

SIFT를 이용한 위성사진의 정합기법[☆]

A Scheme for Matching Satellite Images Using SIFT

강 석 찬* 황 인 택** 최 광 남***
Suk-Chen Kang Inteck Whoang Kwang Nam Choi

요 약

본 논문에서 우리는 위성 영상에 대하여 객체를 지역화한 접근을 제안한다. 우리의 방법은 서술 벡터에 기반한 특징 정합 방법이다. 객체를 지역화하는 방법은 SIFT(Scale Invariant Feature Transform)를 적용시킨다. 먼저, 위성영상의 키포인트를 찾고, 키포인트의 서술 벡터를 일반화한다. 그리고 서술 벡터간에 유사성을 측정하여 키포인트를 매칭시킨다. 마지막으로, 키포인트의 인접 픽셀값에 가중치를 주어 객체에서 위치를 결정한다. SIFT를 이용한 이 실험은 다양한 스케일과 어파인 변환에 대해 좋은 결과를 산출하였다. 본 논문에서 제안된 방법은 구글 어스의 위성영상을 사용하였다.

Abstract

In this paper we propose an approach for localizing objects in satellite images. Our method exploits matching features based on description vectors. We applied Scale Invariant Feature Transform (SIFT) to object localization. First, we find keypoints of the satellite images and the objects and generate description vectors of the keypoints. Next, we calculate the similarity between description vectors, and obtain matched keypoints. Finally, we weight the adjacent pixels to the keypoints and determine the location of the matched object. The experiments of object localization by using SIFT show good results on various scale and affine transformed images. In this paper the proposed methods use Google Earth satellite images.

☞ Keyword : Satellite Images, SIFT, 위성사진

1. 서 론

최근 위성사진 서비스의 비용이 낮아지면서 이를 이용한 여러 응용서비스들이 개발되고 있는 추세이다. 위성사진의 응용과 관련하여 차량 탐색, 도로 내 교통량 분석 등과 같은 다양한 연구들이 진행되어왔으며, 이를 위한 기반이 되는 이미지 정합 기술과 관련, 정합을 효율적으로 수행하기 위한 여러 연구들도 진행되었다. 특히 물체 검색과 인식 분야에서 연구가 진행되면서 이미지

정합과 관련한 다양한 알고리즘들이 연구되었다.

이미지 정합(Matching)에는 원 이미지와 대상 이미지와의 픽셀 간 유사성 비교를 통한 정합 외에 이미지에 대한 주된 특징이 되는 점들을 찾는 뒤, 이 점들에 대하여 이 점들을 보다 식별이 용이하게(Distinctive) 서술할 수 있는 서술 벡터(Description Vector)를 생성하고, 이 벡터들 간의 유사성(Similarity)을 측정하여 정합을 하는 방법이 있다. 이러한 주된 특징이 되는 픽셀 또는 지점들을 특징점(Feature) 또는 Keypoint라고 한다. 이러한 키포인트 기반의 정합을 위해서는 키포인트의 안정성이 중요한데, 키포인트는 이미지 데이터 내에서 해당 이미지 부분에 대해서 추후 이미지에 대한 어느 정도의 왜곡이나 변환이 있다 하더라도 일정한 방법에 의하여 최대한 다시 관찰 될 수 있는 점들 이어야 한다. 이를 수행하는 알고리즘을 키포인트 디텍터(Keypoint Detector)라고 하며,

* 준 회 원 : 중앙대학교 컴퓨터공학과 대학원 졸업(석사)
sckang@vim.cau.ac.kr

** 정 회 원 : 중앙대학교 대학원 컴퓨터공학과 박사과정
itwhoang@vim.cau.ac.kr

*** 정 회 원 : 중앙대학교 컴퓨터공학부 교수(교신저자)
knchoi@cau.ac.kr

[2009/01/19 투고 - 2009/01/28 심사 - 2009/02/18 심사완료]

☆ 이 논문은 2008년도 중앙대학교 학술연구비 지원에 의한 것임

해당 알고리즘이 이용되는 컨텍스트에 따라 특징 검출기(Feature Detector), 주요 지역 탐색기(Interesting Region Detector)라고도 한다.

Moravec은 코너 탐색기(Corner Detector)를 개발, 스테레오 정합에 이용하였다. Harris 와 Stephens는 이를 이미지와 에지(Edge)의 변화에서도 재검출이 가능한 코너 탐색기로 발전시켰다 [2]. 그 이후 이미지 크기-불변(Scale Invariant) 속성에 대한 다양한 연구들이 있었으며, Mikojajczyk는 Harris의 탐색기를 발전시킨 알고리즘을, Kadir는 특이점 지역 탐색기(Salient Region Detector)를 개발하였고, 각각 객체 인식에서 좋은 성능을 보였다 [3]. Lowe는 스케일 공간(Scale-Space)에서의 극점을 찾는 SIFT 키포인트 탐색기(SIFT Keypoint Detector)를 보였다 [4].

