2013년 8월 전자공학회 논문지 제 50 권 제 8 호
Journal of The Institute of Electronics Engineers of Korea Vol. 50, NO. 8, August 2013 http://dx.doi.org/10.5573/ieek.2013.50.8.180

논문 2013-50-8-22

성능을 개선한 해밍 코드 기법을 이용한 데이터 은닉

(Data Hiding using Improving Hamming Code)

김 천 식*

(Cheonshik Kim©)

요 약

스테고(Stego) 이미지 분석 측면에서 스테고-분석(Steganalysis) 툴의 기본적인 공격 목표는 데이터 은닉에 의한 스테고-미디어의 통계적으로 비정상적인 부분을 찾음으로서 데이터 은닉의 존재를 탐지 하는 것이다. 본 논문에서는 디지털 이미지를 위해 (7, 4) 해밍코드를 개선한 방법으로 데이터 은닉 스킴을 제안하였다. 제안한 방법은 원 영상이미지에 9비트 마다 6비트의 데이터를 은닉하는 방법이다. 실험결과 제안한 방법은 0.67bpp의 데이터 은닉 성능을 보이며 기존의 방법들에 비해서 스테고 영상의 질이 평균적으로 약간 더 좋은 것으로 나타났다.

Abstract

The primary goal of attack on steganographic images, termed steganalysis, is to detect the presence of hidden data by finding statistical abnormality of a stego-media caused by data embedding. This paper proposes a novel steganographic scheme based on improving the (7, 4) Hamming code for digital images. The proposed scheme embeds a segment of six secret bits into a group of nine cover pixels at a time. The experimental results show that the proposed scheme achieves a 0.67bpp embedding payload and a slightly higher visual quality of stego images compared with the previous arts.

Keywords : Data Hiding, Steganography, LSB, BMP

Ⅰ. 서 론

현재 인터넷에는 많은 디지털 콘텐츠 (예: 동영상, 음악, 이미지, 디지털 북 등.)가 불법적인 방법으로 공유되고 있다. 예를 들어, 개인이 동의하지 않은 저작물을 사적인 이익을 위해서 상업용 공유사이트를 통해서 공유하거나 왜곡한 개인 정보를 공유사이트나 개인 홈페이지를 통해서 퍼트리는 것이다. 이와 같은 문제점을 해결하기 위해서, 많은 연구자들이 디지털 저작권 보호 방안을 연구 및 개발하고 있다.

워터마크(Watermarking)[21]는 대표적인 저작권 보호 기술이다. 워터마크에 의한 저작권 보호 기술은 영상과 음반 분야에서 많이 적용되어 그 가치를 인정받고 있다. 데이터 은닉(Data Hiding)[5, 11]과 스테가노그래피(Steganography)[3~4, 6~7, 9, 15, 17~19, 20] 기법 또한 디지털 콘텐츠의 보호를 목적으로 연구되는 분야이다. 스테가노그래피(Steganography)[12, 14, 16] 기법은 저작권 보호 및 비밀 통신을 위한 목적으로 활용되고 있다. 데이터 은닉은 디지털 콘텐츠의 저작권 보호[8]와 주석의 용도로 활용되고 있다[1, 10].

데이터 은닉과 스테가노그래피 스킴을 평가할 때 가장 중요한 요소는 비인지성(Imperceptibility)과 데이터 은닉비율이다. 비인지성이란, 공격자들이 디지털 콘텐츠

* 정회원, 안양대학교 디지털미디어공학과
(Dept. of Digital Media Engineering, Anyang Univ.)
© Corresponding Author(E-mail: mipsan@paran.com)
※ This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) by the Ministry of Education, Science and Technology (20120192).
접수일자: 2013년3월25일, 수정완료일: 2013년7월21일

로부터 은닉된 데이터의 존재를 쉽게 감지할 수 없도록 하는 방법이다. 데이터 은닉의 문제점은 데이터 은닉 량과 디지털 콘텐츠의 질이 상관관계에 있다. 즉, 데이터의 은닉 비율을 높이면 디지털 콘텐츠의 질의 나빠지고, 그 반대의 경우 디지털 콘텐츠의 질이 좋아진다. 따라서 실제 스킴을 적용하고자 할 때, 적절한 임계값을 찾아야 한다.

