

위치 관계에 의한 영상 검색을 위한 질의 및 검색 기법

(Query Optimization Algorithm for Image Retrieval by Spatial Similarity)

조수진[†] 유석인^{**}
(Sue-Jin Cho) (Suk-In Yoo)

요약 영상의 시각적 특성을 이용하여 영상 데이터베이스를 검색하는 내용 기반 영상 검색 시스템에서 사용자가 직접 작성한 질의 영상에 존재하는 불완전성을 극복하기 위하여, 물체의 정확한 좌표값 대신 물체간의 위치 관계를 비교하는 기법이 많이 사용된다. 본 논문에서는 물체간의 8 방향 위치 관계 정보를 이용하여 영상을 검색하는 시스템을 위한 질의 변환 알고리즘을 제안한다. 제안된 알고리즘은 영상 내에 존재하는 물체들간의 위치 관계에 추이성(transitivity)이 존재하는 경우 정보가 중복된다는 사실로부터, 질의에 존재하는 추이성을 모두 제거함으로써 질의 영상을 최소 에지의 그래프로 변환한다. 제안된 알고리즘에 의해 생성된 프라임 에지 그래프는 동일한 위상 관계(topology)를 표현하는 그래프 중 최소 개수의 에지를 가지게 되므로 검색 중의 위치 관계 비교 회수를 최적화 할 수 있다. 실험 결과, 위치 관계의 추이성을 고려하지 않은 기존 알고리즘에 비해 평균 비교 회수를 크게 감소시켜 탐색 모듈의 효율을 향상 시킬 수 있다.

Abstract Content-based image retrieval system retrieves an image from a database using visual features. Among approaches to express visual aspects in queries, "query by sketch" is most convenient and expressive. However, every "query by sketch" system has the query imperfectness problem. Generally, the query image produced by a user is different from the intended target image. To overcome this problem, many image retrieval system use the spatial relationships of the objects, instead of pixel coordinates of the objects. In this paper, a query converting algorithm for an image retrieval system which uses the spatial relationship of every two objects as an image feature is proposed. The proposed algorithm converts the query image into a graph that has the minimum number of edges, by eliminating every transitive edge. Since each edge in the graph represents the spatial relationship of two objects, the elimination of unnecessary edges makes the retrieval process more efficient. Experimental results show that the proposed algorithm leads the smaller number of comparison in searching process as compared with other algorithms that do not guarantee the minimum number of edges.

1. 서론

내용 기반 영상 검색 시스템(Content-based image

retrieval system)이란 영상 내용(content)의 시각적 특성(visual features)을 이용하여 데이터베이스에 저장되어 있는 영상을 검색하는 시스템이다. 영상 검색 시스템은 검색 기법, 질의 형태 등에 따라 다양하게 분류할 수 있지만, 근본적으로는 영상의 어떠한 특성을 검색에 이용하는가에 따라 분류된다. 초기의 영상 검색 시스템은 물체의 색채 히스토그램, 질감 등 영상의 포괄적인 특성(global feature)을 검색에 이용하였다. 이러한 포괄적인 특성을 이용하는 경우, 데이터의 양이 적고 일차원적 백

· 이 논문은 2000년도 두뇌한국21사업에 의하여 지원되었음

† 비회원 : 서울대학교 컴퓨터공학과
sue@cs.snu.ac.kr

** 종신회원 : 서울대학교 컴퓨터공학과 교수
siyoo@hera.snu.ac.kr

논문접수 : 1999년 2월 3일
심사완료 : 2000년 2월 23일

터로 표현이 가능하므로 다양한 인덱싱(indexing) 기법을 이용하여 검색의 효율성을 높일 수 있었다. 그러나 영상의 포괄적인 특성을 표현하는 방식은 사용자가 영상을 인지하고 기억하는 형식과 다르므로 질의에 표현할 수 있는 방법은 제한되어, 이와 같은 시스템의 일반적인 질의 형태는 '예제에 의한 질의 (query by example)' 방식으로 한정된다[7,9,10]. 예제에 의한 질의 방식은 사용자가 질의 화면에 제시된 여러 보기 영상 중 검색하고자 하는 영상과 가장 유사한 특성을 갖고 있다고 생각되는 영상을 선택하면 시스템이 해당 예제 영상과 유사한 특성을 가진 영상을 데이터베이스에서 찾아 결과로서 보여 주는 방식이다. 일반적으로 예제 영상은 데이터베이스에 저장되어 있는 영상들 중 무작위로 추출된 보기 영상 중에서 고르게 되므로 질의 영상에 대한 별도의 특성 추출이 필요 없다는 장점이 있지만, 사용자 입장에서는 만족할 만한 예제 영상을 찾기 위해 보기 영상을 일차 검색해야 하는 불편함이 있다.

최근 다양한 분야에서의 디지털 이미지 사용이 확대됨에 따라 일반 사용자의 영상 데이터베이스 접근 역시 늘어나게 되어 점차 질의의 효율과 편리성에도 중점을 두는 시스템이 개발되었다. 사용자가 질의에 직접 표현하기 어려운 포괄적인 특성 외에 다양한 영상의 특성들을 검색에 이용하려는 시도가 있었고, 이에 따른 새로운 형태의 질의 방식인 '스케치에 의한 질의 (query by sketch)'가 제안되었다[1,5,6,12]. 스케치에 의한 질의 방식의 검색 시스템에서는 사용자가 직접 그리기 도구를 이용하여 검색하고자 하는 영상의 전체 또는 일부분을 질의 영상으로 작성하므로 영상 내의 각 물체의 색, 모양, 윤곽선 정보, 물체의 배치 등 다양한 특성들을 표현할 수 있게 되었다. 시스템에 따라 검색의 효율성을 고려하여 이 특성들 중 일부를 채택하여 사용하게 되는데, 검색의 효율성에는 검색 속도와 검색의 성공률 등이 모두 포함된다. 이 방식의 시스템이 가지는 문제점은, 사용자가 작성한 질의는 검색하고자 하는 목표 영상(target image)과 많은 차이를 가진 단순화되거나 왜곡된 영상일 가능성이 높다는 점이다. 특히 위치적 정보(positional information)와 관련된 특성들이나 색(color) 표현은 목표 영상과 직접 비교했을 경우, 상당한 차이를 보일 수 있다. 또, 질의 영상에 목표 영상의 일부분만이 표현될 수도 있다. 이러한 문제들을 해결하기 위해서는 질의 영상으로부터 특성을 추출하는 단계에서 사용자가 의도한 특성만을 판별해 낼 수 있어야 하며, 검색 중에도 이를 반영하여야 한다. 그 중, 사용자의 질의로부터 물체의 정확한 위치 대신 상대적 위치 관계만으로 위치

적 유사성을 판별하는 기법은 [2,3,4,8,11] 등에서 제안되었다. 이 기법들은 물체를 기호로 표현한 후, [3,11]에서는 잠정적인 격자 상에서의 위치 관계를 2D String을 사용해 표현하고, [4]에서는 그래프 형태로 표현하여 각 에지에 두 물체의 중심점을 잇는 직선의 경사를 가중치로서 표시한다. 이러한 방법은 인덱싱과 검색에 있어 효율적이지만 물체를 영상으로부터 인식하여 기호로 표시하는 작업은 고도의 영상 이해를 필요로 하므로 완전 자동화하기에는 어려움이 있다. 또한 질의에 있어서도 의미 있는 물체를 기호로 표시하므로 색상(color)과 같은 영상의 시각적 특성을 표현할 수 없다는 제약도 있다. 또한, 두 방식 모두 물체의 개수에 따라 위치 관계 비교회수는 제곱비로 증가하게 된다.

