것을 결과 집합 Y라 정의한다. 질의 Q에 대한 유사도 $\mu Q(x)$는 $\mu Q(x) = t(\mu A1(x), ..., \mu A_m(x))$로 정의된다. 여기서 결과 통합 함수 t는 min 또는 max 함수를 이용한다. $\mu A_i(x)$는 단순 질의 A_i에 대한 객체 x의 유사도 점수를 의미한다.

A_0 알고리즘은 데이타베이스에 저장된 모든 데이타를 대상으로 각 단순 질의에 대한 유사도를 구한 후 그 결과를 통합함으로써 사용자 질의에 대해 정확하게 k개의 이미지를 검색할 수 있다는 장점이 있다. 또한 각 단순 질의 처리 시스템들을 병렬 처리함으로써 순차 수행보다 처리 시간을 단축시킬 수 있다. 그러나 인덱스가 없을 경우, 데이타베이스에 저장된 모든 이미지에 대해 각 단순 질의에 대한 유사도를 구해야 하므로 복합 질의를 처리하기 위해 데이타베이스에 접근하는 회수, 즉 투플 I/O는 총 투플 개수 * 단순 질의 개수가 된다. 즉, 색상과 질감, 모양 질의가 결합된 복합 질의라면 모든 이미지에 대해 색상 질의에 대한 유사도, 질감 질의에 대한 유사도, 모양 질의에 대한 유사도를 계산하기 위해 색상 테이블과 질감 테이블, 모양 테이블을 접근하게 된다. 또한 모든 데이타들을 유사도가 높은 순서대로 정렬한 다음 상위 k개의 결과들을 추출해서 결과 집합을 생성함으로 중간 결과 집합의 크기도 증가한다. 모든 이미지 특징에 대해 인덱스가 존재한다면 투플 I/O는 T*단순 질의개수로 감소하지만 인덱스 구축과 유지비용이 많이 들며, 정렬 접근 단계에서 각 단순 질의의 결과 집합들에 대한 교집합의 개수가 k가 될 때까지 반복적으로 k개의 이미지를 검색하므로 인덱스 효율이 떨어진다.

다른 복합 질의 처리 알고리즘으로 A_0의 알고리즘을 개선한 Multi-step 알고리즘이 있다[9]. Multi-step 알고리즘은 A_0 알고리즘에서 정렬 접근 단계의 시간이 많이 걸린다는 단점을 개선한 것이다. 이 알고리즘은 다음의 여섯 단계로 구성된다.

1. 각 서브시스템에서 단순 질의에 대한 유사도를 계산한 후 가장 유사한 객체 하나를 추출한다. 이때 각 서브시스템 i의 결과는 객체 x와 질의에 대한 x의 유사도의 쌍 $(x, \mu A_i(x))$이다.

2. 각 서브시스템 i에서 추출된 객체 x에 대해 다른 질의 시스템 j에 대한 유사도를 계산한다. 즉, $i \neq j$, $\mu A_j(x)$를 계산한다.

3. 각 서브시스템에서 계산한 객체 유사도에 대한 임계값 $th = t(\mu A_1(x_1), ..., \mu A_m(x_m))$를 계산한다.

4. 각 서브시스템에서 추출된 객체들에 대해 $\mu Q(x) = t(\mu A_1(x), ..., \mu A_m(x))$를 계산해서 $\mu Q(x) \geq th$인 객체를 결과 집합 Y에 추가한다.

5. 집합 Y가 k 개의 원소를 가질 때까지 1단계에서 5단계를 반복한다.

6. 결과 집합 Y를 반환한다.

Multi-step 알고리즘은 정렬 접근 단계의 비용을 줄이기 위해 각 서브시스템에서 가장 유사한 한 개의 객체를 구하고 추출된 객체들의 유사도 점수를 이용하여 임계값을 계산한다. 최종 결과 집합은 유사도 점수가 임계값보다 높은 k개의 객체들로 이루어진 집합이다. 이 알고리즘은 두 가지 측면에서 A_0 알고리즘과 다른 특징을 갖는다.

첫째, A_0 알고리즘에서는 각 서브시스템별로 T개의 결과를 추출해서 유사도가 높은 순서대로 정렬하지만 Multi-step 알고리즘은 각 서브시스템별로 1개의 객체를 추출한다. 따라서 결과 집합 Y를 구하기 위해 A_0 알고리즘에서 한번에 T개씩 R번의 반복 접근이 이루어지는 반면, 1번에 1개씩의 데이타를 접근하는 Multi-step 알고리즘은 분명히 R*T보다 작은 데이타베이스 접근 수를 가짐을 알 수 있다.

둘째, A_0 알고리즘에서는 최종 결과를 산출하기 위해 각 서브시스템에서 추출한 결과 집합들에 대한 교집합을 구하고 이 집합에 속한 객체들에 대해 최종 유사도 $\mu Q(x) = t(\mu A_1(x), ..., \mu A_m(x))$를 계산해서 유사도가 높은 k개의 객체들을 구하는 반면, Multi-step 알고리즘에서는 각 서브시스템에서 추출한 객체에 대해 $\mu Q(x)=t(\mu A_1(x), ..., \mu A_m(x))$가 $th=t(\mu A_1(x_1), ..., \mu A_m(x_m))$ 보다 크거나 같은 객체를 최종 결과 집합 Y에

표 1 A0 알고리즘과 Multi-step 알고리즘 비교

	A_0 알고리즘	Multi-step 알고리즘
데이타베이스 접근 수	한번에 T개씩 R번의 반복 접근	한번에 1개씩 R*T보다 작게 접근
최종 결과 집합	각 단순질의 결과 집합들의 교집합	임계값 $th = t(\mu A_1(x_1), ..., \mu A_m(x_m))$를 넘는 단순 질의 결과 집합들의 교집합
인덱스 효율	T개씩 가져오므로 인덱스 사용 효과 감소	1개씩 가져오므로 인덱스 사용 효과 증가

포함시킨다. 이 알고리즘은 Y가 k개의 원소를 가질 때까지 반복 수행된다.

