

영역기반 스테레오 영상 정합을 위한 고속 SAD 알고리즘

이우영*, 김정길** 종신회원

A Fast SAD Algorithm for Area-based Stereo Matching Methods

Woo-Young Lee*, Cheong Ghil Kim** *Lifelong Member*

요 약

스테레오 비전 분야에서 영역 기반의 영상 정합은 스테레오 영상 분석을 위한 대표적인 방법이다. SAD (Sum of Absolute Difference) 알고리즘은 영역 기반 정합 알고리즘의 한 종류로서 대규모 데이터 집약적 계산을 요구하여 소프트웨어 방식을 사용할 경우 처리속도가 매우 느리게 된다. 본 논문에서는 소프트웨어 기반 SIMD (Single Instruction Multiple Data) 병렬 기법인 SSE (Streaming SIMD Extensions) 명령어를 이용한 고속 SAD 알고리즘을 제안한다. SSE 명령어를 지원하는 CPU는 16개의 128비트 크기의 XMM 레지스터를 보유하여 SIMD 명령어 집합 확장을 가능하게 하였다. 제안하는 소프트웨어 기반 병렬 고속화 기법의 성능 측정을 위하여 일반적 SAD를 이용한 영상 정합 알고리즘과 SSE 명령어를 사용한 알고리즘의 수행 속도차이를 측정하였다. 제안하는 기법은 일반적 SAD 알고리즘보다 평균 4배의 성능 향상의 결과를 보임으로 소프트웨어 기반 고속병렬 처리를 통한 실시간 스테레오 비전 응용분야에 효과적으로 적용될 수 있음을 보였다.

Key Words : SAD, SIMD, SSE, Stereo Matching, Parallel Processing

ABSTRACT

Area-based stereo matching algorithms are widely used for image analysis for stereo vision. SAD (Sum of Absolute Difference) algorithm is one of well known area-based stereo matching algorithms with the characteristics of data intensive computing application. Therefore, it requires very high computation capabilities and its processing speed becomes very slow with software realization. This paper proposes a fast SAD algorithm utilizing SSE (Streaming SIMD Extensions) instructions based on SIMD (Single Instruction Multiple Data) parallelism. CPU supporting SSE instructions has 16 XMM registers with 128 bits. For the performance evaluation of the proposed scheme, we compare the processing speed between SAD with/without SSE instructions. The proposed scheme achieves four times performance improvement over the general SAD, which shows the possibility of the software realization of real time SAD algorithm.

I. 서 론

절대 차이의 합계를 구하는 SAD (Sum of Absolute Difference) 연산은 이미지 사이의 유사성을 측정하기 위하여 사용되는 알고리즘으로 스테레오 이미지 및 움직임 추정을 위한 비디오 압축 등의 분야에서 널리 사용되고 있으며[1-3] 대표적인 스트림 데이터(streaming data) 처리 응용분야라고 할 수 있다. 특히, 최근 들어 다양한 3D 기기 및 스마트폰, 디지털 카메라, 등 차세대 PC 등 환경에서 스트림 데이터 처리를 수행하여야 하는 멀티미디어 응용이 증가하고 있다.

실시간 스테레오 비전 응용분야에서 영상정합 그리고 동 영상 코딩 과정에서 움직임 추정은 핵심 기능으로 이를 위해

서는 보통 수백번 이상의 SAD 연산이 필요하며 고속의 연산을 위하여는 SAD 알고리즘의 효율적 구현[4,7]이 필수적이며 전용 하드웨어를 구현을 위한 여러 연구가 진행되고 있다[1-4]. 일반적으로 스테레오 비전 시스템의 알고리즘을 처리하는데 있어서 소프트웨어 방식을 사용하면 200x200 화소 크기를 가지는 상대적으로 작은 크기의 영상으로 가정하고 응용 분야에 따른 전-후처리 과정을 무시하더라도 초단위의 시간이 필요하게 된다[5].

