

# 실시간 이미지 시스템을 위한 BLoG 기반의 특징점 검출

박이근\*, 김종민\*, 이용기\*\*

## 요약

본 논문은 실시간 파노라마를 위한 영상의 특징점 검출 방법을 제안한다. 파노라마 연구는 최근 실시간 지역탐색이나 DVR 등에 적용하는 연구가 최근 활발히 진행되고 있다. 특히 특징 점 검출은 파노라마를 이루는 가장 중요한 요소이다. 특징 점 검출을 위해서는 어떠한 명암 변화에도 특징점은 불변이어야 하며 이미지의 크기와 회전이 변화하더라도 불변의 점을 찾아야 한다. 기존 연구방법은 고차원적인 벡터와 많은 후보 점을 선점하기 때문에 연산량이 많고 수행시간이 길어 실시간에 활용하기에는 어려운 점이 있다. 따라서 본 논문은 보다 빠른 실시간 특징 점 검출을 위해 LoG 방법을 비트 단위로 분할 후 결합하는 BLoG 방법을 제안하고 다양한 실험을 통하여 속도와 연산량 그리고 검출 성능에 대하여 비교한다.

## The detection of the feature point in the real-time image system used by BLoG

Yi-Keun Park\*, Jong-min Kim\*, Woong-Ki Lee\*\*

## Abstract

In these days mobile handsets have come to be used at almost every user. The performance improvement of mobile devices and networks have made this trend possible. As a great variety of mobile applications are published, the necessity of running large-scale mobile applications becomes greater than before. To accomplish this, the existing researchers have developed mobile cluster computing libraries like Mobile-JPVM. In this paper, we implement a compute-intensive Animated GIF generating application and its cell phone viewer software using Mobile-JPVM library. We find out by the real execution of our softwares on the KTF handsets that they can sufficiently run on cellular phones. Our Animated GIF generator and its viewer are going to be commercially used for the mobile fashion advertisement systems.

Keywords : 파노라마, 특징점

## 1. 서론

이미지 매칭은 다양한 분야에서 사용되어지고 있다. 이미지 매칭을 위한 첫걸음은 특징영역을 검출해야한다는 것이다. 특징 영역을 검출하는 방법에는 점, 선, 에지, 등등이 있는데 초기에

는 주로 점을 위주로 검출하였다. 점을 검출하기 위한 첫 번째 연산자는 Moravec[1] 연산자부터 시작하게 되었는데 이는 비등방성이라는 제한과 잡음이 많이 있었기 때문에 Harris와 Stephens은 계산하는 비용이 많지만 자기상관함수를 사용하여 성능 면에서 큰 효과를 얻는 특징점 검출 방법을 제시하였다. 이 해리스 코너 검출 (Harris Corner Detection)[2]은 지금도 많이 사용하고 있으나 연산하는 양이 많아 속도가 느리다는 단점을 지니고 있다. 그래서 속도를 개선하는 방법을 고안하여 나온 방법이 Susan[3] 과 빠른 코너 검출 (Fast Corner Detection)[4] 등 다양한 방법들이 있다.

이들은 잡음, 크기, 회전등에 영향을 받기 때

※ 제일저자(First Author) : 박이근  
접수일: 2009년 9월 16일, 완료일: 2009년 12월 31일  
\* 조선대학교 일반대학원 전산통계학과  
provezz@naver.com  
\*\* 조선대학교 컴퓨터통계학과  
▣ 본 연구는 조선대학교 2000학년도 교내연구비 지원에 의해 수행되었음

문에 특징점 검출만 가지고는 이미지 매칭을 하는데 있어 큰 어려움이 있다. 그래서 점이 아닌 선을 검출하고 선이 아닌 영역을 검출하게 되었는데 그 중 크기와 회전, 명암의 변화에도 변하지 않는 아주 강건한 특징영역을 검출하는 SIFT

(Scale Invariant Feature Transform)가 2004년 David G. Lowe [5, 6]에 의해 발표되었다. 이는 이미지 매칭에 있어 가장 성능이 우수하여 지금까지 로봇이나 이미지 유사판별 등 많은 분야에서 사용되어지고 있다. SIFT는 명암, 크기, 회전, 잡음 등에 강한 특징을 찾기 위한 방법으로 가우시안 이미지의 차인 DOG(Difference of Gaussian)를 통하여 후보 점들을 추출한다.

