Journal of Korea Multimedia Society Vol. 15, No. 5, May 2012(pp. 639-650)

모듈러 연산과 히스토그램 이동에 기반한 새로운 가역 정보 은닉 기법

김대수†, 유기영††

요 약

Tsai 등은 2009년에 예측 코딩과 히스토그램 이동 기법을 이용한 가역 은닉 방법을 제안하였다. Tsai 등의 방법은 비밀 정보를 숨길 수 있는 양을 향상시키기 위해 예측 코딩을 이용하고, 히스토그램을 두 개 생성하였다. 하지만, 예측 코딩은 삽입과정에서 기준 픽셀이 사용되지 않고, 히스토그램을 두 개로 나누는 방법은 한 블록마다 두 쌍의 최대값과 최소값이 생성되기 때문에 기존 히스토그램 이동 기법보다 많은 양의 추가 전송 데이터가 발생하게 된다. 이러한 문제점을 해결하기 위해 본 논문에서는 모듈러 연산과 히스토그램 이동에 기반한 새로운 가역 정보 은닉 기법을 제안한다. 실험결과를 통해 제안한 방법의 비밀 정보를 숨길 수 있는 양은 Tsai 등의 방법보다 28% 증가 하면서도 추가 전송 데이터의 양은 71% 감소하는 것을 볼 수 있었다.

A novel Reversible Data Hiding Scheme based on Modulo Operation and Histogram Shifting

Dae-Soo Kim†, Kee-Young Yoo††

ABSTRACT

In 2009, Tsai et al. proposed reversible image hiding scheme using linear prediction coding and histogram shifting. Tsai et al.'s scheme improved the hiding capacity of Ni et al.'s scheme by using the prediction coding and two histograms. However, Tsai et al.'s scheme has problems. In the prediction coding, the basic pixel is not used from embedding procedure. Many additional communication data are generated because two peak and zero point pairs are generated by each block. To solve the problems, this paper proposes a novel reversible data hiding scheme based on modulo operation and histogram shifting. In experimental results, the hiding capacity was increased by 28% than Tsai et al.'s scheme. However, the additional communication data was decreased by 71%.

Key words: Steganography(스테가노그래피), Reversible data hiding(가역 정보 은닉), Histogram shifting(히스토그램 이동기법), Modulo operation(모듈러 연산)

1. 서 론

인터넷 보급이 확산되면서 동영상, 음악, 이미지 등 다양한 디지털 컨텐츠(digital contents)들의 유통과 소비가 증가하였고, 디지털 비밀 정보(secret data)를 안전하게 전달하기 위한 정보보호 기술이 필요

※ 교신저자(Corresponding Author) : 유기영, 주소 : 대구광역시 북구 산격 3동 경북대학교 공대 9호관 515호(708-701), 전화 : 053)950-5553, FAX : 053)957-4846, E-mail : yook@knu.ac.kr
접수일 : 2011년 11월 10일, 수정일 : 2012년 1월 25일
완료일 : 2012년 3월 13일

† 정회원, 경북대학교 정보보호학과 (stairways@infosec.knu.ac.kr)
†† 정회원, 경북대학교 IT 대학 컴퓨터학부
※ 본 논문은 2011년도 두뇌한국21사업에 의한 지원과 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2012008348)

하게 되었다. 이러한 정보보호 기술은 목적에 따라 암호(cryptography) 기법과 정보 은닉(data hiding) 기법으로 구분된다. 암호 기법은 데이터가 제 3자에게 누설되거나 조작되는 것을 방지하기 위해 데이터를 암호화(encryption) 하여 데이터의 내용을 알지 못하게 전송한다. 정보 은닉 기법은 제 3자가 디지털 컨텐츠에 비밀 정보를 삽입된 것을 알지 못하게 비밀 정보를 디지털 컨텐츠에 삽입하여 공개된 네트워크에서 전송한다.

디지털 이미지에서의 정보 은닉 기법은 비밀 정보를 삽입할 때 원본 이미지(cover image)의 픽셀을 변경하기 때문에 비밀 정보가 삽입된 이미지(stego image)의 왜곡이 발생한다. 왜곡을 개선하기 위해 여러 방법들이 제안되었지만[1-5], 인간의 시각(human visual system)으로 감지 할 수 없는 왜곡은 여전히 존재한다. 이런 왜곡은 의료 영상, 군용 영상, 예술작품 등의 특정한 분야에서 민감하게 작용한다. 의료 영상의 경우 작은 왜곡에도 의사의 오진으로인해 환자의 생명과 연결되게 된다. 이러한 문제를 해결하기 위해 손실 없는(lossless) 삽입 방법을 이용하여 원본 이미지에 비밀 정보를 삽입하고, 추출하는 과정에서 원본 이미지를 완벽하게 복원할 수 있는 가역 정보 은닉(reversible data hiding) 방법이 제안되었고, 활발히 연구되었다[6-11].

2006년 Ni 등은 히스토그램(histogram)을 이용한 새로운 가역 정보 은닉 방법을 제안하였다[12]. 히스토그램은 도수분포를 나타낼 때 계급을 밑변으로 하고, 직사각형의 면적이 그 계급의 도수에 비례하도록 그린 기둥모양의 그래프이다. 이미지 히스토그램에서는 이미지의 픽셀값(pixel value)과 픽셀값의 빈도(frequency of pixel value)를 각각 가로축과 세로축으로 나타낸다. 히스토그램 이동 방법은 이미지 히스토그램에서 빈도가 가장 많은 픽셀의 이동을 통하여 비밀 정보를 삽입하는 방법이다.

Ni 등의 방법에서 비밀 정보를 삽입하는 양을 증가시키기 위하여 2009년 Tsai 등은 예측 코딩과 히스토그램 이동을 이용한 가역 정보 은닉 방법을 제안하였다[13]. Tsai 등의 방법은 이미지를 블록으로 나누어 블록의 가운데 값과 블록의 나머지 픽셀들의 차를 구하는 예측코딩과 히스토그램을 두 개로 나누어 한 블록에 두 쌍의 최대값(peak point)과 최소값(zero point) 을 생성하여 비밀 정보를 삽입하는 방법이다. 하지만, Tsai 등의 방법인 예측 코딩(prediction coding)은 블록의 모든 픽셀이 삽입과정에 사용되지 않아 비밀 정보를 숨길 수 있는 양(capacity)이 감소되며, 두 개의 히스토그램을 사용하는 방법은 블록마다 두 쌍의 최대값과 최소값이 생성되기 때문에, 많은 양의 추가 전송 데이터가 발생하는 문제점이 있다.

