

블록 효과 감소 알고리즘의 VLSI 회로 구현

김희정, 박성모, 최진호, 김지홍*

부산외국어대학교 컴퓨터전자공학부

*동의대학교 컴퓨터영화공학부

VLSI Circuit Implementation of A Blocking Effect Reduction Algorithm

Hee Jung Kim, Sung Mo Park, Jin Ho Choi, Ji Hong Kim*

Div. of Computer & Electronics Engineering, Pusan University of Foreign Studies.

Div. of Computer & Visual Engineering, Dongeui University

요 약

본 논문에서는 유리 B 스플라인 곡선을 이용한 블록 효과 감소 알고리즘을 VHDL을 이용하여 설계하고, 모의 실험을 통하여 동작을 확인한다. 블록 효과는 매우 낮은 비트율로 블록 기반 부호화 방식을 수행할 때 복원 영상에서 나타나는 블록 형태의 왜곡을 의미한다. 설계된 회로는 유리 B 스플라인 곡선을 적용한 블록 효과 감소 알고리즘으로서, 이 기법은 컴퓨터 그래픽스 분야에서 제어점을 근사하는 부드러운 곡선을 생성하기 위해 사용되는 스플라인 곡선을 적용하여 블록 현상을 효과적으로 감소시킨다. 설계된 회로는 주파수 100MHz에서 동작을 시켰으며, 모의 실험 결과 매우 우수한 블록 효과 감소 기능을 가진 것을 알 수 있다.

1. 서 론

영상 정보의 압축을 위한 블록 기반 부호화 기법은 지역적 상관성이 고려하지 않고 각 블록을 독립적으로 처리하며, 알고리즘의 단순성에 비해 매우 높은 압축 성능을 가진다. 그러나 인접 블록들 간의 경계 영역에서 연속성이 깨어져 복원 영상에서 블록 형태의 왜곡이 일어날 수 있는데, 이러한 현상을 블록 효과(blocking effect)라고 한다[1].

블록 효과 감소를 위한 기법들은 전체 블록 기반 부호화 과정에서 적용되는 위치에 따라 전처리 기법과 후처리 기법으로 분류된다[2]. 전처리 기법

은 영상 부호화 과정에서 적용되는 블록 효과 감소 기법을 의미하며, 후처리 기법은 복원 영상에 적용되는 기법들을 통칭한다.

본 논문에서는 최근 제안된 후처리 기법 중의 하나인 유리 B 스플라인 곡선을 이용한 블록 효과 감소 기법을 VLSI 회로로 구현한다. 회로 설계는 하드웨어 기술 언어인 VHDL을 이용하여 설계하였으며, 모의 실험을 통하여 결과를 확인하였다.

본 논문의 2장에서는 본 논문에서 구현하고자 하는 유리 B 스플라인 곡선을 이용한 블록 효과 감소 기법에 대해 간략히 설명하며, 3장에서는 VLSI 설계 과정을 기술한다. 제 4장에서는 블록 효과를 가진 입력 영상에 대해 설계된 회로를 적용한 모의 실험을

수행하고 성능을 분석한다. 마지막으로 5장의 결론으로 논문을 맺는다

2. 스플라인 곡선을 이용한 블록 효과 감소

본 논문에서 구현하고자 하는 블록 효과 감소 알고리즘은 컴퓨터 그래픽스 분야에서 부드러운 곡선을 생성할 때 사용하는 유리 B 스플라인 곡선을 이용하며, 그 개요는 다음과 같다[3].

유리 B 스플라인은 두 B 스플라인의 비(ratio)를 의미하며,

$$P(u) = \frac{\sum_{k=0}^n w_k P_k B_{k,d}(u)}{\sum_{k=0}^n w_k B_{k,d}(u)} \quad (1)$$

와 같이 정의된다. 식 (1)에서 P_k 는 $n+1$ 개의 제어점을 의미하며, $B_{k,d}(u)$ 는 $d-1$ 차수의 블렌딩 함수(blending function)를 나타낸다. 그리고 파라미터 w_k 는 각 제어점에 대한 가중치이다.

