

# AD-Census 기반 스테레오 비전 시스템의 효율적인 하드웨어 구조

조은별\*, 박태근<sup>o</sup>

## Efficient Hardware Architecture of a Stereo Vision System Based on AD-Census

Eunbyeol Cho\*, Taegeun Park<sup>o</sup>

### 요약

AD-Census 알고리즘은 기존 알고리즘인 AD(absolute difference)방식과 CT(census transform)방식을 결합하여 화소의 색 차이와 주변화소와의 관계를 통해 비용함수를 구하기 때문에 높은 정합률을 보인다. 또한 입력영상의 상관관계에 따라 비용을 누적하는 부정형 가변 윈도우를 사용하여 비교적 단순한 연산으로도 정합의 정확도를 높일 수 있다. 본 논문에서는 AD-Census 알고리즘과 부정형 가변 윈도우 알고리즘을 기반으로 정확도가 높으면서도 효율적인 실시간 스테레오 비전 시스템을 제안하였다. 데이터 재사용과 병렬처리 기법을 사용하여 데이터 처리를 최적화하였으며 성능을 개선하였다. 제안한 구조는 동부하이텍 110nm 표준셀 라이브러리로 합성하여 츠쿠바 영상 기준 2.47%에러율과 192MHz(약 59fps)의 최대 동작 주파수를 나타내며 381K개의 게이트를 사용한다.

**Key Words** : stereo vision, hardware architecture, census transform, realtime system, VLSI design

### ABSTRACT

Since the AD-Census algorithm that combines the existing the AD (absolute difference) and the CT (census transform) algorithms computes the matching cost by using the pixel intensity difference and the relative ordering of the pixel intensity, it shows high matching performance. Also it utilizes the unfixed and variable-sized window, it can obtain high matching quality with relatively simple arithmetic. In this paper, we propose an efficient realtime stereo vision system with high accuracy using the AD-Census algorithm and the unfixed, variable-sized window. We improve the performance by using the data reuse and the parallel processing. The proposed architecture has been synthesized using DonbuHitek 110nm standard cell library. The proposed system shows 2.47% of error rate and can run up to 192MHz (59fps) operation frequency with 381K gates.

### I. 서론

스테레오 비전은 두 장 이상의 평면 이미지를 처리

하여 깊이 정보를 계산하는 기술이다. 이러한 스테레오 비전 알고리즘은 장애물 인식, 로봇비전<sup>[1]</sup>, 무인자 동차<sup>[2]</sup> 등 다양한 응용 분야에서 사용되고 있는 데, 하

\* 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2011-0009513).

※ 본 연구는 2017년도 가톨릭대학교 교비연구비의 지원으로 이루어졌음(M-2017-B0002-00014).

• First Author : Department of Information, Communication, and Electronic Engineering, The Catholic University of Korea, lkrkr02@gmail.com, 학생회원

<sup>o</sup> Corresponding Author : Department of Information, Communication, and Electronic Engineering, The Catholic University of Korea, parktg@catholic.ac.kr, 종신회원

논문번호 : KICS2016-10-298, Received October 10, 2016; Revised April 14, 2017; Accepted June 15, 2017

드웨어로 설계하기 위해서는 정합의 정확도와 실시간 처리 여부가 중요하게 고려되어야 한다<sup>3)</sup>.

스테레오 비전은 정합 방법에 따라 전역 정합과 지역 정합으로 나누어진다<sup>4)</sup>. 전역 정합은 전체 영상의 정보를 사용하여 한 화소의 깊이정보를 얻기 때문에 정합률은 좋지만 복잡도가 크게 증가하여 실시간 처리 시스템을 설계하기 어렵다. 반면 지역 정합은 영상의 일부분만을 사용하므로 정보에 민감하여 정합률이 낮지만 상대적으로 알고리즘이 단순하여 하드웨어로 설계하기 용이하다는 특징이 있다. 전역 정합 방법으로는 동적 계획법(dynamic programming), 그래프 컷(graph cut), 신뢰 확산(belief propagation) 알고리즘 등이 있으며, 지역 정합법으로는 절댓값 누적 정합(SAD, sum of absolute difference), 차이의 제곱 누적 정합(SSD, sum of squared differences), 센서스 변환(CT, census transform) 등이 있다.

다양한 알고리즘과 특징을 가진 실시간 스테레오 비전 시스템에 대한 많은 연구가 진행되어 왔다<sup>7-12)</sup>. Yang 등은 7×7 윈도우를 사용한 SAD기반의 시스템을 FPGA로 구현하였다<sup>5)</sup>. 또한 Jin 등은 직교화(rectification) 과정, 여러 검출 등의 스테레오 매칭 전, 후처리를 포함하고, 64의 탐색거리를 지원하는 Census기반 시스템을 제안하였다<sup>6)</sup>. Ambrosch 등은 변화도 기반의 MCT(modified census transform)와 색상 차이 기반의 SAD를 적절히 응용하여 시스템을 구현하였으며<sup>7)</sup>, Ttofis 등은 입력영상 정보에 따라 가중치를 부여하여 누적한 방식을 사용한 ADSW(adaptive support weight) 시스템을 구현하였다<sup>8)</sup>. 전역 정합을 기반으로 한 시스템으로는 DP 알고리즘에서 수직 방향으로 다시 적분하여 수직 잡음을 개선한 RealDP(real dynamic programming)가 Forstmann에 의해 제안되었다<sup>9)</sup>. 또한, 상관관계를 고려하여 BP알고리즘을 적용한 HBP(hierarchical belief propagation) 시스템이 Yang 등에 의해 구현되었다<sup>10)</sup>.

