

최소 공간관계를 이용한 효율적인 이미지 검색

(Efficient Image Retrieval using Minimal Spatial Relationships)

이 수 철 [†] 황 인 준 ^{**} 변 광 준 ^{***}
 (Soo Cheol Lee) (Een Jun Hwang) (Kwang Jun Byeon)

요약 멀티미디어 데이터베이스에서 이미지를 검색하기 위해 공간관계를 이용하는 것은 비주얼한 인터페이스 시스템을 통해서 효율적으로 수행할 수 있다. 시스템에서 이미지의 객체들은 2D 스트링으로 표현되고, 이것은 이미지 내의 객체를 추상화 하는 방법으로 객체의 심볼릭 프로젝션을 통해서 생성된다. 그러나 2D 스트링 표현기법을 이용한 이미지 객체간의 공간관계는 정확하지 않기 때문에 3D 이미지를 검색할 때 정확도가 떨어진다. 이러한 문제점을 해결하기위해 본 논문에서는 3D 이미지를 위한 공간 연산자를 이용한 공간관계를 제안하고, 이미지의 공간관계에서 중복되는 부분을 제거하기 위해 여러 가지 추론규칙을 이용한다. 이러한 규칙을 기반으로 하는 추론기법은 내용을 이용해 이미지를 검색할 때 질의 처리 시스템에서 사용되고, 기존의 방법보다 정확도와 융통성이 높다.

키워드 : 멀티미디어 데이터베이스, 이미지 데이터베이스, 이미지 검색, 공간 관계, 추석, XML

Abstract Retrieval of images from image databases by spatial relationship can be effectively performed through visual interface systems. In these systems, the representation of image with 2D strings, which are derived from symbolic projections, provides an efficient and natural way to construct image index and is also an ideal representation for the visual query. With this approach, retrieval is reduced to matching two symbolic strings. However, using 2D-string representations, spatial relationships between the objects in the image might not be exactly specified. Ambiguities arise for the retrieval of images of 3D scenes. In order to remove ambiguous description of object spatial relationships, in this paper, images are referred by considering spatial relationships using the spatial location algebra for the 3D image scene. Also, we remove the repetitive spatial relationships using the several reduction rules. A reduction mechanism using these rules can be used in query processing systems that retrieve images by content. This could give better precision and flexibility in image retrieval.

Key words : Multimedia database, Image database, Image retrieval, Spatial relationship, Annotation, XML

1. 서론

최근 컴퓨터 및 통신 기술이 발달하면서, 통신망에서 단순히 문자 정보만을 제공했던 것과는 달리 비디오와 오디오와 같은 멀티미디어 정보 제공에 대한 사용자의 욕구가 증가하고 있다. 하지만 대용량의 데이터베이스로

부터 멀티미디어 데이터를 검색 할 때는 많은 제약 사항을 안고 있다.

우선적으로 데이터베이스 안에 존재하는 멀티미디어 데이터에 대한 보다 효율적이고 정확한 인덱싱 기술의 개발이 어렵다는 사실이다. 기존의 멀티미디어 데이터 분석은 사용자가 멀티미디어 데이터를 직접 분석함으로써 적절한 내용을 텍스트 형태로 입력하는 방식이었다. 이와 같은 방법은 좋은 성능을 가질 수 있는 시스템을 구성 할 수 있지만, 관리자에게 많은 시간과 노력을 요구한다. 또한, 관리자의 개인적인 생각이 데이터에 포함될 수 있으므로 다양한 시각을 갖는 사용자의 욕구를 충족시켜 주기에는 한계가 있다. 따라서 다양한 사용자들의 질의를 충족 시켜줄 수 있도록 여러 특징들을 포함하는 내용기반 검색(Content-based retrieval)시스템

· 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성 지원사업의 연구결과로 수행되었음

[†] 학생회원 : 한국과학기술정보연구원 연구원
juin@kisti.re.kr

^{**} 종신회원 : 고려대학교 전자컴퓨터공학과 교수
(Corresponding author)
ehwang04@korea.ac.kr

^{***} 종신회원 : 아주대학교 정보및컴퓨터공학부 교수
byeon@ajou.ac.kr

논문접수 : 2003년 10월 6일

심사완료 : 2005년 4월 14일

[1]의 개발이 필요하다. 특히 이미지의 경우 내용기반 검색 기법이 가장 많이 사용되고 있다. 내용기반 이미지 검색이란 이미지가 가지고 있는 정보를 시각적인 특성을 이용하여 데이터베이스에 저장되어 있는 이미지를 검색하는 기법이다. 이미지 검색은 검색 기법, 질의 형태 등에 따라 다양하게 분류할 수 있지만, 근본적으로는 영상의 어떠한 특성을 검색에 이용하는가에 따라 분류된다. 초기의 이미지 검색 기법은 이미지의 색채 히스토그램, 질감 등 이미지의 포괄적인 특성을 검색에 이용하였다. 이러한 포괄적인 특성을 이용하는 경우, 데이터의 양이 적고 일차원적인 벡터로 표현이 가능하므로 다양한 형태의 인덱싱 기법을 이용하여 정확도를 높일 수 있었다. 그러나 이미지의 포괄적인 특성을 표현하는 방식은 사용자가 영상을 인지하고 기억하는 형식과 다르므로 질의를 표현할 수 있는 방법이 제한되어 일반적인 질의 방식이 예제에 의한 질의(Query By Example)[2, 3]로 한정된다. 예제에 의한 질의 방식은 사용자가 질의 화면에 제시된 여러 샘플 이미지에서 검색하고자 하는 이미지와 가장 유사한 특성을 갖고 있다고 생각되는 이미지를 선택하면 시스템이 해당 예제 이미지를 데이터베이스에서 찾아 결과로서 보여주는 방식이다. 일반적으로 예제 이미지는 데이터베이스에 저장되어 있는 이미지들 중 무작위로 추출된 샘플 이미지 중에서 고르게 되므로 질의 이미지에 대한 별도의 특성 추출이 필요 없다는 장점이 있지만, 사용자 입장에서 만족할 만한 예제 이미지를 찾기 위해 샘플 이미지를 일차적으로 검색해야 하는 불편함이 있고, 정확도가 떨어진다. 이러한 문제점을 해결하기 위해 이미지의 객체 공간 관계를 이용한 공간 유사도 기반 이미지 검색(Spatial Similarity-Based Image Retrieval)기법[4-10]을 사용한다. 이것은 격자 상에서의 객체간의 위치관계를 2-D string을 사용하여 표현하고, 질의와 유사한 공간 관계를 가진 이미지를 검색하는 기법이다. 그러나 이미지 객체간의 공간 관계를 2차원으로 표현하기 때문에 여러 가지 애매한(ambiguous) 문제가 발생한다. 예를 들면, 실제 이미지에서 두 객체간의 공간 관계는 앞/뒤 관계이지만 2차원으로 표현을 했을 때에는 앞/뒤가 아닌 중복으로 표현이 되기 때문에 검색결과의 정확도가 떨어질 수밖에 없다. 또한 이미지의 객체 수가 증가할수록 객체간의 공간 관계의 수도 기하급수적으로 증가하기 때문에 검색 속도나 정확도 면에서 문제가 발생한다.

본 논문에서는 이러한 문제점을 해결하기 위해서 이미지 객체간의 공간 관계에서 3차원 관계를 인식할 수 있도록 3차원 인터페이스와 중복되는 공간 관계를 제거하고 유추할 수 있는 알고리즘을 제안한다. 이것은 이미지에 존재하는 객체들 중에 사용자가 원하는 객체를 분

리하고, 분리된 객체들의 특성과 그들 간의 공간 관계를 분석한다. 또한 분석된 결과는 XML을 이용해 추색 처리함으로써 보다 정확하고 빠른 결과를 얻을 수 있다.

