

# 메모리 사용률을 개선한 SURF 알고리즘 특징점 추출기의 하드웨어 가속기 설계

정창민\* ·곽재창\* ·이광엽\*\*

\*서경대학교

An Implementation of a Feature Extraction Hardware Accelerator  
based on Memory Usage Improvement SURF Algorithm

Chang-min Jung\* · Jae-chang Kwak\* · Kwang-yeob Lee\*\*

\*SeoKyeong University

E-mail : kenji01@skuniv.ac.kr

## 요 약

SURF 알고리즘은 영상의 특징점 검출 및 서술자를 생성하는 알고리즘으로 크기와 회전, 조명 및 시점 등의 환경 변화에 강인한 특징을 가지고 있다. 이러한 특징 때문에 객체 인식, 파노라마 이미지, 3차원 영상 복원 등 영상처리 분야에서 많이 사용되고 있다. 하지만 SURF 알고리즘과 같은 대부분의 인식 알고리즘은 많은 양의 연산을 필요로 하기 때문에 실시간 구현이 어렵다. 본 논문은 SURF의 메모리 접근 횟수와 메모리 사용량을 분석하여 효율적인 메모리를 설계함으로써 메모리 접근 횟수와 메모리 사용량을 최소화하여 실시간 구현이 가능하도록 설계하였다.

## ABSTRACT

SURF algorithm is an algorithm to extract feature points and to generate descriptors from input images. It is robust to change of environment such as scale, rotation, illumination and view points. Because of these features, it is used for many image processing applications such as object recognition, constructing panorama pictures and 3D image restoration. But there is disadvantage for real time operation because many recognition algorithms such as SURF algorithm requires a lot of calculations. In this paper, we propose a design of feature extractor and descriptor generator based on SURF for high memory efficiency. The proposed design reduced a memory access and memory usage to operate in real time.

## 키워드

SURF, Feature extraction, Integral image, Fast hessian matrix, Non-maximum suppression

## 1. 서 론

고성능 하드웨어를 탑재한 포터블 기기들이 개발되면서 이를 활용한 어플리케이션의 연구 개발이 활발히 이루어지고 있다. 특히 고성능 카메라의 탑재로 인해서 객체 인식 및 영상 처리에 대한 관심이 높아지고 있다.

영상 처리 분야에서 인식 및 처리를 하기 위해 입력된 영상의 특징점을 추출하고, 추출된 특징점을 통하여 서술자를 생성하여 객체 인식 및 추적

등을 하게 되는데 대표적인 방법으로는 SIFT, SURF 등이 있다.

두 알고리즘은 입력된 영상의 밝기 값을 이용하여 크기, 회전, 시점 등의 환경 변화에 강인한 특징점 및 서술자를 생성하는 특징이 있지만 SURF 알고리즘이 더욱 빠른 연산 속도를 가지므로 임베디드 환경에서의 개발에 용이하다는 장점이 있다. [1]

하지만 인식 알고리즘은 입력 영상에 대해 픽셀 단위로 연산하여 객체를 검출하기 때문에 반

복적이고 많은 연산을 처리해야 되며, 영상의 특징적인 위치를 서로 다른 환경에서도 동일한 부분을 반복적으로 검출해야 하므로 소프트웨어로 구현하기에 적합하지 않고 실시간 구동이 어렵다. 이러한 단점을 보완하기 위해 최근 Multi-core CPU의 탑재 및 그래픽 연산을 처리하기 위한 고성능 GPU의 탑재가 이루어지고 있지만 실시간 구동을 위해서는 하드웨어 구현이 필수적이다.

본 논문에서는 SURF 알고리즘 기반 특징점 추출기를 설계하여 실시간 구동에 적합하도록 처리 속도를 개선하였다.

