

이진 영상 워터마킹 알고리즘에 관한 연구

이 덕*, 최종욱**

*상명대학교 컴퓨터과학과

**상명대학교 소프트웨어학부

e-mail : lide@sangmyung.ac.kr

A Study on Watermarking Algorithm for Binary Images

Li De*, Jonguk Choi**

*Dept. of Computer Science, Sangmyung University

**Dept. of Software, Sangmyung University (MarkAny Inc)

요 약

전자정부, 전자상거래의 활성화로 디지털 문서가 빠른 속도로 유통되고 있기에 이에 대한 효과적인 보호대책이 필요한 실정이다. 본 논문에서는 Window Pattern 을 이용하여 Binary Image 에 저작권 정보를 삽입하는 방안을 제안한다. 삽입대상 Window Pattern 을 결정하고 이러한 Pattern 으로 원본 영상을 Scan 하면서 픽셀 값에 변화를 주게 된다. 이렇게 되어 하나의 Pattern 에 1 bit 의 정보의 삽입이 가능하게 되고 추출 시 원본을 필요로 하지 않으며 실용성이 높고 적용분야도 넓다.

1. 서론

전자정부, 전자상거래, 디지털 도서관의 구현 및 활성화로 문서영상을 중심으로 한 이진 영상이 방대하게 유통되고 있어 이에 대한 저작권 보호의 필요성이 고조되고 있는 실정이다. 이진 영상은 하나의 픽셀을 1 bit 로 표현하기 때문에 데이터 size 가 작고 삽입공간이 현저하게 적어 지게 된다. 또한 적합하지 않은 픽셀 값 변경으로 인한 위상 변화는 쉽게 시각적으로 감지될 수 있다.

디지털 문서 이미지에 정보를 삽입하는 기존의 방법 들로는 아래와 같은 몇 가지가 있다. Line or word shifting coding 방법[1]은 텍스트 이미지의 기존의 행간과 자간의 거리, 문자의 회전각 등을 적당히 조절하여 bit stream 을 삽입하게 되는데 정보가 삽입된 텍스트 이미지에서 행간, 자간 거래와 회전각을 변경시켰을 경우에는 정확하게 정보를 추출 할 수가 없게 된다.

문자의 형태를 이용하여 정보를 삽입[2][3]하는 방법은 특정된 언어 또는 글꼴에 한해서 적용 가능한 방법으로서 문자의 구조특성을 이용하여 시작적으로 감지하기가 어렵게 문자형태를 변경시킴으로써 정보

를 삽입하게 된다. 또 Morphological 신호처리를 이용하여 문서 이미지의 문자나 도형에 변화를 주어 정보를 삽입하게 되는데 이는 Noise 나 Offline Detection 에 강인함을 보이지만 제한된 문자나 도형에 한해서만 적용 가능하여 적용범위가 극히 제한적이다.

또한 Signature Seal 에 Text 코드를 삽입하는 방법[4]도 있는데 이는 문서내의 Text 를 코드화하여 코드화 된 비트 stream 을 Seal 이미지에 삽입함으로써 디지털 문서에 포함되어 있는 Seal 이미지를 이용하여 문서내용을 복원함으로써 문서내용에 대한 위, 변조를 막아 주게 된다. 하지만 Seal 이미지가 Noise 나 압축에 Fragile 하여 문서 내용을 복구해내지 못하는 경우가 있고, 또 공문서나 증명서류 등 제한된 분야에만 적용이 가능하고 위조검출에는 사실상 적용이 불가능하다.

이 외에도 자연어 처리를 이용하여 디지털 문서에 정보를 삽입[5]하는 방식이 있는데 문서중의 표현을 의미가 변하지 않게 다른 표현으로 치환하여 정보를 삽입하는 시스템이다. 이 방법은 디지털 문서를 이미 지화 하지 않고도 정보를 삽입할 수 있는 장점이 있는 반면, 원본 문서가 있어야 추출가능하며 또 실질적

으로는 문서의 내용이 바뀌게 되어 상품권, 공문서, 증명서류 등 분야에는 적용하기가 어렵다.

