

논문 2008-45SD-6-10

# 시간 분할 워터마킹 알고리즘의 H.264 적용 및 검증

## ( Application and Verification of Time-Division Watermarking Algorithm in H.264 )

윤진선\*, 최준림\*\*

( Jin Seon Youn and Jun Rim Choi )

### 요약

본 논문에서는 시간분할 워터마크 삽입 알고리즘을 제안하고, 동영상 압축 표준인 H.264/AVC에 적용하였다. 시간분할 워터마크 삽입 알고리즘은 워터마크를 여러 조각으로 나눈 후, 각각의 조각을 여러 프레임의 공간영역에 나눠 삽입하는 것으로, 이러한 삽입 방법을 통해 비가시성과 내성을 가지며, 동영상 인코더의 변경 없이 알고리즘 적용할 수 있는 장점이 있다. 그리고 주파수영역에 워터마크를 삽입하는 기존 방법과는 달리 모든 동영상 압축 표준에 적용 가능한 장점을 가진다. 제안하는 알고리즘의 검증을 위해 워터마크 코어를 Verilog-HDL로 구현 했으며, Excalibur와 H.264/AVC 참조코드를 사용해 하드웨어 및 소프트웨어 통합검증을 수행했다. 검증결과 워터마크가 삽입된 영상과 원본 영상간의 PSNR은 60dB이상 이었으며, 양자화 파라미터 28을 가지는 H.264/AVC 인코딩에서도 워터마크가 80% 이상 유지됨을 확인하였다.

### Abstract

In this paper, we propose watermark algorithm called TDWA(Time-Division Watermarking Algorithm) and we applied the proposed algorithm to H.264 video coding standard. We establish that a proposed algorithm is applied to H.264 baseline profile CODEC. The proposed algorithm inserts a watermark into the spatial domain of several frames. We can easily insert strong and invisible watermarks into original pictures using this method. For verification of the proposed algorithm, we design hardware core using Verilog-HDL and Excalibur for JM 8.7 code with hardware & software co-simulation. As a result of verification, the PSNR between watermarked pictures and original pictures are more than 60dB and we found the watermark is kept more than 80% after encoding of H.264/AVC with quantization parameter of 28 in baseline profile.

**Keywords :** 시간분할 워터마킹(Time-division watermarking), 공간 영역(spatial domain), 워터마킹(watermarking), 저작권 보호(copyright protection)

### I. 서론

디지털 워터마킹은 디지털 콘텐츠의 저작권을 보호하기 위해 콘텐츠에 저작권을 표시해주는 특정 마크(또는 신호)를 삽입하는 기술을 의미한다<sup>[1]</sup>. 디지털 워터마킹의 특성으로는 내성(Robustness) 및 투명성(Transparency)등 여러 가지가 있으며, 워터마크의 삽입 위치에 따라 공간영역(spatial domain) 삽입과 주파수 영역(frequency domain) 삽입으로 구분된다. 추출

방법에 따라서는 블라인드(blind)와 논 블라인드(non-blind) 방법 등으로 크게 분류 된다. 이러한 워터마킹의 특성은 응용되는 곳의 요구에 따라 결정된다<sup>[2]</sup>.

영상의 워터마킹을 위한 대부분의 워터마킹 알고리즘은 공격으로 부터의 강인성을 가지기 위해 주파수영역에 워터마크를 삽입한다<sup>[3~5]</sup>. 그러나 주파수 영역을 기반으로 하는 워터마킹 알고리즘의 경우 워터마크 삽입을 위한 연산이 복잡하며, 이미 하드웨어로 구현되어진 영상 코덱에는 알고리즘의 적용이 불가능한 단점을 가진다. 이에 본 논문에서는 워터마크의 조각인 키 이미지를 여러 프레임의 공간영역에 시간에 나눠 삽입하는 시간분할 워터마킹 알고리즘 (TDWA, Time-Division Watermarking Algorithm)을 제안한다. 제안하

\* 학생회원, \*\* 정회원, 경북대학교 전자공학과  
(Department of Electronics Engineering, Graduate School, Kyungpook National University)  
접수일자: 2007년6월19일, 수정완료일: 2008년5월29일

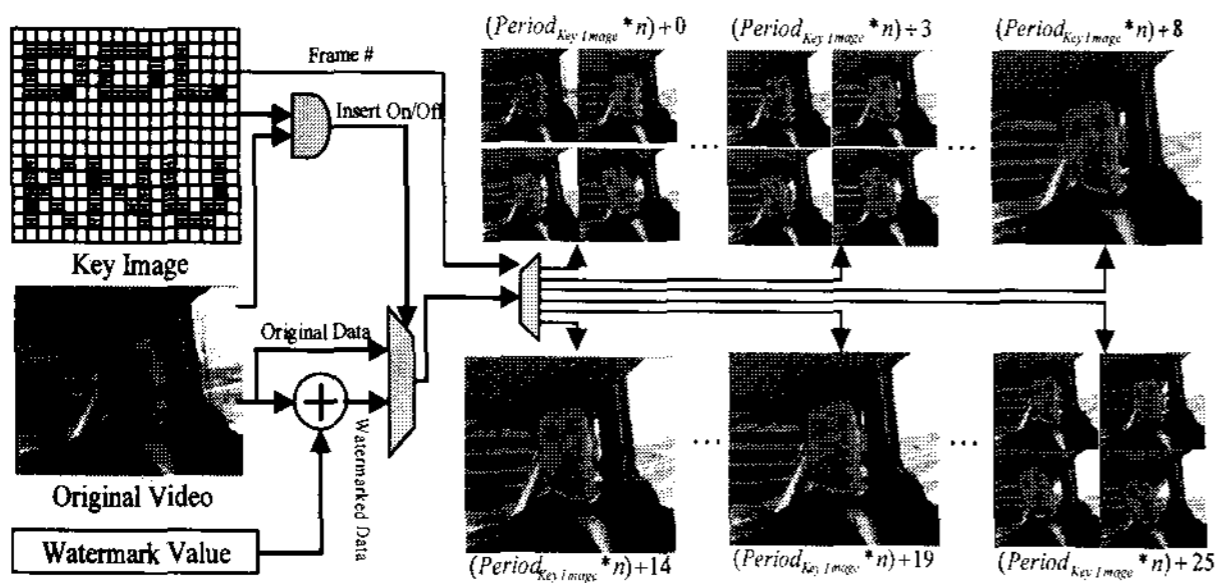


그림 1. 시간분할 워터마킹  
Fig. 1. Time-Division watermarking.

