

# 지형/지물 이미지 데이터베이스를 위한 형태 특징 추출 방법

## (A Shape Feature Extraction Method for Topographical Image Databases)

권 용 일<sup>†</sup>      박 호 현<sup>\*\*</sup>      이 석 룡<sup>\*\*\*</sup>      정 진 완<sup>\*\*\*\*</sup>  
(Yong-Il Kwon)    (Ho-Hyun Park)    (Seok-Lyong Lee)    (Chin-Wan Chung)

**요 약** 항공 및 위성 사진과 같은 지형/지물 이미지는 대부분 비슷한 색상과 질감을 갖는다. 따라서 지형/지물 이미지 데이터베이스에서 질의 이미지를 정확하게 검색하기 위해서는 이미지의 형태 특징을 추출하는 것이 매우 중요하다. 본 논문에서는 지형/지물 이미지 검색을 위한 형태 특징 추출 방법을 제안한다. 이 방법은 이미지에 홀(hole)이 있거나 이미지가 연결되지 않은 영역들로 구성되어 있을 경우에도 형태 특징들을 효과적으로 추출할 수 있다. 그리고 세그멘테이션(segmentation)과 같은 특징 추출의 전처리 과정(pre-processing)에서 발생할 수 있는 오류에 강인하다. 본 논문에서 제안하는 방법이 기존의 방법들보다 우수함을 다양한 이미지 검색 실험을 통해 보인다.

**키워드** : 이미지 데이터베이스, 지형/지물 데이터베이스, 특징 추출, 원형 투영

**Abstract** Topographical images such as aerial and satellite images are usually similar with respect to colors and textures but not in shapes. Thus shape features of the images and the methods of extracting them become critical for effective image retrieval from topographical image databases. In this paper, we propose a shape feature extraction method for topographical image retrieval. The method extracts a set of attributes which can model the presence of holes and disconnected regions in images and is tolerant to pre-processing, more specifically segmentation, errors. Various experiments suggest that retrieval using attributes extracted using the proposed method performs better than using existing shape feature extraction methods.

**Key words** : Image databases, Topographical databases, Feature extraction, Polar projection

### 1. 서 론

이미지 검색이란 사용자로부터 주어진 이미지, 즉 질의 이미지(query image)와 유사한 이미지를 이미지 데이터베이스로부터 찾는 과정이다. 이미지 검색은 이미지에 대한 주석(annotation)과 실제 내용(content)을 이용

해 이루어 질 수 있다. 주석 기반의 이미지 검색은 사람이 직접 각각의 데이터베이스 이미지에 대해 주석을 작성한 후에 검색 시 그 주석을 이용하는 것이다. 이것은 명백히 진보된 이미지 데이터베이스에 적합한 검색 방법이 아니다. 반면에 내용 기반의 이미지 검색(content-based image retrieval)에서는 색상(color), 질감(texture), 형태(shape) 특징들이 이미지로부터 자동적으로 추출되고, 이들은 질의 이미지와 유사한 이미지를 찾는 데 이용된다[1]. 본 논문은 내용 기반의 이미지 검색에 초점을 맞춘다. 내용 기반의 이미지 검색 방법은 다음과 같이 이미지의 종류에 따라 달라진다: 1) 색상이 중요한 이미지(예: 상품 카탈로그, 풍경과 꽃 사진) 2) 질감이 중요한 이미지(예: 벽돌 담장 사진, 직물 사진, 파도 사진) 3) 형태가 중요한 이미지(예: 교통 표지판, 상표). 이것은 한 종류의 이미지에 적합한 검색 방법이 다른 종류의 이미지에 대해서는 효과적인 방법이 아닐 수 있

· 본 연구는 한국과학기술원 영상정보특화연구센터를 통한 국방과학연구소의 연구비 지원과 한국전산원의 광대역통합연구개발망 활용 연구 및 시범 과제를 통한 연구비 지원으로 수행되었습니다.

† 정 회 원 : 중앙대학교 전자전기공학부  
kyi0605@hanmail.net

\*\* 종신회원 : 중앙대학교 전자전기공학부 교수  
hohyun@cau.ac.kr

\*\*\* 종신회원 : 한국외국어대학교 산업정보시스템공학부 교수  
sillee@hufs.ac.kr

\*\*\*\* 종신회원 : 한국과학기술원 전자전산학과 교수  
chungcw@cs.kaist.ac.kr

논문접수 : 2005년 8월 17일  
심사완료 : 2006년 5월 9일

음을 의미한다. 기존의 방법으로 정확한 검색을 할 수 없었던 이미지 종류들 중의 하나로 본 논문에서 다루는 지형/지물 이미지가 있다.

다른 종류의 이미지와 비교할 때 지형/지물 이미지, 특히 항공사진과 위성사진의 경우는 다음과 같은 이유 때문에 정확한 검색이 어렵다. 첫째로 지형/지물 이미지는 대체로 비슷한 색상과 질감을 갖기 때문에 이와 같은 특징들을 이용해 이미지를 구별하기 어렵다. 따라서 지형/지물 이미지의 검색에서 형태 특징은 매우 중요하다. 둘째로 지형/지물 이미지는 복잡한 형태를 갖는 경우가 많아 단순 형태의 이미지보다 정교한 특징 추출이 요구된다. 마지막으로 지형/지물 이미지는 대체로 해상도가 낮고, 지형/지물에 반사되는 빛의 양과 같이 통제 불가능한 요소는 특징 추출의 전처리 과정인 세그멘테이션에 영향을 줄 수 있다. 결국 지형/지물 이미지의 정확한 검색을 위해서는 위의 사항들을 고려할 수 있는 특징 추출 방법이 필요하다.

본 논문에서 지형/지물 이미지는 특징 추출할 객체(예: 섬)와 배경(예: 바다)은 분리되어 있다고 가정한다. 즉 이미지는 특징 추출하기 전에 세그멘테이션 되어 있다. 일단 세그멘테이션 과정이 완료되면 이미지는 이진 이미지(binary image)로 변환된다. 즉 영역과 배경 화소(pixel)는 각각 0과 1로 표현된다[2]. 그리고 지형/지물 이미지 데이터베이스에서 검색의 대상은 주로 이미지 자체보다 그 안에 포함된 객체들(objects)일 것이다. 예를 들어 군도(archipelago) 이미지에서 군도보다는 각 섬에 대한 검색이 이루어질 것이다. 이 때문에 이미지에 포함된 각각의 객체를 분리해서 저장하는 것이 더 유용하다. 이를 위해 본 논문에서는 세그멘테이션 과정을 거친 이미지는 분리 과정(separation process)에 의해 개별의 객체로 나뉘고 이들은 하나의 이미지로 저장된다고 가정한다. 결국 이런 전처리 과정 후에 이미지 데이터베이스가 구축되고, 하나의 이미지는 단일 객체를 갖는다. 예를 들어 그림 1(a)는 원래 이미지가 세그멘테이션 과정을 통해 영역과 배경으로 구분된 이미지를 나

타낸다. 그리고 이 이미지는 검색을 위해 분리 과정을 거쳐서 그림 1(b)와 같이 개별적으로 데이터베이스에 저장된다.

영역의 형태는 간단한 형태(simple shape)와 복잡한 형태(complex shape)로 나뉘질 수 있다. 간단한 형태는 오직 하나의 영역(region)을 갖고 그 영역은 홀(hole)이 없는 다각형(polygon)으로 표현된다. 복잡한 형태는 홀을 포함하거나 다수의 연결되지 않은 영역들을 갖는다. 예를 들어 군도를 이루는 작은 섬들이나 호수를 포함하고 있는 지형은 복잡한 형태이다.

지형/지물 이미지에서 연결되지 않은 인접한 영역들을 단일 객체로 간주하는 것은 중요하다. 이것은 단일 영역이 세그멘테이션 과정에서 발생하는 오류로 인해 연결되지 않은 영역들로 바뀔 수 있기 때문이다. 그러나 이와 같은 세그멘테이션 오류가 검색 과정에 영향을 미쳐서 결과의 정확성을 감소시키면 안 된다.