본 논문에서는 기존의 방법상의 한계점을 극복하고자 문서 이미지 중 Text 의 Edge 부분의 픽셀 값을 적당한 값으로 치환하여 변경시킴으로써 정보를 삽입하게 된다.

우선 수개의 삽입대상 Windows Pattern 을 선택하고 bit 1 과 bit 0 을 나타내는 Pattern 을 결정한다. 다음 문서 이미지 중 Pattern 과 일치한 픽셀 영역에 한해서는 삽입정보에 대응되는 특정된 Pattern 으로 변경시켜 주게 된다. 하여 하나의 Pattern 에 1bit 의 정보가 삽입되게 되며 이와 같이 전체 문서 이미지를 특정된 방향으로 Scan 하여 정보를 반복적으로 삽입하게 된다. 추출 시에는 bit 1 과 bit 0 의 Pattern 으로 Embedded 이미지를 같은 방식으로 scan 하여 일치한 Pattern 찾아냄으로써 삽입한 정보를 추출하게 된다.

2. 정보 삽입 알고리즘

2.1 픽셀 공간 이용형 워터마크

Window Pattern 은 3*3 또는 4*4 의 블록을 선택할 수 있다. [그림 1]은 3*3 블록을 보여주는데, 어두운 색으로 표시된 픽셀은 bit 1 을 표시하고 흰색 부분은 0 을 표시하게 된다. 정보 삽입 연산의 필요로 3*3 bit 를 $P_i(k)$ 의 bit stream 으로 표현하게 된다. $P_0(k)$ 는 픽셀 평면의 중심 픽셀 값이 되고 왼쪽 우측으로부터 시계 바늘 방향으로 번호를 부여한다. 이렇게 되어 블록 패턴은 순서 패턴으로 변화하게 된다. 이러한 정보의 표현방법은 이해를 쉽게 해주게 된다. S_{ij} 는 삽입정보로서 저작권자의 고유의 저작권 데이터, 싸인 또는 로고 이미지 일 수 있다.

대상 문서 이미지에 영향을 줄이고 은닉 효과를 높이기 위하여 3*3 픽셀 중에서 가장 적은 영향을 끼칠 수 있는 픽셀 값을 변화시킨다. 즉 픽셀 값을 0->1로, 1->0로 반전시킨다. 픽셀 값 변경이 이미지에 미세한 변화를 주기 위하여 Edge 부분의 픽셀 값만 변경 시키는데 $P_0(k)$, $P_3(k)$, $P_7(k)$ 등 픽셀이 이에 해당된다.

2.2 삽입대상 Window Pattern 의 결정

삽입 대상 Window Pattern 선택 시 다양한 문서 이미지에 대한 통계적 분석을 통해 출현빈도가 많고 픽셀 값에 대한 치환이 시각적인 효과에 영향을 적은 Pattern 을 선택하였다. 이러한 Pattern 은 [표 1]에서와 같이 서로 다른 유형의 세가지 그룹으로 구분되었으며 전체 12 개의 삽입대상 Pattern 과 삽입후의 Pattern 에 [그림 1]에서의 상대 위치를 적용하였다. 표에서 보다시피 삽입 bit 1 을 표시하는 Pattern 의 종류는 4 가지이고, 삽입 bit 0 을 표시하는 패턴은 8 가지가 된다. 이러한 패턴의 선택은 사용언어, 글꼴,

글꼴크기에 따라 통계적인 방법으로 처리하여 선택된다.

