

윈도우 분할 기반 양방향 필터의 하드웨어 설계

현 용 호*, 박 태 근^o

Hardware Design of Bilateral Filter Based on Window Division

Yongho Hyun*, Taegeun Park^o

요 약

양방향 필터(bilateral filter)는 필터링 시 주변 화소의 평균을 계산하여 경계 보존과 잡음제거에 장점을 가진다. 본 논문에서는 윈도우 분할 기반 양방향 필터에 대하여 실시간 처리가 가능한 시스템을 설계하였다. 윈도우 내부의 주변 화소를 5분할하고 연속된 중심화소와 공유하는 주변 화소를 동시에 연산하는 파이프라인 스케줄링을 적용한 병렬 처리 기법으로 성능을 개선하였다. 비트 폭에 따른 필터 성능과 하드웨어 자원 소모에 대한 상충관계(tradeoff)를 고려하였으며, 필터링 결과 영상의 PSNR 분석을 통하여 비트를 할당하였고 사용된 지수함수는 16단계의 계단함수 LUT를 적용하였다. 설계한 시스템은 verilogHDL로 설계되었으며, 동부하이텍 110nm 라이브러리를 사용하여 Synopsys를 통해 합성하였고 416MHz의 최대 동작주파수에서 416Mpixels/s(397fps)의 처리량(throughput)과 132K 게이트의 하드웨어 자원을 사용한다.

Key Words : bilateral filter, edge preserving, VLSI architecture, pipelining, realtime system

ABSTRACT

The bilateral filter can reduce the noise while preserving details computing the filtering output at each pixels as the average of neighboring pixels. In this paper, we propose a real-time system based on window division. Overall performance is increased due to the parallel architectures which computes five rows in the kernel window simultaneously but with pipelined scheduling. We consider the tradeoff between the filter performance and the hardware cost and the bit allocation has been determined by PSNR analysis. The proposed architecture is designed with verilogHDL and synthesized using Dongbu Hitek 110nm standard cell library. The proposed architecture shows 416Mpixels/s (397fps) of throughput at 416MHz of operating frequency with 132K gates.

I. 서 론

이미지 필터링은 영상 처리와 컴퓨터 비전 분야에서 자주 사용되는 처리 과정이며 기본 단위인 필터링 윈도우가 있는 것이 특징이다. 그 중에서 양방향 필터(bilateral filter)^[1]는 스테레오 정합(stereo matching)^[2],

잡음 제거(denoising)^[3], 고생동폭 영상처리(high-dynamic range imaging)^[4]등의 분야에 활용되고 있다.

양방향 필터는 두 가지 요소로 구성된다. 하나는 영역 필터(range filter)이고 다른 하나는 공간 필터(spatial filter)이다. 양방향 필터의 중요한 특성인 경계 보존은 비선형 특성을 가진 영역 필터로부터 기인

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

※ 본 연구는 2016년도 가톨릭대학교 교비연구비의 지원으로 이루어졌음(M-2016-B0002-00100).

• First Author: Department of Information, Communication, and Electronic Engineering, The Catholic University of Korea, gusdydgh2@naver.com, 학생회원

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

논문번호 : KICS2016-09-247, Received September 5, 2016; Revised December 5, 2016; Accepted December 13, 2016

한다. 반면에 공간 필터는 비슷한 화소값을 가지는 화소들을 선택하여 평균을 계산하는 선형 특성을 가지고 있으며 지역 통과 필터이다. 이러한 선형 특성은 영상의 잡음 제거, 흐림 효과(blurring)를 발생시킨다. 앞서 설명한 필터의 경계 보존과 흐림 효과의 정도를 결정하기 위하여 각 필터마다 파라미터 값이 존재하는데, 이 파라미터들의 변화에 따라 경계 보존의 정도와 흐림 효과의 정도가 달라진다.