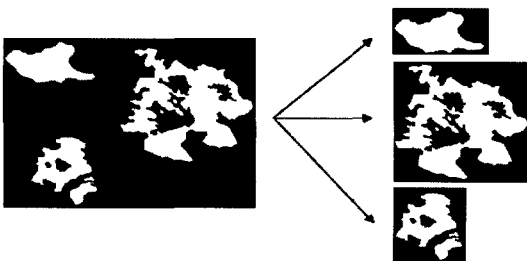
기존의 내용 기반의 이미지 검색 시스템은 색상, 질감 그리고 간단한 형태 특징을 이용하여 이미지를 검색한다[1,3,4]. 이와 같은 시스템이 복잡한 형태를 갖는 지형/지물 이미지 검색에 이용될 때 검색의 정확성은 매우 떨어지게 된다. 따라서 지형/지물 이미지의 내용 기반 검색을 수행하기 위한 새로운 방법이 필요하다.

본 논문에서는 지형/지물 이미지의 형태 특징을 보다 효과적으로 추출 할 수 있는 방법인 원형 투영(polar projection)을 제안한다. 원형 투영은 복잡한 형태의 객체를 기술할 수 있는 속성 벡터들(attribute vectors)을 추출한다. 그리고 이 방법은 다른 기존의 형태 특징 추출 방법과 달리 객체의 표현 정도를 조절할 수 있다. 즉 객체를 자세히 또는 대략적으로 표현하는 것이 가능하다. 그리고 이것은 이미지 검색의 속도와 정확성 그리고 저장 공간과 관계된다.

본 논문의 나머지 부분은 다음과 같이 구성된다. 2절에서는 기존의 형태 특징 추출 방법에 대해 간단히 설명하고, 3절에서는 원형 투영의 특징 추출 과정에 대해 자세히 언급하고 이 방법의 장점을 설명한다. 4절에서는 원형 투영 방법에서 사용될 거리 함수를 정의하고, 어떻게 두 이미지를 효율적으로 검색할 수 있는지에 대해 설명한다. 5절에서는 다양한 실험을 통해 원형 투영 방법의 우수성을 보이고, 6절에서는 본 논문의 전체적인 결론을 짓는다.

## 2. 관련 연구

이미지의 형태 특징 추출 방법은 영역 기반의 특징 추출(region-based shape feature extraction) 방법과 외곽선 기반의 특징 추출(contour-based shape feature extraction) 방법으로 나뉘질 수 있다[5]. 영역 기반의



(a) 세그멘테이션 과정

(b) 분리 과정

그림 1 세그멘테이션된 이미지와 분리된 객체

방법은 영역의 외곽선과 그 내부의 화소 분포를 형태 특징으로 추출하는 것이고, 외곽선 기반의 방법은 오직 영역의 외곽선에 관한 특징을 추출하는 것이다. 본 논문에서 제안하는 원형 투영은 영역 기반의 특징 추출 방법으로 분류된다.

앞서 언급한 대로 세그멘테이션 과정을 거친  $S_X \times S_Y$  크기의 이진 이미지를 생각해 보자. 여기서 각각의 화소  $p(x,y)$ 는 다음에 정의된 함수  $f(x,y)$ 에 의해 주어진 값을 갖는다.

$$f(x,y) = \begin{cases} 1 & \text{if } p(x,y) \text{ belongs to a region} \\ 0 & \text{otherwise} \end{cases}$$

영역 기반의 방법 중 하나로 다음과 같이 정의되는 투영(projection) 방법이 있다. 이 방법은 직각 좌표계(Cartesian coordinates) 위에서 수행되므로 본 논문에서는 이를 직각 투영(Cartesian projection)이라고 부를 것이다.

$$p_X(x) = \sum_{y=1}^{S_Y} f(x,y), \quad x=1,2,\dots,S_X$$

$$p_Y(y) = \sum_{x=1}^{S_X} f(x,y), \quad y=1,2,\dots,S_Y$$

즉,  $p_X(x)$ (또는  $p_Y(y)$ )는 X-축(또는 Y-축) 위의 각 화소에 대해 Y-축(또는 X-축)을 따라 영역 화소의 수를 세는 것을 의미한다[2]. 그림 2는 직각 좌표계 위에서 수행되는 투영의 한 예를 보여 준다. 각각의 정사각형은  $6 \times 6$ (pixels) 이미지에서 화소를 나타낸다. Y-축 왼쪽(또는 X-축 아래쪽)에 있는 수는 각각의  $y$ (또는  $x$ )에 대해 X-축(또는 Y-축)을 따라 투영된 값, 즉 영역 화소의 수를 의미한다. 일단 이처럼 투영된 값을 얻게 되면, 두 이미지는 유클리디안 거리(Euclidean distance)와 같은 어떠한 벡터 기반의 거리로도 비교될 수 있다.

직각 투영 방법은 직각 좌표계에서 추출된 특징 값을 이용하여 두 이미지를 비교하므로 이 방법은 이미지의 방향에 따라 값이 변한다. 다른 말로 직각 투영은 방향

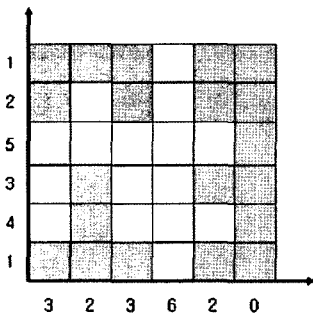
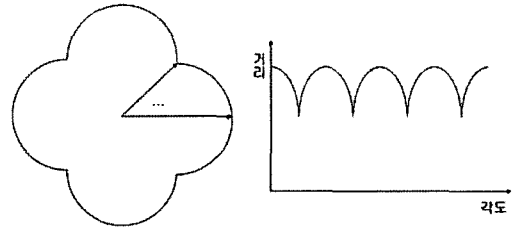


그림 2 직각 좌표계 위의 투영 예

불변(rotation invariant)하지 않다. 그리고 이 방법은 객체 안의 홀과 연결되지 않은 영역들에 대해 고려하지 않는다.

외곽선 기반의 특징 추출 방법으로 시그너처(signature)가 있다[6]. 이 방법은 무게 중심과 같은 영역 내부의 점을 기준으로 영역의 외곽선까지의 거리를 방향에 따라 추출하는 것이다.



(a) 객체의 외곽선 (b) 특징 추출 결과  
그림 3 시그너처를 이용한 특징 추출 예

그림 3은 시그너처를 이용한 객체의 형태 특징 추출을 보여준다. 그림 3(a)의 객체에 대해 시그너처 방법을 적용하면 그림 3(b)의 그래프와 같이 방향에 따라 특징 값을 추출한다. 이 방법은 위의 예와 같이 한 방향에 대해 유일한 특징 값을 갖는 형태에만 이용될 수 있다. 다시 말해 영역의 기준점에서 외곽선까지 있는 선이 다른 외곽선과 만나는 형태의 특징 추출에는 적합하지 않다.

또 다른 외곽선 기반의 방법으로 Curvature Scale Space(CSS)가 [7-9]에 제안 되었다. CSS는 과거 이미지 검색 분야에서 성공적인 특징 추출 방법으로 사용되었고, MPEG 개발에 힘입어 보다 확장되고 진보하였다[5]. CSS의 개념은 영역의 형태를 비교할 때 그 외곽선의 불룩한 부분과 오목한 부분을 형태 특징으로 이용하는 것이다[9]. 그리고 두 부분은 외곽선의 곡률(curvature)이 0을 만족하는 변곡점(inflexion point)에 의해 분리된다. 여기서 곡률은 외곽선 위의 접선 기울기의 변화율을 의미한다. 이미지의 영역에 대한 CSS의 특징 추출 과정은 다음과 같다[5,7].