는 알고리즘은 한 장의 워터마크를 여러 장의 프레임에 나눠서 삽입한다. 이를 통해 원본 콘텐츠의 왜곡을 최소화 하는 동시에 공격으로부터 강한 특성을 가진다. 또한 제안하는 알고리즘은 공간영역에 워터마크를 삽입 함으로써 기존 인코더의 변경 없이 영상 콘텐츠에 워터마크를 삽입할 수 있다. 그림 1은 TDWA의 워터마크 삽입 방법을 나타낸다.

본 논문에서 제안하는 워터마크 알고리즘의 검증은 위해 제안한 TDWA를 Verilog-HDL을 통해 하드웨어로 구현했으며, H.264 video coding standard<sup>[6]</sup> (baseline profile, QP=28)의 표준 참조코드인 JM코드를 사용했다. 검증 결과 워터마크가 삽입된 영상과 삽입되지 않은 영상간의 PSNR은 60dB이상으로 측정 되었으며, H.264(baseline profile, QP=28)로 코딩된 영상에서 워터마크가 80% 이상 유지됨을 실험을 통해 확인할 수 있었다.

본 논문은 제 II장에서 시간분할 워터마킹 알고리즘에 대해 제시하며, 제 III장에서는 시간분할 워터마킹 알고리즘의 하드웨어 구현에 대해 설명 한다. 제 IV에서 테스트 및 검증 결과에 대해 논하고, 제 V장에서 결론을 기술한다.

## II. 시간분할 워터마크 삽입 알고리즘

### 2. 시분할 워터마크 삽입 알고리즘의 개념

사람의 눈은 초당 1/17회 정도의 움직임을 인지한다. 이는 빛이 시신경을 통해 뇌로 전달 될 때에 시간적인 제약으로 인한 것이다<sup>[7]</sup>. 따라서 사람의 눈은 움직이는 영상에 1/17초 보다 빠른 시간으로 지나가는 영상을 기존 영상의 사이에 삽입을 해도 그 영상을 보는 사람들은 영상 사이에 삽입된 다른 영상을 인지하지 못한다. 이러한 특성을 광고에 적용할 수 있는데, 이런 광고기법을 잠재의식 광고<sup>[8]</sup>라 한다.

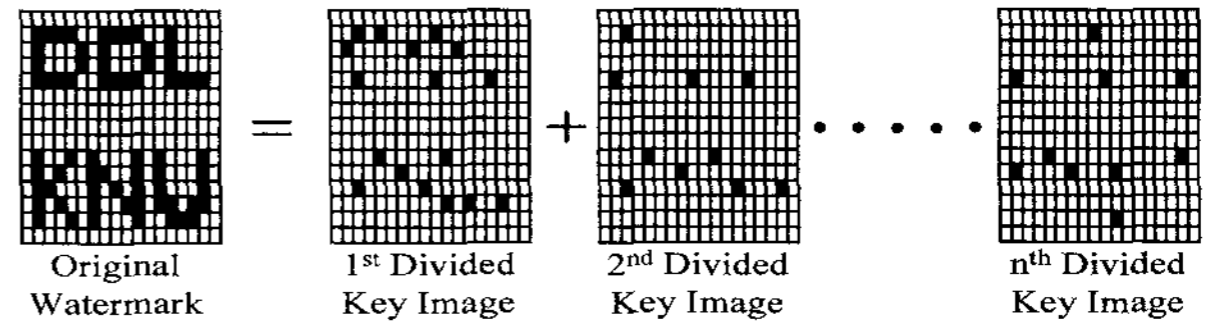


그림 2. 원본 워터마크와 분할된 워터마크  
Fig. 2. Original Watermark and Split Watermarks.

본 논문에서 제안하는 워터마크 삽입기법도 이 원리를 이용하여 워터마크 데이터 중 일부를 1/20 ~ 1/30초 시간주기로 원본영상의 공간영역에 삽입함으로써 원본영상의 왜곡을 줄임과 동시에 영상의 저작권을 보호할 수 있다. 본 논문에서 제안하는 시간 분할 워터마크 삽입 알고리즘의 핵심은 워터마크를 작은 조각으로 분할해 삽입한다는 점과 분할된 워터마크를 여러 프레임의 공간 영역에 삽입한다는 점이다. 그림 1은 TDWA에서의 워터마크 삽입방법을 나타내고 있다.

그림 2는 원본 워터마크와 분할된 워터마크, 즉 키 이미지(Key Image)를 나타낸다. 키 이미지는 정의된 순서대로 프레임의 공간 영역에 삽입되며, 저작권 표시를 위해 키 이미지는 임의로 변경 가능하다.