양방향 필터는 계산복잡도가 필터 윈도우의 크기에 영향을 받기 때문에 윈도우 크기가 커질수록 계산시간도 늘어난다⁵⁾. 이러한 단점을 보완하기 위해 윈도우의 크기가 계산 복잡도에 주는 영향을 줄이는 동시에 필터의 성능을 유지하는 고속 알고리즘들이 제안되었다. 대부분의 고속 알고리즘은 여러 개의 공간 필터를 활용하여 양방향 필터링을 수행하는 것이다. 각 공간 필터는 FFT(Fast Fourier Transform), 확장된 적분 영상, 혹은 순환 필터(recursive filter) 등을 활용하였다. 임의의 공간 윈도우와 다항식 공간 윈도우를 히스토그램에 적용한 Porikli 등의 $O(1)$ (constant time) 알고리즘⁶⁾, 임의의 공간 및 영역 윈도우를 분해한 뒤 여러 개의 공간 필터를 사용하여 양자화와 보간을 하는 Durand와 Dorsey의 알고리즘⁴⁾, 푸리에 분석을 사용하여 가우시안 영역 윈도우를 추정하고 주기를 최적화한 Sugimoto 등의 압축 알고리즘⁷⁾ 등이 제안되었다.

또한 실시간 영상 처리를 수행하기 위한 다양한 하드웨어 구현에 대한 연구가 진행되어 왔다. Tseng 등은 히스토그램을 기반으로 한 설계 기술을 적용하였다⁸⁾. Han 등은 병렬구조를 적용한 양방향 필터를 스테레오 비전 프로세서에 활용하였다⁹⁾. 또한 양방향 필터를 활용한 동적 영역 압축(dynamic range compression)으로 비디오 품질을 개선한 시스템¹⁰⁾, 필터 블록 내부의 화소값 차이에 따라 적응적 계수 마스크를 사용하여 경계 보존 필터를 적용한 뒤 병렬 파이프라인을 적용한 시스템¹¹⁾, MCU(Master Control Unit), 커널 기반의 클럭 장치와 재구성 서버(reconfigurable server) 등으로 구성되어 있는 저전력 고성능의 양방향 필터 시스템¹²⁾ 등이 제안되었다.

양방향 필터는 구조가 단순하며, 다른 이미지 필터에 비하여 경계 보존과 잡음 제거를 효과적으로 할 수 있다. 그러나 순차적인 연산을 진행할 경우 인접한 중심화소간의 커널 윈도우 내부에 겹치는 주변 화소가 발생하고, 이에 따른 입력 데이터의 중복에 의하여 계산복잡도가 증가한다. 본 논문에서는 가중치 연산 후 화소값의 효율적인 저장과 관리를 위하여 lifetime

analysis 방법¹³⁾에 의한 스케줄링을 적용하였다. 또한 양방향 필터에 효과적인 병렬, 파이프라인 구조를 적용한 실시간 하드웨어 시스템을 구현하여 성능을 개선했다. 제안된 구조는 상당한 내부 메모리를 필요로 하는 Tseng⁸⁾과 Han⁹⁾ 등의 구조와 달리 내부 메모리가 필요 없으며 중심 픽셀과 인접한 픽셀 간의 필터링 연산을 담당하는 서브 필터들의 간결한 파이프라인 구조와 효율적인 스케줄링으로 인하여 우수한 성능을 보인다. 알고리즘에 사용되는 지수연산은 PSNR 에러 분석을 통하여 계단 함수형태로 근사화하여 LUT(look-up table)로 설계하였다.

본 논문의 구성은 다음과 같다. II장에서는 기본적인 가우시안 필터 및 본 논문에 적용한 양방향 필터를 설명하였다. III장에서 제안한 하드웨어 구조와 스케줄링에 대해서 설명하고 IV장에서 설계 결과를 다른 선행연구 결과와 비교 분석하였으며 V장에서 결론을 맺는다.

II. 양방향 필터

영상 필터링은 해당 위치의 중심 화소와 필터링 윈도우 내부에 있는 주변부 화소값과의 함수로 계산된다. 특히 가우시안 필터는 중심화소로부터 멀어질수록 감소하는 가중치를 적용하여 주변 화소와 중심 화소간의 가중 평균을 계산한다. 영상은 보통 천천히 변화하기 때문에 서로 가까이 있는 화소들은 비슷한 값들을 가지고 있다는 가정을 할 때 잡음들은 주변 화소들과의 평균을 계산하여 제거될 수 있다. 영상이 영역을 통과하면서 천천히 변화한다는 가정은 경계에서 어긋나게 되며 경계부분은 지역 통과 필터링에 의해 흐려지게(blurring) 되며, 이는 화질 열화의 원인이다¹⁾. 이와 같은 가우시안 필터의 단점을 보완하여 경계 부근에서 의도치 않은 흐림 효과를 방지하고 경계를 보존하면서 가중 평균값을 계산하여 영상의 잡음을 제거하는 필터가 양방향 필터이다.