지역 정합법으로는 영역 기반 알고리즘이 많이 사용되는 데, 주로 차이의 절댓값이나 제곱, 센서스 변환 등과 같이 간단한 비용함수를 정의된 윈도우 영역만큼 누적하여 정합여부를 판단하기 때문에 윈도우 크기와 모양은 정합률에 큰 영향을 미친다. 영역 기반 알고리즘에서 정합하는 대상에 따라 적합한 윈도우를 결정하는 다양한 연구가 진행되었지만, 여전히 NP-hard 문제로 복잡도가 크다.

기존의 영역 기반 알고리즘은 고정된 윈도우의 특성으로 인해 정합률이 낮았다. 예를 들어, SAD의 경

우, 전체적 혹은 부분적 명도, 사물의 경계선 등의 색상 차이에 민감하며 텍스처가 뚜렷하게 드러나지 않는 영역에서 정합의 정교함이 떨어진다. 센서스 변환을 이용한 알고리즘은 주변화소와의 크고 작음만을 구분하기 때문에 화소들의 관계성만으로는 알 수 없는 반복적 영역에서의 오류가 높다. 또한 기존의 고정된 윈도우는 왜곡과 텍스처를 구분하기 어려워 정확도가 낮았다. 이를 개선하기 위해 가중치 알고리즘이 제안되었지만 가중치를 계산하는 과정에서 복잡한 연산이 포함되므로 하드웨어가 상대적으로 복잡하다.

본 논문에서는 AD-Census알고리즘과 부정형 가변 윈도우의 비용값을 누적하는 방식을 사용하여 기존의 영역 기반 알고리즘의 단점을 줄이고, 고정된 윈도우의 문제를 해결하는 방안을 제안하였으며, 비용함수 모듈, 비용 윈도우 모듈, 비용 누적 모듈 등을 포함한 AD-Census 스테레오 시스템을 구현하였다. 비용함수 모듈을 구현할 때에는 비교할 화소수를 최소화하면서도 데이터 재사용을 통해 연산을 줄일 수 있는 포인트 방식의 센서스 연산을 사용하였으며, 비용 누적 모듈에서는 데이터 재사용과 병렬처리를 적절히 활용한 스케줄링을 통해 효율적으로 설계하였다. 알고리즘에 사용되는 지수연산은 LUT(look-up table)로 설계하였으며 이를 위하여 경험적으로 근사화하여 허용할 수 있는 오차 범위를 갖도록 하였다.

본 논문의 구성은 다음과 같다. II장에서는 스테레오 정합 알고리즘으로 사용된 AD-Census알고리즘에 대하여 설명하였고 III장에서 제안한 시스템의 하드웨어 구조와 스케줄링에 대해 설명하였다. IV장에서 설계 결과를 다른 스테레오 시스템과 성능을 비교 분석하였으며, V장에서 결론을 맺는다.

## II. AD-Census 알고리즘

AD-Census 알고리즘은 AD(absolute difference)와 CT(census transform)를 결합해 비용함수를 계산하고, 부정형 가변 영역을 누적하여 정합하는 스테레오 알고리즘이다[11]. AD-Census 알고리즘은 주변 화소와의 관계성뿐만 아니라 색상 차이도 민감하게 살피기 때문에 다른 지역 정합법에 비해 정합률이 크게 개선되었다. 또한 사용된 부정형 가변 윈도우는 가중치 알고리즘과 달리 복잡한 연산 없이 적용 가능하며 기준 화소와 윈도우 내 주변화소의 상관관계에 따라 윈도우의 모양과 크기를 다르게 정의하여 어려움을 줄일 수 있다.