본 논문에서는 이러한 문제점을 개선하기위해서 블록 기반의 역 S-예측코딩을 사용하여 블록의 모든 픽셀을 삽입과정에서 사용할 수 있도록 하여 비밀 정보를 숨길 수 있는 양을 증가 시켰으며, 추가 전송 데이터의 양을 줄이기 위하여 모듈러 연산(modulo operation)을 사용한다. 제안한 방법의 실험 결과, 비밀 정보를 숨길 수 있는 양은 Tsai 등의 방법보다 28% 증가 하면서도 추가 전송 데이터의 양은 71% 감소하는 것을 볼 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 제안한 방법과 관련된 연구로 예측 코딩과 히스토그램 이동을 이용한 가역 정보 은닉 방법과 문제점을 간략히 살펴보고, 3장에서는 제안한 방법의 삽입 과정과 추출 및 복원 과정에 대해서 자세히 살펴본다. 4장에서는 실험 결과를 분석하고, 5장에서는 결론을 도출한다.

2. 관련연구

2.1 히스토그램 기반의 가역 정보 은닉 방법

Ni 등은 2006년에 히스토그램의 최대값과 최소값을 이용한 히스토그램 기반의 가역 정보 은닉 방법을 최초로 제안하였다[12]. 제안한 방법에서는 원본 이미지의 히스토그램에서 빈도가 가장 큰 값을 최대값으로, 빈도가 0이거나 가장 작은 값을 최소값으로 선택을 하고, 최대값에 비밀 정보를 삽입한다. 최대값과 최소값 쌍은 최대 3개까지 생성 가능하다. 원본 이미지 픽셀의 값이 최대값과 같고, 비밀 정보가 0이면 픽셀 값을 변경하지 않고, 비밀 정보가 1이면 픽셀 값을 최소값 쪽으로 1만큼 이동하여 비밀 정보가 삽입된 이미지를 생성한다. 추출 과정에서는 삽입 과정에서 사용한 최대값과 최소값을 이용하여 이미지에서 비밀 정보를 추출하고, 복원된 원본 이미지(recovered original image)를 생성한다. Ni 등이 제안한 히스토그램 이동 방법은 픽셀의 값이 최대값과 같거나 최소값과 최대값의 범위 안에 있을 경우, 1만큼의 픽셀값이 변경되기 때문에 다른 가역 정보 은닉 방법

보다 이미지의 왜곡이 적다는 장점이 있다. 하지만, 비밀 정보를 숨길 수 있는 양이 최대값의 크기에 결정되는 문제점을 가지고 있다. 이후에 제안되는 히스토그램 기반의 가역 정보 은닉 방법은 대부분, 이미지의 품질(PSNR: Peak signal to noise ratio)을 유지하면서 비밀 정보를 숨길 수 있는 양을 증가시키는데 중점을 두고 있다[11,13].

2.2 예측 코딩과 히스토그램 이동을 이용한 가역 정보 은닉 방법

Tsai 등은 블록에서 예측코딩과 히스토그램 이동을 이용한 가역 정보 은닉 방법을 제안하였다[13]. Tsai 등의 방법은 기존 Ni 등의 방법보다 비밀 정보를 삽입할 수 있는 양을 증가하기 위해 원본 이미지를 블록으로 나누고 n번째 블록 $B^{(n)}$의 기준 픽셀(basic pixel) $r^{(n)}$과 나머지 픽셀 값 $c_{(i,j)}^{(n)}$의 차이를 통해 잔여 값(residual value)을 구하는 예측코딩(linear prediction coding)을 수행한다. 또한, 잔여 값을 이용하여 음의 히스토그램(non-negative histogram)과 양의 히스토그램(negative histogram)을 생성하여 두 히스토그램에서 각각의 최대값과 최소값 쌍 ($P_+^{(n)}$, $Z_+^{(n)}$), ($P_-^{(n)}$, $Z_-^{(n)}$) 을 찾아 두 개의 최대값에 삽입과정을 진행하여 비밀 정보를 삽입할 수 있는 양을 증가시켰다. 비밀 정보가 삽입되어 변경된 잔여 값 $e'^{(n)}_{(i,j)}$에 역 예측코딩(inverse linear prediction coding)을 수행하여 비밀 정보가 삽입된 이미지를 생성한다. 추출 과정에서는 각 블록의 최대값과 최소값을 이용하여 이미지에서 비밀 정보를 추출하고, 복원된 원본 이미지를 생성한다.

2.2.1 삽입과정

Input: $N\times N$ 픽셀의 원본 이미지 C 와 픽셀 값 $c_{(i,j)}^{(n)}$, 블록 크기 M, 비밀 정보 d_l

Output: $N\times N$ 픽셀의 비밀정보가 삽입된 이미지 S와 픽셀 값 $s_{(i,j)}^{(n)}$, 각 블록의 최대값 최소값 쌍 ($P_+^{(n)}$, $Z_+^{(n)}$), ($P_-^{(n)}$, $Z_-^{(n)}$)

Step 1: 원본 이미지 C 를 $M\times M$의 블록으로 나눈다.

Step 2: 블록 $B^{(n)}$의 가운데 위치한 픽셀을 기준 픽셀 $r^{(n)}$로 선택하고 블록안의 나머지 픽셀 값 $c_{(i,j)}^{(n)}$을 다음의 식 (1)으로 잔여 값 $e_{(i,j)}^{(n)}$을 계산한다.

$$e_{(i,j)}^{(n)} = c_{(i,j)}^{(n)} - r^{(n)} \tag{1}$$

Step 3: 기준 픽셀 $r^{(n)}$을 제외한 잔여 값 $e_{(i,j)}^{(n)}$을 이용하여 음의 히스토그램과 양의 히스토그램을 생성한다.

Step 4: 각 히스토그램에서 최대값과 최소값을 찾는다.

Step 5: 다음의 조건에 따라 비밀 정보를 삽입한다.

- $e_{(i,j)}^{(n)} = P_+^{(n)}$ 또는 $e_{(i,j)}^{(n)} = P_-^{(n)}$ 이면, 비밀 정보를 삽입하고 다음 식 (2)에 따라 $e_{(i,j)}^{(n)}$을 변경한다.

$$e'^{(n)}_{(i,j)} = \begin{cases} e_{(i,j)}^{(n)}, & \text{if } d_l = 0, \\ e_{(i,j)}^{(n)} + 1, & \text{if } d_l = 1 \text{ and } P_+^{(n)} < Z_+^{(n)} \text{ or } P_-^{(n)} < Z_-^{(n)}, \\ e_{(i,j)}^{(n)} - 1, & \text{if } d_l = 1 \text{ and } P_+^{(n)} > Z_+^{(n)} \text{ or } P_-^{(n)} > Z_-^{(n)} \end{cases} \tag{2}$$

- $e_{(i,j)}^{(n)} \neq P_+^{(n)}$ 또는 $e_{(i,j)}^{(n)} \neq P_-^{(n)}$ 이면, 비밀 정보를 삽입하지 않고 다음 식 (3)에 따라 $e_{(i,j)}^{(n)}$을 변경한다.