2.1 가우시안 필터

가우시안 필터는 필터링할 화소에 대하여 그 화소를 중심으로 주변 화소에 대하여 각각의 가중치를 곱하고 누적하여 평균을 구하며 수식은 다음과 같다.

$$C'_i = \frac{1}{\pi\sigma^2} \sum_{j \in \Omega} \exp\left(-\frac{\|i-j\|}{\sigma}\right)^2 \cdot C_j \quad (1)$$

이 때, i 는 필터링할 중심화소의 좌표를, j 는 필터

링 범위로 설정된 윈도우 Ω 내의 주변 화소의 좌표를 의미한다. C_j 는 좌표 j 에 대한 화소값을, C_i' 은 좌표 i 에 대한 필터링 화소값을 의미하며 σ 는 파라미터이다.

가우시안 필터는 영상의 잡음을 제거하여 전체적인 화소값의 평활화에는 큰 효과를 보인다. 그러나 화소값이 크게 변화하는 경계 부분에서도 흐림 효과가 발생한다. 때문에 가우시안 필터링은 잡음 제거는 가능하지만 전체 영상의 품질이 떨어지는 단점을 가지고 있다.

2.2 양방향 필터

가우시안 필터의 경계 부분에서의 영상 품질 저하 현상을 보완하기 위해서는 경계 부근의 화소값이 필터링 계산에 큰 영향을 주지 않도록 해야 한다. 그래서 비슷한 화소값을 갖는 윈도우 주변 화소들의 평균을 구하는 방법을 적용하여 경계 보존이 가능한 필터가 제안되었는데 그것이 양방향 필터이다.

양방향 필터는 범위 내에 존재하는 이웃 화소들에 대하여 화소 단위로 그 유사 정도에 따라 가중치를 할당하고, 가중치 평균을 계산하여 잡음을 제거하며 양방향 필터의 수식은 다음과 같다.

$$C_i' = \frac{1}{N_i} \sum_{j \in \Omega} R_{i,j} \cdot S_{i,j} \cdot C_j \quad (2)$$

식(2)는 가우시안 필터와 유사하다. 가우시안 필터는 화소 사이의 공간적 거리 차이를 가중치로 사용하는 공간 필터이다. 그리고 화소값의 차이를 가중치로 사용하는 영역 필터를 공간 필터와 합성한 것이 양방향 필터이다.

$$S_{i,j} = \exp\left(-\frac{\|i-j\|}{\sigma_s}\right)^2 \quad (3)$$

식(3)은 공간 유사도를 계산하는 방법이며 유클리드 거리 계산법을 사용한다. σ_s 은 저역 통과 필터링의 양을 조절하는 파라미터이며, 이 값이 커질수록 평활화의 정도가 세지며 영상은 더 흐릿해진다.

$$R_{i,j} = \exp\left(-\frac{\|C_i - C_j\|}{\sigma_r}\right)^2 \quad (4)$$

식(4)는 화소 유사도를 계산하는 방법이며 유클리드 거리 계산법을 사용한다. σ_r 은 화소값의 범위를 정하는 파라미터 값이며 화소값의 차이가 σ_r 을 초과하게

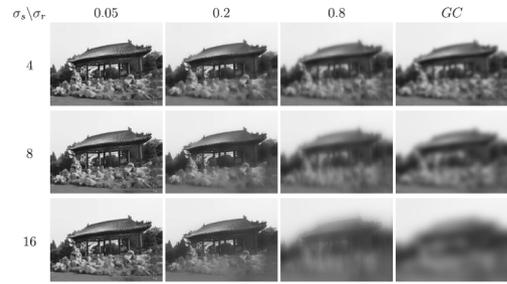


그림 1. 양방향 및 가우시안 필터링
Fig. 1. Bilateral and gaussian filtering

되면 필터링 결과에 큰 영향을 미치지 못하게 된다.