### 2.1 AD-Census 비용함수

AD-Census 비용함수는 AD와 CT 기반의 비용합수를 결합하여 색상 차이에 민감한 AD방식의 단점을 보완하면서 주변 화소와의 관계를 기반으로 판단하는 센서스 방식의 장점을 살려 방사 측정 변화와 이미지 노이즈로 생긴 값의 변화에 강인하다.

$$C_{AD(p,d)} = \frac{1}{3} \sum_{i=R,G,B} |I_i^{L(p)} - I_i^{R(p-d)}| \quad (1)$$

AD기반의 비용값  $C_{AD}$ 는 두 입력영상의 중심화소 값 차이로 정의되며, 식(1)과 같이 두 영상의 R, G, B 색상 차이를 평균내어 AD기반의 비용값으로 사용한다. 이 때, 흑백 영상의 경우,  $C_{AD}$ 는 두 영상의 중심화소 차이값으로 구할 수 있다.

$$CV_{(x,y)} = \begin{cases} 0 & | P_{(x,y)} > R_{(x',y')} \\ 1 & | P_{(x,y)} \leq R_{(x',y')} \end{cases} \quad (2)$$

CT 기반의 비용값  $C_{Census}$ 를 구하기 위해서는 식(2)와 같이 두 입력영상의 센서스 벡터  $CV_{(x,y)}$ 를 구해야 한다.  $CV_{(x,y)}$ 는 중심화소인  $P_{(x,y)}$ 와 주변화소인  $R_{(x',y')}$ 의 값을 비교한 벡터로, 중심화소가 주변화소보다 클 경우, 0이 되고, 중심화소가 주변화소보다 작을 경우, 1이 된다.

$$HD_{(x_l, x_r, y)} = \sum (CV_{(x_l, y)}^L \otimes CV_{(x_r, y)}^R) \\ C_{Census(p,d)} = HD_{(p,p-d,y)} \quad (3)$$

두 입력 영상의  $CV_{(x,y)}$ 를 통해 식(3)과 같이 HD (hamming distance)를 구하며 그 값은  $C_{Census}$ 가 된다.

$$\rho_{(c,\lambda)} = 1 - \exp(-\frac{c}{\lambda}) \quad (4) \\ C_{(p,d)} = \rho(C_{Census(p,d)}, \lambda_{Census}) + \rho(C_{AD(p,d)}, \lambda_{AD})$$

AD-Census의 최종 비용값  $C_{(p,d)}$ 는 식(4)와 같이 지수함수인  $\rho_{(c,\lambda)}$  함수로 정의되며 파라미터  $\lambda$ 에 의해 두 값에 대한 민감성 정도가 조절된다.

### 2.2 부정형 가변 윈도우 누적 알고리즘

고정된 윈도우를 사용한 영역 기반 알고리즘은 높은 정확도를 얻기 힘들며, 가중치 알고리즘은 계산 복잡도가 높다는 단점이 있다. 부정형 가변 윈도우 누적 알고리즘은 윈도우 기반의 알고리즘을 사용하여 하드웨어 설계가 용이하며 화소의 연관성에 따라 비용값을 누적하기 때문에 높은 정확도를 나타낸다.

부정형 가변 윈도우 누적 알고리즘은 그림 1과 같이 상관성이 높은 부정형 영역에 대해 값을 누적하며 정의하면 다음과 같다.

$$D_{c(p_n,p)} = \max_{i=R,G,B} |I_i(p_n) - I_i(p)| \\ D_{s(p_n,p)} = p_n - p \quad (5)$$

부정형 윈도우는 식(5)와 같이 색 차이  $D_c$ 와 공간 거리 차이  $D_s$ 로 정의되며,  $D_c$ 와  $D_s$ 는 상, 하, 좌, 우 4가지 방향에서 (6)의 조건을 만족해야 한다.

$$1. D_{c(p_n,p)} < \tau_1 \text{ and } D_{c(p_n,p_{n-1})} < \tau_1, D_{s(p_n,p)} < L_2 \\ 2. D_{c(p_n,p)} < \tau_2, L_2 < D_{s(p_n,p)} < L_1 \quad (6)$$

부정형 윈도우는 그림 1에서와 같이 먼저 중심화소가 속한 열 방향(상, 하)으로 (6)조건을 만족하는지 판단하고, 이어서 행 방향(좌, 우)으로 탐색하기 시작한다. 탐색된 부정형 윈도우 영역에서만 비용값을 누적하며, 이 때 윈도우의 크기와 모양은 최종 정합값을 결정하는 데에 큰 영향을 미치므로 부정형 윈도우를 정의하고 있는 파라미터  $L_1, L_2, \tau_1, \tau_2$ 를 적절하게 선택해야 한다.

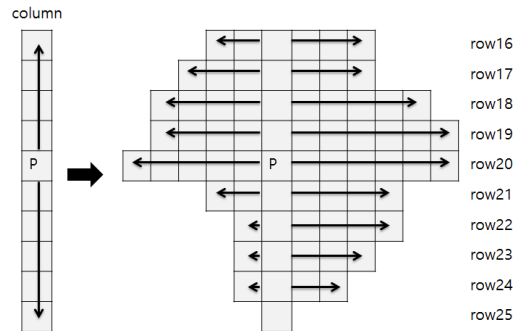


그림 1. 비용 누적 윈도우  
Fig. 1. Cost aggregation window.