$$e'^{(n)}_{(i,j)} = \begin{cases} e_{(i,j)}^{(n)} + 1, & \text{if } P_+^{(n)} < e_{(i,j)}^{(n)} < Z_+^{(n)} \text{ or } P_-^{(n)} < e_{(i,j)}^{(n)} < Z_-^{(n)}, \\ e_{(i,j)}^{(n)} - 1, & \text{if } P_+^{(n)} > e_{(i,j)}^{(n)} > Z_+^{(n)} \text{ or } P_-^{(n)} > e_{(i,j)}^{(n)} > Z_-^{(n)}, \\ e_{(i,j)}^{(n)}, & otherwise \end{cases} \tag{3}$$

Step 6: 변경된 잔여 값 $e'^{(n)}_{(i,j)}$을 이용하여 다음 식 (4)으로 역 예측 코딩을 수행하고, 비밀 정보가 삽입된 이미지 S를 얻는다.

$$s_{(i,j)}^{(n)} = e'^{(n)}_{(i,j)} + r^{(n)} \tag{4}$$

2.2.2 추출 및 복원 과정

Input: $N\times N$ 픽셀의 비밀정보가 삽입된 이미지 S와 픽셀 값 $s_{(i,j)}^{(n)}$, 블록 크기 M, 각 블록의 최대값 최소값 쌍 ($P_+^{(n)}$, $Z_+^{(n)}$), ($P_-^{(n)}$, $Z_-^{(n)}$)

Output: $N\times N$ 픽셀의 복원된 원본 이미지 C와 픽셀 값 $c_{(i,j)}^{(n)}$, 비밀 정보 d_l

Step 1: 비밀정보가 삽입된 이미지 S를 $M\times M$ 의 블록으로 나눈다.

Step 2: 기준 픽셀 $r^{(n)}$과 블록의 픽셀 값 $s_{(i,j)}^{(n)}$을 이용, 다음의 식 (5)으로 변경된 잔여 값 $e'^{(n)}_{(i,j)}$을 계산한다.

$$e'^{(n)}_{(i,j)} = s_{(i,j)}^{(n)} - r^{(n)} \tag{5}$$

Step 3: 각 블록의 최대값 최소값 쌍 ($P_+^{(n)}$, $Z_+^{(n)}$),

($P_{-}^{(n)}$, $Z_{-}^{(n)}$)과 다음의 조건을 이용하여 비밀 정보를 추출하고 잔여 값을 복원한다.

- $P_{+}^{(n)} \geq e'^{(n)}_{(i,j)} \geq Z_{+}^{(n)}$ 또는 $P_{-}^{(n)} \geq e'^{(n)}_{(i,j)} \geq Z_{-}^{(n)}$ 이면 다음의 식 (6), (7)을 따른다.

$$d_l = \begin{cases} 0, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)}, \\ 1, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} - 1 \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)} - 1 \end{cases} \tag{6}$$

$$e_{(i,j)}^{(n)} = \begin{cases} e'^{(n)}_{(i,j)}, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)}, \\ e'^{(n)}_{(i,j)} + 1, & otherwise \end{cases} \tag{7}$$

- $P_{+}^{(n)} \leq e'^{(n)}_{(i,j)} \leq Z_{+}^{(n)}$ 또는 $P_{-}^{(n)} \leq e'^{(n)}_{(i,j)} \leq Z_{-}^{(n)}$ 이면 다음의 식 (8), (9)을 따른다.

$$d_l = \begin{cases} 0, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)}, \\ 1, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} + 1 \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)} + 1 \end{cases} \tag{8}$$

$$e_{(i,j)}^{(n)} = \begin{cases} e'^{(n)}_{(i,j)}, & \text{if } e'^{(n)}_{(i,j)} = P_{+}^{(n)} \quad \text{or} \quad e'^{(n)}_{(i,j)} = P_{-}^{(n)}, \\ e'^{(n)}_{(i,j)} - 1, & otherwise \end{cases} \tag{9}$$

Step 4: 복원한 잔여 값을 이용하여 다음 식 (10)으로 원본 이미지를 생성한다.

$$c_{(i,j)}^{(n)} = e_{(i,j)}^{(n)} + r^{(n)} \tag{10}$$

2.2.3 Tsai 등이 제안한 방법의 문제점

Tsai 등의 방법에서 삽입과정 중 예측코딩을 수행하는 Step 2는 블록의 기준 픽셀 $r^{(n)}$과 나머지 픽셀 값 $c_{(i,j)}^{(n)}$의 차이를 이용하여 잔여 값을 계산한다. 이러한 과정에서 기준 픽셀이 변경되지 않아야 복원과정에서 원본 이미지를 복원할 수 있기 때문에 삽입과정에서 기준 픽셀은 사용되지 않는다. 블록마다 한 픽셀을 삽입과정에 사용하지 않기 때문에 비밀 정보를 숨길 수 있는 양이 감소한다.

또한, 삽입과정 Step 4 에서 한 블록의 음의 히스토그램, 양의 히스토그램에서 각각 한 쌍씩의 최대값과 최소값을 찾는다. 이 두 쌍의 최대값과 최소값들은 비밀정보를 추출하고 원본 이미지를 복원하는데 사용된다. 각 블록마다 두 쌍의 최대값과 최소값이 생성되기 때문에 기존 히스토그램 이동 기법보다 많은 양의 추가 전송 데이터가 발생한다.

2.3 블록 기반의 역 S-예측 코딩

Tsai 등의 방법 중 예측코딩은 각 블록의 가운데 픽셀을 삽입과정에서 사용하지 않기 때문에 비밀 정보를 숨길 수 있는 양이 감소하게 된다. 이를 해결하기위해 블록 기반의 역 S-예측코딩을 2010년 Kim 등이 제안하였다[14].

블록 기반의 예측코딩은 블록 내의 모든 픽셀을 삽입 과정에서 사용하기 위해서 역 S-순서(inverse S-order) 방법을 블록에 적용하였다. 그림 (1)과 같이 첫 번째 블록을 제외한 나머지 블록에서 역 S-순서로 이전 블록 $B^{(n-1)}$의 가장 가까운 행 또는 열의 가운데 위치한 값을 기준 픽셀 $b^{(n)}$을 선택한 다음, 각 블록의 픽셀 $c_{(i,j)}^{(n)}$ 과 기준 픽셀 $r^{(n)}$ 의 차이값을 식 (11)을 이용해 계산한다.

$$e_{(i,j)}^{(n)} = c_{(i,j)}^{(n)} - b^{(n)} \tag{11}$$

추출 및 복원 과정에서는 하나의 블록에서 모든 추출과정과 복원 과정이 완료된 후, 복원된 이전 블록에서 기준 픽셀을 선택하여 차이값을 구하고 추출과정과 복원과정을 수행한다.