추출된 후보가 되는 점들은 수가 많기 때문에 오차를 줄이고 방향성분을 얻어 비교하여 이미지 매칭을 수행하는 방식이다. 하지만 이 방법은 단일 매칭에서는 좋은 성과를 보여주지만 빠른 수행 능력을 요구하는 실시간 이미지 처리 시스템에는 부적절하다. 또한 이 방법은 특허까지 걸려 있어 실용성에서 제약이 많다. 그래서 고안되는 방법으로 다양한 방법들이 나오게 되었다. 그 중 하나는 고차원 벡터를 좀 더 낮은 차원으로 줄여서 계산량을 줄이는 주성분분석(PCA) 방법을 이용한 PCA-SIFT [7]가 있고 Hessian [9] 방법을 기반으로 하는 매우 강건한 특징점을 좀 더 빠르게 찾는 SURF(Speeded Up Robust Features) [8] 등이 제안되어 지금까지 여러 분야에서 사용되어지고 있다.

하지만 이 것 역시 실시간 이미지 매칭에 약간의 부족함이 있어 좀 더 빠른 매칭 방법에 대한 많은 연구가 진행되는 분야중 하나이다.

실시간 이미지 매칭에서는 항상 빠른 연산하는 양이 요구된다. 이는 매칭 이미지간의 유사한 점을 찾을 때 많은 것보다는 이미지 매칭에 영향을 주는 강건한 점을 확실하게 찾는 방법으로 속도를 줄일 수 있다. 더불어 크기와 회전에 영향을 주지 않고 잡음과 밝기에도 영향을 받지 않기 위한 불변한 서술자를 검출해야 한다. 그래서 본 논문에서는 이미지 피라미드를 통하여 후보 점을 얻는 방식에서 많은 연산하는 양을 줄이기 위하여 매칭 이미지들 간에 강건한 점만 남기고 많은 매칭 점들을 줄이는 방법을 제안하

고자 한다.

## 2. 관련 연구

### 2.1 SIFT 기반의 DOG

SIFT [5, 6]에서는 이미지 매칭을 하기 위해서는 그림1과 같이 4단계를 거쳐 특징(feature) 영역을 뽑아낸다.

1. Find Scale-Space Extrema
2. Keypoint Localization & Filtering
3. Orientation Assignment
4. Create descriptor

(그림 1) SIFT 방법

첫 번째는 명암, 회전, 크기, 잡음에 강건한 점을 찾기 위한 후보 점을 검출하는 단계이다. 여기에서는 DoG(Difference of Gaussian)을 이용한다.

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

<수식 1>

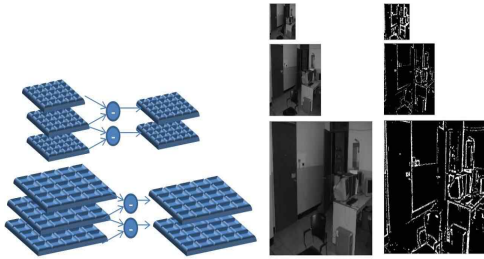
수식1 은 가우시안(Gaussian) 함수를 사용하게 된다. 여기서  $\sigma$  에 따라서 가우시안 곡선 첨도 값이 다르게 나타난다. 수식 2와 같이 가우시안 함수의 임계값인  $\sigma$ 을 다르게 설정하여 나온 각각의 영상을 서로 빼주어서 특징이 되는 점들만 남기게 되는 DoG 영상이 된다. 이는 LoG(Laplacian of Gaussian)의 속도를 향상 방법으로 LoG와 유사한 결과를 가진다.