### 2. 시간분할 워터마크 삽입용 키 이미지 생성

시간분할 워터마크 삽입 알고리즘에서 키 이미지의 생성은 키 이미지가 삽입될 원본 영상의 크기와 초당 재생 빈도(frame/sec)에 따라 달라진다. 각각 키 이미지의 세로 크기와 가로 크기를 구하는 식은 다음과 같이 정의될 수 있다.

$$Key\ Image\ Size_H \leq \frac{Original\ Picture\ Size_H}{One\ Macro\ Block\ Size_H} \quad (1)$$

$$Key\ Image\ Size_V \leq \frac{Original\ Picture\ Size_V}{One\ Macro\ Block\ Size_V} \quad (2)$$

식 (1)과 (2)에서 나타낸 것과 같이, 키 이미지의 크기는 원본 영상을 매크로 블록의 크기로 나눈 값이 된다. 이는 제안한 알고리즘에서 키 이미지의 삽입은 매크로 블록 단위로 이루어지기 때문이다.

키 이미지를 구성하는 다른 요소는 프레임 번호이며, 키 이미지의 조각이 몇 번째 프레임에 삽입되는지를 나타낸다. 본 논문에서는 이 프레임 번호를 기초로, 키 이미지들의 조각을 원본 영상에 주기적으로 삽입한다.

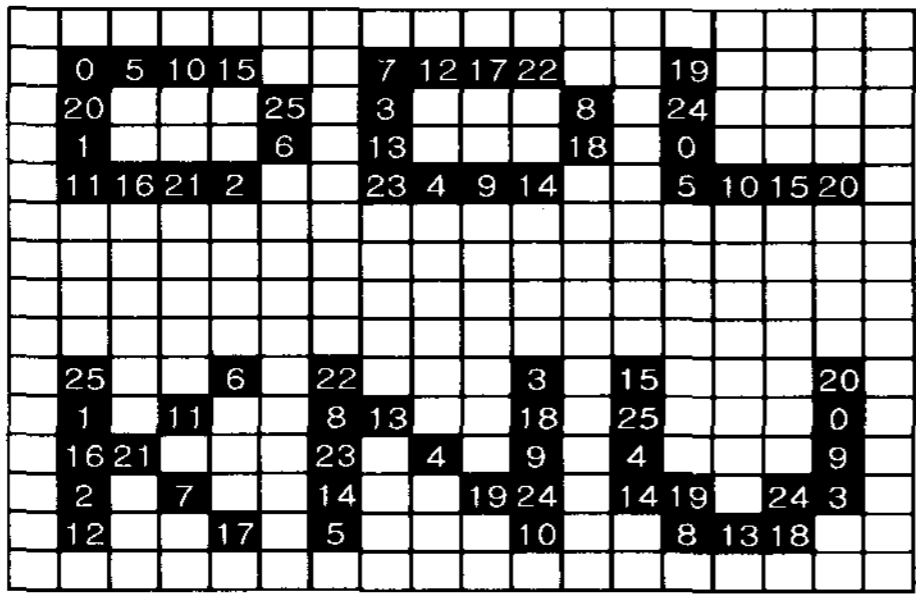


그림 3. 생성된 워터마크 Key Image  
Fig. 3. Key Image of generated watermark.

그림 3은 생성된 키 이미지를 나타낸다. 그림 3에서 검은 블록안의 번호는 키 이미지가 삽입 되어질 프레임을 나타내며, 그림 3의 키 이미지는 26 프레임을 주기로 삽입된다. 그림에서 키 이미지의 크기는 18x15 조각이며, 최소 288x240 픽셀크기의 영상에 삽입 될 수 있다.

### 3. 시간분할 워터마크 삽입 알고리즘의 키 이미지의 삽입

시간분할 워터마크 삽입 알고리즘은 키 이미지의 삽입을 위해 WV(Watermark Value)가 필요하다. WV는 워터마크의 강도를 조절하는 파라미터로 절대 값 4이상의 값을 사용하는데, 이는 WV 값이 절대 값 4 미만일 경우 키 이미지의 대부분이 인코딩 과정에서 손실되기 때문이다. 코드 1은 TDWA의 슈도코드를 나타낸다. TDWA는 모든 처리를 매크로 블록 단위로 처리하며,

```

if(Frame_NBn_piece_of_KEY_IMAGE==Frame_NBORI) {
    if(Position_Yn_piece_of_KEY_IMAGE == Position_YORI)
        if(Position_Xn_piece_of_KEY_IMAGE==Position_XORI)
            for(i=Position_YORI;i<4+Position_YORI;i=i+1)
                for(j=Position_XORI;j<4+Position_XORI ; j=j+1)
                    pixelWatermarked (i, j) = pixelORI(i,j)+WV;
        } else
            for(i=Position_YORI;i<4+Position_YORI;i=i+1)
                for(j=Position_XORI;j<4+Position_XORI;j=j+1)
                    pixelWatermarked (i, j) = pixelORI(i,j);
}
    
```

코드 1. 키 이미지의 입력을 위한 슈도 코드  
Code 1. Pseudocode for inserting key image.

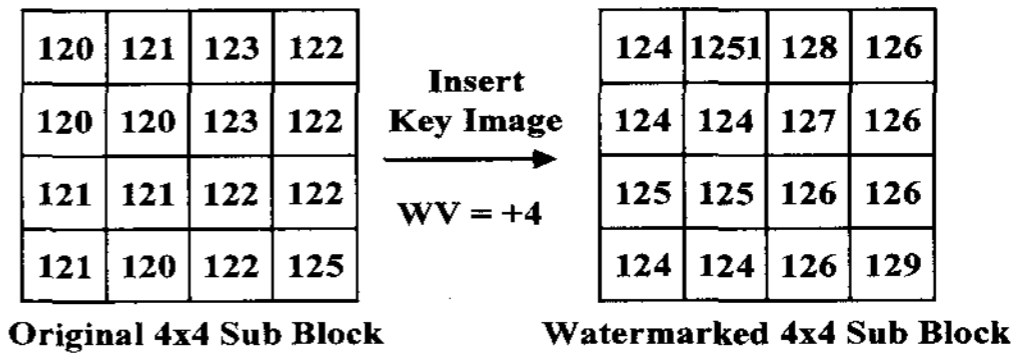


그림 4. 원본 블록과 워터마크된 블록  
Fig. 4. Original block & watermarked block.