디지털 이미지를 이용한 데이터 은닉 기법에는 크게 두 가지 방법이 있다.

하나는 공간영역(Spatial Domain)에 기반을 둔 기법이고 다른 하나는 주파수 영역 (Frequency Domain)에 기반을 둔 기법이다. LSB (Least Significant Bit)를 활용한 기법은 공간영역에 기반을 둔 기법으로, 일반적으로 사용된다. 이 방법은 각 픽셀의 LSB를 0혹은 1로 바꿈으로써 데이터를 은닉하는 방법이다. 이 방법의 장점은 LSB만을 활용할 경우 데이터 은닉 량과 이미지의 질이 매우 높은 특성을 보이는 장점에 있다. 단점으로는 간단한 포맷의 변환만으로 저장된 데이터가 사라지는 단점이 있다[9].

주파주 영역에 기반 한 기법은 이미지를 DCT[19] 혹은 DFT[19]등으로 주파수를 변환 한 후에 데이터를 저장하는 방법으로서 데이터 은닉 량이 공간 영역에 기반을 둔 방법에 비해서 적은 단점이 있다. 하지만 다양한 공격에 대해서 강한 특성을 보인다.

본 논문에서는 데이터를 해밍 코드기법을[2] 사용해서 데이터를 은닉하고자 한다. 기존에 [3]의 기법이 데이터 은닉에 해밍 코드 기법을 적용했었다. 이 방법은 7비트의 커버 이미지 비트열에서 1비트를 바꾸고, 3비트를 저장하는 방법이다. 본 논문에서 9비트열에서 6비트를 저장하는 새로운 방법을 제안하고자 한다. 제안한 방법은 데이터 은닉 량과 이미지의 질에서 좋은 성능을 보인다.

본 논문은 구성은 Ⅱ장에서 본 연구의 기반이 되는 해밍 코드를 를 설명하며, Ⅲ장에서는 데이터를 은닉하는 알고리즘을 제안한다. Ⅳ장에서는 제안한 방법에 대해서 실험과 성능을 비교한다. Ⅴ장에서 결론과 함께 앞으로의 연구 방향을 제시한다.

Ⅱ. 관련 연구

해밍 코드는 1950년에 해밍에 의해서 소개된 해밍 (7, 4) 부호를 의미한다. 4비트의 자료를 전송하기 위해서 3비트의 패리티 비트를 추가하기 때문에 이런 이름을 가진다. 이 부호는 모든 1비트 오류를 감지해서 바로 잡을 수 있고, 2비트 오류도 탐지가 가능하다. 해밍 코드는 해밍 행렬을 활용하여 해밍 코드의 생성 및 탐지가 가능하다. 이 행렬에는 부호화에 사용하는 생성 행렬 G와 오류를 감지하고 바로 잡는 데 쓰는 확인 행렬 H가 포함되며, (7,4) 부호의 경우 이들은 다음과 같다.

$$G=\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\\0&1&1&1\\1&0&1&1\\1&1&0&1\end{pmatrix} \tag{1}$$

$$H=\begin{pmatrix}0&1&1&1&1&0&0\\1&0&1&1&0&1&0\\1&1&0&1&0&0&1\end{pmatrix} \tag{2}$$

예를 들어 “1011”에 대응하는 코드 p=(1 0 1 1)이다. 이 코드를 위한 해밍 코드의 생성은 다음과 같은 수식으로 생성이 가능하다.

$$Gp^T = c \tag{3}$$

수식 (3)에서 c는 생성된 해밍 코드를 의미한다.