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ \approx L(x, y, k\sigma) - L(x, y, \sigma)$$

<수식 2 >

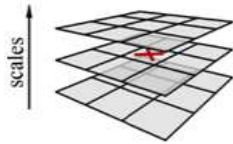
크기, 회전, 잡음, 명암에 불변하는 후보 점들을 찾기 위하여 (그림 2)와 같이 이미지 크기를 다르게 설정해주는 이미지 피라미드를 사용한다. 이미지의 크기가 변해도 강건한 점은 남아 있는 점을 이용한다. 특징점이 지만 오차로 인하여 검출되지 않는 점까지 검출하기 위하여 많은 후보

점을 선출하겠다는 것이다.



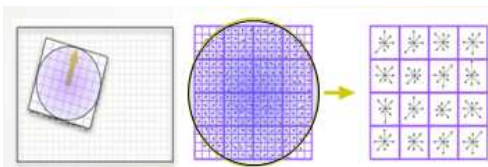
(그림2) 이미지 피라미드에 의한 다양한 크기의 DoG

후보 점에는 오차와 잡음으로 후보가 된 점이 포함되었다. 그래서 이런 오차와 잡음을 제거하기 위하여 (그림 3)과 같이 생성된 후보 점 DoG 가운데 이미지 와 다른 DoG 위 아래 이미지 2장을 가지고 특징을 비교한다. 이는 특징 점을 주변 8개와 다른 임계값의 DoG 이미지 각각 9개씩 총 26개를 비교하여 극값을 확인하여 극값을 아니면 후보 점에서 제거하여 많은 후보 점의 수를 줄인다.



(그림3) 3장의 DoG 이미지

이 후 다른 오차와 잡음 제거하고 나온 후보 점을 가지고 (그림 4)와 같이 고유 방향 벡터를 구하여 이 벡터를 가지고 특징을 가지게 된다. 이 특징을 가지고 두 이미지를 비교하여 가장 근접한 특징을 연결을 하여 이미지 매칭을 하게 된다.



(그림 4) 후보 점 주변 고유 방향 벡터

## 2.2 SURF 기반의 LOG

SURF는 SIFT의 속도를 개선한 방법으로 라플라시안(Laplacian)함수와 가우시안 함수를 같

이 이용하는 LOG을 사용한다. 라플라시안은 영상을 회전시키고 적용과 적용 후 회전시켜도 같은 결과를 가지는 등방성 2차 미분 함수이다.수식3은 가우시안 함수를 적용한 LoG 식이다.

$$LoG(x, y, \sigma) = \frac{1}{\pi\sigma^4} \left[ 1 - \frac{x^2 - y^2}{2\sigma^2} \right] \exp \frac{-(x^2 + y^2)}{2\sigma^2}$$

<수식 3>

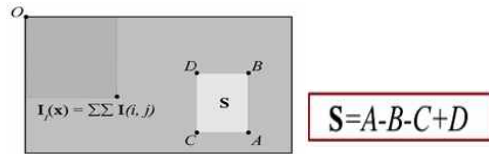
우리는 수식 3을 대신 유사한 결과를 가지는 수식 4를 사용한다.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$L(x, y) = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

<수식 4>

LoG에 의하여 특징점이 나오면 잡음을 제거하고 Hessian 함수를 기초로 하여 그림5와 같이 넓이를 이용하여 속도를 개선시키는 방법이다.

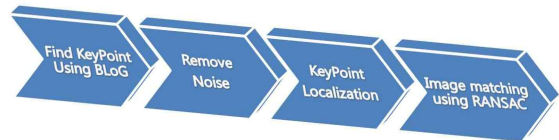


(그림 5> SURF 주요 방법

## 3. 제안하는 BLoG 기법

### 3.1 전체 구성도

보다 빠른 이미지 매칭을 위하여 제안한 알고리즘 방법을 적용한 전체적인 구성도 이다.



