

논문 2007-44SD-11-15

JPEG2000 DWT에 기반한 적응형 블록 워터마킹 구현

(Adaptive Block Watermarking Based on JPEG2000 DWT)

임 세 윤*, 최 준 림**

(Se-Yoon Lim and Jun Rim Choi)

요 약

본 논문에서는 JPEG2000 DWT에 대한 워터마킹 알고리즘들의 화질 저하와 에지 부근의 블록화 문제를 해결하기 위하여 두 개의 스케일링 변수를 사용하여 블록간의 워터마킹 신호가 영상에 따라 자동 조절되는 적응형 블록 워터마킹을 제안하고 검증하였다. 저주파 LL 부대역의 모든 계수들의 평균값과 블록 평균값과의 비를 스케일 변수로 사용하여 1차 강도 조절을 하고, 현재 블록 평균값과 다음 인접 블록 평균값의 비를 2차 스케일링 변수로 사용하여 강도 조절을 함으로써 비가시성과 화질 저하의 문제를 해결하였다. 적응형 블록 워터마킹은 원본 영상에 의해 자동으로 생성되며, 영상의 밝기 값에 따라 워터마크의 강도가 조절된다. 동일한 강도의 워터마크를 삽입하는 대신에 블록 단위별로 서로 다른 강도를 가지는 워터마크를 삽입함으로써 기존 알고리즘들보다 비가시성적인 특성이 4~14dB 향상되었으며 필터 공격, JPEG2000 압축, 리사이즈, 자르기 등과 같은 다양한 워터마크 공격에도 더욱 견고한 특성을 보였다. 또한 제안한 방식을 JPEG2000 칩에도 적용하기 위해 Hynix 0.25 μm CMOS 공정을 사용하여 ASIC으로 구현하여 검증하였다.

Abstract

In this paper, we propose and verify an adaptive block watermarking algorithm based on JPEG2000 DWT, which determines watermarking for the original image by two scaling factors in order to overcome image degradation and blocking problem at the edge. Adaptive block watermarking algorithm uses 2 scaling factors, one is calculated by the ratio of present block average to the next block average, and the other is calculated by the ratio of total LL subband average to each block average. Signals of adaptive block watermark are obtained from an original image by itself and the strength of watermark is automatically controlled by image characters. Instead of conventional methods using identical intensity of a watermark, the proposed method uses adaptive watermark with different intensity controlled by each block. Thus, an adaptive block watermark improves the visibility of images by 4~14dB and it is robust against attacks such as filtering, JPEG2000 compression, resizing and cropping. Also we implemented the algorithm in ASIC using Hynix 0.25 μm CMOS technology to integrate it in JPEG2000 codec chip.

Keywords: JPEG2000, DWT(Discrete Wavelet Transform), 워터마킹(Watermarking)

I. 서 론

오늘날 모든 데이터들은 디지털로 처리되고 있으며,

디지털의 특성상 화질의 열화 없이 무단 복제, 도용, 변형, 불법 배포가 가능하여 저작권을 가진 소유자에게 큰 피해를 줄 수 있다. 디지털 데이터에 대한 소유권을 효율적으로 보호하기 위한 해결책으로 워터마킹(watermarking) 기술이 제시 되었으며, 이 기술의 특징은 정보 보호 목적에 따라 여러 가지 방법으로 삽입이 가능하다는데 있다.

워터마킹은 워터마크의 삽입 영역에 따라 공간 영역, 주파수 영역 삽입 방식으로 분류되는데, 공간 영역 워터마크 삽입 방식은 초기적인 형태의 워터마킹 시스템으로 쉽게 구현 될 수 있는 장점이 있지만 일반적인 필

* 학생회원, ** 정회원, 경북대학교 전자공학과, WRG 연구원

(Department of Electronics Engineering, Graduate School, Kyungpook National University)

※ 본 논문은 정부통신부 출연금으로 ETRI SoC 산업진흥센터에서 수행한 IT-SoC 핵심설계인력양성사업의 연구 결과이며 IDEC의 tool 지원에 의해 결과를 검증하였습니다.

접수일자: 2007년5월3일, 수정완료일: 2007년11월1일

터와 같은 신호 처리나 압축 등의 변형 공격에 쉽게 깨지는 특성을 가지기 때문에 주파수 영역 삽입 방식으로 바뀌고 있는 추세이다. 주파수 영역 삽입 방식을 사용하면 견고한(robust) 특성을 갖게 되며, HVS(Human Visual System) 특성을 고려하여 삽입하면 가시성 면에서도 우수한 성능을 나타낸다. 주파수 영역 삽입 방식은 크게 DCT(Discrete Cosine Transform)를 이용한 방식과 DWT(Discrete Wavelet Transform)를 이용한 방식으로 나뉘어진다. 정지 영상의 경우 더 좋은 화질과 압축률을 얻기 위해 DCT에서 DWT기반의 설계 방식으로 옮겨가고 있으며, 차세대 정지영상 알고리즘인 JPEG2000에서도 DWT를 사용함에 따라 이에 대응하는 워터마킹의 연구도 점차 늘어가는 추세이다. 그러나 기존의 DWT를 기반으로 하는 워터마크 알고리즘들^[1~3]은 이미지에 대해 동일한 워터마크를 삽입하여 비가시적인 측면에서 성능 저하가 있었다.

