

# 히스토그램과 블록분할을 이용한 매칭 알고리즘

## Matching Algorithm using Histogram and Block Segmentation

박성곤\*, 최연호\*\*\*, 조내수\*\*, 임성운\*\*\*, 권우현\*\*\*\*  
(Sung-Gon Park, Youn-Ho Choi, Nae-Su Cho, Sung-Woon Im, Woo-Hyun Kwon)

**Abstract** - The object recognition is one of the major computer vision fields. The object recognition using features(SIFT) is finding common features in input images and query images. But the object recognition using feature methods has suffered of difficulties due to heavy calculations when resizing input images and query images. In this paper, we focused on speed up finding features in the images. we proposed method using block segmentation and histogram. Block segmentation used diving input image and than histogram decided correlation between each block and query image. This paper has confirmed that the matching time reduced for object recognition since reducing block.

**Key Words** : SIFT, Block Segmentation, Histogram, Feature Detect, Object Recognition

### 1. 서론

매칭에 사용되는 특징점은 객체인식, 파노라마와 같이 입력된 영상과 기준 영상을 매칭하기 위해 사용하는 공통된 중요한 위치 정보를 말한다. 이러한 특징점을 검출하고 매칭하기 위하여 많은 방법들이 컴퓨터 비전 분야에서 여러 가지 방법으로 연구 되고 있다[1]. SIFT 알고리즘은 영상의 모서리 같은 특징이 되는 부분을 벡터방식으로 특징점을 추출하는 방법으로 영상의 크기나 회전에 강력한 매칭성능을 가지고 있는 매칭 방법이다[2][3][4]. 영상의 크기를 조절하여 특징점을 추출하는 매칭 특성상 많은 연산량으로 인해 매칭 시간이 오래 걸린다는 단점을 가지고 있다.

본 논문에서는 입력된 영상 전체의 특징점을 추출하는 것이 아니라 매칭 시간을 단축하고 매칭 확률을 증가시키기 위하여 입력 이미지를 블록분할 하여 물체인식에 필요한 부분에 대해서만 특징점을 추출하는 방법을 사용하였다. 최근의 Video Camera나 CCTV들은 컬러영상을 지원한다는 것을 이용하여 각 블록과 기준영상의 상관관계를 컬러 이미지의 히스토그램을 비교하여 판단하는 방법을 사용하였다.

제안한 방법은 특징점을 추출하기 위해 입력 이미지를 일정 크기의 블록 분할하고, 각 블록의 히스토그램을 기준영상의 히스토그램과 비교하여 상관관계를 얻은 후 두 이미지의 상관 관계가 일정 값보다 높을 경우 해당 블록의 특징점을

구하여 database에 저장하게 된다. 모든 블록의 상관관계 비교와 특징점이 구해지고 나면 최종적으로 기준 영상의 특징점과 비교하여 최종 매칭이 끝나게 된다. 제안된 방법으로 매칭을 수행할 경우 전체 영역에서 특징점을 구하여 매칭하는 기존의 방법보다 매칭 수행시간이 줄어드는 것을 확인 한다.

### 2. 특징점 매칭 알고리즘

#### 2.1 SIFT (Scale Invariant Feature Transform)

SIFT는 1999년 David G. Lowe가 제안한 알고리즘으로 이미지의 회전이나 크기에 강인한 특징점 추출 알고리즘이다 [2][3]. 본 논문에서는 기본적인 SIFT 알고리즘과 제안된 알고리즘의 비교를 위하여 Oregon State University의 Rob Hess가 공개한 알고리즘 소스를 참고 하였다.

#### 2.1.1 Scale-space extrema detection

후보 특징점은 DOG(Difference of Gaussian) 이미지[3]를 생성하여 구할 수 있다. DOG 이미지를 생성하기 위하여 여러단계의 가우시안 이미지를 생성해야 한다. 각 단계의 가우시안 이미지가 생성되면 각 이미지의 차 이미지를 구한다.