본 논문의 구성은 다음과 같다. 제2장에서는 관련연구로 현재 나와 있는 이미지 검색 시스템과 여러 검색 기법에 대해 기술한다. 제3장에서는 이미지 검색을 위한 인덱싱에 대해서 설명한 후, 각 객체간의 관계를 나타내는 그래프를 정의한다. 제4장은 중복되는 공간 관계를 제거하기 위한 추론 규칙에 대해서 설명한다. 5장은 구현과 실험에 관련된 내용으로서 3가지 검색방법을 이용하여 3가지 태스크를 수행했을 때의 유효성과 효율성의 비교에 대해서 기술한다. 마지막으로 제6장에서 결론을 맺는다.

2. 관련연구

컴퓨터 하드웨어와 소프트웨어가 발전함에 따라서 숫자나 문자를 위주로 처리하던 기존의 데이터베이스 시스템은 다양한 미디어를 처리하게 되었다. 텍스트, 그래픽, 이미지, 오디오, 비디오 등의 비정형 데이터를 멀티미디어 데이터라 하고 이를 효율적으로 관리하기 위해 멀티미디어 데이터베이스가 사용된다.

기존의 정형 데이터에 대해서는 데이터 값에 기반한 검색을 하기 쉽지만 멀티미디어 데이터의 경우 내용을 기반으로 한 검색이 어렵다. 따라서 완전히 일치되는 대상을 찾아내기 어려우므로 검색 대상 공간을 축소한 다음 데이터를 검색 순서대로 탐색하는 방법을 사용한다. 이미지의 경우 검색 대상이 되는 이미지의 여러 특성을 추출하고, 사용자가 질의로 준 이미지의 특성과 유사한 특성을 가진 이미지를 검색 결과로 알려준다. 사용되는 이미지 특성에는 색상, 질감, 형상 등이 있다. 이미지에 텍스트로 설명을 붙이는 경우에는 키워드를 사용하여 원하는 데이터를 질의하고 검색할 수도 있다. QBIC[11]은 내용기반 검색을 이용한 대표적인 시스템으로 이미지를 효율적으로 관리, 조직하고 탐색하는 도구이다. IBM에서 개발되었으며, 데이터베이스에 저장된 이미지에 대해 시각적인 내용으로 질의를 할 수 있다. 이미지에 포함된 객체는 다르지만 색상이 유사한 경우 더 정확한 질의를 하기 위해 키워드나 텍스트를 사용한다. 또 다른 내용기반 검색 시스템들은 보다 정확한 이미지 표현 방법을 사용하는데 대표적인 시스템으로는 Virage[12]와 Chabot[13]이 있다. 하지만 대부분의 검색 시스템들은 이미지에 존재하는 객체간의 공간 정보에 대해서는 전혀 고려를 하고 있지 않다. 일반적으로 색상과 공간정보를 이용한 검색 기법은 이미지를 여러 개의 영역으로 분할하고 각 영역에서 색상 히스토그램 값을 뽑아내고, 이미지내의 색상의 공간 분포는 각 영역에서 색상의 교차점에 의해 표현된다. Quad-tree를 기반으로 한 영역

기본 이미지 검색기법에서는 동종의 이미지 객체 클러스터를 얻기 위해 객체들을 $2^N * 2^N$ 개의 이진배열로 이미지를 표현하게 된다.

2D-string을 기반으로 하여 효율적인 인덱싱, 융통성 있는 정보검색 공간 추론과 시각화를 위한 기법이 제안되었다. 공간 관계는 퍼지 개념이기 때문에 예제에 의한 이미지 검색에서는 유사 검색의 능력이 필수적이다. 2D-String[14]을 이용한 공간관계 표현 기법은 x축과 y축에 따라서 이미지 객체를 표현하는 것으로 이미지에 있는 객체간의 방향(direction) 관계를 스트링 형태로 표현한다.

질의 처리는 질의로 들어온 이미지를 2D string과 비교하여 같으면 질의 이미지와 같은 이미지라 하여 검색하는 방식으로 진행된다. 이렇게 이미지 검색을 위해 픽셀 이미지를 모두 비교하는 것과는 달리 스트링 매칭으로 하여 좀 더 빠르게 검색할 수 있다. 만약 이미지 데이터베이스에 저장된 이미지의 수가 굉장히 많다면 하나의 이미지 질의를 처리할 때 데이터베이스 있는 전체 2D string을 다 비교하는 저장된 이미지 데이터 수에 따른 선형적 검색이 되어 문제가 된다. 따라서 2D-H, 2D-PIR과 같은 확장된 형태의 객체 표현법을 사용한다. 2D-H string과 2D string은 단지 방향 관계만을 표현하지만 2D-PIR string의 경우는 이미지 객체간의 방향과 위상 관계를 표현함으로써 다른 표현 기법보다 효율성 면에서 뛰어나다. 표 1은 기존의 2D-String을 이용한 공간관계 표기법과 제안하는 기법간의 비교를 나타내는 표이다.

3. 이미지 검색을 위한 인덱싱

이미지 데이터베이스에서 효율적인 이미지 검색과 저장을 위해서 가장 중요한 것은 인덱싱이다. 이미지는 이미지 객체에 대한 일반적인 정보와 이러한 정보에 관련된 모양(shape), 공간위치(spatial location)와 같은 두

가지 정보를 포함하고 있다. 이미지 데이터베이스 내의 이미지들을 효율적으로 관리하기 위해서 각 이미지에 공간정보를 내포(embed)시킨다. 본 논문에서 이미지 객체의 공간정보는 x축, y축, z축에 따라서 이미지 객체를 표현하는 3D방식을 사용하고, 이미지 데이터베이스에 질의를 하기위해서 사용자는 자신이 관심 있는 이미지 객체를 선택하는 ROI(Region of Interest)[9,14]를 지정한다. 이렇게 지정된 객체들은 2D-string 형태로 표현되는 것이 아니라 3차원 형태로 표현된다.

3.1 3차원 공간 관계의 표현

데이터베이스에 저장된 모든 이미지들은 각각이 독특한 특성을 가진 객체들로 구성되어 있고, 이러한 객체들 간에는 다양한 공간 관계가 존재할 수 있다. 공간 관계는 상대좌표와 절대좌표로 표현할 수 있고, 3차원 공간에서의 경우 객체 O의 공간 위치 좌표 $P_0=(X_0, Y_0, Z_0)$ 로 나타낼 수 있다. 따라서 만약 한 이미지들에 n개의 객체가 존재하는 경우 전체 이미지는 $P=(P_1, P_2, \dots, P_n)$ 의 위치 좌표의 집합으로 표현될 수 있다. 각 위치 좌표에 해당하는 객체는 의미적 정보를 가지고 있기 때문에, 이러한 정보를 주석 처리할 수 있다. 이와 같은 위치 좌표를 공간 위치 점(Spatial location point)[9]이라고 부른다. 단순화를 위해서 이미지 객체는 하나의 공간 위치 좌표로 표현된다. 이미지의 공간 위치 점들 간의 공간관계를 표현하기 위해서, 이미지를 같은 크기의 3차원 공간으로 분할한다. 그림 1은 공간 위치 점이 서로 다른 공간에 위치한 공간 위치 좌표의 예를 보여준다.

Allen은 시간 간격으로 표현되는 이벤트간의 시간 관계를 추론하고 이것을 표현하기 위해서 표 2의 시간 간격 대수(Temporal interval algebra)를 제안하였다. 대수의 각 요소들은 7가지의 기본 관계와 그것의 역(Inverse)으로 구성되어 있다.

표 1 기존 기법과의 비교

	9DLT	2D-String	2DC-String	제안기법
표기법	9개의 방향에 정수 0, 1, 2, ..., 7, 8을 할당하여 표기.	Type 0, 1, 2의 3종류의 공간관계를 표현. 예) Type 0 $V_{ab}="0"$ if $r(a) = r(b)$ $V_{ab}="1"$ or $V_{ab}="0"$ if $r(a) < r(b)$ $V_{ab}="2"$ or $V_{ab}="1"$ if $r(a) > r(b)$	객체의 영역들간의 공간관계를 표현. 총 13개의 관계를 가지고 있음. 예) • After • Meet with right adjacency • Is include	객체의 공간좌표와 MBR을 이용. 총 12개의 관계를 가지고 있음. 예) • Left • Upper • Below
특징	표현할 수 있는 공간관계가 제약이 있다.	공간관계 계산을 위해서 많은 연산이 필요함	아이콘 이미지 사용으로 반복되는 이미지 이해작업 요구를 피할 수 있다. 고수준 사용자의 질의에는 부적합하다.	3차원 공간관계를 표현할 수 있다. 이미지 객체의 수가 증가하면 공간관계가 복잡해진다. X, Y, Z 축을 이용해 객체의 좌표를 계산한다.