최근 CPU의 발전으로 하나의 명령어로 다수의 데이터를 처리할 수 있는 SIMD (Single Instruction Multiple Data) 병렬처리 기반의 MMX (Multimedia Extension) 명령어 등이 소개되고 있다[6]. 대부분의 스트림 데이터 처리를 위한 프로

* 본 연구는 문화체육관광부 및 한국콘텐츠진흥원의 2012년 콘텐츠산업기술기원사업으로 수행되었음.

*건아정보기술(주) 건아부속연구소(lwyllove@naver.com), **남서울대학교 컴퓨터학과 (cgkim@nsu.ac.kr), 교신저자 : 김정길

접수일자 : 2012년 7월 6일, 수정완료일자 : 2012년 7월 17일, 최종게재확정일자 : 2012년 7월 23일

그림은 루프 구조를 사용하여 반복적인 명령을 수행하기 때문에 시간 지역성과 공간 지역성을 갖고 있으며 공통적으로 데이터 병렬처리에 적합한 SIMD 구조가 효율적으로 적용 가능하다.

본 논문에서는 소프트웨어 기반 SIMD 병렬처리 기법인 SSE (Streaming SIMD Extensions) 명령어를 이용하여 스테레오 비전 시스템의 대표적인 알고리즘 중 하나인 SAD 연산의 소프트웨어 기반의 고속 병렬화 기법을 제안한다.

본 논문의 구성은 2장에서는 SAD 알고리즘에 대하여 소개하고 3장에서는 SSE를 기술한다. 4장에서는 제안하는 SAD 알고리즘의 고속화 기법을 소개하며 5장에서의 실험 방법 및 결과를 기술한다. 마지막으로 6장에서는 결론을 맺는다.

II. SAD 알고리즘

영역 기반 영상 정합은 윈도우(window: 일정 크기의 이웃 화소들의 집합 단위)의 영역을 연속적으로 비교하는 알고리즘이다. (그림 1)은 영역 기반 정합 알고리즘을 이용한 영상 정합 과정을 보여주며 오른쪽 영상의 기준 윈도우 대하여 왼쪽 영상의 윈도우를 가로축으로 이동 시키면서 각 윈도우 쌍간의 유사성을 계산하여 그 대응점을 결정한다[5].

본 논문에서는 영역 기반 정합 알고리즘의 한 종류인 SAD 알고리즘을 구현하였다. 이는 최적의 정합을 각 윈도우의 대응되는 화소간 차이에 대한 절대값의 합(SAD값)이 최소가 되는 화소의 변이 값으로 정의한다. 식(1)은 개별 윈도우의 SAD 값을 구하는 과정을 나타낸다.

$$SAD(X, Y, R) = \sum_{Y=0}^{H-1} \sum_{X=0}^{W-1} |SI(X, Y) - TI((X+R), Y)| \quad (1)$$

H = 윈도우의 열길이
 W = 윈도우의 행길이
 R = 윈도우 차이값
 SI = StandardImage
 TI = TargetImage

두 영상 사이의 최적의 정합점을 계산하기 위하여 윈도우 차이값 R 을 0에서 최대 차이값까지 변화시키면서 계산되는 최대 차이값+1개의 SAD 결과 중 최소값을 갖는 차이값 R 을 최적값으로 선택한다. SAD 정합 알고리즘은 최대 차이값에 따라 계산할 수 있는 거리 범위가 결정되며, 최대 차이값이 커질수록 계산해야 하는 SAD 값의 수가 증가하므로 계산량이 증가하게 되고, 비슷하게 생긴 다른 물체가 최대 시차 범위 내에 존재할 경우 최적값 계산에 있어 오류가 발생할 가능성이 증가한다. 반면 최대 차이값을 지나치게 작게 설정하는 실제 유사 영역이 기준 시야 이내에 들어오지 못하는 오류가 발생함으로 적절한 최적값을 얻어낼 수 없다. 따라서 최대 차이값은 계산량을 고려하여 설정하되 지나치게 작아

저서 오류가 생기지 않도록 해야 한다[5].

