이미지 왜곡을 줄인 이진 이미지 인증을 위한 정보 은닉 기법

(A Data Hiding Scheme for Binary Image Authentication with Small Image Distortion)

이 윤 호 † 김 병 호 ††
(Younho Lee) (Byoungho Kim)

요 약 본 연구에서는 삽입되는 정보에 의한 이미지의 왜곡을 최소화하는 이미지 인증을 위한 정보 은닉 기법을 제안한다. 제안 방법은 해밍 코드를 이용한 메시지 삽입 방법을 이용하여 적은 화소의 왜곡만으로 많은 양의 인증 정보의 삽입이 가능하다. 또한 정보 삽입으로 인한 이미지 영역의 훼손을 줄이기 위해 Yang 등이 제안한 변조 가능 기준(flippablity criteria)에 의해 선택된 변조 가능 화소(flippable pixel) 만을 정보 삽입에 사용한다. 마지막으로, 인증 정보가 각 변조 가능 화소에 삽입되는 순서를 은폐함으로써, 적법한 검증자가 아닐 경우, 이미지로부터 인증 정보를 추출해 내기 어렵게 한다. 제안 방법의 우수성을 보이기 위해, 기존 연구들과 반전되는 화소의 수, 오탐율에 대하여 비교 분석을 수행하며 그 결과로써 제안 방법이 적은 양의 화소값의 변화만으로 매우 낮은 오탐율을 보장함을 보인다. 이에 부가하여, 다양한 이진 이미지에 대해 제안 방법과 Yang 등의 방법을 적용하여 정보를 삽입하는 실험을 수행한다. 실험 결과에 대한 이미지 영역 분석을 통해 제안 방법이 이전의 방법보다 적은 왜곡을 갖게 됨을 보이고, 최근에 제안된 이진 이미지 정보 은닉 방법에 대한 공격에도 이전의 방법들보다 좀 더 안정성이 있음을 보인다.

키워드 : 정보 보호, 정보 은닉, 이미지 인증, 디지털 워터마킹, 메시지 인증 코드

Abstract This paper proposes a new data hiding scheme for binary image authentication with minimizing the distortion of host image. Based on the Hamming-Code-Based data embedding algorithm, the proposed scheme makes it possible to embed authentication information into host image with only flipping small number of pixels. To minimize visual distortion, the proposed scheme only modifies the values of the flippable pixels that are selected based on Yang et al's flippablity criteria. In addition to this, by randomly shuffling the bit-order of the authentication information to be embedded, only the designated receiver, who has the secret key that was used for data embedding, can extract the embedded data. To show the superiority of the proposed scheme, the two measurement metrics, the miss detection rate and the number of flipped pixels by data embedding, are used for the comparison analysis between the proposed scheme and the previous schemes. As a result of analysis, it has been shown that the proposed scheme flips smaller number of pixels than the previous schemes to embed the authentication information of the same bit-length. Moreover, it has been shown that the proposed scheme causes smaller visual distortion and more resilient against recent steg-analysis attacks than the previous schemes by the experimental results.

Key words : Security, Information Hiding, Image authentication, Digital watermarking, Authentication Codes

† 정 회 원 : 영남대학교 전자정보공학부 전임강사
yhlee@ynu.ac.kr
†† 정 회 원 : 경성대학교 컴퓨터공학과 교수
bkim@ksu.ac.kr
(Corresponding author임)
논문접수 : 2006년 9월 20일
심사완료 : 2008년 12월 24일

정보과학회논문지: 시스템 및 이론 제36권 제2호(2009.4)

1. 서 론

정보 은닉(data hiding) 기법은 디지털 컨텐츠에 컨텐츠 왜곡을 최소화하여 특정한 정보를 은닉하는 기법이다. 정보 은닉 기법은 디지털 워터마킹(digital watermarking) 및 스테가노그래피(steganography) 분야를 포함한다. 디지털 워터마킹은 디지털 컨텐츠에 대한 저작권 정보나 인증 정보를 은닉시킴으로써 저작권 관리, 디지털 컨텐츠의 인증 등에 사용되며, 스테가노그래피는 정보의 은닉에 중요성을 둔 기법으로 군사적 용도로 이용된다.

이미지에 대한 정보 은닉 기법은 주로 그레이스케일 또는 컬러 이미지에 대해 많이 연구되었으며, 화소의 색이나 밝기 값을 눈에 띄지 않게 끔만 바꾸어 이미지 왜곡을 최소화 하는 방법이 주를 이루었다. 그러나 이진 이미지에 대한 정보 은닉 기법은 팩스, 육필 전자 서명 등에 여전히 널리 사용되고 있음에도 불구하고, 최근에야 비로소 주목을 받기 시작하였다. 일반적으로 이진 이미지에 대한 안전한 정보 삽입은 컬러/그레이스케일의 경우보다 훨씬 어려운데, 그 이유는 이진 이미지는 각 화소가 0 혹은 1의 값만을 갖기 때문이다.

최근 이진 이미지 인증을 위한 몇몇 정보 은닉 기법이 제안되었다[1-7]. 하지만 방법들 중의 일부[1,6]은 오탐율[1](miss detection rate)이 2^{-20} 이상으로써 매우 큰 단점이 존재한다. 또한 나머지 방법들도 인증 정보 삽입을 위한 많은 수의 화소 값을 변화에 따른 큰 이미지의 왜곡이라는 단점이 존재하며, 이러한 이미지 왜곡의 유형 파악을 통한 정보 은닉 검출 기법에 취약한 면을 보이고 있다[8,9].

본 논문에서는 현재 암호학적 알고리즘들이 보장하는 2^{-80} 이하의 오탐율을 갖으면서, 이미지 왜곡 측면에서 기존의 방법들보다 개선된 새로운 이진 이미지 인증을 위한 정보 은닉 기법을 제안한다. 제안 방법은 해밍 코드[10]를 이용한 정보 삽입 방법[11]을 이용하여 삽입 정보당 변환 되는 화소의 수를 기존의 방법보다 감소시킨다. 이 때, 정보 삽입으로 인한 이미지 영역에서의 왜곡을 줄이기 위해 Yang 등이 제안한[7] 변환 가능 기준을 이용하여 원 이미지로부터 변환 가능 화소를 선별한 후, 해당 화소들만을 사용하여 해밍 코드 삽입 방법을 적용한다.

해밍 코드 기반 삽입 방법은 적은 화소 변환만으로 많은 양의 정보를 삽입할 수 있으나, 공격자가 해밍 코드가 삽입된 화소들을 추출할 수 있다면, 해당 화소 값들을 변화시키면서도, 즉 이미지를 변조시켜도, 원래의 삽입 정보의 추출이 가능하도록 하는 공격 방법이 존재한다. 본 연구에서는 이러한 문제점을 보완하기 위해서 변환 가능 화소에 정보 삽입 시, 삽입되는 정보의 순서를 의사 난수 생성기를 이용하여 변환시킨다. 따라서 공격자는 정확한 해밍 코드 정보를 읽어낼 수 없기 때문에 위에서 언급한 공격 방법을 사용할 수 없게 된다.

제안 방법에서 이미지 인증을 위해 삽입 되는 정보는 변환 가능 화소를 모두 0값으로 변환시킨 전체 이미지에 대한 메시지 인증 코드(MAC: Message Authentication Codes)이다. 이러한 MAC을 이용하여 제안 방법의 오탐율을 2^{-80} 이하로 보장한다.

제안 방법의 우수성을 보이기 위해, 다음의 세 가지 과정을 수행하였다. 첫째로 제안 방법과 이전의 방법들에 대하여 인증 정보 삽입으로 인하여 원래의 이미지로부터 변환되는 화소 수 및 오탐율에 대하여 분석하였다. 둘째로, 이미지 영역에서의 왜곡 여부에 대한 분석을 위해 다수의 이진 이미지에 대하여 제안 방법과 기존의 연구들 중 이미지 영역에서 가장 적은 이미지 왜곡을 발생시키는 Yang 등의 방법[7]을 적용하여 인증 정보를 삽입한 후, 삽입된 이미지들에 대하여 이미지 왜곡을 분석한다. 이미지 왜곡은 이미지 분야에서 가장 많이 사용되는 척도인 극 신호 대 잡음비(Peak Signal-to-Noise Ratio: PSNR)와 최근에 이진 이미지에 대한 왜곡 분석 척도로 제안된 경계선 구획 유사도 기반 측정 방법(Edge Line Segment Similarity Measure: ELSSM)[12]을 이용하였다. 분석 결과 제안 방법은 PSNR, ELSSM에서 모두 Yang 등의 방법에 비해 좋은 성능을 보임을 알 수 있었다. 마지막으로, 이진 이미지에 대한 정보 은닉을 효과적으로 감지할 수 있는 Cheng 등의 공격 방법[8,9]이 제안 방법에는 적용시키기 어려움을 실험을 통해 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 설명하며, 3장에서는 제안 이미지 인증 기법에 대해 기술한다. 4장에서는 제안 기법과 이전 방법들과의 성능 분석 및 비교 평가 결과에 대해 기술하고, 5장에서 결론을 맺는다.

2. 관련 연구

최근까지 제안된 이진 이미지 인증을 위한 정보 은닉 기법들은 다음과 같다. Tzeng[1] 등은 워터 마크를 삽입하는 대상 이미지인 원 이미지(host image)를 블록으로 나누어 공유된 코드 워드 정보를 블록 별로 삽입하는 방법을 사용하였다. 코드 워드 정보를 삽입하기 위해 코드 워드 정보를 저장하는 화소의 위치를 나타내는 코드 홀더 정보를 공유하고, 여러 개의 코드 홀더 중 필요 반전 화소 수를 가장 최소로 하는 코드 홀더를 선택하

1) 오탐율: 인증 정보가 삽입된 이미지가 아닌 다른 이미지가 이미지 인증 검증 테스트를 통과할 확률을 의미한다[1].

