SIFT를 이용한 유사 위성 영상 검색
(Similar Satellite Image Search using SIFT)

김 정 범 † 정 진 완 †† 김 덕 환 ††† 김 상 희 †††† 이 석 룡 †††††
(Jung-Bum Kim) (Chin-Wan Chung) (Deok-Hwan Kim) (Sang-Hee Kim) (Seok-Lyong Lee)

요 약 영상 데이타의 증가로 인해 여러 분야에 걸쳐 유사 영상 검색에 대한 수요는 계속해서 증가하고 있으며, 좀 더 정확한 유사 영상 검색을 위해 내용기반영상검색 기술 분야에 대한 많은 연구가 이루어지고 있다. 내용기반영상검색에서는 유사한 영상을 검색하기 위해 색상, 모양, 질감 등의 영상 자체의 내용을 사용한다. 하지만 위성 사진과 같이 색상 정보의 사용이 어렵고 영상 자체가 매우 복잡한 영상의 경우는 내용기반영상검색 기술을 이용해도 유사성 검색이 쉽지 않다. 그 이유는 색상 정보의 사용이 어려워 모양 정보를 이용하려면 영상 세그멘테이션을 통해 물체의 모양을 분리해내야 하는데 영상이 매우 복잡한 경우 영상 세그멘테이션의 결과가 좋지 않기 때문이다.

따라서 본 논문에서는 영상 세그멘테이션을 사용하지 않고 유사성 검색을 하는 방법을 제안하여 유사 위성 영상을 검색하고자 한다. 이를 위해 영상 세그멘테이션을 사용하지 않는 모양 기술자인 SIFT keypoint descriptor를 기반으로 영상의 유사도를 정의하여 유사성 검색을 한다. 실험을 통해 제안한 방법을 이용함으로써 색상 정보의 사용이 어렵고 영상 자체가 매우 복잡한 위성 영상의 유사성 검색의 정확성을 향상 시킬 수 있음을 보인다.

키워드 : 내용기반영상검색, 위성 영상, 유사성 검색, 영상 세그멘테이션

Abstract Due to the increase of the amount of image data, the demand for searching similar images is continuously increasing. Therefore, many researches about the content-based image retrieval (CBIR) are conducted to search similar images effectively. In CBIR, it uses image contents such as color, shape, and texture for more effective retrieval. However, when we apply CBIR to satellite images which are complex and pose the difficulty in using color information, we can have trouble to get a good retrieval result. Since it is difficult to use color information of satellite images, we need image segmentation to use shape information by separating the shape of an object in a satellite image. However, because satellite images are complex, image segmentation is hard and poor image segmentation results in poor retrieval results.

In this paper, we propose a new approach to search similar images without image segmentation for satellite images. To do a similarity search without image segmentation, we define a similarity of an image by considering SIFT keypoint descriptors which doesn't require image segmentation. Experimental results show that the proposed approach more effectively searches similar satellite images which are complex and pose the difficulty in using color information.

Key words : content-based image retrieval, satellite image, similarity search, image segmentation

· 본 연구는 한국과학기술원 영상정보특화연구센터를 통한 국방과학연구소의 연구비 지원으로 수행되었습니다.

† 비 회 원 : KAIST 전산학과
jbkim@islab.kaist.ac.kr
†† 종신회원 : KAIST 전산학과 교수
chungcw@kaist.edu
††† 정 회 원 : 인하대학교 전자공학과 교수
deokhwan@inha.ac.kr
†††† 정 회 원 : 국방과학연구소 Image Intelligence Processing Group principal researcher
falcon@add.re.kr
††††† 종신회원 : 한국외국어대학교 산업경영공학부 교수
sllee@hufs.ac.kr

논문접수 : 2007년 12월 10일
심사완료 : 2008년 6월 3일

Copyright@2008 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.
정보과학회논문지: 데이타베이스 제35권 제5호(2008.10)

1. 서 론

위성 사진, 지형 사진 등의 영상 데이타는 지금 이 순간에도 지속적으로 생산되고 있으며, 그 양은 계속해서 증가할 것이다. 또한 최근 디지털 카메라의 보급 등으로 인해 누구나 손 쉽게 영상 데이타를 생산해 낼 수 있기 때문에 영상 데이타의 양은 급격하게 증가하고 있다. 이렇게 증가하고 있는 수많은 영상 데이타 속에서 원하는 영상을 빠르고 정확하게 찾기 위해서는 유사 영상 검색 기술이 필수적이다. 영상 데이타의 증가로 인해 여러 분야에 걸쳐 유사 영상 검색에 대한 수요는 계속해서 증가하고 있다.

많은 경우에 유사 영상 검색을 위하여 영상의 제목과 같은 서지정보 또는 주석(annotation)을 이용하는 방법이 사용되고 있다. 이는 기존의 문서 검색에서 검색의 대상을 영상으로 바꾼 것이라 할 수 있다. 하지만 영상은 문서와 여러 가지 다른 특징을 가지고 있기 때문에 이 방법은 몇 가지 단점을 가지고 있다. 영상의 주석은 제작자의 주관적 입장에서 만들어 지기 때문에 동일한 영상에 대해서도 각자 다른 주석을 작성할 수 있다. 또한 검색을 위해서는 모든 영상에 대해서 서지정보나 주석이 작성되어야만 한다. 위와 같은 단점을 극복하기 위해서는 영상의 서지정보나 주석이 아닌 영상 자체의 정보(content)를 바탕으로 영상을 검색해야 한다. 영상 자체의 정보로는 색상, 질감, 모양 등이 있으며, 이러한 영상 자체의 정보를 바탕으로 영상을 검색하는 방법을 내용기반영상검색(CBIR, Content Based Image Retrieval)이라고 한다. 내용기반영상검색 기술을 통해서 영상에 대한 서지, 주석 정보를 이용하지 않고 영상 자체만으로 영상의 검색이 가능해지므로 이 분야에 대한 많은 연구가 이루어지고 있다.

대표적인 내용기반영상검색 시스템으로는 QBIC[1], VisualSEEK[2], MARS[3] 등이 있으며 국내에서도 COSMIC[4], SCARLET[5] 등의 내용기반영상검색 시스템이 개발되어 있다.

앞서도 말했듯이 내용기반영상검색에는 색상, 질감, 모양 등의 영상 자체의 정보가 사용되는데 영상이 그레이스케일인 경우 색상의 정보는 이용할 수 없게 된다. 따라서 질감이나 모양의 정보를 사용해야 하는데 모양 특징을 사용하기 위해서는 영상에서 영상 내의 object를 분리해서 object의 모양 특징을 추출해야 한다. 영상에서 object를 분리해내기 위해서는 영상 세그멘테이션 기술이 필요하다. 하지만 색상 정보를 사용하기 힘들면서 영상 자체가 매우 복잡한 위성 사진 또는 지형 사진의 경우 모양 특징을 이용하기 위해 영상 세그멘테이션을 해도 영상에서 object를 정확히 분리하기가 어렵다. 따라서 모양 특징을 제대로 추출하기가 힘들고 이는 유사성 검색의 정확도 저하로 이어지게 된다. 그렇기 때문에 위성 사진 또는 지형 사진처럼 영상 자체가 매우 복잡하고 색상정보를 사용하기 힘든 경우에는 영상 세그멘테이션을 사용하지 않고 유사성 검색을 할 수 있다면 유용할 것이다. 따라서 본 논문에서는 지형 사진 또는 위성 사진 같은 색상 정보의 사용이 힘들고 매우 복잡한 영상의 유사성 검색에 초점을 맞추고 이러한 유사성 검색의 정확성을 높이는 목적을 갖는다. 또한 본 논문에서는 오프라인에서 유사성 검색을 하는 응용을 고려하였다. 따라서 빠른 수행 시간보다는 정확한 결과에 더 비중을 둔다.