키포인트 탐색과 함께 같이 특정 구조의 서술 벡터(Description Vector)로 표현하는 서술(Descriptor) 알고리즘도 같이 발전하였다. 초기에는 모멘트 기반 불변 속성(Moment Invariant properties)이 쓰였다 [5]. SIFT 서술자(SIFT Descriptor)는 4*4*8 차원의 서술 벡터를 생성하며, 정합과 관련하여 매우 뛰어난 성능을 보인다 [6][7]. 추후 SIFT 서술자를 기반으로 알고리즘을 확장한 PCA-SIFT, GLOH등과 같은 알고리즘들도 개발되었으며 상황에 따라 SIFT보다 좀 더 나은 성능을 보이기도 하나, 전반적으로는 SIFT 와 유사한 성능을 보이는 것으로 알려져 있다.

또한 위성사진을 이용한 응용과 관련하여 기존에 여러 연구들이 있었고, 비슷한 응용으로써 항공사진(Aerial Image)에서의 응용들도 이루어졌다. 전소희는 위성영상을 분석, 교통정보 분석에 응용하는 방법을 제안하였다 [1]. 위성 영상 내에서 NGC(Normalized Gray-level Correlation) 방법하의 템플릿 정합을 적용, 분석 대상이 되는 물체나 부분 영상에 대한 중심 좌표를 찾아내고, 이를 이용하여 자동차나 건물 등의 수 등의 통계적인 데이터를 추출하고 이를 분석하는 방법을 제안하였다. 하지만 템플릿 정합의 경우 점점 더 고해상도 화

되는 이미지를 분석함에 있어서 계산이 많다는 점과 비교 대상이 되는 사진의 방향이나 촬영 위치에 따른 왜곡에 따라 결과가 불안정해질 수 있는 단점이 있었다.

본 논문에서는 이런 영상의 특징 탐색 및 서술을 위한 여러 가지 기술의 장단점 및 특징을 조사하고, 그 중 최근 대두되고 있는 특징 기반 탐색 방법의 하나로써 SIFT 키포인트를 이용한 위성 영상 검색 시스템의 제안 및 성능 향상을 제안하고자 한다.

2. 정합알고리즘

2.1 키포인트 탐색

SIFT 키포인트 탐색기는 스케일 공간상에서 극점을 찾는 알고리즘이다. 스케일 공간상에서의 극점은 이미지의 스케일 변화나 이미지 변환에 대하여도 다시 탐색될 가능성이 높은 지점으로 알려져 있다.

Lindeberg은 다양한 가정 하에서 가능한 스케일 공간을 생성하는 커널 함수는 가우시안 함수임을 보였다 [4]. 함수 G 는 가우시안 필터로 다음과 같이 정의된다. 여기서 σ 는 scale factor가 된다.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (1)$$

* 연산은 컨볼루션(convolution)연산이라 할 때, 이미지 L 은 원본 이미지 $I(x,y)$ 에 대하여 가우시안 필터가 적용된 이미지로 다음과 같이 정의된다.

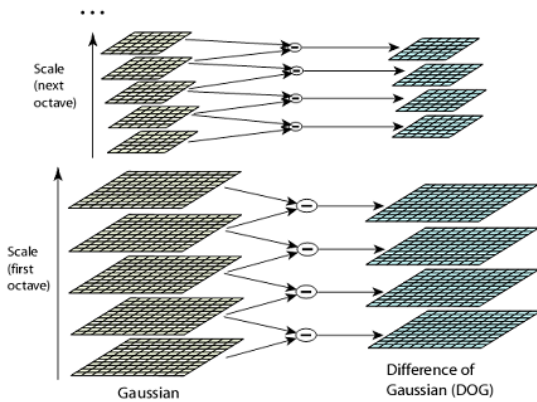
$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2)$$

가우시안 이미지의 차연산 이미지(Difference of Gaussian)인 D 는 다음과 같이 일정한 배수의 가우시안 필터가 적용된 이미지 사이의 차이로 정의된다.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (3)$$