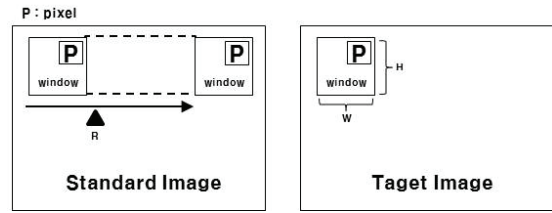


그림 1. 영역 기반 정합 알고리즘의 영상 정합 과정

III. MMX와 SSE

MMX 는 하나의 명령어로 다수의 데이터를 처리 할 수 있는 병렬처리 구조로서 대표적인 프로세서 제조회사인 Intel과 AMD의 CPU에는 SIMD 연산에만 사용되는 128비트 레지스터와 SIMD 명령어의 집합인 SSE(Streaming SIMD Extensions)와 3DNOW가 각각 설계되었다. Intel의 SIMD 명령어는 1997년 Pentium II의 MMX에서 출발하여 현재 SSE4까지 지속적으로 발전하고 있으며 MMX를 명령어를 설계하여 작은 규모의 SIMD 병렬처리가 가능하게 함으로써 범용 컴퓨터에서도 고속의 과학 연산 및 영상처리 응용에 활용되고 있다[8].

MMX 초기에는 부동 소수점 연산용 레지스터를 공유하여 실행되었으나, SSE는 128비트의 전용 레지스터를 사용한다. 이를 이용하여 사용하고자 하는 연산 자료형 별로 지정하여 데이터를 저장하여 동시 연산을 수행한다. 지원하는 자료형은 8bit, 16bit, 32bit 이다. 예를 들어, 8bit형의 경우 한번에 16개의 다른 데이터를 동일 연산을 동시에 처리 가능하다. (그림 2)는 SSE 명령어가 처리 가능한 자료의 형태를 보여준다.

32bit				32bit				32bit				32bit			
16bit		16bit		16bit		16bit		16bit		16bit		16bit		16bit	
8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit	8bit

그림 2. SSE의 처리 가능 자료 형태

본 논문에서 제안되는 SAD를 이용한 동일 추측 영역 추측 연산은 윈도우를 기준 영상과 목표 이미지에 대하여 각각 추출하고 그 윈도우에 관하여 SAD 연산을 수행할 때 SSE 명령어를 사용하여 데이터 병렬처리 기반의 고속 연산을 목표로 한다. (그림 3)에서는 유사 이미지에서의 윈도우 추출 방식을 그림으로 표현 해주며 (그림 4)에서는 SSE 명령어를 사용하지 않은 일반적인 SAD 연산을 그림으로 표현한다.

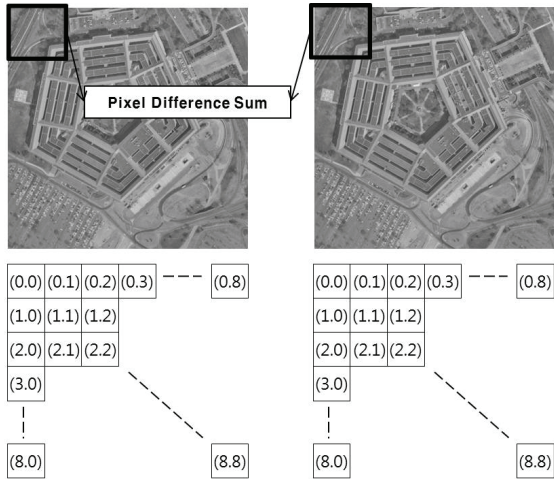


그림 3. 유사 이미지에서들 간의 윈도우 추출

$$\left(\begin{matrix} (0,0) \\ (0,0) \end{matrix} \right) + \left(\begin{matrix} (0,1) \\ (0,1) \end{matrix} \right) + \left(\begin{matrix} (0,2) \\ (0,2) \end{matrix} \right) + \dots + \left(\begin{matrix} (0,8) \\ (0,8) \end{matrix} \right)$$