(그림6) BLoG을 이용한 이미지 매칭 전체 구성도

첫 번째 단계로 BLoG의 방법을 이용하여 특징이 되는 후보 점 들을 뽑아내고 두 번째로 잡

음으로 인하여 생긴 후보 점 들을 제거하고 세 번째로 오차로 형성된 부분을 제거 하고 지역화 시킨다. 마지막으로는 이미지 매칭을 위하여RANSAC (Random Sample and Consensus) [10] 방법을 이용하여 이미지 매칭을 시킨다.

### 3.2 BLoG(BitRate Laplacian Of Gaussian)

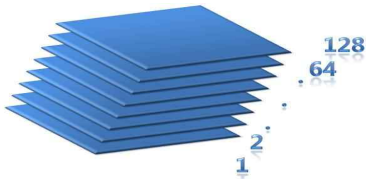
제안하는 BLoG(BitRate Laplacian Of Gaussian)은 라플라시안 함수와 가우시안 함수를 이용하여 보다 빠르게 특징이 되는 후보 점 을 검출한다. 먼저 이미지를 라플라시안 함수를 이용하여 등방성을 가진 이미지를 만들어 낸다. 라플라시안 함수는 2차 미분 함수이기 때문에 등방성 성질을 가지게 된다. 이는 변화가 있는 점이나 선 고립점에서 반응을 가지게 되는데 이 모두 후보 점이 된다. 라플라시안 함수로 얻은 이미지를 식가우시안 함수를 이용하여 새로운 이미지를 생성해야 한다. DoG와 LoG와는 달리 BLoG에서는 연산 하는 양을 줄이기 위하여 이미지 피라미드는 따로 설정하지 않는 대신 수식5와 같이 라플라시안 이미지를 임계값이 다르게 가우시안 함수를 적용하여 이미지를 만든다. 여기에서 n 은 비트분할 단위이며 분할 한 n값에 따라 LoG의 값  $\sigma$  가 달라진다.

$$BLoG(x, y, n) = ((L(x, y), \sigma) * I(x, y)) | (2^{5\sigma})$$

$$(\sigma = 0.4n)$$

<수식 5 >

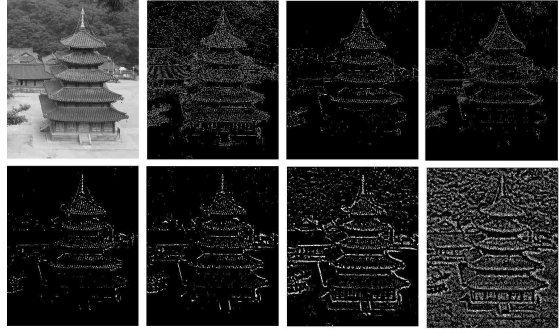
만들어진 이미지를 그림7과 같이 비트단위로 분할로 뽑아내어 특징이 되는 후보 점 들을 검출 할 것이다.



(그림 7) 비트단위 분할 이미지

(그림 8)은 8비트로 BLoG 영상을 적용한 그림이다. 첫 번째 비트인 0은 잡음이 많아 처음부터 구할 필요가 없어 두 번째 비트인 1부터 마지

막 비트인 7까지 각각의 BLoG 영상을 보여준다.



(그림8) 각각의 임계값으로 BLoG 한 이미지

각각의 BLoG 영상을 수식6에 적용하여 새로운 이미지를 생성한다. 여기서 마지막 비트인7은 잡음과 오차가 많이 포함되어 있기 때문에 첫 번째 비트인0 번과 같이 포함하지 않는다. 그래서 총 6개의 BLoG 이미지를 비교하여 일정 계 된다.

$$Robust\ Point = \sum_{n=1}^6 BLoG(x, y, n)$$

<수식 6>

이렇게 재설정 된 이미지에서 2비트 이하의 점들을 제거해 주면 강건한 특징을 가진 후보 점 들이 검출이 된다. (그림 9)은 그 후보 점들을 나타낸 것이다.



