

객체인식을 위한 FAST와 BRIEF 알고리즘 기반 FPGA 설계

FPGA based Implementation of FAST and BRIEF algorithm for Object Recognition

허 훈 *, 이 광 엽**

Hoon Heo*, Kwang-Yeob Lee**

Abstract

This paper implemented the conventional FAST and BRIEF algorithm as hardware on Zynq-7000 SoC Platform. Previous feature-based hardware accelerator is mostly implemented using the SIFT or SURF algorithm, but it requires excessive internal memory and hardware cost. The proposed FAST & BRIEF accelerator reduces approximately 57% of internal memory usage and 70% of hardware cost compared to the conventional SIFT or SURF accelerator, and it processes 0.17 pixel per Clock.

요 약

본 논문은 기존의 FAST와 BRIEF 알고리즘을 Zynq-7000 Soc Platform에서 하드웨어로 구현했다. 대표적으로 SIFT 나 SURF 알고리즘을 사용하여 특징점 기반 하드웨어 가속기로 구현 하지만, 하드웨어 비용과 내부 메모리가 많이 필요하다. 제안하는 FAST & BRIEF 가속기는 기존의 SIFT 나 SURF 가속기 보다 내부 메모리 사용량을 약 57%, 하드웨어 비용을 약 70% 정도 감소하고, 수행 시간은 Clock 당 0.17 Pixel를 처리한다.

Key words : FAST, BRIEF, Hamming Distance, SURF, SIFT

1. 서론

컴퓨터 비전에서 물체인식 기법은 카메라 또는 각종

센서 정보를 활용하여 일정 범위에서 물체의 존재 여부를 판단하는 기술로 최근 많은 응용분야에서 연구되고 있다. 응용분야로 Markerless AR과 로봇을 예로 들 수가 있다.

물체 인식으로는 특징점 기반 알고리즘을 많이 사용한다. 대표적인 알고리즘은 SIFT[1]와 SURF[2]가 있다. 하지만 SIFT나 SURF는 특징점 검출 과정과 서술자 생성 과정에서 많은 연산 량이 요구된다. 이 점을 보완하기 위하여 SIFT 또는 SURF 하드웨어 가속기가 제안되었다. 하지만 SIFT와 SURF 알고리즘은 DoG(Difference of Gaussian)와 Gradient Vector를 보완하기 때문에 내부 메모리가 많이 필요하다는 단점이 있다. 이 단점을 보완하기 위해서 본 논문은 SIFT나 SURF 같은 특징 기반 인식 알고리즘인 기존

* Dept. of Computer Engineering, Seokyeong University

**Dept. of Computer Engineering, Seokyeong University, kylee@skuniv.ac.kr, 010-8225-2870

★ Corresponding author

※ Acknowledgment

"This work was supported by Ministry of Trade, Industry and Energy[MOTIE] and IDEC Platform center[IPC]."

Manuscript received Jun. 21, 2013; revised Jun. 27, 2013; accepted Jun. 27, 2013

의 FAST[3]와 BRIEF[4] 알고리즘을 하드웨어로 구현했다. FAST Corner Detection 알고리즘은 특징점 검출 알고리즘 중에서 처리 속도가 빠른 알고리즘이고, BRIEF 알고리즘은 Descriptor 생성 알고리즘 중에서 방향성에는 약하지만, 생성 속도가 빠른 알고리즘이다.

FAST 알고리즘은 가감산 및 비교 연산만을 필요하여 하드웨어로 구현했을 때 적은 하드웨어 비용으로 구현할 수 있다. BRIEF 알고리즘도 간단한 연산으로 이루어져 있기 때문에 하드웨어 비용이 적게 든다. 실험 환경으로 Zynq-7000 SoC Platform에서 FAST와 BRIEF 하드웨어를 구현하였고, 기존의 SIFT 또는 SURF 가속기와 비교하여 하드웨어 비용과 내부 메모리 사용량을 검증하였다.

기존의 SIFT나 SURF 가속기와 비교한 결과 내부 메모리는 약 57%, 하드웨어 비용은 약 70% 정도 감소하였다.

본 논문은 2장에서 알고리즘을 설명하고, 3장에서는 구현된 하드웨어 디자인을 소개한다. 4장에서는 시물레이션을 통한 구현 및 성능 분석을 설명한다.

