

# 히스토그램 이동을 이용한 고용량 리버서블 워터마킹

배 성 호<sup>†</sup>

## 요 약

리버서블 워터마킹은 디지털 영상에 정보를 은닉하고 인증된 상대에게만 은닉된 정보를 복호화하고 원래의 상태로 영상을 복구하는 방법이다. 본 논문에서는 히스토그램 이동을 이용한 고용량 리버서블 워터마킹을 제안한다. 본 논문에서는 높은 삽입량을 위하여 영상을 2×2 블록으로 분할하고, 각 블록에 수평, 수직, 대각에 대한 화소쌍을 이용하여 가장 빈도가 높은 최대 삽입공간을 찾는다. 최대 삽입공간을 포함하는 위치맵을 이용하여 오버플로우와 언더플로우를 제거하여, 반복적인 워터마크 삽입을 통해 삽입량을 증가시킨다. 실험을 통하여 제안한 방법이 기존의 리버서블 워터마킹 방법과 비교하여 좋은 화질과 높은 삽입량을 나타냄을 확인하였다.

## A High Capacity Reversible Watermarking Using Histogram Shifting

Sung-Ho Bae<sup>†</sup>

### ABSTRACT

Reversible watermarking hides some information in a digital image in such a way that an authorized party could decode the hidden information and also restore the image to its original state. In this paper, a high capacity reversible watermarking method using histogram shifting is proposed. In order to increase embedding capacity, the proposed method divides the image into 2×2 blocks and uses a paring(horizontal, vertical, diagonal) inside each block, then finds a maximum embedding bin which has the most frequent difference values among the parings. Also, the proposed method removes the overflow and underflow by using location map which including the maximum embedding bin and increases the embedding capacity by embedding iteratively. The experimental results show that the proposed method provides a high embedding capacity and good visual quality compared with the conventional reversible watermarking methods.

**Key words:** Reversible watermarking(리버서블 워터마킹), Histogram shifting(히스토그램 이동), Location map(위치맵), Maximum embedding bin(최대 삽입 공간)

### 1. 서 론

최근 네트워크의 발달과 보급을 통해 쉽게 디지털 콘텐츠 전송이 가능하여 디지털 멀티미디어 워터마크의 생산과 사용이 급진전되고 있다. 멀티미디어 워터마크의 상업적 가치와 사용 폭이 증가된 반면, 멀티미디어 워터마크들이 네트워크와 인터넷을 통해

쉽게 불법적인 복제나 변조가 가능하여 저작권과 보안상의 문제점이 발생된다[1]. 이러한 문제점을 해결하기 위하여 디지털 워터마크 내부에 소유권을 보장할 수 있도록 개인 정보를 은닉하여 확인할 수 있는 기술인 디지털 워터마킹 방법에 대한 연구가 활발히 진행되고 있다[1-5].

지금까지 기존의 디지털 워터마킹 방법은 비가시

※ 교신저자(Corresponding Author): 배성호, 주소: 부산광역시 남구 신천로 179(608-711), 전화: 051)629-1212, FAX: 051)629-1212, E-mail: baesh@tu.ac.kr

접수일: 2009년 8월 13일, 수정일: 2009년 9월 14일  
완료일: 2009년 9월 16일

<sup>†</sup> 정희원, 동명대학교 의용공학과 부교수

성을 만족시키기 위해 디지털 콘텐츠의 저작권 정보를 사람들이 인지할 수 없을 정도로만 변경을 수행하여 저작권을 보호하였지만, 디지털콘텐츠의 원래 정보를 복원할 수는 없었다[6].