본 논문에서는 사용자가 그리기 도구를 사용하여 작성한 질의로부터 각 물체의 색과 크기, 물체간의 8 방향 위치 관계를 추출하여 목표 영상을 검색하는 시스템을 제안한다. 이 방식은 질의의 형태가 단순하여 작성이 간편하며 각 물체의 색과 크기 정보를 이용하여 유사성을 비교하고 서로간의 위치 관계의 일관성을 검사하므로, 질의 영상이 부분 정보만을 담고 있거나 구체적인 위치 정보가 정확히 입력되지 않더라도 성공적인 검색이 가능하다. 특히, 질의 변환 알고리즘을 통해 두 물체간의 위치 관계를 다른 물체들과의 위치 관계로부터 유추할 수 있는 경우 이런 불필요한 위치 관계에 대한 정보를 질의에서 추출된 특성으로부터 제외한다. 제안된 질의 변환 알고리즘은 그래프 형식으로 표현된 물체간의 위치 관계에 존재하는 추이성(transitivity)을 모두 삭제하여 질의를 최적화하므로 검색의 효율성을 더욱 높일 수 있다. 이에 의해 생성되는 프라임 에지 그래프는 최소 개수의 에지로 질의 영상에 존재하는 모든 물체 사이의 위치 관계를 표현할 수 있다.

논문의 구성은 다음과 같다. 2장에서는 제안된 검색 시스템의 전반적인 검색 원리를 설명하고, 3장에서는 질의 영상의 표현 형식인 프라임 에지 그래프를 정의하고 질의 변환 알고리즘을 설명한 후, 생성된 프라임 에지 그래프가 정의를 만족하며, 에지의 최소성을 보장함을 증명하고, 4장에서는 위치 관계에 의한 유사성 비교 기법을 설명하며, 5장에서는 제안된 알고리즘의 검색의 정확성과 질의 변환의 효과를 분석한 후, 6장에서는 결론과 함께 향후 연구 과제를 제시한다.

2. QBM (Query by Blob Map) 영상 검색 시스템

QBM은 스케치에 의한 질의 방식의 영상 검색 시스

템으로서 사용자는 색이 지정된 영역(colored blob)의 배치를 통해 질의 영상을 표현한다. QBM 시스템에서는 검색에 각 물체의 색과 크기, 물체간의 위치 관계만을 사용하므로, 데이터베이스에 저장되는 영상과 사용자의 질의 영상은 모두 물체(objects)의 집합으로 간주된다. 이와 유사한 표현 방법은 [3]에서 기호 그림(symbolic picture)이라는 개념으로 제안된 바 있지만, QBM 시스템에서의 물체(object)라는 개념은 [3]에서의 물체, 또는 현실 세계에서 말하는 물체와는 다른 의미로 사용된다. [3]에서는 집, 나무, 전화기 등과 같이 의미를 가진 사물을 물체로 보아 이를 기호로 표시하였으나 QBM에서는 사용자가 질의 영상을 작성할 때 한 가지 색으로 한 번에 표현할 수 있는 단위를 물체로 정의한다. 이러한 정의는 영상 분할에 의한 동질적인 영역(homogeneous region)에도 동일하게 적용되므로 데이터베이스에 입력되는 영상은 영상 처리를 거쳐 자동으로 물체의 집합으로 저장될 수 있다.

[정의 1] : 물체란 영상 내에서 같은 색을 가진 연결된 화소(pixel)들의 집합으로서 네 개의 원소로 이루어진 순서쌍(4-tuple) $O = \langle c, a, x, y \rangle$ 로 표현된다. 여기에서,
 c 는 물체의 색(color),
 a 는 물체의 면적(area),
 x 와 y 는 각각 물체 중심점(centroid)의 x 와 y 좌표를 나타낸다.

QBM 시스템은 질의 처리 모듈과 탐색 모듈, 영상의 인덱싱 모듈로 구성된다. 인덱싱 모듈은 데이터베이스에 저장하기 위해 입력된 영상으로부터 특성을 추출하여 구조화하고, 질의 처리 모듈은 사용자가 작성한 질의 영상으로부터 특성을 추출하고 이를 최적화하며, 탐색 모듈은 질의 처리 모듈에서 추출된 특성과 인덱싱 모듈에서 구조화된 데이터베이스 영상의 인덱스와의 비교를 통해 검색된 영상을 유사성 순위에 따라 보여준다.

2.1 질의 처리 모듈

질의 처리 모듈은 세부적으로 질의 인터페이스와 질의 변환기로 구성된다. 질의 인터페이스는 사용자의 질의 영상 작성을 지원하는 도구로서, 캔버스(canvas)와 팔레트(palette)를 제공하며, 사용자가 물체의 윤곽선을 그리면 지정된 색으로 내부를 채우는 동시에 물체의 면적과 중심점을 계산한다. 따라서 질의 영상은 분할 과정이 필요치 않으므로 물체의 특성을 쉽게 추출할 수 있다. 질의 변환 과정에서는 물체간의 위치 관계를 그래프

로 표현한다. 일반적으로 물체간의 위치 관계는 왼쪽, 오른쪽 등의 방향성에서 겹치거나 포함되는 등의 위상 관계까지 다양하게 기술할 수 있다. [8]에서는 물체간의 위치 관계를 총 169가지로 분류했지만 QBM에서는 8방향 위치 관계만을 고려한다. 사용자는 두 물체의 일부가 겹치거나(overlapping) 전체가 겹치는(containing) 관계보다는 좌우 또는 상하 관계만을 염두에 둔다고 가정했기 때문이다. 물체와 물체간의 위치 관계로 이루어지는 영상은 그래프로 표현할 수 있다. 각 물체는 그래프의 노드로 표현하고 두 물체간의 위치 관계는 에지로 표현한다. 위치 관계는 상대적인 개념이므로 모든 에지는 방향성을 가진다. 또한 각 에지에는 위치 관계의 정보를 포함하는 레이블을 지정한다. 레이블이 가질 수 있는 값은 8 방향 위치 관계만을 고려하므로 모두 9 가지이다. 정의 1에 따르면 모든 물체에는 중심점의 좌표에 대한 정보가 저장되어 있으므로 에지에 레이블을 지정하지 않아도 두 물체의 위치 관계를 쉽게 알 수는 있다. 그러나 에지에 레이블을 지정하는 이유에는 두 가지가 있다. 첫째로는 검색의 속도를 높이기 위해서이다. 위치 관계를 표현하는 목적은 결국 탐색 모듈에서의 비교를 위한 것인데, 에지에 레이블을 지정하지 않으면 위치 관계를 비교할 때마다 좌표를 비교해야만 한다. 이에 비해 레이블을 지정한 경우에는 단지 레이블 값만을 비교하면 되므로 속도를 높일 수 있다. 둘째는 질의 영상을 변환할 때 추이성(transitivity) 검사를 용이하게 하기 위해서이다. 이는 첫 번째의 이유와 같이 검색 효율을 높이기 위한 작업의 일환이기도 하다. 이에 대한 자세한 설명은 다음 장에서 다루게 된다. 질의 영상에서는 물체의 중심점 좌표는 사용하지 않으므로 에지에 레이블을 지정한 후 삭제한다. 결론적으로 질의 영상은 다음 정의와 같은 레이블 그래프(labeled graph)로 변환할 수 있다.