- (1) 특징 추출할 영역의 외곽선을 추출한다.
- (2) 외곽선을 저역 통과 필터(low pass filter)에 통과시켜 부드럽게 만든다. 이 과정은 외곽선의 오목한 부분을 점차 제거한다.
- (3) 저역 통과 된 새로운 외곽선의 곡률이 계산되고, 변곡점의 위치와 개수를 기록한다.
- (4) 외곽선의 모든 오목한 부분이 제거될 때까지 과정 (2)와 (3)을 반복한다.
- (5) 변곡점이 제거될 때 그것의 위치와 필터링한 횟수를 외곽선의 특징 값으로 사용한다.

직각 투영과 시그너처와 같이 CSS 또한 오직 간단한 형태의 객체에 대해서만 적용 가능하다. 따라서 위의 방법들은 홀이 있거나 연결되지 않은 영역들을 갖는 복잡한 객체에 대해서는 적합하지 않다. 다음 절에서 제안하는 형태 특징 추출 방법은 이와 같은 문제점을 해결할 수 있다.

### 3. 원형 투영

본 논문에서 제안하는 방법의 기본 개념은 이미지에 대해 방사선(ray)을 이용한 사용자 정의의 이산 표현(user-defined discrete expression)을 갖는 것이다. 여기서 방사선은 객체의 원점에서 시작하여 외곽선으로 일정 각도마다 향하고 이것은 이미지의 내용을 표본화(sampling)한다. 제안한 방법은 원형 좌표계(polar coordinates) 위에서 투영이 수행되므로 원형 투영(polar projection)의 이름을 갖는다. 각 방사선에 대해 속성들이 계산되고 모든 방향에서 얻어진 속성들은 객체의 형태를 기술하는데 사용된다.

속성 추출을 위해 먼저 해야 할 일은 이미지의 원점을 찾는 것이다. 본 논문에서는 그 원점을 이미지의 무게 중심이라 가정하고 어떻게 무게 중심을 구하는지 간단히 설명할 것이다. 다음으로 객체의 무게 중심에서 시작하여 외곽선에서 끝나는 일정 간격 방향의 방사선을 얻는다. 각 방사선 사이의 각도를  $\theta$ 라 하면  $k=360/\theta$ 개의 방사선  $R_1, R_2, \dots, R_k$ 이 얻어지고 방사선  $R_i (i=1, 2, \dots, k)$ 는 다음과 같이 정의된다.

$$R_i = \langle P_1^i, P_2^i, \dots, P_{n_i}^i \rangle$$

여기서 각각의  $P_j^i (j=1, 2, \dots, n_i)$ 는 이미지 화소  $p(x, y) (x \in 1, 2, \dots, S_x, y \in 1, 2, \dots, S_y)$ 를 나타내고 다음의 특성을 갖는다.

- $P_1^i$ 은 객체의 무게 중심이다.
- $P_{n_i}^i$ 은 방향  $(i-1)\theta$ 로 뻗은 방사선  $R_i$  상에서 가장 멀리 떨어진 영역 화소이다.
- $d(P_1^i, P_{j+1}^i) > d(P_1^i, P_j^i), \forall j=1, 2, \dots, n_i-1$

즉, 각도  $(i-1)\theta$ 에서 방사선은 객체의 무게중심부터 외곽선까지의 화소들이 순서대로 나열된 리스트(list)를 의미한다. 각각의 방사선에 대해서 다음의 세 가지 속성들을 추출한다:

- (1) 영역 화소의 빈도(Region Pixel Frequency, RPF): 방사선이 갖는 영역 화소의 수.
  - (2) 영역 교대의 빈도(Region Alternation Frequency, RAF): 영역 화소와 배경 화소 사이의 변화 수.
  - (3) 방사선의 길이(Radial Length, RL)
- 기존의 방법들에 비해 위의 속성들을 사용하는 것은

몇 가지 장점을 갖고 있다. 첫째로 영역 교대의 빈도는 객체 안의 홀의 여부를 알게 해준다. 둘째로 한 이미지에  $360/\theta$ 개의 방사선이 존재하고 속성들은 각 방사선마다 얻어지며,  $\theta$ 는 사용자 정의의 매개 변수(parameter)이므로 객체는 사용자에게 의해 자세히 또는 대략적으로 표현될 수 있다. 두 이미지를 비교할 때 소비되는 계산 시간과 추출된 속성들을 위한 저장 공간을 고려해  $\theta$ 를 조절할 수 있다.

지금부터 객체의 무게 중심과 위의 세 가지 속성들을 어떻게 계산하는지에 대해 설명한다. 객체의 무게 중심을 찾기 위해서 먼저 이진 이미지에서 영역 화소의 총 수를 계산해야 한다.

$$XY_{sum} = \sum_{x=1}^{S_x} \sum_{y=1}^{S_y} f(x, y)$$

그 다음으로 영역 화소의  $x$ 와  $y$  좌표 값의 합을 각각 구한다.

$$X_{sum} = \sum_{x=1}^{S_x} \sum_{y=1}^{S_y} f(x, y) \times x$$

$$Y_{sum} = \sum_{x=1}^{S_x} \sum_{y=1}^{S_y} f(x, y) \times y$$

마지막으로 무게 중심은 아래와 같이 계산된다.

$$(x_c, y_c) = \left( \frac{X_{sum}}{XY_{sum}}, \frac{Y_{sum}}{XY_{sum}} \right)$$

무게 중심이 얻어지면, 사용자 정의의  $\theta$ 는 객체를 기술하는 방사선의 수, 즉  $k$ 를 결정하기 위해 사용된다. 앞으로 언급할 식의 편리한 표현을 위해, 2절에서 정의된 직각 좌표계에서의 함수  $f(\cdot, \cdot)$ 을 다음과 같이 원형 좌표계에서 다시 정의한다.

$$F(P_i^k) = \begin{cases} 1 & \text{if pixel } P_i^k \text{ belongs to a region} \\ 0 & \text{otherwise} \end{cases}$$

각 방사선  $R_k$ 에 대해 세 가지 속성들은 다음과 같이 얻어진다.

- $RPF_k = \sum_{P_i^k \in R_k} F(P_i^k)$ . (하나의 방사선 안에 있는 영역 화소의 수를 의미한다.)
- $RAF_k = \sum_{P_i^k \in R_k} |F(P_i^k) - F(P_{i+1}^k)|$ . (영역과 배경 화소 사이의 교대 수를 의미한다.)
- $RL_k = d(P_1^k, P_{n_k}^k)$ . (방사선의 길이를 의미한다.)

그림 4는 지형/지물 이미지에 대한 원형 투영을 실행한 예를 보여준다. 왼쪽 그림은 오른쪽 지형/지물 이미지에서 타원으로 표시된 부분의 방사선을 확대해 나타낸 것이다. 각각의 사각형은 방사선의 화소를 나타내고 가장 오른쪽 화소는 객체의 무게중심이다. 무게 중심에서 시작해서 화살표 방향으로 투영 연산을 하면 9개의

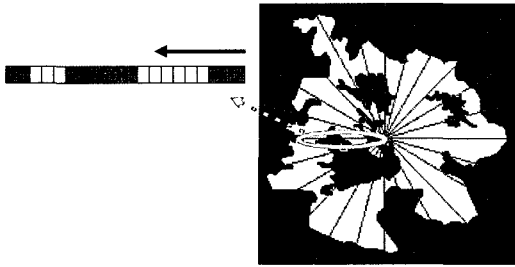


그림 4 원형 투영을 이용한 특징 추출 예

원색 화소를 발견할 수 있다. 따라서 영역 화소의 빈도(RPF)는 9가 된다. 투영 연산 과정에서 3번째 화소(검은색)와 4번째 화소(흰색) 사이에서 색의 변화가 나타난다. 이 밖에도 9~10번째 화소, 15~16번째 화소 사이에서 색의 변화가 나타난다. 그러므로 영역 교대의 빈도(RAF)는 3이다. 마지막으로 18번째 화소가 무게 중심에서 가장 멀리 떨어져 있는 영역의 화소이므로 방사선의 길이(RL)는 17이 된다.