(그림 9) 강건한 특징점 결과

### 3.3 노이즈 제거 및 오차 낮추기

하지만 여기서 나온 강건한 특징을 가진 후보 점 들은 여전히 수가 많음을 알 수 있다. 위의 후보 점 들은 잡음과 오차를 아직 지니고 있는 후보점도 포함되어 있다. 그래서 수식7을 통하여 잡음 제거를 하였다.

$$\begin{aligned} \text{Horizontal} &: f(x, y) = f(x-1, y) \cap f(x+1, y) \\ \text{Vertical} &: f(x, y) = f(x, y-1) \cap f(x, y+1) \\ \text{Digonal}(+) &: f(x, y) = f(x+1, y-1) \cap f(x-1, y+1) \\ \text{Digonal}(-) &: f(x, y) = f(x-1, y-1) \cap f(x+1, y+1) \end{aligned}$$

<수식 7 >

이는 x축 y축 대각선 방향 모두 4방향의 미분을 통하여 주변의 잡음들을 줄일 수 있다. (그림 10)은 수식7을 적용하여 잡음을 제거하기 전과 제거한 후의 차이를 나타내는 그림이다.



(그림 10) Noise 제거 전과 후(4방향)

잡음을 줄였지만 아직도 이미지는 오차로 생긴 다수의 후보 점들이 남아 있다. 오차가 없는 완전한 특징을 가진 후보 점으로만을 추출하기 위하여 수식8을 이용하면 오차로 생긴 점들을 줄일 수 있어서 완전한 강건한 특징을 가진 점을 추출하게 된다.

$$\text{KeyPoint} = \sqrt{(L_{x+1,y} - L_{x-1,y})^2 + (L_{x,y+1} - L_{x,y-1})^2}$$

<수식 8>

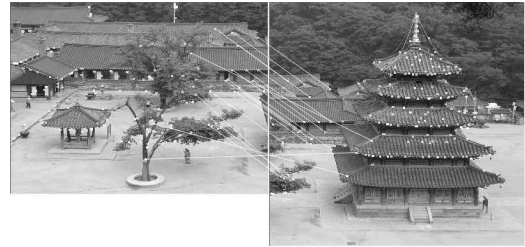
식8을 적용하기 위해서는 먼저 전에 나온 후보 점을 중심으로 하는 16×16 영역을 생성한다. 이 영역 안에서 수식8을 적용하여 최대 극값을 측정하면 이 점이 우리가 찾는 강건한 특징을 갖는 점이 되는 것이다. 이 때 영역 안에 2개 이상의 특징 후보 점이 들어오면 극 값으로 설정한다. 이는 주변의 다수의 강건한 값을 하나의 가장 강건한 점으로 만들 수 있을뿐더러 주변 오차로 형성된 점들을 다시 재설정 하고 오차를 줄여 준다. (그림 10)은 수식8을 적용전과 후 이미지이다. 이미지 매칭에 필요한 강건한 점들을 검출하였다. 이제 비교하여 이미지 매칭을 시켜주기만 하면 된다.



(그림11) 이미지 오차제거

### 3.4 RANSAC을 이용한 이미지 매칭

BLoG로부터 추출한 강건한 특징을 가진 점들을 이미지매칭 시키기 위한 방법으로 RANSAC (Random Sample and Consensus) [10] 방법을 이용하여 이미지 매칭을 실험 하였다.



(그림 12) RANSAC을 이용한 매칭포인트 결정

(그림 12)은 RANSAC방법을 이용하여 이미지 매칭 포인트를 결정짓는 그림이다. 이 매칭 포인트를 가지고 서로 다른 3장의 이미지를 매칭 시켜 (그림 13)의 결과를 얻었다.