[정의 2] : 레이블 그래프(labeled graph)란 그래프 내의 각 에지에 레이블이 지정되어 있는 방향성 그래프(directed graph)로서 세 개의 원소로 이루어진 순서쌍(3-tuple) $G = \langle V, E, \lambda \rangle$ 로 표현된다. 여기에서,
 V 는 노드의 집합,
 $E \subseteq V \times V$ 는 에지의 집합,
 $\lambda: E \rightarrow L_E$ 는 각 에지에 레이블을 지정하는 함수,
 L_E 는 레이블이 가질 수 있는 값의 집합을 나타낸다.
 $L_E = \{0101, 0110, 0111, 1001, 1010, 1011, 1101, 1110, 1111\}$

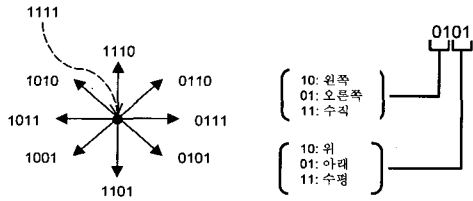


그림 1 에지 방향과 레이블

일반적으로 영상을 그래프로 변환하면 에지의 집합 E 는 $V \times V$ 와 같아진다. 모든 두 물체 사이의 위치 관계를 규정할 수 있기 때문이다. 그림 1은 9 가지의 에지 방향과 각각에 해당하는 레이블을 보여준다. 각 방향에 해당하는 레이블은 4 비트로 이루어진 코드로 표현된다. 이 코드의 앞 두 비트는 x 방향, 즉 가로 방향의 위치 관계를 표시하고 뒤의 두 비트는 y 방향, 즉 세로 방향의 위치 관계를 표시한다. 예를 들어 만일 물체 O_1 이 물체 O_2 보다 왼쪽 위에 있다면 O_1 에서 O_2 의 방향으로 이어진 에지의 레이블의 앞 두 비트는 '01'이 되고 뒤의 두 비트는 '01'이 되므로 레이블은 '0101'이다.

2.2 탐색 모듈

탐색 모듈에서는 질의 변환을 통해 생성된 그래프와 데이터베이스에 저장되어 있는 각 영상의 추출된 특성들과의 유사성(similarity score)이 계산된다. 유사성의 척도는 두 영상에서 짝을 이루는 물체간의 색의 차이와 면적의 차이, 그리고 위치 관계의 일치 여부이다. 최종적으로, 유사성 순위에서 상위를 차지한 영상들이 검색의 결과로서 제시된다.

2.3 인덱싱 모듈

QBM 시스템은 물체 단위로 질의를 처리하므로 인덱싱 모듈에서는 영상 분할(segmentation) 과정이 포함된다. 영상 분할이란 원래의 영상을 동질적인(homogeneous) 영역들로 나누는 과정으로 영상 분할의 기준은 정의 1에 따라 물체의 색이 된다. 그러나 실제로 영상에 표현된 모든 색을 다 사용하는 것은 아니고, 색의 양자화(color quantization)를 통해 각 화소(pixel)의 색을 대표색(representative color)으로 매핑(mapping)하여 사용한다. 즉, 하나의 물체를 구성하는 화소들의 색은 원래는 다를 수도 있지만 모두 같은 대표색으로 매핑된다는 공통점을 가진다. 분할된 각 물체의 색과 면적, 중심점의 좌표는 탐색을 위해 따로 저장된다. 유사성 비교 대상은 각 물체의 중심점 좌표가 아닌 두 물체간의 위치 관계이지만 각 물체의 중심점 좌표 대신 데이터베이스 영상에 존재하는 물체간의 위치 관계를 명

시적으로 표현하면 별도의 자료 구조가 필요할 뿐만 아니라 유사성 비교시 해당 물체들에 해당하는 위치 관계 정보를 다시 검색해야 하므로 각 물체에 저장된 중심점의 좌표만을 저장하는 방법이 더 효율적이다. 데이터베이스에 저장되는 영상을 이루는 물체의 정보는 색과 면적에 따라 정렬된다.

3. 프라임 에지 그래프

사용자가 작성한 질의에 나타나는 위치 정보의 부정확성을 검색에서 배제하기 위하여 물체의 중심점 좌표를 직접 이용하는 대신 위치 관계만을 검색의 대상으로 하는 기법이 제안되었다. 그러나 이 기법의 문제는 위치 관계가 물체의 개수에 따라 비약적으로 증가한다는 점이다. 질의 영상 내의 물체의 개수를 n 이라 하면 레이블 그래프의 에지의 개수, 즉 서로 다른 물체간의 위치 관계 비교 회수는 최대 $n(n-1)/2$ 회이다. 본 논문에서는 이 문제를 해결하기 위하여 프라임 에지 그래프라는 영상 데이터의 구조를 제안한다. 프라임 에지 그래프는 일직선상에 놓인 물체들의 위치 관계는 서로 인접한 두 물체들간의 위치 관계만 일치해도 모두 일치한다는 사실로부터 위치 관계 비교의 효율성을 높이기 위해 제안된 개념이다. 물체 사이의 위치 관계에는 추이성(transitivity)이 존재할 수 있다. 그림 2와 같이 물체 O_1 이 물체 O_2 의 왼쪽 수평 방향에 있고, 물체 O_2 는 물체 O_3 의 왼쪽 수평 방향에 있다면 물체 O_1 이 물체 O_3

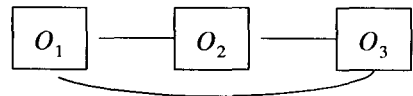


그림 2 에지의 추이성

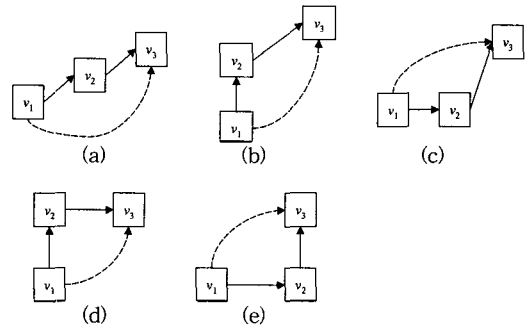


그림 3 추이 에지의 예

의 왼쪽 수평 방향에 있다는 사실은 달리 명시하지 않아도 자명하다. 이를 레이블 그래프로 바꾸어 말한다면 O_2 에서 O_1 으로 연결하는 에지의 레이블이 1011이고 O_3 에서 O_2 로 연결하는 에지의 레이블도 1011이라면 O_3 에서 O_1 으로 연결하는 에지의 레이블은 따로 참조하지 않아도 1011이라는 것을 알 수 있다. 이러한 성질을 추이성이라고 정의한다. 추이성을 가지는 에지, 즉 추이 에지(transitive edge)의 정의는 다음과 같다.

[정의 3] v_1, v_2, v_3 가 서로 다른 노드이고 $e_i = \langle v_1, v_2 \rangle$ 와 $e_j = \langle v_2, v_3 \rangle$ 가 다음 조건 중 한 가지를 만족하는 경우, $e_k = \langle v_1, v_3 \rangle$ 를 추이 에지(transitive edge)라고 한다.

- (1) e_i 와 e_j 의 방향이 같다.
- (2) e_i 또는 e_j 가 수직방향이고 e_i 와 e_j 의 상하 방향이 같다.
- (3) e_i 또는 e_j 가 수평방향이고 e_i 와 e_j 의 좌우 방향이 같다.
- (4) e_i 가 수직방향이고 e_j 가 수평방향이다.
- (5) e_i 가 수평방향이고 e_j 가 수직방향이다.

그림 3은 정의 3의 조건을 만족하는 예로서, 점선으로 표시된 에지가 추이 에지이다. 정의 3의 에지의 추이성은 레이블 사이의 bitwise-AND 연산을 통해 쉽게 판별할 수 있다.