[표 1]은 두 알고리즘의 특징을 정리한 것이다. 지금까지 살펴 본 바와 같이 Multi-step 알고리즘은 정렬 접근 단계에서의 데이타베이스 접근 비용을 감소시킬 수 있었지만 여전히 각 서브시스템에서 추출된 모든 객체에 대해 임의접근 단계를 수행하고 있으므로 계산 비용이 많이 들게 된다.

본 논문에서는 위에서 기술한 시스템들에서 사용하는 결과 통합 방법을 사용하지 않고, 하나의 단순 질의를 수행한 결과 집합의 최소 유사도를 계산하고 이를 모든 단순 질의 결과 집합의 최소 유사도로 추정한 다음 최소 추정 유사도를 기준으로 나머지 단순 질의들을 순차적으로 처리한다. 이로 인해, 모든 단순 질의를 계산한 다음 결과를 통합하는 방법들에서 발생하는 투플 I/O와 검색 시간을 줄일 수 있다.

3. 단순 질의 처리 방법

본 논문에서는 복합 질의문 계획을 수행하는데 사용하기 위한 단순 질의 형태로 텍스트 키워드 질의와 색상, 질감, 모자익, 위치, 공간 질의를 사용한다. 이 장에서는 각각의 단순 질의 처리 방법에 대해 설명한다.

3.1 텍스트 키워드 질의 처리 방법

텍스트 키워드 질의는 데이타베이스에 저장된 이미지의 의미정보를 대상으로 검색을 수행하므로 완전 일치 검색 방법을 이용한다. 텍스트 키워드 질의의 질의 결과는 조건을 만족하는 이미지들의 집합이 된다. 따라서 텍스트 키워드 질의는 기본 특징 질의들과 함께 사용될 경우 조사 범위를 줄일 수 있는 필터로 사용될 수 있다[14].

3.2 색상 질의 처리 방법

본 논문에서는 이미지에 대한 색상 정보로서 이미지에 대한 대표 색상과 색상 히스토그램, 모자익 정보를 사용한다. 색상 정보는 먼저 각 픽셀(pixel)에 대한 RGB 색상 정보를 추출한 다음 사람의 시각 능력에 유사한 색상 모델인 HSV 색상으로 변환하여 사용한다. 대표 색상은 색상 히스토그램을 구성하는 색상 막대(color bin) 중에서 가장 많은 픽셀 수를 가지는 색상을 사용한다. 모자익 정보는 이미지를 64개의 블록으로 나눈 후 각 블록에 대한 대표 색상을 추출한 정보이다.

이미지의 색상 정보는 해상도에 따라 다양한 색상 계수를 가지므로 모든 색상 계수를 다 고려하는 것은 효율적인 방법이 아니다. 그래서 본 논문에서는 색상 계수를 빨강, 노랑, 녹색, 청녹색, 파랑, 자홍색의 여섯가지 색도와 채도와 명도를 각기 3가지 값으로 분류한 54개의 색상 계수에 흑백 색상 4가지를 더한 58가지 색상 계수로 정량화한다. 또한 이미지의 크기가 달라지면 픽셀 수가 달라지고 색상 히스토그램의 크기 값 또한 달라지므로 모든 이미지에 대한 색상 히스토그램의 채널 값의 합이 1이 되도록 표준화한다[14].

색상 비교 함수는 데이타베이스에 저장된 이미지 데이타의 색상 정보 $T(h_t, s_t, v_t)$와 질의 색상 $Q(h_q, s_q, v_q)$에 대한 유사도를 계산하고 유사도가 높은 순서대로 k개의 이미지 ID를 반환한다. 유사도는 [0, 1]사이의 값을 갖도록 표준화한다.

3.3 질감 질의 처리 방법

본 논문에서 사용된 질감 정보는 각 질감 이미지의 특징을 모멘트(moment)[11]를 이용하여 7가지 계수로 나타낸다. 이들 모멘트의 집합은 이동, 회전, 확대, 축소에 변화가 없이 일정하다는 특징을 가진다. 모멘트 비교 함수는 유클리드 거리 함수를 이용해서 유사도를 계산하고 유사도가 높은 순서대로 k개의 이미지 ID를 반환한다[14].

3.4 위치 질의 처리 방법

이미지 내 객체의 위치 정보는 절대 위치 정보라고 말할 수 있으며 이미지를 데이타베이스에 저장할 때 추출한 객체 MBR의 중심 좌표이다. 위치 질의를 처리하는 함수는 질의 인터페이스에서 주어진 객체의 중심 좌표 정보와 데이타베이스에 저장된 객체들의 중심 좌표 사이의 거리를 측정하고 가장 가까운 객체를 포함하는 k개의 이미지 ID를 반환한다.

3.5 공간 질의 처리 방법

공간 정보는 이미지 내 객체들의 위치 관계 정보이다. 공간 정보는 각 객체의 중심 좌표를 이용해 객체간의 거리 정보를 추출하고 그 사이각을 이용해 방향 정보를 추출한 다음 객체들간의 포함, 겹침, 근접, 등의 공간 정보를 추출한다[14]. 질의에 사용된 공간 정보는 질의 인터페이스 상에서 공간 정보 추출 기법을 이용해서 추출되며 추출된 질의 정보는 공간 관계 테이블을 대상으로 검색을 수행하는 데 이용된다. 따라서 공간 질의는 텍스트 키워드 질의 처리 기법과 동일한 방법을 사용한다.