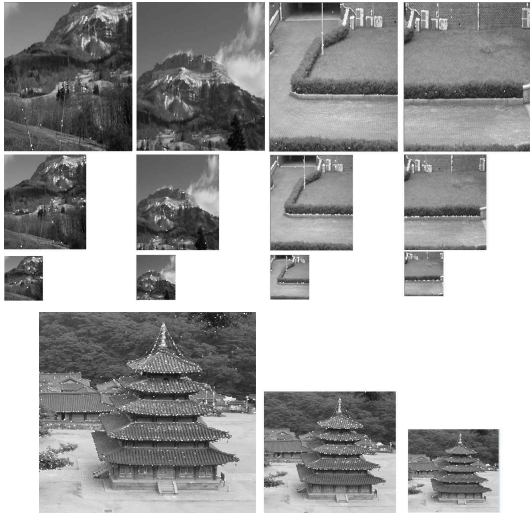


(그림13) 최종 이미지

## 4. 실험 결과 및 분석

### 4.1 크기 변화 실험 분석

다양한 크기 상태를 변화를 주어서 실험을 해 보았다. 변화에 따른 특징 점을 추출해본 결과 크기 변화에 따라 각각 불규칙적으로 특징 점들이 추출되었음을 그림14를 보고 알 수 있다.



(그림14) Scale 변화에 따른 특징점 추출

하지만 이 점들을 잡음 제거 후 크기의 변화에 따라 특징 점을 뽑아내어 비교 해 보았을 때 주 매칭 포인트가 되는 점들은 크기가 변해도 검출되는 것을 (표 1>을 통하여 알 수 있다.

<표 1> Scale 변화에 따른 결과

| 이미지                 | 50%이미지 / 잡음제거 | 원본이미지 / 잡음제거 | 170%이미지 / 잡음제거 | Robust Points | Key Points |
|---------------------|---------------|--------------|----------------|---------------|------------|
| 1. 팔상정 (768 * 576)  | 102 / 57      | 405 / 102    | 1230 / 271     | 39            | 33         |
| 2. 산 1 (320 * 240)  | 57 / 22       | 125 / 63     | 96 / 59        | 17            | 14         |
| 3. 산 2 (320 * 240)  | 51 / 21       | 108 / 52     | 51 / 40        | 19            | 17         |
| 4. 정원 1 (256 * 256) | 53 / 47       | 160 / 74     | 128 / 78       | 38            | 31         |
| 5. 정원 2 (256 * 256) | 56 / 36       | 155 / 41     | 154 / 42       | 31            | 25         |

<표 1> 을 보면 크기 변화에 추출된 강건한 점이 주 매칭 포인트가 되는 것이 83.3% 정도가 되는 것을 볼 수 있다. 즉 이는 특징점이 스케일

변화에 영향을 미치지만 이미지 매칭에 영향을 미치는 강건한 점들은 거의 유지됨을 알 수 있다.

### 4.2 회전 변화 실험 분석

제시한 BLoG가 회전에 영향을 얼마나 미치는지 확인하기 위하여 2차원 유클리드 공간상에서의 회전변환과 3차원 공간상에서의 회전변환을 실험 하였다.



(그림 15>회전 변화에 따른 특징점 추출

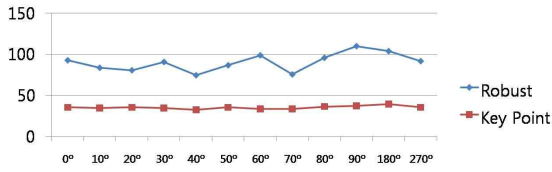
(그림 15)는 2차원 유클리드 공간상의 회전을 10도 간격으로 회전변환을 주어서 얻은 영상들이다.

<표 2> 10 간격의 회전변화에 따른 결과

|              | 0  | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90  | 180 | 270 |
|--------------|----|----|----|----|----|----|----|----|----|-----|-----|-----|
| Robust Point | 93 | 84 | 81 | 91 | 75 | 87 | 99 | 76 | 96 | 110 | 104 | 92  |
| Key Point    | 36 | 35 | 36 | 35 | 33 | 36 | 34 | 34 | 37 | 38  | 40  | 36  |

<표 2> 는 위 결과에 따른 강건한 점과 매칭에 영향을 미치는 점들을 비교한 표이다. 이 표에서 특징 점은 81.7%가 유지됨을 알 수 있었다.