대부분의 환경에서 콘텐츠의 변경은 적법한 목적으로 제공된다. 그러나 의료영상에 대한 부주의한 변경은 오진의 결과를 가져오고, 법정에서 증거가 되는 영상의 악의적인 훼손은 잘못된 유죄판결의 결과를 가져온다. 그러므로, 콘텐츠가 반드시 변경이 되지 않아야 하는 의료영상과 군사영상과 같이 특별한 영상들은 콘텐츠의 무결성에 대한 인증이 필요하다. 그러나 대부분의 워터마킹에 기반한 인증은, 인증 그 자체에 의하여 원영상이 왜곡될 수밖에 없다. 전형적으로 그 왜곡은 비트 대체(replacement), 양자화, 명도 0과 255에서의 절단(truncation) 등으로 완전히 제거될 수 없다[7]. 어떤 영상이 의료나 군사 응용분야에서 매우 높은 전략적 중요성을 가진다면 비록 그 왜곡이 매우 작더라도 허용이 되지 않는다. 그러므로, 만약 영상의 인증이 증명되면, 인증을 위해 삽입된 왜곡이 제거 되어야 한다. 이러한 조건을 만족하는 데이터 삽입 기술을 리버서블 워터마킹이라 한다.

리버서블 워터마킹은 정보를 영상에 은닉하고 인증된 상대에게만 은닉된 정보를 복호화하고 원영상을 초기상태로 복원한다. 이에 최근 수년간 저작권 정보를 추출 후 원래 정보를 복원할 수 있는 리버서블 워터마킹에 대한 연구가 활발히 진행되고 있다[1, 6-9]. 리버서블 워터마킹의 기본적인 요구사항은 데이터 삽입후 영상의 화질 왜곡이 적어야하며, 삽입한 데이터를 제거하면 원영상을 복원할 수 있어야 한다. 리버서블 워터마킹의 성능은 다음과 같이 측정된다[8].

- 1) 페이로드량(payload capacity): 삽입되어지는 정보의 최대량
- 2) 화질: 삽입된 영상의 화질 정도
- 3) 복잡성: 알고리즘의 복잡도

본 논문에서는 히스토그램 이동을 이용하여 좋은 화질을 유지하면서도 높은 페이로드량을 가지는 리버서블 워터마킹을 제안한다. 제안한 방법은 영상을 2x2 크기의 블록으로 분할하고, 각 블록에서 기준 화소를 정하여, 수평, 수직, 대각에 대한 3개 화소쌍의 차분값을 이용하여 워터마크를 삽입한다. 이는 기존

의 이웃한 화소쌍을 선택하여 삽입하는 방법보다 최대 15배의 워터마크 삽입공간을 더 만들어 낼 수 있다. 또한, 워터마크가 가장 많이 삽입될 수 있는 최대 삽입공간을 영상에 따라 적응적으로 구하여, 히스토그램 이동과 워터마크 삽입에 이용함으로써, 페이로드량을 최대화하였다. 또한 최대 삽입공간을 포함하는 위치맵(location map)을 이진 비트스트림(binary bitstream)으로 변환하여 영상에 워터마크로 삽입함으로써, 오버플로우(overflow)나 언더플로우(underflow)를 제거하고 반복적으로 워터마크를 삽입하여 페이로드량을 증가시켰다.

## 2. 기존의 리버서블 워터마킹 알고리즘

리버서블 워터마킹은 최근 군사용 또는 의료 영상에서 각광 받고 있는 워터마킹의 한 분야로 콘텐츠에 워터마크를 삽입한 후, 콘텐츠 인증 및 무결성 검사와 같은 콘텐츠 인증의 목적으로 사용된다[6]. Fridrich 등[9]은 무손실 압축후 최하위 비트에 데이터를 삽입하는 리버서블 영상 인증 방법을 제안하였다. 한편, Tian[8]은 영상의 화질 왜곡이 적으면서도 고용량의 워터마크를 삽입 방법을 제안하였다. 이는 영상의 차이 값과 평균을 이용하여 영상의 차이를 2배로 확장하고, 확장된 차이 값에 워터마크를 더하여 영상에 삽입한다. 이 방법은 위치맵을 사용하여 오버플로우와 언더플로우를 제거하였으며, 영상의 차이 값을 이용하므로 인접된 화소 간의 차가 작은 저주파 영상에 효율적이다. 그러나 고주파가 많은 영상에서는 화소 간의 차이가 2배로 되면서 화질의 왜곡이 현저히 나타나며 확장 가능한 차이 값을 가지는 화소쌍이 감소하여 페이로드량이 감소한다.