본 논문에서는 이와 같은 화질저하의 문제점을 해결하기 위하여 JPEG2000에서 사용될 수 있는 적응형 블록 워터마킹을 제안하고 구현하였다. JPEG2000 DWT를 사용하여 원본 이미지를 레벨1로 분해한 후 LL 부대역내의 모든 계수들을 블록 단위로 나누어서 블록별로 연산을 수행하였다. 영상의 밝기 성분의 상관관계를 이용하여 두 개의 스케일링 변수를 생성하고 이를 사용하여 워터마크의 강도를 조절하였다. 동일한 강도의 워터마크를 삽입하는 대신에 블록 단위별로 다른 강도를 가지는 워터마크를 삽입함으로써 기존 알고리즘들보다 비가시성과 견고성을 향상시킬 수 있었다. 또한 이를 ASIC으로 구현하여 JPEG2000 상에서 검증하였다.

제 II장에서는 JPEG2000 DWT개요와 적응형 워터마킹 방식에 대해 설명하였고, 제 III장에서는 앞장에서 설명한 알고리즘을 하드웨어로 설계한 구조를 설명하였다. 제 IV장에서는 제안한 방식을 기존의 방식과 비교하고, ASIC으로 구현한 결과를 설명하였으며 제 V장에서 결론을 기술하였다.

II. 적응형 블록 워터마킹 알고리즘

1. JPEG2000 DWT개요

DWT는 이미지를 고주파수 성분과 저주파수 성분으로 분해하는 영상의 주파수 변환 과정이다. JPEG2000에서는 표에 있는 Daubechies 계수 값을 사용하는 고역필터와 저역필터를 사용하여 DWT를 수행한다. 각각의 필터를 통과한 영상 데이터는 신호의 각 두 샘플들

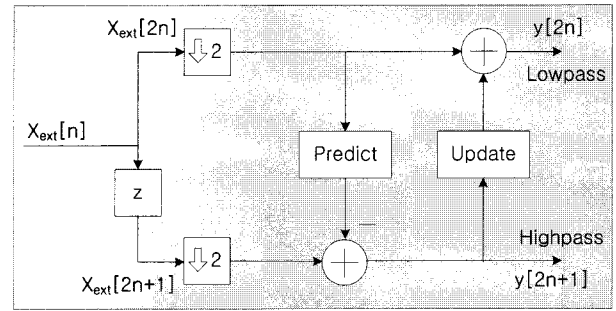


그림 1. 리프팅을 이용한 웨이블릿 변환
Fig. 1. The forward wavelet transform using lifting scheme.

중 하나를 제거하는 2다운 샘플링을 거쳐 주파수 대역별 영상의 크기가 가로 세로 각각 반씩 줄어들게 된다.

DWT를 구현하는 방법은 두 가지가 있다. 하나는 컨벌루션 방식이고 또 한 가지 방식으로 1995년에 제안된 리프팅 방식^[7]이 있다. 그림 1과 같이 predict, update, scaling 세 과정을 통해 원 신호를 다시 읽지 않고 이웃 샘플의 결과 값을 사용하여 계산 하는 방법으로 컨벌루션 방식보다 계산량이 절반으로 줄어들며, 하드웨어 구성에도 용이하다.

식 (1)은 필터의 분해와 합성을 위해 각각 5개와 3개의 계수를 사용하는 5/3 필터에 사용되는 predict 단계를 나타낸다.

$$Y(2n+1) = X_{ext}(2n+1) - \left[\frac{X_{ext}(2n) + X_{ext}(2n+2)}{2} \right] \quad (1)$$

업데이트(update) 단계는 predict 단계에서 연산된 홀수 번째 신호를 이용하여 평균값을 구하고, 그것을 짝수 번째 신호에 더해서 짝수 번째 신호를 업데이트 시킨다. 식 (2)는 5/3필터에 사용되는 업데이트(uptdate) 단계를 나타낸다.

$$Y(2n) = X_{ext}(2n) + \left[\frac{Y(2n-1) + Y(2n+1) + 2}{4} \right] \quad (2)$$

DWT의 필터는 사용하는 필터 계수에 따라 두 가지로 나뉘어진다. JPEG2000에서 손실 압축에 사용하는 필터 연산의 분해와 합성을 위해 각각 9개와 7개의 계수를 사용하는 9/7필터와 무 손실 압축에 사용하는 5/3필터가 있다.^[4~6]

DWT 연산을 수행할 때, 현재 픽셀의 좌우로 이웃한 픽셀이 9/7필터의 경우 4개, 5/3필터의 경우 2개씩 존재해야 필터연산이 가능하다. 따라서 영상의 가장자리에



그림 2. 리프팅 기반의 9/7 필터 테스트 예
Fig. 2. Lifting-based 9/7 filter test example.