[정리 1] v_1, v_2, v_3 가 중심점이 일치하지 않는 서로 다른 노드이고, 에지 e_i, e_j, e_k 가 각각 $e_i = \langle v_1, v_2 \rangle$, $e_j = \langle v_2, v_3 \rangle$, $e_k = \langle v_1, v_3 \rangle$ 라 할 때, $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과가 $\lambda(e_k)$ 와 같다면 에지 e_k 는 추이 에지(transitive edge)이다. 역으로, e_k 가 추이 에지라면 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과는 $\lambda(e_k)$ 와 같다.

증명 $\lambda(e_i) = i_1i_2i_3i_4$, $\lambda(e_j) = j_1j_2j_3j_4$, $\lambda(e_k) = k_1k_2k_3k_4$ 라 하자.

(\Rightarrow)

$\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과가 $\lambda(e_k)$ 와 같다면,

$i_1 \cdot j_1 = k_1, i_2 \cdot j_2 = k_2, i_3 \cdot j_3 = k_3, i_4 \cdot j_4 = k_4$ 이다. (\cdot 은 AND 연산이라 하자.)

$k_1k_2k_3k_4 \in L_E$ 이고 v_1 과 v_3 의 중심점이 일치하지 않으므로, $k_1k_2k_3k_4$ 가 가질 수 있는 값은 모두 8가지이다.

표 1의 (a), (b), (c), (d), (e), (f), (g), (h)는 각각 $\lambda(e_k)$ 가 가질 수 있는 값인 0101, 0110, 1001, 1010,

0111, 1011, 1101, 1110에 대해, $\lambda(e_i)$ 와 $\lambda(e_j)$ 가 가질 수 있는 값의 가능한 조합을 모두 보여준다. 모든 경우 정의 3의 조건 중 하나를 만족함을 알 수 있다. $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과가 $\lambda(e_k)$ 와 같으면 언제나 정의 3의 조건 중 하나를 만족하므로 e_k 는 추이 에지이다.

(\Leftarrow)

역을 증명하기 위해 에지 e_k 가 추이 에지라고 가정하면, e_i 와 e_j 는 정의 3의 조건 중 하나를 만족한다.

i) 조건 (1)을 만족하는 경우

e_i 와 e_j 의 방향이 같으면 v_1 의 x 좌표와 v_2 의 x 좌표, v_2 의 x 좌표와 v_3 의 x 좌표 사이의 부등호가 같을 것이므로 부등호의 추이성(transitivity)에 의해 v_1 의 x 좌표와 v_3 의 x 좌표 사이의 부등호도 같을 것이고, y 좌표 또한 마찬가지이므로 e_k 의 방향은 e_i 와 e_j 의 방향과 같다. 즉, $\lambda(e_i) = \lambda(e_j) = \lambda(e_k)$ 이다. $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 네 비트가 모두 같으므로 bitwise-AND 연산 결과는 $\lambda(e_i)$, $\lambda(e_j)$ 와 같고, 따라서 $\lambda(e_k)$ 와도 같다.

ii) 조건 (2)를 만족하는 경우

e_i 와 e_j 의 상하 방향이 같으므로 i)과 같은 이유로 e_k 의 상하 방향은 e_i 와 e_j 의 상하 방향과 같다. 즉, $\lambda(e_k)$ 의 뒤의 두 비트는 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 뒤쪽 두 비트와 같다. 만일 e_i 가 수직 방향이라면 v_1 의 x 좌표와 v_2 의 x 좌표가 같으므로 v_1 의 x 좌표와 v_3 의 x 좌표 사이의 부등호는 v_2 의 x 좌표와 v_3 의 x 좌표 사이의 부등호와 같게 된다. 즉, $\lambda(e_k)$ 의 앞 두 비트는 $\lambda(e_j)$ 의 앞 두 비트와 같다. $\lambda(e_i)$ 의 앞 두 비트는 '11'이므로 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과의 앞 두 비트는 $\lambda(e_j)$ 의 앞 두 비트와 같아져 $\lambda(e_k)$ 의 앞 두 비트와도 같고, $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과의 뒤의 두 비트는 $\lambda(e_i)$, $\lambda(e_j)$ 의 뒤쪽 두 비트와 같으므로 $\lambda(e_k)$ 의 뒤쪽 두 비트와도 같다. e_j 가 수직 방향인 경우에도 마찬가지이다. 따라서 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과는 $\lambda(e_k)$ 와 같다.

iii) 조건 (3)을 만족하는 경우

ii)와 유사하게 증명 가능하다.

iv) 조건 (4)를 만족하는 경우

e_i 가 수직 방향이므로 ii)에서와 마찬가지로 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과의 앞 두 비트는 $\lambda(e_k)$ 의 앞 두 비트와 같고 e_j 가 수평 방향이므로 비슷한 과정을 거치면 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과의 뒤쪽 두 비트는 $\lambda(e_k)$ 의 뒤쪽 두 비트와 같다는 것을 알 수 있다. 따라서 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과는 $\lambda(e_k)$ 와 같다.

표 1 $\lambda(e_k)$ 가 각각 (a) 0101, (b) 0110, (c) 1001, (d) 1010, (e) 0111, (f) 1011, (g) 1101, (h) 1110인 경우의 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 가능한 값의 조합과 정의 3에서의 해당 조건

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
0101	0101	(1)
0101	0111	(3)
0101	1101	(2)
0111	0101	(3)
0111	1101	(5)
1101	0101	(2)
1101	0111	(4)

(a) $\lambda(e_k) = 0101$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
1001	1001	(1)
1001	1101	(2)
1001	1011	(3)
1011	1001	(3)
1011	1101	(5)
1101	1001	(2)
1101	1011	(4)

(c) $\lambda(e_k) = 1001$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
0111	0111	(1)

(e) $\lambda(e_k) = 0111$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
1101	1101	(1)

(g) $\lambda(e_k) = 1101$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
0110	0110	(1)
0110	0111	(3)
0110	1110	(2)
0111	0110	(3)
0111	1110	(5)
1110	0110	(2)
1110	0111	(4)

(b) $\lambda(e_k) = 0110$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
1010	1010	(1)
1010	1011	(3)
1010	1110	(2)
1011	1010	(3)
1011	1110	(5)
1110	1010	(2)
1110	1011	(4)

(d) $\lambda(e_k) = 1010$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
1011	1011	(1)

(f) $\lambda(e_k) = 1011$

$\lambda(e_i)$	$\lambda(e_j)$	만족하는 조건
1110	1110	(1)

(h) $\lambda(e_k) = 1110$

v) 조건 (5)를 만족하는 경우

iv)와 유사하게 증명 가능하다.

따라서 e_k 가 추이 에지라면 $\lambda(e_i)$ 와 $\lambda(e_j)$ 의 bitwise-AND 연산 결과는 $\lambda(e_k)$ 와 같다.■

추이 에지의 레이블은 다른 두 에지의 bitwise-AND 연산 결과와 같으므로 추이 에지가 연결하고 있는 두 노드의 위치 관계는 다른 에지들의 레이블만으로도 알 수 있다. 다시 말해, 영상을 변환한 레이블 그래프에서 추이 에지를 모두 삭제해도 임의의 두 노드 사이의 위치 관계 정보는 여전히 남아 있게 된다. 추이 에지 외에도 세 가지 불필요한 에지가 존재할 수 있다. 첫째로, 모든 루프(loop)는 유효한 정보를 담고 있지 않고, 둘째로, 두 노드 사이에 방향만 다른 대칭(symmetric) 에지가 존재할 경우, 그 중 하나만으로도 위치 관계를 알 수 있으므로 다른 하나는 불필요하다. 셋째로, 서로 다른

두 개 이상의 노드들의 중심점의 x, y 좌표가 모두 같은 경우, 이 노드들 서로간을 잇는 에지를 제외한 다른 에지들은 모두 중복되므로 그 중 하나씩만 남겨 놓아도 무방하다. 그림 4에서 에지 $\langle v_1, v_3 \rangle$ 와 $\langle v_2, v_3 \rangle$ 는 중복된 위치 관계를 표시하므로 이 중 하나는 불필요하다.