Lee 등[7]은 차영상의 히스토그램 특성을 이용하여 워터마크가 삽입된 영상의 PSNR(Peak Signal To Noise Ratio)이 최소 51.14[dB] 이상을 유지하면서도 보다 많은 워터마크를 삽입하는 방법을 제안하였다. Lee 등은 원영상  $I$ 에서 식 (1)과 같이 이웃한 화소쌍을 선택하고 홀수열에서 짝수열을 빼어 차영상을 생성하였다.

$$D(i, j) = I(i, 2j+1) - I(i, 2j) \quad (1)$$

생성된 차영상  $D$ 에서 -2와 2의 히스토그램 공간을 비우기 위하여 식 (2)와 같이 원영상의 히스토그램을

이동하여 식 (3)의 수정된 차 영상을 생성하였다.

$$\tilde{I}(i, 2j+1) = \begin{cases} I(i, 2j+1) + 1 & \text{if } D(i, j) \geq 2 \\ I(i, 2j+1) - 1 & \text{if } D(i, j) \leq -2 \\ I(i, 2j+1) & \text{otherwise} \end{cases} \quad (2)$$

$$\tilde{D}(i, j) = \tilde{I}(i, 2j+1) - I(i, 2j) \quad (3)$$

위터마크가 1이면 식 (4)와 같이 이웃 화소와의 차분값이 2 또는 -2로 되고 위터마크가 0이면 차분값을 1 또는 -1로 되는 위터마크가 삽입된 영상을 생성한다.

$$I_w(i, 2j+1) = \begin{cases} \tilde{I}(i, 2j+1) + 1 & \text{if } \tilde{D}(i, j) = 1 \& (w=1) \\ \tilde{I}(i, 2j+1) - 1 & \text{if } \tilde{D}(i, j) = -1 \& (w=1) \\ \tilde{I}(i, 2j+1) & \text{otherwise} \end{cases} \quad (4)$$

Lee 등의 방법은 위터마크 삽입공간을 -1 또는 1로 고정하여 영상에 삽입되는 위터마크의 전체 삽입량은 차영상에서 -1과 1의 개수에 의존한다. 이는 대부분의 차분값이 0 주위에 분포하고 있다는 가정에 기반을 둔다. 위터마크 추출은 이웃한 화소쌍에 대해 홀수열에서 짝수열을 뺀 차영상을 구하여 -1 또는 1이면 위터마크 0을 추출하고 -2 또는 2이면 위터마크 1을 추출한다. 히스토그램이 수정된  $I_w$ 를 원영상으로 복원하기 위해 홀수행에서 -2 이하인 화소는 1을 더하고 2 이상인 화소는 1을 빼서 원영상으로 복원한다.

Lee 등의 방법은 위치맵을 사용하지 않고 위터마크의 추출 및 원영상의 복원을 간단한 알고리즘으로 구현하였다. 그러나 이 방법은 차영상에서 이웃 화소쌍에서 -1과 1을 가지는 차분값의 개수에 따라 위터마크의 삽입량이 제한되는 단점이 있다. 또한 히스토그램의 변경시 0 또는 255가 원영상에 없다는 가정하에 알고리즘을 전개하여, 0이나 255가 있는 영상에서는 오버플로우와 언더플로우를 발생시킨다.

Bae 등[1]은 차영상에서 수직과 수평으로 화소쌍을 달리하여 위터마크를 삽입하고, 차영상에서 위터마크를 최대로 삽입할 수 있는 최대 삽입공간을 구하여 위터마크 삽입량을 증가시키는 방법을 제안하였다. 그러나 이 방법은 위터마크의 반복 삽입과 영상에 따라 최대 위터마크 삽입공간이 다르게 생성되므로, 위터마크 추출시 최대 위터마크 삽입공간에 대한 정보가 필요하며, 여전히 오버플로우와 언더플로우를 발생시켜 원영상을 복원하지 못하는 문제점을 가지고 있다.

0의 개수	255의 개수	0의 좌표 (X,Y)	...	255의 좌표 (X,Y)	...
0	9	18	36	...	

그림 1. 위치맵 L의 구조도

Hwang 등[6]은 Lee 등의 방법의 오버플로우와 언더플로우 문제를 해결하기 위해 그림 1과 같은 비트스트림(bitstream)으로 구성되어 있는 위치맵 L을 이용하였다.