유리 B 스플라인 곡선을 이용한 블록 효과 감소 알고리즘에서는 인접한 블록들의 화소들 중 블록 경계를 기준으로 2 개씩의 화소값을 조정한다. 따라서 제어점의 수를 6 으로 선택하며 곡선 함수의 계산량을 감소시키기 위해 스플라인 곡선의 차수를 3 으로 설정한다. 그리고 제어점에 대한 가중치는 처리될 화소와 블록 경계 간의 거리를 고려하여 차등적으로 설정되도록 한다. 이를 위해 블록 경계에서 동일 위치에 있는 화소값의 절대차(absolute difference)와 거리의 비율을 각 제어점에서의 가중치로 설정한다. 즉, 그림 1 에서 보는 것처럼 제어점 p_0 와 p_5 , p_1 과 p_4 , p_2 와 p_3 이 블록 경계를 기준으로 서로 동일한 위치에 존재하며 각각 5, 3, 1 의 거리를 가지므로, 각 제어점에 대한 가중치는

$$\begin{aligned} w_0 = w_5 &= |p_0 - p_5|/5 \\ w_1 = w_4 &= |p_1 - p_4|/3 \\ w_2 = w_3 &= |p_2 - p_3|/1 \end{aligned} \quad (2)$$

와 같이 설정한다. 그리고 노트 벡터는 {0, 0, 0, 0, 1, 2, 3, 3, 3, 3}로 설정하며, 이에 따라 기준 화

소값 $p_0 \sim p_5$ 가

$$\begin{aligned} p'_0 &= P(0) \\ p'_1 &= P(3/5) \\ p'_2 &= P(6/5) \\ p'_3 &= P(9/5) \\ p'_4 &= P(12/5) \\ p'_5 &= P(3) \end{aligned} \quad (3)$$

로 각각 수정된다.

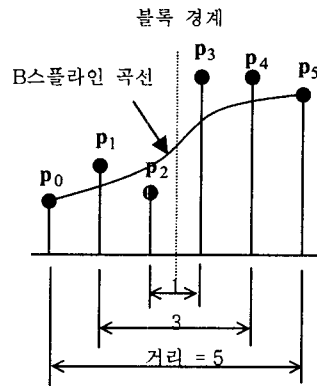


그림 1. B 스플라인 곡선과 가중치 적용 예

3. 회로 구현

제안되어진 알고리즘은 VHDL 하드웨어 설계언어를 이용하여 설계하였다[4]. 그림 2 는 설계된 회로의 블록도이다.

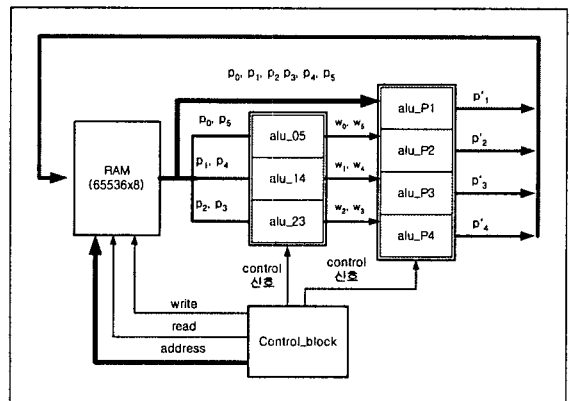


그림 2. 설계된 회로의 블록도
각 회로 블록의 동작 및 신호는 그림 3 에 나타

내었다. 회로의 동작을 살펴보면 256x256 크기의 영상 데이터를 읽어 메모리에 저장한 다음, 블록 경계에서 $p_0, p_1, p_2, p_3, p_4, p_5$ 의 여섯 개의 영상 데이터를 읽는다. 이 데이터는 alu_05, alu_14, alu_23 블록으로 입력되고, 여기서는 수식 (2)를 이용하여 w_0 에서 w_5 까지의 가중치를 연산한다. 그리고 계산되어진 가중치 $w_0, w_1, w_2, w_3, w_4, w_5$ 의 값과 원 영상데이터 $p_0, p_0, p_1, p_2, p_3, p_4, p_5$ 의 값은 수식 (1)에 따라 alu_p1, alu_p2, alu_p3, alu_p4 블록에서 p'_1, p'_2, p'_3, p'_4 를 계산한다. 계산되어진 데이터는 메모리의 해당 번지에 다시 저장되어진다.