가우시안 함수는

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

와 같이 표현되고, 가우시안 이미지를  $L(x,y,\sigma)$ 라 하면,

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

와 같이 표현 된다.

식(2)를 이용하여 구하여지는 DOG 이미지를  $D(x,y,\sigma)$ 라 하면

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

저자 소개

- \* 準會員 : 慶北大學 電子工學科 碩士課程
- \*\* 正會員 : 慶北大學 電子工學科 博士課程
- \*\*\* 正會員 : 慶北大學 制御電氣工學部 教授
- \*\*\*\* 正會員 : 慶北大學 電子工學科 教授

와 같이 구할 수 있다.

DOG가 구하여 지면 DOG가 가진 극값들을 추출해야 한다. 연속된 3개의 영상을 이용하여 3×3×3마스크를 이용하여 최대, 최소의 극값을 추출한다.

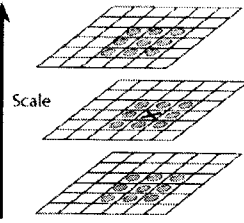


그림 1. DOG 이미지에서 최대, 최소 값 추출  
Fig 1. Maxima and minima of the DOG images.

### 2.1.2 Keypoint localizaton

3×3×3마스크를 이용하여 구하여진 특징점은 작은 노이즈에서 반응하게 되는 저 저도(bad contrast) 특징점과 에지를 따라 존재하게 되는 중복적인 특징점을 제거한다..

### 2.1.3 Orientation assignment

제거되고 남은 각 후보 특징점들을 중심으로 주위의 16×16 픽셀의 데이터를 각 특징점들의 방향성 표시를 하게 된다.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (4)$$

$$\theta(x,y) = \tan^{-1} \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (5)$$

방향성은 식(4),(5) 으로 표현되고 크기는  $m(x,y)$ 로, 방향은  $\theta(x,y)$ 로 구하여 진다. 이때 구해진 특징점의 방향, 크기는 이미지가 회전하여도 같은 값을 가지고 있으므로 매칭시 회전된 이미지에서도 매칭을 가능하게 해준다.

### 2.1.4 Keypoint descriptor

식(4),(5)에 의해 구해진 16×16 픽셀 각각의 방향, 크기 데이터를 이용하여 4×4 크기 데이터와 8개의 방향 데이터로 기술하여 총 128개의 벡터데이터로 표현되어진다.

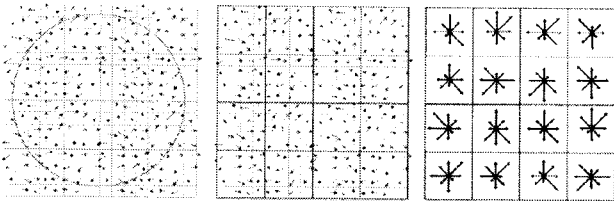


그림 2. 특징점 묘사  
Fig 2. Keypoint descriptor.

## 2.2. 매칭 알고리즘

입력 이미지와 비교 이미지의 특징점 묘사가 끝나면 각각의 특징점을 가지고 매칭을 하게 되는데 이때 사용되는 방법이 유클리안 거리(Euclidean Distance)인데 다음과 같은 식으로 표현된다.

$$D(a,b) = \sqrt{\sum_{i=1}^n (D_a[i] - D_b[i])^2} \quad (6)$$

식 (6)에서와 같이 표현되며 두 특징점  $D_a, D_b$  의 묘사자

가 비슷할수록  $D(a,b)$ 는 0에 가까워 진다.

## 2.3 히스토그램의 비교

히스토그램은 컬러 이미지가 가지고 있는 색 데이터로써, 본 논문에서는 이미지를 HSV로 변경 하고 Hue, Saturation, Value 중 Hue 이미지, 즉 색조 이미지로 최종변경 후 해당 블록 히스토그램과 비교 이미지의 히스토그램을 비교하게 된다.

$$result = \frac{\sum_1^n \hat{q}_i \hat{v}_i}{\sqrt{\sum_1^n \hat{q}_i^2 \sum_1^n \hat{v}_i^2}} \quad (7)$$

식(7)은 두 Hue 이미지의 히스토그램 비교 계산식이다. n 은 히스토그램 명암의 개수,  $q_1, v_1$ 은 해당 명암의 픽셀 합을 나타낸다. 계산된 값이 1에 가까울수록 두 이미지는 유사하다고 판단한다.