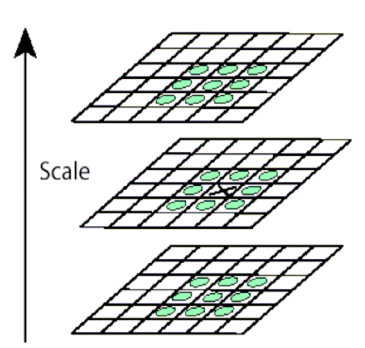
스케일 공간이론과 관련한 Lindeberg의 연구에서 Lindeberg는 σ^2 를 factor로 가지는 라플라시안(Laplacian)의 일반화(Normalization)는 스케일-불변 속성(scale invariance)를 위해 필요한 요소임을 밝혔다. 보다 상세한 실험적인 비교에 따르면, $\sigma^2 G$ 의 극대점과 극소점은 그라디언트(Gradient)나 헤시안(Hessian), 헤리스 코너(Harris corner) 함수와 같은 다른 이미지 함수와 비교하여 가장 안정한 이미지 특징(Feature)을 만들어냄을 밝혔다 [6]. 그리고 이는 곧 다음과 같이 가우시안 이미지의 차 연산 이미지로 근사할 수 있음을 밝혔다.

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \nabla^2 G \quad (4)$$



(그림 1) Gaussian 차에 의한 영상

이 함수들을 기반으로 스케일 공간을 구축하는 방법은 다음 그림 1과 같다. 그림 1의 왼쪽 편의 이미지들은 가우시안 필터가 적용된 이미지이며, 오른쪽 편의 이미지들은 인접한 두 가우시안 이미지의 차 연산(difference) 통해 얻어진 가우시안 이미지의 차 연산 이미지이다.



(그림 2) Gaussian 차에 의한 영상

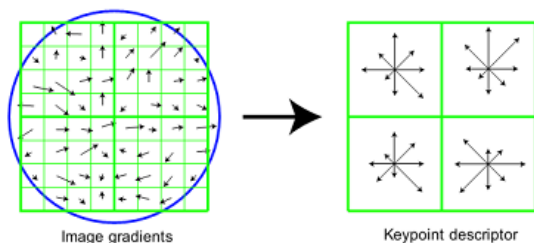
그림 2에서는 스케일 공간에서의 극대점과 극소점을 찾기 위하여 각 포인트들은 현 이미지 내에서의 주변 8개의 점들과 인접한 이미지에서의 각각 9개의 점들과 비교를 한다. 현재 기준이 되는 점이 가운데 이미지의 'X' 위치라 한다면, 'X' 점을 기준으로 인접한 총 26개의 점들에 대하여 비교를 한다. 그 결과, 이 점들 중 모든 인접한 점들에 비해서 가장 작거나 가장 큰 값을 가지는 점이 선택되고 이를 계속 반복하여 극대점과 극소점을 찾게 된다. 대부분의 후보 점들은 첫 번째 검색에서 대부분 제거되므로 이를 위한 검색 비용은 낮다. 검색 결과 선택된 점들은 후처리를 통하여 최종적으로 가장 안정한 점들을 선택하게 되고, 이 점들 주변 값들의 그라디언트 값들을 통하여 이 키포인트의 방향이 결정된다. 그리고 키포인트 주변 지역의 점들에 대한 방향 히스토그램을 생성하게 된다.

방향 히스토그램은 360도를 기준으로 36개의 구역으로 나눈다. 히스토그램에 더해지는 각 샘플들은 그라디언트 값에 의해서 가중치가 곱해지고, 키포인트의 scale의 1.5 배 값을 가지는 σ 로 원형 윈도우에 대하여 가우시안 가중치가 추가로 곱해진다.

2.2. 서술과 정합

키포인트 지점들에 대한 최종 선택이 끝나면, 이를 서술 벡터(Description Vector)로 서술하는 과

정이 진행된다. 일단, 키포인트의 주변 픽셀에 대하여 이미지 그라디언트와 방향들을 얻는다. 방향에 대한 불변(rotate-invariant) 속성을 위해 서술자와 그라디언트 방향은 키포인트 방향에 대하여 상대적으로 회전된다. 효율성을 위해 모든 레벨의 그라디언트들이 미리 계산되며 이는 그림 3에서의 작은 화살표들로 표현된다. 서술자는 모든 방향 히스토그램 엔트리의 값들을 포함하는 벡터의 형태로 형성되며, 그림 3의 오른쪽의 각 화살표들의 길이들은 엔트리의 값들에 대응된다. 아래의 그림 3에서는 2 x 2 배열로 방향 히스토그램을 표현했으나, 실험에 의하면 4 x 4 배열의 히스토그램으로 8개 방향에 대하여 수행한 결과가 가장 좋은 결과를 보이며, SIFT를 이용한 정합을 수행 시 이러한 4 x 4 배열에 8개 방향에 대한 히스토그램의 벡터 ($4 * 4 * 8 = 128$ 차원) 형태로 많이 사용된다. 키포인트 탐색과 서술이 끝난 뒤, 실제 이미지의 정합이 일어난다. 간단한 형태의 이미지 정합은 키포인트 간 유사성 비교에 의하여 일어나는데, 정합 템플릿이 될 이미지의 키포인트의 수가 N 이고 위성사진에서 추출된 키포인트의 수가 M 이라 한다면 총 $N * M$ 의 키포인트 간 정합이 일어난다. 이는 기존 픽셀 대 픽셀 비교에 의한 템플릿 정합 방법에 비하여 적은 수의 연산을 필요로 하는 장점이 있다.



