DBpia

경계선 보존 알고리즘 기반의 디블로킹 필터와 효율적인 VLSI 구조

정회원 트룽퀑빈*, 준회원 김 지 훈**, 정회원 김 영 철**°

Deblocking Filter Based on Edge-Preserving Algorithm And an Efficient VLSI Architecture

Truong Quang Vinh* *Regular Member*, Ji-Hoon Kim** *Associate Member*,
Young-Chul Kim**° *Regular Member*

요 약

본 논문은 새로운 경계선 보존 알고리즘을 이용하여 블록화 현상을 제거하는 디블로킹 필터와 HD해상도의 실시간 영상처리가 가능한 디블로킹 필터의 VLSI구조를 제안한다. 기존의 블록 분류 기반의 접근 방법과 달리 제안된 알고리즘은 픽셀 분류 기반 접근을 사용한다. 또한 제안된 경계선 보존 맵은 픽셀을 경계선 영역과 평탄 영역으로 분류하며, 블록화 현상 제거에 사용되는 오프셋 필터와 경계선 보존 필터의 기반이 된다. 이를 바탕으로 제안된 디블로킹 필터의 VLSI구조는 고연산량 처리를 위하여 블록 전체에 파이프라인 기법을 적용하였다. 또한 블록 버퍼를 위한 메모리 절감 구조는 메모리의 사용을 최적화 시킨다. 본 필터는 VHDL을 이용한 설계를 통하여 CycloneII FPGA상에서 구현된 구조의 동작을 검증 후, Synopsys의 Design Compiler와 ANAM 0.25 μm CMOS cell library로 합성하여 칩으로 구현하였을 때의 성능을 예측하였다. 제안된 알고리즘의 실험 결과는 세밀한 영상 성분을 보존하면서 효과적으로 블록화 현상을 제거하며, 픽셀 분류 기반에서 제안된 알고리즘은 블록 분류 기반보다 PSNR 성능이 우수함을 보였다.

Key Words : Edge-Preserving Filter, Offset Filter, Deblocking Filter, VLSI Architecture, HD

ABSTRACT

This paper presents a new edge-preserving algorithm and its VLSI architecture for block artifact reduction. Unlike previous approaches using block classification, our algorithm utilizes pixel classification to categorize each pixel into one of two classes, namely smooth region and edge region, which are described by the edge-preserving maps. Based on these maps, a two-step adaptive filter which includes offset filtering and edge-preserving filtering is used to remove block artifacts. A pipelined VLSI architecture of the proposed deblocking algorithm for HD video processing is also presented in this paper. A memory-reduced architecture for a block buffer is used to optimize memory usage. The architecture of the proposed deblocking filter is prototyped on FPGA Cyclone II, and then we estimated performance when the filter is synthesized on ANAM 0.25 μm CMOS cell library using Synopsys Design Compiler. Our experimental results show that our proposed algorithm effectively reduces block artifacts while preserving the details.

※ 본 연구는 교육과학기술부 및 한국연구재단의 지역혁신인력양성사업 과 부분적으로 지식경제부 및 정보통신산업진흥원의 대학IT연구센터 지원 사업으로 수행된 연구결과임 (NIPA-2011-C1090-1111-0008)

* University of Technology, HoChiMinh City, VietNam Faculty of Electrical-Electronics Engineering(tqvinh@hcmut.edu.vn),
** 전남대학교 전자컴퓨터공학과 정보통신 시스템온칩 연구실(bbode79@nate.com, yckim@jnu.ac.kr), (° : 교신저자)

논문번호 : KICS2011-08-336, 접수일자 : 2011년 8월 3일, 최종논문접수일자 : 2011년 11월 21일

www.dbpia.co.kr

Ⅰ. 서 론

최근의 비디오 후처리(post processing) 기술은 대부분 고화질 디스플레이 제품의 영상 품질을 향상시키기 위하여 사용되고 있다. 그러나 대역폭과 미디어 저장 공간의 제한 때문에 영상 압축 기술 없이 고화질의 영상을 제공하기 힘들다. 따라서 이러한 제한 조건을 고려하여 다양한 국제 압축 표준 방식이 개발 되고 있다. 압축 표준 방식에는 MPEG-1/2[1], MPEG-4[2] 와 같은 기본적인 변환 방식이 있으며, 최근에는 블록 기반의 이산 코사인 방식(BDCT)을 이용한 H.264/AVC[3] 변환 방식이 있다. 블록 기반의 이산 코사인 변환 방식은 널리 알려진 효율적인 변환 방식 중 하나로써, 이미지의 블록화가 드물게 나타나며, 압축 어플리케이션에 주로 활용된다. 그러나 저전송율 압축에서, 블록 기반의 이산 코사인 변환을 이용한 코딩 기술은 격자 잡음(grid noise), 계단 잡음(staircase noise), 모서리 잡음(outlier noise)을 현저하게 발생시킨다. 그러므로 비디오 후처리시스템에서 이러한 블록화 현상 제거를 위한 디블로킹 필터는 필수적인 요소이다.

블록화 현상을 제거하기 위해서 현재까지 다양한 디블로킹 방법이 제안 되어왔다. 이들 방법은 크게 공간 영역과 주파수 영역을 이용한 방법으로 나눌 수 있다. 공간 영역을 통한 방법은 공간 적응형 필터를 이용하는 방법[4-7], 통계모델을 이용하는 방법[8]과 POCS(Projection Onto Convex Sets) 이론을 이용하는 방법[9,10]이 제안 되었다.

반면 주파수 영역을 통한 방법에는 이산 코사인 변환(DCT)을 이용하는 방법[11,12]과 이산 웨이블렛 변환(DWT)을 이용하는 방법[13], 이미지를 주파수 영역으로 전환하기 위해서 인접 블록들 간의 상관관계를 분석 후 블록화 현상을 최소화하기 위해서 필터링을 적용한 방법[14] 등이 있다. 위에서 제안된 방법들의 하드웨어 설계구조는 찾아보기 힘들며 일부 알고리즘들[8,11-14]은 고연산량을 필요로 하기 때문에 높은 설계비용과 지연을 유발시켜 실시간 처리에는 부적합하다. 따라서 대부분 디블로킹 필터의 하드웨어 설계 시 저연산량과 저복잡도를 지닌 공간 영역 필터링 방법을 사용한다.