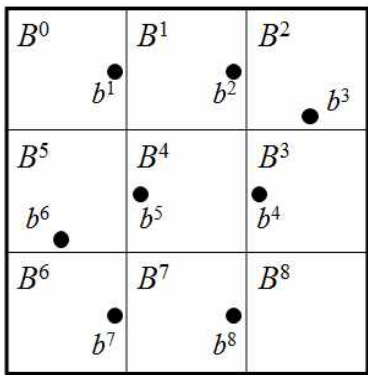

그림 1 블록 $B^{(n)}$의 순서와 기준 픽셀 $b^{(n)}$의 위치

3. 제안한 방법

Tsai 등의 방법 중 예측코딩의 문제점을 해결하기 위해 블록 기반의 역 S-예측코딩을 사용하여 블록의 모든 픽셀을 삽입과정에서 사용할 수 있도록 하였다. 또한, 추가 전송 데이터의 양을 줄이면서 비밀 정보를 숨길 수 있는 양을 증가시키기 위해 모듈러 연산(modulo operation)을 사용하였다.

제안하는 방법은 원본 이미지에 블록 기반의 역 S-예측코딩을 수행하여 블록 $B^{(n)}$에서 잔여 값 $e_{(i,j)}^{(n)}$을 계산한다. 이후, 잔여 값(residual value)에 모듈러 연산을 수행하여 잉여 값(residue value) $k_{(i,j)}^{(n)}$을 계산

하고, 히스토그램을 생성한다. 히스토그램은 모듈러 $t^{(n)}$에 관한 표준 잉여계(standard residue system)인 집합 $\mathbb{Z}_t = \{0, 1, \cdots, t-1\}$ 의 범위 안에서 생성이 되며, 모듈러 $t^{(n)}$을 최소값이 존재할 때까지 최초 2부터 1씩 증가시킨다. 빈도수가 0인 최소값이 존재하면, 최대값 $P^{(n)}$과 최소값 $Z^{(n)}$을 한 블록 당 한 쌍을 찾아 최대값에 비밀정보를 삽입한다. 추출 및 복원 과정에서 비밀 정보가 삽입된 이미지에 블록 기반의 역 S-예측코딩을 수행하여 변경된 잔여 값 $e'^{(n)}_{(i,j)}$ 을 계산한 후, $t^{(n)}$로 잉여 값 $k^{(n)}_{(i,j)}$을 모듈러 연산하여 최대값 $P^{(n)}$과 최소값 $Z^{(n)}$을 이용하여 비밀 정보를 추출하고 원본 이미지를 복원한다.

그림 2는 3×3 블록에서 모듈러 연산을 이용한 히스토그램 생성의 예를 보여주고 있다. 최초 모듈러의 값을 2로 잉여 값을 계산하고 $\mathbb{Z}_2 = \{0, 1\}$ 로 히스토그램을 생성한다. 빈도수가 0인 최소값이 존재하지 않기 때문에 $t^{(n)}$을 1 증가하여 잉여 값을 계산한다. 모듈러 3으로 계산 후, $\mathbb{Z}_3 = \{0, 1, 2\}$ 로 히스토그램을 생성한다. 모듈러 3의 경우에도 빈도수가 0인 최소값이 존재하지 않기 때문에 다시 $t^{(n)}$을 1 증가하여 모듈러 4로 계산하고 $\mathbb{Z}_4 = \{0, 1, 2, 3\}$ 으로 히스토그램을 생성한다. 모듈러 4의 경우에는 빈도수가 0인 최소값이 존재하기 때문에 모듈러 $t^{(n)}$의 값은 4가 되며, 최대값 $P^{(n)}$는 1이고, 최소값 $Z^{(n)}$은 0이 된다.

3.1 삽입과정

이 절에서는 제안한 방법의 삽입과정에 대해서 설명한다. 그림 3은 삽입과정의 흐름도이다. 원본 이미지에 블록 기반의 역 S-예측코딩을 수행하여 잔여 값 $e^{(n)}_{(i,j)}$을 계산하고, 빈도수가 0인 최소값이 존재하는 $\mathbb{Z}_t$의 히스토그램을 찾을 때 까지 $t^{(n)}$을 1증가시키면서 모듈러 연산을 하여 블록 $B^{(n)}$의 히스토그램을 생성한다. 최대값과 최소값을 찾은 후에 히스토그램 이동기법으로 최대값에 비밀 정보를 삽입하고 비밀 정보가 삽입된 이미지를 얻는다. 다음은 삽입과정의 알고리즘이다.

Input: $N \times N$ 픽셀의 원본 이미지 C 와 픽셀 값 $c^{(n)}_{(i,j)}$, 블록 크기 M, 비밀 정보 d_l

Output: $N \times N$ 픽셀의 비밀정보가 삽입된 이미지 S 와 픽셀 값 $s^{(n)}_{(i,j)}$, 각 블록의 최대값 최소값 쌍 ($P^{(n)}$, $Z^{(n)}$), 각 블록의 모듈러 $t^{(n)}$