## 2.4 제안된 알고리즘

SIFT 알고리즘은 회전, 크기에 강인하나, 알고리즘의 특성 상 입력 이미지의 축소, 확대 등으로 인한 많은 연산량을 단점으로 가지고 있다.

본 논문에서는 SIFT 알고리즘의 단점을 개선 하고자 필요한 영역에서만 DOG를 적용하여 특징점을 구하여 데이터베이스에 저장 하고 만들어진 데이터베이스에서 기준 이미지의 데이터베이스와 매칭을 하여 최종적으로 매칭이 완료 하게 된다. 필요한 영역을 구하기 위하여 입력 이미지를 일정 크기의 블록으로 분할하였으며, 일반적인 이미지 사이즈인 4:3 비율 중 800×600 픽셀을 기준으로 나머지 부분이 생기지 않는 40×40 픽셀로 분할을 하였다.

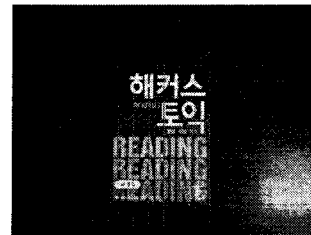


그림 4. 블록 분할된 이미지  
Fig 4. Image block segmentation.

그림 4은 800×600 픽셀의 이미지로써 40×40 픽셀로 나누었을 때 20×15총 300개의 블록이 생성 되게 된다.

블록 분할된 이미지는 히스토그램을 이용하여 왼쪽 위의 블록을 시작으로 오른쪽 밑의 마지막 블록까지 기준 이미지의 히스토그램과 순차적으로 비교하여 식(7)에 의해 각 블록에 대한 유사도 상관관계 값을 계산하게 된다. 이때 계산된 상관관계 값을 이용하여 threshold 값을 적용, 일정 값 이상의 경우에만 유사하다고 판단한다.

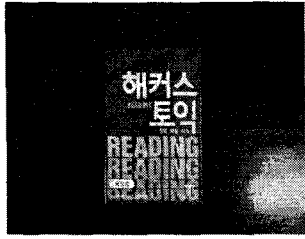


그림 5. 임계값 적용 된 블록 이미지

Fig 5. Block image adapted threshold value.

그림 5는 threshold 값 적용 후 상관관계가 있는 블록만 나타낸 그림이다.

threshold가 넘는 블록의 경우 DOG를 적용하여 특징점을 추출하게 된다. 추출된 특징점은 데이터베이스에 저장하고 데이터베이스에 저장된 특징점과 기준 이미지의 특징점을 비교하여 입력이미지의 전체가 아닌 상관성 있는 블록만 매칭하는 것으로 최종적인 매칭이 완료하게 된다.

### 2.5 모의 실험

제안된 알고리즘은 입력 이미지에서 물체 이미지를 비교하여 찾는 것으로 실험하였다.

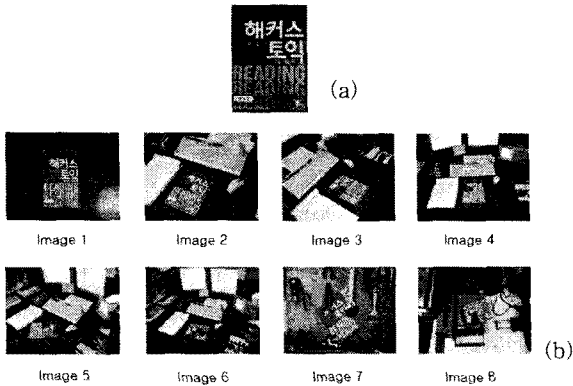


그림 6. (a) 대상이미지 (b) 입력이미지

Fig 6. (a) Object image. (b) Input image.