그 중 매칭에 영향을 미치는 특징 점은 90% 이상이 유지됨을 알 수 있다. 즉 유클리드 공간에서의 회전 변화에 따른 이미지 매칭에 영향을 미치는 강건한 특징 점들은 거의 변화가 없음을 알 수 있다. 그림16는 표2를 그래프로 표현한 것이다



(그림 16) 회전 변화 그래프

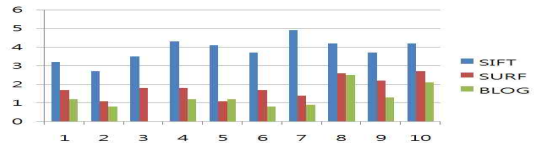
2차원에 이어서 3차원적인 회전 변화에 대한 실험이 (그림 17)에서 보여준다. 그 결과 3차원적 실험에서는 강건한 특징 점은 많이 다르게 나타났으나 이미지 매칭을 위한 특징 점은 약간 줄어들 뿐 이미지 매칭 시에 필요한 조건에는 충분히 유지됨을 발견 할 수 있었다.



(그림 17) 3차원 공간의 회전변화에 따른 결과

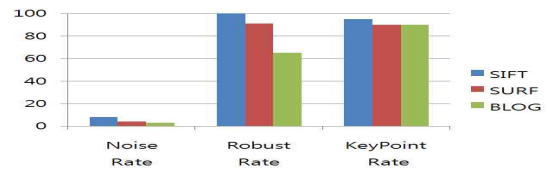
### 4.3 강건한 특징점 검출 속도와 메모리 효율

제안하는 BLoG방법을 다른 이미지 매칭 방법과 비교하기 위하여 연관 이미지 2장씩을 각각 다른 10번 이미지 매칭 실험을 하였다. 먼저 Fast 코너와 헤리스 코너를 검출하여 RANSAC를 적용하여 이미지 매칭을 하는 방법과 SIFT 와 SURF로 하는 이미지 매칭을 하는 방법을 비교해 보았다. 먼저 Fast코너와 헤리스 코너는 특징을 검출하는데 있어서는 빠른 결과를 나타냈지만 특징인자가 많아 이걸 비교 하여 이미지 매칭을 하는데 상당한 시간이 오래 걸렸다. SIFT 와 SURF를 비교해본 결과는 (그림 18)과 같았다. SIFT는 4ms의 평균속도로 가장 느린 결과를 얻었고 SURF는 1ms의 평균 속도를 얻은 거에 비하여 BLoG는 1ms이하의 평균속도로 가장 빠른 결과를 얻을 수 있었다.



(그림 18) 이미지 매칭 비교 결과

하지만 제안한 BLoG는 속도가 가장 빠르지만 이미지 매칭에 실패한 경우가 1번 있었다. 그래서 이미지 매칭을 하기 전 특징 점 검출을 실험해 보았다. 실험은 위에서 사용되던 10개의 이미지를 가지고 3가지를 해 보았다. 먼저 검출된 점에 대하여 잡음의 비율을 해보았다. 잡음은 별 차이가 나지 않았다. 두 번째로 SIFT를 기준으로 SURF와 BLoG에서 검출 된 특징 점의 수의 비율을 조사 해보니 제안한 BLoG의 방식은 특징점을 찾는데 있어 저조함을 알 수 있다. 하지만 마지막으로 이미지 매칭에 영향을 미치는 점들을 지니고 있는 비율을 계산해보면 모두 가지고 있는 실험 결과를 얻었다. (그림 19)는 3가지 실험에 대한 결과이다.