여 해당 화소 위치에 코드 워드 정보를 삽입하는 방법을 사용하였다. 코드 홀더 선정 및 코드 워드 생성시에 정보 삽입자는 정보 추출/검증자와 공유된 비밀키를 초기값으로 하는 유사 난수 생성기를 사용하였다. 해당 방법은 코드 홀더 개수가 적을 경우 빠른 처리 속도를 갖고, 블록별로 코드 워드를 삽입하여 검증하기 때문에 블록 별로 이미지가 변조되었는지 확인할 수 있는 장점이 존재하나, 블록 별로 독립적으로 코드 워드를 처리하기 때문에, 오탐율이 전체 이미지의 크기가 아닌 하나의 블록의 크기에 의존하게 되어 오탐율이 전체 이미지를 대상으로 처리하는 방법에 비해 증가하게 되는 단점이 존재한다. 이러한 이유로 Tzeng의 방법은 오탐율이 2^{-20} 이상으로써 현재 암호학적으로 사용되는 일방향 해쉬 알고리즘 적용 시 도달 가능한 수준인 2^{-80}에 비해 매우 큰 단점이 존재한다.

2004년 김해용 등은 이진 이미지의 인증 정보 삽입을 위해 인증 정보가 들어갈 화소의 위치를 공개적으로 공유하는 방법을 제안하였다[2,3]. 해당 방법은 임의의 원 이미지에 인증 정보를 삽입을 할 경우, 공개적으로 알려진 인증 정보 삽입 대상이 되는 화소의 위치를 초기화시킨 후, 초기화된 이미지를 메시지 부분으로 한 전자 서명 또는 메시지 인증 코드(MAC)를 구하여 해당 값을 초기화된 화소의 위치에 삽입하는 방법을 제안하였다. 본 방법은 인증 정보 생성자와 추출/검증자 사이에 공유되는 비밀키(삽입되는 정보가 MAC일 경우) 또는 정보 생성자의 개인키(삽입되는 정보가 전자 서명일 경우)를 통해 정보 삽입이 가능하다. 본 방법은 암호학적으로 안전하다고 알려진 알고리즘을 사용하기 때문에 현재 실용적으로 쓰이는 전자 서명 또는 메시지 인증 코드 알고리즘을 적용할 경우 오탐율이 2^{-80} 이상이 되는 상당히 안전한 방법이다. 그러나 본 방법은 이미지 영역의 왜곡에 대한 고려 없이, 공유된 키의 정보를 이용한 난수 정보를 이용하여 정보 삽입 화소를 결정하기 때문에, 정보 삽입시 이미지 영역의 왜곡이 많은 단점이 존재한다.

이 후 이미지 영역의 왜곡을 고려한 많은 방법들이 있었다. Wu 등은 이진 이미지 영역에서의 왜곡을 줄이기 위해 화소의 변조 가능 기준(flippability criterion)을 도입한 새로운 정보 은닉 기법을 제안하였다[6]. 이후 Yang 및 김해용 등은 Wu의 방법을 개선한 새로운 변조 가능 기준을 도입한 정보 은닉 기법을 제안하였다[4,5,7]. Wu의 방법과 차별되는 해당 방법들의 특징은 이미지 수신자 측에서도 변조 가능 화소(flippable pixel)의 위치를 알 수 있다는 것이다. 따라서 변조 가능 화소 부분을 제외한 영역을 정보 삽입자와 이미지 수신자인 정보 추출/검증자가 공유할 수 있으며, 이러한 이유로 전체 이미지 영역을 입력값으로 하는 MAC 및 전자 서명의 사용이 용이한 장점이 있다.

그러나 이러한 변조 가능 화소 기준을 이용한 방법들도, 기본적으로 메시지 삽입을 위해 삽입 비트당 평균적으로 0.5 화소의 값을 변환시키는 점은 변화가 없으며 이것은 [11]의 연구 결과에 따르면 매우 높은 수준의 화소 변조이다. 본 연구에서는 이러한 문제점을 개선하여 기존의 연구들보다 적은 수의 화소 변조만으로 이진 이미지 인증 정보를 삽입할 수 있도록 하는 방법을 제안하는 것을 목적으로 한다.

3. 제안 방법

본 절에서는 새로운 이진 이미지 인증을 위한 정보 은닉 기법을 제안한다. 3.1절에서는 제안 방법이 가정하는 환경에 대해 기술한다. 3.2절에서는 제안 방법에 적용하게 되는 Yang 등이 제안한 변환 가능 화소 및 그 평가 기준에 대해 기술한다. 3.3절에서는 사용되는 용어 및 기호, 그리고 알고리즘에 대하여 기술하고, 마지막 3.4절에서 제안 방법을 기술한다.

3.1 제안 방법의 환경 및 가정

본 연구에서 가정하는 시나리오는 아래의 그림 1과 같다. (1) 정보 삽입자는 자신이 소유한 원이미지에 대해서 정보 추출자와 공유한 비밀키 $K(\in\{0,1\}^*)$ 및 원 이미지 정보를 이용하여 인증정보를 생성한다. (2) 이후 생성된 인증 정보를 원 이미지, 비밀 키 K, 그리고 (1)의 과정에서 생성된 인증 정보를 입력값으로 하는 정보 삽입 알고리즘을 수행한다. (3) 정보 삽입 알고리즘의 수행 결과로써 생성되는 정보 삽입 이미지(stego-image)를 정보 추출자에게 전달한다. (4) 정보 추출자는 정보 삽입자로부터 전송된 정보 삽입 이미지로부터 공유 비밀키 K를 이용하여 인증 정보를 추출해 낸다. (5) 마지막으로, 추출된 인증 정보 및 공유 비밀키 K를 이용하여 전달받은 정보 삽입 이미지를 인증한다.

(3)의 과정에서 감시자는 해당 정보 삽입 이미지에 대한 공격이 가능하다. 감시자의 공격은 다음의 두 가지이다. 하나는 해당 정보 삽입 이미지를 변형시켜서 정보 유효성 검증을 통과하는 위조된 정보 삽입 이미지를 생성시키는 공격이다(A)-(a). 두 번째 공격은 이미지 자체에 인증 정보가 삽입되었는지의 여부를 판별해 내는 공격이다(A)-(b).

본 연구에서는 이러한 시나리오에서 사용될 수 있는 정보 삽입 및 추출 알고리즘을 제안한다. 3.4절에서는 이러한 네 가지 알고리즘에 대해 각각 기술한다.

3.1.1 성능 분석 척도

이진 이미지 인증을 위한 정보 은닉 방법은 위에서 언급한 (A)-(a) 및 (A)-(b)의 두 가지 공격을 성공하지 못하게 하는 것을 목적으로 한다. 이러한 공격의 성공

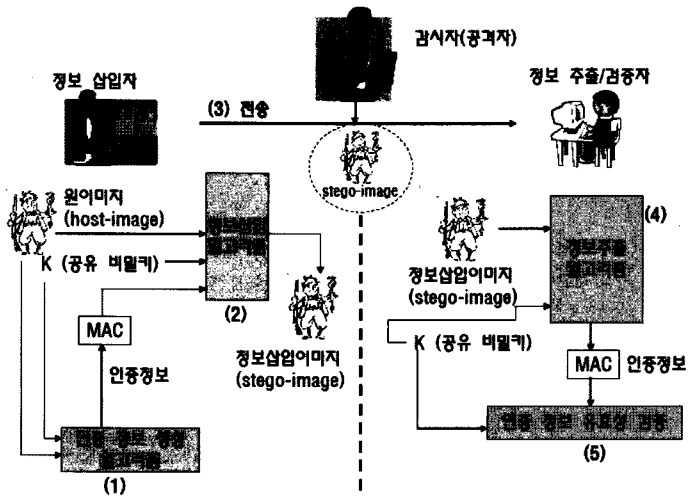


그림 1 제안 방법이 가정하는 시나리오

가능성을 가늠하기 위한 이진 이미지 인증을 위한 정보 은닉 기법의 성능 분석 척도는 아래와 같다.

1) **오탐율**(miss detection rate): 정보 삽입자가 적법한 과정을 통하여 생성한 스테고 이미지가 아닌 훼손된 이미지가 이미지 검증 과정을 통과할 확률을 의미한다.
2) **왜곡량**(image distortion): 정보 삽입 이미지 생성 시, 인증 정보 삽입으로 인한 원 이미지의 훼손된 정도의 양을 의미한다. 왜곡량은 이미지의 화소값의 변화 뿐만 아니라, 이미지 영역에서의 시각적 왜곡도 동시에 고려되어야 한다. 본 왜곡량이 작아질수록 공격자가 (A)-(b) 공격을 성공할 확률이 감소하게 된다. 왜곡량에 대한 정량적 측정 방안에 대해서는 4장에서 자세히 기술한다.

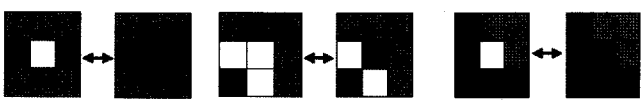
3.2 변환 가능 화소

본 세부 절에서는 제안 방법에 사용되는 Yang 등이 제안한 변환 가능 화소 선택 방법에 대해 기술한다[7]. 해당 변환 가능 화소 선택 기준은 임의의 화소가 있을 때, 해당 화소 주변의 8 화소의 값들로 결정된다. Yang 등의 변환 가능 화소 기준은 이러한 9개의 화소 중 검은색 화소들 간의 연결성을 유지를 기준으로 설정하기 때문에 연결성 보존 기준(Connectivity-Preserving Criteria)이라 부른다.