대상 이미지와 입력 이미지의 크기는 각각 200×278, 800×600 픽셀의 크기로 Visual Studio 6.0에서 OpenCV library를 사용하여 실험 하였다.

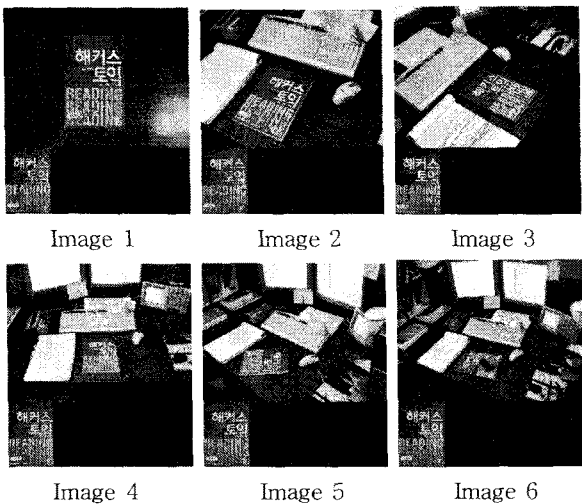


Image 7

Image 8

그림 7. 제안된 알고리즘 결과

Fig 7. Result of proposed algorithm.

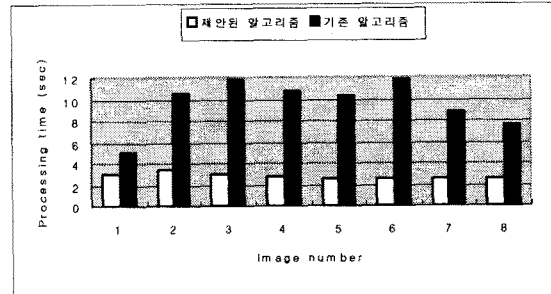


그림 8. 매칭 시간

Fig 8. Performance time for matching.

그림 7, 8은 제안된 알고리즘의 매칭 결과를 보여준다. 그림 8에서와 같이 기존 알고리즘 대비하여 평균 69.4%의 매칭 시간 개선, 기존 알고리즘 평균 90%, 제안된 알고리즘 평균 95.4%로써 평균 5.4%의 개선된 매칭 성공률을 알 수 있다.

### 3. 결론

물체 인식분야 중 하나인 특징점 매칭 알고리즘은 입력 이미지와 대상 이미지에서 특징점을 추출하고 두 이미지 사이의 특징점을 매칭 시켜 입력 이미지에서 빠르고 정확하게 대상 이미지를 찾는데 목적이 있다. 본 논문에서는 기존 알고리즘의 문제점인 매칭 시간을 보완하는데 목적을 두었으며, 제안된 알고리즘은 모의실험을 통하여 기존 알고리즘에 비해 매칭 시간이 줄어드는 것을 확인할 수 있었다.

### 참 고 문 헌

- [1] F. Arman and J.K. Aggarwal, "Model-Based Object Reconition in dense-range image - a review" *ACM Computing Surveys*, Vol.25, No.1, 1993
- [2] David G.Lowe. "Object Recognition from Local Scale-Invariant Features." *Proc. Of the International Conference on Computer Vision (ICCV)*, Corfu, September 1999, pp.1150-1157
- [3] David G.Lowe. "Distinctive Image Features from Scale Invariant Keypoints." *Accepted for publication in the International Journal of Computer Vision*, pp.91-110, 2004
- [4] C.J. Harris and M.Stephes, "A combined conner and edge detector", *In Fourth Alvey Vision Conference*, Manchester, UK, pp.147-151, 1998
- [5] Rafel C. Gonzalez. Et al, "Digital Image Processing using MATLAB", Prentice-Hall, 2004
- [6] Rafael C. Gonzalez, "Digital Image Processing", Prentice-Hall, 2008