그림 4. 일반적인 윈도우별 SAD 연산

(그림 4)의 연산은 각 픽셀마다 검색 범위만큼 연산을 수행하여 추출된 데이터를 비교하여 한 지점의 깊이 데이터를 추출 하게 된다. 결국 이미지의 크기와 검색 범위의 크기에 비례하여 연산 횟수가 증가하게 된다. (그림 5)는 이러한 데이터 병렬 연산을 SSE 명령어를 이용하여 SAD 연산에 적용한 연산 과정을 그림으로 보여준다.

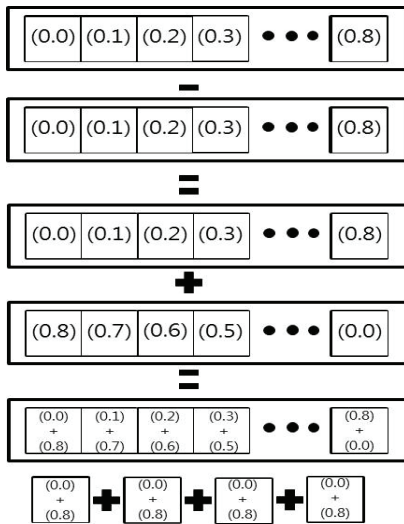


그림 5. SSE 기반의 SAD 연산

IV. 실험 환경 및 결과

본 장에서는 실험 환경 및 방법을 소개하고 결과를 기술한다. <표 1>은 실험에 사용된 하드웨어를 요약하였으며 사용된 유사 이미지는 같은 지점을 서로 다른 각도에서 촬영된

카네기 멜론 대학의 StereoImage[9]를 사용하였다. 크기는 가로 512픽셀, 세로 512픽셀의 크기를 사용하였다. 동일한 위치 추출에 대한 윈도우 사이즈는 8X8을 사용하였다. 검색 범위는 127로 설정하였으며 정해진 픽셀 위치에서 0~127까지의 방향을 순차적으로 검색한다. 8X8 마스크는 SSE의 128bit의 메모리를 효율적으로 사용할 수 있으며 넓은 범위를 통한 정확성을 위해 선택 되었다. <표 2>는 실험에 사용된 유사 이미지 및 설정 사양을 요약한다.

표 1. 실험에 사용된 하드웨어 환경

	소개
CPU	Intel Core I7 CPU Q 740 @1.73GHz
RAM	8.00GB
System	Windows 7 (64bit)
Compiler	Microsoft Visual Studio 2010

표 2. 실험에 사용된 이미지 및 설정 사양

	소개
Image sizes	512 X 512 pixels
Search rage	127(0 ~ +127)
Mask size	8 X 8

본 실험 방법은 유사 이미지에 대하여 SAD 연산을 통하여 동일 추출 지역을 추출해 내고 그 부분의 검색 위치에 따라 이미지화 Gray 톤으로 표시하였다. 유사 이미지는 동일 지역을 다른 각도에서 촬영하였기 때문에 동일한 물체에 대하여 비슷한 검색 위치를 가지게 되어 같은 물체에 관한 검색위치에 따른 데이터가 같은 값을 가지게 된다. 이러한 데이터를 이미지화 했을 경우 동일 물체는 동일한 Gray 톤으로 표시되어 동일 지역에 대한 이미지가 표현되도록 하였다. (그림 6)은 본 실험 과정을 도표로 나타내었다.

연산 속도 측정 구간은 이미지의 모든 픽셀에 대한 SAD 연산 및 최적값이 구해질 때까지의 계산 시간을 측정하였다. 초기 ImageLoad 및 GrayScale 과정과 계산된 최적값을 통한 Image화 단계는 동일한 함수를 사용하므로 연산 속도 측정 부분에서 제외하였다. 속도 측정 방법은 시스템 타임을 사용하였으며 1/100초 단위의 ms단위로 표시하며 실행 시간의 차를 이용 백분율로 표시하였다.