Step 1: 원본 이미지 C 를 $M \times M$ 의 블록으로 나

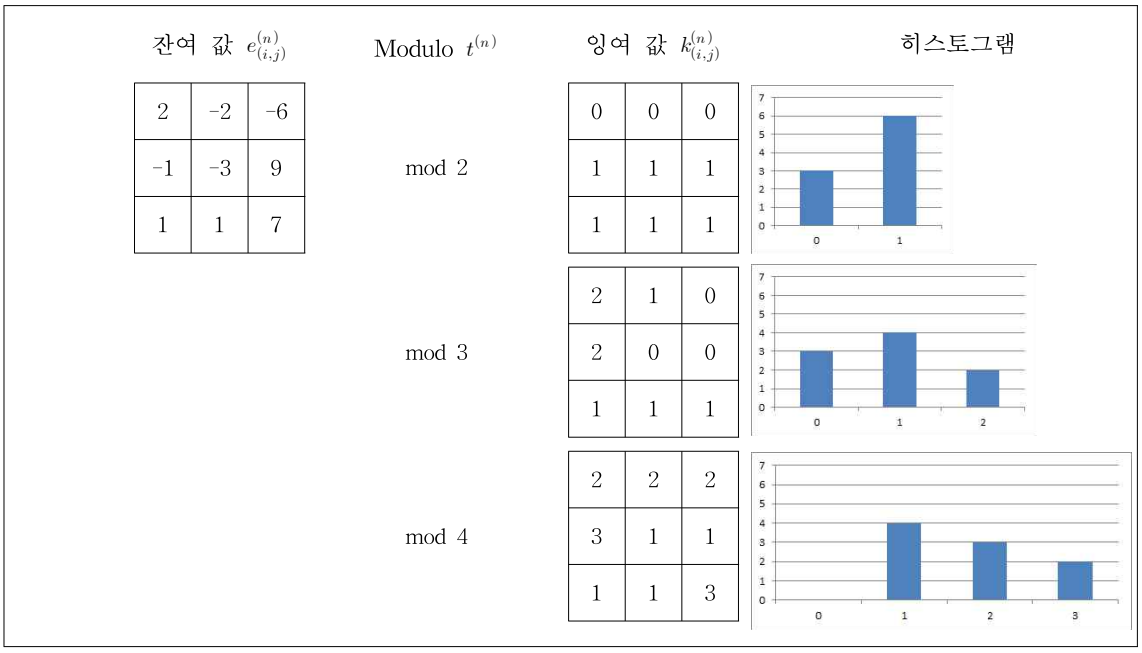

그림 2. 3×3 블록에서 모듈러 연산을 이용한 히스토그램 생성의 예

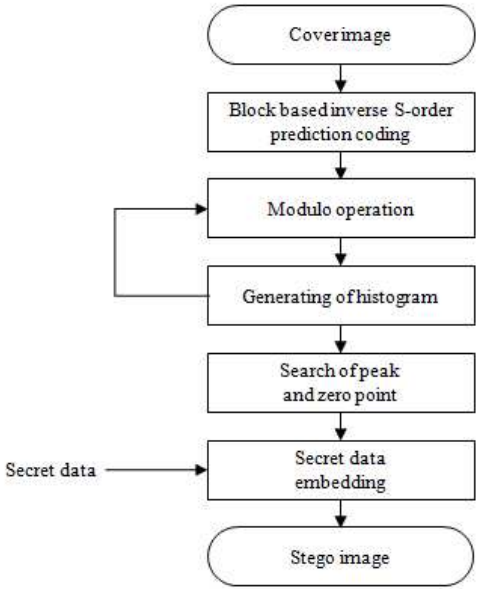

그림 3. 삽입과정의 흐름도

눈다.

Step 2: 블록 기반의 역 S-예측코딩을 수행하여 기준 픽셀 $b^{(n)}$과 블록의 픽셀 값 $c_{(i,j)}^{(n)}$으로 다음의 식 (12)을 이용하여 잔여 값 $e_{(i,j)}^{(n)}$을 계산한다.

$$e_{(i,j)}^{(n)} = c_{(i,j)}^{(n)} - b^{(n)} \tag{12}$$

Step 3: 잔여 값 $e_{(i,j)}^{(n)}$에 다음 식 (13)을 이용하여 잉여 값 $k_{(i,j)}^{(n)}$을 계산한 다음 블록 $B^{(n)}$ 의 히스토그램을 집합 $\mathbb{Z}_t$의 범위에서 생성한다. $t^{(n)}$은 최초 2이며, 빈도수가 0인 최소값이 존재할 때 까지 $t^{(n)}$을 1씩 증가시킨다.

$$k_{(i,j)}^{(n)} = e_{(i,j)}^{(n)} \bmod t^{(n)} \tag{13}$$

Step 4: 최소값 $Z^{(n)}$이 존재하는 히스토그램에서 최대값 $P^{(n)}$을 찾는다.

Step 5: 다음의 조건에 따라 비밀 정보를 삽입하고 비밀정보가 삽입된 이미지를 얻는다.

- $e_{(i,j)}^{(n)} = P^{(n)}$ 이면, 비밀 정보를 삽입하고 다음 식 (14)에 따라 $c_{(i,j)}^{(n)}$을 변경한다.

$$s_{(i,j)}^{(n)} = \begin{cases} c_{(i,j)}^{(n)}, & \text{if } d_l = 0, \\ c_{(i,j)}^{(n)} + 1, & \text{if } d_l = 1 \text{ and } P^{(n)} < Z^{(n)}, \\ c_{(i,j)}^{(n)} - 1, & \text{if } d_l = 1 \text{ and } P^{(n)} > Z^{(n)} \end{cases} \tag{14}$$

- $e_{(i,j)}^{(n)} \neq P^{(n)}$ 이면, 비밀 정보를 삽입하지 않고 다음 식 (15)에 따라 $c_{(i,j)}^{(n)}$을 변경한다.

$$s_{(i,j)}^{(n)} = \begin{cases} c_{(i,j)}^{(n)} + 1, & \text{if } P^{(n)} < e_{(i,j)}^{(n)} < Z^{(n)}, \\ c_{(i,j)}^{(n)} - 1, & \text{if } P^{(n)} > e_{(i,j)}^{(n)} > Z^{(n)}, \\ c_{(i,j)}^{(n)}, & otherwise \end{cases} \tag{15}$$

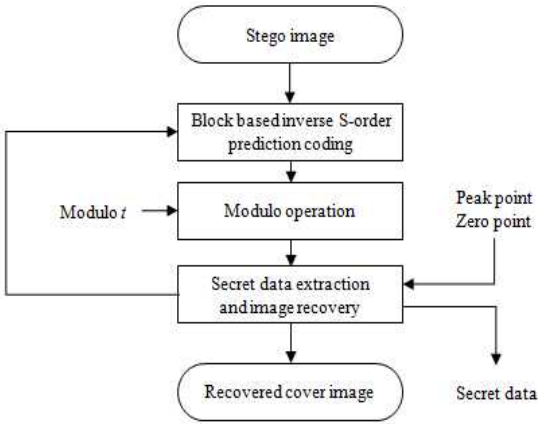

그림 4. 추출 및 복원 과정의 흐름도

3.2 추출 및 복원 과정

이 절에서는 추출 및 복원과정에 대해서 설명한다. 그림 4는 추출 및 복원 과정의 흐름도이다. 비밀정보가 삽입된 이미지에 블록 기반의 역 S-예측코딩을 수행하여 변경된 잔여 값 $e'^{(n)}_{(i,j)}$을 계산한다. 삽입과정에서 생성된 모듈러 $t^{(n)}$와 최대값 최소값 쌍 ($P^{(n)}$, $Z^{(n)}$)을 이용하여 블록 $B^{(n)}$에 대한 모듈러 연산을 하고, 히스토그램 이동기법으로 삽입된 비밀 정보를 추출하고 원본 이미지를 복원한다. 추출 및 복원 과정의 알고리즘은 다음과 같다.

Input: $N \times N$ 픽셀의 비밀정보가 삽입된 이미지 S 와 픽셀 값 $s_{(i,j)}^{(n)}$, 블록 크기 M, 각 블록의 최대값 최소값 쌍 ($P^{(n)}$, $Z^{(n)}$), 각 블록의 모듈러 $t^{(n)}$

Output: $N \times N$ 픽셀의 복원된 원본 이미지 C 와 픽셀 값 $c_{(i,j)}^{(n)}$, 비밀 정보 d_l

Step 1: 비밀정보가 삽입된 이미지 S 를 $M \times M$ 의 블록으로 나눈다.