이는 차영상의 히스토그램이 식 (2)와 같이 히스토그램을 이동해야 하는 경우, 홀수열의 화소값이 255일 경우에 해당 위치 좌표를 위치맵에 저장하면서 홀수열의 화소값은 변화 시키지 않고 짝수열에서 1을 감소한다. 이 때 짝수열이 화소 값이 0일 경우는 1을 감소하지 않고 0인 화소의 위치를 위치맵에 저장한다. 차영상의 히스토그램이 2이하인 경우는 위의 조건을 반대로 하여 적용하였다. 위치맵 L과 비밀키로 생성된 실제 위터마크 W를 식 (5)와 같이 합한 이진 비트스트림을 생성하여 영상에 위터마크로 반복 삽입하였다.

$$Capacity = LUW \quad (5)$$

그러나, 이는 위터마크가 삽입되지 않는 화소들에 대한 위치맵만을 생성하여, 실제 위터마크가 삽입공간인 차영상의 값이 -1과 1이 되는 위치에 255와 0이 있을 경우에는 여전히 오버플로우와 언더플로우의 문제점을 가진다. 또한 위치맵을 적용하는 방법 외에는 Lee 등의 방법과 동일하여 -1과 1을 가지는 차분값의 개수에 따라 위터마크의 삽입량이 제한되는 단점도 여전히 존재한다.

그러므로 기존의 방법의 단점을 보완하여 위터마크된 영상에서 오버플로우와 언더플로우 현상을 제거하여 원영상을 완벽하게 복원하고, 서로 다른 영상 또는 반복횟수에 따라 적응적으로 최대 위터마크 삽입공간을 저장하고 추출하는 방법이 필요하다.

### 3. 제안한 고용량 리버서블 위터마킹

본 논문에서는 2장에서 제시한 기존의 방법의 단점을 개선시켜 오버플러우 및 언더플러우 문제점을 완벽하게 해결하고 보다 많은 양의 위터마크를 반복 삽입하는 방법을 제안한다. 또한 Bae등의 방법에서

나타나는 최대 삽입공간 값의 저장과 전송으로 워터마크를 추출하는 것이 아니라 최대 삽입공간 값을 이진 비트스트림으로 변환하여 영상에 워터마크로 삽입함으로써, 워터마크 추출시 영상 또는 삽입 반복 횟수에 따른 최대 삽입공간 값을 적용적으로 저장하고 추출하여 원영상을 완벽하게 복원한다.

### 3.1 블록에 기반한 워터마크 삽입

본 논문에서는 워터마크의 삽입량을 증가시키기 위하여 영상을 2x2 크기의 블록으로 분할하고 각각의 블록에서 기준화소와 이웃한 3개의 화소와의 차분 값을 구한다. 즉, 블록의 첫번째 위치 화소를 기준으로 가로, 세로, 대각선 위치에 있는 화소를 두 화소씩 한 쌍으로 하여 한 블록당 1개의 기준화소와 나머지 3개의 화소와 화소쌍을 만든다. 그리고 MxN 크기의 원영상에서 0 ≤ i ≤ M-1과 0 ≤ j ≤ N-1 일 때, 식 (6)과 같은 0.75MN 크기의 1차원 차분배열 D를 생성한다.

$$\begin{aligned} D(k) &= I(2i, 2j+1) - I(2i, 2j) \\ D(k+1) &= I(2i+1, 2j) - I(2i, 2j) \\ D(k+2) &= I(2i+1, 2j+1) - I(2i, 2j) \end{aligned} \quad (6)$$