## III. 제안한 구조

### 3.1 제안한 스테레오 시스템의 구조

그림 2는 제안한 AD-Census 스테레오 시스템의 전체 구조로 입력 영상을 저장하는 두 개의 메모리와

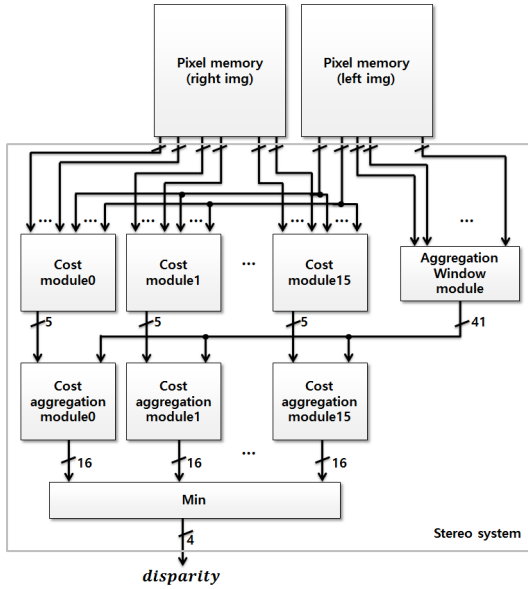


그림 2. 제안된 AD-Census 스테레오 시스템.  
Fig. 2. Proposed AD-Census stereo system.

AD-Census기반으로 비용을 계산하는 비용함수 모듈(cost module), 부정형 윈도우 영역을 정의하는 누적 윈도우 모듈(aggregation window module), 누적 윈도우 정보와 비용값을 입력받아서 비용값을 누적하는 비용 누적 모듈(cost aggregation module), 최종 정합값을 결정하는 비교기(min)로 구성되었다.

본 연구에서는 탐색거리로 16을 사용하였으며 16개의 AD-Census기반의 비용함수 모듈과 비용 누적 모듈을 병렬 처리하여 최종 정합값을 계산하였다. 16개의 비용 함수 모듈의 결과는 비용 누적 모듈 내의 메모리에 저장되며 비용 누적 연산에 이용된다. 비용 누적 모듈은 부정형 윈도우 영역을 나타내는 윈도우 비트 스트림을 입력받아 해당하는 위치의 비용값들을 누적한다. 이 때, 모든 비용 누적 모듈에는 같은 비트 스트림을 입력값으로 받아들인다. 하나의 병렬처리를 통해 계산된 탐색거리에 따른 16개의 정합값들은 토너먼트 형식으로 구성된 비교기를 통해 정합값을 결정하도록 하였다. 입력 영상 메모리와 비용함수 모듈들 간의 비교적 많은 입출력은 설계에 부담이 될 수 있다. 이런 문제점은 메모리를 적절히 분할(partition)하여 동시에 많은 데이터에 접근(access)하거나 메모리 파이프라인 기법인 인터리빙(interleaving) 방법을 적용하여 처리량을 높이는 방식으로 개선할 수 있다.

그림 3은 비용함수 모듈의 구조를 나타낸다. 비용 함수 모듈은 두 픽셀 이미지의 중심 화소 차이를 계산

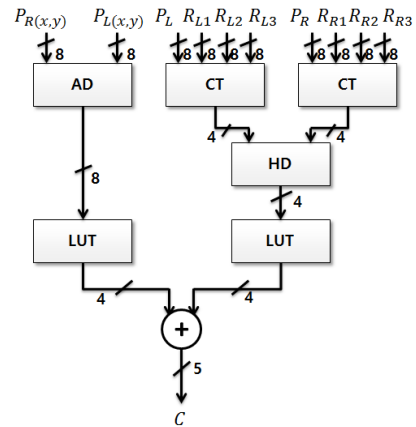


그림 3. 비용함수 모듈의 구조.  
Fig. 3. Structure of a cost module.

하는 AD 모듈, 중심 화소와 주변 화소의 관계에 따라 벡터를 계산하는 CT 모듈, 계산된 두 벡터의 해밍거리를 계산하여  $C_{Census}$  결정하는 HD 모듈, 식(4)로 정의된 지수함수를 근사화한 LUT(look-up table) 그리고 최종 비용을 계산하는 덧셈기로 이루어져 있다.

지수함수를 위한 LUT는 다양한 실험을 통해 에러를 분석한 후 표 1과 같이 효율적으로 근사화하였다. AD와 CT에 대한 비용함수 LUT는 값의 범위가 다르기 때문에 별도로 구성하였으며 모두 4비트의 출력을 갖는다. 센서스 벡터를 계산하는 CT에는 중심화소와 주변화소 값들이 입력으로 들어온다. 이 때, 주변화소의 수와 위치는 계산량과 정합률에 영향을 준다. 본 연구에서는 다양한 패턴의 주변화소의 경우를 모의적으로 실험하였고 간단하지만 에러분석에서 비교적 좋

표 1. 지수함수가 포함된  $\rho(c,\lambda)$ 함수의 근사값 구현.  
Table 1. Approximation of function.