이렇게 불필요한 에지들을 모두 제외하고 남은 에지를 프라임 에지라 한다. 선택의 가능성이 여러 가지이므로 프라임 에지의 집합이 유일하게 결정되는 것은 아니지만 결국 하나의 위상 구조(topological structure)를 표현하게 된다. 이와 같이 프라임 에지로만 이루어진 그래프를 프라임 에지 그래프라 한다.

[정의 4] 영상 I 를 변환한 레이블 그래프 $G = \langle V, E, \lambda \rangle$ 의 부분 그래프(subgraph)인 $G_p = \langle V, E_p, \lambda \rangle$ 가 다음과 같은 조건을 만족하면 G_p 를 영상 I 의 프라임 에지 그래프라 한다.

- (1) $E_p \subseteq E$
- (2) $e \in E_p$ 인 모든 에지 e 는 추이 에지가 아니다.
- (3) 모든 $v \in V$ 에 대해 $\langle v, v \rangle \notin E_p$ 이다.
- (4) $\langle v, u \rangle \in E_p$ 이면 $\langle u, v \rangle \notin E_p$ 이다.
- (5) $\lambda(\langle u, v \rangle)$ 가 1111이라면 모든 $w \in V$ 에 대해, $((\langle u, w \rangle \in E_p \rightarrow \langle v, w \rangle \notin E_p) \wedge (\langle v, w \rangle \in E_p \rightarrow \langle u, w \rangle \notin E_p)) \wedge (\langle w, u \rangle \in E_p \rightarrow \langle w, v \rangle \notin E_p) \wedge (\langle w, v \rangle \in E_p \rightarrow \langle w, u \rangle \notin E_p)$ 이다.
- (6) 서로 다른 노드 u, v 에 대해 에지 $\langle u, v \rangle$ 가 추이 에지가 아니고, $\lambda(\langle u, w \rangle)$ 가 1111인 w 가 존재하지 않으면 $\langle u, v \rangle \in E_p$ 또는 $\langle v, u \rangle \in E_p$ 이다.

프라이미 에지 그래프란 정의 4의 조건 (1)에서와 같이 영상을 변환한 레이블 그래프의 부분 그래프(subgraph)이다. 조건 (2)의 의미는 프라이미 에지 그래프에는 추이 에지가 하나도 포함되어 있지 않다는 것이고 조건 (3), (4), (5)는 앞에서 설명한 불필요한 에지들을 배제하기 위한 조건이고 조건 (6)은 영상을 변환한 레이블 그래프의 프라이미 에지들은 모두 프라이미 에지 그래프에 포함된다는 의미이다. 조건 (6)에 의해 임의의 두 물체 사이의 위치 관계를 알아낼 수 있는 정보가 보존된다.

본 논문에서 제안하는, 질의 영상으로부터 프라이미 에지 그래프를 생성하는 절차는 다음과 같다.

[질의 변환 알고리즘]

단계 1 (초기 그래프의 생성)

질의 영상 Q 로부터 방향성 그래프 $G_q = \langle V, E \rangle$ 를 생성한다. 이 그래프에서 V 는 질의 영상 Q 에 포함된 모든 물체들의 집합이고 E 는 V 의 서로 다른 두 물체 사이를 연결하는 방향성 에지들의 집합이다.

단계 2 (에지에 레이블 지정)

그림 1의 규칙에 따라 각 에지의 방향에 해당하는 레이블을 지정한다.

단계 3 (추이 에지 표시)

서로 다른 두 에지의 각 쌍에 대해 정의 3의 조건에 해당되는 항목이 있는지 검사하여 추이 에지로 판명되는 에지는 표시를 해 놓는다.

단계 4 (추이 에지 삭제)

단계 3에서 추이 에지로 표시된 에지들을 그래프에서 삭제한다.

단계 5 (대칭 에지 삭제)

단계 4의 결과로 남은 그래프에서 각 에지는 관련된 노드는 같으나 방향만 다른 대칭 에지의 쌍을 갖게 된다. 대칭 에지는 단계 3에서 필요하므로 초기 그래프에

는 반드시 모두 포함되어야 하지만 추이 에지를 모두 삭제한 이후에는 필요 없으므로 이 중 하나의 에지만 남기고 나머지 하나는 삭제한다. 즉, $V = \{O_i \mid 1 \leq i \leq n\}$ 이라면 $i > j$ 인 경우 에지 $\langle O_i, O_j \rangle$ 는 삭제한다.

단계 6 (중복 에지 삭제)

단계 5의 결과로 남은 그래프에서, 중심점이 일치하는 노드가 존재한다면 그로 인해 중복된 에지 중 하나만 남기고 모두 삭제한다. 즉, $i < j$ 인 경우 $\lambda(\langle O_i, O_j \rangle)$ 가 1111이라면 O_j 와 연결된 에지들은 모두 삭제한다.

위의 알고리즘에 따라 생성된 그래프는 프라이미 에지 그래프의 정의를 만족한다.

[정리 2] 제안된 알고리즘에 의해 생성되는 그래프는 정의 4를 만족한다.

증명: 영상 I 를 변환한 레이블 그래프 $G = \langle V, E, \lambda \rangle$ 를 위의 절차에 따라 변환하여 생성되는 그래프를 $G_p = \langle V, E_p, \lambda \rangle$ 라 하자. 변환 알고리즘에서는 에지가 삭제만 될 뿐, 추가되지는 않으므로 $E_p \subseteq E$ 를 만족한다는 사실은 쉽게 알 수 있다. 단계 3에서 추이 에지로 표시된 에지들은 단계 4에서 모두 삭제되므로 E_p 에는 추이 에지가 포함되지 않고, 나머지 에지들은 단계 3에서 추이 에지로 표시되지 않으므로 모두 E_p 의 원소로 남는다. 단계 5에 의해 대칭 에지 중 하나는 삭제되고 단계 6에서는 중복된 에지들이 하나만 남기고 모두 삭제된다. 따라서 그래프 G_p 는 정의 4의 일곱 가지 조건을 모두 만족한다. ■

이러한 절차에 따라 생성되는 그래프는 위치 관계를 보존하는 최소의 에지 개수를 가지며 대칭 에지의 선택을 제외하면 유일하다. 생성된 그래프의 에지 개수가 최소라는 사실을 증명하기 위해 다음과 같이 유사 방향성 경로(pseudo-directional path)라는 개념을 정의한다.

[정의 5] 노드 v_1 에서 v_n 까지의 유사 방향성 경로(pseudo-directional path)란 다음과 같은 조건을 만족하는 경로 $P = v_1, v_2, \dots, v_n$ 을 말한다.

