

무효 블록의 격리를 이용한 새로운 이미지 인증 방법

(A New Image Authentication Method through Isolation of Invalid Blocks)

김효철[†] 김현철^{**} 유기영^{***}
 (Hyo Chul Kim) (Hyun Chul Kim) (Ki-Young Yoo)

요약 본 논문에서는 이미지 인증을 목적으로 무효한 블록들을 격리하는 새로운 워터마킹 방법을 제안한다. 이 과정에서 워터마크의 비인식성과 깨지기 쉬운 워터마크를 보장하기 위하여 DCT 도메인의 고주파 영역을 사용하였다. 여러 가지 실험을 통하여 예러가 보정된 원본 영상들과 워터마크된 이미지들 사이에 유효한 블록들이 동일함을 확인하였다. 결과적으로 워터마크 추출 과정에서 원본 이미지나 워터마크와 같은 여분의 정보가 필요 없으며, 워터마크된 이미지만으로 2진 워터마크를 추출할 수 있다는 사실을 발견하였다. 실험 결과들은 제안된 방법이 이미지 인증을 위하여 성공적으로 사용될 수 있음을 보여준다.

키워드 : 워터마크, 퍼블릭 워터마킹, 이미지 인증

Abstract In this paper, we propose a new watermarking method using isolation of invalid blocks for image authentication. We embed messages into the high frequency band in the DCT domain because of its imperceptibility and fragility. And we were able to inspect the identity of valid blocks between error corrected original images and watermarked images through experiments. As a result, we found that the watermarked image is enough to extract binary messages. Therefore, the extra information such as the original image or watermark was not necessary in our extracting process. Experiment results show that the proposed method can be used successfully for image authentication.

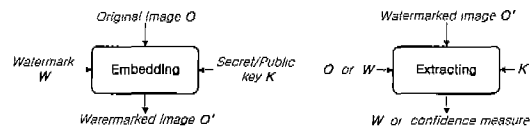
Kye words : Watermark, Public watermarking, Image authentication

1. 서론

최근 멀티미디어 정보를 많이 이용하게 됨에 따라 정지 영상의 저작권 보호와 인증을 위한 효율적인 방법의 필요성이 증가하고 있다. 이를 위하여 정지 영상에 워터마크와 같이 보이지 않는 정보를 삽입하는 여러 가지 방법들이 연구되었다[1]-[3]. 그림 1은 전형적인 워터마크의 삽입과 추출 과정에 대한 것이다[4].

워터마크 삽입 과정에서는 그림 1(a)와 같이 비밀/공개 키 K 로 워터마크 W 혹은 삽입 위치를 암호화하여 원본 이미지 O 를 배포용으로 사용되는 워터마크된 이미지 O' 로 변환한다. 이미지 인증을 위한 워터마크 추출

과정에서는 그림 1(b)와 같이 워터마크된 이미지 O' 에 대하여 O 또는 W , K 를 이용하여 삽입된 W 또는 신뢰할만한 척도를 추출한다. 워터마킹 체계는 그 성질과 입력 출력의 조합의 차이에 따라 다음과 같은 3가지 유형으로 나누어질 수 있다[4].



(a) 워터마크 삽입 과정 (b) 워터마크 추출 과정

그림 1 전형적인 워터마크의 삽입과 추출 과정

[†] 비 회 원 : 계명문화대학 전산정보처리과 교수

khc@km-c.ac.kr

^{**} 비 회 원 : 포항1대학 정보통신과 교수

hckm@purple.knu.ac.kr

^{***} 통신회원 : 경북대학교 컴퓨터공학과 교수

yook@bh.knu.ac.kr

논문접수 : 2000년 12월 4일

심사완료 : 2001년 10월 13일

(1) Private 워터마킹에서는, 워터마크를 추출할 때 적어도 원본 이미지를 필요로 한다 : $O' \times O \times W \times K \rightarrow \{0,1\}$.

(2) Semiprivate 워터마킹에서는, 워터마크를 추출할 때 원본 이미지를 필요로 하지 않는다 : $O' \times W$

$$\times K \rightarrow \{0,1\}.$$

(3) Public 워터마킹(blind 워터마킹)에서는, 워터마크를 추출할 때 원본 이미지나 삽입된 워터마크를 필요로 하지 않는다: $O' \times K \rightarrow W$.