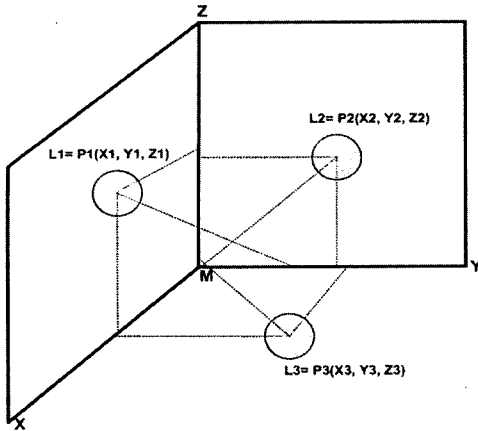


그림 1 공간 위치 좌표

표 2 Allen의 시간 간격 대수

Relation	Symbol	Inverse	Meaning
A before B	b	bi	AAA BBB
A equal B	e	ei	AAA BBB
A meets B	m	mi	AAABBB
A overlaps B	o	oi	AAA BBB
A starts B	s	si	AAA BBBBB
A finishes B	f	fi	AAA BBBBB
A during B	d	di	AAA BBBBB

본 논문에서는 표 2의 Allen의 시간 간격 대수를 이용하여 X, Y, Z 좌표에 있는 이미지 객체의 이미지 관계 대수(Image relation algebra)를 표 3과 같이 정의한다. 이미지 관계 대수는 8개의 방향성 관계와 4가지 위

상 관계로 구성되어 있다. 방향성 관계는 다음과 같이 2가지로 나뉘어져 있다.

- 방향성 관계: upper, below, left, right
- 혼합형 방향성 관계: leftupper, leftbelow, rightupper, rightlower

이미지 객체를 표현하기 위해서 A, B와 같은 임의의 문자를 사용하고 x, y 축과의 프로젝션(Projection)은 A_x , A_y 와 같이 나타낸다. 표 3에서 기호 \wedge 과 \vee 은 표준 논리 연산자인 AND와 OR를 의미하고, 기호 $\{ \}$ 은 간격 연산자(Interval operator)를 논리 연산자와 같이 사용하기 위해서 사용한다. 예를 들면, $A_x(b, m, o)B_x$ 는 $A_x b B_x \vee A_x m B_x \vee A_x o B_x$ 와 같은 의미이다.

n개의 객체들로 구성된 이미지가 있다고 가정할 때, 그래프를 이용해서 객체간의 공간 관계를 정의할 수가 있는데, 이것을 공간 그래프(Spatial graph)라고 한다. 그림 2는 원본 이미지와 공간 그래프를 나타낸다.

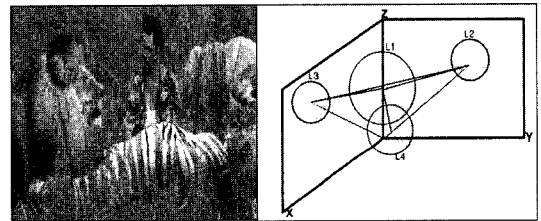


그림 2 원본 이미지와 공간 그래프

4. 공간 관계 추론규칙

이미지 객체의 증가에 따라서 공간 관계도 이에 비례하여 증가하게 된다. 그러나 이렇게 나온 공간 관계들은 중복되는 경우가 많다. 객체간의 중복된 공간 관계를 제

표 3 이미지 관계 대수

Relation	Meaing	Definition
A BL B	Below	$A_x\{d, di, s, si, f, fi, e\}B_x \wedge A_y\{b, m\}B_y$
A UP B	Upper	$A_x\{d, di, s, si, f, fi, e\}B_x \wedge A_y\{bi, mi\}B_y$
A LT B	Left	$A_x\{b, m\}B_x \wedge A_y\{d, di, s, si, f, fi, e\}B_y$
A RT B	Right	$A_x\{bi, mi\}B_x \wedge A_y\{d, di, s, si, f, fi, e\}B_y$
A LU B	Leftupper	$(A_x\{b, m\}B_x \wedge A_y\{bi, mi, oi\}B_y) \vee (A_x\{o\}B_x \wedge A_y\{bi, mi\}B_y)$
A LL B	Leftlower	$(A_x\{b, m\}B_x \wedge A_y\{b, m, o\}B_y) \vee (A_x\{o\}B_x \wedge A_y\{b, m\}B_y)$
A RU B	Rightupper	$(A_x\{bi, mi\}B_x \wedge A_y\{bi, mi, oi\}B_y) \vee (A_x\{oi\}B_x \wedge A_y\{bi, mi\}B_y)$
A RL B	Rightlower	$(A_x\{b, m\}B_x \wedge A_y\{b, m, o\}B_y) \vee (A_x\{oi\}B_x \wedge A_y\{b, m\}B_y)$
A OL B	Overlaps	$A_x\{d, di, s, si, f, fi, o, oi, e\}B_x \wedge A_y\{d, di, s, si, f, fi, o, oi, e\}B_y$
A IS B	Inside	$A_x\{d\}B_x \wedge A_y\{d\}B_y$
A OS B	Outside	$A_x\{di\}B_x \wedge A_y\{di\}B_y$
A IN B	In front of	$A_z\{di\}B_z \wedge A \text{ behind } B \Leftrightarrow A_z\{bi, mi\}B_z$

거하게 위해서 본 논문에서는 몇 가지 추론 규칙을 제안한다. 여기서 제거되는 공간 관계는 중복되거나 유추(Inferable)가 가능한 것이다.

규칙들은 지식표현(Knowledge representation)과 전산학의 추론에서 사용되는 것으로 특히 인공지능에서 널리 이용된다. 이것은 규칙이 쉽게 이해할 수 있고, 구현하기가 쉽다는 것에 기인한다. 추론 규칙을 설명하기 전에 논문에서 사용하는 몇 가지 표기법을 정의한다. 만약 x 를 통해서 y 를 유추할 수 있다면 $x \Rightarrow y$ 라고 표기한다. 또한 $x \Rightarrow y$ 이고 $y \Rightarrow x$ 이면 $x \Leftrightarrow y$ 라고 표기한다.

- **규칙 1(반사성)** 다음의 위상 관계는 반사성(Reflexivity)을 가지고 있다.

$$A e A \quad A o A$$

- **규칙 2(대칭성)** 다음의 위상 관계는 대칭성(Symmetry)을 가지고 있다.

$$A e B \Leftrightarrow B e A \quad A o B \Leftrightarrow B o A$$

- **규칙 3(역 속성)** 다음의 방향 관계들은 서로가 역(Inverse)의 관계가 있다.

$$A_{UP} B \Leftrightarrow B_{BL} A \quad A_{RU} B \Leftrightarrow B_{LL} A$$

$$A_{LU} B \Leftrightarrow B_{RL} A \quad A_{RT} B \Leftrightarrow B_{LT} A$$

- **규칙 4(전이성)** $\Theta \in \{LU, RU, LL, RL, LT, RT, IS, E\}$ 라면 아래와 같이 규칙을 정의 할 수 있다.

$$A \Theta B \wedge B \Theta C \Rightarrow A \Theta C$$

- **규칙 5(유추성)** 몇몇 관계들로부터 다른 관계들을 유추(Implication)할 수 있다.

$$A_{IS} B \Rightarrow A_{OL} B \quad A_{E} B \Rightarrow A_{OL} B$$

- **규칙 6** 이 규칙은 아래처럼 IS와 {OL, OS}간의 관계를 이용한다.