색상 정보의 사용이 어렵고 매우 복잡한 영상을 대상으로 유사성 검색을 할 때 영상 세그멘테이션을 사용하지 않기 위해서는 그레이스케일 영상에서 영상 세그멘테이션을 하지 않고 특징을 추출하는 기술이 필요하다. 영상 세그멘테이션을 하지 않으면서 그레이스케일 영상에서 특징을 추출하는 방법으로는 SIFT(Scale-Invariant Feature Transform)[6]가 있다. SIFT는 영상에서 특징을 추출할 때 영상에서 특징이 되는 특징점(keypoint)을 찾고 그 주변에서 특징(feature)을 추출하므로 영상 세그멘테이션 기술을 필요로 하지 않는다. SIFT를 제안한 [6]에서는 특징(feature)을 추출하는 방법과 함께 영상에서 추출해낸 특징을 이용해 영상을 매칭하는 방법 또한 제안하였다. SIFT의 매칭 방법은 object recognition에 사용하기 위해 고안되었기 때문에 동일한 영상의 검색에는 효과적이지만 유사성 검색에는 적합하지 않다. 또한 SIFT는 keypoint 주변 픽셀만을 고려하므로 false match가 다소 있을 수 있다. 따라서 본 논문에서는 SIFT의 특징 추출 방법은 그대로 이용하고 SIFT에서 추출된 특징을 기술하는 SIFT keypoint descriptor를 이용해 영상의 유사도를 판단하여 이를 기반으로 유사성 검색을 하는 방법을 제안하고자 한다.

SIFT의 매칭 방법을 사용하지 않고 제안한 방법을 통해 영상의 유사도를 정의하여 유사성 검색을 한 경우 영상 세그멘테이션을 이용하여 모양 특성을 추출하는 경우나, SIFT 매칭 방법을 그대로 이용한 것보다 2배 이상의 좋은 유사성 검색의 결과를 보이는 것을 확인할 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문을 설명하기 위한 관련 연구에 대해 살펴보고 3장에서 이 논문에서 제안한 방법에 대해 설명한다. 4장에서는 실험을 통한 결과에 대해 설명하고 5장에서 본 논문의 결론을 맺는다.

2. 관련 연구

본 장에서는 본 논문과 관련되는 몇 가지 연구에 대

해서 설명한다.

2.1 영상 세그멘테이션을 이용한 내용기반영상검색

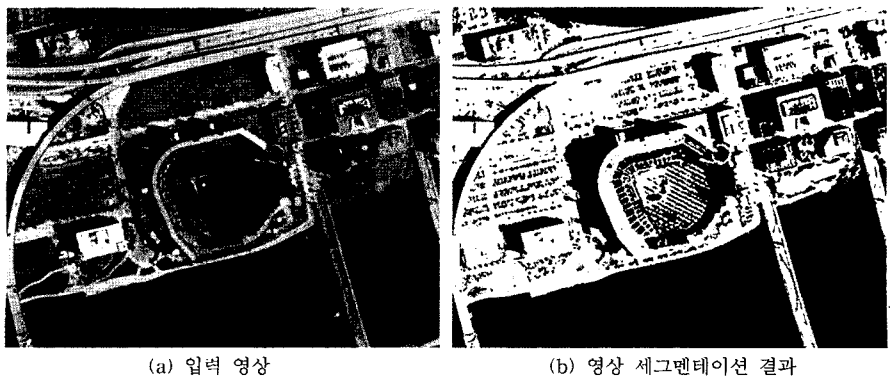
영상 데이타베이스에서 질의 영상의 object와 유사한 object가 포함된 영상을 검색하기 위해서는 object의 색상, 모양, 질감 등의 정보를 이용하는 [1-3] 등과 같은 내용기반영상검색 기술을 이용할 수 있다. 내용기반영상검색에서는 영상에서 색상, 모양, 질감 등의 특징을 추출하고 추출된 특징을 비교하여 두 영상 간의 유사도를 계산한다. 찾고자 하는 object가 영상에 포함되어 있는 경우 유사한 object를 찾기 위해서는 object를 영상에서 분리하는 영상 세그멘테이션 기술이 필요하다. 따라서 영상의 특징을 추출하기에 앞서 영상 세그멘테이션이 선행되고 그 결과로 나온 세그먼트를 대상으로 특징을 추출한다. 그렇기 때문에 영상 세그멘테이션의 결과가 유사성 검색의 결과에 큰 영향을 미친다.

그러나 본 논문에서 고려하고 있는 위성영상과 같이 색상 정보의 사용이 힘들고 영상 자체가 복잡한 경우는 영상 세그멘테이션을 수행하여 영상에서 object를 성공적으로 분리해내는 것이 어렵다. 또한 위성영상의 경우 영상에서 object가 차지하고 있는 부분이 작기 때문에 영상 세그멘테이션이 더욱 어렵다. 잘못된 영상 세그멘테이션의 결과로는 object가 다른 세그먼트에 붙어있는 경우, 여러 세그먼트로 조각난 경우, 세그먼트로 발견되지 않는 경우 등이 있다. 여러 가지 영상 세그멘테이션 방법이 있으나 복잡한 영상에서 다양한 형태의 object를 성공적으로 분리해내는 데에는 한계가 있다. 그림 1에서 영상 세그멘테이션의 결과가 좋지 않은 예를 볼 수 있다. 그림의 예에서는 object의 형태가 잘못 나왔을 뿐만 아니라 object가 다른 세그먼트에 붙어있어 결과가 좋지 않다. 이와 같이 영상에서 object를 제대로 분리해내지 못한 경우 영상에서 모양 특징을 추출하는 [7,8]와 같은 기술 들을 이용해서 특징을 추출해도 좋은 유사성 검색의 결과를 얻기 힘들다.

2.2 SIFT(Scale-Invariant Feature Transform)

SIFT는 영상에서 특징(feature)을 추출하는 방법의 일종으로 영상에서 특징을 추출하기 위해 먼저 영상에서 특징이 되는 특징점(keypoint)을 찾고 그 주변 픽셀의 gradient로 히스토그램을 구성한 keypoint descriptor를 특징(feature)으로 사용한다. SIFT에서는 영상에서 keypoint를 찾기 위해 영상에 다양한 스케일 σ를 가진 Gaussian function을 적용한다. 그렇게 얻어진 다양한 스케일의 smooth된 영상에서 DoG(Difference of Gaussian)[6]영상을 계산한다. 다양한 스케일과 사이즈에 대해서 DoG 영상을 찾기 때문에 SIFT의 특징(feature)은 scale-invariant한 특성을 가질 수 있게 된다. 이렇게 얻어진 여러 DoG 영상에서 각 픽셀을 그 픽셀의 주변 8개의 픽셀과 양쪽으로 인접한 두 스케일의 DoG 영상에서 그 픽셀과 위치가 동일한 픽셀의 주변 9*2개 픽셀의 명도를 비교한다. 만약 어떤 픽셀이 주변의 26개 픽셀 전부보다 명도가 가장 높거나 가장 낮다면 이 픽셀을 keypoint로서 찾는다. 그 후 몇 가지 과정을 거쳐 영상의 keypoint를 결정한다.