II. 기존의 FAST & BRIEF 알고리즘

1. FAST Corner Detection

특징점 기반 인식 알고리즘은 특징점 검출, 서술자 생성, 매칭 단계로 수행된다. 특징점 검출 단계로 FAST(Features from Accelerated Segment Test) Corner Detection 알고리즘[3]을 사용하여 특징점을 검출한다.

특징점은 Corner와 Blob을 나타낸다. Corner는 Edge에서 모서리 부분을 뜻하고, Blob 영역은 주변보다 더 밝거나 어두운 부분을 뜻한다. FAST 알고리즘

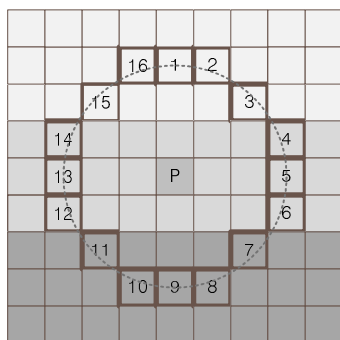


Fig. 1. FAST Algorithm
그림 1. FAST 알고리즘

은 FD(Feature Detection), FS(Feature Score), NMS(Non-Maximal Suppression)단계로 구성된다.

FD 단계는 P를 중심으로 그림 1과 같이 영역을 잡는다. Ip를 P에 있는 밝기 값이라고 가정했을 때, 1, 5, 9, 13에 있는 밝기 값(I1, I5, I9, I13)을 비교한다. threshold를 t로 가정한다면, Ip-t보다 어둡고, Ip+t보다 밝은 점이 3개 이상이면 P는 후보 특징점이 된다. P가 후보 특징점이 되면, 1 ~ 16(I1~I16)의 밝기 값과 위와 같은 방법으로 검사하게 된다.

비교 결과 더 어둡거나 더 밝은 점이 12개 이상이면 특징 후보로 정의한다. FS 단계에서는 후보 특징마다 Score를 할당한다. FD에서는 threshold를 임의로 결정하지만, FS 단계에서는 bmin과 bmax를 활용하여 threshold를 정의한다. bmin과 bmax가 같아졌을 때, 그 값이 Score가 된다.

Non Maximum Suppression(NMS) 단계는 이웃한 특징점 중에서 최댓값을 가지는 부분을 최종적인 특징점으로 선택하게 된다. 즉, 각각 후보 특징점 들의 Score를 비교하여 가장 높은 Score 가진 후보 특징점이 최종 특징점으로 선택된다.

2. BRIEF

Descriptor 생성 단계는 BRIEF 알고리즘[4]을 사용한다. BRIEF 알고리즘은 Bit String Description을 생성하며, 0과 1로 구성된다.

$$\tau(p; x, y) := \begin{cases} 1 : p(x) < p(y) \\ 0 : p(x) \geq p(y) \end{cases} \quad (1)$$

binary test τ 는 수식 1과 같이 정의한다. p는 smoothed image patch를 뜻하고, p(x)와 p(y)는 x와 y 위치인 밝기 값을 나타낸다. 두 위치의 밝기 값과 비교를 하여 작으면 1, 크거나 같으면 0인 binary test를 구한다.

$$f_n(p) := \sum_{1 \leq i \leq n} 2^{i-1} \tau(p; x_i, y_i) \quad (2)$$

수식 2에서의 n은 vector length이다. 기존의 BRIEF 논문에서는 128, 256 그리고 512 vector length를 사용하고, 본 논문에서는 256 vector length를 사용하였다. 각각 특징점들 주위로 31x31 영상을 만든 후, 임의로 그 주변의 위치 x와 y를 결정하고, binary test를 구한다. 수식 2를 이용하면 총 256 binary descriptor를 생성할 수 있다.

3. Hamming Distance

매칭 알고리즘으로는 Hamming Distance 알고리즘

[5]을 사용한다. DB에 학습된 데이터를 저장하고, 입력 영상에서 FAST와 BRIEF 알고리즘에 의해 나온 Descriptor 정보를 XOR 연산을 통하여 거리를 측정한다. 예를 들어 DB에 학습된 Descriptor 정보가 1000_1000이고, 입력 영상에서 1001_0111이라면, 거리는 5가 된다. 각각 특징점마다 256 bit descriptor를 가지고 있다. DB와 입력 영상에서 검출된 특징점 위치 그리고 descriptor를 가지고, 1대 N 형식으로 입력 영상 특징과 DB 영상 특징이 일치하는 부분을 찾게 된다. 특징점 수가 많이 검출될수록 비교할 것이 많아 속도가 저하된다. 이 점을 보완하기 위해서 병렬로 검사해야 할 필요성이 있다.