동일한 위치와 동일한 프레임 번호를 가진 원본 영상의 블록에 WV 값만큼 더하거나 빼는 방법으로 키 이미지를 삽입 한다. 그림 4는 원본 데이터의 4x4 서브 블록(sub block)과 키 이미지가 삽입된 블록을 나타낸다.

### 4. 시간분할 워터마크 삽입 알고리즘의 키 이미지 추출

제안하는 알고리즘에서 삽입된 키 이미지는 그림 5, 그림 6과 같이 디코딩된 원본 영상과 워터마크가 삽입된 영상간의 차이 값을 통해 추출 된다.

키 이미지가 삽입된 블록(4x4 Sub Block)은 키 이미지가 삽입되지 않은 블록에 비해 WV의 16배에 해당하는 차이 값을 가진다. 이는 키 이미지가 삽입될 조건을 만족하는 매크로 블록들 중 4x4 픽셀 크기의 서브 블록에만 WV 만큼의 값 변화가 발생하기 때문이다. 그러나 영상의 인코딩 과정에서 발생하는 데이터 손실로 인해 원본 영상과 키 이미지가 삽입된 영상 간의 오차가 발

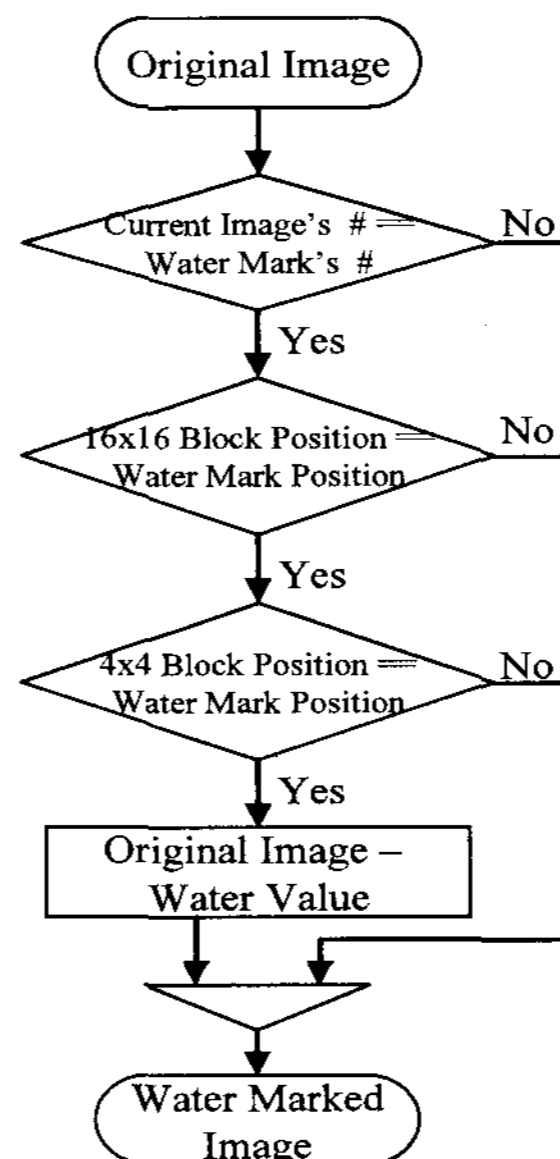


그림 5. 워터마크 삽입을 위한 블록 다이어그램  
Fig. 5. Inserting block diagram.

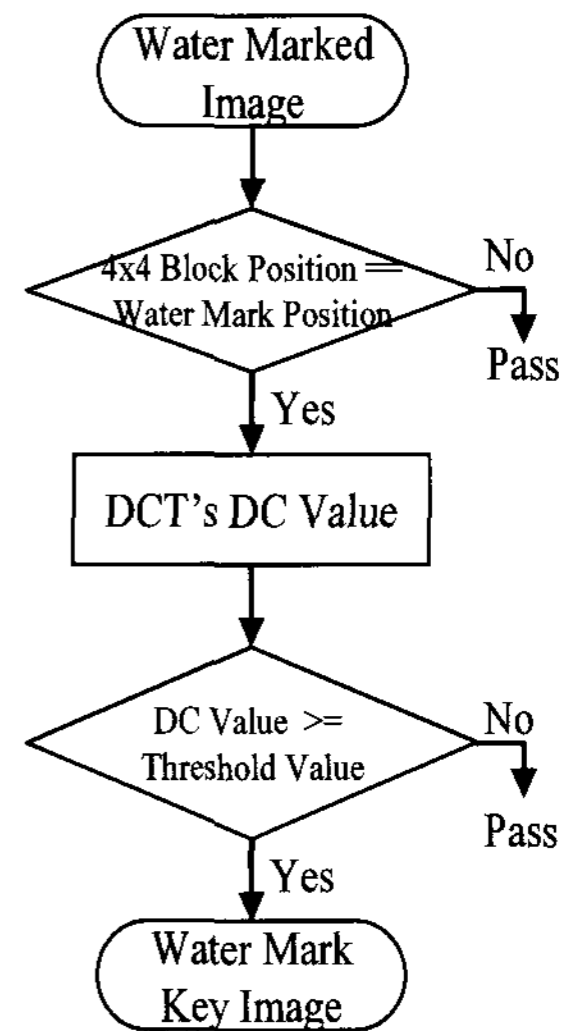


그림 6. 워터마크 추출을 위한 블록 다이어그램  
Fig. 6. Extracting block diagram.

생하게 된다. 이런 데이터 손실로 인해 WV의 16배에 미치지 않는 차이 값이 추출 된다. 그림 6은 키 이미지의 추출을 나타낸 블록 다이어그램이다.