영상에서 keypoint를 찾은 후에는 각 keypoint에 대해서 특징(feature)을 추출한다. 특징을 추출하기 위해서 그림 2과 같이 각 keypoint의 주변 영역 픽셀의 gradient의 크기와 방향을 구한다. 그 후 각 주변 픽셀의 gradient 값을 기반으로 이 keypoint의 오리엔테이션을 할당하고 할당된 오리엔테이션을 기준으로 모든 gradient의 방향을 회전한다. 이렇게 모든 gradient의 방향을 할당된 오리엔테이션을 기준으로 회전함으로써 rotation-invariant한 특성을 갖춘다. 최종적으로 keypoint의 특징을 추출하기 위하여 그림 2와 같이 keypoint 주변 영역을 4×4 영역으로 나누고 각 영역의 gradient를 합산하여 오리엔테이션 히스토그램을 구성한다. 이렇게 구성



(a) 입력 영상 (b) 영상 세그멘테이션 결과

그림 1 영상 세그멘테이션 결과

한 오리엔테이션 히스토그램을 keypoint descriptor라고 하며, keypoint descriptor가 keypoint의 특징(feature)이 된다. 그림 2에서는 한 keypoint의 descriptor로써 주변 8×8 영역에 대해서 4개의 오리엔테이션 히스토그램을 사용하지만 실제로는 주변 16×16 영역에 대해서 16개의 오리엔테이션 히스토그램을 사용한다. 따라서 keypoint의 특징값(feature value)인 keypoint descriptor는 128차원을 갖는다.

SIFT를 제안한 [6]에서는 object recognition에 적합한 matching 방법을 제안하였다. 이러한 matching 방법은 동일한 영상의 검색에는 적합하나, 본 논문에서 해결하고자 하는 문제인 위성 영상을 대상으로 유사한 object를 검색하는 데에 그대로 사용하는 것에는 한계가 있다. 따라서 본 논문에서는 SIFT의 특징 추출 방법은 그대로 이용하고 SIFT에서 추출된 특징을 기술하는 SIFT keypoint descriptor를 이용해 영상의 유사도를 판단하여 이를 기반으로 유사성 검색을 하는 방법을 제안하여 SIFT를 유사성 검색에 이용하고자 한다.

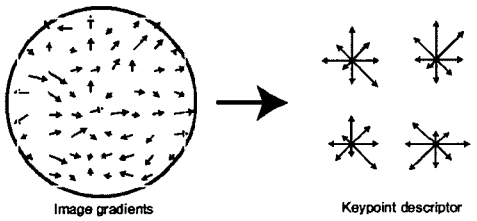


그림 2 keypoint descriptor의 생성

2.3 Angular Radial Transform shape descriptor

모양 특징을 나타내는 shape descriptor는 크게 region-based shape descriptor와 contour-based shape descriptor로 나눌 수 있다. Contour-based shape descriptor의 경우 object의 contour를 특징으로 기술하기 때문에 object가 한 개의 닫혀진 contour로 이루어져 있는 경우에 적절하다. Region-based shape descriptor의 경우 여러 region 또는 contour로 이루어진 object는 물론 hole이 포함된 object의 특징 또한 기술할 수 있어 단순한 모양은 물론이고 복잡한 모양을 기술하는 데에도 적절하다. Angular Radial Transform (ART) [7,8]는 MPEG-7의 표준으로 지정된 region-based shape descriptor이다. ART shape descriptor는 ART계수의 집합으로 이루어진다. ART는 극좌표상에서 단위 원 안에서 정의되는 2차원의 복잡한 변환이다.

$$F_{nm} = \langle V_{nm}(\rho,\theta), f(\rho,\theta)\rangle = \int_0^{2\pi}\int_0^1 V^*_{nm}(\rho,\theta) f(\rho,\theta)\rho d\rho d\theta$$

여기서, F_{nm}은 차수가 n과 m인 ART 계수를 나타내며, $f(\rho, \theta)$은 극좌표 상의 영상 함수, $V_{nm}(\rho, \theta)$은 ART 기저함수를 나타낸다. ART 기저함수는 아래와 같이 각도와 반경 두 축의 방향을 따라 분리되어 정의 된다.

$$V_{nm}(\rho,\theta) = A_m(\theta)R_n(\rho)$$

각도 기저함수와 반경 기저함수는 다음과 같이 정의된다.

$$A_m(\theta) = \frac{1}{2\pi}\exp(jm\theta)$$

$$R_n(\rho) = \begin{cases} 1 & n = 0 \\ 2\cos(\pi n\rho) & n \neq 0 \end{cases}$$

ART를 이용한 shape descriptor는 ART 계수의 정규화된 절대값을 이용하며, rotation, scale invariant한 특성을 갖는다.

ART가 상당히 좋은 성능을 나타내는 모양 기술자이기는 하나, 2.1 장에서 언급하였듯이, 영상 세그멘테이션이 어려운 위성영상을 대상으로는 ART와 같은 모양 기술자를 사용하여도 좋은 유사성 검색 결과를 얻는 것이 어렵다. 따라서 본 논문에서는 영상 세그멘테이션을 하지 않고 유사성 검색을 하는 방법을 제안하고자 한다.

3. SIFT Keypoint Descriptor를 이용한 유사성 검색

SIFT는 특징을 추출하기 위해 영상의 gradient를 이용하고 영상 세그멘테이션을 필요로 하지 않는 특징이기 때문에 앞서 고려했던 색상 정보의 사용이 어려우면서 매우 복잡한 영상에 쓸 수 있는 특징 추출 기법으로 적당한 것으로 생각된다. SIFT를 제안한 [6]에서는 한 영상의 object와 같은 object가 다른 영상에 존재하는 지를 판별하기 위한 object recognition에 적합한 매칭 방법을 제안하였다. SIFT의 매칭 방법은 다음과 같다. 먼저 여러 keypoint 중 찾고자 하는 object의 keypoint와 descriptor값이 가장 가까운 keypoint와 두 번째로 가까운 keypoint를 찾는다. 그 후 가장 가까운 keypoint의 descriptor값의 거리(distance)와 두 번째로 가까운 keypoint의 descriptor값의 거리의 차이가 크면 해당 keypoint와 descriptor값이 가장 가까운 keypoint를 매칭된 것으로 판단한다. SIFT의 매칭 방법은 찾고자 하는 object가 포함된 영상의 모든 keypoint와 모든 비교할 keypoint의 거리를 계산해야 하므로 계산 비용이 크다. 이러한 계산 비용은 query keypoint와 descriptor값이 가까운 keypoint를 찾아야 하는 본 논문의 방법 역시 공통적으로 가지고 있다. 그러나 [6]에서 인덱스를 사용해 이러한 계산 비용을 절감했듯이 본 방법에도 인덱스를 사용함으로써 계산 비용을 절감할 수 있을 것으로 생각된다. 이 때 두 keypoint의 descriptor값의 거리

는 Euclidean distance를 이용한다. 이런 식으로 object가 포함된 영상의 모든 keypoint와 매칭되는 keypoint를 찾고 몇 가지 과정을 거쳐 최종적으로 object가 매칭되었는지를 결정한다. 이와 같이 SIFT의 매칭 방법은 object recognition을 고려하여 고안되었기 때문에 유사성 검색에는 적합하지 않다. 또한 SIFT는 keypoint 주변의 픽셀만을 고려하므로 false match가 다소 있을 수 있다. 여기서 false match라 함은 query keypoint가 유사한 object에 위치하고 있지 않는 keypoint와 잘 못 매칭되는 것을 의미한다.