4. 복합 질의문 계획 생성 알고리즘

단순 질의는 유사도 계산 함수를 이용하여 유사도를 계산하고 유사도 점수가 높은 객체들을 순서대로 추출하거나 인덱스를 구축하여 가장 유사한 객체를 검색한다. 그러나 (color = 'yellow')∧(shape = 'round')와 같이 여러 개의 단순 질의들이 논리 연산자(∧, ∨, ¬)

로 결합된 복합 질의는 단순 질의와는 다른 질의 처리 알고리즘이 필요하다. 본 절에서는 퍼지 집합 이론을 이용하여 이미지 데이타베이스 검색 시스템에서 발생하는 복합 질의를 수행하기 위한 복합 질의문 계획 생성 알고리즘을 설명한다.

이미지에 대한 시각 질의는 유사도 함수를 이용하여 질의 조건에 대한 유사도를 [0, 1]사이의 수로 표현하고 유사도가 가장 높은 이미지를 검색함으로써 모호한 집합에 대한 연산을 수행하는 퍼지 집합 이론을 이용하여 수학적으로 해석할 수 있다. 데이타베이스에 있는 모든 데이타로 구성된 전체 집합 U에 대해 단순 질의 A, B에 대한 퍼지 집합은 $A = \{x \mid x \in U, (x, \mu C(x))\}$, $B = \{x \mid x \in U, (x, \mu S(x))\}$로 정의된다. 여기서 x는 데이타를 의미하고, $\mu C(x)$, $\mu S(x)$는 질의 C, S에 대한 유사도 점수이다. 검색할 이미지의 수를 k라고 하고 복합 질의 $A \cap B$에 대한 결과 집합은 $\mu C \wedge S(x)$가 가장 높은 k개의 원소를 가지는 U의 부분 집합 T가 된다.

본 논문에서는 질의 최적화에 응용하기 위해 퍼지 집합 이론인 Bellman-Giertz 이론[12]을 이용하였다. 이 이론에 의해, 질의 Q_1, Q_2가 교집합 연산과 합집합 연산에만 적용되고 논리적으로 동등한 질의이면 모든 객체 x에 대해 $\mu Q_1(x) = \mu Q_2(x)$임이 증명되었다. 즉, $\mu A \wedge A(x) = \mu A(x)$, $\mu A \wedge (B \vee C)(x) = \mu(A \wedge B) \vee (A \wedge C)(x)$이 성립한다. 이를 질의 최적화에 이용하면 논리적으로 동등하면서 같은 결과를 가지는 비용이 적게 드는 질의로 대치할 수 있다. Bellman-Giertz는 또한 교집합에 대한 집합 함수 min과 합집합에 대한 집합 함수 max가 단조성을 가짐을 보였다. 즉, $\mu A(x) \leq \mu A(x')$ 이고 $\mu B(x) \leq \mu B(x')$이면 $\mu A \wedge B(x) \leq \mu A \wedge B(x')$이고 $\mu A \vee B(x) \leq \mu A \vee B(x')$이다.

4.1 텍스트 질의와 시각 질의가 결합된 복합 질의문 계획 생성

텍스트 질의와 시각 질의가 결합된 질의의 예를 들면 다음과 같다.

[예1] : (name = '사과' AND color = 'red')

[예2] : (name = '사과' OR color = 'red')

텍스트 질의는 처리비용이 적고 구현이 용이하지만, 시각 질의는 유사도 함수 구현이 어렵고 계산 비용이 비싸다는 단점을 가진다. 따라서 질의 [예1]의 경우와 같이 텍스트 질의와 시각 질의가 AND 연산자로 결합된 경우는 텍스트 질의를 먼저 수행한 결과 집합에 대해 시각 질의에 대한 유사도를 계산하면 계산 비용을 줄일 수 있다. 질의 [예2]의 경우는 텍스트 질의와 시각 질의들이 OR 연산자로 결합되어 있다. OR 연산자의 특성상 질의 [예2]의 결과는 텍스트 질의만을 만족하거나 시각 질의만을 만족할 수도 있다.

4.2 논리곱 복합 질의문 계획 생성 알고리즘

동질 연산자 복합 질의는 여러 개의 단순 질의가 한 종류의 연산자로 결합된 질의를 말한다. 즉 단순 질의 $A_1,...,A_m$이 AND 연산자로 결합된 m-차원 논리곱 질의 $F(A_1 \wedge ... \wedge A_m)$ 또는 단순 질의 $A_1,...,A_m$이 OR 연산자로 결합된 형태 m-차원 논리합 질의 $F(A_1 \vee ... \vee A_m)$를 말한다. 이 절에서는 동질 연산자 복합 질의를 처리하는 알고리즘을 제안하고 이 알고리즘이 정확하게 k개의 유사 객체를 찾음을 증명한다.

논리곱 복합 질의(Conjunction Combination Query) $Ft(A_1 \wedge ... \wedge A_m)$은 단순 질의 $A_1,...,A_m$을 모두 만족하는 객체 x를 찾는 것이다. 다시 말해서 모든 질의 조건을 만족하는 객체를 찾아 만족도가 높은 순서대로 상위 k개를 찾아오는 것이다. 각 단순 질의 A_i에 대해 이를 만족하는 상위 유사 집합을 X^i라 정의하면 결과 객체는 적어도 k개의 원소를 포함하는 집합 $\bigcap_{i=1}^{m} X^i$에 속한다는 것을 알 수 있다.