### 5. 키 이미지의 재구성

키 이미지의 추출은, 단지 키 이미지의 조각들은 추출한 것에 불과하다. 따라서 여러 프레임에 나눠 삽입되었던 키 이미지의 조각들을 재구성 해야지만 삽입된 워터마크를 확인할 수 있다. 그림 7은 추출된 키 이미지의 조각과 재구성된 워터마크를 나타낸다.

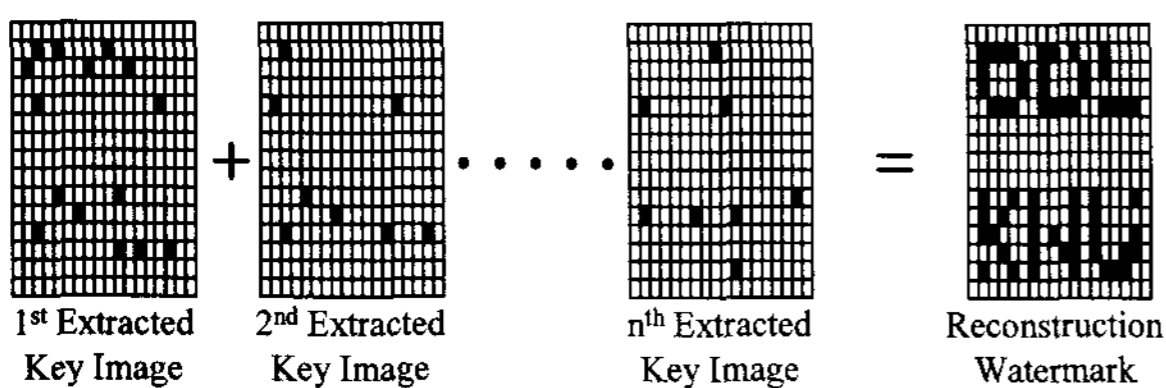


그림 7. 워터마크의 재구성  
Fig. 7. Reconstructing a watermark.

### III. 시분할 워터마크 삽입 알고리즘의 구현

본 논문에서는 제 II장에서 제안한 알고리즘의 검증 및 하드웨어적용을 위해 알고리즘을 FPGA상에서 구현하였다. 알고리즘의 구현에는 Verilog-HDL을 사용했으며, Excalibur가 내장된 SoC Kit 상에서 JM 코드와 연계한 통합 검증을 수행했다.

구현한 시간분할 워터마크 코어는 WM\_COMPWM과 WM\_SUM, WM\_COUNT의 3개의 블록으로 구성했다. WM\_COMPWM는 키 이미지에 대한 정보를 가지며, WM\_SUM은 키 정보에 따라 키 이미지를 삽입한다. 그리고 현재 매크로 블록에 대한 정보를 가지는 WM\_

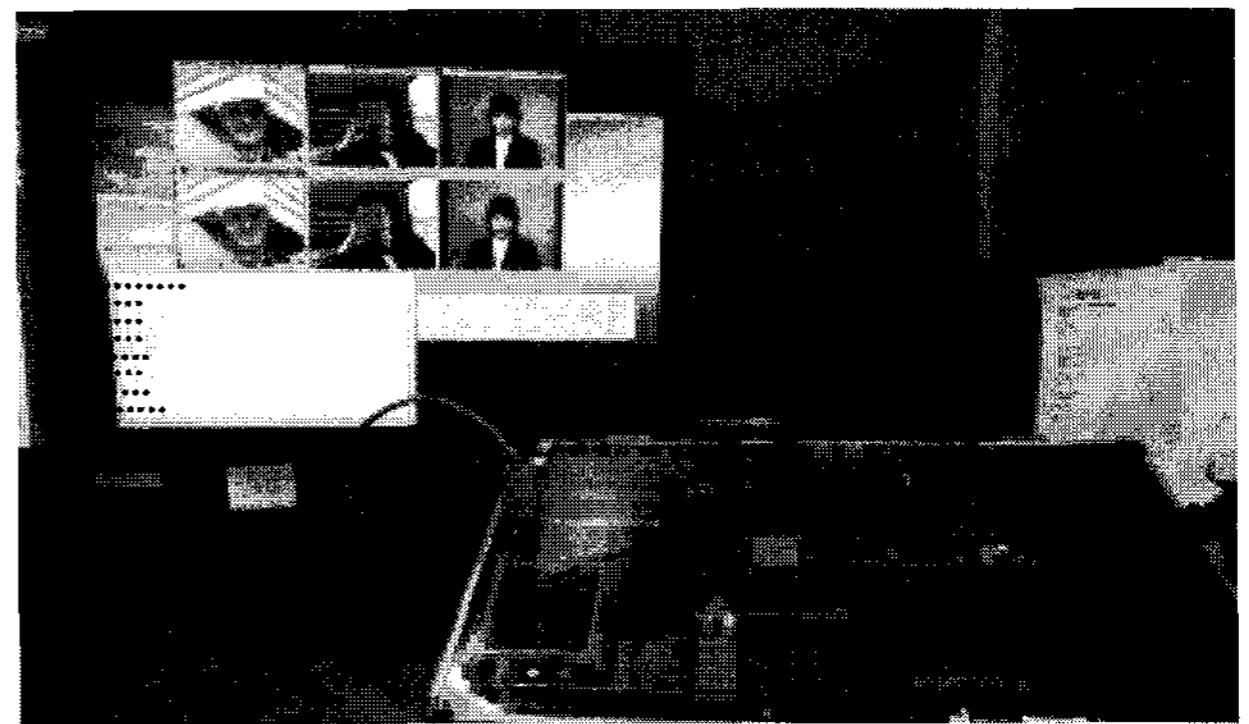


그림 8. SoC Kit과 컴퓨터 상에서의 TDWA 구현  
Fig. 8. Implementation of TDWA on a SoC Kit & Computer.