변환 가능 화소의 기준을 수식적으로 표현하기 위해 그림 2의 (a)와 같이 변환 가능 화소 판단 대상이 되는 현 화소를 p_c라 하고, 주변의 화소를 각각 $w_1, w_2, \ldots, w_8$로 표기하도록 한다. 이러한 표기시에 p_c의 값이 현재의

w_6	w_7	w_8
w_5	p_c	w_1
w_4	w_3	w_2

(a) P_c 화소의 변환 가능 화소 여부 판단을 위한 주변 8 화소의 표기



(b) Yang 등의 변환 가능 화소의 선정 기준에서 중앙 화소가 변환 가능 화소가 아닌 경우 (: 어떤 화소값을 갖어도 관계없는 화소)

그림 2 변환 가능 화소 선정 기준

화소값에서 다른 값으로 바뀌어도 아래의 네 개의 수식으로 계산되는 N_{vw}, N_{vb}, N_{ir}, N_C 값이 변화되지 않을 경우, p_c는 변환 가능 화소가 된다. 본 수식에서는 0은 흰색 화소, 1은 검은색 화소값을 의미하며, $\bar{}$ 는 이진수 값에 대한 논리적 부정 연산자이며 $\cdot$는 논리적 곱 연산자이다.

$$N_{vw} = \sum_{k=1,3} \overline{p_c} \cdot \overline{w_k} \cdot \overline{w_{k+4}},\ N_{vb} = \sum_{k=1,3} p_c \cdot w_k \cdot w_{k+4} \quad (1)$$

$$N_{ir} = \sum_{k=1}^{4} \overline{p_c} \cdot w_{2k} \cdot \overline{w_{2k+1}} \cdot \overline{w_{2k-1}} \quad (w_1 = w_9) \quad (2)$$

$$N_C = \sum_{k=1}^{4} p_c \cdot w_{2k} \cdot w_{2k+1} \cdot w_{2k+2} \cdot w_{2k+3} \cdot w_{2k+4} \quad (w_t = w_{t \bmod 8}) \quad (3)$$

위에서 기술한 식 (1), (2), (3)의 이미지 영역의 의미는 그림 2(b)의 중앙 화소가 변환 가능 화소가 아닌 경우 1), 2), 3)에 각각 대응된다.

최종적으로, 이러한 변환 가능 기준을 알고리즘으로 나타내면 아래의 알고리즘 1과 같다. 본 알고리즘 1은 제안 방법에 사용된다.

변환 가능 화소 결정 알고리즘 (FlipDecision)
입력값: 원 이미지 $I \in \{0,1\}^{width \times heigth}$ ($width$: 이미지의 너비, $height$: 이미지의 높이), 변환 가능 여부 결정 대상 화소의 위치 $p=(x,y)(0 \le x < width, 0 \le y < height)$
결과값: 1 (p위치의 화소가 변환 가능 화소일 경우)
0 (그 이외의 경우)
STEP 1: I의 p 위치의 주변 화소가 8개 미만일 경우 0을 반환
STEP 2: $p_c \leftarrow (p$ 위치의 화소값), $w_1,\ldots,w_8 \leftarrow$ (p위치 주변 8개의 화소값) (그림 2(a)의 규칙에 따라 각각 저장)
STEP 3: 수식 (1)-(3)을 이용, $N_{vb(1)}$, $N_{vw(1)}N_{ir(1)}$, $N_{C(1)}$ 계산
STEP 4: $p_c \leftarrow \overline{p_c}$
STEP 5: 수식 (1)-(3)을 이용, $N_{vb(2)}$, $N_{vw(2)}N_{ir(2)}$, $N_{C(2)}$을 다시 계산
STEP 5: $N_{vb(1)} \neq N_{vb(2)}$ 또는 $N_{vw(1)} \neq N_{vw(2)}$ 또는 $N_{ir(1)} \neq N_{ir(2)}$ 또는 $N_{C(1)} \neq N_{C(2)}$이면 0을 반환, 그렇지 않을 경우 1을 반환

알고리즘 1 변환 가능 화소 결정 알고리즘[7]

3.3 사용되는 알고리즘 및 기호들

본 세부 절에서는 제안 방법에 사용되는 외부 알고리즘에 대해 정의하고 제안 방법에 사용대한 기호의 설명을 기술한다.

3.3.1 사용되는 기호들

1) $K \in \{0,1\}^*$: 정보 삽입자와 정보 추출/검증자 사이에 공유된 키 정보이다. 단지 K값을 갖고 있는 경우에만 정보 삽입 이미지로부터 정보를 추출할 수 있으며, 정보 삽입 이미지에 대한 검증이 가능해야 한다.
2) $I \in \{0,1\}^{width \times height}$: 비밀 정보 삽입 대상의 원 이미지를 의미한다. 해당 이미지의 너비는 $width$, 높이는 $height$값을 갖는다.
3) $b_i \in Z_{width} \times Z_{height}$, $s_i \in \{0,1\}$: b_i는 변환 가능 화소 i의 위치를 나타내며 s_i는 화소 값을 의미한다.
4) t: 원 이미지 I의 변환 가능 화소의 수를 의미한다.
5) $I(b_i)$: 이미지 I의 b_i 위치의 화소를 의미한다. 해당 값이 변경될 경우 이미지 I 중 b_i의 위치의 화소값이 변경된다.
6) $|A|$: A의 비트 길이를 나타낸다.
7) $\|$: 비트 정보간의 연결(concatenation) 연산을 나타낸다.
8) $\bar{s}$: 이진수인 s값에 대한 논리적 부정값을 나타낸다.
9) M: 원 이미지에 삽입 되는 인증 정보를 의미한다.
10) k: 인증 정보의 비트 길이를 나타낸다($k \le t$).

3.3.2 사용되는 알고리즘들

1) $FindFlip(I)$ $(0,1^{width \times height} \rightarrow (Z_{height} \times Z_{width} \times 0,1)^t)$

이미지 I에서 변환 가능 화소를 찾아 해당 화소들의 위치와 화소의 값을 반환 하는 역할을 수행한다. 단, 변환 가능 화소들이 서로 이웃에 있을 경우, 중복된 변환 가능 화소 중 하나의 값을 바꾸면, 나머지 변환 가능 화소가 변환 불가능 화소로 바뀔 수 있기 때문에, 서로 이웃한 변환 가능 화소들은 해당 화소들 중 이미지의 왼쪽 위 모서리 화소의 좌표를 (0, 0)으로 했을 때 해당 화소의 y좌표가 더 작거나, 만약 y좌표가 같다면 x좌표가 더 작은 화소만을 변환 가능 화소로 설정하고, 나머지 화소는 변환 불가능 화소로 설정한다[7]. 해당 알고리즘은 3.2절의 *FlipDecision* 알고리즘을 이용하면 쉽게 구현이 가능하다.

2) $RandPerm(((p_1,s_1),\ldots,(p_t,s_t)),K)$
$((Z_{height} \times Z_{width} \times 0,1)^t \times 0,1^* \rightarrow (Z_{height} \times Z_{width} \times 0,1)^t)$

순열값 $((p_1,s_1),\ldots,(p_t,s_t))$ 및 공유키 K를 입력받아 순열값의 순서를 섞어주는 의사 난순열 생성(Pseudo-Random Permutation) 작업을 수행하는 알고리즘이다. 여기서 K는 난순열 생성시 씨앗(seed) 값으로 사용된다. 본 알고리즘은[13-15]에서 제안된 방법들을 이용하여 표준 대칭키 암호 알고리즘을 기반으로 구현이 가능하다.

3) $MACGen(I,K)$ $(\{0,1\}^{width \times height} \times \{0,1\}^* \rightarrow \{0,1\}^k)$

I를 메시지로 하고 K값을 키로 사용하는 MAC(메시지 인증 코드) 생성 알고리즘을 나타낸다. MAC 생성시 사용된 키 값을 알고 있는 경우, 생성된 MAC값의 유효성에 대한 검증이 가능하다. MAC 알고리즘은 HMAC[16] 등 다양한 표준 MAC알고리즘을 이용하여 구현이 가능하다.

4) $HCBEmbed(I,m)/HCBExt(S)$

해밍 코드 기반 정보 삽입/추출 알고리즘을 나타낸다. $HCBEmbed$는 c 비트의 원 이미지 I로부터 최대 $\log_2\lfloor(c+1)\rfloor$ 비트의 정보 m의 삽입을 I 내부의 최대 1개의 화소값 변경만으로 삽입이 가능하다. $HCBExt$는 정보 삽입 이미지 S로부터 삽입 정보를 추출하는 알고리즘이다. 좀 더 자세한 사항은 본 논문의 부록 A에 기술되어 있다.

3.4 제안 정보 삽입/추출 알고리즘

본 세부 절에서는 제안 방법에 대해 기술한다. 3.1절의 그림 1에서 설명한 네 가지 알고리즘에 대하여 기술한다.

3.4.1 인증 정보 생성 알고리즘

제안 인증 정보 생성 알고리즘은 아래의 알고리즘 2와 같이 기술된다. 제안 알고리즘은 원 이미지에서 변환 가능 화소를 찾아내어 해당 화소를 0(흰색)으로 초기화 후, 해당 이미지에 대한 MAC값을 계산한다. 생성된 인증 정보는 바로 MAC값이 된다.

인증 정보 생성 알고리즘
입력값: 원 이미지 $I\in\{0,1\}^{width\times heigth}$ ($width$: 이미지의 너비, $height$: 이미지의 높이), 공유키 $K\in\{0,1\}^*$
결과값: 인증 정보 $\sigma\in\{0,1\}^k$
STEP 1: $((p_1,s_1),\ldots,(p_t,s_t))\leftarrow FindFlip(I)$
STEP 2: $I(p_1)\leftarrow 0,\ldots,I(p_t)\leftarrow 0$
STEP 3: $\sigma\leftarrow MACGen(I,K)$
STEP 4: σ를 반환한다.