제안하는 알고리즘의 기본 개념은 질의 결과 집합에 대한 최소 유사도를 유추해서 각 단순 질의에 대해 최소 유사도를 넘는 후보 객체들만을 선출한 후 모든 단순 질의를 만족하는 객체들을 최종적으로 추출한다는 것이다. 즉, 하나의 단순 질의 A_i를 먼저 수행한 후 X^i에 속하는 객체들의 최소 유사도를 α라 정의하고 나머지 단순 질의들에 대해 α를 넘는 유사도를 가지는 객체들을 추출하여 추출된 원자들의 교집합을 구하는 것이다. 직관적으로 단조 함수 t에 대해 $x \in \bigcap_{i=1}^{m} X^i$, $\mu Q(x)$가 최소인 x의 유사도를 α라 정의하면 $y \in \bigcap_{i=1}^{m} X^i$인 y에 대해 $\mu Q(y) \geq \mu Q(x) = \alpha$임을 알 수 있다. [정리 1]은 두 객체 x, z에 대해, x가 최종 결과 집합에 속하며 x의 유사도가 최종 결과 집합의 최소 유사도이고, 임의의 객체 z의 유사도가 x의 유사도보다 작으면, z는 모든 단순질의들을 만족하지 않음을 증명한다. 즉, $z \notin \bigcap_{i=1}^{m} X^i$이고 $z \notin \bigcup_{i=1}^{m} X^i$이다.

[정리 1] 두 객체 $x \in \bigcap_{i=1}^{m} X^i$, 임의의 객체 z에 대해 $\mu Ft(A1...Am)(x) = \alpha$이고 $\mu Ft(A1...Am)(x) > \mu Ft(A1...Am)(z)$이면 $z \notin \bigcap_{i=1}^{m} X^i$이고 $z \notin \bigcup_{i=1}^{m} X^i$이다.

[증명] 이제부터는 질의 $Ft(A_1...A_m)$를 간단히 Q로 표기한다.

i) Fagin의 정리 4.1[6]에 의해 두 객체 $x \in \bigcap_{i=1}^{m} X^i$와 z에 대해 $\mu Q(z) > \mu Q(x)$이면 $z \in \bigcup_{i=1}^{m} X^i$이라는 것이 증명되었다. 만일 $z \in \bigcup_{i=1}^{m} X^i$라고 한다면 $\mu Q(z) >$

$\mu Q(x)$이어야 한다. 이는 가정에 위배되므로 $z \notin \bigcup_{i=1}^{m} X^i$이다.

ii) 만일 $z \in \bigcap_{i=1}^{m} X^i$이라면 $\mu Q(z)$는 $\bigcap_{i=1}^{n} X^i$의 최소 유사도인 α보다 크거나 같은 값이어야 한다. 가정에서 $\alpha = \mu Q(x)$이므로 $\mu Q(z) \geq \mu Q(x)$가 성립된다. 이는 가정에 위배되므로 $z \notin \bigcap_{i=1}^{m} X^i$이다. □

다음은 논리곱 복합 질의를 처리하는 검색 알고리즘 SSCC(Similarity Search for Conjunction Combination query)를 기술한 것이다. 질의 $Ft(A_1 \wedge ... \wedge A_m)$에 대해 SSCC 알고리즘은 다음의 각 단계로 기술된다.

[SSCC 알고리즘]

[단계1] 데이타베이스에 저장된 객체 x에 대해 $\mu A_i(x)$를 계산하고 유사도가 높은 순서대로 정렬하여 상위 k개의 유사도를 가지는 객체들을 원소로 갖는 X^1를 생성한다. 모든 $y \in X^1$에 대해 최소 유사도 $\alpha = \mu A_i(y)$를 구한다.

[단계2] $\forall x \in X^1$인 x에 대해 $A_i(1<i\leq m)$의 유사도 $\mu A_i(x)$를 계산하고 $\mu A_i(x) \geq \alpha$를 만족하는 원소들의 집합 $T=\{x \mid x \in X^1, ((\alpha \leq \mu A_2(x)) \wedge \cdots \wedge (\alpha \leq \mu A_m(x))\}$를 생성하고 $n(T) \geq k$가 될 때까지 단계2를 반복 수행한다.

[단계3] $\forall x \in T$, $z_i = \mu A_i(x)$라 하고 가중치를 $w_1, \cdots, w_m (w_1+w_2+\cdots+w_m=1)$라 하면 질의 Q의 결과 집합에 속하는 객체 x의 유사도 $\mu Q(x) = \sum_{i=1}^{m} w_i z_i / m$를 계산하고 유사도가 높은 순서대로 k개를 추출한다.

SSCC 알고리즘이 정확하게 질의문을 만족하는 k개의 객체로 구성된 결과를 구할 수 있다는 것을 다음 [정리 2]로 알 수 있다.

[정리 2] 모든 단조(monotone) 질의에 대해 SSCC 알고리즘이 정확하게 상위 k개의 객체를 산출한다.

[증명]

i) 단조 질의를 Q, 총 객체 수를 N이라 하자. $A^i{}_\alpha = \{x \in U \mid \mu Ai(x) \geq \alpha\}$, $\alpha \in (0, 1]$로 정의하면 $\bigcup_{\alpha \in (0,1]} (\bigcap_{i=1}^{m} A^i{}_\alpha)$는 모든 객체를 포함한다. $k \leq N$이므로 $\bigcup_{\alpha \in (0,1]} (\bigcap_{i=1}^{m} A^i{}_\alpha)$는 적어도 k개의 객체를 포함함을 알 수 있다. 또한 $\bigcap_{i=1}^{m} A^i{}_\alpha$가 적어도 k개의 객체를 갖도록 알고리즘에서 α가 잘 정의되었으므로 정의에서 집합 T는 적어도 k개의 원소를 가진다.