공간 영역 필터링 방법 중 제안된 방법으로는 8×8 정사각형 크기의 디블로킹 블록을 이용하여 이미지를 수평활동 영역과 수직활동 영역으로 분류한 후 다시 임계값을 통하여 균일, 수평, 수직, 복잡 디블로킹 블록 총 4개로 분류한 후 블록화 현상 종류에 따라 각각 필터링을 적용하는 방법이 있다[4]. 또 다른 방법으로는 영역별 엔트로피를 분석하여 블록화 현상을 제거하는 방법이 제안 되었다[5]. 이 방법은 각 영역을 엔트로피 값 기반으로 정밀, 평탄, 중간 영역 총 3개의 영역으로 나눈다. 그 밖에도 블록을 균일, 비균일 블록으로 분류하여 필터링을 수행하는 방법[6]이 제안 되었다.

위에서 제안된 방법들은 블록을 여러 부류로 나눈 후, 분류된 블록의 특징을 고려하여 각각 다른 필터링을 적용한다. 평탄한 영역에서는 블록화 제거 강도를 강하게 적용하고, 경계선을 포함한 세밀하고, 복잡한 영역에서는 블록화 제거 강도를 약하게 적용한다. 따라서 블록화 현상을 제거하면서, 이미지의 세밀함과 선명함이 보존 될 수 있다. 그러나 고정된 블록사이즈를 통한 분류가, 서로 다른 영역을 항상 완벽하게 분류 할 수 있는 것은 아니다. 이러한 분류 방법은 고정된 블록 사이즈에 평탄 영역과 경계선 영역이 함께 평탄 영역의 블록으로 분류 될 수 있다는 단점이 있다. 경계선 영역이 포함된 블록을 평탄 영역으로 판단하고, 블록화 제거 강도를 강하게 처리 하였을 경우, 포함된 경계선을 구성하고 있는 픽셀은 손상된다. 그러므로 고정된 블록 사이즈를 통한 분류 후 디블로킹 필터를 적용 할 경우 의도치 않게 부적절한 영역을 평탄화 시킬 수 있다. 그림 1 은 경계선을 포함하고 있지만 평탄 지역으로 분류된 경우를 보여준다.

본 논문에서는 블록화 현상 제거를 위하여 픽셀 분류를 통한 새로운 디블로킹 알고리즘과 이를 구현하기 위한 효율적인 하드웨어 구조를 제안한다. 제안한 알고리즘은 고정된 블록 크기를 이용한 블록 분류 방법과 달리, 경계선 검출 처리를 통하여 각각의 픽셀을 분류 후 경계선 맵을 만든다. 생성된 경계선 맵의 정보를 바탕으로 오프셋 필터와 경계선 보존 필터를 적응적으로 적용한다. 하드웨어 구

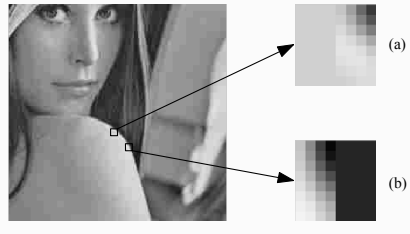

그림 1. (a) 평탄 영역으로 분류되었지만 경계선 픽셀을 포함한 경우 (b) 경계선 영역으로 분류 되었지만 평탄 영역을 포함한 경우

www.dbpia.co.kr

조 구현은 HD 실시간 비디오 영상의 후처리 시스템을 위해서, 파이프라인 구조를 적용하고 메모리 사용을 줄이면서 높은 연산량을 처리 할 수 있는 설계 면적이 최소화된 블록 버퍼를 사용한다.

본 논문의 구성은 다음과 같다. Ⅱ 장에서는 제안한 경계선 보존 디블로킹 알고리즘을 설명하며, Ⅲ 장에서는 제안한 하드웨어 구조를 상세히 설명한다. Ⅳ 장에서는 알고리즘 시뮬레이션과 구조 검증, 기존의 다른 디블로킹 방법과의 비교를 통한 성능을 분석하며. 마지막으로 Ⅴ 장에서 결론을 맺는다.

Ⅱ. 제안된 경계선 보존 디블로킹 알고리즘

제안한 경계선 보존 알고리즘 기반의 디블로킹 필터는 경계선 검출 처리 결과를 통한 경계선 보존 맵을 바탕으로 2단계 적응형 필터를 이용하여 블록화 현상을 제거하며, 필터의 처리과정은 그림 2 와 같다.

픽셀분류 단계에서는 영상으로 부터 입력된 픽셀을 경계선 검출 처리를 통해서 픽셀을 분류한다. 이는 수직, 수평경계선 검출, 경계선 강도에 대한 데이터를 획득하기 위한 과정이다. 다음 단계로서 이들 데이터를 근간으로 이진 경계 맵을 만든다. 다음으로 수평과 수직 경계선 맵을 근거로, 평탄 영역에서 격자 잡음을 제거 할 수 있는 오프셋 필터를 수행한다. 마지막으로 경계선 강도 맵을 근거로, 계단 잡음, 모서리 잡음을 제거할 수 있는 경계선 보존필터링작업을 수행한다.

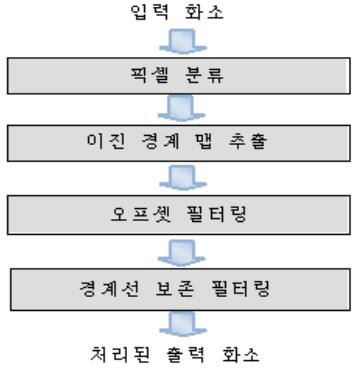

그림 2. 경계선 보존 알고리즘 기반의 디블로킹 필터 처리 과정

2.1 픽셀 분류

그림 3 처럼 8×8 크기의 블록이 있다.

기존의 관련 연구에서는 이 블록을 평탄 영역과 경계 영역으로 분류 할 수 없었다. 본 논문에서는 이 블록의 모든 픽셀을 경계선 검출 처리에 의해서 분류한다. 경계선 검출 처리는 소벨(Sobel) 또는 프리윗(Prewitt) 등의 다양한 회선 마스크(Convolution mask)를 사용 할 수 있다. 본 논문에서는 실시간 처리를 위한 하드웨어 구현에 적합한 저연산량으로 경계선 방향과 경계선 강도 측정이 가능한 프리윗 마스크를 사용 하였다.