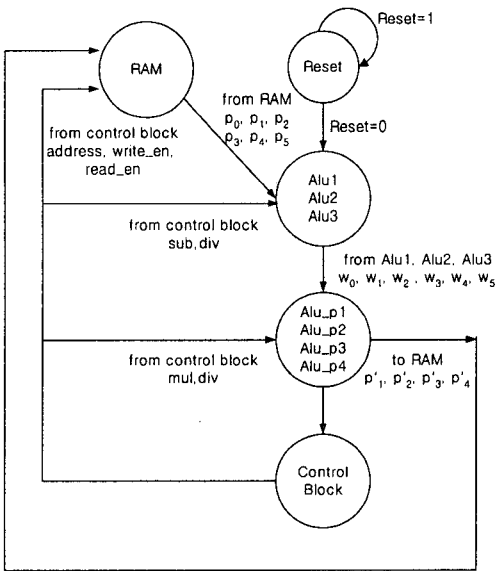
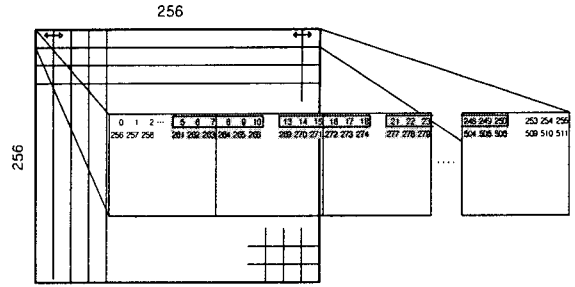
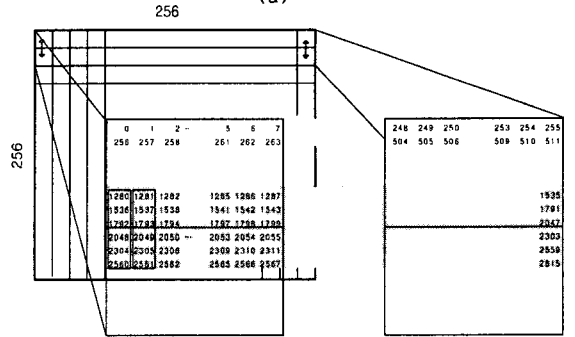


그림 3. 회로 블록의 입력 및 출력 신호

이러한 연산은 블록의 경계 영역에 위치하는 수평과 수직 방향의 모든 화소들에 대하여 적용한다. 그림 4(a)는 수평 방향으로 적용되어지는 순서를 나타낸 것이고 그림 4(b)는 수직방향으로 적용되어지는 예를 나타낸 것이다. 설계되어진 회로에서는 먼저 수평방향으로 블록 효과 감소를 위하여 경계부분에서 데이터를 읽고 연산을 수행한 다음, 메모리에 저장한다. 그리고 수평방향으로의 연산이 완료되면 그림 4(b)에서와 같이 블록 경계면에서 수직방향으로 데이터를 읽은 다음, 연산을 수행하고 저장한다.



(a)



(b)

그림 4. 수직 및 수평 방향의 연산순서
(a) 수평방향의 연산 (b) 수직방향의 연산

그림 5(a)는 설계되어진 회로에서 메모리 주소 그리고 alu_05, alu_14, alu_23의 입력 신호인 블록 경계에서의 영상 데이터와 출력 신호인 가중치의 값을 나타낸 것이다. 그리고 그림 5(b)는 alu_p2, alu_p3, alu_p4에서 유리 B 스프라인 곡선을 이용하여 계산되어진 p'_1, p'_2, p'_3, p'_4 의 출력을 나타낸 것이다.