$C_{AD(p,d)}$	0~	1~	2~	3~	4~
$\rho(c,\lambda)$	0	0.0951	0.1812	0.2591	0.3296
approx.	0	0.125	0.1875	0.25	0.3125
$C_{AD(p,d)}$	5~	6~	7~	8~	10~
$\rho(c,\lambda)$	0.3934	0.4511	0.5034	0.5506	0.6321
approx.	0.375	0.4375	0.5	0.5625	0.625
$C_{AD(p,d)}$	11~	13~	16~	19~	24~
$\rho(c,\lambda)$	0.6671	0.7274	0.7981	0.8504	0.9092
approx.	0.6875	0.75	0.8125	0.875	0.9375
$C_{Census(p,d)}$	0	1	2	3	4
$\rho(c,\lambda)$	0	0.8646	0.9816	0.9975	0.9996
approx.	0	0.875	0.9375	0.9375	0.9375

은 결과를 보인 4-point CT를 적용하였다.

그림 4는 CT에 필요한 화소 값들의 위치들과 CT의 구조를 보여주며 CMP는 비교기를 나타낸다. CT에 입력될 중심 화소  $P_{(x,y)}$ 와 4 개의 주변화소 중, 중심화소  $P_{(x,y)}$ 와 주변화소  $R_3$ 는 그림4(b)와 같이 2 클럭이 지나서 입력되어야 하는 중심화소  $P_{(x,y)}$ 와 주변화소  $R_4$ 가 되며, 입력 값과 비교하는 정보가 같아진다. 따라서 본 논문에서는  $C_3$  값을 재사용하여 그림 4(c)와 같이 구성하였다.

그림 5는 부정형 윈도우 영역을 정의하는 누적 윈도우 모듈의 세부 구조를 나타낸 것이다. 부정형 윈도우를 초기에 최대 윈도우의 크기인  $L_1$ (상, 하, 좌, 우, 한 방향에 대한 크기)만큼의 윈도우로 설정하여 열과 행 방향 각각  $2L_1$ 만큼 탐색한다. 조건(6)을 만족하는 위치의 값을 1, 만족하지 않은 위치의 값을 0으로 할당하여 비트 스트림 형태로 부정형 윈도우 정보를 보낼 수 있도록 설계하였다.

부정형 윈도우는 먼저 열 방향으로 비트 스트림을 계산하는 데, 열 방향으로는 조건(6)과 같이 비교를 해야 하므로 열 방향을 처리할 때에는 cb(center bit)를 항상 1로 내보낸다. 그렇게 계산된 열 방향의 비트 스트림은 레지스터에 저장되어 각 41개의 행 방향의 cb가 된다. 열 방향의 윈도우 연산이 끝나면 41번의 행 방향의 연산이 시작되는데, 만약 열 방향의 비트 인 cb가 0으로 할당되었다면 그 행은 열 방향으로 조

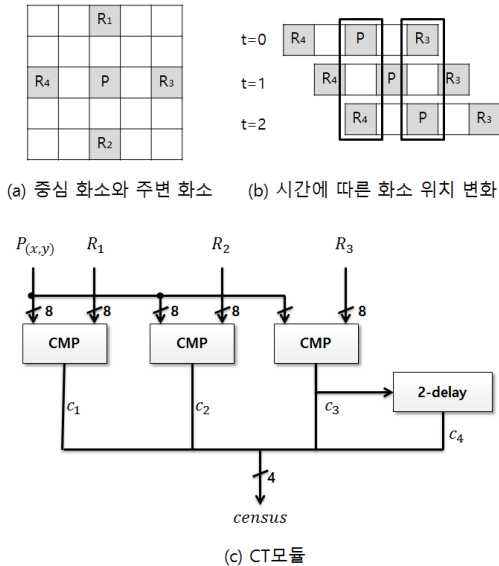


그림 4. CT에 필요한 화소들의 위치와 CT의 구조.  
Fig. 4. Pixel locations and the structure of CT.

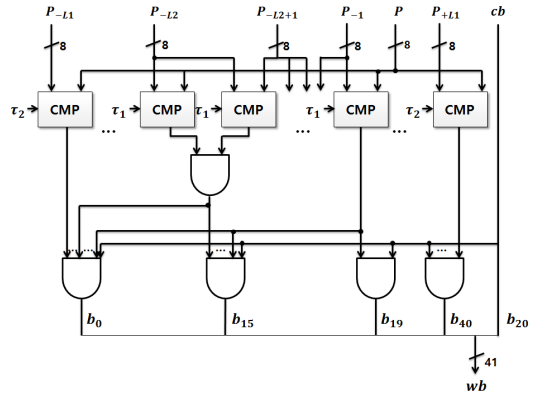


그림 5. 누적 윈도우 모듈의 구조.  
Fig. 5. Structure of aggregation window module.

건을 만족하지 못하였으므로 누적범위가 아니며 비트 스트림은 cb에 의해 모두 '0'으로 채워지게 된다.