여기서  $k = 3(i\frac{M}{2} + j)$ 이다. 이는 기존의 이웃 화소 2개를 한쌍으로 하여 4개의 화소에 최대 2개의 워터마크를 삽입할 수 있는 것에 비해 이론적으로 1.5 배의 삽입량을 증가시킨다. 오버플로우와 언더플로우 문제를 해결하기 위해, 기준화소를 제외한 화소에서 255와 0이 나타날 경우, 0 또는 255의 위치 정보를 위치맵에 저장한 후 각각의 화소값을 254와 1로 변경하여 수정된 영상 I'와 차분배열 D'을 생성한다. 이는 Hwang 등의 방법에서 실제 워터마크 삽입 화소에 위치맵을 생성하지 않아 오버플로우와 언더플로우가 발생하는 단점을 보완하여, 히스토그램 이동 이전에 모든 오버플로우와 언더플로우가 발생할 수 있는 화소에 대하여 안정된 값으로 히스토그램을 이동하여 오버플로우와 언더플로우를 제거한다. 위치맵 구조는 1번째 워터마크 삽입에서는 Hwang 등에서 사용한 그림 1의 위치맵 L을 사용하였으며, 비밀키를 이용하여 무작위로 이진 난수를 발생시켜 생성된 실제 워터마크 W를 합하여 원영상에 삽입하였다.

본 논문에서는 -1과 1을 가지는 차분값의 개수에 따라 워터마크의 삽입량이 제한되는 Lee등과 Hwang

등의 단점을 개선하기 위해 Bae 등의 방법에서 사용한 D'의 절대값 빈도를 나타내는 히스토그램 H에서 최대 삽입공간 dmax(1)을 식 (7)과 같이 구하여, D'에 dmax(1)와 -dmax(1)값을 가지는 화소쌍에 워터마크를 삽입하여 페이로드량을 증가시켰다.

$$\begin{aligned} H(|D'(k)|) &= H(|D(k)|) + 1, \quad 0 \leq k < 0.75MN - 1 \\ d_{max}(1) &= \text{argmax}(H(|D'(k)|)), \quad 0 \leq d_{max}(1) \leq 253 \end{aligned} \quad (7)$$

여기서 dmax(1)에서 1의 의미는 1번째 워터마크 삽입을 나타낸다. 이는 기존의 -1 또는 1로 고정된 차분 값을 영상에 따라 적용적으로 최대로 워터마크를 삽입할 수 있는 공간을 찾는 것이다. 히스토그램의 이동과 워터마크 삽입은 구하여진 dmax(1)값을 제외하고는 식 (2)와 (3) 및 (4)의 방법과 동일하며, 워터마크 검출시 dmax(1) 값이 필요하다.

제안한 방법은 워터마크를 반복적으로 삽입해도 원영상의 완벽한 복원 가능하므로 그림 2와 같이 하나의 블록에서 기준화소를 바꾸어 가며 워터마크를 삽입하였다. 그림 2에서 P는 블록내의 화소를 나타내며, 흑색으로 표시된 화소가 기준화소를 나타낸다.

①, ②, ③은 기준화소와 대응되는 화소쌍을 나타내며, 2번째 반복 삽입에서는 P2를 기준화소로 하여 나머지 3개의 화소에 워터마크를 삽입한다. 이는 기준화소를 고정할 경우 블록의 나머지 화소들이 기준화소보다 큰 화소는 계속 증가하고, 기준화소보다 작은 화소는 계속 감소되는 현상을 방지하기 위함이다. 2번째 워터마크 삽입부터는 그림 1의 위치맵 L을 수정하여 n번째 워터마크 삽입시 n-1번째 워터마크 삽입에 사용된 dmax(n-1) 값을 그림 3과 같이 n번째

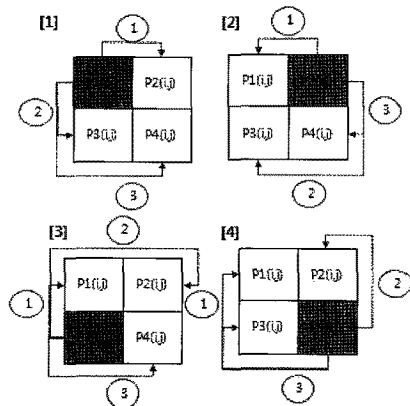


그림 2. 반복 삽입 방법

0의 개수	255개 수	0 좌표	...	255 좌표	...	$dmax(n-1)$
-------	--------	------	-----	--------	-----	-------------

그림 3. 수정된  $n$ 번째 위치맵  $L$ 의 구조도

위치맵에 추가한다.