$$Hc^T = s \tag{4}$$

수식 (4)에서 c은 전송받은 비트들 중 7비트를 의미하며, 계산된 s는 신드롬(syndrome)을 나타낸다. s가 0이라면 c은 에러가 없는 코드를 의미한다.

c이 2라면 두 번째 비트가 잘못되었음을 의미한다. 이 방법은 2비트 이상의 오류가 일어나지 않을 때만 오류를 고칠 수 있음을 어렵지 않게 보일 수 있다. 2비트 오류가 발생할 때도 H와의 곱이 영 벡터가 되기 때문에 오류를 감지할 수는 있지만, 1비트 오류와 2비트 오류를 구별할 수 없기 때문에 정정은 불가능하다.

Ⅲ. 제안 방법

3.1 해밍 코드의 수식 정의

패리티 비트(Parity Bit)에 의한 오류 검출은 단지 오류 검출만 되지만 해밍코드 (Hamming Code)는 오류 검출 후 오류 정정까지 가능하다. 두 해밍 행렬은 코드

생성자 (G)와 패리티 체크 행렬 (H) 로 정의 된다. d는 전송 비트를 의미한다.

$$c = \mathrm{mod}((G \times d^T)^T, 2) \quad (5)$$

엘리스(Alice)가 $d = (1101)$를 전송하기 위해서 인코딩 한다면 $c(codeword)$는 (1101001)이 된다.

$$s = (\mathrm{mod}(H \times c^T)^T, 2) \quad (6)$$

수신측에서는 공식(6)을 이용해서 데이터가 패리티 코드를 제외한 값이 정상적인 값인가 또는 그렇지 않은 값인가를 판단한다. 이때, s는 신드롬(syndrome)으로서 데이터가 무결성인 경우 $syndome$은 0이 된다.

3.2 해밍코드를 이용한 데이터은닉

코드를 보내는 측은 커버 이미지 I를 전송한다. I는 n개의 요소 (x_i^n)로 구성된다. 이 경우, $x_i(\in I)$는 i번째의 x를 의미한다.

I로부터 9개씩을 분리하여 사용한다. 이중 7개의 x를 수식 (7)과 같이 추출하여 코드워드 c로 사용한다.

$$c = (LSB(x_1), LSB(x_2), ..., LSB(x_n)) \quad (7)$$

추출한 코드워드(Code-word) c를 공식 (6)에 적용하여 신드롬을 구한다. 이때, 구한 첫 번째, 신드롬을 $syndrome_1$으로 한다. 메시지 M={m_1, m_2, m_3}은 3개의 비트 (m_i^3)로 구성되며, 전체 메시지는 M_i^n로 나타낸다. 데이터를 저장하기 위해서 $syndrome_1$ 3비트와 메시지 3비트에 대해서 xor 연산을 수식 (8)과 같이 실행한다.

$$fpos = exor(syndrome_1, M_i) \quad (8)$$

$fpos$를 10진수로 바꾼 값의 범위는 1~ 7이 되며 오류가 있는 위치를 의미한다. 해당 위치의 값이 0이면 1로 1이면 0으로 뒤집는다. $fpos$의 값이 0이면 오류가 없음을 의미한다.

$$c = (2LSB(x_1), 2LSB(x_2), ..., 2LSB(x_n)) \quad (9)$$

수식 (9)의 $2LSB()$는 픽셀 x로부터 오른쪽에서 2번째 비트를 추출하는 함수이다. 이 수식으로부터 코드워드 c를 추출한 후 공식 (6)에 적용하여 신드롬 $syndrome_2$를 구한다.

$$fpos = exor(syndrome_2, M_i) \quad (10)$$

$syndome_2$와 두 번째 3-비트 메시지 M_2의 값과$exor$ 연산을 실행한 후 $fpos$를 구한다. $fpos$의 비트 값을 각각을 $LSB(x_8)$과 $LSB(x_9)$에 나누어 저장한다. 총 3비트를 저장해야 하므로 x_8의 LSB에 1비트를 x_9의 LSB에 2비트를 저장한다.