COUNT로 구성했다.

코어의 최대 동작 주파수는 50.9MHz며, 532 bit의 레지스터와 551개의 Logic Elements를 사용했다. Target chip은 Altera의 Excalibur 40만 게이트(gate)사용했다. 그림 8은 SoC Kit와 PC를 사용해 시간분할 워터마크 삽입 알고리즘의 통합 검증의 모습을 찍은 사진이며 그림 9는 Synplify Pro로 합성한 블록도를 나타낸다.

### IV. 실험 결과 및 고찰

#### 1. 비디오 코딩을 위한 H.264 표준

제안한 시간분할 워터마크 삽입 알고리즘은 매크로 블록 단위로 처리되는 모든 동영상 압축 표준에 적용 가능하다. 본 논문에서는 제안한 알고리즘을 높은 압축률을 가지는 H.264 동영상 압축 표준에 적용했다. 그림 9는 H.264 인코더 블록도를 나타낸다<sup>[9]</sup>. 그림에서 짙게 표시된 부분이 시간분할 워터마크 알고리즘이 적용되는 위치다. 그림에서 알 수 있듯, H.264 인코더로 입력되는 원본 영상에 키 이미지를 바로 삽입한다.

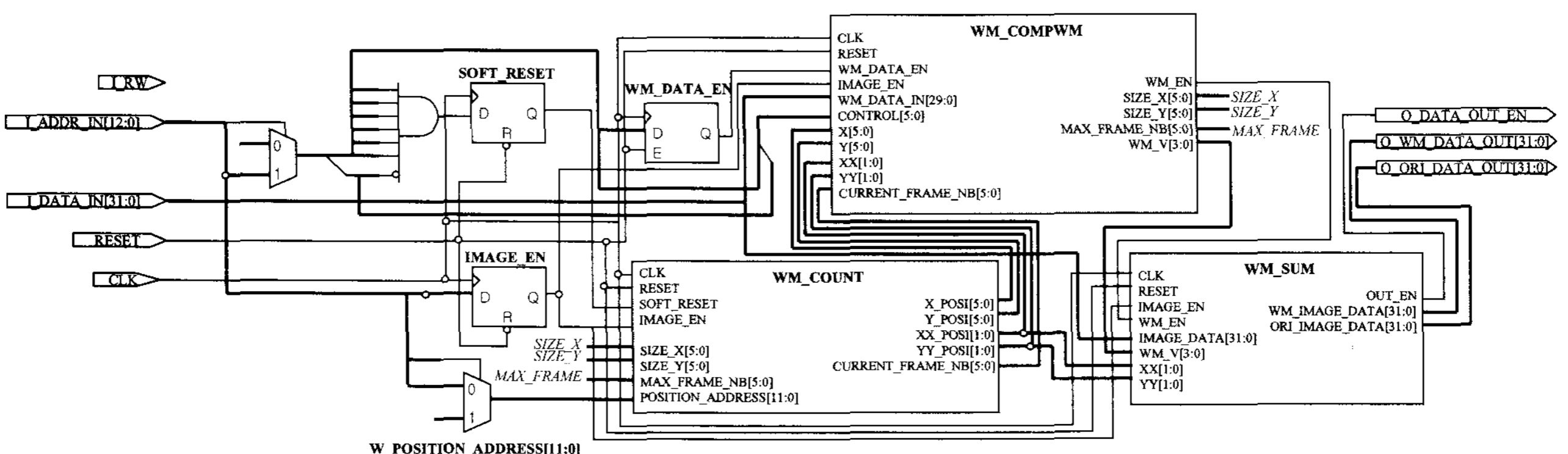


그림 9. Synplify Pro상에서 구현된 TDWA의 하드웨어 블록도.  
Fig. 9. Time-Division Watermark Algorithm Hardware Block Diagram by Synplify Pro.

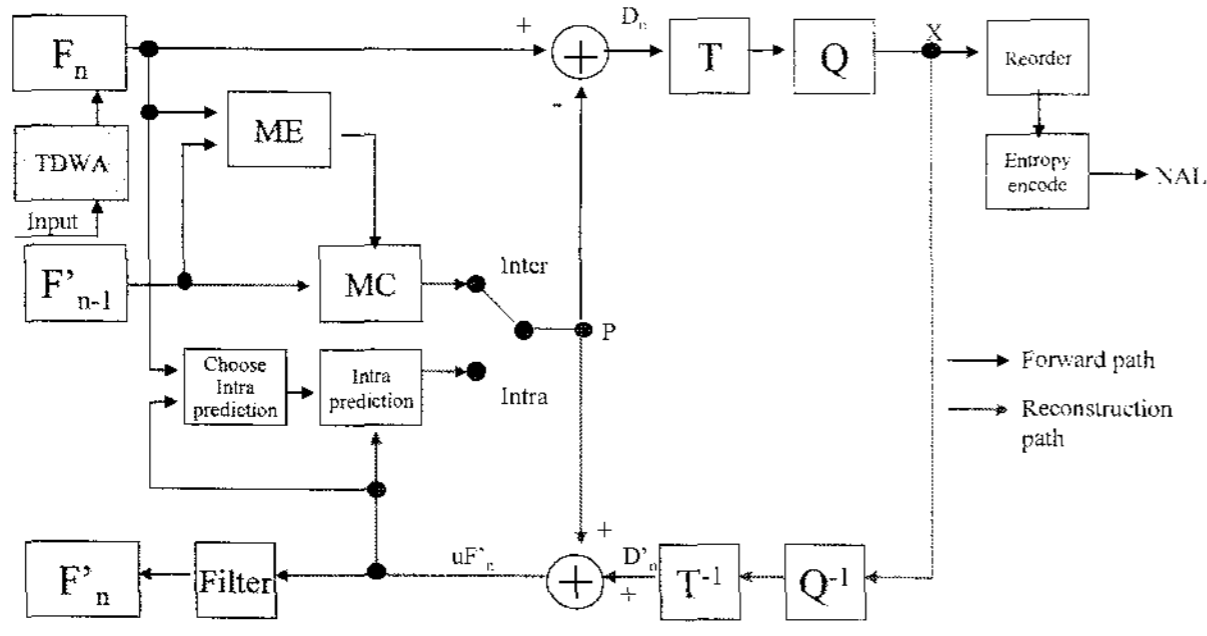


그림 10. H.264 인코더를 위한 블록도  
Fig. 10. Block diagram for H.264 encoder.