추출된 속성들은 이미지내 영역의 위치에 따라 변하지 않는다. 이것은 영역이 다른 위치로 이동하면 무게 중심도 따라서 움직이고 결국 얻어지는 값은 같기 때문이다. 그러나 이 속성들은 방향 불변(rotation invariant)을 만족하지 못한다. 그리고 지형/지물 이미지는 임의의 방향에서 촬영될 수 있으므로 방향 불변의 특징 추출은 매우 중요하다. 방향 불변을 위해 한 속성에 대해서 가장 큰 값을 기준점으로 이용할 수 있는데 본 논문에서는 방사선의 길이(RL) 속성을 사용한다. 여기서 방사선의 길이가 가장 긴 방사선을 참조 방사선(reference ray)이라 한다. 이 방법은 참조 방사선을 기준으로 객체의 메타데이터(meta data)를 이동시켜 모든 객체의 참조 방사선이 속성 벡터의 첫 번째 원소가 되도록 하는 것이다. 그 결과 원형 투영이 방향 불변의 성질을 갖게 해준다. 2절에서 언급한 직각 투영 방법을 이용할 때 이와 같은 방식의 속성 이동은 수행될 수 없다. 방향 불변을 위한 위 방법의 사용은 4절에서 보다 자세히 설명될 것이다.

메타데이터의 크기 불변(scale invariant)을 위해 각각의 속성 값은 모든 방사선에서 계산된 그 속성의 평균값으로 나뉘진다. 정규화 계수로서 최대값(maximum value) 대신 평균값(mean value)을 사용하는 이유는 최대값보다 평균값이 세그멘테이션 과정에서 발생하는 잡음(noise)에 더 강인하기 때문이다.

지금까지의 내용은 다음과 같이 요약된다. 지형/지물 이미지와 방사선 사이의 각도  $\theta$ 가 주어지면,  $360/\theta$ 개의 방사선에 대해 각각의 세 가지 속성들(RPF, RAF, RL)이 계산되고 이 값들은 검색을 위해 저장된다. 이때 형

태의 비교를 빠르게 하기 위해 참조 방사선의 메타데이터는 속성 벡터의 첫 번째 원소로 저장된다. 그리고 이것을 기준으로 반시계 방향의 방사선들에 대한 속성 값들이 연속적으로 저장된다.

기존의 세 가지 방법과의 비교를 통해 본 논문에서 제안하는 원형 투영의 장점을 보인다. 직각 투영은 특징 값으로 오직 영역 화소의 빈도만을 이용한다. 이에 반해 원형 투영은 영역 화소의 빈도뿐만 아니라 영역 교대의 수와 방사선의 길이를 이용한다. 방사선의 길이는 객체의 외곽선 정보를 얻는데 사용되고 교대의 수는 객체 내부의 홀과 연결되지 않은 영역들을 확인하는데 이용될 수 있다.



그림 5 영역 화소 빈도가 같은 두 이미지

영역 화소의 빈도만을 사용한 경우는 그림 5의 두 이미지를 구별 할 수가 없다. 그러나 두 이미지의 형태는 매우 다르다. 즉 왼쪽 이미지는 한 개의 홀을 갖는 고리 형태이지만 오른쪽 이미지는 단순한 원이다. 오른쪽 이미지에서 영역의 반지름은 왼쪽 이미지에서 고리의 두께와 같기 때문에 임의의 방향( $k$ )에 대해 두 이미지의 영역 화소의 빈도, 즉  $RPF_k$ 는 같다. 두 이미지를 구별하기 위해 영역과 배경 사이의 교대의 수와 방사선의 길이가 사용된다. 이 경우 왼쪽 이미지의 교대 수는 모든 방향에서 1이지만 오른쪽 이미지에서는 0이다.

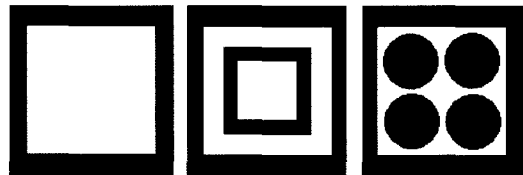


그림 6 방사선의 길이가 같은 세 이미지

그림 6의 세 이미지는 서로 다른 형태이지만 동일한 외곽선을 갖는다. 따라서 시그너처 방법은 오직 객체의 무게중심에서 외곽선까지의 거리를 특징으로 하기 때문에 위의 이미지들을 구별할 수 없다. 이처럼 시그너처는 객체 내부의 영역 화소들을 고려하지 않는다. 그러나 원

형 투영 방법은 영역 화소의 빈도(RPF)와 영역 교대의 빈도(RAF)를 이용해서 외곽선 내부의 영역 화소의 분포를 추출할 수 있다.

결국 원형 투영 방법은 객체의 외곽선과 그 내부의 영역 화소에 관한 특징을 추출하므로 직각 투영과 시그너치 방법으로 비교하기 어려운 그림 5와 그림 6의 이미지들을 구별할 수 있다. 이런 장점을 갖는 원형 투영은 지형/지물 이미지 데이터베이스에 적용될 경우, 질의 이미지와 유사한 이미지를 정확하게 검색할 수 있다.

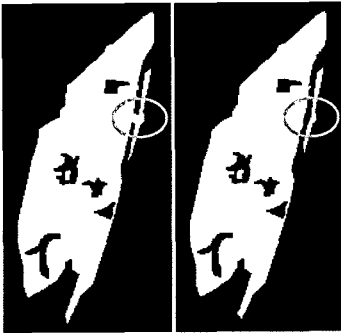


그림 7 세그멘테이션 오류의 예

그림 7은 세그멘테이션 오류가 일어나는 상황을 보여 준다. 원으로 표시된 부분의 두 영역은 왼쪽 이미지에서는 연결되었지만 오른쪽 이미지에서는 분리되어 있다. 왼쪽 이미지에서 두 영역의 연결은 그 외곽선이 모퉁한 부분 갖게 하여 오른쪽 이미지에서 영역의 외곽선과 매우 다르게 된다.

결국 두 이미지에 대해서 CSS 방법은 매우 다른 메타데이터를 만들어내지만, 원형 투영 방법에 의해 얻어진 메타데이터는 그 차이가 매우 작다. 그리고 이 차이는 두 이미지가 그 만큼 다르다는 것을 의미한다.

#### 4. 거리 함수(Distance Function)

본 절에서는 3절에서 추출된 속성 값들을 이용하여 이미지를 비교하는 방법에 대해 설명한다. 질의 이미지가 갖는 형태와 데이터베이스에 저장된 이미지의 형태 비교는 그것들로부터 추출된 특징들 간의 거리를 계산함으로써 이루어진다.

특징 추출 과정 후의 두 형태(I, Q)는 다음과 같은 리스트로 표현될 수 있다.

$$I = \langle (RPF_1^I, \dots, RPF_k^I), (RAF_1^I, \dots, RAF_k^I), (RL_1^I, \dots, RL_k^I) \rangle$$

$$Q = \langle (RPF_1^Q, \dots, RPF_k^Q), (RAF_1^Q, \dots, RAF_k^Q), (RL_1^Q, \dots, RL_k^Q) \rangle$$

여기서  $RPF_j^I$ ,  $RAF_j^I$ ,  $RL_j^I$ 는 방사선  $R_j$ 에서 얻어진 속성 값들이다. 그리고 3절에서 언급한 참조 방사선 때문에 모든  $j$ 에 대해  $RL_j^I \geq RL_j^Q$ 를 만족한다. 이것은 형

태 Q에서도 마찬가지다. 길이가 같은 방사선이 여러 개일 경우 다수의 방사선이 참조로서 선택될 수 있는데, 이 경우 본 논문에서는 그것들 중 임의로 하나를 선택한다.