Step 2: 블록 기반의 역 S-예측코딩을 수행하여 첫 번째 블록 또는 복원된 블록의 기준 픽셀 $b^{(n)}$과

블록의 픽셀 값 $s_{(i,j)}^{(n)}$으로 다음의 식 (16)을 이용하여 블록 $B^{(n)}$의 변경된 잔여 값 $e'^{(n)}_{(i,j)}$을 계산한다.

$$e'^{(n)}_{(i,j)} = s^{(n)}_{(i,j)} - b^{(n)} \tag{16}$$

Step 3: 모듈러 $t^{(n)}$을 이용하여 변경된 잔여 값 $e'^{(n)}_{(i,j)}$을 다음의 식 (17)로 모듈러 연산을 수행하여 변경된 잉여 값 $k'^{(n)}_{(i,j)}$을 구한다.

$$k'^{(n)}_{(i,j)} = e'^{(n)}_{(i,j)} \bmod t^{(n)} \tag{17}$$

Step 4: 각 블록의 최대값 최소값 쌍 ($P^{(n)}$, $Z^{(n)}$)과 다음의 조건을 이용하여 비밀 정보를 추출하고 원본 이미지 블록 $B^{(n)}$을 복원한다.

- $P^{(n)} \geq k'^{(n)}_{(i,j)} \geq Z^{(n)}$ 이면 다음의 식 (18), (19)을 따른다.

$$d_l = \begin{cases} 0, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)}, \\ 1, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)} - 1 \end{cases} \tag{18}$$

$$c^{(n)}_{(i,j)} = \begin{cases} s^{(n)}_{(i,j)}, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)}, \\ s^{(n)}_{(i,j)} + 1, & otherwise \end{cases} \tag{19}$$

- $P^{(n)} \leq k'^{(n)}_{(i,j)} \leq Z^{(n)}$ 이면 다음의 식 (20), (21)을 따른다.

$$d_l = \begin{cases} 0, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)}, \\ 1, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)} + 1 \end{cases} \tag{20}$$

$$c^{(n)}_{(i,j)} = \begin{cases} s^{(n)}_{(i,j)}, & \text{if} \quad k'^{(n)}_{(i,j)} = P^{(n)}, \\ s^{(n)}_{(i,j)} - 1, & otherwise \end{cases} \tag{21}$$

Step 5: Step 2, Step 3, Step 4를 마지막 블록까지 반복수행하여 비밀정보를 추출하고 원본 이미지를 복원한다.

3.3 추가 전송 데이터

Tsai 등의 방법과 제안한 방법의 최대값과 최소값 쌍의 개수를 비교해 봤을 때, Tsai 등의 방법은 한 블록 마다 두 개의 최대값 최소값 쌍이 생성되고, 추출 및 복원과정에서 사용된다. 제안한 방법은 한 블록 마다 모듈러 $t^{(n)}$이 필요한 대신, 한 블록마다 하나의 최대값과 최소값 쌍을 이용하여 비밀 정보를 추출하고 원본 이미지를 복원할 수 있다.

그리고, Tsai 등의 방법에서 최대값과 최소값 쌍의 비트크기와 제안한 방법에서 모듈러 $t^{(n)}$와 최대값 최소값의 비트크기를 비교하면, Tsai 등의 방법에서 최대값과 최소값의 범위는 최소 -255에서 최대 255의 값을 가질 수 있다. 512가지의 수를 표현하기 위해서는 최소 9비트가 필요하고, 4개의 값을 모두 표현하기 위해서는 32비트가 필요하다. 하지만, 제안한 방법에서 $t^{(n)}$의 범위는 최소 2에서 최대 $M \times M+1$의 값을 가질 수 있다. 또한, 최대값과 최소값의 범위는 최소 0에서 최대 $t^{(n)}-1$의 값을 가질 수 있으며, 다음의 식(22)을 이용하여 각 블록의 추가 전송 데이터 비트의 크기 $L^{(n)}$를 구할 수 있다.

$$|t^{(n)}| = \lceil \log_2 M \times M+1 \rceil, \; |P^{(n)}| = \lceil \log_2 t^{(n)} \rceil,$$
$$|Z^{(n)}| = \lceil \log_2 t^{(n)} \rceil \quad L^{(n)} = |t^{(n)}| + |P^{(n)}| + |Z^{(n)}| \tag{22}$$

예를 들어, Tsai 등의 방법에서는 3×3블록에서 두 개의 최대값 최소값 쌍을 가지기에 한 블록 마다 32비트의 추가전송 데이터를 가지게 되며, 제안한 방법에서 3×3 블록에서 $t^{(n)}$은 4이고, 최대값 최소값 쌍은 (2, 3)일 때 $t^{(n)}$의 크기는 4비트이고, 최대값과 최소값은 각각 2비트의 크기를 가지기 때문에, 이 블록의 추가 전송 데이터 크기 $L^{(n)}$은 8비트가 된다.

4. 실험 결과

이 장에서는 Tsai 등의 방법과 본 논문에서 제안한 방법의 실험 결과를 비교하여 살펴보겠다. 512×512 크기의 이미지 10개와 의료 이미지 8개를 이용하여 실험을 하였으며, 비밀 정보는 난수(random number)를 생성하여 만들었다. 각 결과는 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡 정도를 나타내는 PSNR으로 나타내었다. PSNR은 다음 식 (23)으로 정의된다[15].

$$PSNR = \left\{ 10 \cdot \log_{10} \left(\frac{255^2}{MSE} \right) \right\} \tag{23}$$

MSE(Mean squared error)는 원본 이미지와 비밀 정보가 삽입된 이미지 간의 평균제곱오차를 뜻하며, 다음 식 (24)으로 구할 수 있다.

$$MSE = \left\{ \frac{1}{MN} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I_{(i,j)} - I'_{(i,j)})^2 \right\} \tag{24}$$

$I_{(i,j)}$와 $I'_{(i,j)}$는 $M \times N$ 크기의 원본 이미지와 비밀

그림 5. 실험에 사용한 이미지

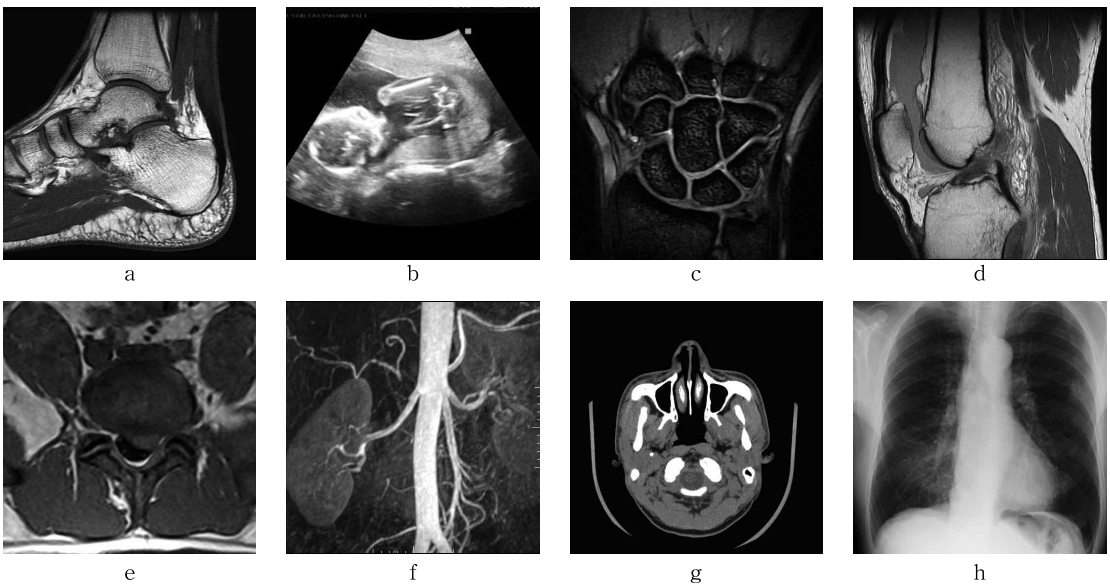

그림 6. 실험에 사용한 의료 이미지

정보가 삽입된 이미지이며, PSNR 값이 클수록 좋은 이미지 품질을 가진다.