서 이웃하는 픽셀의 수가 부족하거나, 존재하지 않을 경우 점대칭 확장과 주기적 대칭확장 두 가지 방식 중의 하나로 신호를 확장하여 필터 연산을 수행한다. 그림 2는 128×128 화소 크기를 가지는 Lena 영상에 적용한 예이다.

영상에 대해 DWT를 수행하면 그림 2의 세 번째와 같이 주파수 성분에 따라 LL, LH, HL, HH로 표기되는 4개의 영역으로 분리 되는데, 첫 번째 문자는 행 방향에 대한 필터링을, 두 번째 문자는 열 방향에 대한 필터링을 나타내며, L은 저역 필터, H는 고역필터를 나타낸다. 왼쪽 상단의 영역을 LL 부대역이라 하며, 행과 열 방향 모두 저역 필터링을 통과한 저주파 성분들의 집합으로 영상의 가장 중요한 성분들의 집합이다. LH 부대역은 수평 방향의 Edge성분들의 집합이며, HL은 수직 방향, HH는 대각선 방향 Edge성분들의 집합이다.

2. JPEG2000 DWT영역에 워터마크 삽입

워터마킹은 마크의 삽입으로 인한 화질의 저하가 적어야 하며, 삽입된 워터마크가 보이지 않는 비가시성과 다양한 공격에도 견고한 특성을 지녀야한다. 그러나 견고성과 비가시성은 동시에 만족 될 수 없으며, trade off 관계를 가진다. 즉, 견고성을 위해 워터마크의 강도를 높이면 화질의 열화를 가져오게 된다. 기존의 DWT에 대한 워터마킹 알고리즘들은 고주파 대역의 신호가 인간에 미치는 시각화만을 고려하거나^[1] 동일한 강도를 가지는 워터마크를 삽입함으로써^[2-3] 비가시성은 물론 화질이 저하가 불가피 하였다.

본 논문에서는 이러한 문제점과 이미지의 에지(edge) 부근의 블록화 문제를 해결하기 위해서 두 개의 스케일링 변수(scaling factor)를 사용하여 블록간의 워터마크 신호가 영상에 따라 자동 조절되는 적응형 워터마크의 삽입을 제안한다. 삽입된 워터마크 정보는 추출 시스템에서 원본 이미지를 통해 재생될 수 있기 때문에 따로 저장, 보관할 필요가 없다. 그림 3과 같이 LL 부대역

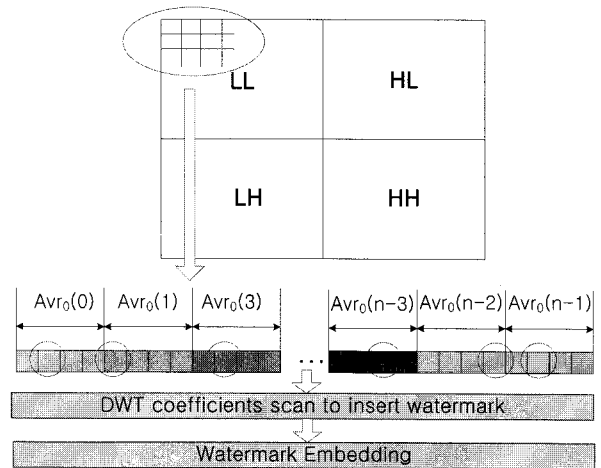


그림 3. 워터마킹 삽입 알고리즘
Fig. 3. Watermarking embedding algorithm.

계수인 $a_{LL}(u,v)$ 를 행 방향으로 1 블록 당 4개의 샘플이 들어가도록 한다. LL 부대역 내에 있는 모든 행에 대해서 같은 방법을 사용하여 블록 단위로 나눈다.

첫 번째 행을 예로 들면, $a_{LL}(0,0)$, $a_{LL}(0,1)$, $a_{LL}(0,2)$, $a_{LL}(0,3)$ 을 1개의 블록으로 설정하고, 4개의 샘플에 대한 평균값을 $Arv_0(0)$ 라 한다. 다음 블록은 $a_{LL}(0,4)$, $a_{LL}(0,5)$, $a_{LL}(0,6)$, $a_{LL}(0,7)$ 를 샘플로 가지며, 블록 평균값은 $Arv_0(1)$ 이 된다.