- (1) 경로 P 에서 각 노드는 한 번씩만 나타난다. 즉, $i \neq j$ 이면 $v_i \neq v_j$ 이다.
- (2) $1 \leq i < n$ 에 대해 v_i 와 v_{i+1} 사이에 에지 $\langle v_i, v_{i+1} \rangle$ 또는 $\langle v_{i+1}, v_i \rangle$ 가 존재한다.
- (3) $i < j < k$ 인 임의의 세 노드 v_i, v_j, v_k 에 대해 에지 $\langle v_i, v_j \rangle$ 와 $\langle v_j, v_k \rangle$ 의 레이블의 bitwise-AND 연산 결과가 에지 $\langle v_i, v_k \rangle$ 의 레이블과 같다. 이 에

지들은 프라임 에지 그래프에서는 이미 삭제되어 존재하지 않을 수도 있지만 레이블은 알 수 있다.

임의의 두 노드 사이에 유사 방향성 경로가 존재한다면 조건 (3)에 의해 그 사이의 위치 관계를 알 수 있으므로 원래의 그래프의 위치 관계를 보존한다고 말할 수 있다. 제안된 변환 알고리즘은 발견된 추이 에지와 대칭 에지를 삭제하는 방식이므로 이 절차에 따라 생성되는 프라임 에지 그래프에서는 임의의 두 노드 사이에 유사 방향성 경로가 존재한다.

[정리 3] 제안된 알고리즘에 의해 생성되는 그래프는 대칭 에지와 중심점 일치에 의해 중복된 에지의 선택을 제외하면 유일하다.

증명. 만일 제안된 알고리즘에 의해 두 개의 그래프 $G = \langle V, E \rangle$ 와 $G' = \langle V, E' \rangle$ 이 생성되고 $E \neq E'$ 이라면 E 에는 포함되고 E' 에는 포함되지 않는 에지 e 가 존재한다. 정리 2에 의해 G 는 프라임 에지 그래프이고 $e \in E$ 이므로 정의 4에 의하면 에지 e 는 프라임 에지이다. G' 역시 프라임 에지 그래프이고 프라임 에지인 e 는 E' 에 포함되어야 하므로 이는 모순이다. 따라서 $E=E'$ 이고 $G=G'$ 이다. 즉, 제안된 알고리즘에 의해 생성되는 그래프는 유일하다. ■

[보조정리 1] 제안된 알고리즘에 의해 생성되는 프라임 에지 그래프에서, 두 노드 사이에 에지가 존재하면 길이 2 이상의 유사 방향성 경로가 존재하지 않는다.

증명. 생성된 프라임 에지 그래프를 $G_p = \langle V, E_p, \lambda \rangle$ 라 하고 v_i, v_j 를 V 에 속하는 서로 다른 두 노드라 하자. $\langle v_i, v_j \rangle \in E_p$ 라면 정리 2와 정의 4에 의해 $\langle v_i, v_j \rangle$ 는 프라임 에지이다. 만일 v_i 에서 v_j 까지의 길이 2 이상의 유사 방향성 경로가 존재한다면 이 경로가 지나거나 또 다른 노드 v_k 가 존재할 것이다. 그러나 정의 5에 의하면 $\langle v_i, v_k \rangle$ 와 $\langle v_k, v_j \rangle$ 의 레이블의 bitwise-AND 연산 결과가 $\langle v_i, v_j \rangle$ 의 레이블과 같으므로 이는 $\langle v_i, v_j \rangle$ 가 프라임 에지라는 전제에 위배된다. 따라서 두 노드 사이에 에지가 존재하면 이 두 노드 사이에는 길이 2 이상의 유사 방향성 경로가 존재하지 않는다. ■

[정리 4] 제안된 알고리즘에 의해 생성되는 프라임 에지 그래프는 레이블 그래프의 위치 관계를 보존하는 부분 그래프 중 최소 개수의 에지를 가진다.

증명. 제안된 알고리즘에 의해 생성된 프라임 에지 그래프를 $G = \langle V, E \rangle$ 라 하자. 그리고 G 보다 적은 수의 에지를 가진 다른 부분 그래프가 존재한다고 가정하고 이 그래프를 $G' = \langle V, E' \rangle$ 이라 하자. E 의 에지의 개수가 E' 의 에지의 개수보다 크므로 E 에는 포함되고, E' 에는 포함되지 않는 에지 $e = \langle v_i, v_j \rangle$ 가 존재한다. G' 도 위치 관계를 보존하는 그래프이므로 모든 두 노드 사이에는 유사 방향성 경로가 존재한다. v_i 와 v_j 를 연결하는 에지는 E' 에 포함되지 않으므로 G' 에서 v_i 에서 v_j 까지의 유사 방향성 경로의 길이는 2 이상이다. 즉, 이 경로가 지나거나 또 다른 노드 v_k 가 V 에 존재한다. 유사 방향성 경로의 정의에 의해 $\lambda(\langle v_i, v_k \rangle)$ 와 $\lambda(\langle v_k, v_j \rangle)$ 의 bitwise-AND 연산 결과는 $\lambda(\langle v_i, v_j \rangle)$ 와 같으므로 정리 1에 의해 에지 e 는 원래의 레이블 그래프에서 추이 에지이다. 정의 4에 의하면 프라임 에지 그래프의 에지 집합에는 추이 에지가 포함될 수 없으므로 이는 G 가 프라임 에지 그래프라는 전제에 위배된다. 따라서 제안된 알고리즘에 의해 생성되는 프라임 에지 그래프보다 적은 수의 에지를 가지면서 위치 관계를 보존하는 부분 그래프는 존재하지 않는다. ■

4. 유사성 비교

두 영상의 유사성은 크게 물체의 유사성과 위치 관계의 유사성의 합으로 계산된다. 물체의 유사성은 추출된 색과 크기 정보에 의해 간단히 계산할 수 있으며, 질의 영상의 각 물체와 가장 유사성 높은 물체를 데이터베이스의 영상에서 찾아 서로간의 위치 관계를 비교한다. 위치 관계의 유사성은 레이블이 일치하는 에지의 개수에 의해 계산할 수 있는데, 삭제된 추이 에지를 감안하여 각 에지의 가중치를 달리한다. 에지 가중치 할당의 원리는 삭제될 추이 에지 e 의 가중치 $w(e)$ 가 관련된 두 에지에 동일하게 분산된다는 것이다. 즉, 초기에는 각 에지에 균일한 가중치가 할당되지만, 질의 변환 알고리즘에서 에지가 삭제될 때마다 삭제된 에지의 가중치는 관련 에지에 균일하게 분배된다. 추이 에지의 판별에 관련된 에지도 역시 추이 에지일 수 있으므로 가중치 할당 과정은 재귀적으로(recursively) 처리된다. 그림 4는 가중치 할당 과정의 예를 보여준다. 각 에지의 가중치는 프라임 에지 생성 단계에서 계산되어 에지 레이블과 함께 저장된다.