중심 화소가 바뀔 때마다 누적 윈도우( $41 \times 41$ )만큼의 비용 데이터를 누적한다면 동시에 필요한 입력 데이터가 많아져 대역폭 문제가 생기게 되고 같은 위치에서의 비용 데이터를 다시 계산해야하기 때문에 불필요한 사이클 수가 증가한다. 그림 6은 비용 누적 모듈의 구조를 나타낸 것으로 대역폭 증가와 불필요한 사이클 증가문제를 해결하기 위해 재사용되는 비용 데이터를 저장하는 메모리를 포함한다. 비용 누적

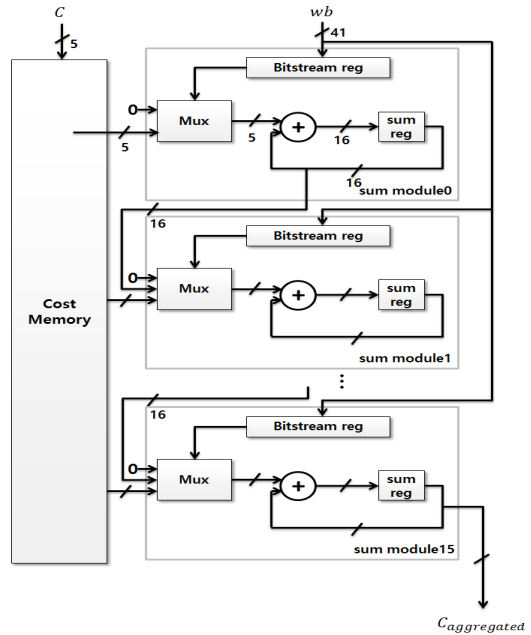


그림 6. 비용 누적 모듈의 구조.  
Fig. 6. Structure of a cost aggregation module.

모듈에는 실시간 시스템을 위해 41개의 행 단위 누적 모듈(sum module)을 병렬로 사용하였으며, 누적 윈도우 정보를 입력받아 정보에 따라 누적할 수 있도록 설계하였다.

비용 모듈에서 계산된 비용값은 각 행에 해당하는 메모리 위치에 저장되며 비트 스트림 비트에 따라 메모리에 저장된 해당 비용값을 사용하여 다중화기(MUX)를 통해 비용을 누적하도록 되어 있다. 비용 누적 모듈은 누적 윈도우 모듈에서 계산된 행 방향의 윈도우 비트 스트림이 0번째에서 40번째까지 행 단위 누적 모듈에 순차적으로 입력되어 계산을 시작한다. 하나의 행 방향으로 41번의 덧셈 연산이 끝나면 상위 행 모듈의 누적값을 받아 열 방향으로 덧셈 연산을 하도록 설계하였다.

### 3.2 스케줄링과 데이터 재사용

지역정합의 특성상 윈도우 내에서 비용연산 및 누적 시에 많은 중복연산을 포함하기 때문에 적절한 병렬처리와 데이터 재사용을 통하여 효율적인 처리와 성능 개선이 가능하다. 부정형의 누적 윈도우는 저장된 입력 영상의 데이터를 사용하여 정의되며, 이는 탐색거리와 상관없이 중심화소가 같으면 같은 형태의 부정형 윈도우를 갖게 된다. 이 방법으로 윈도우를 정의할 경우, 16개의 병렬처리를 사용하여 불필요한 계산을 줄일 수 있다.

그림 7은 비용 누적 모듈에서 데이터를 재사용하기 위한 과정으로 하나의 비용 누적 윈도우는 41개 행으로 구성되어 있으며, 하나의 행은 41개의 열로 되어있다. 현재 윈도우의 계산을 수행할 때 메모리에는 각 행에 해당하는 값이 저장되어 있다. 저장된 값 중, Row0~Row40의 p0~p40에 해당하는 값을 사용하여 현재 윈도우를 계산하게 된다. 이 후, 행 방향으로 윈도우가 움직이면서 다음 윈도우 누적을 하게 되며 한 행에 대한 비용 누적 계산을 마치게 된다. 이 때, 윈도

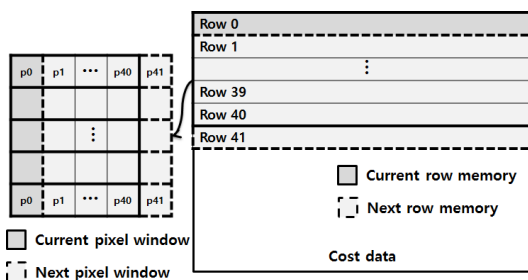


그림 7. 비용 누적 모듈의 데이터 재사용.  
Fig. 7. Reusing data of cost aggregation module.

우의 가장 위에 위치한 Row0는 더 이상 다음 연산에서 사용되지 않으므로 메모리에서 지워지고, 그 위치에 다음 윈도우 계산에 필요한 Row41을 계산하면서 새롭게 저장된다. 새롭게 계산한 Row41은 이전에 계산된 Row1~Row40과 함께 다음 행의 윈도우 연산을 수행하게 된다.