III 제안하는 FAST & BRIEF 하드웨어 구조

그림 2는 하드웨어로 구현한 Block Diagram을 보여준다. FAST Corner Detection Block에서 특징점 검출을 한 후, BRIEF Block에서 Descriptor를 생성한다. 그 후, Matching 단계에서 검출된 특징을 비교한다.

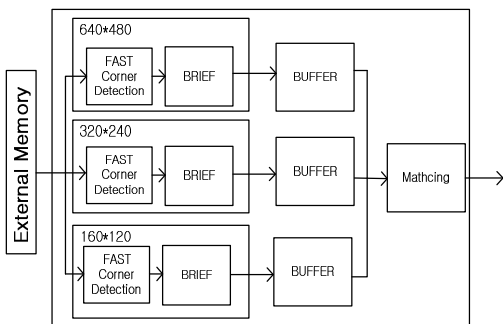


Fig. 2. Designed hardware structure
그림 2. 설계된 하드웨어 구조

1. 특징점 검출 모듈

그림 3은 특징점 검출 모듈을 보여준다. 기본적인 구조 흐름은 FD와 FS 단계에서 검출된 후보 특징점 위치와 Score를 Buffer에 저장한 후, NMS 단계에서 최종 특징점을 검출한다. FD, FS 그리고 NMS 단계는 파이프라인으로 처리된다.

FD 단계에서 후보 특징점이 선택되면, FS 단계로 넘어간다. 후보 특징점이 아니라면, 다음 중심점 픽셀을 읽어 와서 FD 단계에서 후보 특징점인지 판단한다. FS 단계에서 특징점에 대한 Score를 정한 후, Buffer에 특징 좌표와 Score를 저장한다. NMS 단계에서는 특징 좌표와 Score를 Buffer에서 읽어 와서

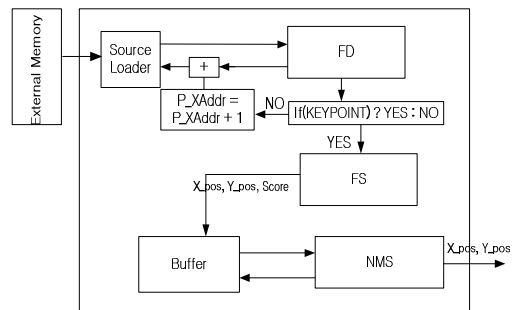


Fig. 3. Keypoint Detection Block
그림 3. 특징점 검출 모듈

주변 인접한 특징점의 Score와 비교한다. 주변 인접한 특징점의 Score보다 값이 크다면, 최종 특징점으로 판단해 특징점 위치를 출력한다.

물체인식을 하기 위해서는 Scale에 강해야 된다. Scale 별로 특징점을 검출하기 때문에 내부 메모리는 640x480, 320x240, 160x120 크기를 저장할 공간이 필요하다. 하지만 그림 4와 같이 Dual Port SRAM을 사용하여 7-Line memory는 Read, 1-Line memory는 Write 하여 640x8KB를 저장할 수 있는 공간만으로 FAST 알고리즘 구현이 가능하다. NMS에서도 이와 같은 방법으로 640x4KB를 저장할 수 있는 Line Memory를 사용한다. 총 640x12 Line Memory를 사용하여 내부 메모리 사용량을 대폭 줄이 수가 있다.

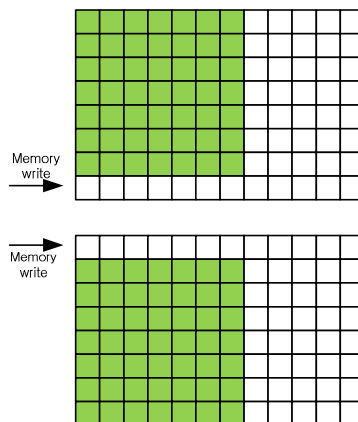


Fig. 4. Memory access pattern
그림 4. 메모리 접근 패턴

설계된 FAST 구조에서 사용하는 총 메모리는 FD와 FS 단계에서 사용하는 640x8, 320x8, 160x8 Memory와 NMS 단계에서 사용하는 640x4, 320x4, 160x4 Memory로 내부 메모리 사용량을 감소 시켰다.