$$N_{i,j} = \sum_{j \in \Omega} S_{i,j} \cdot R_{i,j} \quad (5)$$

$N_{i,j}$ 는 필터링 결과가 범위를 벗어나지 않도록 조절해주는 정규화 상수이다. 양방향 필터링에서는 중심 화소와 화소값이 비슷한 주변 화소가 더 큰 가중치값을 갖게 되고, 이 가중치가 곱해진 평균값이 계산된다. 따라서 밝은 화소들은 밝은 화소끼리, 어두운 화소들은 어두운 화소끼리 평활화를 진행하여 경계가 잘 보존된다.

그림1에서 좌측부터 3개의 열은 양방향 필터링의 경우이고 최우측 열은 가우시안 필터링의 경우이다. 그림에서 보는 바와 같이 일정한 가중치값을 갖는 가우시안 필터에서 보다 화소 유사도를 측정하여 가중치에 반영하는 양방향 필터의 경우에서 경계 보존이 비교적 잘 되고 있음을 알 수 있다.

III. 제안된 스케줄링 및 하드웨어 구조

3.1 데이터 스케줄링

본 논문에서는 5×5 의 필터링 윈도우를 사용하며 제안된 양방향 필터는 하나의 중심화소에 대하여 자신을 포함한 25개의 주변 화소값과 필터링 연산을 한다. 그림2(b)의 5번째 단계에서 중심화소를 기준으로 필터링할 때 주변 화소값의 데이터 중복은 빗금친 20개이다. 행방향의 연속된 5개의 중심화소에 대한 필터링 시 중복되는 열의 결과값을 재사용할 수 있도록 파이프라인 스케줄링을 적용하였으며, 동시에 병렬처리 구조를 이용하여 성능을 개선하였다.

그림2(a)와 같이 한 클럭 당 하나의 중심 화소값과 윈도우 좌측 5개의 주변 화소값이 입력으로 들어가게 된다. 5개의 주변 화소값은 저장되어 있던 이전 4개의

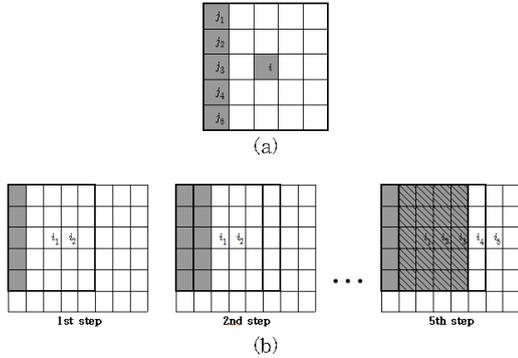


그림 2. (a) 커널 윈도우, (b) 데이터 스케줄링 과정
Fig. 2. (a) Kernel window, (b) Data scheduling process

중심 화소값과 서브 필터링 계산을 한다. 그림2(b)는 중심화소 i 에 대한 필터링 과정을 나타낸다. 윈도우 내부에 있는 주변 화소 25개를 세로로 5분할하여 한번에 5개씩 가중치와 가중치곱 화소값을 계산하여 누적한다. 이 과정을 5번 반복하면 하나의 중심화소에 대하여 주변 화소 25개의 가중치 누적합과 가중치곱 화소값의 누적합을 얻으며 나눗셈을 통하여 가중치 평균값이 결과값으로 출력된다.

앞서 설명한 계산과정이 행으로 연속된 5개의 중심 화소에서 동시에 이루어진다. 1번 화소가 5번의 서브 필터링 연산을 하는 동안 옆에 위치한 2번 화소는 그림2(b) 5번째 단계의 빗금 친 20개의 화소들에 대한 필터링 연산을 진행한다. 이는 4번의 서브 필터링 연산을 한 것이다. 이어지는 3, 4, 5번 위치의 중심화소는 동일한 서브 필터링연산으로 15, 10, 5개의 주변 화소값의 가중치 누적합과 가중치곱 화소값의 중간결과가 저장된다.