첫 번째 스케일링 변수는 식 (3)과 같이 정의되며, m행 n번째 블록과 다음 블록간의 상관관계를 나타낸다.

$$\beta_m(n) = \frac{Arv_m(n+1)}{Arv_m(n)} \tag{3}$$

블록 자체 내에 에지가 나타나지 않거나 블록과 블록 사이에 에지(edge)가 나타나지 않으면 $\beta_m \approx 1$ 의 값을 가진다. 어느 한쪽의 블록 평균값이 크게 되면 β_m 은 1보다 크거나 작은 값을 가지게 되어 블록간의 관계에 따라 워터마크의 강도가 조절되어진다.

두 번째 스케일링 변수는 식 (4)와 같이 정의되며, 전체 LL 부대역내의 계수 평균값에 대한 m행 n 블록 평균값을 나타낸다. 분모 항은 LL 부대역내의 모든 샘플에 대한 평균값이다.

$$\alpha_m(n) = \frac{Arv_m(n)}{\sum_{m=0}^{LL} \sum_{n=0}^M Arv_m(n)} \tag{4}$$

LL 부대역내의 전체 계수 평균값을 기준으로 워터마크의 신호 세기를 1차 조절하여 블록에 DC 바이어스(bias)를 걸어주고 β_m 을 사용하여 블록간의 상관관계를

기준으로 워터마크의 신호를 한 번 더 조절한다. β_m 은 인접블록과 비슷한 색상을 지닐 경우에는 1에 가깝게 되며 인접블록과 색상차가 급격히 차이가 나게 되면 0에 가까운 값을 가지게 된다. 새로 제안된 워터마킹 알고리즘은 식 (5)와 같다.

$$Y'_m(n) = Y_m(n) \cdot [1 + \alpha_m(n) \cdot w_m(n)] \cdot [1 + \beta_m(n) \cdot w_m(n)] \quad (5)$$

여기에서 $Y_m(n)$ 은 m행 n번째 블록에 있는 4개의 계수 중에서 워터마크가 삽입될 신호이며 매 블록마다 한 개의 워터마크가 삽입된다. 영상 픽셀의 비트수가 8이라고 가정하고, 인간의 시각은 밝기가 강한 곳의 변화에 상대적으로 둔감하므로, 밝기 성분 값이 큰 위치 $Y_m(n)$ 을 선정하는 방법은 아래와 같다.

- m행 n번째 블록 평균값인 $Arv_m(n)$ 을 구한다.
- $Arv_m(n) \leq Y_m(n) < 255$ 을 만족하는 계수 중에서 가장 큰 $Y_m(n)$ 을 선택한다.
- $Arv_m(n) = 255$ 일 경우 m행 n번째 블록의 워터마크는 삽입하지 않는다.
- $Arv_m(n) = 0$ 일 경우 m행 n번째 블록의 워터마크는 식 (5)에 관계없이 1을 삽입하며 블록의 첫 번째 신호에 삽입한다.

$w_m(n)$ 은 워터마크 신호 강도이고 $Y'_m(n)$ 은 워터마크가 삽입된 신호이다. $w_m(n)$ 은 0과 0.1사이의 값을 가지며 $w_m(n)$ 의 값이 크면 클수록 견고해진다. 그러나 너무 큰 값을 선택하게 되면 비가시적인 특성이 나빠진다. 따라서 본 논문에서는 견고성을 유지하면서 비가시적인 특성이 높은 $w_m(n) = 0.01$ 의 값을 택하여 실험하였다. 워터마크 신호는 알고리즘에 의해 원본 이미지로부터 자동생성 된다.

3. JPEG2000 DWT영역에서 워터마크 추출

워터마킹을 인지하는 방법은 크게 추출 방식과 감지 방식으로 나뉘어 진다. 추출 방법의 경우에는 워터마크를 추출하기 위해 원본 이미지를 필요로 하며, 감지 방식의 경우에는 암호화된 키를 삽입하고 이를 검출하기 때문에 원본 이미지는 필요 없고 삽입된 키의 정보만 알면 감지가 가능하다.

본 논문에서는 추출방식을 이용하여 워터마킹을 판

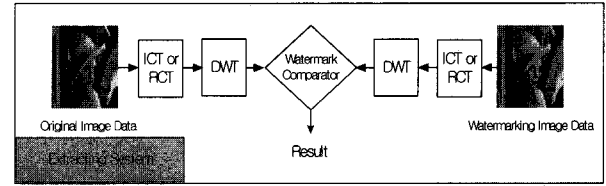


그림 4. JPEG2000 워터마킹 추출 구조

Fig. 4. JPEG2000 watermarking extraction structure.