2. 테스트 및 검증

테스트는 다음의 순서로 진행 했으며, 테스트를 위해 TDWA 하드웨어 코어와 JM 8.7 코드를 통합 검증 했으며, Excalibur 기반의 SoC Kit상에서 통합 검증하였

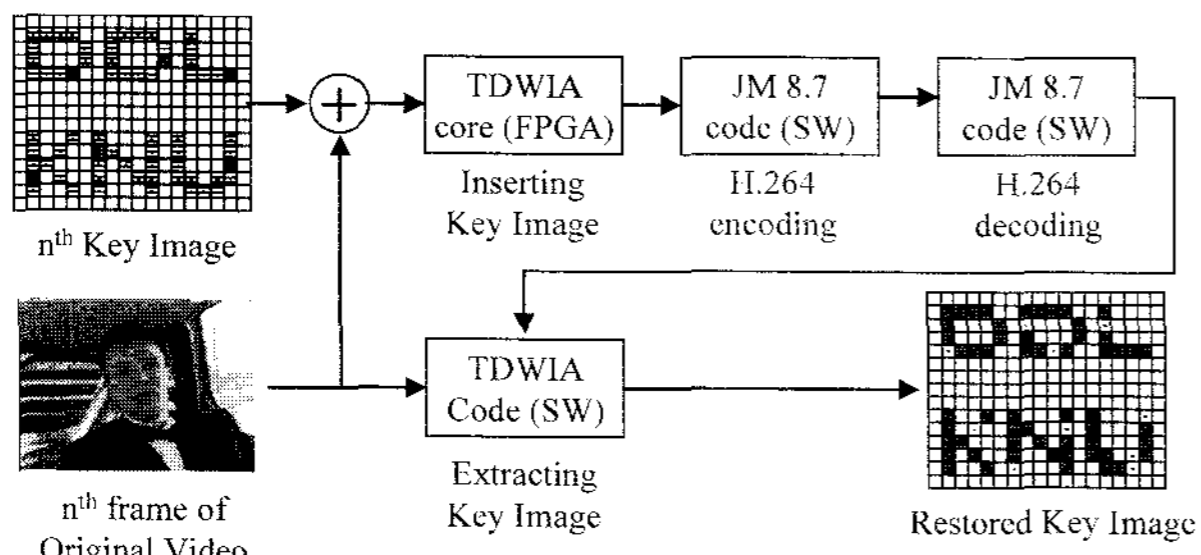


그림 11. TDWA의 테스트 과정  
Fig. 11. Test process of TDWA.



그림 12-1. 원본 비디오      그림 12-2. 워터마크된 비디오  
Fig. 12-1. Original Video.      Fig. 12-2. Watermarked Video.

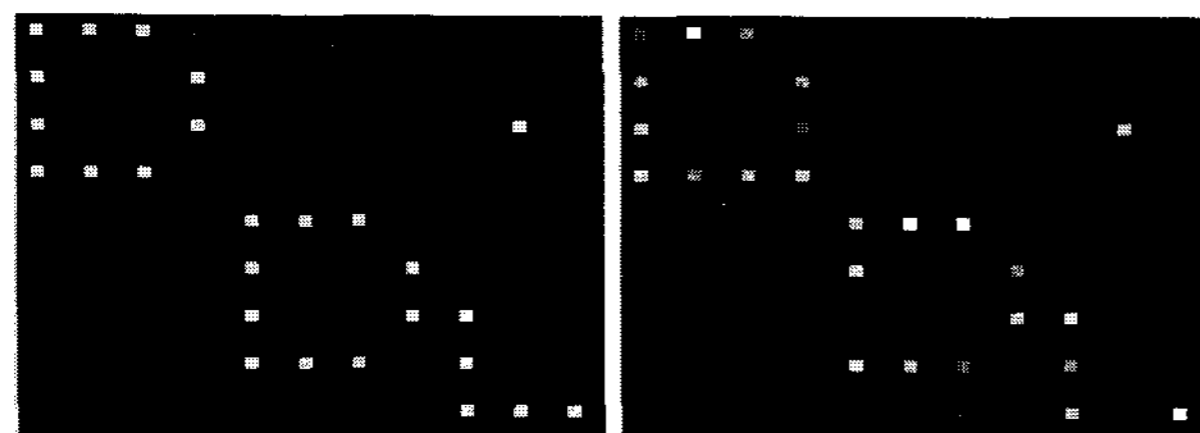


그림 13-1. 원본 키 이미지      그림 13-2. 추출된 키 이미지  
Fig. 13-1. Original Key Image.      Fig. 13-2. Extracted Key Image.

표 1. TDWA의 검증 결과(Baseline profile, qP = 28)  
Table 1. Results of TDWA(Baseline profile, qP = 28).

Encoding of Original Video		Encoding of Watermarked Video	
Image	PSNR (dB)	Image	PSNR WM
	40.71		40.51
	36.67		36.65
	38.04		37.97

다. 그림 11은 검증 과정을 나타낸다.