알고리즘 2 인증 정보 생성 알고리즘

3.4.2 정보 삽입 알고리즘

제안 정보 삽입 알고리즘의 핵심은 세 가지이다. 첫째는 시각적 왜곡을 줄이기 위해 변환 가능 화소만을 이용하여 인증 정보를 삽입하는 것이고, 둘째는 해밍 코드 기반 정보 삽입 알고리즘[2]을 통해 전체 이미지에서 변환되는 화소의 수를 줄이는 것이며, 마지막으로 해밍 코드 기반 정보 삽입 알고리즘의 문제점인 하나의 정보 표현을 다수개의 코드가 나타낼 수 있다는 성질을 없애기 위해 공유키를 이용한 유사 난수열 생성기를 이용하여 정보가 삽입되는 순서를 숨기는 방법을 사용한다는 것이다.

해밍 코드 기반 정보 삽입 방법은 최대 1 화소값의 변화만으로 c비트 이미지에 $\log_2\lfloor(c+1)\rfloor$ 비트 정보를 삽입할 수 있어 적은 이미지 왜곡으로 많은 정보를 삽입할 수 있는 장점이 있다. 그러나 해밍 코드의 특징으로 인하여, 정보 삽입 이미지와 동일한 $\log_2\lfloor(c+1)\rfloor$ 비트 정보가 삽입된 다른 $2^{c-\lfloor\log_2(c+1)\rfloor}$ 가지의 c비트 길이의 이미지가 존재하며 이는 해밍 코드 계산법을 이용하여 쉽게 찾아낼 수 있다. 이러한 성질은 일반적인 이미지 정보 은닉 기법에는 별 문제가 되지 않으나, 이미지의 인증 및 무결성을 보장하기 위한 정보 은닉 방법에서는 좋지 않은 것이다. 왜냐하면, 정보 삽입자가 생성한 이미지가 아닌 변조된 이미지가 동일한 인증 정보를 내놓을 수 있기 때문에, 정보 추출/검증자는 해당 정보 삽입 이미지가 변조되었다는 것을 판별할 수 없기 때문이다. 본 제안 방법에서는 이러한 문제점을 해결하기 위해 해밍 코드 삽입 방법으로 정보가 삽입된 후 생성된 해밍 코드 워드 값을 공유된 키 와 의사 난순열 생성기를 이용하여 순서를 변화시킨 후, 원 이미지의 변환 가능 화소에 저장한다. 따라서 공유키 K를 모르는 공격자는 해밍 코드워드 값을 알 수 없게 되기 때문에 위에서 언급한 변조 공격을 수행할 수 없다.

이에 부가하여, 제안 방법에서는 전체 삽입 가능한 정보의 양을 늘리기 위해 삽입 가능 화소를 여러 개의 블록으로 나누어서 각 블록에 대해 해밍 코드 삽입 방법을 사용한다. 본 블록 기반 기법은 전체 삽입 가능 정보의 양을 증가시키는데 기여한다. 예를 들어, 원 이미지에 삽입 가능 화소가 $1024(=2^{10})$개 있을 경우, 전체 삽입 가능 화소를 1블럭으로 생각한다면, 삽입 가능 정보의 양은 $\lfloor\log_2(\lfloor 2^{10}+1\rfloor)\rfloor=10$비트이다. 이에 반하여 1024개의 화소를 두 개의 블록으로 나누어 1 블록당 512개의 화소를 갖는 것으로 생각한다면, 전체 삽입 가능 정보의 양은 $2\cdot\lfloor\log_2(\lfloor 2^9+1\rfloor)\rfloor=18$비트가 되어 삽입 가능한 정보의 양이 증가한다. 그렇지만 블록 당 최대 1개의 화소가 변경되므로 나누어지는 블록의 수가 증가할수록 이미지가 훼손되는 정도인 왜곡량은 증가하게 된다.

알고리즘 3에 제안 정보 삽입 알고리즘을 정형화하여 기술한다. 제안 알고리즘에 사용된 부 알고리즘 및 기호들의 대한 내용은 3.3절에 기술되어 있다.

정보 삽입 알고리즘
입력값: 원 이미지 $I\in\{0,1\}^{width\times heigth}$ ($width$: 이미지의 너비, $height$: 이미지의 높이), 공유키 $K\in\{0,1\}^*$, 인증 정보 $\sigma\in\{0,1\}^k$
결과값: 정보 삽입 이미지 $S\in\{0,1\}^{width\times heigth}$
STEP 1: $((p_1,s_1),\ldots,(p_t,s_t))\leftarrow FindFlip(I)$
STEP 2: $((p'_1,s'_1),\ldots,(p'_t,s'_t))\leftarrow RandPerm((p_1,s_1),\ldots,(p_t,s_t),K)$
STEP 3: $k\le n(\lfloor\log_2(\lfloor t/n\rfloor+1)\rfloor)$를 만족하는 최소 n을 찾아낸다. 만약 찾아내지 못하면 삽입 불가능이므로 종료한다.

2) 해밍 코드 기반 정보 삽입 방법에 대한 내용은 부록 A에 자세히 기술되어 있다.

STEP 4: $s'_1 \| s'_2 \| \cdots \| s'_t$ 를 n개의 블록 $B_1, \ldots, B_n$으로 나눈다.
$(B_j = s'_{(j-1)\cdot \lfloor t/n \rfloor + 1} \| \cdots \| s'_{j \cdot \lfloor t/n \rfloor} (1 \le j \le n))$
STEP 5: σ를 $\lfloor \log_2(\lfloor t/n \rfloor + 1) \rfloor$ 비트 길이로 각각 나눈다. 이렇게 나누어진 σ의 조각들을 $m_1, \cdots, m_{n'} (n' \le n)$이라 한다.
STEP 6: $B'_i \leftarrow HCBEmbed(B_i, m_i)$를 수행한다. $(i = 1, \cdots, n')$
STEP 7: B'_i와 B_i의 값을 비교하여 틀린 부분의 위치값을 찾아낸다. 해당 위치 값을 $f_i (\in \{1,2,\cdots, \lfloor t/n \rfloor\})$라 하자. $(i = 1, \cdots, n')$ 만약 틀린 부분이 없으면 $f_i \leftarrow 0$을 수행한다.
STEP 8: $f_i \neq 0$이라면, $I(b'_{(i-1)\cdot \lfloor t/n \rfloor + f_i}) \leftarrow \overline{s'_{(i-1)\cdot \lfloor t/n \rfloor + f_i}}$을 수행한다. $(i = 1, \cdots, n')$
STEP 9: $S \leftarrow I$를 수행하고 S를 반환한다.

알고리즘 3 정보 삽입 알고리즘

3.4.3 정보 추출 알고리즘

정보 추출 알고리즘은 정보 삽입 알고리즘과 비슷한 과정으로 진행된다. 전달 받은 정보 삽입 이미지로부터 삽입 가능 화소를 찾아낸 다음, 해당 화소들을 미리 동의한 비밀 정보의 비트 길이에 맞추어 적당한 수의 블록으로 나눈 후, 해밍 코드 기반 정보 추출 알고리즘을 통하여 추출해낸다. 정보 추출 알고리즘의 정형화된 과정은 아래의 알고리즘 4에 기술되어 있다.

정보 추출 알고리즘
입력값: 정보 삽입 이미지 $S \in \{0,1\}^{width \times heigth}$ ($width$: 이미지의 너비, $height$: 이미지의 높이), 공유키 $K \in \{0,1\}^*$, 인증 정보의 비트 길이 k
결과값: 인증 정보 $\sigma \in \{0,1\}^k$
STEP 1: $((p_1, s_1), \ldots, (p_t, s_t)) \leftarrow FindFlip(S)$
STEP 2: $((p'_1, s'_1), \ldots, (p'_t, s'_t)) \leftarrow RandPerm((p_1, s_1), \ldots, (p_t, s_t), K)$
STEP 3: $k \le n(\lfloor \log_2(\lfloor t/n \rfloor + 1) \rfloor)$를 만족하는 최소 n을 찾아낸다. 만약 찾아내지 못하면 삽입된 정보가 없으므로 종료한다.
STEP 4: $s'_1 \| s'_2 \| \cdots \| s'_t$를 n개의 블록 $B'_1, \ldots, B'_n$으로 나눈다.
$(B'_j = s'_{(j-1)\cdot \lfloor t/n \rfloor + 1} \| \cdots \| s'_{j \cdot \lfloor t/n \rfloor} (1 \le j \le n))$
STEP 6: $m_i \leftarrow HCBExt(B'_i)$를 수행한다. $(i = 1, \cdots, n)$
STEP 7: $m_1 \| \cdots \| m_n$ 중에 앞에서부터 k비트 만을 잘라내어 해당 값을 σ로 설정한다.
STEP 8: σ를 반환한다.

알고리즘 4 정보 추출 알고리즘

3.4.4 인증 정보 검증 알고리즘

추출된 인증 정보 σ및 정보 삽입 이미지 S를 입력값으로 하여 해당 이미지의 인증 정보를 검증한다. 본 알고리즘에서 특이한 사항은 S에서 변환 가능 화소를 모두 0으로 설정한 이미지 S'과 원 이미지 I에서 변환 가능 화소를 모두 0으로 설정한 이미지 I'은 같은 이미지라는 것이다. 따라서 본 알고리즘은 S로부터 변환 가능 화소를 모두 0으로 설정한 이미지 S을 입력값으로 하여 메시지 인증 코드 생성 작업을 수행하여 생성된 결과인 σ'과 인증 정보 추출 알고리즘의 결과로 S로부터 추출된 값 σ가 같은 값인지 확인함으로 인증 정보 검증 알고리즘 수행을 마치게 된다. 이를 정형화한 본 알고리즘에 대한 기술은 다음의 알고리즘 5에 자세히 기술되어 있다.