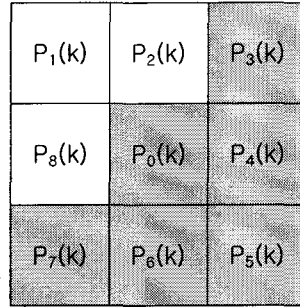


그림 1 Window Pattern 의 구성

G	P_i	S	P_1	P_0
G1	[1 0 1 1 1 1 1 1 1]	1	[1 0 0 1 1 1 1 1 1]	
		0	[1 0 1 1 1 1 1 1 1]	
	[1 1 1 0 1 1 1 1 1]	1	[1 1 0 0 1 1 1 1 1]	
		0	[1 1 1 0 1 1 1 1 1]	
	[1 1 1 1 1 0 1 1 1]	1	[1 1 1 1 1 0 0 1 1]	
		0	[1 1 1 1 1 0 1 1 1]	
G2	[1 1 1 1 1 1 1 1 0 1]	1	[1 1 1 1 1 1 1 0 0 1]	
		0	[1 1 1 1 1 1 1 1 0 1]	
	[1 1 0 0 1 1 1 1 1 1]	1	[1 1 0 0 1 1 1 1 1 1]	
		0	[1 1 1 0 1 1 1 1 1 1]	
	[1 0 0 1 1 1 1 1 1 1]	1	[1 0 0 1 1 1 1 1 1 1]	
		0	[1 0 1 1 1 1 1 1 1 1]	
G3	[1 1 1 1 1 1 1 0 0 1]	1	[1 1 1 1 1 1 1 0 0 1]	
		0	[1 1 1 1 1 1 1 1 0 1]	
	[1 1 1 1 1 1 0 0 1 1]	1	[1 1 1 1 1 1 0 0 1 1]	
		0	[1 1 1 1 1 1 0 1 1 1]	
	[1 1 1 1 0 0 1 1 1 1]	1	[1 1 1 1 0 0 1 1 1 1]	
		0	[1 1 1 0 1 1 1 1 1 1]	
	[0 1 1 1 1 1 1 1 1 0]	1	[0 1 1 1 1 1 1 1 1 0]	
		0	[0 1 1 1 1 1 1 1 1 1]	

표 1 삽입 대상 Pattern 과 삽입 Pattern

2.3 픽셀 치환에 따른 정보의 삽입과정

Original Document Image 에 정보를 삽입하기 위하여 [표 1]에서 선택한 총 12 개의 Pattern 으로 Image 를 X 또는 Y 축 방향으로 Scan 하게 된다. [그림 3]에서 보다시피 Scan 과정 중 선별된 임의의 하나의 Pattern 과 일치하지 않으면 삽입대상에서 제외되며, 만일 일치하면 정보의 삽입이 이루어 지게 된다. 삽입 정보가 1 일 경우 선택된 Pattern 을 표에 나

와 있는 P_1 값으로 치환하고, 삽입정보가 0 일 경우 P_0 을 치환함으로써 정보를 삽입하게 된다. 하나의 Pattern 에 대한 정보 삽입 끝나면 Scan 카운터를 증가하고 계속하여 다음 선택된 Pattern 에 대하여 반복적인 과정을 적용시킨다. 전부의 삽입정보에 대한 삽입 끝나면 삽입 과정을 종료 시킨다.

그림 2 에서는 표에 나와있는 패턴중의 하나를 선택하여 정보의 삽입을 보여준다. 이 패턴은 픽셀 하나를 변화시켜 bit 1 을 삽입하고 대상 패턴 원형 그대로 유지하는 방식으로 bit 0 을 삽입하게 된다. 이렇게 되어 보다 적은 픽셀의 변화로 정보를 삽입하게 되며 삽입 후의 텍스트 이미지의 화질도 일정한 정도 유지할 수 있다.