2. 서술자 생성 모듈

그림 5는 서술자(Descriptor) 생성 모듈을 보여 준다. 특징점 검출 단계에서 검출된 특징점 위치인 XPos와 YPos는 FIFO에 저장된다. BRIEF module은 FIFO에 저장된 XPos와 YPos 그리고, Pattern Lookup Table에서 2개의 위치를 읽어 온다.

Descriptor를 생성하기 위해서 FIFO에서 읽어온 특징점 위치와 Pattern에서 읽어온 2개의 위치를 더한 곳에서 밝기 값을 비교한다. 비교하기 위해서 Pattern Lookup Table에서 2개의 위치를 읽어 온다. 특징점 위치 중심으로 Pattern에서 읽어 온 2개의 위치를 더하여 2개의 픽셀의 밝기 값을 검사한다. Pattern은 임의로 지정된 위치인 Lookup Table이다. 이와 같은 방법을 256번 수행하여 256bit Descriptor를 생성하게 된다.

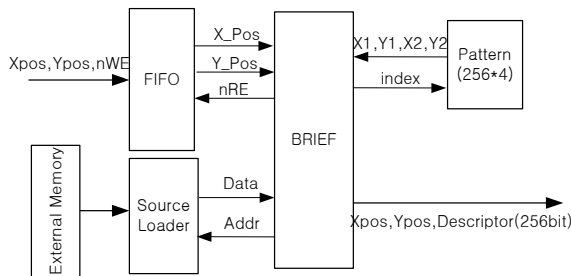


Fig. 5. Descriptor Generation Block.
그림 5. 서술자 생성 모듈.

IV 실험 및 비교

실험 환경으로 Zynq-7000 SoC Platform을 사용했다. 검증 이미지로는 640x480 해상도의 Graffiti와 Sunflower 이미지를 사용하였다. 그림 6은 특징점 검출 모듈에서 검출된 특징점을 보여준다.

표 1은 기존의 SIFT[6], SURF[7] 그리고 FAST & BRIEF 가속기의 LUTs, Registers, Block Rams 사용량을 보여 준다. LUTs 경우, FAST&BRIEF 가속기는 SIFT 가속기 보다 86% 감소했고, SURF 가속기 보다 81% 감소했다.

Registers 와 Block Rams 경우, FAST&BRIEF 가속기는 SIFT 가속기 보다 69% 와 57% 감소를 했고, SURF 가속기보다 64%와 88% 감소했다.

표 2는 기존의 FAST 가속기[8]와 비교한 결과이다. 수행 시간은 오래 걸렸지만, 적은 하드웨어 비용으로



Fig. 6. Resulting Screen for detecting keypoints.
그림 6. 특징점 검출 결과 화면

Table 1. Comparison of Hardware Implementation Cost.
표 1. 하드웨어 구현 비용 비교.

	SIFT[6]	SURF[7]	FAST&BRIEF
Resolution	320x240	640x480	640x480
LUTs	31,357	20,879	4,257
Registers	10,020	8,681	3,187
Block Rams	168KB	589KB	72KB

구현하였다. BRIEF에서 각 특징점마다 Descriptor 생성할 시간이 필요하기 때문에, 특징점이 빠르게 검출될수록, 많은 특징점을 FIFO에 보관해야 된다. FIFO 저장 공간이 커질수록 하드웨어 비용이 많이 필요하다. 하지만, FAST 수행 시간과 BRIEF 수행 시간이 비슷하면, FIFO 에 보관할 특징점 수가 적어 하드웨어 비용이 적게 든다.

표 3은 수행 속도를 보여준다. 본 논문에서 제안한 FAST & BRIEF 가속기를 기존의 SIFT, SURF 가속기 그리고, Exynos4421에서 SW로 구현한 FAST

Table 2. Comparison between the conventional FAST accelerator and proposed FAST accelerator

표 2. 기존의 FAST와 제안하는 FAST 가속기 비교.