인증 정보 검증 알고리즘
입력값: 정보 삽입 이미지 $S \in \{0,1\}^{width \times heigth}$ ($width$: 이미지의 너비, $height$: 이미지의 높이), 공유키 $K \in \{0,1\}^*$, 인증 정보 $\sigma \in \{0,1\}^k$
결과값: 1 (인증 성공: S는 공유된 키 K를 이용하여 정상적인 과정으로 생성된 정보 삽입 이미지라는 것을 의미) 0 (인증 실패: 인증 성공의 경우를 제외한 나머지 경우)
STEP 1: $((p_1, s_1), \ldots, (p_t, s_t)) \leftarrow FindFlip(I)$
STEP 2: $S(p_1) \leftarrow 0, \ldots, S(p_t) \leftarrow 0$
STEP 3: $\sigma' \leftarrow MACGen(S, K)$
STEP 4: 만약 $\sigma = \sigma'$이면 1 그렇지 않은 경우 0을 반환한다.

알고리즘 5 인증 정보 검증 알고리즘

4. 성능 비교 분석

본 절에서는 제안 방법 및 이전 방법들에 대한 성능 비교 분석을 수행한다. 성능 비교 분석은 3.1.1절에서 기술한 성능 비교 분석 척도인 오탐율과 왜곡량에 대하여 비교 한다. 왜곡량에 대한 명확한 정의는 기존의 연구에서 명확히 정의되지 않은 관계로 본 논문에서는 본 연구와 다른 연구들의 왜곡량을 비교하는 척도로 다음의 3가지 방법을 제시한다.

1) 변환되는 화소의 수: 본 척도는 특정한 양의 정보 삽입 시, 정보 삽입을 위해 변환되는 화소의 수를 비교한다. 본 척도는 전체 이미지의 크기, 변환 가능 화소의 수등이 나오면 정량적인 분석이 가능하다. 본 연구에서는 본 척도에 대한 비교를 정량적인 수식 분석을 통해 기술한다.
2) 시각적 이미지 왜곡: 본 척도는 인간의 시각 시스템이 정보 삽입으로 인해 얼마나 이미지 영역에서의 왜곡을 인식하느냐를 구분하는 척도이다. 일반적으로, 이미지 영역에서의 왜곡도를 측정하는 척도로 널리 알려진 것은 PSNR(Peak-Signal-to-Noise Ratio)이다. 본 연구에서는 원 이미지와 정보 삽입 이미지 간에 PSNR 분석을 통해 제안 방법이 얼마나 이미지 영역에 대해 왜곡되었는지 분석한다. 원이미지와

정보 삽입 이미지 간의 PSNR은 아래와 같이 정의된다.

$$PSNR(dB) = 10\log_{10}\frac{P^2 \times width \times height}{\sum_{x=0}^{width-1}\sum_{y=0}^{height-1}(g(x,y)-f(x,y))^2}$$

여기서 $f(x,y)$는 원 이미지의 (x,y) 좌표[3]에서의 화소값이며, $g(x,y)$는 정보 삽입 이미지의 화소값이 된다. 또한 P 는 최대 peak-to-peak 신호 차이값을 의미하는데, 이진 이미지에서는 화소값이 0 또는 1밖에 없으므로 $P=1$이 된다. 또한 $width$,$height$는 이미지의 너비 및 높이가 된다.

PSNR에 부가한 또 한 가지의 척도로, 이진 이미지에 대한 이미지 영역에서의 시각적 왜곡의 정도를 정량화 시키려는 최근 연구 중 [12,17,18] 가장 좋은 성능을 나타내는([12] 참조) "Edge Line Segment Similarity Measur(ELSSM)"를 이용하여 제안 방법의 이미지 영역에서의 왜곡을 분석한다[12].

왜곡량 분석시, 시각적 이미지 왜곡과 관련된 PSNR 및 ELSSM 은 정보 삽입에 사용되는 원 이미지에 따라 많은 차이를 보이므로, 본 연구에서는 다수개의 이진 이미지를 원 이미지로 하여 제안 방법과 기존까지의 연구 중 최선의 성능을 보이는 방법 중의 하나인 Yang 등의 방법[7]을 구현하여 직접 정보 삽입 이미지를 생성한 후, 해당 이미지들의 PSNR 및 ELSSM 값의 비교를 통해 제안 방법의 이미지 영역에서의 왜곡량에 대한 상대적 분석을 수행한다.

본 절의 구성은 다음과 같다. 우선 첫째 세부절에서 제안 방법에 대한 오탐율 및 변환되는 화소 수에 대한 분석을 수행하고, 두 번째 세부절에서는 제안 방법과 이전의 방법들 간의 오탐율 및 왜곡량 측정 요소 중의 하나인 삽입으로 인한 변환 화소의 개수에 대한 비교 분석 결과를 기술한다. 세 번째 세부절에서는 제안 방법과 Yang 등의 방법을 구현하여 실제 이진 이미지에 정보 삽입을 수행한 결과를 보이고, PSNR 및 ELSSM 등의 이미지 영역에서의 성능 비교를 수행한다. 마지막 세부절에는 제안 방법의 공격에 대한 안정성에 대해 논의한다.

4.1 제안 방법의 분석

제안 방법에서는 정보 삽입에 의해 값이 변화하는 화소의 수는 각 블록당 최대 1개이며, 블록의 개수는 삽입 정보의 비트 길이에 따라 달라진다. 만약, 삽입 정보의 비트 길이가 k일 경우, 삽입 가능 블록의 수 n은 $k=n\lfloor(\log(\lfloor t/n \rfloor+1))\rfloor$을 만족 한다. 이 경우, 각 블록에서 임의의 한 점이 변화될 확률은 $(1-(1/2)^{\log_2(\lfloor t/n \rfloor+1)})$이 된다. 이러한 이유는 *HCBEmbed* 함수의 입력값으로 들어가는 이미지 블록이 삽입하고자 하는 메시지 자체를 이미 포함하고 있는 경우, 해당 블록에 대해서는 픽셀값 변환이 일어나지 않는다. 이러한 확률은 $1/(2^{\lfloor(\log(\lfloor t/n \rfloor+1))\rfloor})$이 되며, 이외의 경우에는 모두 입력값인 이미지 블록에서 1 화소값이 변화하게 되므로 결과적으로 블록당 $(1-(1/2)^{\log_2(\lfloor t/n \rfloor+1)})$ 만큼의 화소가 평균적으로 변하게 된다. 따라서 제안 방법을 사용할 때, 평균적으로 변화하는 화소의 수는 $n(1-(1/2)^{\log_2(\lfloor t/n \rfloor+1)})$이 된다.

제안 방법의 오탐율은 다음과 같다. 우선 삽입 가능 화소 이외의 화소를 바꿨음에도 불구하고 정보 추출/검증자의 검증 테스트를 통과할 확률은 정보 추출자가 추출하는 MAC값을 암호학적으로 위조할 수 있는 가능성과 같은 값이 된다. 이것은 삽입 가능 화소 이외의 화소들은 인증 정보 삽입 알고리즘에서 수행하는 MAC 생성 알고리즘의 메시지 부분을 차지하기 때문이다. 만약, 삽입 가능 화소를 변조할 경우에, 해당 화소가 MAC 값 계산에 참여하는 화소일 경우에는 무조건 인증 정보 검증 알고리즘을 통과 할 수 없다. 왜냐하면 동일한 입력 메시지 및 동일한 키 값으로 생성되는 MAC값은 유일하기 때문이다. 따라서 제안 방법의 오탐율은 MAC값을 암호학적으로 위조할 수 있는 가능성과 같게 된다. 이 가능성은 매우 작은 값을 갖게 되며 생성되는 인증 정보의 비트 길이 k값에 의존하며 k비트 인증 정보를 사용할 때의 비트 안전도(Security in Bits)는 약 $k/2$가 된다[19]. 부록의 표 4에 비트 안전도 및 다른 공개키 암호 알고리즘의 안전도에 대한 상관관계가 나타나 있다.

4.2 제안 방법의 타 방법들과의 분석을 통한 비교 결과

제안 방법과 타 방법과의 비교 분석 결과가 표 1에 자세히 기술되어 있다. 직접적인 분석 결과로 비교가 가능한 결과는 제안 방법과 김해용 및 Yang 등의 방법이다. 제안 방법은 삽입 정보량이 변환 가능 화소의 수보다 클 경우, 김해용 및 Yang 등의 방법 보다 값이 변하는 화소의 개수에서 좀 더 유리함을 알 수 가 있다. 또한 표 1에서 알 수 있듯이 Wu 등의 방법[6]은 매우 높은 오탐율을 갖기 때문에 이미지 인증을 위한 정보 삽입 알고리즘으로는 적합하지 않다.[4]

실제적으로 사용되는 이진 이미지는 약 1만 화소 (100×100) 이상의 화소를 갖는 이미지 이고, 이 경우 일반적으로 5% 정도의 화소가 변환 가능 화소이다[6].

3) 원 이미지 및 정보 삽입 이미지의 두가지 모든 경우에 대해, 이미지의 가장 왼쪽의 최상단 화소의 좌표를 (0,0)으로 정의하고, 가장 오른쪽의 최하단 화소의 좌표를 $(width-1,height-1)$로 정의한 것을 가정한다.

4) 이러한 이유에 대해서는 부록 B에 좀 더 자세히 기술되어있다. Wu 등의 방법[6]은 높은 오탐율을 갖고 있는 이유로 그림 3의 비교에서는 제외하였다.