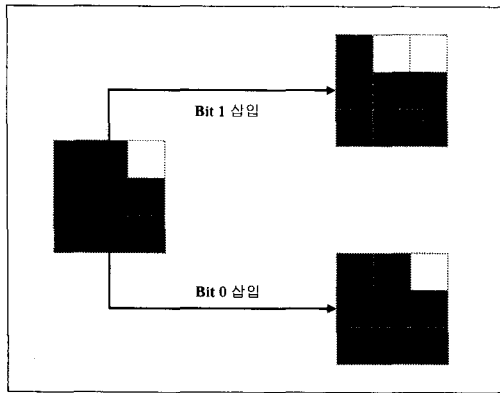


그림 2 픽셀 변화에 따른 정보의 삽입

표에서 보여준 삽입 Pattern 들은 픽셀 값을 임의로 변경시킨 것이 아니고 시각적인 은닉효과를 높이기 위한 방법을 사용하였다. Cropping 등 변화에 강인하기 위해서는 하나의 삽입정보를 충분히 수용할 수 있는 직사각형 영역을 설정하고 이 영역에 Parity bit 를 추가하여 반복적으로 정보를 삽입할 수 있다. 직사각형 영역은 삽입정보의 양에 따라 변할 수 있다. 두 가지 Scan 방향과 P_1 , P_0 값들의 적당한 연산 조합은 추출시의 키로서 정보의 삽입위치와 Parity bit 의 위치지정에 사용될 수 있다.

$$dx=010; \quad dy=101$$

$$Ix=dx \oplus [L_3(P_{1S2})] \quad (1)$$

$$Iy=dy \oplus [L_3(P_{0S4})] \quad (2)$$

$$P_i = P_{1S2} \parallel P_{0S4} \quad (3)$$

I_x 와 I_y 는 각각 scan 방향이 X 방향과 Y 방향일 때의 삽입위치를 나타내는데 P_{1S2} 는 1bit 를 나타내는 모든 패턴들을 XOR 한 결과값을 좌측 2bit Shift 한 값이며 $L_3(P_{1S2})$ 은 해당 값의 하위 3bit 를 취한 것이다. I_y 도 마찬가지로 좌측 4bit Shift 한 것 만 다를

뿐이다. 하여 이렇게 계산된 위치로부터 정보를 삽입하게 된다. P_1 는 P_{1S2} 와 P_{0S4} 를 연결한 값으로 Parity bit 를 나타낸다. Parity bit 를 체크 해냄으로써 정보의 삽입 영역인 직사각형 영역을 추출해낼 수 있게 된다.

대상 이미지의 size 가 $3n$ 과 일치하지 않으면 나머지 부분에는 정보를 삽입하지 않는다. 이미지 size 를 늘이고 확장된 부분엔 zero padding 해주어 정보를 삽입 할 수 도 있다. 하지만 이렇게 되면 확장된 부분에 정보가 삽입이 될 경우, 해당 정보를 추출해 내지 못할 수도 있다.

$4*4$ Pattern 을 이용하면 문서 이미지 중의 보다 많은 Edge 부분의 픽셀을 포함할 수가 있어 어떤 픽셀 값들을 변경시킬 것인지를 효과적으로 분석하고 결정할 수가 있어, 보다 적합한 패턴을 얻을 수 있다. 하여 삽입 이미지의 화질을 향상시킬 수 있으나, 패턴 size 의 증가로 정보의 삽입공간이 줄어들게 된다.

워터마크의 추출은 P_1 , P_0 패턴으로 Embedded Image 를 삽입 과정에서와 동일한 방식으로 Scan 하여, 원본 없이 정확한 삽입정보를 추출해내게 된다. 추출 시 우선 Parity bit 를 계산하여 직사각형 영역을 추출하고, 하나의 직사각형 영역에서의 정보의 삽입 시작위치를 결정한 후 추출하게 된다. 삽입 시와 마찬가지로 삽입위치 및 Parity bit 는 식 (1), (2), (3)으로부터 계산해내게 된다.

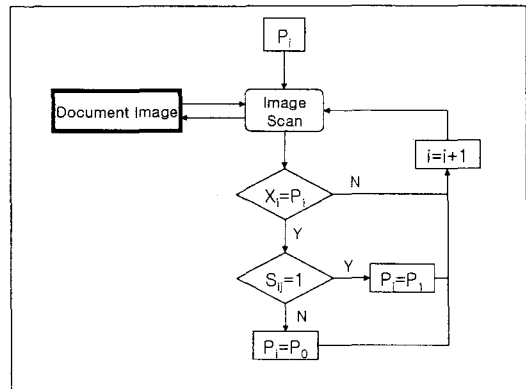


그림 3 삽입과정 흐름도

3. 실험결과

본 실험에서는 대상 문서는 한글 문서로부터 추출해낸 $256*256$ bmp Image 를 사용하였고 88bit 의 Random Sequence 를 삽입하였다. 삽입할 수 있는 정보의 양은 대상 이미지와 선택한 Windows Pattern 에 의해 결정된다. 그림 4 과 그림 5 는 삽입전후의 Image 를 보여준다.