## II. 본 론

SURF 알고리즘은 적분 이미지 생성, 특징점 추출, 서술자 생성을 통해 객체를 검출하는 알고리즘이다. Hessian Matrix를 이용하여 크기 변화에 강인한 특징점을 추출하고, 이에 대한 주방향을 검출하여 회전 변화에 강인한 서술자를 생성한다.

본 논문에서 설계한 특징점 추출기는 SURF 알고리즘 내에서 반복적인 연산이 많고 메모리 사용률이 높은 부분으로, 적분 이미지와 Box Filter의 메모리 접근 패턴을 분석하여 메모리 사용률을 줄여 특징점을 추출한다.

### 1. 적분 이미지

적분 이미지는 입력 영상에 대해 지정된 사각형을 구성하는 픽셀의 밝기 값의 합을 의미한다. 이는 Box Filter의 크기에 상관없이 지정된 범위의 픽셀 밝기 값의 합을 구할 수 있다.[2]

그림 1은 생성된 입력 영상 전체 영역에 대한 적분 이미지에서 지정된 사각형의 밝기 값을 구하는 방법을 나타낸다.

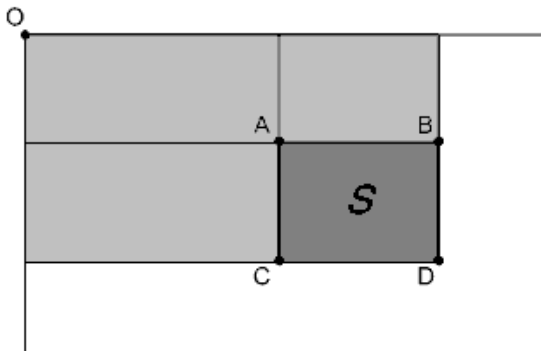


그림 1. 적분 이미지 내 사각형 S의 밝기 값 계산

그림 1에서 점 A, B, C, D로 이루어진 사각형 S의 밝기 값은 식 (1)로 구할 수 있다.

$$S = II_{\Sigma}(D) + II_{\Sigma}(A) - II_{\Sigma}(B) - II_{\Sigma}(C) \quad (1)$$

### 2. Hessian 검출기

Hessian 검출기는 식 (2)와 같은 Hessian Matrix를 기반으로 특징점 후보를 추출하는 알고리즘이다.  $L_{xx}$ 는 x 위치의 입력 이미지의 픽셀의 밝기 값과  $\sigma$ 의 분산을 가지는 가우시안 x방향 2차 미분 값과의 Convolution 값을 의미한다. 마찬가지로  $L_{yy}$ 는 y방향,  $L_{xy}$ 는 xy 방향으로 미분된 가우시안 필터와 Convolution 값을 의미한다.[3]

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (2)$$

SURF 알고리즘에선 식 (2)와 같은 Hessian Matrix의 근사화 된 Box Filter를 사용하여 빠른 속도로 크기 변화에 강인한 Hessian Matrix의 근사값을 계산 할 수 있다.

그림 2는 SURF 알고리즘에서 사용하는 x, y, xy 방향의 근사화 된 Box Filter이다.

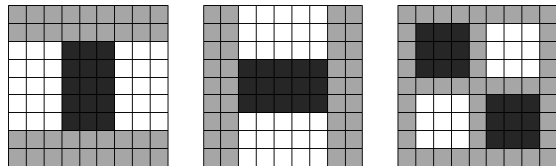


그림 2. x, y, xy 방향의 근사화 된 Box Filter

### 3. Non-Maximum Suppression

SURF 알고리즘은 크기 변화에 강인한 특징점을 얻기 위해 각 Scale 별로 Box Filter의 크기를 변화시켜 Determinant 값을 구한다.