각 속성의 유형에 대해 유사성 측정은 다음과 같이 유클리디안 거리(Euclidean Distance)로 계산 될 수 있다.

$$D_{RPF}(I, Q) = \sqrt{\sum_{j=1}^k (RPF_j^I - RPF_j^Q)^2}$$

$$D_{RAF}(I, Q) = \sqrt{\sum_{j=1}^k (RAF_j^I - RAF_j^Q)^2}$$

$$D_{RL}(I, Q) = \sqrt{\sum_{j=1}^k (RL_j^I - RL_j^Q)^2}$$

원형 투영이 이용되는 이미지의 종류에 따라 어떤 속성 값은 다른 속성 값보다 형태 비교에 더 중요할 수 있다. 이를 반영하기 위해 위에서 언급한 각 속성의 거리에 대한 가중치(weight)가 적용된다. 그 결과 다음의 거리 함수를 얻는다.

$$D(I, Q) = w_{RPF} * D_{RPF}(I, Q) + w_{RAF} * D_{RAF}(I, Q) + w_{RL} * D_{RL}(I, Q)$$

여기서  $w_{RPF} + w_{RAF} + w_{RL} = 1$ 이고 사용자는 가중치를 원하는 값으로 조절할 수 있다.

그림 8의 세 가지 형태에 대한 유사성 거리(similarity distance)의 계산을 보인다. 예를 간단히 하기 위해  $\theta = 90$ 을 사용한다. 객체 A는 객체 B를 시계 방향으로 90도 회전 시키고 약간의 형태 차이를 갖는다. 반면에 객체 C는 객체 A와 B와 매우 다르다. 따라서 유사성 함수는  $D(A, B) < D(A, C)$ 을 만족한다.

이전 절에서 설명한 정의와 위의 표기를 이용했을 때, 원형 투영을 통해 얻어진 객체 A, B, C의 메타데이터는 각각 다음과 같이 표현된다.

$$A = \langle (3, 2, 3, 4), (4, 2, 2, 0), (4, 2, 3, 3) \rangle$$

$$B = \langle (3, 3, 3, 5), (4, 4, 2, 0), (4, 4, 3, 4) \rangle$$

$$C = \langle (3, 3, 3, 3), (0, 0, 0, 0), (2, 2, 2, 2) \rangle$$

먼저 객체 A를 예를 들어 설명하면  $RL_2^A$ 의 값이 RL 속성에서 가장 큰 값이기 때문에 위의 메타데이터는 이미  $R_2^A, R_3^A, R_4^A, R_1^A$  순서로 속성 값들이 나열된다. 객체 B의 경우에  $R_2^B, R_3^B, R_4^B$ 가 참조 방사선이 될 수 있지만 여기서는  $R_3^B$ 를 선택한다. 객체 C는 어떤 방사선도 참조 방사선이 될 수 있다. 위의 유사성 거리 계산식과 모든 가중치에 1/3을 정의하여 이용하면  $D(A, B) = 1.88$ 과  $D(A, C) = 2.92$ 의 값을 얻게 된다. 이 결과로부터 객체 A가 객체 C보다 객체 B와 더 유사하다는 것을 알 수 있다. 그리고 이것은 사람의 눈으로 확인했을 경우와 일치한다.

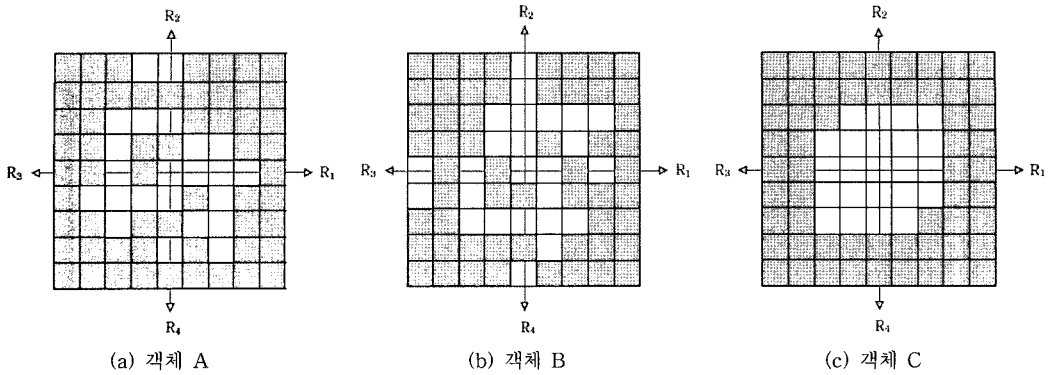


그림 8 원형 투영과 거리 함수의 예

지금부터는 제안된 원형 투영 방법의 복잡도(complexity)를 분석해 본다. 세그멘테이션과 객체의 분할 과정은 검색 과정과는 별개이기 때문에 오프라인(off-line) 상태에서 수행된다. 그리고 특징 추출은  $k$ 개의 방사선에 대해 이루어진다. 한 방사선에 포함된 화소의 수는  $M = \max_{i=1, \dots, k} |R_i|$ 에 의해 상한 경계(upper-bounded)가 된다. 이 화소들에 대한 한 번의 검사(scan)로 세 가지 속성을 추출하기 때문에 형태 특징을 추출하기 위한 시간 복잡도(time complexity)는  $O(k \times M)$ 이다. 다시 말해  $M$ 은 상수이므로 시간 복잡도는  $k$ 에 대해 선형적이다.

거리 계산 또한 방사선의 개수  $k$ 에 대해  $O(k)$ 의 선형의 복잡도를 갖는다. 참조 방사선을 이용하면  $k$  길이의 특징 벡터에 대해 유클리디안 거리는 각 속성에 대해 한 번만 계산되기 때문이다. 반면에 참조 방사선을 사용하지 않으면 질의 이미지의  $k$  길이의 특징 벡터에 대해 데이터베이스 이미지의  $k$  길이의 특징 벡터를 하나씩 회전시키면서  $k$  번의 거리함수를 구한 다음 각각을 비교하여야 하므로  $O(k^2)$ 의 복잡도를 갖는다.

원형 투영 방법에서 특징 추출과 유사성 계산이 선형적 시간으로 이루어지고 추출된 특징들은 방향/크기/위치에 불변하다. 또한 기존의 특징 추출 방법들과 달리

원형 투영은 복잡한 형태의 중요한 특징들을 추출할 수 있다. 이 때문에 기존의 직각 투영, 시그너처, CSS 보다 원형 투영이 더 우수하다. 다음 절에서는 기존의 방법들과 비교 실험을 통해 본 절에서 언급한 원형 투영의 장점을 보인다.

5. 실험 결과

본 논문에서 제안한 형태 특징 추출 방법의 성능을 평가하기 위해 다양한 실험 결과를 보일 것이다. 다음 실험들에서 가중치  $w_{RPF}$ ,  $w_{RAF}$ ,  $w_{RL}$ 는 각각 0.27, 0.27 그리고 0.46으로 하였다. 그리고 검색된 이미지가 위쪽과 왼쪽에 위치할수록 질의 이미지와 유사한 이미지이다.