$$A_{IS} B \wedge A_{OL} C \Rightarrow B_{OL} C$$

$$A_{IS} B \wedge A_{OS} C \Rightarrow B_{OS} C$$

- **규칙 7** 이 규칙은 위상 관계 IS와 방향 관계인 {LT, RT, UP, BL}간의 관계를 나타낸다. 만약 $\Theta \in \{LT, RT, UP, BL\}$ 라면 아래와 같이 규칙을 정의 할 수 있다.

$$A_{IS} B \wedge B \Theta C \Rightarrow A \Theta C$$

- **규칙 8** 만약 $\Theta \in \{LT, RT, UP, BL\}$ 라면, OL과 {LU, RT, UP, BL}간의 규칙은 다음과 같이 정의 할 수 있다.

$$A \Theta B \wedge B_{OL} C \wedge C \Theta D \Rightarrow A \Theta D$$

- **규칙 9** 이 규칙은 {LT, RT, UP, BL}와 {LU, RU, RL, LL}간의 관계를 나타내는 것으로 아래와 같이 정의할 수 있다.

$$A_{UP} B \wedge B_{LU} C \Rightarrow A_{LU} C$$

$$A_{UP} B \wedge B_{RU} C \Rightarrow A_{RU} C$$

$$A_{BL} B \wedge B_{LL} C \Rightarrow A_{LL} C$$

$$A_{BL} B \wedge B_{RL} C \Rightarrow A_{RL} C$$

$$A_{LT} B \wedge B_{LU} C \Rightarrow A_{LU} C$$

$$A_{LT} B \wedge B_{RU} C \Rightarrow A_{LT} C$$

$$A_{RT} B \wedge B_{RU} C \Rightarrow A_{RU} C$$

$$A_{RT} B \wedge B_{RL} C \Rightarrow A_{RL} C$$

- **규칙 10** 이 규칙은 {LT, RT, UP, BL}와 {LU, RU, RL, LL}간의 관계를 나타내는 것으로 아래와 같이 정의할 수 있다.

$$A_{LT} B \wedge B_{\{LU, LL\}} C \Rightarrow A_{LT} C$$

$$A_{RT} B \wedge B_{\{RU, RL\}} C \Rightarrow A_{RT} C$$

$$A_{UP} B \wedge B_{\{RU, LU\}} C \Rightarrow A_{UP} C$$

$$A_{BL} B \wedge B_{\{RL, LL\}} C \Rightarrow A_{BL} C$$

4.1 추론 규칙의 적용

그림 3과 같은 동물원 기차에서 눈에 띄는 객체들은 사람(man), 기관사실(cab), 창문(window), 왼쪽 큰 바퀴(lbw), 오른쪽 작은 바퀴(rsw), 굴뚝(smokestack), 기차 몸통(body) 이렇게 7가지임을 알 수 있다. 이미지에서 다음과 같은 공간 관계들을 생각 할 수 있다.

{window IS cab, cab OS lbw, cab LT body, cab LU lbw, body UP lbw, smokestack RU body, lbw LT rbw, man LT cab}

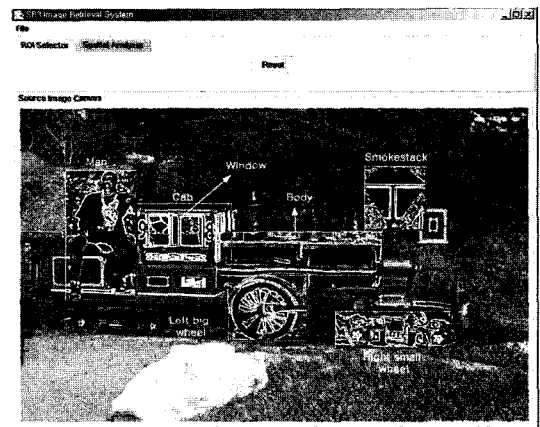


그림 3 동물원 기차 이미지와 객체들

위에서 언급한 추론 규칙을 적용하면 다음과 같은 공간 관계를 유추할 수 있다.

- 기관사 실은 기차 몸통의 왼쪽에 있고 기차 몸통은 오른쪽 작은 바퀴의 왼쪽 위에 위치한다(cab LT body \wedge body LU rsw). **규칙 10**을 적용하여 기관사 실이 오른쪽 작은 바퀴에(cab LT rsw) 있음을 유추할 수 있다.
- 창문은 기관사 실의 안에 있고 기관사 실은 왼쪽 큰 바퀴의 바깥에 위치한다(window IS cab \wedge cab OS

lbw). 규칙 6을 적용하면 창문이 왼쪽 큰 바퀴의 바깥쪽에 위치함(window os lbw) 유추.

- 그림에서 창문은 기관사 실의 안쪽에 위치하고 기관사 실은 왼쪽 큰 바퀴의 위에 위치함을 알 수 있다(window is cab ^ cab up lbw). 규칙 7을 이용하여 창문이 왼쪽 큰 바퀴위에(window up lbw) 있음을 유추.
- 사람은 왼쪽 큰 바퀴의 왼쪽에 위치하고 왼쪽 큰 바퀴는 굴뚝의 왼쪽 아래에 위치하고 있다(man LT lbw ^ lbw LL smokestack). 규칙 10을 적용하여 사람이 굴뚝 왼쪽(man LT smokestack)에 있음을 유추.
- 왼쪽 큰 바퀴는 몸통의 아래에 있고 몸통은 굴뚝의 왼쪽 아래에 위치하고 있다(lbw BL body ^ body LL smokestack). 규칙 9를 적용하여 왼쪽 큰 바퀴는 굴뚝의 왼쪽에(lbw LT smokestack) 있음을 유추.

위에서 여러 가지 추론 규칙을 적용하여 새로운 공간 관계를 유추 하였지만, 유추 가능한 모든 관계를 기술한 것은 아니다.

5. 구현 및 실험

본 논문에서는 이미지간의 3차원 공간 유사도와 주석을 이용한 검색 기법을 제안하였고, 이것을 기반으로 보다 진보된 형태의 이미지 검색 시스템을 구현하였다. 이미지에 대한 주석은 XML 데이터베이스에 저장된다. 이때 저장되는 데이터의 형태는 최근에 많이 사용하는 웹 언어인 XML(eXtensible Markup Language)이다. XML은 SGML에서 유래된 것으로 보다 간결하고, 융통성이 좋다. 그래서 XML은 화학, 수학과 같은 여러 분야에서 널리 사용되고 있다. 시스템에서 분석된 이미지 객체들의 공간정보와 특징들은 XML형태로 표현되고, 저장된다.

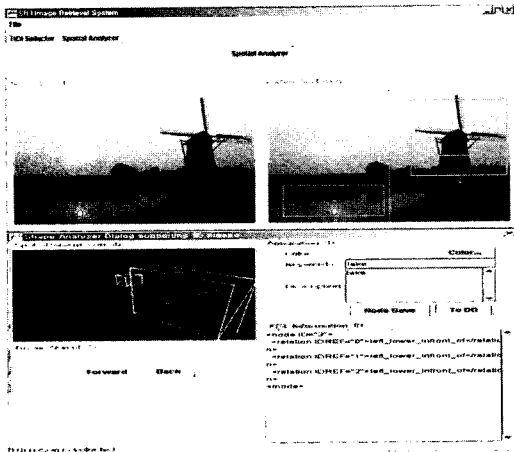


그림 4 이미지 분석 인터페이스

그림 4는 이미지 검색 인터페이스를 나타내는 것으로, 각 이미지 객체들은 사각형으로 표시되고, 사각형에 포함되어 있는 객체들의 엣지들이 추출된다. 그리고 선택된 객체들의 의미 정보들은 XML 코드에 포함되어 진다. 왼쪽 버튼이 있는 윈도우에는 사용자가 선택한 이미지 객체들 간의 공간관계 그래프를 나타낸다. 공간관계 그래프의 이미지 객체들은 전/후 버튼을 이용해서 객체의 3차원 위치를 조정할 수 있다. 그림 5는 질의 윈도우 인터페이스를 보여준다.