ii) 앞서 "상위 k개"에 대한 정의를 $\mu A_i(x) \geq \alpha$인 객체들의 집합 $A^i{}_\alpha$에 대해 집합 $T=\{x \mid x \in \bigcap_{i=1}^{m} A^i{}_\alpha\}$에 속하는 원소로 정의하였다. 여기서 우리는 집합 T에 속하는 원소들이 정확하게 유사도가 높은 상위 k개임을 증명하면 된다. 즉, 집합 T에 속하는 객체 y와 T에 속하지 않으면서 동시에 $\bigcup_{i=1}^{m} A^i$에도 속하지 않는 객체 z에 대해 $\mu Q(y) > \mu Q(z)$임을 보이면 된다. 이는 [정리 1]에 의해 입증되었다. 즉, $z \notin T$ and $z \notin \bigcup_{i=1}^{m} A^i$, $y \in T$인 z, y에 대해 $\mu Q(y) > \mu Q(z)$를 알 수 있다.

i)과 ii)에 의해 SSCC 알고리즘이 논리곱 복합 질의에 대해 정확하게 상위 k개의 객체를 추출함을 알 수 있다. □

4.3 논리합 복합 질의문 계획 생성 알고리즘

논리합 복합 질의(Disjunction Combination Query) $Ft(A_1 \vee ... \vee A_m)$은 단순 질의 $A_1 ... A_m$에 대해 하나 이상의 단순 질의를 만족하는 객체 x를 찾는 것이다. 따라서 논리합 질의를 처리하는 알고리즘은 논리곱 질의를 처리하는 알고리즘에 비해 간단한 형태를 갖게 된다. 논리곱 질의와는 달리 각 단순 질의 A_i를 만족하는 상위 유사 집합을 X^i라 정의하면 우리가 찾는 객체는 적어도 k개의 원소를 포함하는 집합 $\bigcup_{i=1}^{m} A^i$에 속한다는 것을 알 수 있다. 질의 $Ft(A_1 \vee ... \vee A_m)$에 대해 SSDC (Similarity Search for Disjunction Combination query) 알고리즘은 다음의 단계로 기술된다.

[SSDC 알고리즘]

[단계1] 모든 단순 질의 $A_1, ..., A_m$에 대해 $\mu A_i(x)$를 병렬로 계산한 다음 각 단순 질의에 대해 상위 k개의 유사도를 가지는 객체들을 원소로 갖는 Xi를 생성한다.

[단계2] 집합 $T=\{x \mid x \in \bigcup_{i=1}^{m} X^i\}$를 생성하고 $\forall x \in T$, $\mu A_1(x)$, ..., $\mu A_m(x)$를 계산한다.

[단계3] $\forall x \in T$, $z_i = \mu A_i(x)$라 하고 가중치를 $w_1, \cdots, w_m (w_1+w_2+\cdots+w_m=1)$라 하면 질의 Q의 결과 집합에 속하는 객체 x의 유사도 $\mu Q(x) = \sum_{i=1}^{m} w_i z_i / m$를 계산하고 유사도가 높은 순서대로 k개를 추출한다.

SSDC 알고리즘이 정확하게 질의문을 만족하는 k개의 객체로 구성된 결과를 구할 수 있다는 것을 다음 [정리3]으로 알 수 있다.

[정리 3] 모든 단조 질의에 대해 SSDC 알고리즘이 정확하게 상위 k개의 객체를 산출한다.

[증명] 단조 질의를 Q, 총 객체 수를 N, X^i가 N개의 객체를 갖는 유사도 집합이라고 하면, $\bigcup_{i=1}^{m} X^i$는 모든 객체를 포함한다. 또한 $k \leq N$이므로 $\bigcup_{i=1}^{m} X^i$는 적어도 k개의 객체를 포함함을 알 수 있다. 또한 X^i가 적어도 k개의 객체를 갖도록 정의되었으므로 정의에서 집합 T는 질의 Q를 만족하는 적어도 k개의 원소를 가진다.

5. 실험 및 분석

이 장에서는 본 논문에서 제안한 복합 질의문 계획 생성 알고리즘과 A_0 알고리즘, Multi-step 알고리즘에 대한 성능 평가를 위해 투플I/O와 중간 결과의 크기, 검색 정확도와 재현율 측면에서 비교, 분석한다. 이를 위해 내용 기반 이미지 검색을 제공하는 프로토타입 시스템을 구축하였다. 구축된 시스템은 이미지에서 색상 히스토그램과 모자익, 질감, 대표 색상을 추출하기 위한 특징 추출 모듈과 예제 이미지 질의를 처리하기 위한 질의 처리 모듈, A_0 알고리즘을 구현한 모듈, Multi-step 알고리즘을 구현한 모듈, 본 논문에서 제안한 SSCC 알고리즘을 구현한 모듈로 구성된다. 이 시스템은 Windows NT 운영체제에서 VisualC++6.0을 이용하여 구현하였다. 실험용으로 사용된 이미지는 Corel Photo CD에 있는 이미지들 중 1000여 개의 이미지를 이용하였다.

실험에 이용된 A_0 알고리즘과 Multi-step 알고리즘은 각각의 단순 질의 처리 프로시져를 병렬로 처리하도록 작성하였다. 세 알고리즘은 동일한 환경에서 작성되었으며 동일한 실험 데이타를 대상으로 실험하였다.

예제 이미지 질의를 복합 질의로 사용하였으며 복합 질의를 이루는 원자 질의는 대표 색상 질의, 색상 히스토그램 질의, 모자익 질의, 질감 질의의 4가지를 사용하였다. 각 알고리즘에 대해 검색 결과의 수, k를 증가시키면서 정확도와 재현율을 비교하였으며 이미지 질의 수를 증가시키면서 10개의 이미지 질의에 대한 투플I/O, 정확도, 재현율을 비교하였다.