표 1 제안 방법과 이전 방법들과의 성능 비교

방법들	Tzeng 등 [1]	김해용 및 Yang 등 [2-5,7]	Wu 등 [6]	제안 방법
값이 변하는 화소의 갯수	$N\sum_{i=0}^{k}(i*(A_i-\sum_{j=0}^{i-1}A_j))$ $(A_l=1-(\frac{1}{2^k}\sum_{j=l+1}^{k}\binom{k}{j})^q)$	$h/2$	$h/2$	$n(1-(1/2)^{\log_2(\lfloor t/n \rfloor+1)})$ $\approx n(1-(1/2)^{(h/n)})$ $(h=n\lfloor(\log(\lfloor t/n \rfloor+1))\rfloor$ 일 경우)
오탐율	$\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}2^{-ik}$	C	$1/h$	C
기호 설명	k: codeword의 길이, N: 사용된 블록의 수, q: code holder의 수 C: 사용 알고리즘의 안전성 (표. 4 참조)			

따라서 이 경우, 실제적으로 사용 가능한 안전도를 갖는 224 비트 길이의 HMAC 알고리즘을 인증 정보로 사용될 경우의 제안 방법과 Tzeng[1] 및 김해용/Yang [4,5,7] 등의 알고리즘을 사용할 경우의 값이 변화는 화소의 개수는 아래의 그림 3과 같이 나타내어진다. 그림 3에서 알 수 있듯이 원 이미지의 크기가 증가 할수록, 제안 방법이 타 방법들에 비해 좀 더 값이 변하는 화소의 개수가 줄어듦을 알 수 있다. 그림 3에서 각 알고리즘은 모두 같은 오탐율($\approx 2^{-112}$)을 갖고 있는 경우의 비교이다.

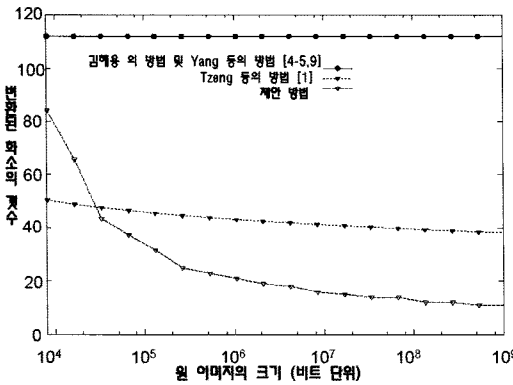


그림 3 224 비트 길이의 HMAC 알고리즘을 이용하여 생성한 MAC 정보를 원이미지에 삽입할 경우, 원 이미지 크기의 변화에 따른 변환된 화소의 개수의 변화도(단, Tzeng의 알고리즘은 나머지 알고리즘들과 같은 오탐율을 갖도록 인자를 설정한 경우임)

4.3 실험을 통한 제안 방법의 Yang 등의 방법과의 비교 결과

본 절에서는 제안 방법의 이미지 영역의 분석을 위해 다양한 문자 및 그림 이미지에 대하여 제안 방법 및 Yang 등의 방법에 대한 정보 삽입 결과 및 이미지 영역에서의 왜곡 분석 척도인 ELSSM(Edge Line Segment Similarity Measure)[12] 및 PSNR(Peak Signal-to-Noise Ratio)을 통한 분석 결과를 보여준다. ELSSM은 이진 이미지의 시각적인 이미지의 왜곡 정도를 정량적으로 분석하기 위해 최근 제안되었다. ELSSM에서의 왜곡량 측정은 이미지가 왜곡되었을 때, 이미지를 내부에 존재하는 각 세그먼트(segment: 서로 연결되어 있는 검은색 점들의 집합)로 나누어 각 세그먼트 마다 왜곡되기 전후의 외곽선의 변화를 측정하는 것이다. 본 방법은 이전까지 알려진 이진 이미지에 대한 이미지 왜곡 측정 척도인 DRDM(Distance Reciprocal Distortion Measure)[17] 및 CSCM(Change in Smoothness and Connectivity Measure)[18]에 비해 인간의 시각 시스템이 인지할 수 있는 시각적 왜곡 정도를 좀 더 정확히 반영할 수 있는 방법으로 알려져 있다[12].

본 연구에서는 6개의 문자 이미지와 3개의 그림 이미지에 대한 실험을 수행하였다. 각 이미지에 대한 정보는 표 2에 기재되어 있다. 이미지에 삽입하는 정보는 224-bit의 HMAC 정보[16]를 삽입하였다. 삽입된 이미지에 대한 ELSSM 및 PSNR 을 각각 측정하였다. 그림 4는 English-1 이미지와 hunter 이미지[6]에 제안 방법과 Yang 등의 방법[7]을 이용하여 정보를 삽입했을 때, 변화된 화소의 위치를 나타낸다. 변화된 화소의 개수는 제안 방법이 Yang 등의 방법보다 상당히 적음을 알 수 있다.

표 2는 다양한 이미지에 대해 224-bit HMAC을 삽입시에, 제안 방법과 Yang 등의 방법의 PSNR 및 ELSSM을 이용한 왜곡량을 나타낸다. 여기서 ELSSM은 값이 적을수록 이미지가 시각적으로 덜 왜곡되었음을 나타내며, PSNR은 높은 값을 가질수록, 이미지의 왜곡이 덜 되었음을 나타낸다. 결론적으로, 제안 방법은 기존의 연구보다 이미지 영역에서 좀 더 적은 왜곡량을 나타냄을 알 수 있다.

4.4 기존의 이진 이미지 정보 삽입 방법에 대한 공격에 대한 안정성

이진 이미지의 정보 삽입 방법에 대한 공격 방법은 최근에 2가지가 제안되었다[8,9]. 하나는 이진 문자 이미지에서의 정보 삽입 방법에 대한 공격이고[8], 다른 하

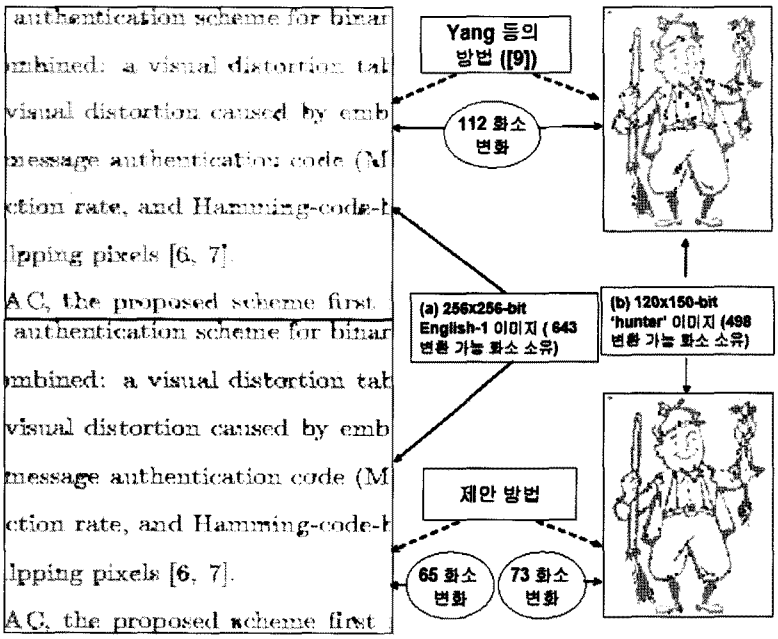


그림 4 제안 방법과 Yang 등의 방법을 이용하여 224-bit HMAC을 삽입한 결과

표 2 PSNR과 ELSSM에 대한 제안 방법과 Yang의 방법과의 비교 (224-bit HMAC 삽입시)

	ELSSM [12]		PSNR (dB)	
	Proposed	Yang et al. [7]	Proposed	Yang et al.
English-1 (256 × 256)	61.433	196.091	32.367	27.673
English-2 (389 × 214)	77.267	233.239	34.152	28.711
Chinese (327 × 101)	94.945	250.503	28.754	24.697
Japanese (342 × 178)	75.022	263.053	32.793	27.352
Korean-1 (295 × 227)	93.161	218.081	31.726	27.766
Korean-2 (217 × 177)	87.549	252.186	30.046	25.352
Photo (640 × 337)	49.615	180.951	38.287	32.885
hunter [6] (120 × 150)	90.182	247.029	25.563	22.061
Clinton's signature [6] (288×48)	156.487	210.257	22.112	20.914

나는 그림 이미지에 대한 공격이다[9].

첫 번째 공격 방법은 주어진 이진 이미지에 정보가 삽입되었는지의 여부를 판별하는 공격이다. 해당 방법은 기존의 이진 이미지를 이용한 정보 삽입 방법이 시각적으로 왜곡이 덜 되는 3×3 화소에서의 L-유형에 많은 정보가 삽입된다는 것을 근거로 공격을 수행하였다(L-유형은 아래의 그림 5에 나타내어져 있다.) 해당 방법은 우선 전체 이미지를 여러개의 세그먼트들로 나눈 후, 해당 세그먼트들을 서로 모양의 유사성을 근거로 하여 유사도가 높은 세그먼트들을 묶어 그룹을 만들었다. 그 후 그룹 내의 각 세그먼트들의 각 위치의 화소값들을 평균을 내어 반올림한 값으로 그룹 대표 세그먼트를 만든 후, 그룹 대표 세그먼트와 그룹 내의 각 세그먼트의 L-유형의 중앙 화소값을 다음과 같은 방법으로 비교하였다.

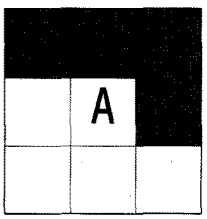


그림 5 L-유형의 한 예(A는 L-유형의 중앙 화소) 좌우 대칭, 회전 및 화소 반전 등에 의해 16개의 L-유형이 존재한다.

만약, 그룹 G_i의 대표 세그먼트를 s_r이라 하고 G_i의 각 멤버 세그먼트를 $s_1,\cdots,s_k$라 했을 때, $E_{G_i}, E_{G_i(e_1)}, E_{G_i(e_2)}$는 다음과 같이 정의된다.