따라서 본 논문에서는 SIFT의 특징 추출 기법은 그대로 이용하여 영상에서 특징을 추출하면서 SIFT의 매칭 방법을 이용하지 않고 SIFT를 이용해 추출해낸 특징을 기술한 SIFT keypoint descriptor를 기반으로 영상 데이타베이스 내의 각 영상의 유사성을 정의하여 유사성 검색을 할 수 있도록 한다. 즉 SIFT의 특징 추출 방법을 통해 얻어진 SIFT keypoint descriptor를 이용하여 유사성 검색을 하는 방법을 제안하고자 한다. 또한 keypoint pruning을 통해 false match된 keypoint들을 제거하여 유사성 검색의 정확성을 향상시킨다.

그림 3의 예제를 통해 기본적인 개념과 과정을 간략히 설명하고, 그 후에 각 과정에 대해 자세히 살펴보자. 그림 3의 왼쪽 영상은 질의 영상이고 오른쪽 영상은 데이타베이스 내의 영상이다.

질의가 요청되면 그림 3과 같이 각 query keypoint에 대해 데이타베이스 영상에서 keypoint descriptor값이 가까운 m개의 keypoint를 찾는다. 이 예제에서는 m=3으로 가정하겠다. Query keypoint는 각각 □, △, ○라고 하자. 오른쪽의 영상에서 query keypoint와 같은 모양의 keypoint들이 각 query keypoint와 descriptor값이 가까운 keypoint들이고 아래의 숫자는 가까운 순서를 나타낸다. 즉 □1은 query keypoint □과 가장 descriptor값이 가까운 keypoint이고 □2는 두 번째로 가까운 keypoint이다. Query keypoint □는 가장 가까운 descriptor를 가진 keypoint □1이 유사한 object에 제대로 위치하고 있지만 △1과 ○1은 각각 다른 object에 위치하고 있다. 즉 false match가 일어난 경우라고 볼 수 있다. 이러한 false match를 없애기 위해 keypoint pruning을 거쳐 유사한 object에 위치하지 않을 것으로 생각되는 keypoint들을 pruning한다. Pruning의 결과로 밀집되어 있지 않은 △1과 ○1이 pruning될 수 있다. pruning되어 있지 않은 keypoint중 가장 가까운 keypoint와 query keypoint를 대응시키면 비로소 그림과 같이 모든 query keypoint가 유사한 object의 keypoint와 대응되게 된다. 이 때 각 대응된 keypoint 쌍의 descriptor의 distance를 계산하고, 모든 대응의 distance의 평균으로 오른쪽 영상의 distance를 정의하게 된다. SIFT keypoint descriptor는 keypoint 주변 영역만을 고려하기 때문에 유사한 object가 없음에도 불구하고 △1나 ○1과 같이 흩어진 유사한 keypoint가 존재할 수 있다. Keypoint pruning을 통해 이런 false match된 keypoint를 제거할 수 있으므로 좀 더 정확한 유사성 검색이 가능해진다.

이제 본격적으로 SIFT keypoint descriptor를 이용한 유사성 검색에 대해 알아보자.

SIFT의 keypoint descriptor를 이용해서 유사성 검색을 하는 과정을 다음과 같이 간략히 설명할 수 있다.

(1) Pruning circle 설정

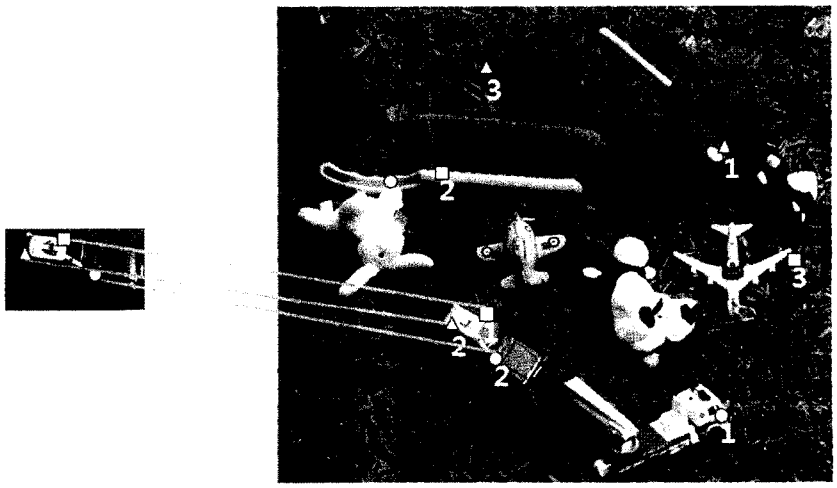


그림 3 질의 영상이 있을 때 데이타베이스 내의 영상의 유사도를 계산하는 예

(2) 영상의 distance 설정
  a. query keypoint의 closest set 탐색
  b. Keypoint pruning
  c. Representative keypoint 설정
(3) Image ranking

pruning circle의 설정 단계에서는 keypoint pruning의 기준이 되는 원인 pruning circle을 설정하고 영상의 distance 설정 단계에서는 각 query keypoint와 keypoint descriptor 값이 가까운 keypoint들을 찾고, 각 query keypoint와 대응되는 keypoint를 찾아 그 들의 평균 distance로 영상의 distance를 설정한다. 이 때 false match를 줄이기 위해 keypoint pruning 과정을 거쳐 대응되는 keypoint를 찾는다. 마지막으로 각 영상을 정의된 distance에 따라 랭킹한다.

각 단계를 설명하기에 앞서 사용되는 notation과 정의에 대해 살펴보자. 표 1에 본 논문에서 사용하는 notation과 정의가 설명되어 있다.

표 1 Notation과 정의

Q	the set of query keypoints
q_i	i th query keypoint in Q
I	an image in databases
C_i	the closest set of q_i
C_i'	the closest set without the the farthest keypoint
r_i	the representative keypoint of C_i
RemainedSet	the set of keypoint to be remained after pruning
$\|.\|$	cardinality of a set
$D(.,.)$	distance between two elements or two sets
m	$\|C_i\|$

그럼 지금부터 각 단계에 대해 살펴보자.

3.1 Pruning circle 설정

Pruning circle이란 질의 영상 내에서 가장 큰 원으로 keypoint를 pruning하는 기준이 된다. 이 원 내에 특정 수 k개 이상의 keypoint가 존재하면 이 원 내의 keypoint들은 pruning하지 않는다. Pruning의 기준으로 원을 사용함으로써 rotation invariant한 pruning이 가능하게 된다. Pruning circle의 지름은 질의 영상의 넓이와 높이 중 작은 값으로 설정되며, 그림 4를 통해 pruning circle의 설정 과정을 볼 수 있다.