이는  $n$ 번째 워터마크 추출시, 추출된 워터마크로부터 실제 삽입된 워터마크와 위치맵을 분리하고, 위치맵에서  $n-1$ 번째 워터마크 추출에 필요한  $dmax(n-1)$ 을 추출하여, 다음 워터마크를 반복 추출하는데 사용된다. 그림 3에서  $dmax(n-1)$ 을 저장하기 위한 비트 할당은 대부분의 영상이  $dmax$ 가 7 이하로 나타나므로 실험에서는 3비트로 할당하였다. 그러나 할당 비트수를 많이 하여도 전체 위치맵에서 차지하는 부분은 매우 미소하다. 제안한 방법에서는 워터마크를 삽입할 공간의 개수  $c$ 를  $dmax(n)$ 을 이용하여 미리 계산한 후,  $c$ 에서 수정된 위치맵  $L$ 의 비트길이  $l$ 을 뺀 만큼  $c-l$  비트 길이의 실제 이진 워터마크  $W$ 를 비밀키로 생성하여 원영상에 삽입하였다. 그러므로 영상에 삽입되는 실제 워터마크의 개수, 즉 페이로드는 위치맵 비트 길이를 제외한  $c-l$  비트 길이이다. 제안한 방법은 워터마크를 삽입할 공간의 개수  $c$ 가 위치맵의 비트 길이  $l$ 보다 크다면 무한히 반복하여 워터마크를 삽입할 수 있다. 만약 영상내에 255와 0의 개수가 많이 존재하여,  $c < l$ 이 발생할 경우 워터마크의 반복삽입을 중단한다. 워터마크의 삽입을 중단할 경우에 워터마크 삽입 반복 횟수  $n$ 과  $dmax(n)$ 을 이진 비트스트림  $B$ 로 저장하고 식 (8)과 같이 RSA[10]와 같은 공개키(public key) 암호시스템을 사용하여 암호화한다.

$$S = Z_E(K_p, B) \tag{8}$$

여기서  $K_p$ 는 인증된 사용자의 공개키를 나타내고,  $Z_E$ 는 공개키 암호시스템의 암호화 함수를 나타낸다. 그림 4는 제안한 워터마크 삽입 방법의 전체 흐름도를 나타내었다.

### 3.2 워터마크 추출 및 원영상 복원

워터마크 추출 및 원영상 복원은 워터마크 삽입 순서의 역으로 한다. 먼저 식 (9)와 같이 공개키  $K_p$ 에 대응되는 개인키(private key)  $K_s$ 를 이용하여 복호화함으로써 워터마크 삽입 반복 횟수  $n$ 과 최대  $dmax(n)$ 을 나타내는 비트스트림  $B$ 를 추출한다.