$E_{G_1} = \{(x,y)|s_i(x,y) \neq s_r(x,y) \wedge (x,y)$는 $L-$유형의 중앙 화소$(i=1,\cdots,k)\}$

$(s(x,y)$는 세그먼트 s의 좌표 (x,y)에서의 화소값)

$E_{G_1(e_1)} = \{(x,y)|(x,y) \in E_{G_1} \wedge (x,y)$의 인접화소에 $L-$유형의 중앙 화소가 존재$\}$

$E_{G_1(e_2)} = \{(x,y)|(x,y) \in E_{G_1} \wedge (x,y)$의 인접화소 중 $L-$유형의 중앙 화소가 존재하지 않음$\}$

해당 논문에서는 $E_{G_1}, E_{G_1(e_1)}, E_{G_1(e_2)}$을 계산한 후, G_1의 대표 값 $\xi_{1(G_1)} = |E_{G_1(e_1)}|/|E_{G_1}|$, $\xi_{2(G_1)} = |E_{G_1(e_2)}|/|E_{G_1}|$에 대한 계산을 수행하였다. 이 후 모든 다른 그룹들에 대하 동일한 과정을 수행한 후, 생성된 ξ_1, ξ_2 값들을 각각 평균을 내어 평균값 사이의 차이를 비교하였다. 해당 논문에서는 정보가 삽입된 이진 이미지는 (해당 차이값을 λ라 한다면) $\lambda > 0.04$이며 그렇지 않을 경우 $\lambda \leq 0.04$의 경향을 보인다는 것을 실험을 통해 증명하였다.

본 연구에서는 [8]의 논문의 방법을 구현하여 제안 방법을 이용한 정보 삽입 이미지에 대해서 [8]의 분석을 수행하였다. 해당 공격 방법은 이진 문자 이미지에 대한 삽입 방법에 대한 공격이므로 이진 문자 이미지에 대하여만 분석을 수행하였다. 분석 결과는 아래의 표 3에 나타나 있다. 분석 결과로부터, 제안 방법은 [8]의 공격에 안정성이 있음을 알 수 있다.

표 3 제안 방법의 [8]의 공격에 대한 안정성($\lambda > 0.004$ 이면 정보 삽입 이미지로 판별됨[8])

실험 이미지	제안 방법을 사용한 정보 삽입 이미지의 λ	원 이미지의 λ
English-1	0.0067	0.0156
English-2	0.0126	0.0132
Chinese	0.0069	0.0000
Japanese	0.0012	0.0067
Korean-1	0.0212	0.0128
Korean-2	0.0046	0.0068

두 번째 공격 방법은 이진 그림 이미지에 대한 정보 삽입 알고리즘에 대한 공격이다[9]. 본 방법은 사전에 정의된 규칙을 이용하여 정보가 삽입된 것으로 추정된 이미지를 보정한 이미지를 생성한다. 이후 보정된 이미지와 원래의 정보 삽입 이미지 사이의 이미지 왜곡을 분석하여 만약, 왜곡 정도가 클 경우는 정보 삽입 이미지로 판별하는 방법을 사용하였다. 이미지 왜곡 분석에 사용된 요소는 PSNR, ELSSM 등이다.

제안 방법은 기존의 다른 방법보다 원 이미지와 정보 삽입 이미지간의 PSNR 및 ELSSM의 차이가 상당히 적다. 따라서 본 공격 방법을 사용할 경우, 보정된 이미지와 제안 방법으로 정보가 삽입된 이미지 사이의 PSNR, ELSSM 값이 이전의 방법들 보다 더욱 적은 값을 나타낼 것이며, 결과적으로 [9]의 공격 방법은 제안 방법을 정보를 삽입한 이미지에 대해서 정보가 삽입되었는지의 여부를 판별하기 어렵게 된다. 따라서 제안 방법은 [9]의 공격에 대해 이전의 다른 방법들 보다 좀 더 높은 안정성을 갖는다.

4. 결 론

본 논문에서는 적은 양의 이미지 왜곡으로 정보를 삽입할 수 있는 이진 이미지 인증용 정보 삽입 방법을 제안하였다. 제안 방법을 사용할 경우, 동일한 인증 정보 삽입 시 기존의 방법들보다 반전되는 화소의 수가 크게 감소함을 보였으며 또한 PSNR 및 ELSSM의 분석을 통해 제안 방법이 기존의 방법들에 비해 시각적인 이미지 왜곡도 줄일 수 있음을 보였다. 또한 기존의 정보 삽입 여부 검출 공격들에 대하여도 좀 더 안정성을 갖음을 실험을 통해 증명하였다.

참 고 문 헌

[1] C. Tzeng, and W. Tsai, "A New Approach to Authentication of Binary Images for Multimedia Communication With Distortion Reduction and Security Enhancement," IEEE Communications Letters, Vol.7, No.9, pp. 443-445, Sep. 2003.

[2] H. Kim, and A. Afif, "Secure Authentication Watermarking for Binary Images," *Brazilian Symposium on Computer Graphics and Image Processing*, pp. 199-206, 2003.

[3] H. Kim, and A. Afif, "A Secure Authentication Watermarking for Halftone and Binary Images," International Journal of Imaging Systems and Technology, Vol.14, No.4, pp. 147-152, 2004.

[4] H. Kim, "A New Public-Key Authentication Watermarking for Binary Document Images Resistant to Parity Checks," in the Proceedings of IEEE Int. Conf. Image Processing(ICIP) 2005, Vol.2, pp. 1074-1077, 2005.

[5] H. Kim, R. Queiroz "Alteration-Locating Authentication Watermarking for Binary Images," In Proceedings of International Workshop on Digital Watermarking (IWDW) 2004, LNCS Vol.3304, pp. 125-136, 2005.

[6] M. Wu, and B. Liu, "Data Hiding in Binary Images for Authentication and Annotation," IEEE Trans. Multimedia, Vol.6, No.4, pp. 528-538, 2004.

[7] H. Yang, and A. C. Kot, "Pattern-Based Data Hiding for Binary Image Authentication by Connectivity-Preserving," IEEE Trans. Multimedia, Vol.9, No.3, pp. 475-486, 2007.

[8] J. Cheng, A. C. Kot, J. Liu, and H. Cao, "Detection of Data Hiding in Binary Text Image," In proc.

IEEE Int. Conf. on Image Processing (ICIP) 2005, Vol.3, pp. 73-76, 2005.

[9] J. Cheng, A. C. Kot, and S. Rahardja, "Steganalysis of Binary Cartoon Image Using Distortion Measure," In proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP) 2007, Vol.2, pp. 261-264, 2007.

[10] Tim Downey, *Calculating the Hamming Code*, [Online], Available: http://www.cs.fiu.edu/~downeyt/cop3402/hamming.html, 2004.

[11] J. Fridrich and D. Soukal, "Matrix Embedding for Large Payloads," in Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents VIII, Vol. 6072, San Jose, CA, January 16-19, pp. W1-W15, 2006.

[12] J. Cheng, and A. C. Kot, "Objective Distortion Measure for Binary Text Image Based on Edge Line Segment Similarity," IEEE Tran. Image Processing, Vol.16, No.6, 2007.

[13] S. Myers, "Efficient-Amplification of the Security of Weak Pseudo-random Function Generators," In Proc. EUROCRYPT 2001, Lecture Notes in Computer Sciences, Vol.2045, pp. 358-372, 2001.

[14] S. G. Akl, and H. Meijer, "A Fast Pseudo Random Permutation Generator with Applications for Cryptology," In Proc. CRYPTO 1985, Vol.196, pp. 269-275, 1985.

[15] M. Luby, and C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," SIAM J. on Computing, Vol.17, No.2, pp. 373-386, Apr. 1988.

[16] D. Eastlake and T. hansen, "US Secure Hash Algorithms (SHA and HMAC-SHA)," IETF RFC 4634, Available: http://www.ietf.org/rfc/rfc4634.txt?number=4634, 2006.

[17] H. Lu, A. C. Kot, and Y. Shi, "Distance-Reciprocal Distortion Measure for binary document images," IEEE Signal Processing Letters, Vol.11, No.2, pp. 228-31, 2004.

[18] J. Cheng, and A. C. Kot, "Objective distortion measure for binary images," In proc. IEEE TENCON, pp. 355-358, 2004.

[19] "Security Requirements for Cryptographic Modules: Compromising the security of the key establishment method," in FIPS 140-2, NIST.

부 록

A. 해밍 코드 기반 정보 삽입/추출 방법

해밍 코드는 2^n-1-n 비트 길이의 데이터가 있을 때, n비트의 패리티 비트를 추가하여 총 2^n-1 비트의 길이를 갖게 되는 코드이다. 해밍 코드는 1비트의 오류 검출에 사용되며 전체 해밍 코드 중 1비트의 오류가 발생했을 경우 해당 오류 비트의 위치를 알려준다. 해밍 코드에 대한 내용은 [10]에 기술되어 있다.

해밍 코드는 데이터 비트와 패리티 비트 부분으로 되어 있으며, 패리티 비트는 전체 해밍 코드중 2^k $(k=1,\ldots,n-1)$ 번째의 위치에 차례로 위치하며 데이터 비트는 나머지 부분에 순서대로 위치한다. 예를 들어, $7(=2^3-1)$비트 해밍 코드 $1110011_{(2)}$에서 패리티 비트 부분은 첫 번째 비트값인 1, 두 번째 비트 값인 1, 네 번째 비트 값인 0이 되며 패리티 비트 부분만을 값으로 나타낼 경우 해밍 코드에 기술된 순서의 역순인 $011_{(2)}$로 나타낸다. 만약 해밍 코드 중에서 패리티 비트 부분에 속하는 a번째 비트의 값을 변환시키면, 그 때의 패리티 비트 부분의 값은 원래의 패리티 비트 부분값에 a를 XOR시킨 값이 된다. 예를 들어 위의 해밍 코드 $1110011_{(2)}$에서 네 번째 비트 값을 0에서 1로 변환시키면, 패리티 비트 부분의 값은 $011_{(2)}$ XOR $100_{(2)}$ = $111_{(2)}$가 된다.

원 이미지의 크기가 2^n-1비트일 경우, 본 정보 삽입 방법을 이용하면 n비트 길이의 정보 m에 대한 삽입이 가능하며 이를 일반화하여 원 이미지의 크기가 c비트일 경우, 최대 $\lfloor \log_2(c+1) \rfloor$ 비트 길이의 정보 삽입이 가능한 해당 알고리즘은 아래와 같이 기술된다.