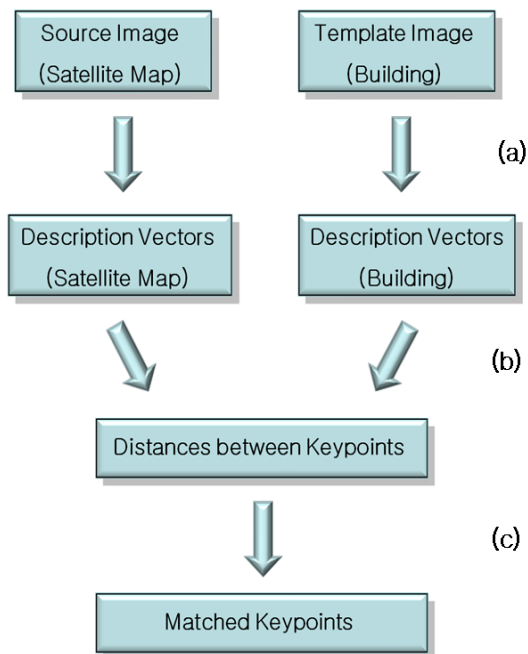
(그림 3) SIFT Descriptor

정합 판정의 경우 매칭 포인트인 하늘색 라인 부분에 대하여 가우시안 모델로 Weight 를 준다. 그리고 최종 정합 판정시에는 각 포인트들에 대하여 총 점수 합산한다. 즉 하늘색 라인 부분이

집중된 곳이 높은 스코어를 받게 되고 정합이 이루어 진다.

3. 설계 및 구현

실험에서의 전반적인 흐름은 그림 4와 같다. 일단 위성사진과 찾는 대상이 되는 템플릿 이미지에 대하여 각각 키포인트 탐색과 서술을 수행한다. 그리고 위성사진에서 검색된 키포인트의 서술 벡터와 템플릿 이미지의 서술벡터에 대하여 유사성을 비교한다. 유사성을 비교한 뒤 스코어 행렬을 구축한다. 키포인트 위치에 따른 스코어 행렬 내 대응되는 위치를 기준으로 인접하는 위치에 대하여 가우시안 함수로 점수를 준다. 각 키포인트들의 유사성을 기준으로 임계값에 따라 가중치를 다르게 주어 점수를 주게 된다. 최종적으로는 스코어 행렬을 탐색, 가장 높은 스코어를 가진 위치를 찾음으로써 객체의 위치를 찾는다.



(그림 4) 시스템 흐름도 : (a) Keypoint detection and description, (b) Calculate Distances between keypoints, (c) Thresholding





실험을 위한 데이터는 구글 어스(Google Earth)를 이용한 위성사진을 이용하였다. 위성사진의 크기는 지상으로부터 얼마나 떨어진 상공에서 찍은 사진인지를 기준으로 세 크기로 분류된다. Large size map은 1000ft 상공에서, Middle size map은 2000ft 상공에서 그리고 Small size map은 3000ft 상공에서 촬영된 영상이다.

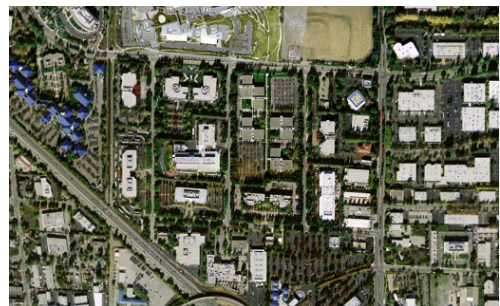
스케일 변화에 따른 결과를 확인하기 위하여 사진 축적에 대하여 변화를 주었다. 그림 5에서 'Large size map' 으로 표현된 샘플은 대축적지도의 형태로 지형 요소들의 사이즈들이 큰 반면, 그림 6에서의 'Small size map' 은 소축적지도의 형태로 지형 요소들의 사이즈들이 전반적으로 작다. 그림 7은 중간 축적도의 지도를 'Middle size map' 으로 정의한다.