3.2 제안하는 하드웨어 구조

그림 3은 제안하는 양방향 필터 구조를 나타낸다. 중심화소와 5개의 세로열 화소를 필터링하는 서브 필터(SF) 어레이 5개와 가중 평균값을 계산하는 나눗셈기로 구성된다. 한 번의 서브 필터링 단계를 완료한 뒤 가중치와 가중치곱 화소값의 중간 누적 결과를 저장하는 가중치합(weight sum), 픽셀합(pixel sum) 레지스터가 있다. 첫 단계에서 5개, 두 번째에서 10개, 세 번째에서 15개, 네 번째에서 20개, 다섯 번째 단계에서 25개의 주변 화소의 가중치곱 화소값과 가중치 누적합이 저장한다. 파이프라인 구조로 이루어진 5번의 SF 계산 후 가중치 화소곱을 가중치 누적합으로 나누는 나눗셈기가 있다. 나눗셈기는 비복원 나눗셈(nonrestoring division) 알고리즘을 적용하였다^[14].

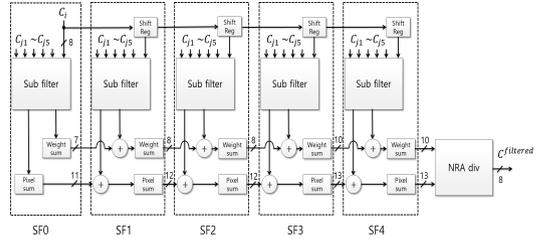


그림 3. 제안하는 양방향 필터 구조
Fig. 3. Structure of the proposed bilateral filter

가중치곱 화소값의 누적합과 가중치 누적합의 비트 폭은 알고리즘 차원에서 오버플로우를 고려하여 할당하였다. 그 결과 가중치곱 화소값은 13비트, 가중치의 누적합은 9비트(정수 5비트, 소수 4비트)로 설정하였다. 13비트의 피제수와 부호비트 포함 10비트의 제수를 계산하는 나눗셈기는 임계경로(critical path)를 고려하여 3-스테이지 파이프라인 구조로 설계하였다. 따라서 하나의 중심화소를 필터링한 최종 결과값은 8 클럭 뒤에 나오게 된다.

그림 4(a)는 서브필터 구조를 나타낸다. 하나의 중심화소와 하나의 주변 화소를 이용하여 계산하는 PE(processing element) 5개로 구성된다. 그림 4(b)는 PE 내부의 구조이다. 화소 및 공간 유사도를 계산하는 유사도(similarity) 모듈, 지수함수 LUT(look-up table) 모듈과 가중치와 주변 화소값의 곱셈을 위한 곱셈기로 구성된다.

유사도 모듈에는 하나의 중심화소와 주변 화소가 입력된다. 유클리드 거리에 의해 화소 및 공간 유사도가 계산되며, 결과는 지수함수 LUT 모듈에 입력된다. 지수함수 LUT에 의하여 주변화소의 가중치값이 출력된다. 그리고 가중치와 주변 화소값의 곱셈 결과가 가중치곱 화소값이다. 곱셈기는 주변 화소값(8비트 정수)과 가중치값(정수 1비트, 소수 4비트)을 입력으로

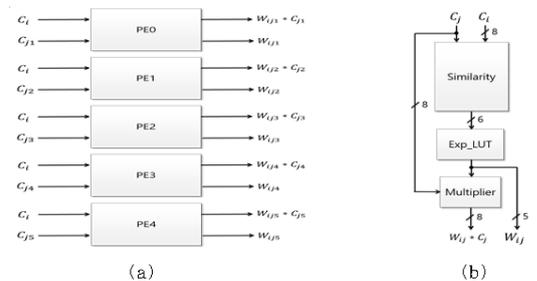


그림 4. (a) 서브필터의 구조, (b) PE의 구조
Fig. 4. (a) Structure of sub-filter module, (b) Structure of processing element

받는다. 출력은 필터링 결과에 큰 영향을 주지 않는 소수부를 잘라낸 8비트 정수값을 갖는다.