3절에서 언급했듯이, 원형 투영 방법은 이미지의 방향/크기/위치에 관계없이 특징들을 추출할 수 있다. 그림 9는 이와 같은 특성들을 확인 할 수 있는 실험 결과이다. 그림 9의 첫 번째 실험에서 질의 이미지와 방향은 다르지만 동일한 영역을 포함하고 있는 이미지들이 높은 순위에 있는 것을 볼 수 있다. 이것은 질의 이미지와 유사한 이미지들이 방향에 관계없이 검색되는 것을 의미한다. 두 번째와 세 번째 실험을 통해서 이미지의 크기와 위치는 다르지만 같은 형태의 영역을 포함하는 이

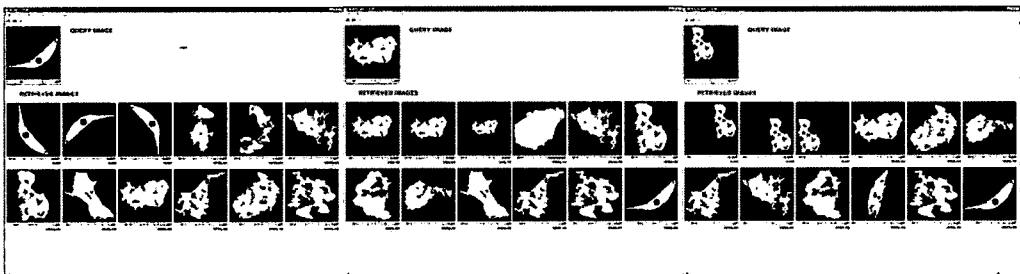
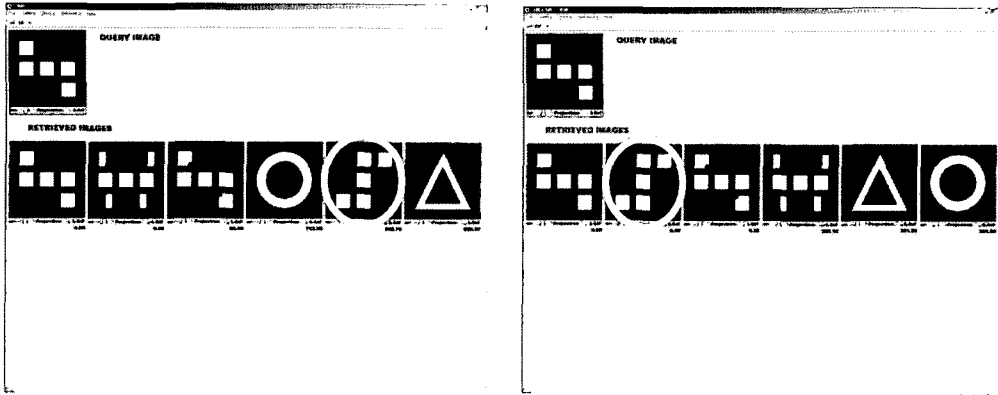


그림 9 원형 투영의 방향/크기/위치 불변의 특성



(a) 직각 투영

(b) 원형 투영

그림 10 직각 투영과 원형 투영을 이용한 이미지 검색 실험

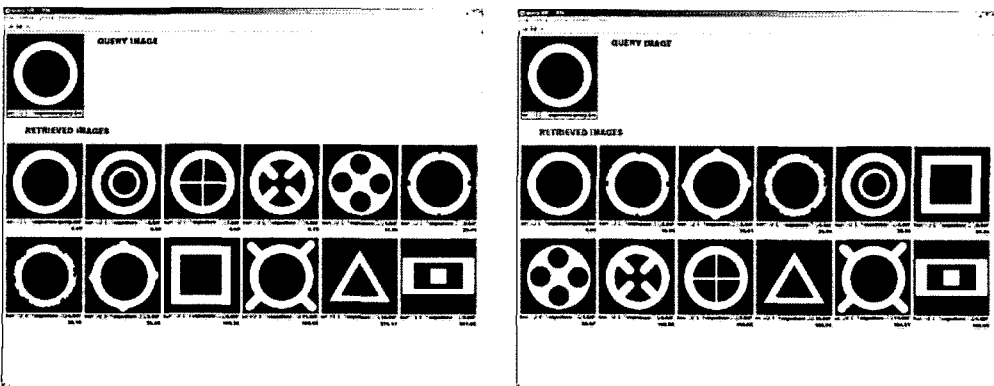
미지가 질의 이미지와 유사한 이미지로 판단됨을 볼 수 있다.

지금부터 기존의 형태 특징 추출 방법과의 비교 실험을 통해 원형 투영 방법의 우수성을 보일 것이다.

그림 10은 기존의 직각 투영 방법과 본 논문에서 제안한 원형 투영 방법의 비교 실험을 나타낸다. 그림 10(a)에서 질의 이미지와 방향이 다른 동일한 형태를 갖는 이미지가 5위에 위치하지만, 다른 형태를 갖는 이미지는 2위로 검색되는 것을 볼 수 있다. 이것은 직각 투영에서 2위의 이미지가 영역의 형태는 다르지만 같은 투영 값을 갖고 5위의 이미지는 형태는 같지만 방향이 달라 질의 이미지와는 다른 투영 값을 갖기 때문이다. 그러나 그림 10(b)의 실험 결과에서 원형 투영은 이미지의 방향에 상관없이 유사한 형태의 이미지를 높은 순위(2위)로 검색하는 것을 볼 수 있다. 그리고 그림 10(a)의 검색 결과와 달리 그림 10(b)의 실험은 사람이

인지하는 것과 같이 질의 이미지와 유사한 순서의 결과 이미지들을 얻었다. 결국 2절에서 언급했듯이 기존의 직각 투영은 방향 불변의 특성을 갖지 못하고, 질의 이미지와 유사한 이미지를 정확하게 찾을 수 없음을 확인할 수 있다.

그림 11은 기존의 시그너처 방법과 원형 투영 방법의 이미지 검색 실험 결과를 보여준다. 시그너처 방법은 원형 투영 방법에서  $w_{RPF} = w_{RAF} = 0, w_{RL} = 1$ 의 가중치를 갖는 것과 같다. 그림 11(a)의 실험에서 질의 이미지와 유사한 이미지가 1, 6, 7, 8 순위로 검색되는 것을 볼 수 있다. 그러나 그림 11(b)에서는 유사한 이미지가 검색된 이미지 중 1, 2, 3, 4 순위를 차지한다. 두 실험을 통해 시그너처 방법은 오직 방사선의 길이( $RL$ )만을 이용하기 때문에 외곽선 내부의 영역 화소를 고려하지 않는 것을 확인할 수 있다. 따라서 시그너처 방법을 이용한 검색 실험에서는 질의 이미지와 비슷한 외곽선(원



(a) 시그너처

(b) 원형 투영

그림 11 시그너처와 원형 투영을 이용한 이미지 검색 실험

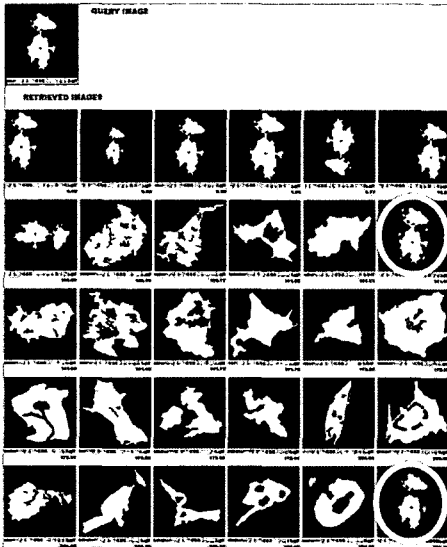


의 형태)만을 갖는 이미지가 높은 순위에 있다. 그러나 원형 투영을 이용한 실험에서는 질의 이미지의 형태와 유사한 이미지가 실제로 높은 순위에 있다. 이 실험에서 원형 투영이 시그너처 보다 우수하다는 것을 알 수 있다.

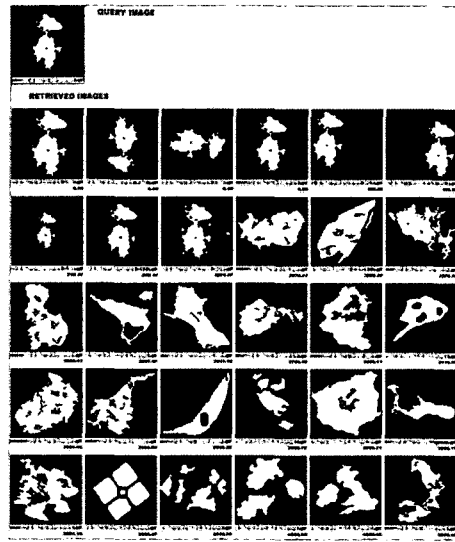
그림 12는 CSS 방법과 원형 투영 방법을 이용한 지형/지물 이미지 검색 결과를 나타낸다. 그림 12(a)에서 CSS 방법이 질의 이미지와 방향/크기/위치는 다르지만 유사한 영역을 갖는 이미지를 높은 순위(1~7위)로 검색하는 것을 볼 수 있다. 그러나 세그멘테이션 과정에서 발생하는 잡음과 영역의 분리로 인해 외곽선이 변화된 경우의 이미지는 낮은 순위(각각 12, 30위)를 차지한다. 그림 12(b)의 원형 투영을 이용한 실험에서는 질의 이

미지와 유사한 이미지를 모두 높은 순위(1~9)로 검색할 수 있다. 이와 같이 CSS 방법과 달리 원형 투영은 지형/지물 이미지에서 빈번히 발생할 수 있는 세그멘테이션 오류에 강인함을 알 수 있다.