실제 정합에 쓰일 이미지는 'Small size map' 이미지 내의 부분 이미지들을 추출하여 템플릿으로 이용하였다. 각 이미지들이 추출된 위치는 다음의 그림 8에의 빨간 사각형 내의 위치들과 같으며, 이 위치들은 추후 정합 때 실제 이미지들을 제대로 정합 했는지 테스트를 시도하는데 이용될 것이다. 그림 8에서 1번 위치의 이미지는 일반적인 건물의 옥상 사진으로 'building 1' 이라 정의한다. 2번 위치의 이미지는 도로 내 교차점으로 'building 2' 라고 정의한다. 'building 2' 에 비하여 이미지 내에 여러 라인들이 존재하여 변화가 심하다. 3번 위치의 이미지는 'building 3' 로 정의한다. 'building 3' 의 특징은 주차장 이미지로, 이미지 내의 변화가 가장 심하다. 4번 위치의 이미지는 건물의 옥상 사진으로 이미지 내의 변화 정도는 'building 1' 과 유사하나 이미지의 크기가 좀 더 크다. 각 템플릿 이미지들의 사이즈는 다음 표 1에 정리되어 있다.

정합은 기존의 픽셀 대 픽셀 비교방식의 템플릿 정합이나 역시 유사한 연산 비용이 드는 NGC 대신, 키포인트의 서술 벡터들 간 유사성을 비교하여 가장 가까운 키포인트를 찾는 키포인트 간 정합으로 진행한다.

(표 1) 'Small size map'에서 추출한 템플릿 이미지

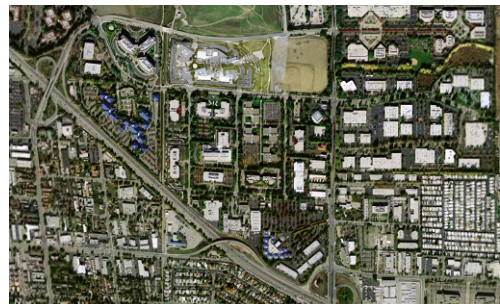
	Building 1	Building 2	Building 3	Building 4
이미지				
폭(pixel)	25	30	30	20
높이(pixel)	25	27	29	54



(그림 5) Large Size Map: 좁은 영역의 대축적지도



(그림 6) Small Size Map: 넓은 영역의 소축적지도



(그림 7) Middle Size Map: 중간축적지도



(그림 8) 'Small size map'에서 템플릿 이미지들의 위치

(표 2) 'Small size map'에서 템플릿 이미지의 픽셀 상의 키포인트 비율

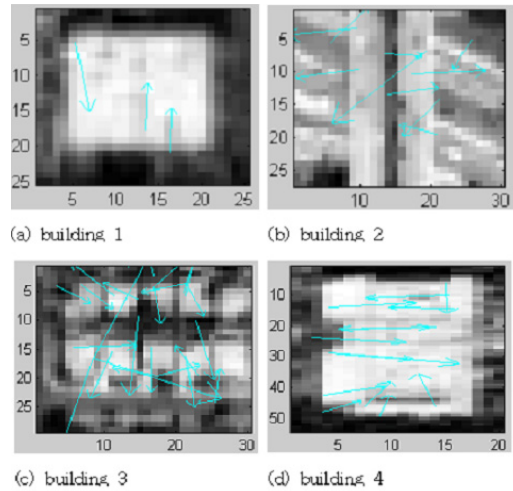
	Building 1	Building 2	Building 3	Building 4
폭(pixel)	25	30	30	30
높이(pixel)	25	27	29	54
키포인트 수	3	10	23	14
키포인트 수 / 폭×높이	0.0048 (3/625)	0.1234 (10/810)	0.2643 (23/870)	0.0130 (14/1080)

4. 실험 및 결과

4.1 키포인트 탐색

소축척지도로 갈수록 이미지 사이즈 대비 표현되는 객체의 수가 많으며, 그에 따른 이미지에서의 변화도 많아진다. 그에 따라 키포인트로 추출되는 지점의 수도 많아진다. 어파인 변환을 적용한 이미지인 'Large-A'와 'Small-A'의 경우 조금 독특한데, 'Large Size Map'의 경우 키포인트의 수가 훨씬 늘어난 반면, 'Small Side Map'의 경우 오히려 키포인트의 수가 줄어드는 현상이 발견되기도 하였다.

정합의 대상이 될 템플릿 이미지들의 키포인트 추출 결과는 그림 9와 같으며, 추출된 키포인트의 수는 아래의 표 2와 같다. 템플릿 이미지의 경우 전체 이미지 사이즈 대비 추출된 키포인트 비율이 일정하지 않은데, 이미지 사이즈가 상대적으로 작고, 이미지 하나에 객체 하나가 표현되면서 상대적으로 이미지 내의 변화 정도가 각 이미지들마다 고르지 않기 때문으로 추측된다. 가장 키포인트가 적은 'building 1'과 가장 키포인트가 많은 'building 3'의 이미지를 비교해 볼 때 그러한 성향에 대해 추측할 수 있다. 또한, 픽셀 수가 많은 'building 4'가 키포인트는 적게 추출된 점으로 볼 때, 이미지의 사이즈 보다는 이미지 내의 변화의 정도가 추출된 키포인트의 수에 더 영향을 미침을 짐작할 수 있다.