표 1에서는 3×3 블록으로 이미지를 나눌 때 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡 정도를 이미지 10개를 이용하여 비교하였다. Lena 이미지에서 실험한 결과 Tsai 등의 방법에서는 93,044 비트를 숨길 수 있고 비밀 정보가 삽입된 이미지의 PSNR 값은 54.14dB 이며, 제안한 방법은 112,339비트를 숨길 수 있고, PSNR 값은 52.18dB 이였다. 두 방법을 비교하였을 때, 비밀 정보를 숨길 수 있는 양은 19,295 비트 만큼 증가하였으며, PSNR 값은 1.96dB 만큼 감소하였다. 하지만, PSNR 값은 30dB 이상일 때 사람의 눈으로는 이미지의 왜곡을 알 수 없어 비밀 정보가 숨겨진 것을 모르기 때문에, 이미지의 왜곡이 적다고 할 수 있다.

평균적인 수치로 봤을 때 비밀 정보를 숨길 수 있는 양은 제안한 방법이 112,426비트로 Tsai 등의 방법보다 25,305비트 증가한 것을 볼 수 있고 PSNR 값은 52.18dB로 2.21bB 만큼 감소하였지만 여전히 50dB 이상으로 이미지의 왜곡이 적다. 또한, Tsai 등의 방법에서는 픽셀 값의 변화가 적은 이미지(smooth image)와 픽셀 값의 변화가 많은 이미지(edge image)에서 비밀 정보를 숨길 수 있는 양의 차이가 1만 비트 이상 나는 것을 볼 수 있지만, 제안한 방법에서는 이미지의 성질에 상관없이 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡이 균등한 것을 볼 수 있었다.

표 2와 표 3은 블록을 다양하게 적용하여 비밀정보를 숨길 수 있는 양과 이미지의 왜곡을 비교하였다. 블록을 작게 할수록 비밀 정보를 삽입할 수 있는

표 1. Tsai 등의 방법과 제안한 방법의 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡정도 비교 (3×3 블록)

Test images	Tsai 등의 방법		제안한 방법	
	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)
Lena	93,044	54.14	112,339	52.18
Airport	80,842	55.19	112,437	52.19
F16	97,977	53.70	112,439	52.15
Baboon	70,470	56.31	113,605	52.19
Boat	82,995	55.15	113,550	52.17
Man	88,052	51.81	114,216	52.24
Peppers	88,756	54.51	111,651	52.18
Toy	101,693	53.20	109,216	52.13
Village	83,632	55.02	112,480	52.16
Woman	83,750	54.85	112,322	52.18
Average	87,121	54.39	112,426	52.18

표 2. Tsai 등의 방법의 블록 크기별 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡정도

Test images	2×2 block		3×3 block		4×4 block		5×5 block	
	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)
Lena	113,814	54.30	93,044	54.14	78,310	53.64	68,155	53.11
Airport	109,154	54.67	80,842	55.19	65,518	54.95	55,569	54.48
F16	117,903	54.01	97,977	53.70	83,175	53.25	73,058	52.81
Baboon	104,197	55.03	70,470	56.31	54,184	56.84	44,478	56.84
Boat	108,609	54.71	82,995	55.15	67,714	55.03	57,903	54.53
Man	112,178	51.78	88,052	51.81	71,935	51.76	61,710	51.58
Peppers	111,261	54.51	88,756	54.51	74,155	53.99	64,500	53.32
Toy	118,292	53.92	101,693	53.20	90,495	52.50	82,522	52.10
Village	110,416	54.59	83,632	55.02	68,217	54.92	57,761	54.60
Woman	109,573	54.56	83,750	54.85	68,087	54.62	58,134	54.26
Average	111,540	54.21	87,121	54.39	72,179	54.15	62,379	53.76

표 3 제안한 방법의 블록 크기별 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡정도

Test images	2×2 block		3×3 block		4×4 block		5×5 block	
	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)
Lena	171,829	51.70	112,339	52.18	82,364	52.31	63,535	52.32
Airport	171,523	51.71	112,437	52.19	80,607	52.37	61,033	52.51
F16	169,316	51.72	112,439	52.15	83,488	52.24	66,042	52.29
Baboon	172,085	51.71	113,605	52.19	81,805	52.44	61,967	52.56
Boat	172,402	51.72	113,550	52.17	82,280	52.37	62,258	52.49
Man	171,563	51.80	114,216	52.24	84,676	52.40	66,084	52.49
Peppers	171,668	51.72	111,651	52.18	80,349	52.33	61,121	52.41
Toy	167,960	51.73	109,216	52.13	83,113	52.17	69,216	52.10
Village	170,946	51.72	112,480	52.16	81,510	52.37	61,908	52.48
Woman	170,927	51.73	112,322	52.18	81,334	52.38	62,455	52.49
Average	171,022	51.73	112,426	52.18	82,153	52.34	63,562	52.41

양이 증가하는 것을 볼 수 있지만 블록을 작게 할수록 최대값 최소값 쌍이 많이 생성되어 추가 전송 데이터가 커지는 문제가 생긴다. 하지만, 제안한 방법에서는 최대값 최소값 쌍을 한 블록에 한 쌍만 생성하여 추가 전송 데이터의 양을 감소시켰다.

표 4에서는 Lena 이미지에서 Tsai 등의 방법과 제안한 방법의 추가 전송 데이터의 양을 비교하였다. 3×3 블록에서 제안한 방법은 Tsai 등의 방법보다 추가 전송 데이터의 양이 71% 감소되는 것을 볼 수 있었다. 2×2 블록에서 제안한 방법의 추가 전송 데이터의 양이 444,003 비트이고, 3×3 블록에서의 Tsai 등의 방법에서 추가 전송 데이터의 양이 924,800비트로 제안한 방법에서는 블록의 크기를 줄여 비밀 정보를 숨길 수 있는 양을 증가시키면서도 Tsai 등의 방법에서 추가 전송 데이터의 양의 48% 밖에 되지 않는 것을 볼 수 있었다.