별한다. 그림 4와 같이 먼저 원본 이미지 데이터를 ICT(irreversible component transform)^[4] 또는 RCT(reversible component transform)^[4] 변환 한다. ICT의 경우에는 Y, Cb, Cr 중에서 Y 성분에 대해 9/7 필터 DWT를 수행하고, RCT의 경우에는 식 (3)의 Y_0, Y_1, Y_2 성분 중에서 Y_0 성분에 대해서 5/3 필터 DWT를 수행한다.

같은 방법으로 워터마킹 된 이미지를 ICT 또는 RCT 변환하고 DWT를 취한다. 워터마킹 된 이미지로부터 구한 DWT 계수 값에서 원본 이미지로부터 구한 DWT 계수 값을 빼준다. 두 결과의 차로 구한 값이 워터마크 신호가 되며 원래 삽입한 워터마크 정보와 같은지를 비교하게 된다. 9/7 필터의 경우에는 이미지내에 포함된 워터마크의 정보도 일부 손실이 되므로 100% 정확히 일치하지 않는다. 그레이(gray) 스케일 에서는 ICT, RCT 과정을 거치지 않으므로 컬러 이미지보다 더 좋은 결과를 보인다.

영상 압축알고리즘의 성능 및 이미지의 질을 평가하기 위해서 BER, SNR, PSNR, MSE등이 주로 사용되며, 이미지의 정확성을 측정하기 위해서 PSNR(peak signal to noise ratio)을 주로 사용하고 있는 추세이다. 본 논문에서도 PSNR을 사용하여 이미지의 정확성을 평가하였다. 식 (6)에서 $I_{i,j}$ 는 원본 신호를 나타내고 $I'_{i,j}$ 는 워터마크가 삽입된 신호를 나타낸다. N^2 은 이미지의 화소수를 나타낸다.

$$PSNR = 20 \times \log_{10} \left(\frac{255}{\sqrt{\frac{\sum (I_{i,j} - I'_{i,j})^2}{N^2}}} \right) \quad (6)$$

다양한 워터마킹 공격에 대한 견고성을 평가하기 위해서 본 논문에서는 식 (7)에 나타난 정규화된 유사도를 사용한다.

$$Z'(X, X^*) = \left(\frac{X \cdot X^*}{\sqrt{X^* \cdot X^*}} / \frac{X \cdot X}{\sqrt{X \cdot X}} \right) \times 100 \quad (7)$$

X는 원래의 워터마크이고, X*는 검출된 워터마크를 나타내며 < · >는 벡터의 내적을 나타낸다. 추출된 워터마크와 원본에 삽입한 워터마크가 같을 경우 위 식은 100%를 나타낸다.

III. 적응형 블록 워터마킹의 ASIC 구현

제 II장의 내용을 바탕으로 이를 JPEG2000 칩에도 적용할 수 있도록 ASIC으로 구현하였다. 실제 워터마킹 연산을 수행하는 부분과 입출력 인터페이스를 포함하여 크게 세 부분으로 구성하였다.

그림 5는 워터마킹 연산부 하드웨어 이다. 동작은 4개의 계수를 입력 받아 16bit 비교기(comparator)를 통해 워터마크가 삽입될 신호를 추출하고 블록의 평균값을 구한다. 외부에서 입력되는 LL부대역 평균의 역수와 곱셈을 하여 블록의 평균값과 전체 LL부대역의 평균값과의 비율(α)를 구한다. 위의 두 가지 과정이 완료되면 워터마크 수식에 따라 워터마크를 삽입한다.

워터마킹 하드웨어는 32비트의 입력을 받도록 구성하여, 한 번에 두개의 DWT계수를 입력받을 수 있다. 입력 인터페이스의 구성은 그림 6에 나타내었다. 제일 처음으로 LL부대역의 전체 평균값의 역수가 전달되며, 이후 DIN 신호에 따라 두개의 계수가 전달된다.

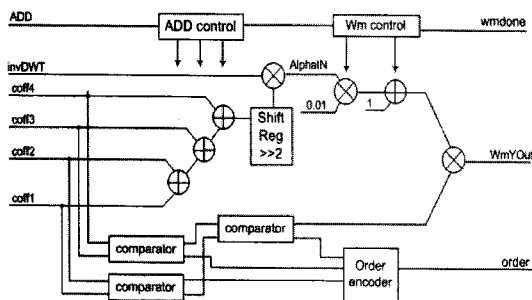


그림 5. 워터마킹 하드웨어 블록도
Fig. 5. Hardware block diagram of watermarking.

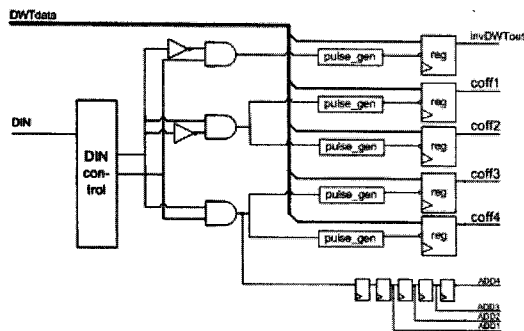


그림 6. 입력 인터페이스 블록도
Fig. 6. Input interface block diagram.