일반적인 응용에 있어서 워터마크는 원본 이미지와 부호화된 이미지 사이에 식별 가능한 차이가 없도록 삽입되어야 한다. 그리고 이미지 프로세싱이나 손실 이미지 압축과 같은 가능한 공격들에 대하여 지속될 수 있도록 강한 메시지 코딩 방법을 사용해야 한다[5]. 그런데, 비인식성(imperceptibility)은 어떤 응용에서나 중요한 것이지만 강인성(robustness)은 어떤 응용에서는 부적합하다. 위조 여부의 판별이 중요한 암호화적인 이미지 인증 문제에는 깨지기 쉬운 워터마크(fragile watermark)[6][7]가 의미를 가지며, 원본 이미지나 다른 정보를 사용하지 않고 워터마크를 추출하는 것이 요구된다: $O' \times K \rightarrow W$. 이러한 응용에 부합하는 Public 워터마킹에서는 워터마크 추출 과정에서 워터마크된 이미지 이외의 어떤 정보도 사용하지 않으므로 원본 이미지와 워터마크된 이미지 사이의 적절한 상호연관성(cross-correlation property)을 사용하여야 한다.

워터마크 삽입을 위한 유효 블록을 선택하는 방법에 대한 연구로는 PIM(picture information measure)이라는 블록의 복잡도에 대한 척도를 이용하는 방법[8]과 AC 성분의 에너지 Ea 를 척도로 하는 방법[9] 등이 있다. 그러나 이 연구들은 원본 이미지와 워터마크된 이미지의 상호연관성을 고려하지 않고 있어서 Public 워터마킹을 위해서는 사용될 수 없다. 식 (1)은 PIM을 구하기 위한 것이다. 여기에서 L 은 블록 내에서 그레이 레벨의 수이며, $h(i)$ 는 블록의 그레이 레벨 i 에 대한 히스토그램(histogram)이다.

$$PIM = \sum_{i=0}^{L-1} h(i) - Max_i[h(i)] \quad (1)$$

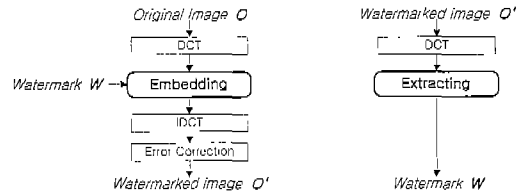
다음의 식 (2)는 $N \times N$ 픽셀 블록에서 Ea 를 구하기 위한 것이며, F 는 DCT 함수이다.

$$Ea = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} |F(u, v)|^2 \quad (2)$$

본 연구에서는 비인식성 유지와 깨지기 쉬운 워터마크를 위하여 DCT 도메인의 고주파 영역에 워터마크를 삽입하였으며[10], 이로 인한 고주파 영역 계수값들의 변동폭이 비인식성 유지와 워터마크 추출에 영향을 미치지 않는 유효 블록들만을 선별하여 워터마크 삽입과 추출에 이용하였다. 이어지는 2장에서는 워터마크의 삽입과 추출에 대하여 설명하였으며, 3장에서는 실험 결과를 다루었다. 마지막으로 4장에서 결론을 언급하였다.

2. 워터마크의 삽입과 추출

그림 2는 본 논문에서 제안한 워터마크 삽입과 추출 과정이다.



(a) 워터마크 삽입 과정 (b) 워터마크 추출 과정

그림 2 제안된 워터마크 삽입과 추출 과정

O 는 식 (3)과 같은 $N \times N$ 픽셀의 원본 그레이 이미지이고, W 는 식 (4)와 같은 최대 $(N/4) \times (N/4)$ 비트의 워터마크이다. 워터마크는 단극성(unipolar)인 0과 1의 스트림, 또는 양극성(bipolar)인 -1과 1의 스트림으로 구성될 수 있는데, 여기에서는 연산의 효율성을 위해 양극성 스트림을 사용하였다. 또한 보안을 강화하기 위하여 비밀/공개키 K 를 이용한 메시지의 순서변환(permutation)을 사용할 수 있으나 여기서는 다루지 않았다.

$$O = \{ o(i,j), 0 \leq i, j < N \}, o(i,j) \in 0, 1, \dots, 2^L - 1$$

where L is the number of bits used
in each pixel. (3)

$$W = \{ w(i), 0 \leq i < N^2/16 \}, w(i) \in -1, 1 \quad (4)$$

2.1 고주파 항의 선택과 처리

먼저 원본 이미지를 8×8 블록들로 나누고 각 블록을 DCT로 변환한 다음, 그림 3과 같이 고주파 영역의 5개 항을 선택하여 사용하였다. 이 5개항 중에서 실제로 워터마크는 4곳에만 삽입되며, 원본 이미지에 최소한의 영향을 주면서 니블(nibble) 단위의 연산을 가능케 하기 위하여 선택 항의 수를 최소로 하였다.