5.1 이미지 검색 프로토타입 시스템

본 논문에서 구현한 이미지 검색 시스템은 이중 그래프 데이타 모델[14]을 기반으로 구축되었다. 이 시스템은 복합 질의 처리 알고리즘에 대한 성능 분석을 위해 작성된 화일 기반 프로토타입 시스템으로서 내용 기반 이미지 데이타베이스 검색 시스템의 특징 추출 모듈과 질의 처리 모듈에 사용될 수 있다.

사용된 이미지는 이미지 고유의 특징 정보를 정확하게 추출하기 위해 원래의 크기를 유지하였다. 질의에 사용된 예제 이미지는 이미지 특징 추출에 사용된 이미지 중 하나를 사용자가 선택할 수 있도록 처리하였다. 질의 결과는 결과 집합 개수 k에 따라 k개의 가장 유사한 이미지로 이루어진 집합을 반환하며 유사도 점수가 높은 순서대로 정렬되어 있다.

5.2 세 알고리즘의 성능 비교

세 알고리즘의 성능 비교를 위해 4가지 단순 질의들이 AND 연산자로 결합된 형태를 복합질의로 사용하였다. 결과 분석은 먼저 검색 인수 k를 10으로 고정한 후 질의 개수를 증가시키면서 평균 정확도와 재현율, 투플 I/O, 중간결과 크기와 인덱스 효율을 비교하였다. 그런 다음 검색 인수 k를 1부터 10까지 증가시키면서 10개의 질의에 대한 세 알고리즘의 평균 정확도와 평균 재현율의 변화를 측정하였다.

[그림 1]은 예제 이미지 질의의 수를 증가시키면서 측정한 세 알고리즘에 대한 평균 정확도를 나타내며 [그림 2]는 평균 재현율을 나타내고 있다.

정확도 비교에서 보면 SSCC 알고리즘이 평균 정확도가 0.46이고 A_0 알고리즘의 평균 정확도가 0.32, Multi-step 알고리즘의 평균 정확도가 0.38로 SSCC 알고리즘이 가장 높음을 알 수 있다. A_0 알고리즘이 각 단순 질의 처리 시스템에서 추출한 객체들에 대한 교집합에서 유사도가 높은 k개의 객체들을 추출하는 단순 비교를 수행한 반면 SSCC 알고리즘은 질의 처리 정확도가 높은 단순 질의를 먼저 수행한 다음 그 결과 집합에 대해 다른 질의 처리 시스템에서의 유사도를 구해 임계값 이상인 객체들을 추출함으로써 정확도를 증가시킬 수 있었다. 또한 Multi-step 알고리즘은 각 단순 질의 처리 시스템에서 가장 유사한 객체 하나씩을 추출한 다음, 이들 중 유사도가 가장 낮은 객체의 유사도를 임계값으로 사용해서 임계값 이상인 객체들을 추출함으로써 모든 단순 질의에 대해 질의 조건을 만족하는 결과를 찾기보다는 하나의 단순 질의를 만족하는 결과를 찾게 되는 경우가 많게 된다. 따라서 검색 결과 정확도가 SSCC 알고리즘에 비해 낮아지게 된다.

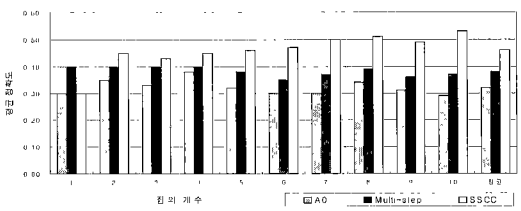

그림 1 질의 개수 증가에 따른 평균 정확도 측정(k=10)

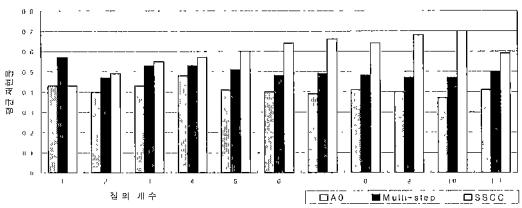

그림 2 질의 개수 증가에 따른 평균 재현율 측정(k=10)

재현율에 있어서도 SSCC 알고리즘이 평균 재현율 0.59, Multi-step 알고리즘이 0.5, A_0 알고리즘이 0.41로 SSCC 알고리즘이 가장 높음을 알 수 있다. 이는 SSCC 알고리즘이 질의 처리 정확도가 높은 단순 질의 처리 시스템을 먼저 수행한 다음 그 결과 집합의 최소 유사도 α를 임계값으로 사용함으로 인해 최종 결과 집합의 유사도가 나머지 두 알고리즘의 유사도보다 월등히 높아진 때문이다.

[표 2]와 [표 3]은 인덱스를 사용하지 않았을 경우와 각 단순 질의에 대한 인덱스를 사용하였을 경우, 세 알고리즘에 대한 투플 I/O, 중간 결과 크기에 대한 비교표이다. 표에서 N은 데이타베이스에 저장된 총 이미지 개수를 의미하고, M은 사용된 단순 질의의 수이다. P는 A_0 알고리즘의 반복회수이며 Q는 Multi-step 알고리즘의 반복회수, L은 SSCC 알고리즘의 반복회수이다. 여기서 P는 단순 질의당 t개씩 읽어왔을 경우의 반복회수, Q는 각 단순 질의당 1개씩 읽어왔을 경우의 반복회수, L은 첫째 단순 질의에 대해 k개를 읽어왔을 경우의 반복회수이므로 P와 Q에는 P=Q/t, L과 Q에는 L=Q/k라는 관계가 성립한다. 이때 각 이미지에 대한 특징 정보는 각 정보별로 테이블에 저장된다고 가정한다.