유사 이미지 추출의 경우 배경과 같은 반복된 이미지 혹은 특징점이 없는 부분에 대해서는 동일 추출 지역에 있어 추출 정확도가 낮았으나 객체에 대하여는 동일 추출 지역에 대한 이미지가 추출 되었다. 또한 SSE 명령어를 사용하지 않은 경우 연산 시간은 약 14687.8401ms 소요되었으며, SSE 명령어를 사용한 경우는 3288.1881ms의 처리 시간이 소요되어 고속화 결과를 보여주었다. 이는 영상 정합 알고리즘의 연산 부분에 대하여 일반적 SAD 기반 동일 추출 지역 추출 연산에 비하여 SSE 명령어를 사용한 경우가 약 440%의 정

도의 고속화 성능 향상 결과를 보여준다. <표 3>은 속도 성능 비교를 나타내고 있으며 (그림 7)은 결과 화면을 표시하고 있다.

V. 결론

본 논문에서는 소프트웨어 기반 SIMD (Single Instruction Multiple Data) 병렬 기법인 SSE (Streaming SIMD Extensions) 명령어를 이용한 고속 SAD 알고리즘을 제안하였다. SSE 명령어를 지원하는 CPU는 16개의 128비트 크기의 XMM 레지스터를 보유하여 SIMD 명령어 집합 확장을 가능하게 하였다. 제안하는 소프트웨어 기반 병렬 고속화 기법의 성능 측정을 위하여 일반적 SAD를 이용한 영상 정합 알고리즘과 SSE 명령어를 사용한 알고리즘의 수행 속도차이를 측정된 결과, 제안하는 기법은 일반적 SAD 알고리즘보다 평균 4배의 성능 향상의 결과를 보임으로 소프트웨어 기반 고속병렬 처리를 통한 실시간 스테레오 비전 응용분야에 효과적으로 적용될 수 있음을 보였다.