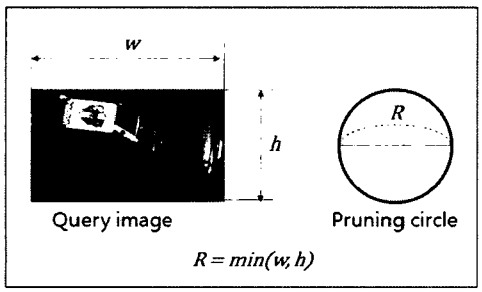


그림 4 Pruning circle의 설정

3.2 영상의 distance 설정

앞선 예에서 살펴보았듯이 영상의 distance를 설정하기 위해서는 먼저 각 query keypoint와 descriptor값이 가까운 몇 개의 keypoint들을 찾고, 찾아진 keypoint들을 대상으로 keypoint pruning을 하여 pruning 되지 않은 keypoint들과 query keypoint들을 대응시켜 각 대응의 distance의 평균으로 영상의 distance를 설정한다. 본 논문에서 가까운 keypoint들을 찾을 때에는 Euclidean distance를 사용하며, 영상 간의 similarity measure로써는 본 장에서 설정하는 영상의 distance간의 Euclidean distance를 사용한다.

3.2.1 Query keypoint의 closest set 탐색

SIFT의 매칭의 경우 각 query keypoint와 descriptor값이 가장 가까운 keypoint와 두 번째로 가까운 keypoint를 찾는다. 하지만 본 논문의 방법에서는 각 query keypoint에 대해 여러 개의 descriptor값이 가까운 keypoint를 찾으며, 이러한 keypoint들의 집합을 closest set이라고 한다. Closest set의 정의는 다음과 같다.

정의 1. 주어진 수 m에 대해서, query keypoint q_i의 closest set C_i은 어떤 영상 I에서 q_i와 descriptor값이 가장 가까운 m개의 keypoint의 집합이다.

한 query keypoint와 한 영상 내의 모든 keypoint와의 descriptor간의 distance를 계산함으로써 한 query keypoint의 closest set을 찾을 수 있다. Closest set을 찾는 데에 있어 closest set 내의 keypoint의 수 m을 정하는 것이 중요하다. 그렇다면 지금부터 m을 구해보자.

m의 계산

m을 계산하기 위해 먼저 영상 내에 n개의 random point가 있을 때 임의의 pruning circle 내에 k개 이상의 point가 포함될 확률을 구해보자. (여기서 random point는 영상 내에 random하게 찍은 한 점을 말한다.) 이 확률이 작음에도 불구하고 pruning circle 안에 k개 이상의 keypoint가 포함된다면 해당 영역에는 유사한 object가 있을 가능성이 높을 것이다. 따라서 이 확률을 작게 하는 n을 구하고 이를 바탕으로 m을 계산한다.

확률 변수 X가 있을 때,

X: 임의의 pruning circle 안에 포함되는 point의 수라고 하자.

영상에 random으로 한 개의 point를 찍었을 때 임의의 pruning circle에 포함될 확률 p는 다음과 같이 계산할 수 있다.

$$p=\frac{size\ of\ pruning\ circle}{database\ image\ width \times database\ image\ height}$$

영상에 n개의 random point를 찍을 때 임의의 pruning circle 안에 x개의 point가 포함될 확률은 $P(X=x)$로 나타낼 수 있으며, 주어진 수 n, 확률 p에 대해서

$$P(X=x)=\binom{n}{x}p^x(1-p)^{n-x}, x=0,1,...,n$$

이다. 따라서, $X\sim b(n, p)$, 즉 확률 변수 X는 이항 분포를 따름을 알 수 있다. 구하고자 했던 영상 내에 n개의 random point가 있을 때 임의의 pruning circle 내에 k개 이상의 point가 포함될 확률은 $P(X \geq k)$로 나타낼 수 있고 이항 분포를 이용하여 쉽게 계산할 수 있다.

앞서도 말했듯이 m을 계산하기 위해 $P(X \geq k)$를 작게 하는 n을 구해야 한다. 이러한 n의 값을 구하기 위해 작은 확률 값 $P\tau$를 파라미터로 받고 $P(X \geq k) \leq P\tau$를 만족하는 가장 큰 n을 구한다. 가장 큰 n의 값을 n_1이라 하자. 구하고자 했던 m은

$$m = \left\lceil \frac{n_1}{|Q|} \right\rceil + 1$$

이 된다. n_1개의 point가 영상 내에 있을 수 있으므로 n_1을 query keypoint의 수로 나눠 한 closest set에 포함될 수 있는 keypoint의 수를 구하고 closest set의 모든 keypoint가 pruning될 때를 대비하여 1을 더한다. 이 때 추가된 1개의 keypoint는 pruning에서 배제되며 query keypoint q_i의 closest set에서 추가된 한 개의 keypoint를 제외한 set을 C'_i라고 하자. 즉 keypoint pruning을 할 때에는 C'_i에 포함된 keypoint들만을 고려한다. 따라서 지금부터 keypoint pruning과 관련해서 keypoint라고 부르는 것은 특별한 언급이 없으면 C'_i에 포함된 keypoint를 말한다.

k의 계산

k는 pruning의 기준이 되는 수로 pruning circle 내에 k개 이상의 keypoint가 있으면 이 영역 내의 keypoint들은 pruning에서 제외된다. k는 다음과 같이 계산한다.

$$k=\left\lceil |Q| \times \alpha \times \frac{size\ of\ pruning\ circle}{size\ of\ query\ image} \right\rceil, 0<\alpha \leq 1$$

α는 파라미터로 0보다 크고 1보다 같거나 작은 실수이다. 질의 영상과 동일한 영상이 아닌 경우 비슷한 영역 위에 query keypoint의 수보다 적은 수의 C'_i에 포함된 keypoint가 존재할 수 있으므로 어느 정도 비율 이상의 C'_i에 포함된 keypoint가 있으면 pruning을 하지 않아야 비슷한 영역 위에 위치하는 keypoint를 pruning하지 않을 수 있다. 따라서 query keypoint의 수에 pruning circle의 면적과 query image의 면적의 비율을 곱하고 특정 비율 α를 곱하여 k를 구한다. 또한 α를 곱하므로 해서 어느 정도의 scale invariant한 pruning이 가능해진다. 예를 들어 질의 영상의 object보다 2배 정도 큰 아주 유사한 object가 어떤 영상에 있다면 α를 0.5 이하로 설정하면 유사한 object에 위치한 keypoint를 pruning하지 않을 수 있다. SIFT에서는 3개 이상의 keypoint가 군집해 있는 경우 object가 존재할 가능성이 있는 것으로 판단한다. 따라서 사용자가 파라미터 α를 제공하지 않는 경우에는 기본적으로 k의 값이 3으로 설정된다.

3.2.2 Keypoint pruning

각 query keypoint의 closest set을 찾은 후에는 false match를 제거하기 위해 유사한 object에 위치하지 않을 것으로 생각되는 keypoint를 pruning하게 된다. Keypoint pruning에서는 C'_i에 포함된 keypoint들만을 고려하므로, keypoint pruning과 관련해서 keypoint라고 부르는 것은 특별한 언급이 없으면 C'_i에 포함된 keypoint를 말한다.