[유사성 계산 알고리즘]
단계 1 (초기화)

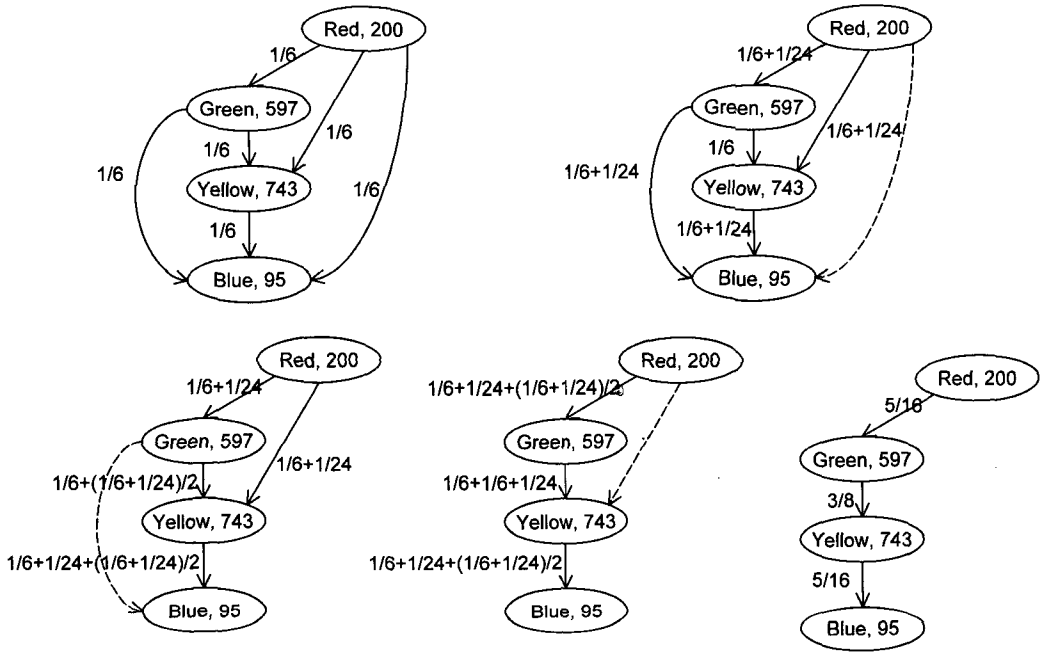


그림 4 가중치 할당 과정의 예

$SP \leftarrow 0.0$

$SO \leftarrow 0.0$

단계 2 (물체의 매칭과 유사성 계산)

질의 영상의 각 물체와 데이터베이스 영상의 물체를 비교하여 물체 Q_i 와의 유사성 값이 가장 큰 데이터베이스 영상의 물체 D_j 를 찾는다. 질의 영상의 물체 Q 와 데이터베이스 영상의 물체 D 의 유사성 $S(Q, D)$ 는 다음과 같이 계산된다.

$$S(Q, D) = w_a * (1 - |a_q - a_d|) + (50 - w_a) * \delta(c_q, c_d)$$

여기에서,

a_q, a_d : Q, D 의 정규화된 크기 ($0 \leq a_q, a_d \leq 1$),

c_q, c_d : Q, D 의 색 인덱스,

w_a : 유사성 계산에서 물체의 크기에 주어지는 가중치 ($0 \leq w_a \leq 50.00$),

$$\delta(c_q, c_d): \begin{cases} 1 & (c_q = c_d) \\ 0 & (c_q \neq c_d) \end{cases} \text{이다.}$$

Q_i 와 D_j 의 유사성을 SO 에 더한다.

$$SO \leftarrow SO + S(Q_i, D_j)$$

질의 영상의 모든 물체에 대해 이 과정을 반복한다.

단계 3 (위치 관계의 유사성 계산)

질의 영상을 변환한 그래프의 각 에지 $\langle Q_i, Q_j \rangle$ 와 해

당되는 데이터베이스 영상의 에지 $\langle D_i, D_j \rangle$ 에 대하여,

$\lambda(\langle D_i, D_j \rangle)$ 와 $\lambda(\langle Q_i, Q_j \rangle)$ 가 같다면,

$$SP \leftarrow SP + 50.00 * \omega(\langle Q_i, Q_j \rangle)$$

모든 에지에 대해 이 과정을 반복한다.

$\lambda(\langle D_i, D_j \rangle)$ 는 데이터베이스에 명시적으로 표현되지 않지만 D_i 와 D_j 의 x, y 축 좌표값을 단순 비교하여 $\lambda(\langle Q_i, Q_j \rangle)$ 와 같은지 확인할 수 있다.

단계 4 (영상의 유사성 계산)

최종적으로 두 영상의 유사성은 각각 단계 2와 단계 3에서 계산한 물체의 유사성 SO 와 위치 관계의 유사성 SP 의 합으로 계산된다.

$Similarity \leftarrow \frac{1}{n} * SO + SP$ (n : 질의 영상에 포함된 물체의 개수)

SO 는 0에서 $n * 50.00$ 사이의 값을 갖게 되며, 프라임 에지 그래프에서 에지의 가중치 $\omega(e)$ 를 모두 더하면 1 이므로 SP 는 0에서 50.00 사이의 값을 갖게 되어 영상의 유사성은 0에서 100.00 사이의 값이 된다. 유사성 값이 100.00이면 두 영상의 모든 매치된 물체의 색과 크기가 같고 물체간의 위치 관계도 모두 일치하는 경우이고 유사성 값이 작을수록 유사성이 낮은 영상이다.

5. 실험 결과

그림 5는 QBM 시스템의 실제 실행 예를 보여준다. 실행 윈도우의 왼편에는 질의 인터페이스가 제공된다. 사용자가 왼편의 사각 영역, 즉 캔버스에 질의 영상을 그리면 시스템은 이 질의 영상으로부터 특성을 추출하여 데이터베이스에 저장되어 있는 영상들과 차례로 비교하여 유사성 순위가 높은 영상들을 오른편의 결과 윈도우에 표시한다.

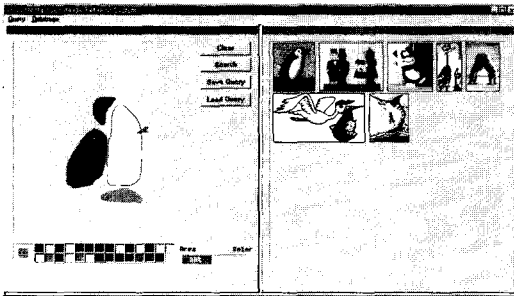


그림 5 QBM 시스템의 실행 예

실험을 위해 일반 사용자로부터 총 100개의 질의 영상을 입력받았다. 위치 관계에 의한 검색의 정확성, 특히 목표 영상의 일부분만 표현된 질의와 전체적으로 이동된 질의, 일부만이 이동된 질의에 대한 검색 효율을 평가한다. 또한, 질의 변환 모듈을 두 가지로 달리 구현하여 위치 관계를 나타내는 에지 개수의 비교를 통해 질의 변환 알고리즘의 효과를 알아본다.

5.1 검색의 효율

QBM에서 사용하는 영상의 특성과 질의 영상의 형태는 기존의 기법과 차이가 있으므로 단순한 성능 비교는 불가능하지만, 유사성 계산 알고리즘을 두 가지로 다르게 구현하여 불완전한 질의에 대한 성능의 개선을 확인해 본다. 첫 번째로, 부분 질의에 대한 성능을 실험하기 위하여 유사성 계산 알고리즘의 단계 4를 다음과 같이 수정한다.

단계 4 (영상의 유사성 계산)

$$Similarity1 \leftarrow \frac{1}{N} * SO + SP \quad (N : \text{데이터베이스 영상에 포함된 물체의 개수})$$

두 번째 알고리즘에서는 위치 관계의 유사성 SP를 계산하지 않고, 물체의 유사성에 물체의 중심점 위치 비교를 포함시킨다. 따라서, 질의의 프라임 에지 그래프로

의 변환 과정은 생략되고, 물체의 중심점 위치는 삭제되지 않는다. 유사성 알고리즘의 단계 2 이하를 다음과 같이 수정한다.

단계 2 (물체의 매칭과 유사성 계산)

$$S(Q, D) = w_s * (1 - |a_q - a_d|) + (50 - w_s) * \delta(c_q, c_d) + 25 * (1 - |x_q - x_d|) + 25 * (1 - |y_q - y_d|) \quad (0 \leq S(Q, D) \leq 100.0)$$

$SO \leftarrow SO + S(Q, D)$

단계 3 (영상의 유사성 계산)

$$Similarity2 \leftarrow \frac{1}{n} * SO$$

두 알고리즘과 제안된 알고리즘을 적용하여 실험한 결과는 그림 6과 같다. 그림에서 y축은 목표 영상을 x축의 값에 해당하는 순위 안에 결과로 제시한 질의의 개수를 의미한다.