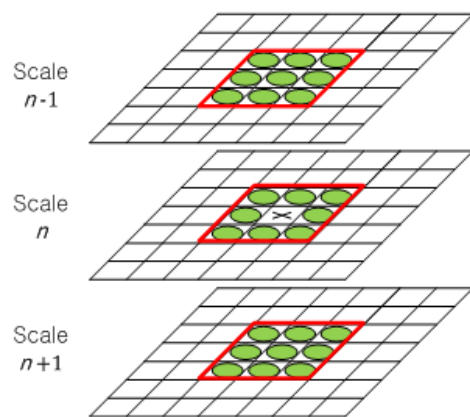


그림 3. Non-Maximum Suppression

Hessian 검출기에서 검출된 각 Scale 별 Determinant 값은 Threshold 값과 비교를 하게 되

고 Determinant 값이 더 클 경우, 같은 Scale 내의 주변 픽셀과 크기 비교를 하게 된다. 이 때, Determinant 값이 더 크면 Scale n-1의 9개의 데이터와 Scale n+1의 9개의 데이터와 비교를 하게 되는데 Determinant 값이 가장 클 경우 특징점으로 추출된다.[4]

그림 3은 특징점을 추출하는 과정이다.

### III. 실험

본 논문에서는 SURF 알고리즘의 특징점 추출기를 하드웨어로 구현했으며 설계 검증은 100MHz 동작 주파수의 Zynq Artix-7 FPGA를 이용하였다. Xilinx ISE 14.4 환경에서 Verilog HDL을 사용하였고 I-Sim으로 시뮬레이션을 진행하였다.

SURF 알고리즘의 특징점 추출기는 반복적인 연산과 많은 메모리 접근을 필요로 한다. 즉, 재사용되는 데이터가 많으며 그에 따른 메모리 접근 횟수가 많아져서 속도의 저하를 가져온다. 이 때, 메모리를 라인 메모리로 사용하면 메모리 접근 횟수를 병렬로 처리하여 속도의 저하를 개선시킬 수 있다.

그림 4는 SURF 알고리즘에서 생성된 적분 이미지로 접근하는 Box Filter의 메모리 접근 영역이다.[5]

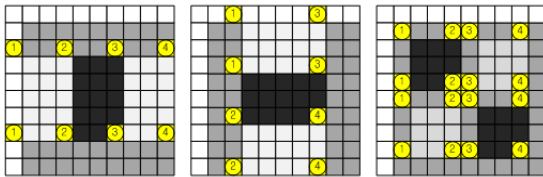


그림 4. Box Filter 계산을 위한 메모리 접근 영역

표 1은 VGA(640\*480) 사이즈의 이미지가 필요로 하는 각 Scale 별 메모리 크기이다. x, y, xy 방향으로 적용 되는 Box Filter의 접근 패턴을 분석하여 라인 별로 메모리를 구성하여 동시에 Read 하여 연산하는 것이 메모리 사용량과 속도에서 더 효율적임을 알 수 있다.[6]

표 1. VGA(640\*480) 사이즈 적분 이미지의 메모리 사용량

Scale	0	1	2	3	4
Filter Size	9×9	15×15	21×21	27×27	33×33
기존 메모리	8,601,600				
라인 메모리	179,200	286,720	394,240	501,760	609,280

(단위 : bit)

본 논문에서는 그림 5와 같이 각 라인을 FIFO로 구성하고 Box Filter의 패턴을 분석하여 특징점 추출에 필요한 데이터만을 받아 연산에 사용함으로써 라인 메모리로 설계한 특징점 추출기의 메모리 사용량보다 제안한 설계 기법을 적용한 특징점 추출기의 메모리 사용량이 감소하였음을 검증하였다.

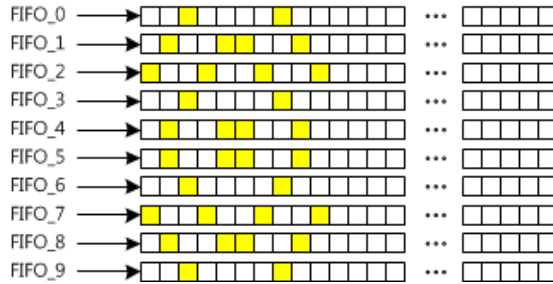


그림 5. 행 단위 FIFO로 구성한 메모리 구조

그림 6은 라인 메모리로 설계한 메모리 구조를 사용하였을 때의 특징점 추출기의 gate count와 제안하는 라인 메모리 설계 기법을 적용한 특징점 추출기의 gate count를 나타내는 그래프이다. 표 2는 특징점 추출기의 연산 처리 시간이다.