그림 13은 4절에서 언급한 참조 방사선의 사용 유무의 경우에 이미지 검색의 정확도를 보이기 위한 실험이다. 그림 13(a)는 참조 방사선을 이용한 원형 투영의 이미지 검색 실험이고 그림 13(b)는 참조 방사선을 사용하지 않은 실험이다. 두 실험에서 질의 이미지와 유사한 이미지 모두 높은 순위로 검색되었다. 4절에서도 언급했듯이 참조 방사선의 사용은 두 이미지의 유사성을 계산할 때 시간 복잡도를 줄여준다. 즉 참조 방사선을 사용

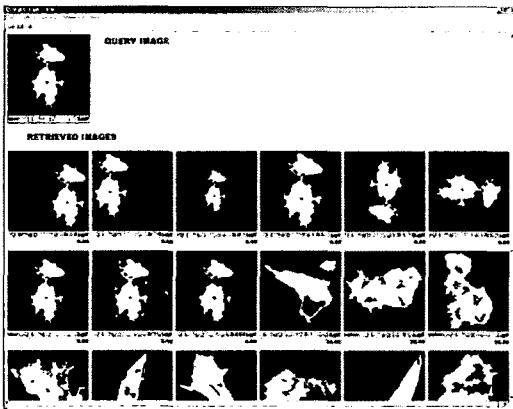


(a) CSS

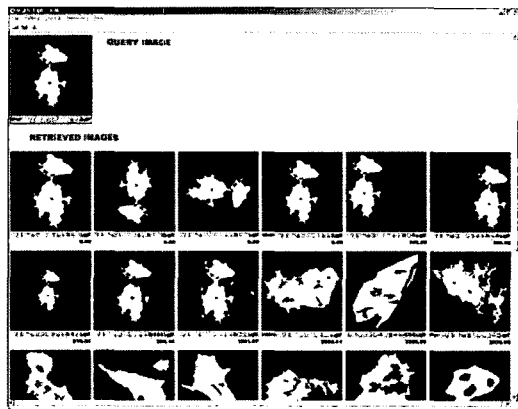


(b) 원형 투영

그림 12 CSS와 원형 투영을 이용한 이미지 검색 실험



(a) 참조 방사선을 사용한 경우



(b) 참조 방사선을 사용하지 않은 경우

그림 13 참조 방사선의 사용 유무에 따른 이미지 검색 실험

하지 않은 경우  $O(k^2)$ 이지만 사용할 경우  $O(k)$ 이 된다. 이 실험을 통해 참조 방사선의 사용은 검색 결과의 정확도에 큰 영향을 미치지 않고 이미지 비교 시간, 더 나아가 검색 시간을 줄일 수 있다는 것을 알 수 있다.

지금까지 기존 방법들과의 비교 등 다양한 실험을 통해 원형 투영의 장점을 보였다. 지금부터는 원형 투영의 성능을 객관적으로 보이기 위해 대용량의 이미지 데이터베이스에서 CSS와의 비교 실험 결과를 보일 것이다. 본 실험에서는 원형 투영과 CSS 만을 비교 실험 할 것이다. 그 이유는 직각 투영은 방향 불변의 성질이 없고, 시그니처는 원형 투영의 방사선 길이만을 이용하는 방법이고 원형 투영이 이미 직각 투영과 시그니처의 특징을 모두 포함하기 때문에 원형 투영이 항상 직각 투영과 시그니처 보다 좋은 성능을 가지기 때문이다. 이는 그림 10과 그림 11의 실험에서 이미 확인된 결과이다.

먼저 그림 14는 2000장의 지형/지물 이미지에 대해 원형 투영과 CSS의 비교 검색 실험을 나타낸다. 그림 14의 실험에서 질의 이미지와 유사한 데이터베이스 이미지는 20장이 있고, 두 실험 결과로 27장의 이미지가 각각 검색된다. 두 방법의 성능은 결과 이미지 안에서 유사한 이미지의 재현율(recall)에 따른 정확도(precision)를 계산함으로써 비교할 수 있다. 그림 15는 그림 14의 두 실험에서 검색된 유사한 이미지의 정확도-재현율 그래프를 나타낸다.

그림 15에서 보듯이 재현율에 따른 정확도가 CSS보다 원형 투영에서 높음을 알 수 있다. 이것은 원형 투영이 CSS보다 유사한 이미지를 결과 이미지 내의 높은 순위로 검색할 수 있음을 의미한다. 그리고 원형 투영과 CSS 실험에서 재현율이 각각 90%, 70%이고, 이를 통해 저장된 유사한 이미지가 원형 투영 실험에서 더 많

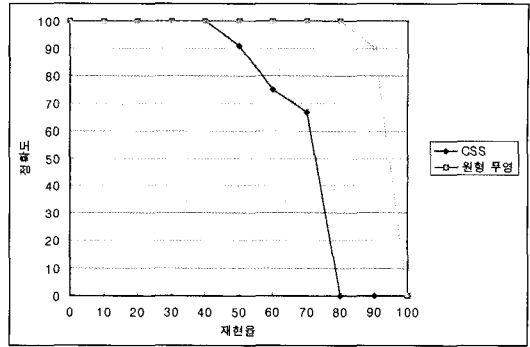


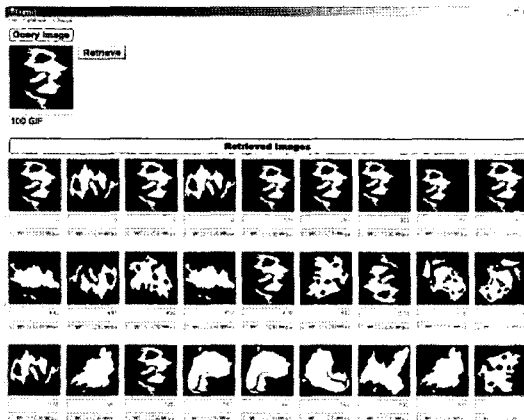
그림 15 그림 14 실험의 정확도-재현율 그래프

이 검색됨을 말해준다. 따라서 그림 14의 실험 결과로부터 지형/지물 이미지 검색에서 원형 투영이 CSS보다 더 나은 성능을 갖는 것을 알 수 있다.

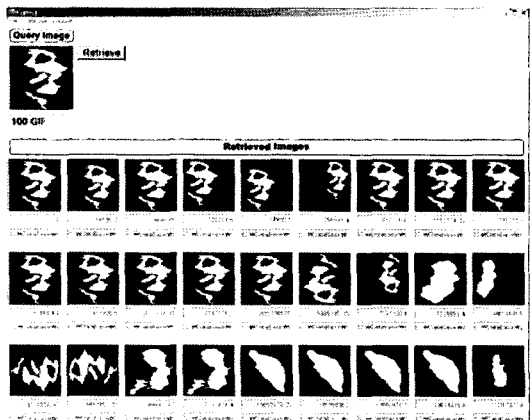
다음으로 그림 16은 MPEG-7 Core Experiment CE-Shape-1 Part B에서 사용된 1399장의 이미지 집합에 대해 원형 투영과 CSS의 성능을 비교한 실험이다 [10].