수신측에서는 9개의 픽셀 단위로 읽어가면서 이미지에 은닉된 비트열을 추출한다. 공식 (7)을 이용하여 블록에서 $LSB(x_1) \sim LSB(x_7)$의 값을 추출한다. 이 추출된 값이 c가 된다. c에 대해서 수식(6)을 적용하여 신드롬(syndrome)을 구한다. $syndome_1$1은 블록에 은닉된 첫 번째 은닉된 비트이다. 두 번째로 은닉된 3비트를 추출하기 위해서 공식(9)를 블록에 적용한다. 적용결과 두 번째 c를 추출하게 된다. $LSB(x_8)$, $LSB2(x_9)$, $LSB(x_9)$에서 3비트를 추출한다. 추출한 3비트는 수식(10)의 $fpos$이다. $fpos$가 1~7의 범위의 값이면 c에 대해서 $fpos$위치의 값 0이면 1로, 1이면 0으로 변경한다. 이후에 c에 대해서 수식(6)을 적용하여 $syndome_2$를 구한다. $syndome_2$가 이 블록에 저장된 두 번째 3비트가 된다. 이와 같은 과정을 남은 이미지의 블록에 모두 적용하면 은닉한 비트열을 모두 추출할 수 있다.

Ⅳ. 실험 결과

제안한 데이터은닉 기법을 평가하기 위해서, 매트릭스 인코딩 즉, "$Hamming+1$" 스킴과 제안한 방법을 구현 및 비교하기 위하여 MATLAB 7.0 소프트웨어 사용 하였다. 실험에 사용된 이미지[13]는 9개의 512×512 크기의 256색상 (gray-scale)의 회색 이미지이다 (그림 1). 이미지에 은닉을 위해서 사용한 데이터는 랜덤 비트를 생성하여 사용했다.

본 실험에서 중요한 두 가지 요소는 스테고(Stego)이미지의 해상도(Quality)와 원본 이미지에 저장한 데이터의 은닉 량 (Embedding Payload) 이다. 이미지의 질과 데이터 저장량은 제안한 시스템의 성능을 평가하는데 중요한 요소다. 이미지의 질적 평가를 위해 주관적(예, 사람의 시각에 의한 평가)인 평가는 설득력이 부족 하므로 객관화를 위해서 PSNR 공식이 제안되었다. PSNR (Peak

(a) Baboon (b) Barbara

(c) Boat (d) Goldhill

(e) Jet(F16) (f) Lena

(g) Pepper (h) Tiffany

(i) Zelda

그림 1. 실험 이미지들 (256 회색 이미지)[13]
Fig. 1. Experimental Images (grayscale image)[13]

Signal to Noise Ratio)은 스테고 영상과 원본 영상에 적용된다. PSNR은 영상물의 질을 평가하는데 가장 보편적으로 사용되는 측정 방법이다.

$$PSNR = 10 \times \log_{10}\left(\frac{I_{\max}^2}{MSE}\right) dB \tag{11}$$

수식 (11)에서 MSE는 원본 영상 I와 스테고 영상 I′의 차이 값이다. MSE의 정의는 수식 (12)과 같다.

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(|I_{i,j} - I'_{i,j}|)^2 \tag{12}$$

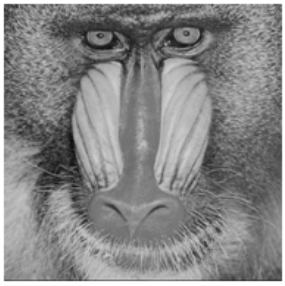
수식 (11)과 (12)의 평가에 따라서, PSNR이 큰 값일 경우 스테고 이미지가 원본 이미지와 가까운 것을 의미한다. 반대로, PSNR의 값이 작을 경우, 원본 영상물과 유사하지 않음을 의미한다. 일반적으로 PSNR이 30dB를 넘을 때, 스테고 이미지의 왜곡은 인간의 시각으로는 탐지되기 어려움을 의미한다. 원본 이미지에 은닉된 데이터량(Quantity)을 측정하기 위해서, bpp(bits-per-pixel) 단위가 평가에 사용된다. bpp는 얼마나 많은 비트들이 은닉될 수 있는가를 측정하는 단위이다. 은닉 비율은 다음과 같이 수식 (13)를 사용하여 측정이 가능하다.