5.1 실험

본 논문에서는 실험을 위해서 3가지 검색 방법을 비교하였다. 3차원 관계를 이용한 질의(Query-by-Spatial relationships using 3D, QbS³), 2차원 관계를 이용한 질의(QbS²), 내용기반 질의(Query-by-Content, QbC) 방법이다. 이 3가지 검색 방법을 위해 사용되는 이미지 데이터베이스에는 약 1000개의 이미지가 저장되어 있다.

다음은 실험을 위한 3가지의 다른 검색 태스크(Task)를 나타낸다.

1. 유사한 구조(Structure)를 가지고 있는 이미지를 검색한다. 즉, 해당 태스크의 이미지들은 서로가 비슷한 레이아웃을 가지고 있지만, 세부적으로는 다르다.
2. 색상과 영역(Boundary)이 다르고 구조가 유사한 이미지를 검색한다.
3. 구조가 서로 다른 이미지를 검색하고, 해당 이미지들은 복잡한 풍경의 비디오에서 얻어진 것이다.

5.1.1 유효성

각각 다른 태스크에 따른 질의 방법을 비교하기 위해서, 본 논문에서는 검색 유효성 P(i)를 평가한다. P(i)는 주어진 i 번째 질의에서 검색된 이미지의 수와 해당 이미지가 타겟 셀에 부합되는 수의 비율로 정의할 수 있다.

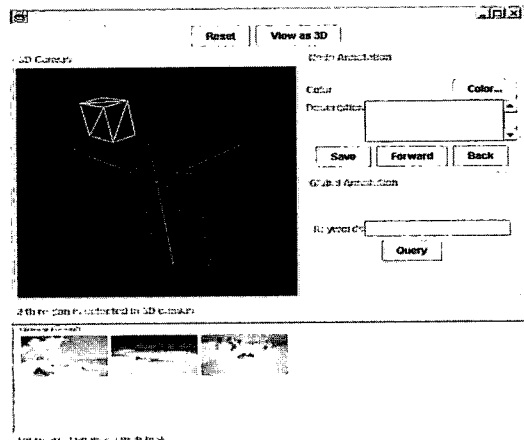


그림 5 질의 윈도우 인터페이스

$$P(i) = \frac{\text{Numer of images which belong to the target set}}{\text{Number of images retrieved with the } i^{\text{th}} \text{ query}}$$

그림 6은 3가지 검색 기법에 대한 종합적인 검색 유효성을 보여주는 그래프이다. 각 성능 그래프들은 QbS₃, QbS₂와 QbC를 나타낸다.

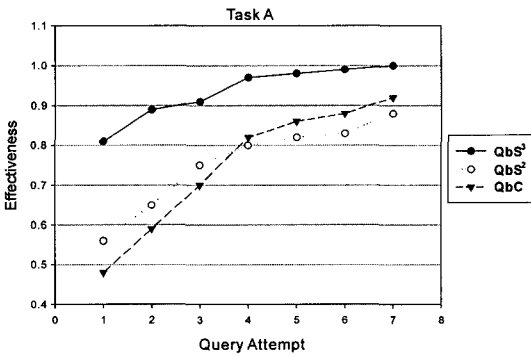


그림 6 검색 유효성

그림 6을 보면 QbS³기법이 다른 기법보다 검색 유효성이 높은 것을 알 수 있다. 3번째 질의 시도(Query attempt)를 보면 P_{QbS³}(3) = 81%, P_{QbS²}(3) = 66%, P_{QbC}(3) = 54%의 검색 유효성을 보여준다. QbS²와 QbC기법은 검색 성능을 향상시키기 위해서 질의 시도 횟수를 증가 시켜야 하고, 그 횟수가 7번째 정도 되면 QbS³와 성능이 비슷해짐을 알 수 있고, 또한 QbC는 다른 두 가지 기법보다 성능이 떨어짐을 확인 할 수 있다.

그림 7, 8, 9는 각 태스크에 대한 3가지 검색 기법의 유효성을 보여주는 그래프이다. 첫 번째 태스크(그림 7)에서는 QbS³이 가장 좋은 성능을 보이고, QbS²가 성능이 좋지 않음을 볼 수 있다. QbS³기법은 객체의 선택과 객체간의 관계 설정 위해서 가상공간을 손쉽게 이동할 수 있다. 하지만 QbS²에서는 정확한 객체의 선택과 관계 설정이 어렵다. QbC기법에서는 예제 이미지에서 타겟 이미지와 유사한 내용의 이미지를 찾기가 힘들다. 두 번째 태스크(그림 8)에서 QbS²와 QbC는 동일한 성능을 보이고 QbS³보다는 성능이 좋음을 알 수 있다. 이것은 타겟이 되는 이미지들의 객체들이 두 가지 색상과 서로의 영역이 다른 간단한 형태의 이미지이기 때문이다. 이러한 이미지 검색에는 2차원 공간관계를 이용한 질의가 적합하다.

세 번째 태스크(그림9)에서는 QbS³와 QbS²가 QbC보다 월등하게 성능이 좋음을 알 수 있다. 타겟 이미지들은 같은 풍경에서 단지 시각(Viewpoint)의 차이에 의해서 다른 이미지들로 구성되어 있다. QbS³기법을 이용하여 이미지 객체의 공간 관계를 가상공간에서 설정한 후에 손쉽게 질의를 만들 수 있다. 타겟 이미지의 복잡한

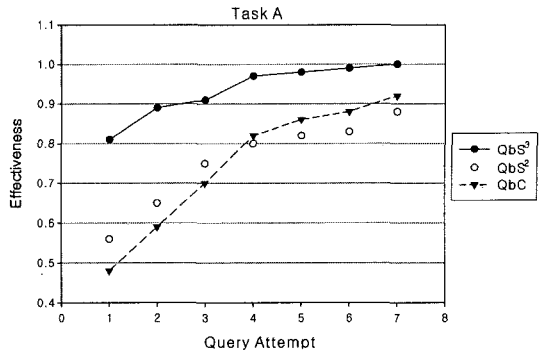


그림 7 이미지 셋(A)의 검색 유효성

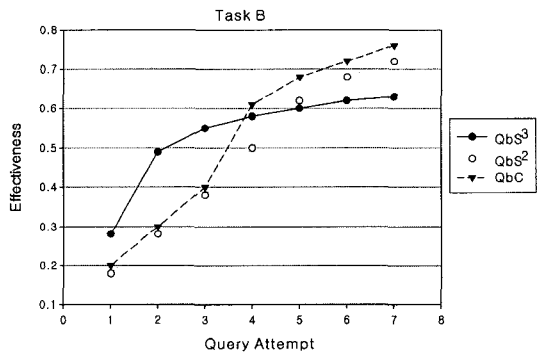


그림 8 이미지 셋(B)의 검색 유효성

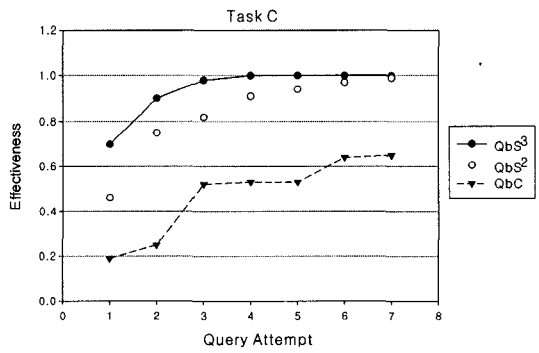


그림 9 이미지 셋(C)의 검색 유효성

풍경 때문에 상대적으로 QbS²의 성능이 떨어진다. 일반적으로 QbC의 성능은 타겟 셋과 데이터베이스의 크기에 좌우되고, 이것은 그만큼 질의에 부합하는 이미지가 많다는 것을 의미한다.

5.1.2 효율성

3가지 기법에 대한 효율성은 태스크(T_{task})를 수행하는 평균시간에 따라서 측정 할 수 있다. 해당 값은 아래의 표 4와 같다.

표 4 평균시간

Scheme	T _{taskA}	T _{taskB}	T _{taskC}
QbS ³	4.4	17.0	8.1
QbS ²	11.6	10.6	16.1
QbC	3.7	6.9	4.1

표 4에서 QbC가 다른 기법에 비해서 평균시간이 가장 좋음을 알 수 있고, 태스크 C를 제외하고는 QbS²가 시간이 가장 오래 걸린다.