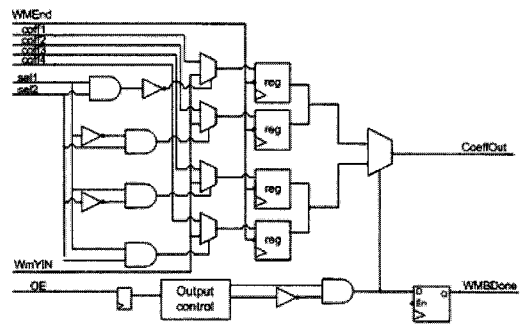


그림 7. 출력 인터페이스 블록도
Fig. 7. Output interface block diagram.

그림 7은 워터마크가 삽입된 후 블록의 계수들을 출력하는 출력 인터페이스 부분을 나타낸다. 4개의 계수 값과 워터마크가 삽입된 계수 값을 전달 받아 몇 번째 계수에 워터마크가 삽입되었는지 알리는 sel1, sel2신호에 따라 워터마크가 삽입된 계수가 데이터 선택기에서 선택 되어 원래의 계수 값 대신 출력되게 된다. 4개의 계수는 총 64비트 이므로 출력가능 신호인 OE를 통해서 첫 번째 계수부터 두 개씩 차례로 출력 된다.

IV. 실험 결과 및 고찰

본 논문에서 제시한 적응형 블록 워터마킹은 PC 환경에서 C언어와 매트랩(matlab)을 사용하여 검증하였다. 실험 이미지로 256×256 화소 크기를 가지는 Lena, Goldhill, Baboon을 사용하였다. 총 4096개의 워터마크를 LL 부대역에 삽입하여 견고성을 높였으며 두 개의 스케일링 변수를 사용하여 이미지의 비가시성을 높였다. 삽입 워터마크의 개수는 고정된 것이 아니라 이미지의 크기에 비례하여 증가하게 된다. 소프트웨어로 검증한 알고리즘은 Verilog-HDL을 사용하여 하드웨어로 설계하였으며, Hynix 0.25 μ m CMOS 공정 라이브러리를 사용하여 ASIC으로 구현하여 테스트 하였다.

1. 비가시성 평가

본 논문에서 제시된 적응형 블록 워터마킹은 9/7 필터와 5/3 필터에 대해서 테스트 하였고 비가시성에 대한 평가 척도로 식 (6)을 사용하였다. 리프팅 기반의 9/7 필터와 5/3 필터에 대한 원본 이미지와 워터마크 이미지의 테스트 결과를 그림 8에 나타내었다.

본 논문에서 제안하는 적응형 블록 워터마킹을 사용할 경우 표 1에서 나타낸 바와 같이, 단일 이미지에 대해 동일한 워터마크 강도를 적용하는 기존 알고리즘과

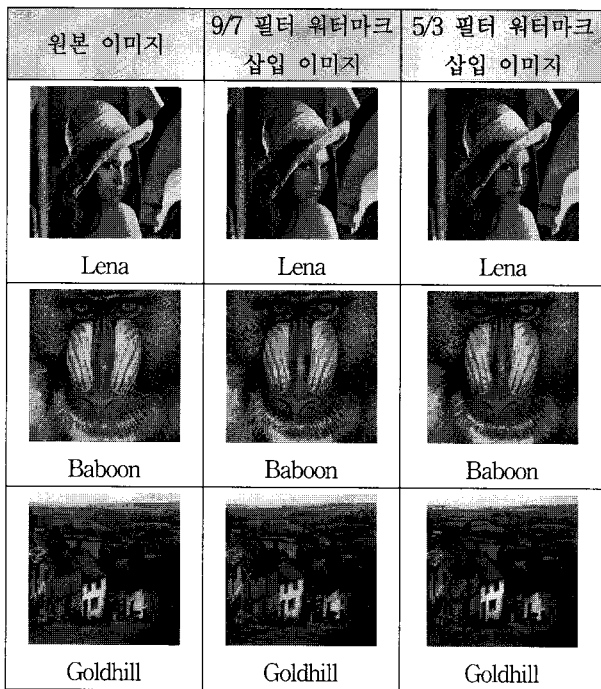


그림 8. 원본 이미지와 워터마크 이미지 비교
Fig. 8. Comparison between original image and watermarking image.

표 1. 워터마크 이미지의 PSNR(dB) 비교
Table 1. PSNR(dB) comparison with watermarking image.