또한, 의료 이미지에 대한 실험에서는 이미지의 특성상 비슷한 픽셀 값이 많이 분포하여 비밀 정보를 숨길 수 있는 양의 증가량은 12,258 비트로 앞서 실험한 결과보다 차이가 크지 않았다.

표 4. Lena 이미지에서 블록 크기별 발생하는 추가 전송 데이터 비교 (bit)

구 분	2×2 block	3×3 block	4×4 block	5×5 block
Tsai 등의 방법	2,097,152	924,800	524,288	332,928
제안한 방법	444,003	265,786	180,361	124,007
감 소 율	78%	71%	65%	62%

표 5. 의료 이미지에서 Tsai 등의 방법과 제안한 방법의 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡정도 비교 (3×3 블록)

Test images	Tsai 등의 방법		제안한 방법	
	Capacity (bit)	PSNR (dB)	Capacity (bit)	PSNR (dB)
a	101,996	53.80	120,160	52.06
b	146,624	54.10	169,465	54.16
c	118,981	52.51	115,467	52.03
d	98,769	54.07	117,717	52.11
e	119,339	52.51	116,568	52.01
f	104,502	53.14	112,809	52.11
g	205,949	52.09	235,778	51.38
h	104,380	53.09	110,644	52.13
Average	125,068	53.16	137,326	52.25

5. 결 론

본 논문은 Tsai 등이 제안한 예측 기법과 히스토그램을 이용한 가역 은닉 방법의 비밀 정보를 숨길 수 있는 양을 증가시키고 추가 전송 데이터의 양을 감소시키기 위해 제안하였다. 블록 기반의 역 S-예측 코딩과 모듈러 연산을 이용하여 이미지의 품질을 유지하면서 숨길 수 있는 양을 증가시키고, 추가 전송 데이터의 양을 줄이는 방법을 제안하였다. 블록 기반의 역 S-예측코딩을 사용하여 블록의 모든 픽셀을 삽입과정에서 사용할 수 있고, 추가 전송 데이터의 양을 줄이기 위하여 모듈러 연산을 이용하여 블록마다 최대값과 최소값 한 쌍만 사용하도록 하였다.

실험 결과 Tsai 등의 방법보다 비밀 정보를 숨길 수 있는 양이 28% 증가하고 이미지의 품질을 유지하는 것을 확인하였으며, Tsai 등의 방법보다 추가 전송 데이터의 양이 71% 감소하는 것을 볼 수 있었다. 그리고 Tsai 등의 방법에서는 픽셀 값의 변화가 적은 이미지와 픽셀 값의 변화가 많은 이미지에서 비밀 정보를 숨길 수 있는 양의 차이가 많이 나는 것을 볼 수 있지만, 제안한 방법에서는 이미지의 성질에 상관없이 비밀 정보를 숨길 수 있는 양과 이미지의 왜곡이 균등한 것을 볼 수 있었다.

참 고 문 헌

[1] D. Artz, "Digital Steganography: Hiding Data Within Data," *IEEE Internet Computing*, Vol. 5, No.3, pp. 75-80, 2001.

[2] R.Z.Wang, C.F. Lin, and J.C. Lin, "Image Hiding by Optimal LSB Substitution and Genetic Algorithm," *Pattern Recognition*, Vol.34, No.3, pp. 671-683, 2001.

[3] C.C. Chang, Y.H. Yu, and Y.C. Hu, "Hiding Secret Data in Images Via Predictive Coding," *Pattern Recognition*, Vol.38, No.5, pp. 691-705, 2005.

[4] Y.C. Hu, "High Capacity Image Hiding Scheme Based on Vector Quantization," *Pattern Recognition*, Vol.39, No.9, pp. 1715-1724, 2006.

[5] C.L Liu and S.R Liao, "High-Performance JPEG Steganography using Complementary Embedding Strategy," *Pattern Recognition*, Vol. 41, No.9, pp. 2945-2966, 2008.

[6] M.U. Celik, G. Sharma, A.M. Tekalp., and E. Saber, "Reversible Data Hiding," *Proc. of International Conference on Image Processing, Rochester*, NY, USA, Vol.2, pp. 157-160, September 24, 2002.

[7] J. Tian, "Wavelet-based Reversible Watermarking for Authentication," *Proc. Security and Watermarking of Multimedia Contents IV, Electronic Imaging 2002*, Vol.4675, pp. 679-690, 2002.

[8] J. Tian, "Reversible Watermarking by Difference Expansion," *Proc. of Workshop on Multimedia and Security*, pp. 19-22, 2002.

[9] H.C. Wu, C.C. Lee, C.S. Tsai, Y.P. Chu, and H.R. Chen, "A High Capacity Reversible Data Hiding Scheme with Edge Prediction and Difference Expansion," *The Journal of Systems and Software*, Vol.82, No.12, pp. 1966-1973, 2009.

[10] H.C. Wu, H.C. Wang, C.S. Tsai, and C.M. Wang, "Reversible Image Steganographic Scheme Via Predictive Coding," *Displays*, Vol.31, No.1, pp. 35-43, 2010.

[11] Y.C. Li, C.M. Yeh, and C.C. Chang, "Data Hiding Based on the Similarity Between Neighboring Pixels with Reversibility," *Digital Signal Processing*, Vol.20, Issue4, pp. 1116-1128, 2010.

[12] Z. Ni, Y.U. N.Q. Shi, N. Ansari, and W.E.I. Su, "Reversible Data Hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol.16, No.3, pp. 354-362, 2006.

[13] P. Tsai, Y.C. Hu, and H.L. Yeh, "Reversible Image Hiding Scheme using Predictive Coding and Histogram Shifting," *Signal Processing*, Vol.89, No.6, pp. 1129-1143, 2009.

[14] D.S. Kim, G.J. Lee, and K.Y. Yoo, "A Reversible Image Scheme Using Novel Linear Prediction Coding and Histogram Shifting," *The 2011 International Conference on*

Security & Management, pp. 282-287, 2011.
[15] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, Vol.1. Morgan Kaufmann, 2 edition, 2008.
[16] S.H, Bae, "A High Capacity Reversible Watermarking Using Histogram Shifting", *Journal of Korea Multimedia Society*, Vol. 13, No. 1, pp. 76-82, 2010.

김 대 수

2010년 2월 대구가톨릭대학교 인터넷공학 공학사
2010년 3월~현재 경북대학교 정보보호학과 석사과정
관심분야 : 암호학, 정보보호, 네트워크보안, 스테가노그라피

유 기 영

1976년 2월 경북대학교 수학교육과 이학사
1978년 2월 한국과학기술원 전산학과 공학석사
1992년 3월 미국 Rensselaer Polytechnic Institute 전산학과 공학박사
1978년 3월~현재 경북대학교 IT대학 컴퓨터학부 교수
관심분야 : 암호학, 정보보호, 유비쿼터스보안, 네트워크보안, 데이터베이스보안, 스테가노그라피, 인증프로토콜