총 소요 시간은 각 질의 시도에 걸리는 시간의 합으로 결정된다. 질의 시도 시간은 질의 명세(Query specification), 매칭(Matching), 비주얼화(Visualization), 사용자 피드백(User feedback)의 합으로 식으로 표현하면 다음과 같다.

$$T_{task} = \sum_{q=1}^n T_{query}(q)$$

$$T_{query} = T_{query\ specification}(q) + T_{matching}(q) + T_{visualization}(q) + T_{feedback}(q)$$

T_{matching}은 QbC보다 QbS³와 QbS²가 높는데 이것은 QbS³와 QbS²가 실행시간에 이미지의 공간 관계를 분석하기 때문이다.

T_{query specification}은 실험 세션의 로그에서 얻어지는 값으로 표 5와 같다.

표 5 질의명세 평균시간

Scheme	T _{query specification}
QbS ³	1.89
QbS ²	1.65
QbC	1.04

실험 세션의 분석을 통해서 3가지 다른 기법을 이용한 검색 태스크는 질의 시도가 증가하면 질의명세 시간이 줄어들음을 알 수 있다. 그림 10은 질의 시도에 따른 T_{query specification} 값을 나타낸다.

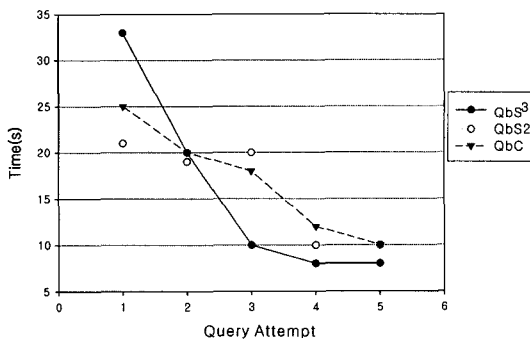


그림 10 실험 세션에서 질의를 명세하기 위한 소요시간

질의를 명세하기 위해서 QbS³와 QbS²가 소요되는 시간은 점점 줄어들고, QbC의 시간도 일정하게 줄어든다. 표에서 QbS³와 QbS²의 시간은 첫 번째 질의 시도에서 많은 시간이 소요됨을 알 수 있는 데, 이것은 두 기법이 공간 관계를 사용하기 때문이다. 이와 달리 QbC는 거의 일정한 수치를 보이는데 이것은 공간 관계를 분석하기 위한 일련의 연산이 필요가 없기 때문이다.

6. 결론

본 논문에서는 공간 정보를 이용한 기존의 검색시스템이 가지고 있는 부정확성을 해결하기 위해서 3차원 정보가 포함된 공간 연산자와, 연산자간의 중복문제를 해결하기 위해 공간 관계 추론규칙을 제안하였고, 이러한 기법을 이용한 검색 시스템을 구현하였다. 제안된 추론규칙은 질의 처리 시에 발생하는 오버헤드문제와 저장 공간문제를 해결하였다. 또한 이미지 검색의 정확도와 융통성 측면에서 효율적임을 알 수 있었다. 또한 위치 연산자 간에 유사도 값을 정의함으로써 질의와 데이터베이스 내의 이미지 간의 유사도를 수치화함으로써 보다 정확한 결과 값을 얻을 수가 있었다.

실험 결과에서 알 수 있듯이 공간 정보와 색상이나 키워드 같은 특징을 복합적으로 이용하여 질의를 처리할 경우 정확도가 높아짐을 알 수 있었다.

이미지 데이터베이스의 인덱싱 역시 검색의 효율을 결정짓는 중요한 요소이기 때문에 향후에는 구현된 시스템에 다차원 인덱싱 기법을 적용하여 대용량의 이미지 데이터에 대해서도 효율적인 검색이 가능하도록 할 것이다.

참고 문헌

- [1] A. Pentland, R. Picard and S. Sclaroff, "Photo-book: Content-based manipulation of image databases," *SPIE Proc. Storage of Retrieval for Image and Video Databases*, February 1994.
- [2] A. Gupta and R. Jain, "Visual information retrieval," *Comm. Assoc. Comp. Mach.*, May 1997.
- [3] N. Chang and K. Fu, "Query by pictorial example," *IEEE Trans. Software Eng.*, Vol. SE-6, pp. 519-524, Nov. 1980.
- [4] C. Carson, S. Belongies, H. Greenspan and J. Malik, "Region-based image querying," *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, June 1997.
- [5] Egenhofer M. J. and Franzasa R. D, "Point set topological spatial relations., *Journal Geogr. Information System*, vol 5-2, pp. 161-174, 1991.
- [6] Lee, S.Y. and F.J. Hsu, "Spatial reasoning and similarity retrieval of images using 2D-C String knowledge representation," *Pattern Recognition*,

Vol 25-3, pp. 305-318, 1992.

[7] M. Nabil, A.H.H. Ngu, and J. Shepherd, "Picture Similarity Retrieval Using the 2D Projection Interval Representation," *IEEE Trans. Knowledge and Data Eng.*, vol. 8, no. 4, pp. 533-539, Aug. 1996.

[8] S. Chang, Q. Shi and S. Yan, "Iconic indexing using 2-D strings," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, Vol. 9, No. 3, pp. 413-428, 1987.

[9] S. Lee and E. Hwang, "Spatial Similarity and Annotation-Based Image Retrieval System," *IEEE Fourth International Symposium on Multimedia Software Engineering*, Newport Beach, CA, December 2002.

[10] S.L. Tanimoto, "An iconic/symbolic data structuring scheme," in *pattern Recognition and Artificial Intelligence*. NY: Academic, 1976, pp. 452-471.

[11] W. Niblack, et al. "The QBIC project: Query images by content using color, texture and shape," *SPIE V 1908*, 1993.

[12] Bach JR, Fuller C, Gupta A, Hampapur A, Horiwitz B, Humphrey R, Jain RC, Shu C, "The virage image search engine: An open framework for image management," *SPIE Proceedings of Storage and Retrieval for Still Images and Video Databases IV*, Feb 1996, San Jose, CA, pp. 76-86.

[13] V. E. Ogle and M. Stonebraker, "Chabot: Retrieval from a Relational Database of Images," *IEEE Computer*, Vol. 28, No. 9, September 1995.

[14] L. H. Rodrigues, Building Imaging Applications with Java Technology, Addison Wesley, 2001.

[15] Eakins, J. P., "Automatic image content retrieval: Are we going anywhere?," In *proceedings of the 3rd Int. Conf. on Electronic Library and Visual information Research.*, May. 1996.

[16] Jagadish HV, "A retrieval technique for similar shape," *Proc. ACM SIGMOD conf.*, May 1991, pp. 209-217.

[17] J. R. Smith and S. Fu Chang, "Visual SEEK: A Fully Automated Content Based Image Query System," *Proc. ACM Mult. Conf., Boston Ma., Nov. 1996.*

[18] J. R. Smith and S. Fu Chang, "Tools and techniques for color image retrieval," *Proc. IEEE Int. Conf. on Image Proc.*, pp. 52-531, 1995.

부 록

A-1 추론규칙 증명

규칙의 증명을 위해서 Allen의 전이성 테이블의 결과와 다음의 몇 가지 명제(Lemma)를 이용한다.