(그림 19) 특징 점 비율 실험 결과

## 5. 결론 및 향후계획

이미지 매칭을 하기 위해서는 크기, 회전, 명암에 강한 특징 점을 찾는 것이 중요하다. 속도 제약이 없는 완벽한 이미지 매칭을 하기 위해서는 SIFT에서 사용되는 DOG를 이용한 특징 점 검출법을 사용하여 매칭 하는 가장 적합하다. 하지만 실시간 이미지 매칭을 하기에는 부적절하다. 속도를 중시하는 실시간 이미지 매칭에서는 SURF의 LOG를 이용한 방법도 있겠지만 BLoG에서는 이미지 피라미드를 이용하지 않기 때문에 메모리 효율 면과 속도 면에서 뛰어나다. 하지만 모든 특징 점을 검출하지 못한다는 단점을 가지고 있지만 이미지 매칭에 필요한 강건한 특

정 점을 거의 유지 할 수 있다. 그러나 본 논문이 제안한 BLoG 알고리즘은 향후 속도 면과 메모리 효율을 유지하면서 보다 정확한 매칭 점에 대한 검출연구가 수행되어야 한다.

**참 고 문 헌**

[1] H. P. Moravec. Towards Automatic Visual Obstacle Avoidance. Proc. 5th International Joint Conference on Artificial Intelligence, pp. 584, 1977, <http://library.chosun.ac.kr/meta.asp>

[2] C. Harris and M.j. Stephens "A combined corner and edge detector", In Alvey Vision Conference, 1988. pp.147-151, <http://library.chosun.ac.kr/meta.asp>

[3] S.M. Smith and M. Brady. SUSAN - A New Approach to Low Level Image Processing. International Journal of Computer Vision, Vol. 23(1), pp. 45-78, 1997. <http://library.chosun.ac.kr/meta.asp>

[4] M. Trajkovic and M. Hedley. Fast Corner Detection. Image and Vision Computing, Vol. 16(2), pp. 75-87, 1998. <http://library.chosun.ac.kr/meta.asp>

[5] David G. Lowe. "Object Recognition from Local Scale - Invariant Features". Proc of the 7th Int. Conf. on Computer Vision, pp. 1150 -1157, Corfu, Greece, 1999. <http://library.chosun.ac.kr/meta.asp>

[6] David G. Lowe. "Distinctive image features from scale-invariant keypoints" Accepted for publication Int. Journal of Computer Vision, 2004, <http://library.chosun.ac.kr/meta.asp>

[7] Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: CVPR. 506 - . 513. 2004, <http://library.chosun.ac.kr/meta.asp>

[8] Herbert Bay Tinne Tuytelaars, and Luc Van Gool. "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding , Volume 110 Issue 3, June 2008. <http://library.chosun.ac.kr/meta.asp>

[9] Mikolajczyk, K., Schmid, C.: Indexing based on scale invariant interest points. In:ICCV. Volume 1. 525 - . 531. 2001. <http://library.chosun.ac.kr/meta.asp>

[10] A.J. Lacey, N. Pinitkarn, and N.A. Thacker, " An

evaluation of the performance of RANSAC algorithms for stereo camera calibration", The 11th British Machine Vision Conference, Bristol, UK, 2000.

<http://library.chosun.ac.kr/meta.asp>

**박 이 근**



2009년 : 조선대학교 컴퓨터통계학과 (학사)  
2009년 : 조선대학교 전산통계학과 (석사) 입학

2009년~현 재: 조선대학교 컴퓨터통계학과 석사과정  
관심분야 : 컴퓨터비전, 영상압축, 멀티미디어

**김 종 민**



2004년 : 전남대학교 컴퓨터공학과 (공학석사)  
2008년 : 조선대학교 전산통계학과 (이학박사)

2007년~현 재: 조선대학교, 광주대학교 외래교수  
관심분야 : 컴퓨터비전, 생체보안, 멀티미디어

**이 용 기**



1975년 : 조선대학교 전자공학과 (학사)  
1981년 : 명지대학교 대학원 전자공학과 (공학석사)  
1984년 : 순천대학교 전자계산학과 (공학박사)

1985년~1986년: 미국 Oregon Graduate Center 객원 교수

1988년~현 재: 조선대학교 컴퓨터통계학과 교수  
관심분야 : 컴퓨터그래픽스, 멀티미디어, 컴퓨터구조