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

그림 3 8×8 블록에서 5개 고주파 항의 선택

다음으로 블록에서 선택된 5개 고주파 항 계수들의 평균값과 최대 값과 최소 값의 차이를 구한다. 여기서 s_n 은 임의의 블록 n 에서 선택된 5개의 계수들을 저장한 배열이다. 그리고 avg_n 은 식 (5)와 같이 블록 n 의 선택된 계수들에 대한 평균이며, gap_n 은 식 (6)과 같이 블록 n 의 선택된 계수들에서 최대 값(mx_n)과 최소 값(mn_n)의 차이(거리)이다.

$$avg_n = \sum s_n(u)/5 \quad \text{where } 0 \leq u < 5 \quad (5)$$

$$mx_n = \max(s_n(u))$$

$$mn_n = \min(s_n(u))$$

$$gap_n = mx_n - mn_n \quad (6)$$

일반적인 이미지에서 각 블록들을 거리(gap)를 기준으로 살펴보면 대략적으로 그림 4와 같은 분포가 나타난다. 여기에서는 gap 이 0에 가까운 블록들이 많고 소수의 블록들이 큰 gap 을 가진다는 것을 볼 수 있다. 여기서 α 는 임의의 상수이다.

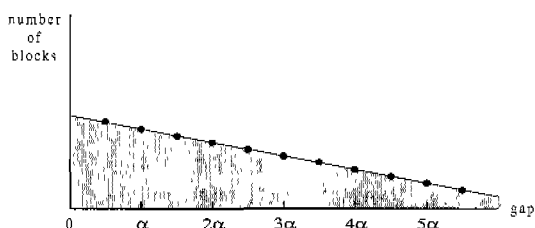


그림 4 일반적인 이미지에서 블록들의 분포

gap 이 큰 블록들은 상대적으로 계수들의 편차가 크므로 평균값을 워터마크 삽입에 이용한다면 예러도 증가하게 될 것이다. 따라서 워터마크는 gap 이 α 보다 작은 유효한 블록들에 삽입하며, gap 이 α 이상인 블록들은 워터마크를 삽입할 수 없는 무효한 블록으로 취급하기로 한다.

2.2 무효한 블록의 격리

무효한 블록들 중에서 gap 이 α 에 인접한 블록들은 IDCT/DCT 변환에 따른 계수 값들의 변화로 인해 유효한 블록으로 오인될 수 있다. 따라서 이러한 블록들을 유효한 블록들로부터 안전하게 격리시켜야 한다. 이를 위하여 gap 이 α 와 3α 사이인 블록의 5개 계수 중에서 최대 값과 최소 값에 $(3\alpha - \alpha)/2$ 를 가감하여 gap 을 2α 만큼 증가시킨다. $\alpha \leq gap_n < 3\alpha$ 인 모든 블록 n 에 대하여 식 (7)과 (8)을 적용한다.

$$mx_n = mx_n + \alpha$$

$$mn_n = mn_n - \alpha$$

$$f_n(i,j) = mx_n \quad \text{where } f_n(i,j) \text{ is the location of maximum value.} \quad (7)$$

$$f_n(k,l) = mn_n \quad \text{where } f_n(k,l) \text{ is the location of minimum value.} \quad (8)$$

그림 5는 gap 이 α 에 인접한 무효 블록들에 대하여 gap 을 2α 만큼 증가시켜 해당 블록들을 유효 블록들로부터 격리한 결과이다.

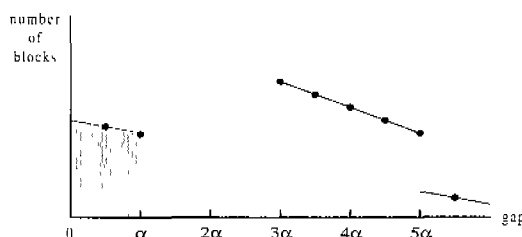
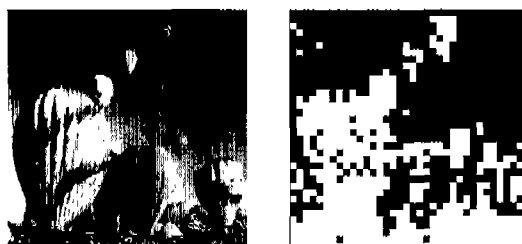


그림 5 유효 블록에 인접한 무효 블록들의 격리

그림 6은 “Balloons” 이미지에서 무효 블록들을 격리하고 표시한 것이다.