- 명제 1 {d, di, s, si, f, fi, e}와 {d, di, s, si, f, fi, o, oi, e}는 대칭적이다.
 $A \{d, di, s, si, f, fi, e\} B \Leftrightarrow B \{d, di, s, si, f, fi,$

- $e\} A$
 $A \{d, di, s, si, f, fi, o, oi, e\} B \Leftrightarrow B \{d, di, s, si, f, fi, o, oi, e\} A$
 증명: 만약 모든 관계와 그것의 역이 하나의 집합이라면 명제 1의 증명은 매우 간단하다. 예를 들면, $A\{d\}B \Leftrightarrow B\{di\}A$ 에서 $A\{d, di\}B \Leftrightarrow B\{d, di\}A$ 가 됨을 손쉽게 알 수 있다.
- 명제 2 $A(o)B \wedge B(o)C \Rightarrow A(b, m, o)C$ (Allen의 전이성 테이블을 이용)
- 명제 3 $A(b, m)B \wedge B(o)C \Rightarrow A(b)C$ (Allen의 전이성 테이블을 이용)
- 명제 4 $A\{bi, mi, oi\} B \wedge B\{bi, mi\}C \Rightarrow A\{bi\}C$
 증명: $A\{bi, mi, oi\}B$ 는 A가 항상 B의 오른쪽에 있음을 나타내고 $B\{bi, mi\}C$ 는 B가 항상 C의 오른쪽에 있음을 의미한다. 그러므로 일차원 공간에서 A는 항상 C의 오른쪽에 있음을 알 수 있다. 게다가 $A\{mi\}C$ 라는 관계는 성립할 수 없다. 왜냐하면 A와 B가 (si)나 (e)의 관계가 없기 때문이다.
- 명제 5 $A\{bi, mi, oi\}B \wedge B\{bi, mi, oi\}C \Rightarrow A\{bi, mi, oi\}C$
 증명: $A\{bi, mi, oi\}B \wedge B\{bi, mi, oi\}C \Leftrightarrow (A\{bi, mi, oi\}B \wedge B\{bi, mi\}C) \vee (A\{bi, mi, oi\}B \wedge B\{oi\}C) \Rightarrow A\{bi\}C \vee (A\{bi, mi, oi\}B \wedge B\{oi\}C)$ (명제 4)
 $\Rightarrow A\{bi\}C \vee A\{bi, mi, oi\}C$ (Allen의 전이성 테이블)
 $\Rightarrow A\{bi, mi, oi\}C$
- 명제 6 $A\{b, m, o, d, s\}B \wedge B\{b, m\}C \Rightarrow A\{b\}C$
 증명: $A\{b, m, o, d, s\}B \wedge B\{b, m\}C \Leftrightarrow (A\{b, m, o\}B \wedge B\{b, m\}C) \vee (A\{d, s\}B \wedge B\{b, m\}C) \Rightarrow A\{b\}C \vee (A\{d, s\}B \wedge B\{b, m\}C)$ (역 속성과 명제 4)
 $\Rightarrow A\{b\}C \vee A\{b\}C$ (Allen의 전이성 테이블)
 $\Rightarrow A\{b\}C$
- 명제 7 $A\{b, m, o, d, s, f, fi, e\} B \wedge B\{b, m\}C \Rightarrow A\{b, m\}C$
 증명: $A\{b, m, o, d, s, f, fi, e\} B \wedge B\{b, m\}C \Leftrightarrow (A\{b, m, o, d, s\} B \wedge B\{b, m\}C) \vee (A\{f, fi, e\} B \wedge B\{b, m\}C) \Rightarrow A\{b\}C \vee (A\{f, fi, e\}B \wedge B\{b, m\}C)$ (명제 6)
 $\Rightarrow A\{b\}C \vee A\{b, m\}C$ (Allen의 전이성 테이블)
 $\Rightarrow A\{b, m\}C$
- 명제 8 $A\{b, m\}B \wedge B\{d, di, s, si, f, fi, o, oi, e\}C \Rightarrow A\{b, m, o, d, s\}C$
 증명: $A\{b, m\} B \wedge B\{d, di, s, si, f, fi, o, oi, e\}C$

$\Leftrightarrow (A\{b\}B \wedge B\{d, di, s, si, f, fi, o, oi, e\}C) \vee$
 $(A\{m\}B \wedge B\{d, di, s, si, f, fi, o, oi, e\}C)$
 $\Rightarrow (A\{b, m, o, d, s\}C \vee (A\{m\}B \wedge B\{d, di, s, si,$
 $f, fi, o, oi, e\}C))$ (**Allen의 전이성 테이블**)
 $\Rightarrow A\{b, m, o, d, s\}C \vee (A\{b, m, o, d, s\}C)$
(Allen의 전이성 테이블)

$\Rightarrow A\{b, m, o, d, s\}C$
• 명제 9 $A\{b, m\}B \wedge B\{d, di, s, si, f, fi, o, oi, m,$
 $mi, e\}C \Rightarrow A\{b, m, o, d, s, f, fi, e\}C$
증명: $A\{b, m\}B \wedge B\{d, di, s, si, f, fi, o, oi, m,$
 $mi, e\}C$
 $\Leftrightarrow (A\{b\}B \wedge B\{d, di, s, si, f, fi, o, oi, e\}C \vee$
 $(A\{b, m\}B \wedge B\{m, mi\}C))$
 $\Rightarrow A\{b, m, o, d, s\}C \vee (A\{b\}B \wedge B\{m, mi\}C) \vee$
 $(A\{m\}B \wedge B\{m, mi\}C)$
 $\Rightarrow A\{b, m, o, d, s\}C \vee (A\{b, m, o, d, s\}C) \vee$
 $(A\{f, fi, e\}C)$ (**Allen의 전이성 테이블**)
 $\Rightarrow A\{b, m, o, d, s\}C \vee A\{f, fi, e\}C$
 $\Rightarrow A\{b, m, o, d, s, f, fi, e\}C$

위에서 설명한 명제를 이용하여 10가지 추론 규칙을 증명하면 다음과 같다.

• 증명 1(규칙 1) $A_X\{e\}A_X$ 와 $A_Y\{e\}A_Y$ 는 정의 e로부터 $A\{e\}A$ 가 됨을 알 수 있고, overlap관계를 나타내는 o도 동일하다.

• 증명 2 (규칙 3) $A \text{ UP } B \Leftrightarrow A_X\{d, di, s, si, f, fi,$
 $e\}B_X \wedge A_Y\{b, m\}B_Y$
 $\Leftrightarrow B_X\{d, di, s, si, f, fi, e\}A_X \wedge A_Y\{b, m\}B_Y$
 $\Leftrightarrow B_X\{d, di, s, si, f, fi, e\}A_X \wedge A_Y\{bi, mi\}B_Y$
 $\Leftrightarrow B \text{ BL } A$

다른 방향성 관계들도 위의 증명과 동일하고, 혼합형 방향성 관계들도 다음과 같이 증명할 수 있다

$A \text{ RT } B \Leftrightarrow A_X\{bi, mi\}B_X \Leftrightarrow A_X\{b, m\}B_X \Leftrightarrow A \text{ LT } B$

• 증명 3 (규칙 4) $A \text{ LU } B \wedge B \text{ LU } C \Leftrightarrow [(A_X\{b,$
 $m\}B_X \wedge A_Y\{bi, mi, oi\}B_Y) \vee (A_X\{o\}B_X \wedge A_Y\{bi,$
 $mi\}B_Y)] \wedge [(B_X\{b, m\}C_X \wedge B_Y\{bi, mi, oi\}C_Y) \vee$
 $(B_X\{o\}C_X \wedge B_Y\{bi, mi\}C_Y)]$

규칙 4에 대한 증명은 다음의 4가지 경우를 생각해 볼 수 있다.