3×3 마스크의 프리윗 경계선 검출 처리 후, 각 픽셀의 수직, 수평 방향으로 방향성 정도를 나타내는 G_x, G_y 를 얻는다. 또한 이들의 크기 즉 경계선 강도는 식 (1)을 통해서 얻을 수 있다.

$$G = |Gx| + |Gy| \tag{1}$$

3상 이진 경계선 보존 맵은 살펴보면 임계값 T, T_d 를 적용하여 블록을 구성하는 픽셀의 x축 경계정보, y축 경계정보, 경계선 강도인 E_x, E_y, E_z를 얻는다. 본 논문에서는 실험을 통하여, 임계값 T, T_d 를 각각 20, 10으로 설정하였다.

$$\begin{aligned} Ex_{i,j} &= \begin{cases} 0, & |Gx| < T_d, \\ 1, & |Gx| \geq T_d, \end{cases} \\ Ey_{i,j} &= \begin{cases} 0, & |Gy| < T_d, \\ 1, & |Gy| \geq T_d, \end{cases} \\ Ez_{i,j} &= \begin{cases} 0, & G < T, \\ 1, & G \geq T. \end{cases} \end{aligned} \tag{2}$$

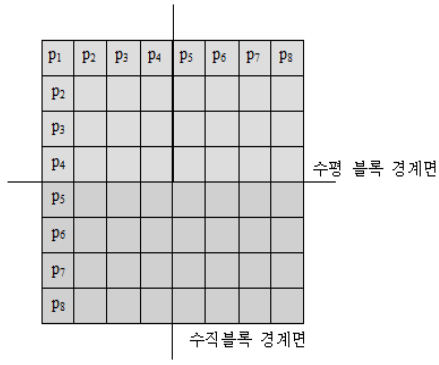

그림 3. 블록에서 픽셀 영역

www.dbpia.co.kr

G_x, G_y, G 와 T, T_d 의 비교 결과 값인 '0' 과 '1'은 각 픽셀이 평탄영역 또는 경계선 영역에 속해 있는지를 나타낸다. 따라서 적응형 필터는 영상의 세밀함을 보존하기 위하여, 3상 이진 경계선 보존 맵을 이용할 수 있다.

2.2 격자 잡음을 위한 오프셋 필터

3상 이진 경계선 보존 맵을 얻은 후 블록화 현상 제거를 위해서 블록의 모든 픽셀에 1차원 오프셋 필터링을 적용한다. 오프셋 필터는 먼저 수평 방향으로 처리 한 후, 수직으로 처리한다. 식 (3)은 1차원 오프셋 필터링의 과정이다.

$$
\begin{aligned}
&offset_i = p_{i,4} - p_{i,5}\ ,\\
&p_{i,1} = p_{i,1} - offset_i / 16 \times \overline{Ex_{i,1}},\\
&p_{i,2} = p_{i,2} - offset_i / 8 \times \overline{Ex_{i,2}}\ ,\\
&p_{i,3} = p_{i,3} - offset_i / 4 \times \overline{Ex_{i,3}}\ ,\\
&p_{i,4} = p_{i,4} - offset_i / 2 \times \overline{Ex_{i,4}} - offset_i / 4 \times Ex_{i,4}\ ,\\
&p_{i,5} = p_{i,5} + offset_i / 2 \times \overline{Ex_{i,5}} + offset_i / 4 \times Ex_{i,5}\ ,\\
&p_{i,6} = p_{i,6} + offset_i / 4 \times \overline{Ex_{i,6}},\\
&p_{i,7} = p_{i,7} + offset_i / 8 \times \overline{Ex_{i,7}}\ ,\\
&p_{i,8} = p_{i,8} + offset_i / 16 \times \overline{Ex_{i,8}},
\end{aligned}
\qquad (3)
$$

여기서 $P_{i,j}$는 그림 2에서 8×8 블록에서 (i,j) 의 위치에 있는 픽셀의 휘도 값을 나타낸다. 그리고 $\overline{Ex_{i,j}}$ 는 $Ex_{i,j}$의 역이다. 즉 $Ex_{i,j}$가 0 이면, $\overline{Ex_{i,j}}$ 는 1이다.

오프셋(offset)은 블록 경계에서 불연속의 정도를 의미하며. 이 값은 블록의 경계에서 2개의 이웃 픽셀간의 휘도 차에 의해 계산된다. 본 논문에서는 블록 경계의 불연속 부분을 제거하기 위해서 offset/2, offset/4, and offset/8 비트이동 연산이 된 오프셋 값을 이용한다[4],[5]. 이 방법은 블록의 중심에서 왼쪽 픽셀인 $p_{i,1}$, $p_{i,2}$, $p_{i,3}$, $p_{i,4}$과 오른쪽 픽셀인 $p_{i,5}$, $p_{i,6}$, $p_{i,7}$, $p_{i,8}$은 블록 경계에서 픽셀간의 서로 다른 휘도 차이를 균형 있게 조정하여 블록화 현상을 제거한다. 블록화 현상은 평탄 영역에서 시각적으로 더욱 두드러지기 때문에, 블록 경계의 $p_{i,4}$, $p_{i,5}$ 2개 픽셀을 제외하고 경계선 보존 맵을 이용하여 필터 처리를 수행한다. 가령 $p_{i,4}$, $p_{i,5}$가 만약 경계선 영역의 픽셀이라면, 가중치 offset/4를 적용하여 휘도 값을 조정 할 것이다. 경계선 보존 맵에서 E_x는 수평 필터 처리를 위해서 적용되며, E_y는 수직 필터 처리를 위해서 적용된다.

오프셋 필터 처리 단계를 거쳐서 평탄 영역의 블록성 잡음은 제거 될 수 있으나, 계단 잡음과 모서리 잡음은 여전히 경계 영역에 남아 있다. 따라서 다음의 경계선 보존 필터를 통해서 이들 잡음을 제거한다.

2.3 모서리, 계단 잡음을 위한 경계선 보존 필터

다음으로 계단 잡음과 모서리 잡음 제거를 위해서 경계선 보존 필터를 제안한다. 경계선 보존 필터의 방법은 경계선을 따라서 평탄화 처리를 하면서 또한 이미지의 세밀한 영상 성분을 향상 시킬 수 있는 bilateral 필터의 기본적인 특성을 이용한다[15].