(a) 원본 이미지

(b) 무효 블록들(흰 영역)

그림 6 “Balloons” 이미지에서 블록의 유효성($\alpha=2.8$)

2.3 워터마크의 삽입

유효한 블록 n 에 대해서는 평균값 avg_n 을 59번 항에 넣고, 나머지 4개항에는 avg_n 에 워터마크 비트 값의 $\alpha/2$ 배를 더하여 삽입한다. 하나의 블록에서 사용하는 고주파 항은 5개이지만, 59번 항에는 워터마크를 판별할 수 있는 기준 값을 넣고 나머지 4개항에 실제 메시지가 삽입된다. 여기에서 δ 는 $\alpha/2$ 이며 $gap_n < \alpha$ 인 모든 블록 n 에 대하여 식 (9)를 적용한다.

$$f_n(6,6) = avg_n \quad \text{where } f_n(6,6) \text{ is the location of the 59th element.}$$

$$f_n(u,v) = avg_n + w(p+i) \delta$$

where $f_n(u,v)$ is the location of the 60th, 61st, 62nd, 63rd element.

$w(p)$ is first watermark bit that will be processed, $0 \leq i < 4$. (9)

단락 블록 n 에 삽입되는 4비트 워터마크가 $-1, -1, -1, -1$ 이나 $1, 1, 1, 1$ 과 같이 동일하다면 gap_n 은 avg_n 과 $avg_n - \alpha/2$ 혹은 avg_n 과 $avg_n + \alpha/2$ 의 차이가 되므로 $\alpha/2$ 가 될 것이다. 그리고 워터마크가 한 비트라도 다르다면 gap_n 은 $avg_n - \alpha/2$ 와 $avg_n + \alpha/2$ 의 차이가 되므로 α 가 될 것이다. 그러므로 4비트 워터마크의 내용에 따라 워터마크 삽입 후의 블록 분포는 그림 7과 같다.

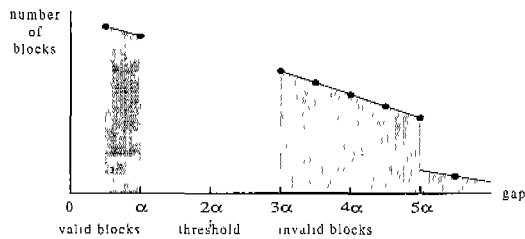


그림 7 워터마크 후의 블록 분포와 경계값

워터마크 추출시 유효한 블록들의 판정을 위해서 사용하는 경계값(threshold)은 DCT/IDCT 변환과 워터마크 삽입 과정에 따르는 계수들의 에러를 고려하여 유효블록의 상한값과 무효블록의 하한값의 중간값인 2α 로 한다.

2.4 에러 보정

이러한 방법을 다양한 이미지들에 적용해 본 결과, 다음과 같은 유형의 블록 판정 에러가, 비록 빈도는 낮지만, 발생하였다.

에러 유형 1 : gap 이 3α 이상으로 처리되었지만 IDCT/DCT 변환에 의해 gap 이 경계값 2α 이하로 되는 경우로, 무효 블록이 유효 블록으로 오인된다.
에러 유형 2 : gap 이 α 보다 작지만 워터마크를 삽입하고 IDCT/DCT 변환을 한 결과 gap 이 경계값 2α 이상이 되는 경우로, 유효 블록이 무효 블록으로 오인된다.

이러한 에러를 보정하지 않으면 원본 이미지와 워터마크가 포함된 이미지의 유효 블록이 다르게 되므로 올바른 추출 과정을 진행할 수 없게 된다. 유형 1의 에러는 상대적으로 발생 빈도가 낮았으며, 해당 블록에서 선택된 계수 중의 최소 값과 최대 값에 α 보다 조금 큰 수를 가감하여 해결하였다. 그리고 유형 2에 대해서는 5개의 항에 일괄적으로 평균을 삽입하여 gap 을 0으로 조정하였다. 이 처리로 인하여 워터마크 추출시 비트 에러가 증가되지만, 블록 판정의 에러를 제거할 수 있었다.

원본 이미지와 워터마크된 이미지 사이의 상호연관성 보장을 위한 유형 1, 2의 에러를 보정한 후에도 워터마크 추출 과정에서 다음과 같은 에러가 발생한다.

에러 유형 3 (비트 에러) : 추출한 워터마크 비트가 원본 워터마크 비트와 일치하지 않는 경우이다

워터마크 추출에서 유형 3의 에러 발생을 없애기 위하여, 미리 워터마크 추출 과정을 진행한다. 비트 에러가 발생하는 블록에 대해서는 식 (9)에서 결함이 있는 δ 를 새로운 것으로 반복 적용하는 방법으로 정확한 추출을 보장하는 계수들을 찾아서 교체한다. 이때 원본 워터마크 비트들은 교체된 계수들의 적용 결과에서 추출된 워터마크 비트들과 비교되며, 변경된 gap 의 유효성도 검사된다. 에러 보정이 포함된 워터마크 삽입에 대한 전체 과정을 요약하면 다음과 같다.

```

 $F_n = FDCT(O)$ 
Isolation of Invalid blocks and watermark embedding
 $O''' = IDCT(F_n)$ 
 $F'_n = FDCT(O''')$ 
Error(includes error type1, type2) correction
 $O'' = IDCT(F'_n)$ 
Watermark extracting
Error(type3) correction
 $O' = IDCT(F_n)$ 
    
```