인덱스가 없을 경우, 투플 I/O에 있어서도 A_0 알고리즘과 Multi-step 알고리즘은 정렬 접근 단계에서 각 서브시스템 별로 데이타베이스에 저장된 모든 객체에 대해 단순 질의에 대한 유사도를 계산하기 때문에 M개의 단순 질의에 대해 N번의 투플 I/O가 발생한다. 반면, SSCC 알고리즘은 먼저 한 개의 단순 질의 처리 시스템에서 N개의 데이타에 대한 유사도 계산을 수행한 다음 첫째 결과 집합에 대해 둘째 단순 질의를 수행하고 그 결과 집합을 다음 단순 질의 수행에 이용하므로 한 번 수행 시 k개의 결과를 추출하고, L 번의 반복 수행이 이루어진다고 하면 $N+L(k_1+\ldots+k_{(M-1)})$개의 투플 I/O가 발생한다. 여기서 k_1은 k과 동일하다.

인덱스가 있을 경우, A0 알고리즘은 인덱스를 이용해 각 단순질의에 대해 t개씩만을 가져오는 과정을 P번 반복한다. Multi-step 알고리즘은 각 단순 질의에 대해 중간 결과 한 개씩만을 가져오므로 Q*M번의 투플 I/O가 발생한다. 반면, SSCC 알고리즘은 인덱스를 이용해 한 개의 단순 질의 처리 시스템에서 k_1개의 결과 데이타를 가져온 다음, 둘째 단순 질의에 대한 유사도를 계산하고, 그 결과 집합을 다음 단순 질의 수행에 이용한다. 여기서 첫째 결과 집합은 k_1, 둘째 결과집합은 k_2, M번째 결과 집합은 $k_{(m-1)}$ 이다.

세 알고리즘의 투플 I/O수 비교를 위해, P=Q/t, L=Q/k를 [표 3]에 있는 세 식에 대입하면 A_0 알고리즘의 투플 I/O수 = Q*M, Multi-step 알고리즘의 투플 I/O도 Q*M, SSCC 알고리즘의 투플 I/O수는 k_1, ... , $k_{(M-1)}$ 이 각각 k보다 같거나 작으므로 $(k_1+\ldots+k_{(M-1)}) <= M*k$ 의 관계가 성립하므로 $Q(k_1+\ldots+k_{(M-1)})/k <= Q*M$임을 알 수 있다. 즉, 최악의 경우에는 A_0알고리즘과 Multi-step 알고리즘과 같은 투플 I/O를 가지지만, 그렇지 않을 경우 두 알고리즘보다 작은 투플 I/O를 갖는다.

둘째 실험은 검색 인수 k를 1부터 10까지 증가시키면서 10개의 질의에 대한 세 알고리즘의 평균 정확도와 평균 재현율의 변화를 측정하였다. [그림 3]은 10개의 예제 이미지 질의에 대해 검색 인수 k의 수를 증가시키면서 측정한 세 알고리즘에 대한 평균 정확도를 나타내며 [그림 4]는 평균 재현율을 나타내고 있다.

[그림 3]을 보면 k가 1일때, 즉, 가장 유사한 1개의 이미지를 검색할 경우 세 알고리즘의 정확도는 1에 가깝다. 이는 질의 이미지를 데이타베이스에 저장된 이미지들 중 하나를 사용했기 때문이다. k가 증가하면서 세 알고리즘 모두 정확도가 감소하고 있다. 이는 질의 이미

표 2 인덱스가 없을 경우의 투플I/O, 중간결과크기 비교

	A0	Multi-step	SSCC
투플 I/O	M*N	M*N	$N+L(k_1+\ldots+k_{(M-1)})$
중간 결과 크기	M*N+T	M*N+Q*M+k	$N+L(k_1+\ldots+k_{(M-1)})+k$

표 3 인덱스가 있을 경우의 투플I/O, 중간결과크기 비교

	A0	Multi-step	SSCC
투플 I/O	P(M*t)	Q*M	$L(k_1+\ldots+k_{(M-1)})$
중간 결과 크기	P(M*t)+T	Q*M+k	$L(k_1+\ldots+k_{(M-1)})+k$

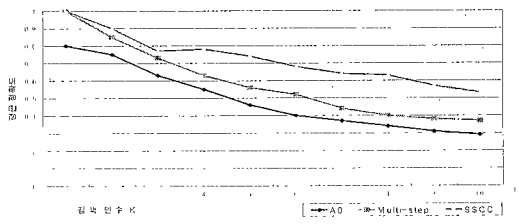

그림 3 k의 변화에 따른 10개의 이미지 질의에 대한 평균 정확도

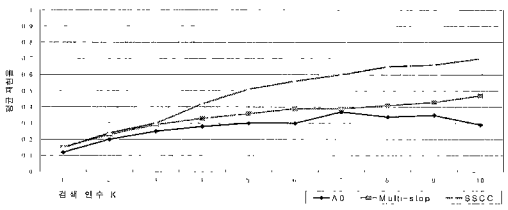

그림 4 k의 변화에 따른 10개의 이미지 질의에 대한 평균 재현율

지와 유사한 이미지의 수가 한정되어 있기 때문이다. 이 비교에서도 SSCC 알고리즘의 평균 정확도가 두 알고리즘에 비해 높음을 알 수 있다.

재현율의 변화를 나타내는 [그림 4]에서도 SSCC 알고리즘의 평균 재현율이 가장 높음을 알 수 있다. 이는 질의 개수를 증가시키면서 측정한 평균 재현율이 나머지 두 알고리즘보다 높았던 것과 동일한 이유 때문이다. 평균 재현율은 세 알고리즘 모두 k가 1일 때 가장 낮으며 k가 증가할수록 증가한다. 이는 질의 이미지와 유사한 이미지의 수가 한정되어 있는데 반해 검색 결과의 수가 증가하므로 산출되는 결과이다.