행렬 (4)처럼 3×3의 픽셀 배열이 있다.

$$
X = \begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix} \qquad (4)
$$

x_i ($i = 1,...,9$)는 휘도 값이며, x_5는 필터 처리 된 픽셀이다. 휘도 간격 d_i는 식 (5)처럼 중심 픽셀인 x_5와 x_i간의 차다.

$$
d_i = \left| x_i - x_5 \right|, \quad i = 1, \cdots, 9 \qquad (5)
$$

필터의 회선 마스크를 위한 상관계수는 다음 식으로부터 산출된다.

$$
c_i = (255 - d_i)^q \qquad (6)
$$

상관계수 c_i는 휘도 간격 d_i 값 보다 더 큰 값이 얻어진다. 즉 중앙 픽셀의 휘도 값과 차이가 많은 픽셀들은 차이가 적은 다른 픽셀들보다 더 적은 가중치가 적용된다. 이러한 이론은 아래의 회선 마스크를 통해 드러난다.

$$
C = \frac{1}{\sum_{i=1}^{9} c_i} \begin{bmatrix} c_1 & c_2 & c_3 \\ c_4 & c_5 & c_6 \\ c_7 & c_8 & c_9 \end{bmatrix} \qquad (7)
$$

www.dbpia.co.kr

가우시안 분포와 비슷한 휘도 분포의 유사성을 기반으로 상관계수를 조절하는 bilateral 필터와 유사하게, 제안된 경계선 보존 필터는 휘도 간격에 지수함수를 적용한 상관계수를 적용했다. 따라서 회선 마스크를 통한 처리는 경계선을 따라서 평탄화 처리를 수행하면서, 계단 잡음과 모서리 잡음을 제거할 수 있다. 식(6)에서 인자 q는 경계선의 평탄화 정도를 조절하기 위한 계수이고, 실험을 통하여 선택되었다. 본 논문에서는 q 값을 8로 설정하였다. 분모 $\sum c_i$는 출력 이미지를 위해서 상관계수 마스크 C 를 정규화 시키는 요소이다. 경계선 보존 필터의 식(8)과 같으며, 경계 영역의 픽셀들에만 적용된다.

$$p_{i,j} = p_{i,j} \times \overline{Ez_{i,j}} + edge_filter(X) \times Ez_{i,j}$$
$$edge_filter(X) = X \times C \quad (8)$$

Ⅲ. 제안된 디블로킹 필터구조

제안된 디블로킹 필터의 구조는 실시간 처리를 위해 파이프라인 기법을 적용하였고, 오프셋 필터와 경계선 필터를 병렬적으로 처리되도록 하였다. 세부사항을 살펴보면 256 단계의 픽셀 휘도 값을 표현하기 위해서 8 비트 부동 소수점을 사용했으며, HDTV 1080p, 30 Hz의 프레임율 수준에서 필터 처리가 가능 하도록 설계 하였다. 요구되는 총계산량은 초당 $1{,}920 \times 1{,}080 \times 30 \approx 75.10^6$ 픽셀이다. 설계된 필터는 2 클록당 1픽셀을 처리하므로 필터의 동작 주파수는 150 MHz이다.

3.1 최상위 블록 설계

디블로킹 필터의 구조는 블록 버퍼(block buffer), 픽셀 분류기(Pixel classifier), 오프셋 필터(offset filter), 경계선 보존 필터(edge preserving filter)와 이들 필터를 위한 컨트롤러(controller)로 총 5개의 요소로 구성되어 있으며 최상위 블록 구조는 그림 3과 같다.

제안된 디블로킹 필터의 블록 버퍼는 래스터 주사(raster scan)방식으로부터 받은 입력 픽셀을 오프셋 필터를 위하여, 8×8 픽셀 블록으로, 픽셀 분류기를 위하여 3×3 픽셀 블록으로 각각 변환한다. 오프셋 필터 적용 후에는 경계선 보존 필터를 위하여 8×8 픽셀 블록이 3×3 픽셀 블록으로 변환된다. 컨트롤러는 유한 상태 기기(FSM)로 구현되어 있으며 데이터 흐름을 제어한다. 파이프라인 단계로 구성된 FIFO 메모리는 지연 유닛으로써, 각 신호들의 동기화를 담당한다.

3.2 메모리 사용 절감 효과를 지닌 블록 버퍼

제안된 구조에서 픽셀 분류기는 입력으로 3×3 픽셀 블록을, 오프셋 필터는 8×8 픽셀 블록을, 경계선 보존 필터는 3×3 픽셀 블록의 데이터 형식을 입력으로 요구한다.

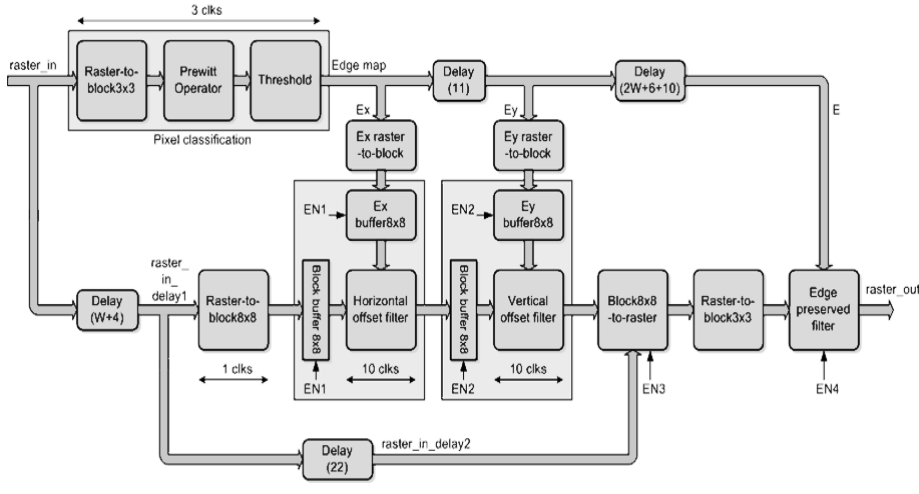

그림 4. 디블로킹 필터의 블록 다이어그램

www.dbpia.co.kr

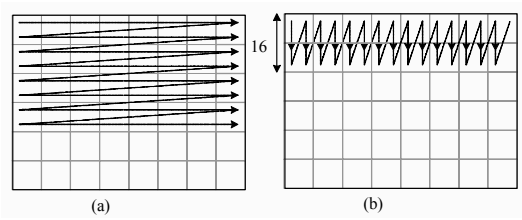

그림 5. (a)일반적인 래스터 주사방식, (b)변경된 래스터 주사방식

그림 5(a)는 래스터 주사 방식을 보여 주고 있다. 일반적으로 영상처리 하드웨어[16]를 위해 그림 6(a)처럼 8×8블록을 생성하기 위하여 8 라인 버퍼를 사용한다. 그러므로 HD해상도를 갖는 입력 영상을 위해서는 매우 큰 FIFO 메모리 버퍼를 필요로 한다.

하지만 본 논문에서는 블록 버퍼를 위한 메모리 사용을 줄일 수 있는 방법으로, 그림 5(b)처럼 변경된 주사 형태를 사용하는 새로운 블록 버퍼를 제안한다. 새로운 블록 버퍼는 수직방향으로부터 픽셀을 순차적으로 불러들인 후 그림 6(b)처럼 오프셋 필터를 위한 8×8 픽셀블록을 생성한다. 새로운 블록 버퍼 구조는 임시적으로 픽셀을 저장하기 위해서 8 바이트 FIFO 메모리를 요구한다.