그림 5(a)는 유사도 모듈의 구조이다. 중심화소와 주변 화소값을 입력으로 받으며, 계산결과를 보정하기 위한 파라미터 σ_s, σ_r 값이 필요하다. 공간 유사도는 식(3)의 유클리드 거리 계산법을 이용하여 위치 차이를 계산한다. 이를 σ_s 로 나뉘어야 하며 나눗셈은 상수 σ_s 값이 제수이므로 쉬프트를 이용하여 단순화하였다. 공간 유사도는 σ_s 값에 의해 1보다 작은 소수값을 가지며 소수 비트 할당에 따른 결과 영상의 PSNR 분석을 통하여 4비트로 제한하였다. 소수 4비트일 때 공간 유사도값을 근사화할 때, $0, \frac{1}{16}, \frac{1}{8}$ 3가지 값을 가진다. 이를 바탕으로 공간 유사도값에 따른 3가지의 유사도 모듈을 설계하였다. 최종적인 공간 유사도의 비트 할당은 정수 1비트, 소수 4비트이다.

화소 유사도 계산을 위해 식(4)를 적용하였다. 중심 화소값과 주변 화소값 차이의 절대값을 상수 σ_r 값으로 나눈 결과값을 제공하여 계산한다. 화소 유사도와 공간 유사도의 합은 6비트(정수 2비트, 소수 4비트)로 할당되어, 이 결과는 그림5(b)에서 지수함수 모듈의 입력이다.

그림 5(b)는 시스템에서 5x5 윈도우 내부의 공간 유사도값이다. 중심화소에서 가까운 거리에 있는 주변 화소와의 공간 유사도는 소수 4비트의 최소 표현단위인 0.0625를 넘지 못한다. 그래서 윈도우 내부 9개의 공간 유사도 값은 모두 0으로 근사화하였다. 중심화소에서 상대적으로 먼 거리에 있는 윈도우 외부의 주변 화소와 공간 유사도 값은 각각 거리가 2, $\sqrt{5}, 2\sqrt{2}$ 인 경우로 나누어진다. 모서리는 0.125, 나머지 2, $\sqrt{5}$ 의 경우 0.0625로 근사화하였다. 공간 유사도 값

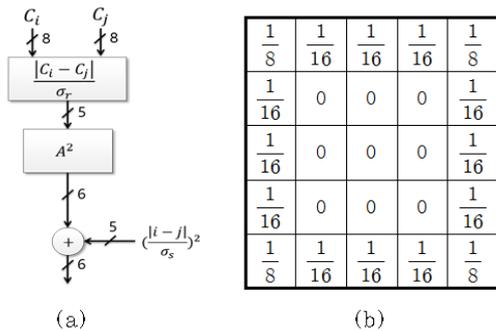


그림 5. (a) 유사도 모듈의 구조, (b) 공간 유사도의 설정값 Fig. 5. (a) Structure of similarity module, (b) Spatial similarity value

이 커질수록 주변 화소값의 가중치는 감소한다. 주변 화소와 중심화소의 거리가 멀어질수록 중심화소의 필터링에 영향을 주지 않는다는 것을 의미한다.

지수함수 모듈은 가중치 계산을 위한 모듈로 입력 값이 클수록 출력값이 작아진다. 본 연구에서는 LUT를 이용하여 지수함수 모듈을 설계하였으며 표1과 같이 16개의 영역으로 구현하였다. 출력인 가중치값의 정밀도가 필터링 성능을 좌우하기 때문에 지수함수 결과의 변화폭이 큰 구간인 1영역에서 11영역까지는 입력 변화폭을 0.0625로 하여 정밀도를 높였다. 결과의 변화폭이 작은 11영역~16영역까지 입력의 변화폭을 0.125로 설정하였으며 변화폭이 작기 때문에 입력 값에 따라 민감하게 변화하지 않았다.