Image	Proposed method 9/7 필터	Proposed method 5/3 필터	Podilchuk's method ^[1]	Dugad's method ^[2]	Lumini's method ^[3]
Lena	45.34	47.70	38.67	37.97	37.12
Baboon	39.20	41.37	35.53	35.28	34.72
Goldhill	45.14	47.50	36.64	37.60	33.72

비교 시 비가시적인 성능이 향상됨을 알 수 있다. Lena 이미지의 경우에는 약 7~10dB, Baboon 이미지는 약 4~7dB, Goldhill 이미지는 약 8~14dB 정도 비가시성이 향상 되었다. 소수점 연산으로 인해 소실되는 데이터를 이유로 9/7 필터 보다 정수 연산을 하는 5/3 필터에서 성능이 더 우수하게 나타났다.

2. 견고성 평가

본 논문에서는 공격에 대한 견고성을 평가하기 위해 식 (11)에 나타난 정규화된 유사도를 사용하여 표 2에 정리하였다. 3×3, 5×5, 7×7 LPF, 스케일링, 자르기, 노이즈 첨가 등의 공격을 이미지에 가했다. 표 2는 256×256 화소 크기를 가지는 Lena 이미지의 다양한 공격에 대한 정규화 유사도를 나타낸다.

표 2. Lena 이미지의 공격에 대한 정규화 유사도
Table 2. The normalized similarity according to a series of attacks of Lena image.

attack	Proposed method 9/7 필터	Proposed method 5/3 필터	Podilchuk's method ^[1]	Dugad's method ^[2]	Lumini's method ^[3]
3×3 LPF	58.1	57.6	35.99	6.24	26.02
5×5 LPF	35.08	38.5	10.31	2.02	9.56
7×7 LPF	18.2	19.8	3.94	0.88	4.21
Scaling	35.78	36.6	34.99	5.90	6.01
Cropping	68.2	67.85	67.78	59.82	19.30
1% Random noise	89.8	90.4	89.20	23.62	4.54

3×3, 5×5, 7×7 LPF 공격에 대해서 Podilchuk 등 [1]이 제안한 기존 알고리즘보다 본 논문에서 제안한 알고리즘이 약 15~28% 더 견고함을 알 수 있다. 스케일링에 대해서는 Podilchuk 등^[1]이 제안한 방법보다 0.78~1.6 %, 자르기에 대해서는 0.07~0.42%, 1% 노이즈 첨가에 대해서는 0.6~8.4% 더 견고함을 알 수 있다. 기존 알고리즘보다 제안한 알고리즘이 더 견고한 것은 블록간의 중요도를 스케일링 변수로 조절하여 워터마크의 강도를 미세 조정되고, 각 영상의 밝기 성분에 따라 워터마크가 랜덤하게 삽입되기 때문이다.

3. ASIC 설계

적응형 워터마크 삽입 하드웨어를 Verilog-HDL로 설계하고, 시뮬레이션과 FPGA 검증을 마친 후 Hynix 0.25 μm CMOS 공정 라이브러리를 사용하여 그림 9와 같이 ASIC으로 구현하였으며, 최대 125MHz로 동작이 가능하다. 제작된 칩의 스펙을 표 3과 같다.

칩의 정상동작을 확인하기 위해 Tektronix사의 TLA704 Logic Analyzer와 Agilent사의 16700B Pattern Generator를 사용하여 측정 하였다. 측정이 완료된 칩

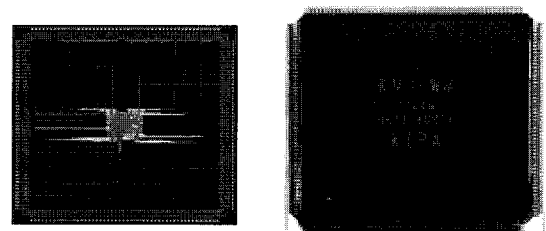


그림 9. 칩 레이아웃(좌) 과 칩(우)
Fig. 9. Chip layout(left) & Chip out(right).

표 3. 적응형 블록 워터마킹 칩 상세서
Table 3. Adaptive watermarking chip specification.

Process Technology	Hynix 0.25 μ m CMOS 1P4M
Gate count	7172 gates
Clock Frequency	125MHz
Supply Voltage	3.3V
Number of I/O pins	69

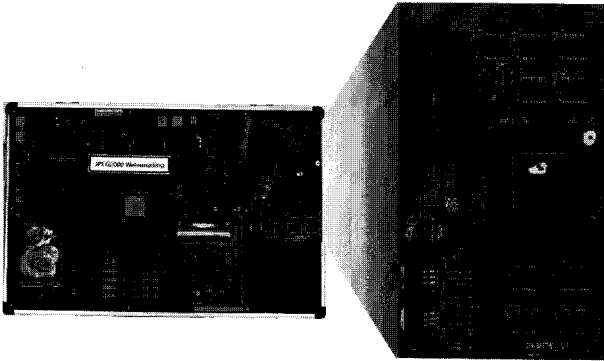


그림 10. 테스트보드와 플랫폼을 이용한 칩의 검증
Fig. 10. Verification of the chip by using test board & platform.