앞서도 말했듯이 pruning circle 내에 k개 이상의 keypoint가 존재하면 그 영역 내의 keypoint는 pruning하지 않는다. Keypoint를 pruning하기 위해서 k개 이상의 keypoint를 포함하는 모든 pruning circle 내에 있는 keypoint들을 찾아야 한다. 즉 pruning하지 말아야 할 keypoint를 모두 찾은 후 나머지 keypoint들을 pruning한다.

표 2의 keypoint pruning 알고리즘을 통해서 k개 이상의 keypoint를 포함하는 모든 pruning circle 내에 있는 keypoint들을 찾을 수 있다. Keypoint pruning 알고리즘은 각 keypoint에 대해서 그 keypoint를 포함하면서 k개 이상의 keypoint를 포함하는 pruning circle이 있는지를 탐색한다. 이를 위해 각 keypoint를 중심으로 후보 영역(candidate region)을 찾는다 후보 영역은 해당 keypoint를 중심으로 pruning circle의 지름을 반지름으로 하는 원형의 영역이며, 해당 keypoint를 포함하는 모든 pruning circle이 존재할 수 있는 영역이다. 후보 영역을 찾은 후에는 중심 keypoint를 포함하면서 후보 영역 내에 존재하는 keypoint들을 포함할 수 있는 pruning circle보다 작거나 같은 최소 경계 원을 찾는다. 결과적으로 후보 영역 내의 모든 keypoint들은 한 개 이상의 최소 경계 원 내에 위치하게 된다. 이렇게 찾아진 최소 경계 원 중에 k개 이상의 keypoint를 포함하는 최소 경계 원이 있으면 그 경계 원 내에 있는 모든 keypoint를 *RemainedSet*에 포함시킨다. 따라서 *RemainedSet*에는 pruning되지 말아야 할 keypoint가 포함된다. *PointList*를 유지함으로써 새로운 최소 경계 원이 추가될 때 현재까지 다른 최소 경계 원에 포함되었던 keypoint들이 새로운 최소 경계 원에 포함될 수 있

는지를 판단할 수 있다. 최소 경계 원(Minimum Bounding Circle, MBC)을 찾는 데에는 여러 개의 점을 기반으로 빠르고 정확하게 최소 경계 원을 구할 수 있도록 제안된 [9]의 알고리즘을 기반으로 하는 Smallest Enclosing Balls of Points 소스 코드[10]를 사용한다.

Keypoint pruning을 제대로 하기 위해서는 표 2의 keypoint pruning 알고리즘의 correctness가 중요하므로 정리 1을 통해 알고리즘의 correctness를 증명하겠다.

정리 1. Keypoint pruning 알고리즘은 k개 이상의 keypoint를 포함하는 모든 pruning circle 내에 있는 keypoint를 찾는다.

증명. keypoint pruning 알고리즘이 k개 이상의 keypoint를 포함하는 pruning circle 안에 있는 어떤 keypoint k_1을 찾지 못했다고 가정하자. 그렇다면 keypoint pruning 알고리즘이 k_1을 포함한 k개 이상의 keypoint를 포함하는 pruning circle을 찾지 못한 것이다.

이런 경우 k_1을 중심으로 하는 후보 영역에는 k개 이상의 keypoint가 존재하고, k_1을 포함한 k개 이상의 keypoint를 포함하는 pruning circle이 존재한다. Keypoint pruning 알고리즘을 사용하면 각 keypoint에 대해서 후보 영역을 찾고, 후보 영역의 중심 keypoint를 포함하면서 후보 영역 내에 존재하는 keypoint들을 포함할 수 있는 pruning circle보다 작거나 같은 최소 경계 원들을 찾는다. k_1을 중심으로 하는 후보 영역 안에 k개 이상의 keypoint를 포함하는 pruning circle이 존재하므로 k_1을 중심으로 하는 후보 영역 안에는 pruning circle 보다 작거나 같으면서 k개 이상의 keypoint를 포함하는 최소 경계 원이 있다. 이 최소 경계 원은 keypoint pruning 알고리즘에 의해 찾아지므로 k_1을 포함한 k개 이상의 keypoint를 포함하는 pruning circle은 찾아진다.

이는 가정과 모순이 되므로 정리 1은 성립한다. □

3.2.3 Representative keypoint 설정

Keypoint pruning을 수행한 후 각 query keypoint q_i의 closest set C_i에는 *RemainedSet*에 포함된 keypoint와 그렇지 않은 keypoint가 존재하게 된다. (pruning에서 제외했던 C'_i에는 포함되지 않고 C_i에만 포함되는 한 개의 keypoint는 모든 C'_i의 keypoint가 pruning될 때를 대비한 것이므로 이미 *RemainedSet*에 있는 것이라고 생각한다.) Representative keypoint는 한 query keypoint와 대응되는 keypoint로 앞선 그림 3의 예제에서 query keypoint △의 representative keypoint는 △2이다. 다시 말하면 representative keypoint는 한 query keypoint의 closest set에서 *RemainedSet*에 포함된 keypoint들 중 query keypoint와 descriptor 값이 가장 가까운 keypoint이다. Representative keypoint의 정의는 다음과 같다.

정의 2. query keypoint q_i의 representative keypoint r_i는 C_i에서 *RemainedSet*에 포함된 keypoint 중 q_i와 가장 descriptor값이 가까운 keypoint이다.

어떤 query keypoint q_i의 representative keypoint r_i는 다음과 같이 구할 수 있다.

$$r_i = \min_{keypoint \in (C_i \cap RemainedSet)} (D(q_i, keypoint))$$

각 query keypoint q_i의 representative keypoint r_i를 찾은 후에는 영상의 distance를 정의해야 한다. 영상의 distance는 각 query keypoint와 그의 representative keypoint의 descriptor값의 distance를 평균 낸 값으로 아래와 같이 정의한다.

$$D(Q, I) = \frac{1}{|Q|} \sum_{i=0}^{|Q|-1} D(q_i, r_i)$$

3.3 Image ranking

영상들의 distance를 정의한 후에는 distance에 따라 영상들을 오름차순으로 정렬하고 사용자가 원하는 수만큼의 상위 영상들을 리턴한다. 이와 같이 데이타베이스 내의 모든 영상의 distance를 정의하면 데이타베이스의 크기가 커지면 속도가 느려질 수 있다. 따라서 인덱스를 사용해 후보 영상을 지정하고 그들만의 distance를 구해 결과를 제공하는 방법을 사용하여 scalability를 향상 시킬 수 있다.

4. 실험 및 결과

표 2 Keypoint pruning algorithm

```
Algorithm KeypointPruning
1. for each keypoint in the closest set of each query keypoint
2.     calculate a candidate region centered on the corresponding keypoint.
3.     find MBCs which cover all keypoint in C'i and in the candidate region.
4.     if there are MBCs containing more than k keypoints
5.         add all keypoints in the MBC to RemainedSet.
6. prune keypoints which are not included in Remainedset.
```

본 장에서는 실험과 비교를 통하여 본 논문에서 제안한 방법의 성능을 기존의 방법들과 비교하여보자.

4.1 실험 환경 및 데이타

실험에 사용된 시스템은 Pentium D 2.80GHz의 CPU와 2.00GB의 RAM의 사양으로 구성되어 있으며 운영체제로는 Windows XP를 사용하였다. 프로그래밍 플랫폼은 마이크로소프트사의 Visual Studio .NET을 사용하였다.