표 2는 데이터를 재사용하지 않을 때와 재사용했을 때의 필요한 사이클 수를 비교한 것이다. 행 단위의 데이터 재사용을 하지 않을 때에는 비용 누적에서 사용되는 행 단위의 병렬처리가 불가능하기 때문에 사이클이 크게 증가한다. 또, 같은 연산의 불필요한 반복으로 인해 효율이 떨어지게 된다. 본 논문에서 제안한 구조는 데이터 재사용을 사용하여 계산량을 약 3M까지 감소시킬 수 있다.

표 2. 데이터 재사용에 따른 계산량 분석.  
Table 2. Performance analysis due to data.

data reuse	window size	search range	image size	cycles
no	41x41	16	384x288	≈77M
yes	41x41	16	384x288	≈3M

## IV. 성능분석

제안한 스테레오 비전 시스템은 verilogHDL로 설계하여 검증하였고, Synopsys를 사용하여 합성하였다. 제안한 구조는 윈도우 크기 41×41, 탐색거리 16, 영상크기 384×288를 처리할 수 있으며, 동부하이텍 110nm 표준셀 라이브러리로 합성한 결과로 381K개의 게이트를 필요로 하며 데이터 재사용을 위해 140KB의 저장공간을 필요로 한다. 최대 클럭 속도는 192MHz로 약 59fps의 속도를 갖는다.

부정형 윈도우 정의에 필요한 파라미터  $L_1, L_2, \tau_1, \tau_2$ 는 정합 에러율에 큰 영향을 미친다.  $L_1, L_2, \tau_1, \tau_2$ 가 허용하는 범위가 너무 좁다면 텍스처가 없는 영역에 대해 상관성을 구분하기 어려우며, 허용하는 범위가 너무 넓다면 모호한 부분에서 많은 에러를 발생시킬 수 있다. 본 논문에서는 최적의 파라미터를 구하기 위해 실험적으로 분석하여  $\lambda_{Census}, \lambda_{AD}, L_1, L_2, \tau_1, \tau_2$ 는 각각 0.25, 10, 20, 5, 20, 6을 사용하였다.

표 3은 여러 스테레오 영상을 입력 영상으로 사용하였을 때, 제안한 시스템의 결과 영상, 에러 영상, 그리고 스테레오 시스템들의 에러율을 나타내는 표이다.

표 3. 깊이정보 처리결과.  
Table 3. Disparity results.




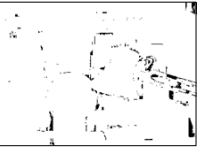

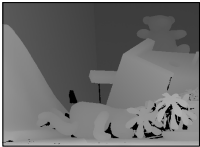
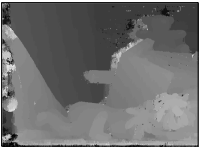
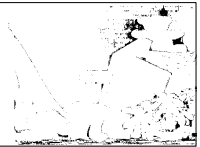

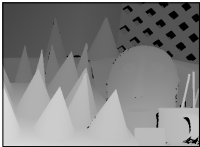
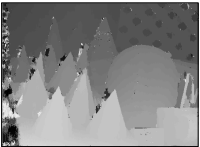
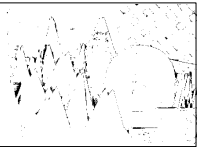
Stereo-pair	Ground truth	Disparity	Bad pixels	Comparison
 <Tsukuba>				Census[6] : 9.79 MCT/SAD[7] : 11.85 ADSW[8] : 4.48 HBP[10] : 1.49 proposed : 2.47
 <Teddy>				Census[6] : 12.5 MCT/SAD[7] : 11.35 ADSW[8] : 21.5 HBP[10] : 8.72 proposed : 4.70
 <Cones>				Census[6] : 7.34 MCT/SAD[7] : 6.97 ADSW[8] : 17.1 HBP[10] : 4.61 proposed : 3.16

표 4. 스테레오 비전 시스템의 성능분석.  
Table 4. Performance analysis of stereo.

algorithm	implementation	imgse size	search range	perf(fps)	throughput(pixel/s)	error(%)
Census[6]	FPGA	640×480	64	230	71M	9.79
MCT/SAD[7]	FPGA/ASIC	750×400	60	60	18M	11.85
ADSW[8]	FPGA	640×480	64	30	9M	4.48
RealDP[9]	AMD AthlonXP 2800	384×288	100	19	2M	2.85
HBP[10]	Nvidia Geforce 7900	320×240	16	16	1M	1.49
proposed	ASIC	384×288	16	59	5M	2.47

본 연구에서는 병렬처리, 파이프라인 스케줄링, 데이터 재사용, 함수 근사화 등의 다양한 설계 기법을 적용하여 기존 알고리즘의 복잡도를 개선하여 실시간처리가 가능하도록 설계하였으며 다른 스테레오 알고리즘과 비교하였을 때 비교적 낮은 에러율을 나타내었다.