1080P 해상도의 영상일 경우, 일반적인 블록 버퍼가 (7×1920+8)13,448 바이트를 사용하는 반면, 입력 데이터를 수직으로 주사 방식에서 메모리 접근을 위한 주소 발생기를 추가로 필요로 하는 새로운 블록 버퍼는 총 (7×16+8)120 바이트의 메모리를 사용한다. 이는 99% 메모리 사용을 절감한 수치이다. raster-to-3×3block은 3×3 레지스터 배열과 FIFO 메모리로 구성되어 있으며, 이는 8×8 블록 버퍼와 유사한 형태의 구조이다.

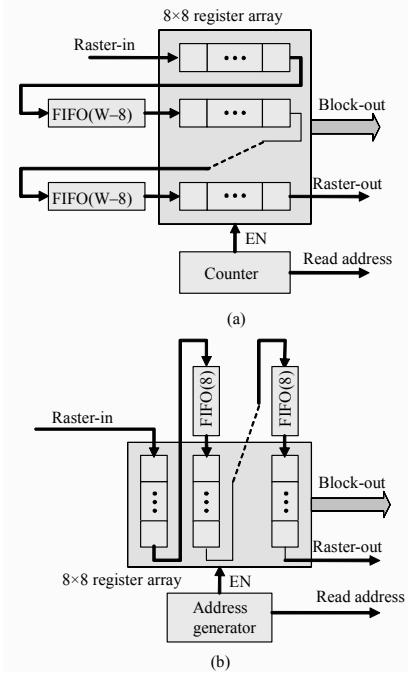

그림 6. (a) 수평주사방식을 사용한 일반적인 8×8 블록 버퍼, (b) 수직주사방식을 사용하여 메모리 사용이 절감된 블록 버퍼

3.3 픽셀 분류기

픽셀 분류기는 경계선 검출을 위한 Prewitt operator 단계와 임계값을 통해 E_x, E_y, E_z 를 추출하는 thresholding 단계로 구성되어 있으며 그림 7 과 같다. 총 3 단계의 파이프라인 구조가 적용 되었으며. 매 클록마다 각 픽셀에 대한 E_x, E_y, E_z 를 추출하여 3상 이진 경계 보존 맵을 생성한다.

3.4 오프셋 필터

오프셋 필터는 하드웨어 복잡도를 줄이기 위해서 비트 이동 연산을 기반으로 설계하였다. 픽셀 벡터와 경계선 보존 맵의 벡터를 입력으로 받으며, 매 클록마다 8 픽셀을 처리한다. 따라서 8×8 픽셀 블록을 모두처리하기 위해서는 8 클록이 소요된다. 수평과 수직 방향의 블록성 잡음 제거를 위해서는 동일한 2개의 오프셋 필터가 사용된다. 그러나 입력 데이터의 형식에서 이들 2 개의 필터는 차이가 있다. 수평 오프셋 필터는 수평 방향의 픽셀과 경계선 보존 맵을 입력 데이터로 받는 반면에 수직 오프셋 필터는 수직 방향의 픽셀과 경계선 보존 맵을 입력

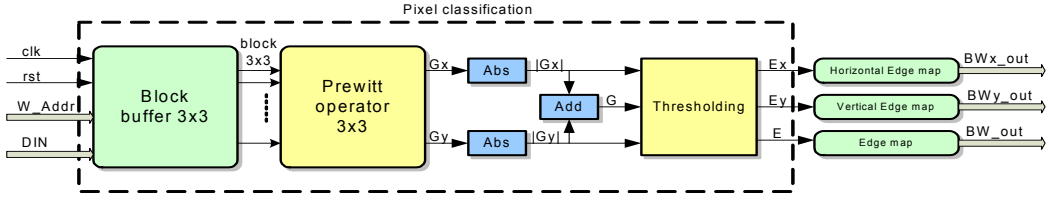

그림 7. 픽셀 분류기 장치의 구조

www.dbpia.co.kr

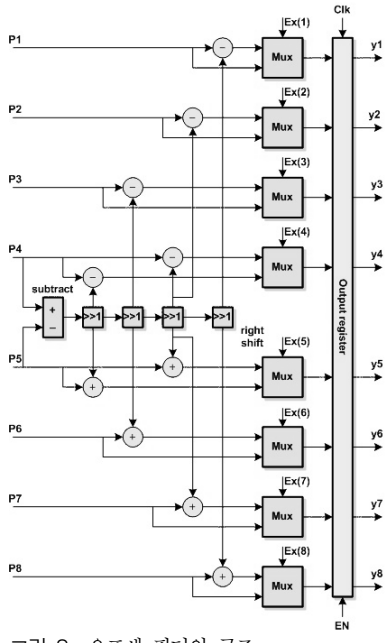

그림 8. 오프셋 필터의 구조

데이터로 받는다. 그림 8 은 제안된 오프셋 필터의 구조를 블록 다이어그램 형태로 보여주고 있다.

3.5 경계선 보존 필터

경계선 보존 필터는 3×3 픽셀 블록과 E_z 맵을 입력 데이터로 받는다. 3×3 입력 블록의 회선 마스크는 휘도 간격 산출, 상관계수 추출, 상관 계수 정규화 총 3단계에 걸쳐서 얻어진다. 마지막으로 이 과정에서 획득된 회선 마스크와 입력 픽셀과의 곱을 통해서 경계선 보존 필터링이 수행된다. 경계선 보존 필터의 블록 구조는 그림 9 와 같으며, 임계 경로는 4개의 곱셈기, 2개의 감산기, 각각 1개의 절대 값 처리기, 가산기, 다중화기, 나눗셈기로 구성되어있다. 이와 같은 임계 경로의 긴 지연시간을 피하기 위해서 12 단계의 파이프라인 구조를 적용하였다.