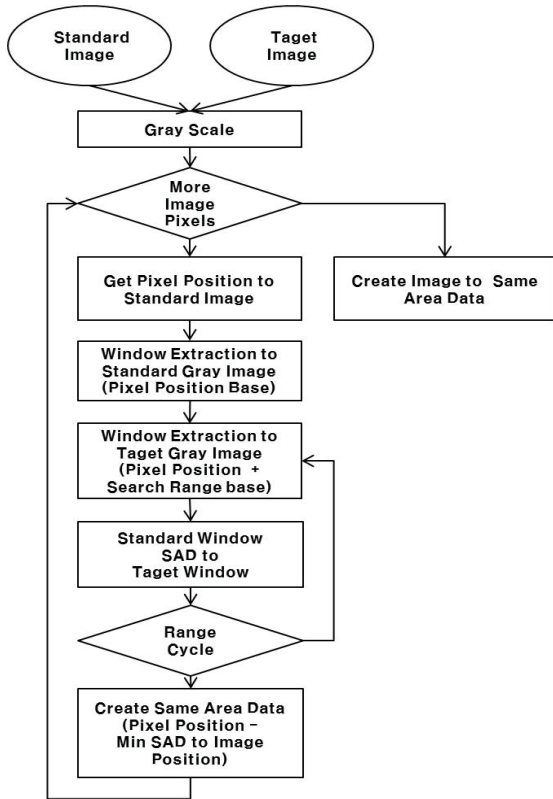


그림 6. SAD 연산 흐름도

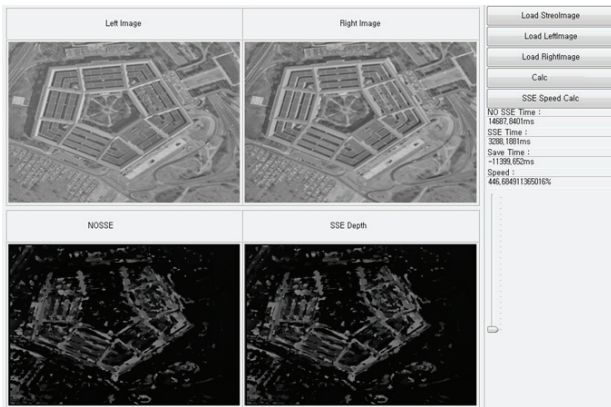


그림 7. 일반적 SAD와 SSE SAD 프로그램 결과 화면

표 3. 일반적 SAD와 SSE SAD의 속도 성능 비교

	소개
SAD w/o SSE	Average 14687.8401ms
SAD w/ SSE	Average 3288.1881ms
Time difference	11399.652ms
Performance improvement ratio	446.%

참고 문헌

- [1] Jongsu Yi, Jaehwa Park, and JunSeong Kim, A Platform-Based SoC Design for Real-Time Stereo Vision, Journal of Semiconductor Technology and Science, Vol. 12, No. 2, June 2012, pp. 212-218.
- [2] P. Muralidhar, C. B. Rama Rao, and I. Ranjith Kumar, Efficient Architecture for Variable block size Motion Estimation of H.264 Video Encoder, 2012 International Conference on Solid-State and Integrated Circuit (ICSIC 2012), pp. 1-6.
- [3] Cheong Ghil Kim, Vason P. Sirmi, and Shin Dug Kim, "High Performance Coprocessor Architecture for Real-Time Dense Disparity Map," 정보처리학회 논문지, 제14-A권 제5호, 2007. 10, pp. 301-308.
- [4] 박장호, 최현준, 박성호, 서영호, 김동욱, "효율적인 SAD 연산을 위한 하드웨어 구현에 대한 연구" 2008년도 한국방송공학 회 학술대회, 2008.11, 223-226
- [5] 이정수, 양승구, 김준성, 실시간 스테레오 비전 시스템을 위한 SAD 정합연산기 설계, 전자공학회지 제 45권 제1호 통권 제 319호, 2008. 1, pp.55-61.
- [6] C. G. Kim, H. S. Kim, S. H. Kang, S. D. Kim and G. H. Han, An Acceleration Processor for Data Intensive Scientific Computing, IEICE Trans. on Information and Systems, Vol. E87-D, 2004. 7, pp.1766-1773.
- [7] 유희재, 정선태, 정수환, 2007. 12, VLIW 기반 고성능 DSP에서의 SAD 알고리즘 최적화 스케줄링, 한국 콘텐츠 학회 논문지 제 7권 제 12호, pp. 262-272.
- [8] Cheong Ghil Kim, "Evaluation of Image Pre-processing using Subword Parallelism," 2010년도 한국산업정보학회 춘 계학술대회 논문집, pp 93-94, 2010. 05.
- [9] <http://vasc.ri.cmu.edu/idb/html/stereo/index.html>

저자

이 우 영(Woo-Young Lee)



- 2011년 2월 : 남서울대학교 컴퓨터학과 학사졸업
- 2011년 2월 ~ 현재 : 건아정보기술(주) 연구소

<관심분야> SIMD, Stereo

김 정 길(Cheong Ghil Kim)

종신회원



- 1987년 : Univ. of Redlands, U.S.A. 컴퓨터과학 학사졸업
- 2003년 : 연세대학교 컴퓨터과학 공학 석사
- 2006년 : 연세대학교 컴퓨터과학 공학 박사

- 2006년 ~ 2007년 : 연세대학교 컴퓨터과학과 박사후연구원
- 2007년 ~ 2008년 : 연세대학교 컴퓨터과학과 연구교수
- 2008년 ~ 현재 : 남서울대학교 컴퓨터학과 교수

<관심분야> 멀티미디어 임베디드 시스템, 이기종 컴퓨팅, 모바일 AR, 3D Contents