$$P = \frac{\|S\|}{H \times W}(bpp) \tag{13}$$

수식 (13)에서, ||S||는 은닉된 메시지 S의 비트 수를 의미한다. H와 W는 이미지의 높이와 폭을 나타낸다.

(그림 2)는 제안한 방법의 실험 결과 이미지의 시각적인 질(Quality)을 보여준다. 모든 이미지는 약 0.67 bpp의 비율로 데이터를 저장할 수 있음을 보여주며, PSNR도 역시 $Hamming+1$ 보다 높은 50dB 이상임을 알 수 있다. (그림 2)에 대해서 원본 이미지와 구분이 어려움을 알 수 있다. 그러므로 본 논문에서 제안한 방법은 공격자에게 쉽게 노출되지 않는 방법임을 알 수 있다.

(표 1)은 매트릭스 인코딩인 "$Hamming+1$" 방법과 본 논문에서 제안한 방법 성능의 결과를 보여준다. 표 1로부터 "$Hamming+1$" 방법에서 이미지의 질이 약간 높은 결과를 보인 것도 있다. 예를 들어, "Baboon", "Goldhill", "Lena", 그리고 "Zelda" 등의 이미지에서는 제안한 방법 보다 이미지의 질이 약간 우세하다. 하지만, 평균적으로 제안한 방법이 약간 더 우위


(a)Baboon (51.8160dB)


(b)Barbara (51.7800dB)


(c) Boat (51.7431 dB)


(d) Goldhill (51.7741 dB)


(e) Jet(F16) (51.7515 dB)


(f) Lena (51.8252 dB)


(g) Pepper (51.7987 dB)


(h) Tiffany (51.7668 dB)


(i) Zelda (51.7695 dB)

그림 2. 실험 결과 스테고 이미지와 PSNR
Fig. 2. Stego Image and PSNR from Experimental Result.

표 1. $Hamming+1$ 방법과 본 논문에서 제안한 방법과의 실험 결과
Table 1. Comparison of Experimental Result between $Hamming+1$ and proposed scheme.

	Hamming + 1		제안한 방법	
Images	PSNR	p	PSNR	P
Baboon	53.71	0.499	51.8160	0.667
Barbara	48.60	0.499	51.7800	0.667
Boats	49.37	0.499	51.7431	0.667
Goldhill	53.73	0.499	51.7741	0.667
Jet(F16)	51.61	0.499	51.7515	0.667
Lena	52.43	0.499	51.8252	0.667
Pepper	47.26	0.499	51.7987	0.667
Tiffany	47.46	0.499	51.7668	0.667
Zelda	54.04	0.499	51.7695	0.667
Average	50.91	0.499	51.7805	0.667

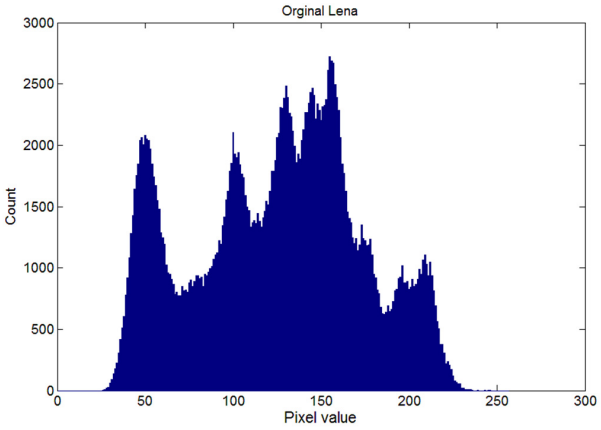


그림 3. 원본 Lena 이미지의 히스토그램.
Fig. 3. Histogram of Original “Lena” Image.