먼저 원본 이미지에 제안된 방법을 이용하여 워터마크를 삽입한 결과(F_n)를 IDCT 변환하여 워터마크된 이미지 O''' 를 얻는다. 다음으로 O''' 를 한번 더 DCT로 변환하여 블록 판정 에러 정보를 얻고, 이 정보를 이용하여 F_n 에 대하여 일단 유형 1, 2의 에러를 보정한 다음 DCT 변환하여 워터마크된 이미지 O'' 를 생성한다. 계속하여 O'' 에서 워터마크를 추출하는 과정에서 비트 에러 정보를 얻는다. 마지막으로, 이 정보를 이용하여 F_n 에 대하여 비트 에러를 보정한 다음 IDCT 변환을 통하여 최종적으로 워터마크된 이미지 O' 를 완성한다.

2.5 워터마크의 추출

삽입된 워터마크를 추출하기 위해서 이미지를 8×8 블록들로 나누고 각 블록을 DCT로 변환한 다음, 5개항에 대하여 gap 을 계산하고 2α 를 경계값으로 하여 유효한 블록을 판정한다. 유효한 블록에 대해서는 4개의 계수 각각에서 기준 값을 뺀 결과의 부호(1, 0)로 4비트의 워터마크를 추출한다. 즉, $gap_n < 2\alpha$ 인 모든 블록 n 에 대하여 식 (10)과 같이 적용한다.

```

 $w'(p+i) = \text{sign}[f'_n(k,l) - f_n(6,6)]$ 
where  $f_n(6,6)$  is the location of the 59th element.
where  $f_n(k,l)$  is the location of the 60th, 61st, 62rd, 63nd element.
    
```

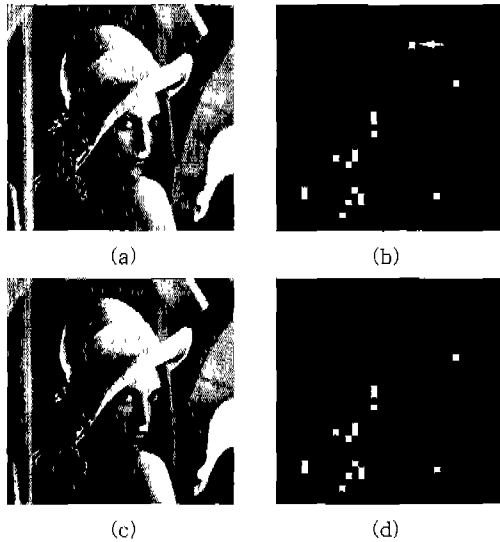
$w'(p)$ is last watermark bit that was processed, $0 < i \leq 4$. (10)

3. 실험 결과

여러 가지 실험들을 위해서 사용한 원본 이미지들은 288×288 크기의 그레이 이미지이며, "Lena" 이미지와 Adobe사의 Photoshop 패키지에 포함되어 있는 샘플 파일들을 변환하여 사용하였다. 워터 마크는 64×64 의 이진 이미지와 다양한 비트 패턴들을 사용하였다.

3.1 블록의 유효성에 대한 결과

그림 8은 "Lena" 이미지에서 블록의 유효성에 대한 결과이며, 표 1은 α 값의 변화에 따른 결과이다.



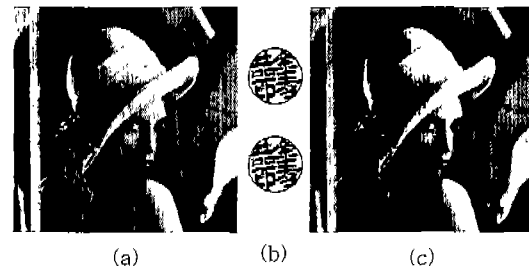
(a) 원본 이미지
 (b) 원본 이미지에서 유효 블록들(검은 영역)과 보정된 블록(화살표 영역)
 (c) 워터마크된 이미지
 (d) 워터마크된 이미지에서 유효 블록들
 그림 8 "Lena" 이미지에서 블록의 유효성 (α 2.8)

표 1 "Lena" 이미지에서 α 값의 변화에 따른 블록의 유효성 (총 블록 수=1,296)

α 값	유효 블록 수	무효 블록 수	유효성 보정 블록 수
2.4	1,273	23	0
2.6	1,275	21	0
2.8	1,280	16	1
3.0	1,282	14	1
3.2	1,286	10	2

표 1에서 보듯이 큰 α 를 사용하면 무효 블록 수가 줄어들지만 다양한 원본 이미지들에서 평균적으로 α 가 2.8일 경우에 가장 좋은 성능을 보였기 때문에 실험의 대부분 결과들을 α 값이 2.8일 경우로 설정하여 표시하였다. 표 2에서는 "Lena" 이미지에서 gap에 따른 블록들의 분포를 보였다. 이 테이블에서는 대다수의 블록이 작은 α 값을 가지며 경계값 이하로 집중됨을 볼 수 있다.