(그림 9) 템플릿 이미지에서의 키포인트

4.2 정합

키포인트들에 대한 정합은 일단 위성사진에의 키포인트와 템플릿 이미지의 키포인트의 서술 벡터 간 유클리디안 거리를 측정 한 뒤, 템플릿 이미지의 각 키포인트와의 유클리디안 거리가 가장 낮은 위성사진에의 키포인트 들을 선택한다. 그리고 그 위치들 대비 정합된 키포인트들의 수를 측정하여 실제 템플릿 이미지가 위치한 곳의 키포인트들이 정합된 경우와 그렇지 않은 경우의 수를 측정하였다. 이는 키포인트의 성능을 측정하는 척도가 될 것이다.

이미지의 축적의 변화에 따른 정합 결과 그림 10(a)의 'Small size map'의 경우 정합은 올바르게 약 (350,240) 부근에 진행이 되었으나 그 외의 다른 정합된 키포인트 들이 전반적으로 흩어져 있

는 것을 볼 수 있다. ‘Small size map’의 특징 객체 단위로 생각할 때 ‘Large Size Map’에의 객체보다 상대적으로 적은 수의 키포인트들이 추출된다. 하지만, 전체 이미지로 보았을 때 단위 크기 대비 객체의 수가 많으며, 유사도는 상대적으로 낮은 키포인트들이 정합된다. 그림 10(b)의 경우에서 보듯이 ‘Middle Size Map’가 ‘Small Size Map’보다 그 정합 정도가 좋은 상태를 보인다.

그림 10의 지도에서 하늘색 선은 매칭 포인트 및 방향, 노란색 원은 정합 되었을 때 기준 scale, 빨간색 원은 매칭 표시가 되었고 스코어도 높으나 잘못 매칭한 부분 그리고 흰색 원은 매칭 표시가 되었고 스코어도 높으며 실제로 제대로 매칭된 부분을 각각 표시한다.



(a) ‘Small size map’ 영상의 정합



(b) ‘Middle size map’ 영상의 정합



(c) ‘Large size map’ 영상의 정합
(그림 10) 척도 변화에 따른 정합결과



(a)

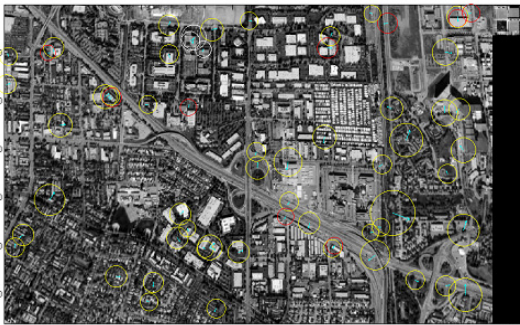


(b)

(그림 11) 어파인변환 된 ‘Large-A’ 영상의 정합결과



(a)



(b)

(그림 12) 어파인변환 된 'Small-A' 영상의 정합결과

그림 10의 결과를 볼때, 원본 이미지인 'Large Size Map'에서 가장 좋은 정합 결과를 보이면서도, 이미지의 크기에 변화가 있는 'Middle Size Map'이나 'Small Size Map'인 경우에도 또한 역시 정합이 잘 이루어졌음을 확인할 수 있다.

또한, 어파인 변환이 적용되어 이미지에 변형이 가해진 이미지인 'Large-A'의 정합 결과 본래의 이미지를 추출하였던 'Large-Size-Map'에서의 정합결과와 유사할 정도로 정합 결과가 좋다. (260, 650) 위치에 잘못 정합이 된 부분이 있으나, 전반적인 키포인트의 가중치는 높지 않으므로 정합 결과에는 영향을 끼치지 않는다. 그림 11(b)의 경우 그림 11(a)와 차이가 있다면, 정합을 위한 템플릿 영상이 도로의 일부로, 이미지 중 중앙 하단으로부터 좌측 중앙 단까지 비슷한 패턴이 등장한다는 점이 있다. 그러한 관계로 상대적으로 다른 예에 비하여 잘못 정합된 키포인트들이 눈에

띈다. 하지만, 최종 정합 판정에서 스코어에 의해 필터링 된다. 그림 10의 정합 결과를 볼 때 어파인 변환에 의한 이미지의 변환이 있음에도 비교적 강건하게 정합을 수행하였음을 알 수 있다.