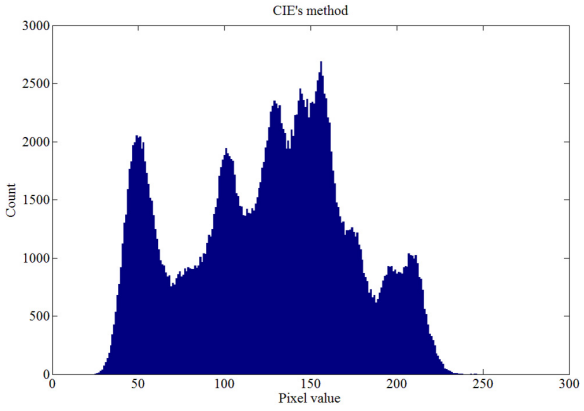


그림 4. 제안한 방법에 의한 “Lena” 스테고 이미지의 히스토그램.
Fig. 4. Histogram of “Lena” Stego Image from Proposed Scheme.

에 있음을 알 수 있다.

(그림 3)과 (그림 4)는 각각 원본 이미지와 제안한 방법에 의한 히스토그램을 나타낸 것이다. 원본이미지에 대해서 제안한 방법에 의한 스테고 이미지의 히스토그램이 큰 왜곡이 없음을 의미한다. 따라서, 다양한 스테고 탐지 분석 기법에 의해서라도 쉽게 탐지 되지 않음으로서 제안한 방법이 향후에 상업용으로 적용되기에 문제가 없음을 보였다.

V. 결 론

본 논문에서는 데이터 은닉 기법에 대해서 전반적으로 리뷰를 하였고, 이들 방법들 중에서 스테가노그래피의 특성을 갖고 있는 기법으로 "$Hamming+1$"의 방법이 성능 면에서 매우 우수한 방법임이 다른 연구자들에 의해서 평가되고 있다. 그러나, 이 방법은 데이터 은닉 비율이 약 0.5 bpp가 된다. 따라서, 이를 개선할 수 있는 방법을 본 논문에서 제안하였고, 성능 비교의 결과 약간의 성능을 개선하였음을 실험을 통해서 입증하였다. 향후, 보다 나은 스테가노그래픽 기법을 제안하여 다양한 분야에 제안한 방법이 활용되도록 할 예정이다.

REFERENCES

[1] Bender W., D. Gruhl, N. Mormoto, A.Lu. "Techniques for data hiding", IBM systems journal, vol. 35, pp.313-336 (1996).

[2] Hamming, R.W., "Error detecting and error correcting codes", Bell System Technical Journal, vol.29, no.2, pp.147-160, 1950.

[3] Zhang, W., Wang, S., and Zhang, X., "Improve embedding efficiency of covering codes for applications in steganography", IEEE Communications Letters, vol.11, no.8, pp.680-682, 2007.

[4] Chang, C. -C., T. D. Kieu, and Y.-C. Chou. "A high payload steganographic scheme based on (7, 4) Hamming code for digital images", International Symposium on Electronic Commerce and Security, Guangzhou, China, pp.16-21, 2002.

[5] Chao, R. M., H. C. Wu, C. -C. Lee, and Y.-P. Chu. "A novel image data hiding scheme with diamond encoding", EURASIP Journal on Information Security, vol. 2009, pp.1-9 (2009).

[6] Fridrich, J., M. Goljan, and R. Du. "Detecting LSB steganography in color, and gray-scale images", IEEE Transactions on Multimedia, vol. 8, pp. 22-28, 2001.

[7] Lee, C. F., C. C. Chang, and K. H. Wang. "Improvement of EMD embedding method for large payloads by pixel segmentation strategy", Image and Vision Computing, vol. 26, no. 12, pp. 1670-1676, 2008.