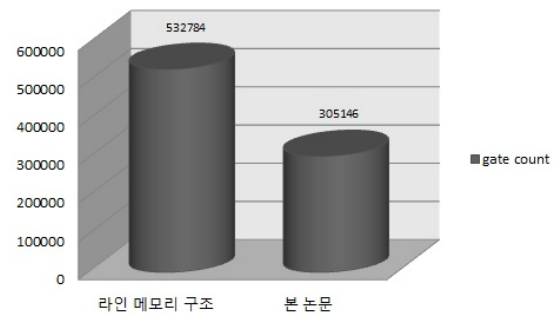


그림 6. 특징점 추출기의 gate count

표 2. 연산 처리 시간

구분	Virtex5 LX330[1]	본 논문 Artix7 100MH
적분 이미지	18	10
Hessian 검출기		

(단위 : msec)

### IV. 결론

본 논문은 임베디드 환경에서 SURF 알고리즘을 실시간으로 구동하기 위해 특징점 추출기를

하드웨어로 구현하여 FPGA로 검증하였다.

실시간 구동을 위해 특징점 추출기의 반복적인 연산을 각 스케일 별로 병렬 처리하여 연산 속도를 향상시켰고 메모리 접근 횟수 및 메모리 사용량을 줄이기 위해 각 라인을 FIFO로 구성하였다.

또한 각 스케일 별로 필요한 적분 영상의 패턴을 분석하여 최소 필요 부분만을 접근하여 연산에 사용함으로써 연산 속도를 개선하였다.

실험 및 검증 결과, 메모리 사용량이 기존의 메모리 사용량에 비해서 약 40% 감소하였음을 확인하였다.

본 논문에서 제안한 SURF 기반 특징점 추출기 하드웨어는 임베디드 환경에서의 객체 인식 및 영상 처리 어플리케이션의 처리 속도를 향상시켜 인식 알고리즘의 실시간 구동에 많이 활용 되어 질 것으로 기대된다.

### Acknowledgement

본 논문은 미래창조과학부 지원으로 수행한 ETRI SW-SoC융합 R&BD센터의 연구결과입니다.

### 참고문헌

- [1] “SURF 알고리즘 기반 특징점 추출기의 FPGA 구현” 류재경, Journal of Korea Multimedia Society, pages 483-492, 03. 2011.
- [2] “A Comparison of SIFT, PCA-SIFT and SURF”, Luo Juan, and Oubong Gwon, International Journal of Image Processing(IJIP), Vol. 3, No. 4. pages. 143-152, 2009.
- [3] “SURF: Speeded Up Robust Features”, H. Bay, T. Tuytelaars, and L.V. Gool, 9th European Conf. Computer Vision, pages 404-417, 2006.
- [4] “FPGA based Speeded Up Robust Features”, J. Svab, T. Krajnik, J. Faigl and L. Preucil, IEEE International Conference on Technologies for Practical Robot Applications, TePRA 2009, Woburn, Massachusetts, USA, pages 35-41, 2009.
- [5] “Object Recognition on a Chip : A Complete SURF-Based System on a Single FPGA”, Michael Schaeferling and Gundolf Kiefer, , 2011 International Conference on Reconfigurable Computing and FPGAs, Volume 2012, Article No. 6, 01. 2012.
- [6] “메모리 효율을 높인 SURF 알고리즘 기반 특징점 추출 하드웨어 가속기 설계”, 2013 SoC 학술대회, 05. 2013.