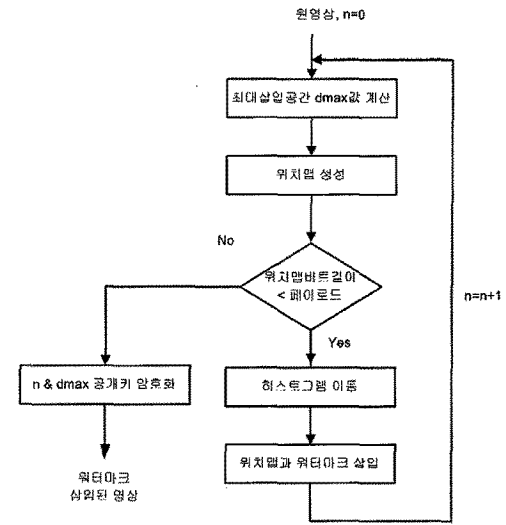


그림 4. 제안한 워터마크 삽입 흐름도

$$B = Z_D(K_s, S) \tag{9}$$

여기서  $Z_D$ 는 공개키 암호시스템의 복호화함수이다. 추출된  $B$ 로부터 반복 회수  $n$ 에 대응되는  $2 \times 2$  블록의 기준화소를 정하고 기준화소와 나머지 3개의 화소와의 차분 값이  $-dmax(n)$ 과  $dmax(n)$ 을 가지는 화소를 만나면 워터마크 0을 추출하고,  $-dmax(n)-1$ 과  $dmax(n)+1$ 을 만나면 1을 추출함으로써 삽입된 워터마크를 간단히 추출한다. 추출된 워터마크로부터 위치맵과 실제 워터마크를 분리하여 생성시 사용된 비밀키를 이용하여 생성된 워터마크와 추출한 워터마크의 상관도를 구하여 워터마크가 동일할 경우 영상을 인증한다. 영상 인증이 수행된 후, 기준화소를 제외한 화소에서  $-dmax(n)$  이하이면 1을 더하고  $dmax(n)$  이상이면 1을 빼서 영상을 1차 복원한다. 다음에는 위치맵에서 0과 255의 위치를 찾아 화소값을 복원하고,  $dmax(n-1)$ 을 추출하여 반복적으로 워터마크를 추출한다.

### 4. 실험 결과

제안한 리버서블 워터마킹 알고리즘의 비 가시성 및 페이로드를 확인하기 위해  $512 \times 512$  크기의 8비트 명암도 영상을 실험영상으로 사용하였다. 표 1에서 그림 1의 사람의 뇌(brain)를 나타내는 의료영상에 대해 Hwang 등[6]의 방법과 제안한 방법으로 6회

표 1. Hwang 등의 방법과 기존의 방법의 성능 비교

반복 횟수	Hwang et al.		Proposed	
	overflow/ underflow [화소 수]	Payload [bits]	overflow/ underflow [화소 수]	Payload [bits]
1	15/1	5,833	0/0	92,564
2	24/3	12,974	0/0	141,087
3	36/4	15,820	0/0	181,985
4	44/4	19,165	0/0	208,871
5	49/4	20,297	0/0	230,539
6	53/4	26,632	0/0	265,018

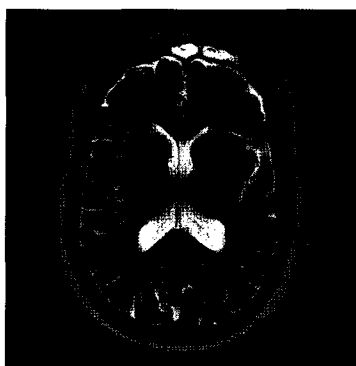


그림 5. 사람의 뇌영상

반복하여 워터마크를 삽입한 실험 결과를 나타내었다. Hwang 등의 방법의 적용은 수직 또는 수평 화소 쌍에 짝수행, 홀수행, 짝수열, 홀수열로 워터마크 삽입 위치를 다르게 하여 반복 삽입하였다. 이는 화소 값이 한쪽 방향으로만 증가 또는 감소되는 것을 방지한다.

표 1의 결과에서 보듯이 Hwang 등의 방법은 인접한 화소의 차영상에서 -1과 1의 개수가 매우 작아 페이로드량이 매우 작게 나타난다.

또한 실제 워터마크 삽입되는 위치에 워치맵을 작성하지 않아 오버플로우와 언더플로우의 개수가 반복 횟수가 많아질수록 증가하는 것을 볼 수 있다. 그러나 제안한 방법은 0과 255의 값을 가지는 화소에 대해 워치맵 저장과 안정된 값으로 화소값을 변경한 후, 히스토그램 이동과 워터마크 삽입을 수행함으로써, 오버플로우와 언더플로우가 없음을 알 수 있다. 또한 반복 삽입시에 최대 삽입이 가능한 공간을 구하여 워터마크함으로써, Hwang 등의 방법보다 10배나

표 2. Tian의 방법과 기존의 방법의 성능 비교

Tian		Proposed	
PSNR [dB]	Payload [bits]	PSNR [dB]	Payload [bits]
44.20	39,566	46.81	56,177
42.86	63,676	43.87	77,896
41.55	84,066	41.88	94,898
40.06	101,089	40.01	112,090
37.66	120,619	38.47	128,125
36.15	141,493	36.16	155,563

많은 페이로드량을 가지는 것을 알 수 있다. 표 2는 Lena 영상에 대해 고용량 워터마킹의 대표적인 방법인 Tian의 방법과 제안한 방법을 적용하여, PSNR과 페이로드량을 비교한 결과이다.