실험을 위해 사용한 dataset은 523개의 위성 사진으로 이루어진 dataset이다. 523개의 위성 사진은 Google Earth[11]를 통해 직접 캡쳐하였으며, 523개의 영상 가운데 22개의 야구 경기장 영상과 30개의 미식축구 경기장 영상이 포함되어 있으며, 야구 경기장을 찾는 것을 목표로 하였다. 영상의 크기는 520픽셀×410픽셀 정도이다. 이 dataset의 특징은 위성 사진으로 이루어져있기 때문에 영상이 매우 복잡하다. 따라서 이 dataset는 본 논문에서 해결하고자 했던 유사성 검색의 환경과 일치한다.

실험에서는 제안한 방법의 성능을 평가하기 위해서 제안한 방법과 두 가지 방법의 성능을 비교한다. 한 가지는 SIFT의 매칭을 이용한 방법이고 다른 한 가지는 ART를 이용한 내용기반영상검색 방법이다. SIFT의 매칭을 이용한 방법은 query keypoint와 영상 데이타베이스의 모든 keypoint를 매칭시키면 동일한 영상 한 개만을 찾을 수 있으므로 query keypoint와 각 영상의 keypoint를 매칭시켜서 여러 개의 영상을 찾을 수 있도록 하였다. ART를 이용한 방법은 특징을 추출할 수 있도록 먼저 각 영상에 region-growing 세그멘테이션을 적용하여 영상에서 object를 분리해내고 그 후 ART를 이용하여 특징을 추출하였다. 질의 영상에서는 제일 큰 세그먼트를 질의 영상의 object로 정하였고, 질의 영상의 object와 영상 데이타베이스 내의 각 영상의 모든 세그먼트와 비교하여 가장 작은 distance를 가진 세그먼트를 각 영상의 object로 정하였다. ART를 이용한 방법과 비교한 이유는 이 논문에서 해결하고자 했던 색상 정보의 사용이 힘들면서 영상 자체가 복잡하여 영상 세그멘테이션이 힘든 영상의 유사성 검색을 할 때 영상 세그멘테이션을 하고 모양 특성을 이용하는 시나리오와 본 논문에서 제안한 방법과 비교하기 위함이다. ART가 [12,13]의 논문 등을 통해 좋은 region-based shape descriptor임이 입증되었고 MPEG-7 표준의 region-shape descriptor로도 지정된 특징 추출 기법이기 때문에, 모양 특징을 추출하는 기법으로서 ART를 사용하였다.

결과를 보여주기 위한 그래프에서 제안한 방법은 Proposed, SIFT 매칭을 이용한 방법은 SIFT matching, ART를 이용한 내용기반영상검색 방법은 CBIR using ART로 나타낸다.

이후의 실험에서 검색의 정확성은 precision과 recall이라는 척도를 사용하여 측정한다. precision은 영상 검색의 결과 중 정확한 영상의 비율을, recall은 정확한 영상 중 검색된 정확한 영상의 비율을 나타내는 값이며 영상 검색의 결과로 나온 영상들을 R, 정확한 결과로 나와야 하는 영상들을 A라고 할 때 precision과 Recall은 다음과 같이 정의된다.

$$Procision = \frac{|R \cap A|}{|R|}$$

$$Recall = \frac{|R \cap A|}{|A|}$$

SIFT의 매칭을 이용한 방법의 경우 유사성 검색의 방법이 아니기 때문에 검색될 영상의 수를 지정하는 것이 힘들다. 따라서 SIFT 매칭방법의 precision과 recall은 조금 다르게 정의하겠다. SIFT 매칭을 했을 때 전체 매칭된 영상들을 M, 정확한 결과로 나와야 하는 영상들을 A라고 하고 검색하고자 하는 영상의 수를 |R|이라고 할 때 SIFT 매칭방법의 precision과 recall을 다음과 같이 정의하였다.

$$Precision = \begin{cases} \frac{|M \cap A|}{|M|} & |R| \le |M| \\ \frac{|M \cap A|}{|R|} & |R| > |M| \end{cases}$$

$$Recall = \begin{cases} \frac{|M \cap A|}{|A|} & |R| \ge |M| \\ \frac{|M \cap A| * |R|}{|M| * |A|} & |R| < |M| \end{cases}$$

Precision은 전체 매칭의 수가 찾고자 하는 영상의 수 |R| 이상인 경우 전체 매칭에 대한 정확한 매칭의 비율로 정의하였다. 즉 전체 매칭의 수만큼 영상을 찾았다 가정하고 비례적으로 precision을 정의하였다. 그렇지 않은 경우는 기존의 precision과 동일하다. Recall은 |R|이 전체 매칭의 수 이상인 경우 기존의 recall과 동일하고 그렇지 않은 경우 전체 매칭에 수에 맞게 비례적으로 recall을 구한다.

4.2 실험 결과

본 실험에서는 위성영상으로 이루어진 dataset을 이용하여 제안한 방법과 SIFT 매칭을 이용한 방법, ART를 이용한 내용기반영상검색 방법의 precision과 recall을 비교한다. 실험에서 사용한 dataset은 위성 사진으로 이루어져 있기 때문에 영상이 매우 복잡하여 세그멘테이션의 결과가 좋지 못하다.

실험에서는 야구 경기장을 질의로 사용하며 8개의 야구 경기장 영상으로 질의를 했을 때의 평균 결과를 비교하였다. 야구 경기장 질의 영상의 크기는 약 50픽셀×50픽셀이다. 그림 5는 8개의 야구 경기장 영상으로 질

의를 했을 때 top10, 20, 22, 30, 40 결과의 평균 precision과 recall을 보여준다. 파라미터는 α=0.3 P_τ=0.001을 사용하였다.

Top22의 결과에서 제안한 방법의 precision과 recall은 약 0.6정도의 결과를 보였으며, 이는 다른 두 방법보다 3배 정도 좋은 결과이다. SIFT 매칭을 이용한 방법의 경우 아주 유사한 영상은 어느 정도 찾을 수 있으나 그렇지 않은 영상의 경우는 잘 찾지 못하므로 검색 영상의 수가 증가하여도 recall의 증가가 거의 없다. ART를 이용한 방법의 경우는 영상 세그멘테이션의 결과에 영향을 받는데 실험에 이용한 dataset의 영상들은 영상에 비해 object의 크기가 작고 영상이 복잡하기 때문에 영상 세그멘테이션의 결과가 만족스럽지 않다. 따라서 ART를 이용한 방법의 결과가 좋지 않은 것으로 생각된다.

그림 6은 제안 방법의 야구 경기장 영상 질의의 결과를 보여준다. 왼쪽 위의 영상이 질의 영상이고 위쪽에 위치하고 왼쪽에 위치할수록 순위가 높은 영상이다.

그림에서 제안한 방법이 야구 경기장 영상 질의에 대해 유사한 야구 경기장 영상을 성공적으로 찾는 것을 볼 수 있다.

그림 7은 파라미터 α의 변화에 따른 유사성 검색의 결과 변화를 보여주고 있다. α가 변화함에 따라 유사성 검색의 결과도 향상되나 어느 시점 이후에는 결과가 나빠지는 것을 볼 수 있다. 그 이유는 α가 증가하면 좀 더 강력한 pruning을 하게 되고, 그로 인해 유사한 object 위에 위치한 keypoint들도 pruning될 수 있기 때문으로 생각된다. 그래프에서는 α의 변화에 따른 topk22의 precision과 recall의 변화를 보여주고 있다.