표 1. 지수 함수의 근사적 구현 Table 1. Approximation of exp(x)

구분	1영역	2영역	3영역	4영역	5영역	6영역
x	0	0.0625	0.125	0.1875	0.25	0.3125
exp(x)	1	0.9375	0.8750	0.8125	0.75	0.6875
구분	7영역	8영역	9영역	10영역	11영역	12영역
x	0.375	0.4375	0.5	0.5625	0.625	0.75
exp(x)	0.625	0.625	0.5625	0.5625	0.5	0.4375
구분	13영역	14영역	15영역	16영역		
x	0.875	1.0	1.125	1.25		
exp(x)	0.375	0.3125	0.3125	0.0		

IV. 설계 및 성능 분석

양방향 필터에서는 공간 유사도의 파라미터 σ_s 값과 화소 유사도 파라미터 σ_r 값의 설정이 필요하다. 본 연구에서는 하드웨어 설계시 효율적인 비트연산을 고려하였으며 σ_s 값은 8, σ_r 값은 128로 설정하였다. 하드웨어 설계시 데이터 정수부의 오버플로우가 발생하지 않도록 정수 비트폭을 할당하였으며 소수부 비트폭은 PSNR 분석을 통하여 결정하였다. 표2는 지수함수 모듈의 소수 비트 할당에 의한 PSNR 값이다. 비트 할당이 적용된 필터링 결과영상과 알고리즘의 결과영상 간의 PSNR을 비교하였으며 비용과 화질의 상충관계(trade-off)를 고려하여 비트폭을 결정하였다. PSNR 47 이하로 내려가지 않는 것을 기준으로 하여 소수부 비트폭은 4비트로 할당되었다.

표 2. 지수 함수 모듈의 소수 비트 할당 Table 2. Bit allocation for exp(x) module

Fractional part	6bit	5bit	4bit	3bit
PSNR	48.437	48.001	47.635	46.980

표 3. 양방향 필터 성능 분석
Table 3. Performance analysis of bilateral filter

구분	Tseng[8]	Han[9]	Gabiger[10]	Vinh[11]	Sakthivel[12]	Proposed
Technology	UMC 90nm	TSMC 180nm	Xilinx Vertex5	Altera Cyclone II	Xilinx Vertex5	Dongbu 110nm
Image size	1920×1080		640×480		1024×1024	
Kernel window	31×31	11×11	5×5	3×3	3×3 ~ 15×15	5×5
Clock speed (MHz)	200	60	220	159	245	416
Frame rate (fps)	60	144	52.45	151.63	78.42	397
Throughput (Mpixels/s)	124	44	54	158	81	416
Gate count (K gates)	356	355	-	-	-	132
Memory (KB)	23	7.8	-	-	-	0
NHE (K)	0.28	0.20	-	-	-	3.15

제안한 양방향 필터는 verilogHDL로 설계하였고 Modelsim을 이용하여 검증하였다. 제안한 구조를 동부하이텍 110nm 표준셀 라이브러리를 이용하여 Synopsys로 합성한 결과 전체 시스템은 132K 게이트를 필요로 하고 최대 클럭 속도는 416MHz이다. 표3은 제안한 양방향 필터를 다른 선행연구 결과와 비교를 통하여 성능을 분석했다. 양방향 필터의 FPGA 구현 방법을 채택한 알고리즘들은 필터 윈도우의 크기가 작은 경우가 더 좋은 성능을 가지고 있으며 윈도우의 크기가 커질수록 성능은 저하된다. ASIC으로 구현한 Han 등의 구조^[9]는 비교적 큰 필터 윈도우를 사용하지만 낮은 처리량을 보인다. 특히 ASIC으로 구현한 Tseng 등의 구조^[8]는 큰 크기의 필터 윈도우를 가지면서도 적절한 처리량을 보여준다. 본 논문에서 제안한 시스템은 다른 연구결과와 비교할 때 우수한 성능을 보여준다. Tseng^[8]과 Han^[9]이 제안한 구조에서는 내부에 수행 시 필요한 정보를 저장하기 위한 메모리가 23KB와 7.8KB가 필요하지만 제안된 구조에서는 내부 메모리가 필요 없다. 또한 필요한 게이트 수도 두 경우보다 적어서 우수하다고 할 수 있다. 좀 더 종합적인 비교를 위하여 면적-속도의 상충관계를 고려한 NHE(Normalized Hardware Efficiency)를 이용하였으며 정의는 식(6)과 같으며 여기에서 110nm는 본 논문에서 사용한 공정기술이다.

$$NHE = \frac{(throughput \times ref - process / 110nm)}{gate\ count} \quad (6)$$

제안한 구조의 NHE는 3.15K를 나타내며 합성결과 132K 게이트의 하드웨어 자원을 요구한다.