을 시스템 레벨에서 검정을 하기 위해 그림 10과 같이 제작된 PCB 보드에 장착하여 확장 커넥터를 통해 외부의 ARM 플랫폼 키트상에 JPEG2000 코덱을 동작시키면서 영상이미지를 입력하여 출력 데이터를 PC로 전송하여 테스트 하였으며 동작을 검증하였다.

V. 결 론

본 논문에서는 멀티미디어 환경에서의 디지털 데이터에 대한 소유권을 보호하기 위한 워터마킹 기술로 JPEG2000 DWT에 기반한 새로운 워터마킹 방법을 제안하고 이를 ASIC으로 구현하였다. 영상에 따라 워터마크 강도를 조절하는 적응형 블록 워터마킹을 제시했으며, 견고하고 우수한 비가시성을 위해 블록을 샘플 단위로 하여 두 개의 스케일 변수를 사용하였다. 워터마크 신호의 세기는 영상의 에지(edge)와 인근 블록의 상관관계에 따라 변하며 블록의 평균값에 의해서도 조정 될 수 있다. JPEG2000 표준에서도 사용이 가능하도록 워터마크 삽입 과정을 표준에 따라 단계별로 테스트 했으며 C와 매트랩(Matlab)으로 검증되었다.

본 논문에서 제시한 워터마킹 알고리즘의 성능을 평가하기 위해서 Lena, Baboon, Goldhill 이미지를 사용하였고, C 언어와 매트랩을 사용하여 테스트 하였다. 동일한 워터마크 강도를 가지는 기존 알고리즘보다 4~

14dB 정도 PSNR 성능이 개선이 되어 비가시적인 특성이 우수하게 나타났으며 필터 공격, JPEG2000 압축, 리사이즈, 자르기 등과 같은 다양한 워터마크 공격에도 기존 알고리즘들보다 견고한 특성을 보였다.

ASIC으로 구현한 결과 약 7K의 gate count를 가지며 최대 125MHz로 동작하였다. ASIC으로 검증된 워터마킹 IP는 JPEG2000 압축코덱 칩과 집적화 될 수 있으며, 프린트, 스캐너, 디지털 카메라 등에 활용될 때 비가시성과 견고성 측면에서 우수한 특성을 나타낼 것으로 기대된다.

참 고 문 헌

- [1] C. I. Podilchuk and W. Zeng, "Perceptual watermarking of still images," Proc. The First IEEE Signal Processing Society Workshop on Multimedia Signal Processing, Princeton, New Jersey, June 1997.
- [2] R. Dugad, K. Ratakonda, and N. Ahuja, "A new wavelet-based scheme for watermarking images," IEEE Int. Conf. on Image Processing, vol. 2, pp. 419-423, 1998.
- [3] A. Lumini and D. Maio, "A wavelet-based image watermarking scheme," Proc. of Int. Conf. on Information Technology: Coding and Computing, pp. 122-127, 2000.
- [4] M. Boliek, C. Christopoulos and E. Majani (editors), "JPEG2000 Part-I Final Draft International Standard," ISO/IEC JTC1/SC29/WG1 N1855, August 18, 2000.
- [5] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: an overview," IEEE Trans. Signal Processing, Vol. 46, No. 4, pp. 1103-1127, November 2000.
- [6] C. Christopoulos, "JPEG2000 Verification Model 8.5 (Thechnical dscription)," ISO/IEC JTC1/SC29/WG1, Sept. 13, 2000.
- [7] Wim Sweldens, "The Lifting Scheme : A New Philosophy in Bi-orthogonal Wavelet Constructions", Proceeding of SPIE, pp. 68-79, 1995.

 저 자 소 개



임 세 윤(학생회원)
 2005년 한국해양대학교 전자통신
 공학과 학사
 2007년 경북대학교 정보보호학과
 석사
 2007년 경북대학교 전자공학과
 박사과정

2007년 WRG 모바일TV IP 사업본부 연구원
 <주관심분야 : 모바일 TV IP 설계, JPEG2000 코
 텍 SoC설계>



최 준 림(정회원)
 1986년 연세대학교 전기공학과
 학사
 1988년 (미) Cornell대학교 전기
 전자공학과 석사
 1991년 (미) Minnesota 대학교
 전기전자공학과 박사

1991년~1997년 LG전자기술원 책임연구원
 1997년~현재 경북대학교 전자전기컴퓨터학부
 부교수
 <주관심분야 : Network 정보보호 SoC 설계, 영
 상압축코덱 SoC 설계>