해밍 코드 기반 정보 삽입 알고리즘 (HCBEmbed) [11]
입력값: 원 이미지 $I \in \{0,1\}^c$, 입력 대상 정보 $m \in \{0,1\}^{\lfloor \log_2(c+1) \rfloor}$
결과값: 정보 삽입 이미지 $S \in \{0,1\}^c$
STEP 1: I를 해밍 코드 워드로 간주한다. 해밍 코드 워드로 간주하였을 때, I의 데이터 비트 부분을 d ($2^{\lfloor \log_2(c+1) \rfloor}-1-\lfloor \log_2(c+1) \rfloor$ 비트), 패리티 비트 부분을 p($\lfloor \log_2(c+1) \rfloor$ 비트)라 한다.
STEP 2: d로부터 새로운 해밍 코드워드 $I_{ham} \in \{0,1\}^c$를 계산한다. I_{ham}의 패리티 부분을 p_{ham}이라 한다.
STEP 3: 만약 $p \oplus p_{ham} \neq m$이라면 I에서 $(p \oplus p_{ham} \oplus m)$ 번째의 화소의 값을 변환시킨다.
STEP 4: $S \leftarrow I$, S를 반환한다.

위의 과정을 통하여 생성된 정보 삽입 이미지 S로부터 삽입 정보 m을 추출해내는 알고리즘은 아래와 같다.

해밍 코드 기반 정보 추출 알고리즘 (HCBExt) [11]
입력값: 정보 삽입 이미지 $S \in \{0,1\}^c$,
결과값: 추출 정보 $m \in \{0,1\}^{\lfloor \log_2(c+1) \rfloor}$
STEP 1: S를 해밍 코드 워드로 간주한다. 해밍 코드 워드로 간주하였을 때, S의 데이터 비트 부분을 d' ($2^{\lfloor \log_2(c+1) \rfloor}-1-\lfloor \log_2(c+1) \rfloor$ 비트), 패리티 비트 부분을 p' ($\lfloor \log_2(c+1) \rfloor$ 비트)라 한다.
STEP 2: d'로부터 새로운 해밍 코드워드 $S_{ham} \in \{0,1\}^c$를 계산한다. S_{ham}의 패리티 부분을 p'_{ham}이라 한다.
STEP 3: $m \leftarrow p' \oplus p'_{ham}$, m을 반환한다.

B. 이전 방법들에 대한 분석

Tzeng 등의 방법 분석

Tzeng 등의 방법은 다음과 같은 가정하에 분석을 수행한다. 코드 워드의 비트 길이를 k라 가정하고, 코드 홀더는 q개가 존재한다고 가정한다. 또한 q개의 코드 워드가 선택한 결과 비트는 균일 분포(uniform distribution)에서 선택된 것으로 가정한다. 마지막으로 제안 방법과 같이, 전체 커버 이미지의 비트 길이는 c이고 N개의 블록으로 나누어 처리한다고 가정한다. Tzeng의 방법에서는 하나의 공유된 키를 이용하여 블록마다 삽입되는 인증 코드 $c_1,...,c_n$를 생성하고, 또 다른 공유키를 이용하여 코드 홀더를 생성한다. 따라서 인증 코드 $c_1,...,c_n$도 집합 $\{0,1\}^k$에서 uniform한 확률 분포로 각각 선택된다고 가정한다. 이러한 경우 삽입되는 인증 코드의 크기는 k^*N bit가 된다. 블록 당 평균 반전 화소 수는 다음과 같다. 임의의 i 번째 블록의 코드 워드 $c_i(i=0,...,N)$에 대하여 q개의 코드 홀더가 선택한 k비트 열을 각각 $t_{1,...,}t_q$라 한다면, i 번째 블럭에서 반전되는 화소의 개수는 $t_{1,...,}t_q$ 중 인증 코드 c_i와 비교하여 가장 비트 차이가 적은 것과 c_i와의 비트 차이의 개수가 된다. $t_{1,...,}t_q$가 독립적으로 $[0,2^{c/N-1}]$에서 균일한 확률 분포로 하나씩 선택된다면 각각과 c_i와의 비트 차이에 대한 확률 분포는 이항 분포를 이루게 된다. 이러한 사실을 바탕으로 c_i와 $t_{1,...,}t_q$중 최소 비트 차이를 갖는 값과의 비트 차이에 대한 확률 분포에 대해서 평균값을 구하면 그 값이 블록당 평균 반전 화소 수가 된다.

(블록 당 반전 화소 개수의 평균값) =

$$\sum_{i=0}^{k}(i^*(A_i-\sum_{j=0}^{i-1}A_j))\ (A_l=1-(\frac{1}{2^k}\sum_{j=l+1}^{k}\binom{k}{j})^q\ )$$

위의 값에 블록의 개수인 N을 곱한 값이 바로 전체 이미지에 대한 반전 화소 수가 된다. Tzeng의 방법의 오탐율은 다음과 같다. 하나의 블록에서 생각해 본다면, 블록 정보 중, 인증 코드를 담고 있는 코드 워드 부분을 제외하고 나머지 블록 정보는 변형시켜도 검증 과정을 통과한다. 또한 원래의 인증 코드를 담고 있는 코드 홀더 이외에, 다른 코드 홀더 위치에 인증 코드가 들어가도록 이미지를 변조 시킬 경우에도 검증 과정을 통과하게 된다. 위의 과정을 종합하여 Tzeng의 방법의 오탐율을 계산하면 아래와 같다. 워터마크는 블록 별로 서로 독립적으로 삽입되고 검증되므로 블록 당 오탐율이 곧 전체 이미지의 오탐율이 된다.

(Tzeng et al's scheme의 오탐율) =

$$\sum_{i=1}^{q}(-1)^{i+1}\binom{q}{i}2^{-ik}$$

김해용 등의 방법 분석

김해용의 방법은 이미지의 크기에 관계없이, 공개된 의사 난수 생성기를 이용하여 워터마크를 삽입할 위치를 선택한 후에, 해당 워터마크가 삽입된 위치를 모두 0으로 초기화한 이미지에 대한 암호학적 서명 또는 MAC 정보를 삽입함으로써 수행된다. 따라서 삽입되는 인증 코드의 크기도 해당 암호학적 서명 알고리즘 및 MAC 알고리즘의 사용 비트 길에 의존한다. 또한 커버 이미지를 스테고 이미지로 변화시키는 데 필요한 평균 반전 화소 수는 사용되는 암호 알고리즘에 좌우된다. 예를 들어, 스테고 이미지에 224bit HMAC 인증 코드가 들어갈 경우 약 112개의 화소가 반전되며, 1024bit RSA 전자 서명이 포함될 경우 약 512개의 화소가 반전된다. 김해용 방법의 안전성은 전적으로 사용 알고리즘의 안전성에 의존한다. 표 4는 NIST의 FIPS 140-2에 기술되어 있는 암호 알고리즘의 안전도에 대한 정의이다. 표 중에서 안전성 항목을 오탐율로 보았을 때, 80 bit의 안전도를 갖을 경우 약 2^{-80}의 오탐율을 갖는다고 간주할 수 있다.

Wu 등의 방법 분석

Wu 등의 방법은 하나의 변환 가능 화소가 블록 내에 존재하고 각 블록당 1비트의 정보 삽입이 가능하며, 블록 내의 검은 화소의 개수에 따른 Even-Odd 삽입 방법을 사용한다. 즉, 정보 추출자는 한 블록에 검은 화소의 개수가 짝수이면 1비트 정보 0이 삽입 되었다고 간주하고, 검은 화소의 개수가 홀수이면 1이 삽입 되었다고 간주한다. 따라서 만약 h비트의 정보를 삽입할 경우, 평균적으로 $h/2$ 개의 화소의 값이 변하게 된다.

Wu 방법의 오탐율은 $1/h$이 된다. 이것은 [4] 논문

표 4 FIPS 140-2에 기술되어 있는 암호 알고리즘의 비트 안전도 [19]

Security (Bits)	Symmetric encryption algorithm	공개키의 최소 비트 길이		
		DSA/DH	RSA	ECC
80		1024	1024	160
112	3DES	2048	2048	224
128	AES-128	3072	3072	256
192	AES-192	7680	7680	384
256	AES-256	15360	15360	512

에서 지적한 Parity Attack이 가능하기 때문이다. 예를 들어, 공격자가 한 블록에 대한 정보를 알고 있으면, 해당 블록의 검은 화소의 개수를 세고, 그 개수를 맞추어서 블록 내의 화소의 값을 변화시키면 인증 정보/검증자는 정보 삽입 이미지가 위조되었는지 확인할 방법이 없다. 이는 매우 높은 오탐율이며, 따라서 Wu의 방법은 이미지 인증을 위한 정보 삽입 방법으로는 부적당한 방법이다.

이 윤 호

2000년 한국과학기술원 전산학과 학사 2002년 한국과학기술원 전자전산학과 전산학 전공 석사. 2006년 한국과학기술원 전자전산학과 전산학 전공 박사. 2006년 9월~2007년 8월 한국과학기술원 정보전자연구소 박사후 연구원. 2007년 10월~2009년 2월 Visiting scholar & Research staff, Georgia Tech Information Security Center. 2009년 3월~현재 영남대학교 전자정보공학부 정보통신공학전공 전임강사. 관심분야는 네트워크 보안, 멀티미디어 보안, 응용 암호

김 병 호

1990년 연세대학교 전산과학과(학사). 1992년 KAIST 전산학과(석사). 1997년 KAIST 전산학과(박사). 1997년~1998년 포스데이터 과장. 1998년~2005년 브레인21 대표이사. 2007년~현재 경성대학교 전임강사. 관심분야는 센서 네트워크