CV(Correlation Value)는 0.9993 을 얻었고, SNR 값은 31.35dB 로 시각상의 차이를 느끼지 못할 만큼 좋은 결과를 얻을 수가 있었다. Embedded

Image 를 JPEG 으로 압축한 Image 를 대상으로 추출 하였을 경우에도 동일한 결과를 얻을 수가 있어 압축 에도 강인함을 보였다. 삽입대상이 Binary 이미지가 므로 JPEG 으로 압축된 이미지에 직접 워터마크를 삽 입하고 추출할 수 도 있다.

참고문헌

[1] S. Low, N. Maxemchuck, J. Brassil and L. O' Gorman, " Document Marking and Identification using Both Line and Word Shifting" , Proceeding of INFOCOM95, Boston, U.S.A., April 1995.

[2] TSUJIAI, UETSUJI, " Digital Watermark in Lettering Images by Using Character Shape" , IEICE Trans, Vol. J82-D-II No.11 pp.2175-2177, Nov 1999.

[3] OKAMOTO, MIYAZAKI, " A Digital Watermark Technique Using Morphological Signal Processing" , IEICE Trans, Vol. J84-A No.8 pp.1037-1044, Aug 2001.

[4] IWAKIRI, MURAKAMI, PIYAPISUIT, NAKAMURA, MATSUI, " Signature Seal with Recoverable Function of Text for Electronic Documents" , SCIS2000, D54, Okinawa, Japan, Jan 2000.

[5] SAMPEI, NAKAGAWA, MATSUMOTO, MURASE, " Meaning Preserving Information Hiding" SCIS2000, D50, Okinawa, Japan, Jan 2000.

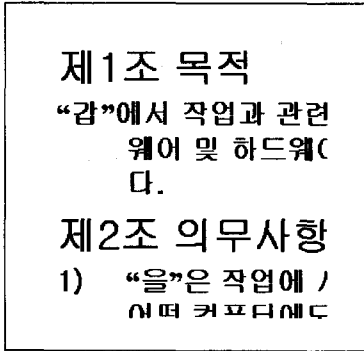


그림 4 Original Image

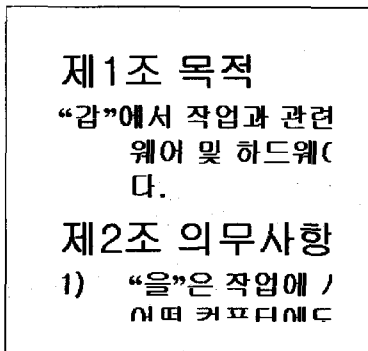


그림 5 Embedded Image

4. 결 론

본 논문에서는 최근 온라인 상에서 급속히 유통되고 있는 Binary Text Image 를 대상으로 저작권 정보를 삽입할 수 있는 효과적인 방안을 제안하였다. 원본 없이 추출 가능하며 Cropping 과 압축 등 변형에 강인하다. 이 방법은 Binary 문서 이미지 뿐만 아니라 도형을 포함한 기타의 다양한 Binary Image 에도 적용 가능하다.

정보 삽입 공간을 늘이고 화질 향상을 위해 통계적인 방법을 통하여 Window Pattern 선별에 대한 보다 체계적이고 확실한 방안이 필요하며, 기타 다양한 기하학적 변형에 대한 강인성 여부도 검토해야 할 것이다. 또한 본 알고리즘을 Offline Detection 에 적용 가능한지도 검토해야 할 사항이다.