6. 결 론

본 논문에서는 내용 기반 이미지 검색을 위해 사용되는 복합 질의문 처리 과정에서 질의 최적화 과정의 복합 질의문 수행 계획을 생성하는 알고리즘을 제안하였으며 그 성능을 분석하였다. 본 논문에서 제안한 SSCC 알고리즘은 기본의 병행 수행 알고리즘들에서 발생하는 투플 I/O수와 검색 시간을 단축시키기 위해 결과 집합에 대한 최소 유사도를 추정하여 각 단순 질의에 대한 검색 조건으로 이용하였다.

SSCC 알고리즘은 데이타베이스에 저장된 모든 특징 테이블의 투플들을 조사하지 않고 하나의 단순 질의를 만족하는 결과 집합에 속하는 이미지에 대한 특징들만을 조사함으로써 투플 I/O와 검색 시간을 단축시킬 수 있었다. 또한 실험 결과, 정확도와 재현율에 있어서도 기존의 병렬 처리 알고리즘보다 높은 성능을 나타냈다. SSCC 알고리즘은 데이타베이스에 인덱스가 구축되어 있을 경우 좋은 성능을 발휘하지만, 인덱스가 없는 경우에도 하나의 단순 질의 결과들에 대해 나머지 단순 질의 조건들을 검사하므로 최악의 상황에서도 A_0 알고리즘과 같은 수의 투플 I/O를 갖게 된다. 그러나 단순 질의를 처리하는 유사도 함수의 정확도가 낮아지면 질의 결과에 대한 정확도도 떨어지는 단점이 있다. 또한 데이타베이스에 저장된 이미지를 질의 이미지로 사용할 경우 검색 정확도가 100%에 가깝지만 축소/확대된 이미지를 질의로 사용할 경우 검색 정확도가 떨어지는 단점이 있다.

마지막으로 본 논문에서 제안한 복합 질의문 계획 생성 알고리즘은 정확한 검색 성능과 빠른 검색 시간으로 대용량 이미지 데이타베이스에서도 적용 가능하며 특정 이미지 영역에 한정되지 않으므로 다양한 이미지 데이타베이스 검색 시스템에서 사용할 수 있다. 또한 구현된 프로토타입 시스템은 내용 기반 이미지 검색을 위한 데이타베이스 검색 시스템에서 여러 특징 질의와 시각 질의들을 처리하는 질의 처리 모듈로 사용될 수 있을 것으로 기대된다.

참 고 문 헌

[1] 김기병, 김형주, "내용 기반 검색 및 주석 기반 검색을 통합하는 비디오 데이타 모델의 설계 및 구현", 한국정보과학회지 제3권 제2호, pp.115-126, 1997.

[2] Virginia E. Ogle and M. Stonebraker, "Chabot : Retrieval from a relational database of images," IEEE Computer, Vol.28, No.9, pp.40-48, 1995.

[3] 윤성민, "이미지 검색의 적중률 향상을 위한 분석기법", 동국대학교 대학원 컴퓨터공학 석사 학위 논문, 1998.

[4] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, W. Equiz, "Efficient and Effective Querying by Image Content," Journal of Intelligent Information System(JIIS), 3(3), pp.231-262, July 1994.

[5] 이석호, 송병호, 김범수, "멀티미디어 데이타베이스 관리 시스템에서의 내용기반 검색 기법에 관한 연구", 한국정보과학회 데이타베이스연구회지, 11권 4호. pp.102-119. 1995.

[6] Ronald Fagin, "Combining fuzzy information from multiple systemsm." in 15th ACM Symposium on Principles of Database Systems, pp.216-226, June 1996.

[7] Guang-Ho Cha, Chin-Wan Chung, "Object-Oriented Retrieval Mechanism for Semistructured Image Collections," in Proceedings of ACM Multimedia 98, Bristol, England, September 12-16, pp.323-332, 1998.

[8] Gholamhosein Sheikholeslami, Wendy Chang and Aidong Zhang, "Semantic Clustering and Querying on Heterogeneous Features for Visual data," in Proceedings of ACM Multimedia 98, Bristol, England, September 12-16, pp.3-12, 1998.

[9] Surya Nepal and M. V. Ramakrishna, "Query Processing Issues in Image(multimedia) Data-

bases," in Proceedings of Data Engineering, March 23-26, Sydney, Australia, pp.22-29, 1999.
[10] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, "Query by Image and Video Content : The QBIC System," IEEE Computer Vol.28 No.9, pp.23-32, Sep. 1995.
[11] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Addison Wesley, pp.45-47, 514-518 1993.
[12] R. Bellman and M. Giertz, "On the Analytic Formalism of the Theory of Fuzzy Sets," Information Sciences 5, pp.149-156, 1973.
[13] Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts International Editions, McGraw-Hill Co. Inc., pp.392-394, 1997.
[14] 박미화, 엄기현, "이미지 정보를 표현하기 위한 이중 그래프 데이타 모델", 한국정보과학회 '98 가을 학술발표논문집(I), pp.262-264. 1998.

박 미 화
1997년 동국대학교 컴퓨터공학과 학사. 1999년 동국대학교 컴퓨터공학과 석사. 1999년 9월 ~ 현재 동국대학교 컴퓨터공학과 박사과정. 관심분야는 데이타베이스, 멀티미디어 데이타베이스, 정보검색

엄 기 현
1975년 서울대학교 응용수학과 학사. 1977년 한국과학기술원 전산학과 석사. 1994년 서울대학교 컴퓨터공학과 박사. 1978년 3월 ~ 현재 동국대학교 컴퓨터멀티미디어 공학과 정교수. 1995년 3월 ~ 1999년 2월 동국대학교 정보관리 처장. 1997년 1월 ~ 현재 한국정보과학회 이사. 1998년 9월 ~ 2000년 8월 한국정보과학회 데이타베이스 연구회 운영위원장. 관심분야는 데이타베이스, 멀티미디어 데이타베이스, 정보시스템