다음의 그림 12은 'Large-A'의 소 축적 이미지인 'Small-A'에서의 정합 결과이다. 이전의 결과에서도 관찰했듯이 대축적 이미지에 비하여 전반적으로 정합된 키포인트들의 분포가 'Large-A'에 비하여 흩어져있는 것을 볼 수 있다. 하지만, 전반적으로 해당 부분 이미지의 위치에서는 정합된 키포인트들의 분포가 높아 정합이 잘 일어남을 볼 수 있다. Fig 12(b)의 경우 상대적으로 다른 정합결과에 좋지 않음을 관찰할 수 있다. 이는 여러 가지 문제들이 있겠지만, 그림 12(b)에서 사용되는 템플릿 이미지에 따른 특성일 가능성이 높다. 즉, 그림 12(b)에서의 템플릿 이미지의 경우 하나의 이미지 내에 여러 이미지들(작은 사각형들)이 합쳐져 있는 형태의 이미지로, 다른 객체에서 이와 유사한 키포인트가 등장할 가능성이 높다. 하지만, 전반적으로 'Small-A'에서 또한 대부분 본래의 부분 이미지들을 잘 검색하였음을 실험 결과로 확인할 수 있다.

4.3 타 알고리즘과의 비교 분석

기존의 템플릿 정합과 NGC와의 성능 비교를 위하여 위에서의 데이터 셋을 가지고 같은 조건에서 각각의 알고리즘에 대하여 정합을 수행한 뒤, 그 수행시간과 정합 성능을 비교하였다. 테스트가 진행된 컴퓨터의 사양은 펜티엄 4 2.4 GHz CPU, 1 GB RAM을 장착한 PC에서 수행하였다. 수행 시간의 경우 프로그램이 시작되는 시간부터 정합이 끝나는 때의 시간을 측정하였으며, 정합을 위한 템플릿으로 'small size map' 템플릿을 이용, 20번의 반복 테스트 뒤 수행시간의 평균을 계산하였다. 결과는 표 3과 같이 SIFT 기반 정합 방법이 키포인트 탐색 과정을 포함하고도 가장 낮은 평균 수행시간을 기록하였다. 템플릿 정합과 NGC의 경우 둘 다 템플릿 이미지의 크기가 클수

록 수행시간이 오래 걸렸다. 템플릿 정합의 경우 가장 작은 템플릿 이미지인 'building 1'의 수행 시간의 경우 평균 85.42 초를 기록한 반면, 가장 큰 템플릿 이미지인 'building 4'의 경우 123.60 초를 기록하였으며, NGC의 경우 이보다 약간 높거나 비슷한 87.68 초와 123.57 초를 기록하였다. SIFT의 경우 32.05 초와 38.64 초를 기록하였다. 이는 전체 정합 과정 중 데이터 간 정합 부분이 연산 작업에서 가장 많이 걸리는 부분임을 증명한다.

(표 3) 평균수행시간의 비교

	템플릿 정합	NGC	SIFT 기반 정합
평균 수행 시간(sec)	106.75	108.47	32.37

정합성능의 경우 크게 3가지 경우로 테스트하였다. 첫 번째는 명암의 변화에 따른 정합 성능 비교로, 'small size map' 템플릿 이미지에 대하여 명암을 20%, 40% 증가시킨 뒤, 이에 대하여 각각 정합을 수행한 뒤 정합 성공률을 측정하였다. 두 번째는 지도의 축적의 변화에 따른 정합 성능 비교로, 템플릿 이미지는 'small size map' 템플릿을 이용하였으며, 'small size map', 'middle size map', 그리고 'large size map' 지도 이미지를 이용 정합을 수행 뒤 성공률을 측정하였다. 세 번째는 어파인 변환에 대한 정합 성능 비교로, 템플릿 이미지는 'small size map' 과 'large size map'의 템플릿을 모두 이용한 뒤, 'small-A' 와 'large-A' 지도 이미지에 대하여 정합을 수행하고 성공률을 측정하였다. 이에 대한 실험 결과는 다음 표 4와 같다. 명암의 변화에 따른 성능의 경우 NGC가 가장 높은 정합 성능을 보였으며, 그 다음으로 SIFT기반 정합 방법이 두 번째로 높은 정합 성능을 보였다. 스케일 변환과 어파인 변환의 경우 SIFT기반 정합방법이 가장 높은 정합 성능을 보였으며, 성능의 차이가 앞의 두 방법보다 큼을 보였다.

각 정합 알고리즘에 대하여 어파인 변환이 적용된 이미지에 대한 정합 결과에서 템플릿 정합의 경우 비교 대상이 되는 사진의 방향이나 촬영

위치에 따른 왜곡, 조명의 변화 등 여러 요소에 대하여 정합 결과가 불안정해 질 수 있음을 확인하였다.