실험 결과를 그림 12와 13에 나타냈다. 그림 12는 원본 영상과 키 이미지가 삽입된 영상을 각각 나타내며, 그림 13은 원본 키 이미지와 복원된 키 이미지를 나타낸다. 워터마크가 삽입된 영상과 워터마크가 삽입된 영상 사이의 PSNR은 평균 60dB이상 이었으며, 양자화 파라미터가 28인 H.264 동영상 압축 표준으로 인코딩한 영상에서도 워터마크 정보가 80% 이상 유지됨을 실험을 통해 확인할 수 있었다.

H.264의 경우 높은 압축 효율로 인해 비디오 코딩에 사용되던 기존의 워터마킹 알고리즘을 적용할 경우 영상 왜곡 및 워터마크 손실등의 문제가 발생할 수 있다<sup>[10]</sup>. 그러나 본 논문에서 제안하는 워터마크 알고리즘의 경우 H.264로 코딩했을 경우에도 영상 및 워터마크가 유지됨을 확인 하였다. 표 1은 각 영상에 대한 TDWA의 적용 결과를 나타낸다.

V. 결 론

본 논문에서는 영상 콘텐츠의 지적 재산권 보호를 위해 시간분할 워터마크 삽입 알고리즘을 제안하고 H.264에 적용하였다. 제안한 알고리즘은 원본영상의 공간영역에 조각으로 나뉜 워터마크를 시간별로 삽입하는 방식이다. 이를 통해 기존 인코더의 변경 없이 인코더로 입력되는 원본영상의 공간영역에 워터마크를 삽입함

으로 알고리즘의 적용을 간단히 구현할 수 있었다. 또한, 워터마크를 여러 프레임에 나눠 삽입함으로써 워터마크 삽입 위치의 예측이 어렵고, 공간 영역의 일부 위치에 워터마크정보를 집중적으로 삽입함으로써 공격으로부터 강한 특성을 가짐을 확인하였다.

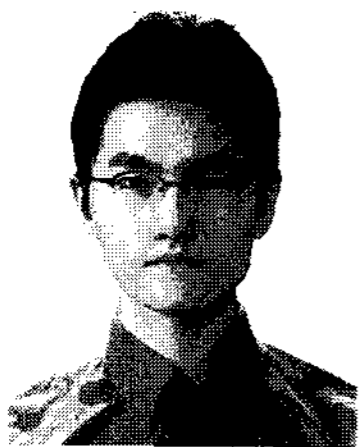
제안한 알고리즘은 Verilog-HDL로 구현했으며, ModelSim 및 Excalibur FPGA를 통해 동작을 확인하였다. 그 결과 양자화 파라미터 28을 가지는 H.264 인코더로 압축된 영상에서도 워터마크가 80% 이상 유지되었으며, 워터마크가 삽입된 영상과 오리지널 영상의 PSNR이 60dB임을 확인하였다. 이를 통해 제안한 워터마크 알고리즘은 원본 영상과 워터마크가 삽입된 영상 간의 영상 열화의 발생이 미미함을 알 수 있었다. 본 논문에서 제안하는 워터마크 알고리즘을 동영상 압축 코덱에 적용함으로써, 동영상 콘텐츠에 비가시성과 견고함의 특성을 가지는 지적 재산권의 삽입이 가능할 것으로 기대된다.

### 참 고 문 헌

[1] I.J. Cox, M.L. Miller, and J.A. Bloom, "Digital Watermarking", Morgan Kaufmann Publishers, San Fransisco, CA, 2001.

[2] 김남득, "디지털 워터마킹 기술소개 및 동향보고", KOSEN/OSTIN, Korea, 2001.  
 [3] J. Y. Park, J. H. Lim, G. S. Kim, C. S. Won, "Invertible Semi-fragile Watermarking Algorithm Distinguishing MPEG-2 Compression from Malicious Manipulation," International Conference on Consumer Electronics, pp. 18-19, June 2002.  
 [4] G. Qiu, P. Marziliano, A. T.S. Ho, "A Hybrid Watermarking Scheme for H.264/AVC Video," Proceedings of the 17th International Conference on Pattern Recognition, pp. 23-26, August 2004.  
 [5] J.J. Chae, K. Khee Pang, "Data Hiding in Video," Proceedings of International Conference on Infotech and Info-net, Vol. 2, pp. 791-796, November 2001.  
 [6] ISO/IEC JTC1/SC29/WG11, "Draft Text of Final Draft International Standard for Advanced Video Coding", DRAFT ISO/IEC 14496-10, 2002.  
 [7] Gi Hung Park, 이론과 실무의 조화 코덱의 세계로의 초대, Hongreung, 2006-08-29.  
 [8] Mikkelson, David P. Subliminal Advertising, 18, 2002 August.  
 [9] Iain E.G Richardson, H.264 and MPEG-4 차세대 영상압축기술, 홍릉과학출판사, 2004년 9월.  
 [10] 김성민, 원치선, H.264/AVC의 4x4 DCT 기반 워터마크에 따른 영상왜곡과 비트율 변화에 대한 연구, 전자공학회, 제42권 제5호, 115쪽, 2005년 9월.

### 저 자 소 개



윤진선(학생회원)  
 2003년 금오공과대학교 물리학과 학사  
 2005년 경북대학교 전자전기컴퓨터학부 석사  
 2005년~현재 경북대학교 전자공학과 박사과정  
 <주관심분야 : 영상압축코덱 SoC 설계>



최준림(정회원)  
 1986년 연세대학교 전기공학과 학사  
 1988년 (미) Cornell대학교 전기전자공학과 석사  
 1991년 (미) Minnesota 대학교 전기전자공학과 박사  
 1991년~1997년 LG전자기술원 책임연구원  
 1997년~현재 경북대학교 전자전기컴퓨터학부 교수  
 <주관심분야 : 영상압축코덱 SoC 설계, Network 정보보호 SoC 설계>