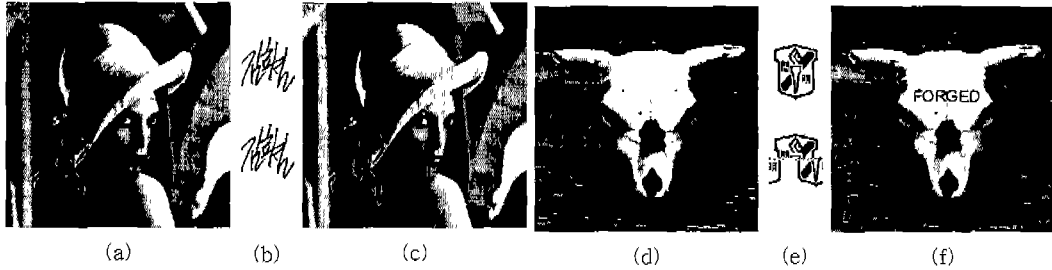
이 실험들에서 여러가 교정된 원본 이미지와 워터마크된 이미지 사이의 유효 블록들이 동일하며, 두 이미지 간의 상호연관성이 지속되고 있음을 확인하였다. 이 결과들은 원본 이미지나 워터마크 정보 없이 워터마크된 이미지만으로 워터마크의 추출이 가능하다는 것을 보여준다.



(a) 원본 이미지
 (b) 삽입된 워터마크와 추출된 워터마크
 (c) 워터마크된 이미지
 그림 9 "Lena" 이미지에서 워터마크의 삽입과 추출 (α =2.8, 여러 유형 1과 2만 보정)

표 2 "Lena" 이미지에서 gap에 따른 블록들의 분포 (총 블록 수=1,296, 경계값=2 α)

	$0 \leq \text{gap} < 0.5\alpha$	$0.5\alpha \leq \text{gap} < 1\alpha$	$1\alpha < \text{gap} < 1.5\alpha$	$1.5\alpha \leq \text{gap} < 2\alpha$	$2\alpha \leq \text{gap} < 2.5\alpha$	$2.5\alpha \leq \text{gap} < 3\alpha$	$3\alpha \leq \text{gap} < 3.5\alpha$	$3.5\alpha \leq \text{gap} < 4\alpha$	$4\alpha \leq \text{gap} < 4.5\alpha$	$4.5\alpha \leq \text{gap} < 5\alpha$
Number of blocks (original image)	1,190	90	11	4	1	0	0	0	0	0
Number of blocks (watermarked image)	2	497	781	0	0	1	13	2	0	0



(a) 워터마크된 "Lena" 이미지
 (b) "Lena" 이미지에 삽입된 워터마크와 추출된 워터마크
 (c) 위조한 워터마크된 "Lena" 이미지(low-pass filtering)
 (d) 워터마크된 "Skull" 이미지
 (e) "Skull" 이미지에 삽입된 워터마크와 추출된 워터마크
 (f) 위조한 워터마크된 "Skull" 이미지(some text insertion)
 그림 10 위조한 워터마크된 이미지들에 대한 결과($\alpha=2.8$, 에러 유형 1과 2만 보정)

3.2 워터마크 삽입과 추출에 대한 결과

그림 9와 표 3은 이러한 상호연관성을 이용하여 워터마크를 삽입하고 추출한 결과에 대한 것이다. 이 실험에서는 에러 유형 1과 2에 대한 보정만 하였으며 워터마크 비트 에러에 대한 보정은 고려하지 않았다.

이 결과에 따르면 원본 이미지와 워터마크된 이미지 사이에서 눈으로 식별할 수 있는 차이를 발견할 수 없었으며, 추출된 워터마크도 약간의 비트 에러만 제외한다면 원본 워터마크의 모양을 그대로 유지하였다. 표 3에서는 α 값의 변화에 따른 비트 에러율을 나타내었다.