표 2의 결과에서 알 수 있듯이, 제안한 방법이 Tian 등의 방법 보다 객관적인 화질을 나타내는 PSNR 뿐만 아니라 페이로드량에서도 성능이 우수하다는 것을 알 수 있다.

## 5. 결 론

본 논문에서는 기존의 히스토그램 이동을 이용한 리버서블 워터마킹 방법을 개선하여 페이로드량을 증가시키고 오버플로우와 언더플로우를 제거하는 방법을 제안하였다. 제안한 방법은 기존의 2개의 화소에 대해 1개의 삽입 가능 공간을 만드는 것에 비해 4개의 화소에 대해 3개의 삽입 가능 공간을 만들어 페이로드량을 증가시킨다. 또한, 워터마크가 최대로 삽입될 수 있는 최대 삽입공간 값을 구하고, 이를 이진 비트스트림으로 변환하여 영상에 워터마크로 삽입하여 워터마크 반복 삽입을 가능하게 하였다. 제안한 방법에서는 여러번의 반복 삽입시, 마지막 삽입에서 반복 횟수와 최대 삽입공간을 공개키로 암호화한다. 이는 인증된 상대에 대해서만 워터마크 추출을 허용하고, 추출된 워터마크로부터 다음 반복 추출에 필요한 최대 삽입공간을 구하여 워터마크의 반복 추출과 원영상의 완벽한 복원 가능하게 한다.

실험을 통하여 제안한 방법이 기존의 리버서블 워터마킹 방법보다 좋은 화질과 높은 페이로드량을 나타내는 것을 확인하였다.

참 고 문 헌

[1] 배성호, 이경환, “다중 히스토그램 쉬프팅을 이용한 리버서블 워터마킹,” 한국화상학회지, 제 14권, 제3호, pp. 194-200, 2008.

[2] I. J. Cox, Joe Kilian, F. T. Leighton, and T. Shamoon, “Secure Spread Spectrum Watermarking for Multimedia,” *IEEE Trans. on Image Processing*, Vol.6, No.12, pp. 1673-1687, 1997.

[3] A. Nikolaidis and I. Pitas, “Region-based Image Watermarking,” *IEEE Trans. on Image Processing*, Vol.10, No.11, pp. 1726-1740, 2001.

[4] H. C. Huang, F. H. Wang and J. S. Pan, “Multipurpose Image Watermarking Algorithm Based on Multistage Vector Quantization,” *IEICE Trans. Fund.*, Vol.E85-A, No.7, pp. 1719-1726, 2002.

[5] Z. M. Lu, D. G. Xu and S. H. Sun, “A VQ-based Robust Multi-watermarking Algorithm,” *IEEE Trans. on Image Processing*, Vol.14, No.6, pp. 822-831, 2005.

[6] 황진하, 김종원, 최종욱, “개선된 히스토그램 쉬프팅 기법을 이용한 리버서블 워터마킹,” 한국정보처리학회 논문지, 제14-C권, 제1호, pp. 39-44, 2007.

[7] Sang-Kwang Lee, Young-Ho Suh, and Yo-Sung

Ho, “Reversible Image Authentication Based on Watermarking,” *Multimedia and Expo. 2006 IEEE International Conference*, pp. 1321-1324, 2006.

[8] J. Tian, “Reversible Data Embedding Using a Difference Expansion,” *IEEE Trans. on Circuits Systems and Video Technology*, Vol. 13, No.8, pp. 890-896, 2003.

[9] J. Fridrich, M. Goljan, and R. Du, “Invertible Authentication,” *Proc. SPIE, Security and Watermarking of Multi-media Contents*, pp. 197-208, 2001.

[10] William Stallings, *Cryptography and Network Security: Principles and Practice*, Prentice Hall, 2005.



배 성 호

1991년 2월 경북대학교 전자공학과 학사  
 1993년 2월 경북대학교 전자공학과 석사  
 1997년 8월 경북대학교 전자공학과 박사  
 1998년 8월~1999년 8월 삼성전자 책임연구원  
 1999년 9월~현재 동명대학교 의용공학과 부교수  
 관심분야 : 영상처리, 신호처리, 컴퓨터비전