3.6 주소 생성기와 컨트롤러

주소 생성기는 수직 래스터 주사 방식에서 주소

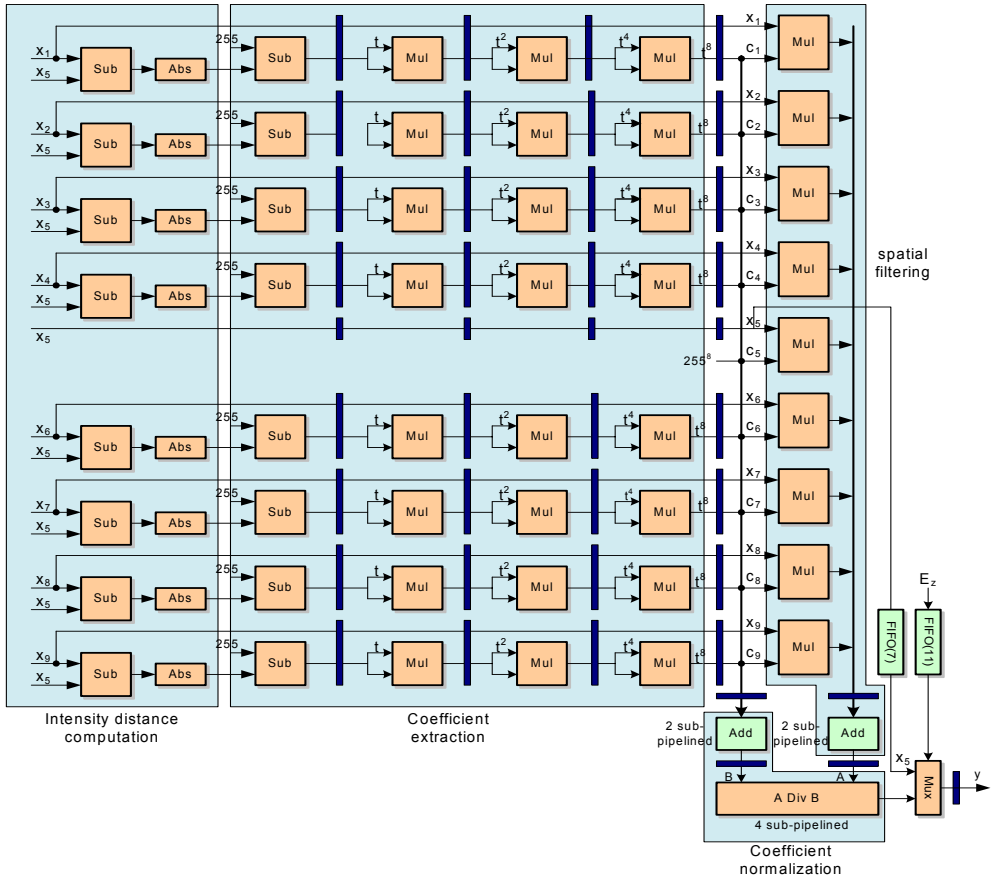

그림 9. 경계선 보존 필터를 위한 전체 파이프라인 구조

www.dbpia.co.kr

카운터를 생성한다. 주소 발생기는 입력 영상의 가로와 세로 크기를 입력으로 받으며, 그림 4(b)처럼 래스터 주사에 관한 주소 값을 계산하기 위해서 유한 상태기기(FSM)를 이용하였다. 컨트롤러 유닛은 활성화 신호인 EN_x, EN_y, EN_z 의 유한상태기기를 통해서, 오프셋 필터와 경계선 보존 필터를 제어한다. 사용된 유한 상태기기는 오프셋 필터를 위한 8×8블록과 경계선 보존 필터를 위한 3×3 블록이 각 필터의 입력으로 모두 사용이 가능할 경우 활성화 시그널을 활성화 시킨다. 예를 들어, 경계영역에서 4 × 4 픽셀 블록의 경우 이웃 픽셀에 대한 정보 부족으로 오프셋 필터를 처리하지 않는다. 디블로킹 필터는 총 9W+47 (W : 입력 영상의 수평 픽셀 수)개의 파이프라인 단계로 구성 되어 있다. 따라서 입력 픽셀과 출력 픽셀 간의 9W+47 클록의 지연이 발생한다.

Ⅳ. 실 험

제안된 디블로킹의 알고리즘 평가 및 검증은 알고리즘 시뮬레이션 통해 수행 되었으며, VHDL을 사용하여 구현된 VLSI 하드웨어 구조 검증은 FPGA 상에서의 구현과 칩으로 구현하였을 때의 합성결과를 분석하였다. 먼저 Matlab 7.4.0을 통해 제안된 알고리즘을 소프트웨어 기반에서 시뮬레이션을 하였고, CycloneII FPGA 기반에서 구현된 구조의 동작을 검증 하였다. FPGA 검증 후 Synopsys Design Compiler 와 ANAM 0.25 μm CMOS cell library 로 합성하였다.

4.1 알고리즘 시뮬레이션

Matlab에서 알고리즘 시뮬레이션을 위해서 먼저 LENA, Barbara, Pepper 테스트 이미지 세트를 이용하여 실험을 수행하였다. 실험에 사용된 테스트 이미지는 8비트 흑백 영상이며 512×512의 영상 크기이다. 제안한 알고리즘을 평가하기 위해서, 공간 영역 기반 분류 방법[4]-[7]과 PSNR 비교를 통해 분석하였다. 분석 결과는 표 1 과 같다. 실험 결과 본 논문에서 제안된 알고리즘이 대부분의 이미지와 비트율에서 가장 높은 PSNR 수치를 보였다. 그림 10은 Lena512 테스트 이미지에 각각의 디블로킹 알고리즘을 적용한 결과이다.

성능비교에 사용된 비교 알고리즘들은 픽셀 활성도에 의해 픽셀 블록을 분류한 후 비트이동 연산을 이용하여 오프셋 필터를 적용한 방법[4], 영역별 엔트로피를 사용하여 블록을 분류하고, 블록화 현상 제거를 위해 저주파 통과 필터를 사용한 방법[5], 픽셀 블록을 균일 블록과 비균일 블록으로 나눈 후 균일 블록에는 WABG 기법을 수행하고, 비균일 블록에는 DFOVS (the deblocking frames of variable size) 기법을 수행하는 방법[6], AC-Coefficient Energy를 활용하여 저활성화 클래스와 고활성화 클래스로 분류 후 각 클래스에 맞는 적응형 필터를 적용하는 방법[7]들이 사용하는 알고리즘 성능비교 대상으로 선정되었다. 위의 모든 알고리즘은 디블로킹 필터를 적용하기 전에 고정된 블록 크기를 바탕으로 픽셀 블록을 분류한다.