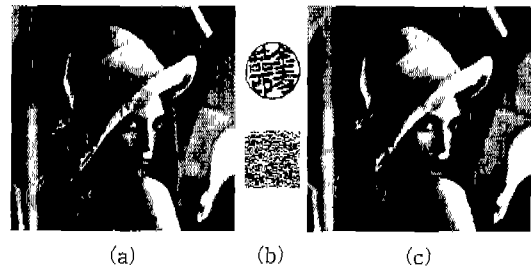
표 3 "Lena" 이미지에서 워터마크의 삽입과 추출 (에러 유형 1과 2만 보정)

α 값	유효 블록 수	무효 블록 수	유효성 보정 블록 수	워터마크 추출 비트 에러율
2.0	1,262	34	16	116/5,048 (2.38%)
2.4	1,273	23	0	55/5,092 (1.08%)
2.6	1,275	21	0	39/5,100 (0.66%)
2.8	1,280	16	1	29/5,120 (0.57%)
3.0	1,282	14	1	19/5,128 (0.37%)
3.2	1,286	10	2	9/5,144 (0.17%)
4.0	1,290	6	6	14/5,160 (0.27%)

그림 10은 위조한 워터마크된 이미지에 대한 결과들이다. 이 실험에서는 워터마크된 이미지들을 로우 패스 필터링과 약간의 문자를 삽입하는 방법으로 위조하였다.

결과적으로 위조한 이미지들에서 추출한 워터마크들은 크게 왜곡되어 나타났으며, 위조 여부를 확실하게 판별할 수 있었다.

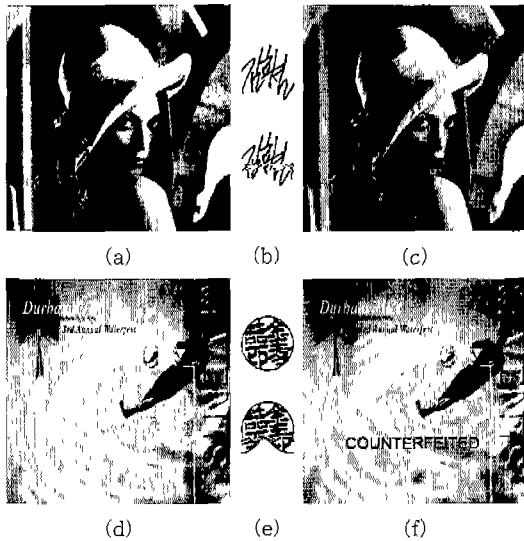
그림 11은 블록 판정 에러들과 비트 에러를 모두 보정하여 삽입된 워터마크에 대한 실험 결과이다. 이 실험을 통하여 추출된 워터마크와 원본 워터마크가 완전히 일치하는 것을 확인하였다.



(a) 원본 이미지
 (b) 삽입된 워터마크와 추출된 워터마크
 (c) 워터마크된 이미지
 그림 11 "Lena" 이미지에서의 워터마크의 삽입과 추출 ($\alpha=2.8$, 모든 에러 보정)

그림 12는 모든 에러를 보정하여 생성한 워터마크된 이미지를 위조하여 실험한 결과들이다. 이 실험에서는 워터마크된 "Lena" 이미지의 눈 부분만을 샤프닝 필터로 처리하였으며, 워터마크된 "Waterskier" 이미지는 약

간의 문자를 삽입하는 방법으로 위조하였다. 결과적으로 위조된 이미지들에서 추출된 워터마크들은 식별할 수는 있지만 왜곡되어 나타났으며, 위조 여부를 확실하게 판별할 수 있었다.



(a) 워터마크된 "Lena" 이미지
 (b) "Lena" 이미지에 삽입된 워터마크와 추출된 워터마크
 (c) 위조한 워터마크된 "Lena" 이미지(sharpening of the eyes portion only)
 (d) 워터마크된 "Waterskier" 이미지
 (e) "Waterskier" 이미지에 삽입된 워터마크와 추출된 워터마크
 (f) 위조한 워터마크된 "Waterskier" 이미지(some text insertion)

그림 12 위조한 워터마크된 이미지들에 대한 결과 ($\alpha = 2.8$, 모든 예러 보정)

이상의 실험 결과들은 본 논문에서 제안된 상호연관성과 워터마킹 방법이 이미지 인증을 위한 각종 응용에 성공적으로 사용될 수 있음을 보여주었다.

4. 결론

본 논문에서는 정지 영상의 효율적이고도 정확한 인증을 위하여 워터마크를 삽입하고 추출하는 새로운 방법을 제안하고 실험한 결과를 기술하였다. 제안된 방법에서는 비인식성과 깨지기 쉬운 워터마크를 위하여 DCT 도메인의 고주파 영역 값들을 처리하였으며, 특정

블록들을 격리시키는 방법을 사용하였다. 또한, 워터마크를 추출할 때 워터마크가 삽입된 이미지 이외의 정보를 사용하지 않는 Public 워터마킹 체계를 위하여 새로운 상호연관성을 이용하였으며, 워터마크 삽입 과정에서 발생하는 블록 유효성에 대한 2가지 유형의 예러들과 워터마크 비트 예러들을 보정하였다.