그림 16의 실험에서도 그림 14의 실험과 마찬가지로 질의 이미지와 유사한 저장된 이미지는 20장이 있고, 두 실험 결과로 27장의 이미지가 각각 검색된다. 그림 17은 그림 16의 두 실험 결과에 대한 정확도-재현율 그래프이다.

그림 17에서 그림 15의 그래프와 마찬가지로 CSS보다 원형 투영에서 재현율에 따른 정확도가 높음을 알 수 있다. 그리고 CSS 실험에서 재현율은 60%인 반면 원형 투영 실험에서는 75%로 저장된 유사한 이미지가 더 많이 검색됨을 알 수 있다. 결국 그림 14와 그림 16의 실험으로부터 원형 투영이 CSS보다 형태 특징을 효과적으로 추출함을 알 수 있다.

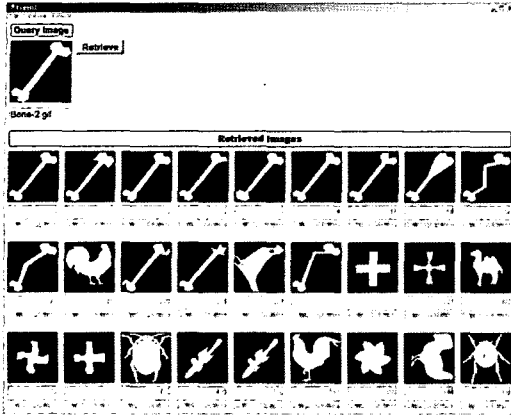


(a) CSS

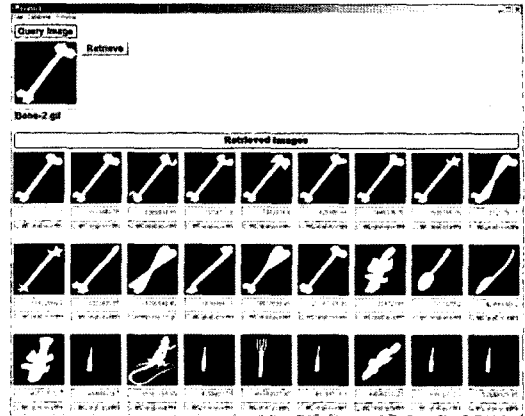


(b) 원형 투영

그림 14 대용량의 지형/지물 이미지에 대한 이미지 검색 실험



(a) CSS



(b) 원형 투영

그림 16 CE-Shape-1의 Part B 이미지에 대한 이미지 검색 실험

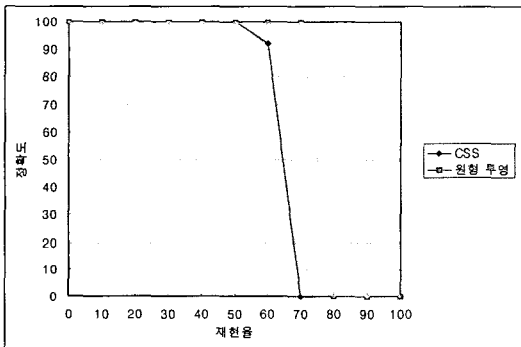


그림 17 그림 16 실험의 정확도-재현율 그래프

## 6. 결론

본 논문은 지형/지물 이미지에 적합한 형태 특징 추출 방법인 원형 투영을 소개하였다. 이 방법은 영역의 무게 중심에서 외곽으로 투영을 하면서 얻어진 특징들을 저장함으로써 형태 정보를 추출한다. 각 방향에서 투영을 하면서 얻어진 특징들은 다음과 같다: (1) 영역 화소의 수 (2) 영역의 변화 수 (3) 무게 중심에서 가장 멀리 떨어진 영역 화소까지의 거리. 원형 투영 방법의 각 특징들은 이전의 방법들에서 추출된 특징들보다 더 포괄적임을 보였고, 이러한 관점에서 영역의 출과 비연결성을 표현할 수 있다. 뿐만 아니라 간단한 참조 각도가 거리 계산 시간을 줄여 그 결과 선행 시간에 유사성을 측정할 수 있었다.

원형 투영은 실험을 통해서 확인된 다음과 같은 장점을 갖는다: (1) 질의 이미지와 유사한 이미지를 방향/크기/위치에 상관없이 검색할 수 있다, 그리고 (2) 세그멘테이션 오류에 강인하고 홀이 있거나 연결되지 않은 영역들을 갖는 이미지의 특징을 추출할 수 있다.

## 참고 문헌

- [1] Gudivada, V.N., Raghavan, V.V., "Content-based Image Retrieval Systems," IEEE Computer, Vol. 28, pp. 18-22, September 1995.
- [2] Milan, S., Vaclav, H. and Roger B., Image Processing, Analysis and Machine Vision, PWS Publishers, 1999.
- [3] Smith, J.R., Shih-Fu, C., "VisualSEEK : a fully automated content-based image query system," Proceedings of the fourth ACM international conference on Multimedia, pp. 87-98, 1997.
- [4] Vasconcelos, N., Kunt, M., "Content-based retrieval from image databases: current solutions and future directions," Proceedings of International Conference on Image Processing, Vol. 3, pp. 48-53, October 2001.
- [5] Manjunath, B.S., Philippe S. and Thomas S., Introduction to MPEG-7 Multimedia Content Description Interface, John Wiley and Sons Publishers, June 2002.
- [6] Gonzalez, R.C., Woods, R.E., Digital Image Processing, Prentice Hall Publishers, 2001.
- [7] Kpalma, K., Ronsin, J., "A multi-scale curve smoothing for generalised pattern recognition (MSGPR)," Proceedings of Seventh International Symposium on Signal Processing and Its Applications, Vol. 2, pp. 427-430, July 2003.
- [8] Mokhtarian, F., Mackworth, A.K., "A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, pp. 789-805, August 1992.
- [9] Mokhtarian, F., and Bober, M., The Curvature Scale Space Representation: Theory, Applications, and MPEG-7 Standardization, Kluwer Academic Publishers, 2002.

- [10] Latecki, L. J., Shape Data for the MPEG-7 Core Experiment CE-Shape-1., <http://www.cis.temple.edu/latecki/TestData/mpeg7shapeB.tar.gz>, 2002.



권 용 일

2004년 중앙대학교 전자전기공학부(학사). 2006년 중앙대학교 전자전기공학부(석사). 2006년~현재 LG전자 OLED연구소 연구원. 관심분야는 이미지 데이터베이스 및 영상 처리



박 호 현

1987년 2월 서울대학교 계산통계학과 학사. 1995년 8월 한국과학기술원 정보및통신공학과 컴퓨터공학전공 석사. 2001년 2월 한국과학기술원 전자전산학과 전산학전공 박사. 1987년 1월~2003년 2월 삼성전자 정보통신연구소 수석연구원. 2003년 3월~현재 중앙대학교 전자전기공학부 조교수. 관심분야는 멀티미디어 스트리밍, 멀티미디어 정보검색, 시공간 데이터베이스



이 석 룡

1984년 연세대학교 기계공학과 학사. 1993년 연세대학교 산업공학과 전자계산전공 석사. 2001년 한국과학기술원 정보및통신공학과 컴퓨터공학전공 박사. 1984년~1995년 한국IBM 소프트웨어 연구소 선임연구원. 1995년~2001년 안산1대학 조교수. 2002년~현재 한국외국어대학교 산업정보시스템공학부 부교수. 관심분야는 멀티미디어 데이터베이스, 데이터 마이닝, 정보검색 등



정 진 완

1973년 서울대학교 공과대학 전기공학과(학사). 1983년 University of Michigan 컴퓨터공학과(박사). 1983년~1993년 미국 GM 연구소 전산과학과 선임연구원 및 책임연구원. 1993년~현재 한국과학기술원 전산학전공 부교수 및 교수. 관심분야는 XML, 시맨틱웹, 멀티미디어 데이터베이스, 스트림 데이터 및 센서 네트워크 데이터베이스