(표 4) 정합성공률 비교

	템플릿정합	NGC	SIFT기반정합
명암의 변화	75.0	100.0	91.6
스케일 변환 적용	33.3	33.3	83.3
어파인 변환 적용	33.3	38.9	77.8

NGC 또한 조명의 변화에 대해서는 안정적인 정합을 보이나, 크기와 스케일의 변화에 따른 정합 결과에 대한 실험결과와 어파인 변환에 의한 이미지 변환 뒤의 정합 결과에 대한 실험 결과를 볼 때 여러 한계가 있음을 확인할 수 있었다. 또한, SIFT를 적용한 키포인트 정합 방법이 이러한 변화에 대하여 강건한 정합을 보임을 실험을 통하여 확인할 수 있었다.

5. 결론

기존의 위성사진을 이용한 분석 서비스의 경우 검색 이미지의 정합을 위하여 검색 대상 이미지와 위성사진 이미지 간의 픽셀 비교 방법을 기반으로 한 템플릿 정합 방법을 이용하였다. 또한, 이미지 내의 밝기 변화에 따른 정합의 불안정함을 해결하기 위하여 템플릿 정합 방법 기반 하에서 NGC(Normalized Grayscale Correlation) 방법을 같이 적용하기도 하였다. 하지만, 픽셀 비교 기반의 템플릿 정합 접근법에서는 방법의 특성 상 이미지의 크기의 변화나 사진촬영시의 상황에 따른 이미지 변환으로 인한 이미지의 왜곡에 대하여 직접 정합을 수행하기 어려웠다.

본 실험에서는 위성사진 틀에서 시점의 변환을 통해 스케일이 다른 이미지들과 어파인 변환이 수행된 이미지를 생성하고, 이를 구현한 정합 시스템에 적용을 하였다. 앞에서의 여러 실험들을 통해 SIFT를 이용한 키포인트 비교 기반의 이미지 정합 시스템이 그러한 스케일과 이미지의 변화에

도 강건한 이미지 정합이 가능함을 확인하였다.

향후 연구로는 위치와 키포인트의 방향을 기준으로 클러스터링을 진행하여 같은 자세(Pose)를 가지는 객체의 키포인트들로 모일 가능성이 높은 키포인트들을 군집화 한 뒤 이 군집에 대한 연산을 진행하는 방법을 생각해볼 수 있다. 이러한 군집화를 먼저 진행할 경우 상대적으로 실제 정합과 연관이 없는 키포인트 들이 필터링 되는 효과를 가져오리라 생각한다.

참 고 문 헌

- [1] 전소희, 이기원, “템플레이트 매칭기법에 의한 위성영상정보의 교통응용”, 대한원격탐사학회 2005년 춘계학술대회 논문집, 37-40.
- [2] C. Harris and M. Stephens, “A combined corner and edge detector”, In Fourth Alvey Conference, 1988.
- [3] K. Timor, Z. Andrew and B. Michel, “An affine invariant salient region detector,” European Conference on Computer Vision, 2004.[4] T. Lindeberg, ” Scale-space theory : a basic tool for analyzing structures at different scales”, Journal of Applied Statistics, 1994.
- [5] G. D. Lowe, “Distinctive image features from scale-invariant keypoints,” International Journal of Computer Vision, 20: 91-110, 2003.
- [6] K. Mikolajczyk and C. Schmid, “An affine invariant interest point detector,” European Conference on Computer Vision, 2002.[7] K. Mikolajczyk and C. Schmid, “A performance evaluation of local description,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10): 1615-1630, 2005.
- [8] H. Moravec, “Rover visual obstacle avoidance,” International Joint Conference on Artificial Intelligence, 785-790, 1981.

● 저 자 소 개 ●



강 석 천 (Suk-Chen Kang)

2005년 중앙대학교 컴퓨터공학과 졸업(학사)
2007년 중앙대학교 컴퓨터공학과 대학원 졸업(석사)
~현재 패턴인식관련연구
관심분야 : 패턴인식, 영상처리, SE
E-mail : sckang@vim.cau.ac.kr



황 인 택 (Inteck Whoang)

2003년 중앙대학교 컴퓨터공학과 졸업(학사)
2005년 중앙대학교 컴퓨터공학과 대학원 졸업(석사)
~현재 중앙대학교 대학원 컴퓨터공학과 박사과정
관심분야 : 패턴인식, 영상처리
E-mail : itwhoang@vim.cau.ac.kr



최 광 남 (Kwang Nam Choi)

2002년 영국, Univ. of YORK, 공학박사
~현재 중앙대학교 컴퓨터공학과 교수
관심분야 : 패턴인식, 영상처리
E-mail : knchoi@cau.ac.kr