V. 결론

본 논문에서는 연속되는 중심화소가 공유하는 주변 화소를 동시에 처리하기 위한 병렬처리와 파이프라인 구조를 적용한 양방향 필터 구조를 제안하였다. 효율적인 설계를 위해 결과 영상의 PSNR 분석을 통하여 소수 비트폭을 할당하였으며, 지수함수는 16단계의 계단함수 LUT를 사용하였다. 제안한 시스템은 동부하이텍 110nm 표준셀 라이브러리를 사용하여 합성한 뒤 계산속도 및 하드웨어 자원 소모량을 분석하였으며 397 fps의 속도로 초당 416M 픽셀의 처리량(throughput)을 보이며 합성결과 132K 게이트의 하드웨어 자원을 요구한다.

감사의 글

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

References

- [1] C. Tomasi and P. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE ICCV*, pp. 839-846, 1998.
- [2] Q. Yang, "Hardware-efficient bilateral filtering for stereo matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 1026-1032, 2014.
- [3] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, and K. Toyama, "Digital

photography with flash and no-flash image pairs,” in *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664-672, 2004.

- [4] F. Durand and J. Dorsey, “Fast bilateral filtering for the display of high dynamic range images,” in *ACM Trans. Graph.*, vol. 21, no. 3, pp. 257-266, 2002.
- [5] L. Dai, M. Yuan, and X. Zhang, “Speeding up the bilateral filter: A joint acceleration way,” in *IEEE Trans. Image Process.*, vol. 25, no. 25, pp. 2657-2672, 2016.
- [6] F. Porikli, “Constant time $O(1)$ bilateral filtering,” in *Proc. IEEE CVPR*, pp. 1-8, Jun. 2008.
- [7] K. Sugimoto and S. Kamata, “Compressive bilateral filtering,” in *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3357-3369, 2015.
- [8] Y.-C. Tseng, P.-H. Hsu, and T.-S. Chang, “A 124 Mpixels/sec VLSI design for histogram-based joint bilateral filtering,” in *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3231-3241, 2011.
- [9] S.-K. Han, “An architecture for high throughput and improved-quality stereo vision processor,” M.S. thesis, Univ. Maryland, College Park, MD, 2010.
- [10] A. Gabiger, M. Kube, R. Weigel, and R. Rose, “An FPGA-based fully synchronized design of a bilateral filter for real time image denoising,” in *IEEE Trans. Ind. Elect.*, vol. 61, no. 8, pp. 4093-4104, 2014.
- [11] T. Q. Vinh, J. H. Park, Y. C. Kim, and S. H. Hong, “FPGA implementation of real-time edge-preserving filter for video noise reduction,” in *Proc. IEEE ICCEE*, pp. 611-614, 2008.
- [12] E. Sakthivel, V. Malalathi, M. Arunraja, and G. Perumalvignesh, “NABI: low power, high speed FPGA based novel approach for bilateral filter,” *Intl. J. Innovation and Scientific Res.*, vol. 25, no. 2, pp. 646-653, 2016.
- [13] K. Pahari, *VLSI digital signal processing systems*, John Wiley & Sons, 1999.
- [14] K. Hwang, *Computer arithmetic: principles, architecture, and design*, Wiley, 1979.

현 용 호 (Yongho Hyun)



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

박 태 근 (Taegeun Park)



1985년 : 연세대학교 전자공학 학사
1988년 : Syracuse Univ. 컴퓨터공학 석사
1993년 : Syracuse Univ. 컴퓨터공학 박사

1991년~1993년 : Coherent Research Inc. VLSI 설계 엔지니어
1994년~1998년 : 현대전자 System IC 연구소 책임 연구원
1998년~현재 : 가톨릭대학교 정보통신전자공학부 교수
<관심분야> VLSI 설계, CAD, 컴퓨터 구조