[8] Lin, P. L., C.-K. Hsieh, and P.-W. Huang, "A hierarchical digital watermarking method for image tamper detection and recovery", Pattern Recognition, vol. 38, no. 12, pp. 2519-2529, 2005.

[9] Mielikainen, J. "LSB matching revisited", IEEE Signal Processing Letters, vol. 13, no. 5, pp. 285-287, 2006.

[10] Provos, N. and P. Honeyman. "Hide and seek: An introduction to steganography", IEEE Security and Privacy, vol. 1, no. 3, pp. 32-44, 2003.

[11] Ni, Z., Y. Q. Shi, N. Ansari, W. Su, Q. Sun, and X. Lin. "Robust Lossless Image Data Hiding Designed for Semi-Fragile Image Authentication", IEEE transactions on circuits and systems for video technology, vol. 18, no.4, pp.497-509, 2008.

[12] Spaulding, J., H. Noda, M. N. Shirazi, and E. Kawaguchi. "BPCS steganography using EZW lossy compressed images", Pattern Recognition Letters, vol. 23, no. 13, pp. 1579-1587, 2002.

[13] University of Southern California. The USC-SIPI Image Database. Retrieved fro http://sipi.usc.edu/database/ on March 1, 2011.

[14] Wang, H. and S. Wang. "Cyber warfare: Steganography vs. steganalysis," Communications of the ACM, vol. 47, no.10, pp.76-82, 2004.

[15] Westfeld, A. "F5: A steganographic algorithm", Proceedings of the 4th International Workshop on Information Hiding 2001, Lecture Notes in Computer Science, vol. 2137, Pittsburgh, PA, USA, pp.289-302, 2004.

[16] Westfeld, A. "Attacks on steganographic systems", Proceedings of the 3rd Information Hiding Workshops, Dresden, Germany, September 28-October 1, pp.61-75, 1999.

[17] Yu, Y. H., C. C. Chang, and Y. C. Hu. "Hiding secret data in images via predictive coding", Pattern Recognition, vol.38, no.5, pp.691-705, 2005.

[18] Zhang, X. and S. Wang. “Efficient steganographic embedding by exploiting modification direction”, IEEE Communications Letters, vol.10, no.11, pp.781-783, 2006.
[19] Chang, C.C., T.S. Chen, and L.Z. Chung. “A steganographic method based upon JPEG and quantization table modification”, Information Sciences-Informatics and Computer Science, vol.141, no.1-2, pp.123-138, 2002.
[20] Lin, W.H., Wang, Y.R., Horng, S.J., “A wavelet-tree-based watermarking method using distance vector of binary cluster”, Expert Systems with Applications, vol.36, no.6, pp.9869-9878, 2009.
[21] 김천식, 윤은준, 조민호, 홍유식, “의료영상을 위한 복원 가능한 정보 은닉 및 메시지 인증”, 대한전자공학회 논문지 CI편, 제 47권, 1호, pp.65-72, 2010.

저 자 소 개

김 천 식(정회원)
1997년 한국외국어대학교 컴퓨터 및 정보통신공학과 (공학석사)
2003년 한국외국어대학교 컴퓨터 및 정보통신공학과 (공학박사)
2010년~2012년 세종대학교 교수
2013년～현재 안양대학교 교수
2007년～2009년 대한전자공학회 컴퓨터소사이어티 멀티미디어 분과위원장
2006년～현재 인터넷 정보학회 학회편집위원
2006년～현재 대한교통학회 정회원
2007년～2008년 인터넷방송통신tv학회 상임이사
2005년～현재 한국데이터베이스학회 정회원
2008년～2009 ICHIT committee member
2009년 ACIIDS 2010 committee member
2009년 대한전자공학회 JUCT 영문저널 위원
2009년 2009 ICACT committee member
2012년 TACT 영문 저널 - 위원
2012년 UMAS 워크샵 프로그램 의장
2013년 GPC 2013 프로그램 의장
<주관심분야: 데이터베이스, 데이터마이닝, Steganography, 영상처리, e-Learning>