결과적으로, 우리는 워터마크 추출 과정에서 원본 이미지나 원본 워터마크와 같은 여분의 정보를 가지지 않고도 워터마크된 이미지만으로 워터마크를 추출하는 새로운 Public 워터마킹 방법을 개발하였으며, 워터마크 삽입 유무만을 나타내는 척도를 넘어서 정확한 워터마크를 성공적으로 재생할 수 있었다. 그리고 각종 실험들을 통하여 본 논문에서 제안된 상호연관성과 워터마킹 방법이 이미지 인증을 위한 여러 가지 응용들에 사용될 수 있음을 확인하였다. 앞으로 이 논문에서 제안한 상호연관성과 예러 보정 알고리즘들을 개선시키고 관련된 응용에 적용하여 그 실효성을 다양한 각도로 검증할 것이다. 또한 여러 가지 가능한 공격에 대한 강인성을 연구하여 이미지 인증은 물론 저작권 보호에도 활용될 수 있는 방법을 찾을 것이다.

참고 문헌

- [1] J. T. Brassil, S. Low, N. F. Maxemchuk, and L. O'Gorman, "Electronic Marking and Identification Techniques to Discourage Document Copying," *IEEE J. Selected Areas Commun.*, vol. 13, no. 8, Oct 1995.
- [2] I. Pitas and T. H. Kaskalis, "Applying Signatures on Digital Images," *Proc. IEEE Nonlinear Signal and Image Processing*, pp. 460-463, June 1995.
- [3] I. J. Cox, J. Killian, T. Leighton, and T. Shamoon, "A Secure Robust Watermark for Multimedia," *IEEE Trans. Image Processing*, vol. 6, no. 12, pp. 1673-1687, Dec 1997.
- [4] Stefan Katzenbeisser, Fabien A. P. Petitcolas and editors, *Information Hiding Techniques for Steganography and Digital Watermarking*, 2000 ARTECH HOUSE, INC., 1999.
- [5] C.-T. Hsu and J.-L. Wu, "Hidden digital watermarks in images," *IEEE Trans. Image Processing*, vol. 8, no. 1, pp. 58-68, Jan 1999.
- [6] F. Mintzer, G. W. Braudaway, M. M. Yeung, "Effective and Ineffective Digital Watermarks," *ICIP*, 1997.
- [7] Don Coppersmith, Fred Mintzer, Charles Tresser, Chai Wah Wu and Minerva M. Yeung, "Fragile imperceptible digital watermark with privacy

- control," *Proceedings of SPIE, Security and Watermarking of Multi-media Contents*, San Jose California, pp. 79-84, Jan 1999.
- [8] Shi-Kuo Chang, *Principles of Pictorial Information Systems Design*, Prentice Hall, pp. 61-81, 1989.
- [9] K. S. NG, L. M. Cheng, "Selective block assignment approach for robust digital image watermarking," *Proceedings of SPIE, Security and Watermarking of Multimedia Contents*, San Jose California, USA, Jan 1999.
- [10] J. J. K. Oruanaidh, W. J. Dowling, and F. M. Boland, "Watermarking digital images for copyright protection," *IEEE Proceedings on Vision, Image and Signal Processing*, 143, pp. 243-246, Aug 1996.



김 효 철

1987년 경북대학교 전자공학과(전산) 졸업(공학사). 1989년 경북대학교 전자공학과(전산) 졸업(공학석사). 1989년 ~ 1996년 국방과학연구소(ADD) 근무. 1999년 경북대학교 컴퓨터공학과 박사수료. 1996년 ~ 현재 계명문화대학 부교수. 관심분야는 멀티미디어, 병렬 및 분산처리, 암호화 등



김 현 철

1995년 경일대학교 컴퓨터공학과 졸업(공학사). 1997년 경북대학교 컴퓨터공학과 졸업(공학석사). 1999년 경북대학교 컴퓨터공학과 박사수료. 2000년 ~ 현재 포항1대학 정보통신과 전임강사. 관심분야는 어레이 프로세서 설계, 병렬 및 분산처리, 암호화 등



유 기 영

1976년 경북대학교 수학교육학과 졸업(이학사). 1978년 한국과학기술원 전산학과 졸업(공학석사). 1992년 미국 Rensselaer Polytechnic Institute 졸업(이학박사). 1978년 ~ 현재 경북대학교 컴퓨터공학과에 재직. 관심분야는 병렬처리, DSP array processor 설계, 암호화 등