1. $A_X\{b,m\}B_X \wedge A_Y\{bi, mi, oi\}B_Y \wedge B_X\{b,m\}C_X$
 $\wedge B_Y\{bi, mi, oi\}C_Y$
 $\Leftrightarrow A_X\{b,m\}B_X \wedge B_X\{b,m\}C_X \wedge A_Y\{bi, mi, oi\}B_Y$
 $\wedge B_Y\{bi, mi, oi\}C_Y$
 $\Rightarrow A_X\{b\}C_X \wedge A_Y\{bi, mi, oi\}B_Y \wedge B_Y\{bi, mi,$
 $oi\}C_Y$ (**명제 6**)
 $\Rightarrow A_X\{b\}C_X \wedge A_Y\{bi, mi, oi\}C_Y$ (Lemma 5)
 $\Rightarrow A \text{ LU } C$

2. $A_X\{b,m\}B_X \wedge A_Y\{bi, mi, oi\}B_Y \wedge B_X\{o\}C_X \wedge$
 $B_Y\{bi, mi\}C_Y$
 $\Leftrightarrow A_X\{b,m\}B_X \wedge B_X\{o\}C_X \wedge A_Y\{bi, mi, oi\}B_Y \wedge$
 $B_Y\{bi, mi\}C_Y$
 $\Rightarrow A_X\{b\}C_X \wedge A_Y\{bi, mi, oi\}B_Y \wedge B_Y\{bi, mi\}C_Y$
(명제 3)

$\Rightarrow A_X\{b\}C_X \wedge A_X\{bi\}C_X$ (**명제 4**)
 $\Rightarrow A \text{ LU } C$

3. $A_X\{o\}B_X \wedge A_Y\{bi, mi\}B_Y \wedge B_X\{b, m\}C_X \wedge$
 $B_Y\{bi, mi, oi\}C_Y$

$\Leftrightarrow A_X\{o\}B_X \wedge B_X\{b, m\}C_X \wedge A_Y\{bi, mi\}B_Y \wedge$
 $B_Y\{bi, mi, oi\}C_Y$
 $\Rightarrow A_X\{o\}B_X \wedge B_X\{b, m\}C_X \wedge A_Y\{bi\}C_Y$ (**명제 4**)
 $\Rightarrow A_X\{b\}C_X \wedge A_Y\{bi\}C_Y$ (**명제 3**)
 $\Rightarrow A \text{ LU } C$

4. $A_X\{o\}B_X \wedge A_Y\{bi, mi\}B_Y \wedge B_X\{o\}C_X \wedge B_Y\{bi,$
 $mi\}C_Y$

$\Leftrightarrow A_X\{o\}B_X \wedge B_X\{o\}C_X \wedge A_Y\{bi, mi\}B_Y \wedge B_Y\{bi,$
 $mi\}C_Y$
 $\Rightarrow A_X\{o\}B_X \wedge B_X\{o\}C_X \wedge A_Y\{bi\}C_Y$ (**명제 4**)
 $\Rightarrow A_X\{b, m, o\}C_X \wedge A_Y\{bi\}C_Y$ (**명제 2**)
 $\Rightarrow A \text{ LU } C$

다른 혼합형 방향성 관계들도 위와 동일하게 증명 할 수 있다.

• 증명 4 (규칙 8) $A \text{ LT } B \wedge B \text{ OL } C \wedge C \text{ LT } D$
 $\Leftrightarrow A_X\{b, m\}B_X \wedge B_X\{d, di, s, si, f, fi, o, oi, e\}C_X$
 $\wedge B_Y\{d, di, s, si, f, fi, o, oi, e\}C_Y \wedge C_X\{b,$
 $m\}D_X$

$\Rightarrow A_X\{b, m\}B_X \wedge B_X\{d, di, s, si, f, fi, o, oi, e\}C_X$
 $\wedge C_X\{b, m\}D_X$ (**Y축 삭제**)
 $\Rightarrow A_X\{b, m, o, d, s\}C_X \wedge C_X\{b, m\}D_X$ (**명제 8**)
 $\Rightarrow A_X\{b\}D_X$ (**명제 6**)
 $\Rightarrow A \text{ LT } D$

다른 관계들도 위와 동일하게 증명 할 수 있다.

• 증명 5 (규칙 10) $A \text{ LT } B \wedge B\{LU, LL\}C \Rightarrow A \text{ LT } C$
 규칙 10에 대한 증명은 2가지 경우를 생각할 수 있다.

1. $A \text{ LT } B \wedge B \text{ LU } C$
 $\Leftrightarrow A_X\{b,m\}B_X \wedge [(B_X\{b,m\}C_X \wedge B_Y\{bi, mi, oi\}C_Y)$
 $\vee (B_X\{o\}C_X \wedge C_Y\{bi, mi\}C_Y)]$

$\Rightarrow A_X\{b,m\}B_X \wedge (B_X\{b,m\}C_X \vee B_X\{o\}C_X)$ (**Y축 삭제**)
 $\Rightarrow (A_X\{b,m\}B_X \wedge B_X\{b,m\}C_X) \vee (A_X\{b, m\}B_X \wedge$
 $B_X\{o\}C_X)$
 $\Rightarrow A_X\{b\}B_X \vee (A_X\{b, m\}B_X \wedge B_X\{o\}C_X)$ (**명제 6**)
 $\Rightarrow A_X\{b\}B_X \vee A_X\{b\}C_X$ (**명제 3**)
 $\Rightarrow A_X\{b\}C_X$
 $\Rightarrow A_X \text{ LT } C_X$

$$2. A_{LT} B \wedge B_{LL} C$$

$$\Leftrightarrow A_X(b,m)B_X \wedge [(B_X(b,m)C_X \wedge B_Y(b, m, o)C_Y) \vee (B_X(o)C_X \wedge B_Y(b, m)C_Y)]$$

$$\Rightarrow A_X(b,m)B_X \wedge [(B_X(b,m)C_X) \vee (B_X(o)C_X)] \text{ (Y 축 삭제)}$$

$$\Rightarrow A_X(b)C_X \text{ (규칙 10의 첫 번째 경우에서 증명)}$$

$$\Rightarrow A_X_{LT} C_X$$

규칙 2와 5-7까지의 규칙은 일반적이기 때문에 증명하는 과정을 생략하고, 또한 규칙 9는 규칙 10과 증명이 유사하기 때문에 본 부록에서는 그 과정을 다루지 않겠다.

A-2 공간관계 분석 알고리즘

```

Algorithm: Analyzing spatial relationship of objects
Input: two image objects (objects1, objects2)
Output: spatial relationship between objects
Iterator iterator1 = object1
Iterator iterator2 = object2
while(iterator1.hasNext())
/* 모든 오브젝트가 체크되지 않았음 */
begin
    object1 = iterator1.next()
    call spacialInfo(object, center)
    /* 중심점과 2차원 공간정보를 추출 */
    call spacial3DInfo(shape, center)
    /* Z축과 3차원 공간정보를 추출 */
while(iterator2.hasNext())
/* 남아있는 오브젝트를 검사 */
begin
    object2 =iterator2.next()
    if(object1==object2) then
    call spacialInfo(object1, object2)
/* 남아있는 오브젝트와의 2차원 공간정보를 검사 */
    call spacial3DInfo(object, center)

/* 남아있는 오브젝트와의 3차원 공간정보와
Z축의 좌표를 검사 */
end
end
    
```



황 인준

1988년 서울대학교 컴퓨터공학과(학사)
 1990년 서울대학교 컴퓨터공학과(석사)
 1998년 Univ. of Maryland at College Park 전산학과(박사). 1998년 6월~1998년 8월 Hughes Research Lab. 연구교수. 1998년 8월~1999년 8월 Bowie State Univ., Assistant Professor. 1999년 9월~2002년 2월 아주대학교 정보통신전문대학원 조교수. 2003년 3월~2004년 8월 아주대학교 정보통신전문대학원 부교수. 2004년 9월~현재 고려대학교 전자컴퓨터공학과 조교수. 관심분야는 데이터베이스, 멀티미디어 시스템, 정보 통합, 전자 상거래, XML 응용, 유비쿼터스 컴퓨팅



변 광준

1985년 서울대학교 컴퓨터공학 학사. 1987년 Pennsylvania of State University 전산학 석사. 1993년 University of Southern California 전산학 박사. 1994년~현재 아주대 정보및컴퓨터공학부 조교수, 부교수. 관심분야는 멀티미디어 데이터베이스, 분산 데이터베이스, 분산 객체 컴퓨팅



이수철

1998년 한남대학교 컴퓨터공학과(학사)
 1998년~2000년 아주대학교 정보통신전문대학원(석사). 2005년 아주대학교 정보통신 전문대학원(박사). 2005년 8월~현재 한국과학기술정보연구원 슈퍼컴퓨팅 센터 선임연구원. 관심분야는 데이터베이스, 멀티미디어 시스템, 정보 통합, XML 응용, 유비쿼터스 컴퓨팅