표 4는 츠쿠바(tsukuba) 영상을 기준으로 제안한 스테레오 비전 시스템과 다른 실시간 스테레오 비전 시스템을 비교한 것이다. 선행연구 [6], [7], [8]은 비교적 빠른 속도를 보였지만, Census<sup>[6]</sup>과 기존 방식을 보완하기 위해 변화도 방식과 SAD 방식을 적절히 적용한 MCT/SAD<sup>[7]</sup>은 상대적으로 에러율이 매우 높았다. 또한 영역기반 알고리즘의 단점을 보완하기 위한 가중치 알고리즘을 적용한 ADSW<sup>[8]</sup>은 다른 지역 정합 알고리즘에 비해 처리율은 낮았으나 에러율이 상대적으로 감소하였다. 전역 정합법을 사용한 방법<sup>[9,10]</sup>

은 지역 정합법에 비해 에러율이 낮은 것을 알 수 있다. 본 논문에서 제안한 구조에서 처리속도는 59fps로 실시간 처리가 가능하며 에러율은 2.47%로 높은 정합률을 나타내었다.

### V. 결론

본 논문에서 제안한 실시간 스테레오 비전 시스템은 AD-Census 비용함수와 부정형 가변 윈도우를 사용하여 다른 영역 기반 알고리즘에 비해 우수한 정합률을 보여준다. 또한 데이터 재사용과 병렬처리 및 파이프라인 스케줄링, 비선형 함수의 효율적인 근사화 기법을 적절히 사용하여 실시간 처리가 가능하였다. 제안한 구조는 윈도우 크기 41×41, 탐색거리 16, 영상크기 384×288를 처리할 수 있으며 동부하이텍

110nm 표준셀 라이브러리를 사용하여 합성한 결과로 381K개의 게이트가 필요하며 최대 클럭 속도는 192MHz(약 59fps)의 속도를 갖는다. 에러율은 츠쿠바 영상을 기준으로 2.47%로 높은 정합률을 나타내었다.

### 감사의 글

저자들은 본 연구를 위하여 설계 소프트웨어를 제공하여 준 IDEC(IC Design Education Center)에 감사드립니다.

### References

[1] M. Bai, Y. Zhuang, and W. Wang, "Stereo vision based obstacle detection approach for mobile robot navigation," *IEEE Int. Conf. Intell. Control and Inf. Process.*, pp. 328-333, 2010.

[2] K. Huh, J. Park, J. Hwang, and D. Hong, "A stereo vision based obstacle detection system in vehicles," *Optics and Lasers in Eng.*, vol. 46, no. 2, pp. 168-178, 2008.

[3] Z. Wang and Z. Zheng, "A region based stereo matching algorithm using cooperative optimization," *IEEE Int. Conf. Computer Vision and Pattern Recognition*, pp. 1-8, 2008.

[4] Z. Gu, X. Su, Y. Liu, and Q. Zhang, "Local stereo matching with adaptive support-weight, rank transform, and disparity calibration," *Pattern Recognition Lett.*, vol. 29, no. 9, pp. 1230-1235, 2008.

[5] D. Yang, L. Chu, C. Chen, J. Wang, and M. Shieh, "Depth reliability based stereo matching algorithm and its VLSI architecture design," *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 25, no. 6, pp. 1038-1050, 2015.

[6] S. Jin, J. Cho, and X. Pham, "FPGA design and implementation of a real-time stereo vision system," *IEEE Trans. Cir. and Syst. for Video technol.*, vol. 20, no. 1, pp. 15-26, 2010.

[7] K. Ambrosch and W. Kubinger, "Accurate hardware-based stereo vision," *Computer Vision and Image Understanding*, vol. 114, no. 11, pp. 1303-1316, 2010.

[8] C. Ttofis and T. Theocharides, "Towards

accurate hardware stereo correspondence: A real-time FPGA implementation of a segmentation-based adaptive support weight algorithm," in *Proc. Conf. Design*, pp. 703-708, 2012.

[9] S. Forstmann, Y. Kanou, and J. Ohya, "Real-time stereo by using dynamic programming," *IEEE Int. Conf. Computer Vision and Pattern Recognition*, pp. 29-29, 2004.

[10] Q. Yang, L. Wang, and R. Yang, "Real-time global stereo matching using hierarchical belief propagation," *BMVC*, pp. 989-998, 2006.

[11] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On building an accurate stereo matching system on graphics hardware," *IEEE Int. Conf. Comp. Vision Workshops*, pp. 467-474, 2011.

### 조 은 별 (Eunbyeol Cho)



2017년 2월 : 가톨릭대학교 정보통신전자공학부 학사  
<관심분야> VLSI 설계, 영상 처리

### 박 태 근 (Taegun Park)



1985년 : 연세대학교 전자공학사  
1988년 : Syracuse Univ. 컴퓨터공학 석사  
1993년 : Syracuse Univ. 컴퓨터공학 박사  
1991년~1993년 : CRI VLSI 설계 엔지니어

1994년~1998년 : 현대전자 책임연구원  
1998년~현재 : 가톨릭대학교 정보통신전자공학부 교수  
<관심분야> VLSI 설계, CAD, 컴퓨터 구조