다음으로 알고리즘 시뮬레이션을 위해서, 동영상 테스트 영상으로 물체의 빠른 움직임과 움직임이 있는 배경, 다수의 블록화 현상을 포함한 Foreman

표 1. 디블로킹 방법에 따른 PSNR 비교

Image	Bit rate (bpp)	PSNR (dB)					
		JPEG	Our method	Ref. [4]	Ref. [5]	Ref. [6]	Ref. [7]
Lena 512	0.25	30.04	**31.48**	31.37	30.91	31.01	31.22
	0.36	32.95	**33.62**	33.48	32.88	32.85	33.37
	0.46	34.26	34.60	**34.62**	34.52	33.70	34.51
Peppers 512	0.25	30.18	**31.29**	31.19	30.62	30.74	30.75
	0.37	32.47	**33.24**	33.06	32.43	32.45	32.72
	0.47	33.61	**34.14**	34.02	33.12	33.22	33.72
Barbara 512	0.44	26.99	**27.40**	27.38	26.67	27.11	27.23
	0.54	28.25	**28.55**	28.55	27.71	28.08	28.23
	0.69	30.16	30.25	**30.28**	30.10	29.33	29.73

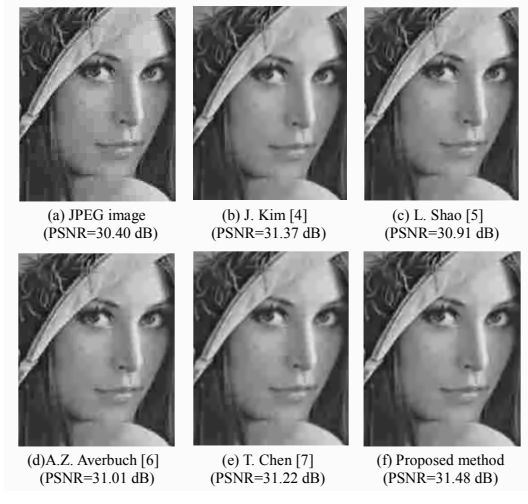

그림 10. Lena512 흑백이미지에 디블로킹 기법 적용결과

www.dbpia.co.kr

(a) Before processing (PSNR = 30.27 dB)　(b) After processing (PSNR = 31.15 dB)

그림 11. 100kbps의 "Foreman" 동영상에서 15번째 프레임에 디블로킹 알고리즘 적용 전 후 비교

표 2. 동영상 테스트를 통한 PSNR 성능비교

Test sequences	Bit rate (kbps)	Average PSNR (dB)	Average PSNR improvement (dB)			
			[4]	[5]	[7]	Our method
Foreman (CIF, 300frs)	100	27.64	0.612	0.511	0.554	0.707
	200	27.71	0.592	0.494	0.562	0.698
	300	27.84	0.561	0.465	0.550	0.689
News (CIF, 300frs)	100	28.10	0.360	0.425	0.401	0.562
	200	28.17	0.333	0.419	0.382	0.549
	300	28.32	0.284	0.410	0.371	0.528
Silent (CIF, 150frs)	100	28.13	0.542	0.498	0.525	0.616
	200	28.19	0.525	0.484	0.506	0.601
	300	28.48	0.476	0.439	0.470	0.571

영상과 물체의 느린 움직임과 움직임 없는 배경, 소수 블록화 현상을 포함한 News 영상, 앞서 소개한 두 영상의 중간적인 특성을 소유한 Silent 영상을 사용하였다. 테스트 영상의 크기는 CIF(352×288)이며, 테스트 영상의 인코딩은 인코딩 설정 항목 중 Profile@Level 설정 항목을 Simple@L3로, Encoding Type 설정 항목을 Single Pass로 설정 한 후 Xvid 인코더를 통해서 각각 100,200,300 Kbps 비트율로 인코딩하였다. 그림 11은 제안된 알고리즘의 성능을 시각적으로 보여주기 위해서 Foreman 영상의 임의 프레임에서 디블로킹 필터 적용 전후 영상을 보여주고 있다. 표 2는 제안된 알고리즘과 공간 영역 기반 분류 방법의 다른 알고리즘들을 실험 영상에 적용 후 성능을 PSNR 비교수치로 보여준다. 실험 결과 제안된 알고리즘은 모든 이미지와 비트율에서 가장 높은 PSNR 개선수치를 보이며 특히 움직임이 많은 영상, 저전송율 압축 영상에서 개선 효과가 더욱 뛰어남을 보여준다.

4.2 하드웨어 구현

제안된 디블로킹 필터의 구조는 RTL 수준에서

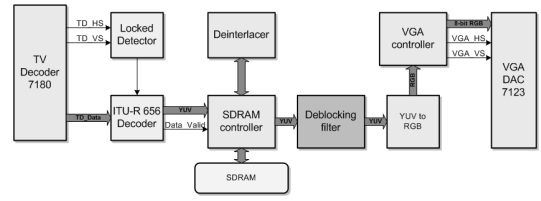

그림 12. 디블로킹 필터를 위한 검증 모델

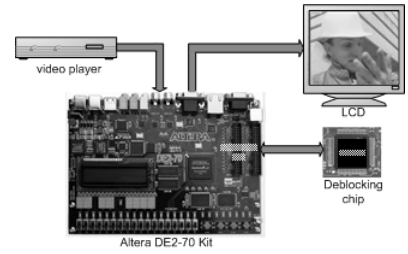

그림 13. 디블로킹 필터를 위한 검증 환경

VHDL를 사용하여 구현된, VLSI 하드웨어 구조를 FPGA CycloneII 상에서 동작을 검증 하였다. FPGA를 통한 검증은 CycloneII FPGA EP2C70F890을 사용한 DE2-70 개발 보드에서 수행되었으며, 그림 12는 FPGA 검증시스템의 구성을 보여준다. 다음으로 FPGA 검증 후 HD 비디오 영상 처리를 위하여 임계 경로의 지연 조건을 6.7ns 로 설정 한 후 Synopsys사의 Design Compiler를 이용하여 0.25 mm ANAM CMOS cell library로 합성 하였다, 구현된 디자인은 1080P 비디오 영상에서 150 Mhz의 동작 주파수로 동작하였다. 표 3 은 제안된 디블로킹 필터와 in-loop 필터를 이용한 다른 디자인과의 비교 합성 결과를 보여준다. 제안된 디블로킹 필터는 50.1K 게이트로써, in-loop 디블로킹 필터 알고리즘 보다 복잡도가 높다. 하지만 PSNR 비교결과 다른 알고리즘에 비해 향상된 수치를 보였고, 실제 시각적으로도 더욱 개선된 이미지를 얻을 수 있었다.