Address	Data
11000	82000 83000 84000 85000 86000
11001	754 762 778 786 784
11002	110 0110001 0000100 0010101 0000011 1111111
11003	000 0000011 0001010 0000011 0010001 0101010
11004	000 0000000 0001000 0000100 0000110 0011001
11005	0011001 1000110 1000100 0001100 1111111
11006	0101010 1001100 0011100 1100110 0000000
11007	1100001 0101100 1000101 0000011 1111111
11008	0110000 1010010 0000000 0000000
11009	0101100 0011001 1111111
11010	0011010 0000000

(a)

	1066	1074	1082	1090	1098	11
/TB_TOTALA/addr	1106					
/TB_TOTALA/pou1	11011	10010001	10010100	10000001	10110001	10011101
/TB_TOTALA/pou2	10111	01110110	01111110	10010010	01011111	10101100
/TB_TOTALA/pou3	01111	01011110	00111101	10011001	01110101	01011011
/TB_TOTALA/pou4	01001	01011001	00101110	10001101	11011100	01100010

(b)

그림 5. 설계되어진 회로의 출력파형 (a) 가중치 연산 결과 (b) 유리 B 스플라인을 적용하여 연산된 최종 출력 데이터

4. 모의 실험

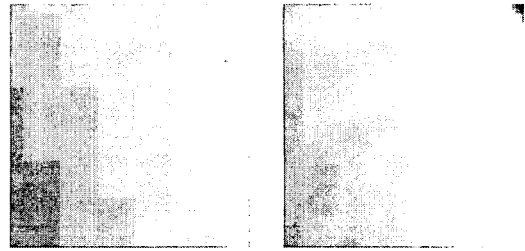
본 논문에서 구현한 VLSI 회로를 시험 영상에 대해 모의 실험하였다. 시험 영상은 JPEG baseline 시스템을 이용하여 압축한 256x256 크기의 Lena 영상으로서 그림 6 (a)와 같다. 그리고 그림 6 (a)의 시험 영상에 대해 본 논문에서 구현한 VLSI 회로를 적용했을 때의 출력 영상을 그림 6 (b)에 나타내었다. 시험 영상과 출력 영상의 어깨 부분에 대한 확대한 영상을 그림 7에 나타내었다. 그림 7 (a)의 블록 효과를 가진 시험 영상과 그림 7 (b)의 출력 영상을 비교할 때, 시험 영상에서 존재하던 블록 효과가 급격히 감소되어 회로가 정상적으로 동작함을 볼 수 있었다.



(a)

(b)

그림 6. 모의 실험의 결과 (a) 시험 영상 (b) 출력 영상



(a)

(b)

그림 7. 어깨 부분의 확대 영상 (a) 시험 영상 (b) 출력 영상

5. 결론

본 논문에서는 유리 개방형 균일 B 스플라인 곡선을 이용한 블록 효과 감소 기법을 VLSI 회로로 구현하였다. 회로 설계는 VHDL을 이용하여 설계하였으며, 모의 실험을 통하여 동작을 확인하였다. 그리고 회로 동작시 주파수는 100MHz를 사용하였다. 회로의 동작 확인 결과 기존 기법들에 비해 소요 계산량이 가장 적을 뿐 아니라, 매우 우수한 블록 효과 감소 기능을 가진 것을 알 수 있다.

[참고문헌]

- [1] Jae S. Lim, Two-Dimensional Signal And Image Processing, Prentice-Hall, Inc., 1990.
- [2] H. C. Reeves and J. S. Lim, Reduction of Blocking Effects in Imaging Coding, Opt. Eng. vol 23, pp.299-302, Feb. 1993.
- [3] 김희정, 김지홍, 유리 개방형 균일 B 스플라인 곡선을 이용한 블록 효과 감소, 한국멀티미디어학회 논문지, 제 5권 4호, pp.386-392, 2002. 8.
- [4] 김영철, 정연모, 조중휘, 홍윤식, 디지털 시스템 설계를 위한 VHDL, 홍릉과학출판사, 1998.
- [5] D. F. Rogers and J. A. Adams, Mathematical Elements for Computer Graphics, McGraw-Hill Publishing Co., 1990.