	Conventional FAST[8]	Proposed FAST
Register	2,217	669
LUTs	5,034	1,371
Block Rams	40.5KB	18KB
Exec. Time	3ms	17ms

Table 3. Comparison for Execution time.

표 3. 수행속도 비교.

	Exec. Time(ms)	Frequency	Pixel per clock cycle
SIFT [6]	60	50 MHz	0.1024
SURF [7]	18	200 MHz	0.0853
FAST&BRIEF (SW)	45	1.4 GHz	0.0048
FAST&BRIEF (HW)	18	100 MHz	0.1706

&BRIEF 수행 시간과 비교한다.

본 논문의 하드웨어 가속기는 $(640 \times 480) / (100\text{MHz} \times 18\text{ms}) = 0.17 \text{ pixel/clock cycle}$ 의 성능을 보여준다. 비교한 결과 기존의 SIFT와 SURF 가속기에 비해 약 1.6배 수행 속도가 빠르다는 것을 확인할 수가 있다.

FAST&BRIEF 가속기는 SURF와 SIFT에 비해 주 방향을 결정하지 않아 방향 변화에 따른 인식률이 낮다. 하지만, 하드웨어 구현 비용 및 수행 성능 측면에서는 성능이 우수하다.

V 결과

본 논문에서는 Zynq-7000 Soc Platform에서 FAST와 BRIEF 알고리즘을 하드웨어로 구현했다. 기존의 SIFT, SURF 하드웨어 가속기와 하드웨어 비용 및 내부 메모리 사용량을 비교했다. 기존의 SIFT와 SURF 하드웨어는 내부 메모리 와 하드웨어 비용을 많이 사용한다. 하지만, 본 논문은 FAST & BRIEF

하드웨어 설계로 간단한 연산만으로 수행이 가능하다. 기존의 SIFT나 SURF 가속기와 비교한 결과, 하드웨어 비용은 약 70%, 내부 메모리 사용량은 약 57%로 감소했다. 추후, 주 방향을 추가하여 방향에 따른 인식률을 증가할 계획이다

References

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", International Journal of Computer Vision, 60(2):91-110. 1, 2.

[2] H. Bay, E. Andreas, T. Tuytelaars and L. V. Gool, "Speeded-up roubst features", Computer Vision and Image Understanding, Vol 110, Issue 3, pp 346-359, June 2008.

[3] E. Rosten and T. Drummond. "Machine learning for high-speed corner detection", In European Conference on Computer Vision, volume 1, 2006. 1.

[4] M. Calonder, V. Lepetit, C. Strecha, and P .Fua, "Brief: Bi-nary robust independent elementary features. In European Conference on Computer Vision. 2010.

[5] A. Gionis, P. Indyk, and R. Motwani. "Similarity search in high dimensions via hashing",In M. P. Atkinson, M. E. Orłowska, P. Valduriez, S. B. Zdonik, andM. L. Brodie, editors, VLDB'99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK, pages 518 -529. Morgan Kaufmann, 1999. 6.

[6] V.Bonato, "A Parallel Hardware Architecutre for Scale and Rotation Inavariant Feature Detection", IEEE Transactions on Circuits and Systems for Video Technology, Vol.18, No.12, 2008.

[7] D. Bouris, T. Tuytelaars, and L.V. Gool, "SURF: Speeded Up Robust Features", 9th European Conf. Computer Vision, pages 404-417, 2006.

[8] K.Taek-Kyu, "An Embedded FAST Hardware Acclerator for Image Feature Detection", Journal of The Institute of Electronics Engineers, pages 28-34, 2012,3

BIOGRAPHY

Heo Hoon (Student Member)

2012: BS degree in Computer Engineering, Seokyeong University
 Present: MS course in Electronics Computer Engineering, Seokyeong University
 <Research interests>
 Microprocessor, Embedded System, Image Processing

Lee Kwang Yeob (Life member)

1985. 8 Seogang University, Dept. of Electronics Engineering(BS)
 1987. 8 Yonsei University, Dept. of Electronics Engineering(MS)
 1994. 2 Yonsei University, Dept. of Electronics Engineering(Ph.D)
 1989 ~ 1995. 2 Hyundai Electronics Inc., Senior Researcher
 1995.3 ~ Seokyeong Univeristy, Dept. of Computer Engineering, Professor
 <Research interests> Microprocessor, Embedded System, 3D Graphics System