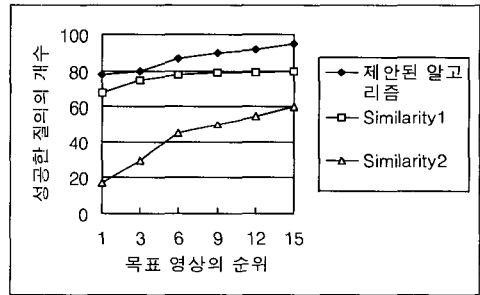


그림 6 세 가지 유사성 계산 알고리즘에 의한 검색의 성공률

5.2 질의 변환의 효과

질의 변환으로 위치 관계 비교의 대상이 되는 에지 개수의 감소율, 즉 매칭 회수의 감소율을 실험으로 알아보기 위해 [4]와 같이 질의 영상을 변환한 레이블 그래프에서 위치 관계의 추이성을 전혀 고려하지 않고 대칭 에지와 루프만을 삭제하는 질의 변환 알고리즘을 적용하여 제안된 기법과 비교한다. 질의 영상에 존재하는 물체의 개수를 n이라 하면 이 그래프의 에지의 개수는 항상 $n(n-1)/2$ 가 된다. 제안된 알고리즘에 의한 프라임 에지 그래프는 원래의 레이블 그래프의 위치 관계를 보존하는 서브 그래프 중에서는 최소의 에지를 가진다. 부분 질의를 허용하는 QBM 시스템의 질의는 대부분 10개 이하의 물체를 포함하지만 실험을 위해 20개 이상의 물체를 포함하는 질의도 요구하였다.

질의에 포함되어 있는 물체의 개수에 따른 그래프의 평균 에지의 개수 변화는 그림 7과 같다. 질의 변환 알고리즘에 따라 물체의 개수가 커질 경우 에지 개수의 차이가 더욱 큰 폭으로 벌어진다. 각각의 질의에 대해서, 추이 에지를 삭제한 결과로 탐색 모듈의 비교 회수가 큰 폭으로 감소됨을 알 수 있다.

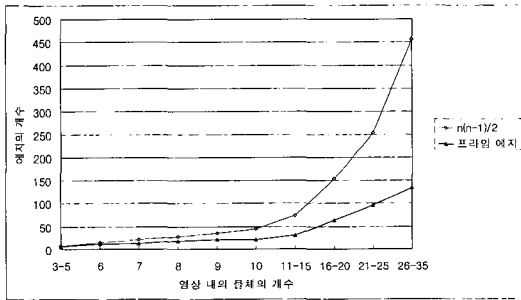


그림 7 물체 개수에 따른 에지 개수의 변화 비교

6. 결론

질의 영상을 사용자가 직접 작성하도록 하는 내용 기반 영상 검색 시스템은 질의 영상의 부정확성으로 인한 특성의 왜곡 현상을 해결해야 한다. 그 중 위치 정보의 왜곡을 완화하기 위해서는 물체간의 위치 관계를 비교하는 기법을 많이 사용한다. 본 논문에서는 이와 같이 물체간의 8 방향 위치 관계 정보를 이용하여 영상을 검색하는 QBM 시스템을 위한 질의 변환 알고리즘을 제안하였다. 제안된 알고리즘은 영상에 포함된 물체들간의 위치 관계에 추이성이 존재하는 경우 정보가 중복되지는 사실을 기반으로 하여 질의에 존재하는 추이성을 모두 제거함으로써 질의 영상을 최소 에지를 가지는 그래프로 변환한다. 질의 변환의 궁극적인 목적은 비교 회수를 최소화하여 탐색 모듈의 효율성을 높이는 것이다. 실험 결과, 에지의 개수와 밀접한 관련이 있는 에지 비교 회수를 최소로 하여 평균 비교 회수를 감소시킬 수 있다.

위치 관계 비교 회수와 함께, 영상 데이터베이스의 인덱싱 역시 검색의 효율을 결정짓는 중요한 요소이다. 향후에는 QBM에 SR-트리[13]와 같은 다차원 인덱스 구조(multidimensional index structure)를 도입하여 10000개 이상의 영상이 저장되어 있는 대형 데이터베이스에서도 효율적인 검색이 가능하도록 보완할 계획이다.

참고 문헌

[1] A. D. Bimbo and P. Pala, "Visual Image Retrieval by

Elastic Matching of User Sketches," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(2), Feb 1997.

[2] C. C. Chang, and S. Y. Lee, "Retrieval of Similar Pictures on Pictorial Databases," *Pattern Recognition*, 24(7), 1991.

[3] S. Chang, Q. Shi and C. Yan, "Iconic Indexing by 2-D Strings," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(3), May 1987.

[4] V. N. Gudivada and V. V. Raghavan, "Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity," *ACM Transactions on Information Systems*, 13(2), April 1995.

[5] C. E. Jacobs, A. Finkelstein, and D. H. Salesin, "Fast Multiresolution Image Querying," *Proceedings of SIGGRAPH '95*, pp.277-286, ACM, New York, 1995.

[6] T. Kato, T. Kurita, N. Otsu, and K. Hirata, "A Sketch Retrieval Method for Full Color Image Database," *Proceedings of 11th ICPR*, pp.530-533, IEEE, 1992.

[7] P. M. Kelly, M. Cannon, and D. R. Hush, "Query by Image Example: the CANDID Approach," *SPIE Vol. 2420 Storage and Retrieval for Image and Video Databases III*, pp.238-248, 1995.

[8] S. Lee and F. Hsu, "Spatial Reasoning and Similarity Retrieval of Images Using 2D C-String Knowledge Representation," *Pattern Recognition* 25(3), pp.305-318, 1992.

[9] F. Liu and R. W. Picard, "Periodicity, Directionality, and Randomness: Wold Features or Image Modeling and Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18(7), pp.722-733, July 1996.

[10] C. Schmid and R. Morh, "Image Retrieval Using Local Characterization," *Proceedings of ICIP-96*, pp.781-783, IEEE, 1996.

[11] J. R. Smith, "Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression," *Ph.D. thesis, Graduate School of Arts and Sciences, Columbia University*, February, 1997.

[12] A. Vailaya, Y. Zhong and A. K. Jain, "A Hierarchical System for Efficient Image Retrieval," *Proceedings of 13th ICPR*, pp.356-359, IEEE, 1996.

[13] Norio Katayama and Shin'ichi Satoh, "The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries," *Proc. ACM SIGMOD*, pp.369-380, 1997.



조 수 진

1992년 서울대학교 자연과학대학 계산통
과학과 학사. 1994년 서울대학교 자연과
학대학 계산통계학과 석사. 2000년 서울
대학교 자연과학대학 전산과학과 박사.
관심분야는 컴퓨터 비전, 지식 기반 시스
템, 데이터베이스, 멀티미디어 등



유 석 인

1977년 서울대학교 전기공학과 졸업(학
사). 1980년 미국 Lehigh University 전
산공학(석사). 1985년 미국 University
of Michigan 전산공학(박사). 1985년 ~
현재 서울대학교 자연과학대학 전산과학
과 교수. 관심분야는 인공지능, 특히 지
식기반 시스템, 기계학습, 컴퓨터비전, Planning, 객체지향
시스템 등