본 논문에서 고려한 응용에서는 유사성 검색을 오프라인에서 수행한다. 따라서 수행 시간보다는 정확도에 비중을 두고 있다. SIFT를 제안한 [6]에서도 정확도에 무게를 두어 수행 시간에 대한 실험결과를 보이지 않고 있기 때문에 본 논문에서도 수행 시간에 대한 실험결과를 보이진 않았으나 SIFT와 같이 유사한 keypoint를 찾을 때 인덱스를 사용함으로써 효율적인 유사성 검색이 가능할 것으로 생각된다.

5. 결 론

지형 사진 또는 위성 사진과 같이 색상 정보의 사용이 힘들고 영상 자체가 복잡한 영상의 경우 유사성 검색이 쉽지 않다. 그 이유는 모양 특성을 사용하기 위해 영상 세그멘테이션을 하여도 object의 모양을 성공적으로 분리할 수 없어서 모양 특징 추출을 제대로 하기 힘들기 때문이며 결과적으로 유사성 검색의 정확성이 저하된다.

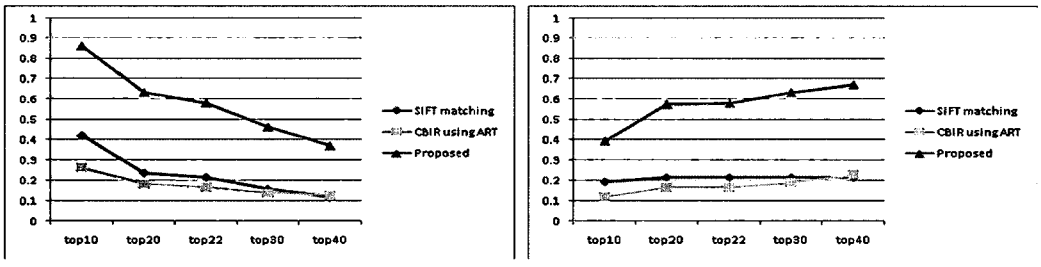
본 논문에서는 색상 정보의 사용이 힘들고 영상 자체가 복잡한 영상을 대상으로 유사성 검색을 할 때, 정확성을 높이기 위해 영상 세그멘테이션을 필요로 하지 않는 특징 기술자인 SIFT keypoint descriptor를 이용해서 유사성 검색하는 방법을 제안하였다. 본 논문에서 제안한 방법은 질의 영상에 대한 pruning circle을 찾고 SIFT keypoint descriptor를 기반으로 각 query keypoint의 closest set을 찾는다. 그 다음 keypoint pruning을 거친 뒤 각 query keypoint에 대응되는 representative keypoint를 찾은 후 영상 데이타베이스 내의 각 영상의 유사도를 계산하여 영상을 랭킹한다. keypoint pruning을 통해 false match를 줄여 좀 더 유사성 검색의 정확성을 향상시킨다.

실험을 통해 위성 영상과 같이 색상 정보의 사용이 힘들고 영상 자체가 복잡한 영상의 유사성 검색을 할 때, 영상 세그멘테이션을 하고 모양 특징을 이용하는 방법이나 SIFT 매칭을 이용하는 방법에 비해 제안한 방법이 더욱 정확한 유사성 검색을 할 수 있음을 보였다.

추후에는 인덱스를 사용하여 본 논문에서 제안한 방법을 이용해 좀 더 효율적인 유사성 검색을 하는 방법을 연구할 생각이다.

참 고 문 헌

[1] Myron Flickner and et. al, "Query by Image and



(a) precision (b) recall

그림 5 야구 경기장 영상으로 질의했을 때의 결과

그림 6 야구 경기장 질의의 결과 영상

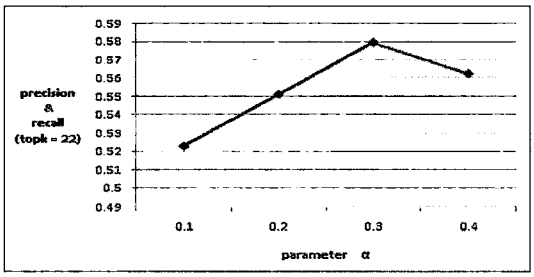


그림 7 α의 변화에 따른 유사성 검색 결과

Video Content: The QBIC system," IEEE Computer, 28(9), pp. 23-32, 1995.

[2] J. R. Smith and S. -F. Chang, "VisualSEEk: A Fully Automated Content-based Image Query System," ACM Multimedia 96,Boston, MA, 1996.

[3] T. S. Huang, S. Mehratra, and K. Ramchandran, "Multimedia Analysis and Retrieval System (MARS) project," Proceedings of the 33rd Annual Clinic on Library Application of Data Processing - Digital Image Access and Retrieval, University of Illinois at Urbana-Champaign, Mar. 1996.

[4] 김덕환, 김시우, 박광순, 이병구, 차광호, 정진완, "COSMIC: 영역 지식과 시각 정보를 이용한 내용 기반 멀티미디어 검색 시스템의 설계 및 구현", 정보과학회 논문지(C), 5권 1호, pp. 14-28, Feb. 1999.

[5] 이동호, 송용준, 김형주, "SCARLET: 웨이브릿 변환을 이용한 내용기반 이미지 검색 시스템의 설계 및 구현", 정보과학회 논문지(C), 3권 4호, pp. 353-364, Aug. 1997.

[6] Lowe, D. G., "Distinctive Image Features from Scale Invariant Keypoints," International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.

[7] MPEG-7 Visual Group, "Text of ISO/IEC 15938-3 /FDIS Information technology. Multimedia content description interface. Part 3 Visual," ISO/IECJTC1/SC29/WG11 N4358, Sydney, July 2001.

[8] B.S. Manjunath, Philippe Salembier and Thomas Sikora, Introduction to MPEG-7: multimedia content description interface, West Sussex, England: John Wiley & Sons, 2002, pp. 238-240.

[9] Bernd Gärtner, "Fast and Robust Smallest Enclosing Balls," Proc. 7th Annual European Symposium on Algorithms (ESA), Lecture Notes in Computer Science 1643 © Springer-Verlag, pp. 325-338, 1999.

[10] http://www.inf.ethz.ch/personal/gaertner/miniball.html

[11] http://earth.google.com

[12] Veltkamp, R. C. and Latecki, L. J., Properties and Performance of Shape Similarity Measures. In Proceedings of IFCS 2006 Conference: Data Science and Classification. July, 2006.

[13] J.P. Eakins, K.J. Riley, J.D. Edwards, Shape feature matching for trademark image retrieval, in: International Conference on Image and Video Retrieval, 2003, pp. 28-38.

김 정 범
2006년 홍익대학교 컴퓨터공학 학사. 2008년 한국과학기술원 전산학과 석사. 2008년~현재 삼성전자 근무. 관심분야는 멀티미디어 데이타베이스, Location based service 등

정 진 완
정보과학회논문지 : 데이타베이스
제 35 권 제 4 호 참조

김 덕 환
정보과학회논문지 : 데이타베이스
제 35 권 제 4 호 참조

김 상 희
정보과학회논문지 : 데이타베이스
제 35 권 제 4 호 참조

이 석 룡
정보과학회논문지 : 데이타베이스
제 35 권 제 4 호 참조