표 3. 제안된 디블로킹 필터의 합성 결과

Features	Ref. [17]	Ref. [18]	Our design
Algorithm	In loop filter	In loop filter	Edge-preserving filter
PSNR improvement	0 to 1 dB	0 to 1 dB	0.2 to 1.5 dB
Pipeline	Non	Non	192 stage
Process (μ)	0.25	0.18	0.25
Cycle/MB	615	250	512
Frequency required for HD at 30fps (MHz)	300	120	150
Memory (bytes)	640	30,720	310
Gate count (k)	20.66k	19.64k	50.1k

www.dbpia.co.kr

V. 결 론

본 논문에서 경계선 보존이 가능한 디블로킹 알고리즘과 이를 하드웨어로 구현하기 위한 효율적인 VLSI 구조를 제안하였다. 블록 기반의 분류 기법을 이용한 다른 디블로킹 기법과는 달리, 픽셀 기반의 분류 기법을 이용하여, 경계선 보존 맵을 생성하고, 생성된 맵을 기반으로 오프셋 필터와 경계선 보존 필터를 적용하였다. 성능비교 실험에서 이들 필터를 사용한 알고리즘은 다른 알고리즘에 비해서 이미지의 세밀한 영상 성분을 보존함과 동시에 블록화 현상을 효과적으로 제거함을 보였다.

제안한 알고리즘 수행을 위한 하드웨어 구조는 구조 전체에 파이프라인 기법 적용과 오프셋 필터와 경계선 보존 필터를 병렬 구조로 배치함으로써 고연산 실시간 비디오 영상 처리에 효율성을 높였고, 블록 버퍼의 사용은 메모리 사용을 최소화 하여. HD 해상도에서 실시간 영상처리가 가능하다.

참 고 문 헌

〔1〕 "Generic Coding of Moving Pictures and Associated Audio Information," Part 2: Video, *ISO/IEC 13818-2,* Nov., 1994.

〔2〕 MPEG Video Group, "MPEG-4 Video Verification Model Version 8.0," *ISO/IEC JTC1/SC29/WG11 N1796*, July, 1997.

〔3〕 Joint Video Team of ITU-T and ISO/IEC JTC 1, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification," *(ITU-T Rec. H.264 ㅡISO/IEC 14496-10 AVC)*, May, 2003.

〔4〕 J. Kim, M. Choi, and J. Jeong, "Reduction of Blocking Artifacts for HDTV Using Offset-and-Shift Technique," *IEEE Trans. Consum. Electron.*, Vol.53, No.4, pp.1736-1743. Nov., 2007.

〔5〕 L. Shao and I. Kirenko, "Coding Artifact Reduction Based on Local Entropy Analysis," *IEEE Trans. Consum. Electron.*, Vol.53, No.2, pp.691-696. May, 2007.

〔6〕 A.Z. Averbuch, A. Schclar, and D.L. Donoho, "Deblocking of Block-Transform Compressed Images Using Weighted Sums of Symmetrically Aligned Pixels," *IEEE Trans. Image Process.*, Vol.14, No.2, pp.200-212. Feb., 2005.

〔7〕 T. Chen, H.R. Wu, and B. Qiu, "Adaptive Postfiltering of Transform Coefficients for the Reduction of Blocking Artifacts," *IEEE Trans. Circuits Syst. Video Technol.*, Vol.11, No.5, pp.594-602. May, 2001.

〔8〕 Z. Li and E.J. Delp, "Block Artifact Reduction Using a Transform-Domain Markov Random Field Model," *IEEE Trans. Cir. Sys. Video Technol.*, Vol.15, No.12, pp.1583-1593. Dec., 2005,

〔9〕 J.J. Zou and H. Yan, "A Deblocking Method for BDCT Compressed Images Based on Adaptive Projections," *IEEE Trans. Cir. Sys. Video Technol.*, Vol.15, No.3, pp.430-435. Mar., 2005.

〔10〕 G.R. Kwon et al., "An Efficient POCS-Based Post-Processing Technique Using Wavelet Transform in HDTV," *IEEE Trans. Cons. Elec*. Vol.51, No.4, pp.1283-1290. Nov., 2005,

〔11〕 Y. Zhao, G. Cheng, and S. Yu, "Postprocessing Technique for Blocking Artifacts Reduction in DCT Domain," Electron. Lett, Vol.40, No.19, pp.1175-1176. Sept., 2004.

〔12〕 Y. Luo and R.K. Ward, "Removing the Blocking Artifacts of Block-Based DCT Compressed Images," *IEEE Trans. Image Process.*, Vol.12, No.7, pp.838-842. Jul., 2003.

〔13〕 H. Choi and T. Kim, "Blocking-Artifact Reduction in Block-Coded Images Using Wavelet-Based Subband Decomposition," *IEEE Trans. Cir. Sys. Video Technol.,* Vol.10, No.5, pp.801-805. Aug., 2000.

〔14〕 N.C. Kim et al., "Reduction of Blocking Artifact in Block-Coded Images Using Wavelet Transform," *IEEE Trans. Cir. Sys. Video Technol.,* Vol.8, No.3, pp.253-257. June, 1998.

〔15〕 C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and color Images," Proc. *IEEE Int. Conf. Comput.* Vision, pp.59-66. 1998.

〔16〕 C.T. Johnston, K.T. Gribbon, and D.G. Bailey, "Implementing Image Processing Algorithms on FPGAs," Proc. Eleventh Electronics New Zealand Conf., Palmerston North, New Zealand, pp.118-123. Nov., 2004.

www.dbpia.co.kr

〔17〕 Y.W. Huang et al., "Architecture Design for Deblocking Filter in H.264/ JVT/AVC," in Proc. *IEEE Int. Conf.* Multimedia Expo, MD, pp.I-693-I-696.2003.

〔18〕 T. M. Liu et al., "A Memory-Efficient Deblocking Filter for H.264/AVC Video Coding," Proc. *IEEE Int. Symp. Circuits Syst., Kobe*, Japan, pp.2140-2143. 2005.

트룽퀑빈 (Truong Quang Vinh) 정회원

1999년 2월 HoChiMinh Univ. 전자공학과 석사

2011년 2월 전남대학교 전자컴퓨터공학과 박사

2011년~현재 HoChiMinh Univ. 전기전자공학과 전임강사

<관심분야> 디지털 회로, SoC, 임베디드 시스템 설계, 영상관련 SoC 및 VDP 설계

김 지 훈 (Ji-Hoon Kim) 준회원

2010년 8월 전남대학교 전자컴퓨터공학부

2010년 9월~현재 전남대학교 전자컴퓨터공학과 석사과정

<관심분야> SoC, 영상관련 SoC 및 VDP 설계

김 영 철 (Young-Chul Kim) 정회원

1981년 2월 한양대학교 전자공학과

1987년 2월 Univ. of Detroit 전자공학과 석사

1993년 2월 Michigan state Univ. 전자공학과 박사

1993년~현재 전남대학교 전자컴퓨터공학부 교수

<관심분야> 임베디드 SoC 설계, 저전력 설계, 영상관련 SoC 및 VDP 설계

www.dbpia.co.kr