論文2003-40SP-5-9

MPEG-2 비트열에서의 인증 및 조작위치 검출을 위한 디지털 워터마킹 기법

(Digital watermarking algorithm for authentication and detection of manipulated positions in MPEG-2 bit-stream)

朴 宰 延 *, 林 載 爀 **, 元 致 善 **

(Jae Yeon Park, Jae Hyuck Li, and Chee Sun Won)

요 약

디지털 워터마킹은 소유권자의 정보나 특정 코드, 혹은 패턴을 디지털화 되어 있는 정지영상, 동영상, 음성 데이터 등에 인간의 시각이나 청각으로는 감지 할 수 없도록 삽입하는 기술이다. 디지털 워터마킹은 크게 강인성 워터마킹과 연성 워터마킹으로 분류 될 수 있다. 강인성 워터마킹은 이미지나 영상에 대한 소유권자의 소유권을 보호하는 것이 주된 목적이며, 연성 워터마킹은 디지털 콘텐츠를 불법적인 변형으로부터 보호하는 것이 목적이다. 특히 준 연성(semi-fragile) 워터마킹은 잡음 첨가 혹은 압축과 같은 데이터의 전송 과정에서 자주 발생하는 비의도적 변형에 대해서는 삽입된 워터마크가 제거되지 않아야 하지만 의도적 변형에 대해서는 삽입된 워터마크가 훼손되어 검출되지 않아야 한다. 본 논문에서는 이러한 요구사항들을 만족시키는 MPEG-2 비트열에서의 인증 및 조작위치 검출을 위한 준 연성 워터마킹 알고리즘을 제안한다. 제안된 알고리즘은 두 가지의 워터마크를 양자화 된 DCT 계수에 삽입한다. 따라서 압축된 비트스트림에 직접적으로 적용할 수 있다. 첫번째 워터마크는 해쉬 함수를 사용하여 비디오 데이터에 대한 인증을 한다. 두번째 삽입되는 워터마크는 양자화 된 DCT DC 계수를 이용하며 조작위치를 검출하는데 사용한다. 제안된 알고리즘은 비트스트림 영역에서의 트랜스 코딩에 의한 비디오 데이터의 변형과 의도적인 변형을 구별 할 수 있으며 만약 비디오 데이터에 의도적인 변형이 있었을 경우 인트라 프레임에 대해서는 변형된 위치를 블록 단위로 검출 가능하다. 또한 제안된 알고리즘은 가역적 특성을 갖고 있기 때문에 비디오 데이터에 변형이 없었을 경우에는 원래의 데이터를 복원 할 수 있다.

Abstract

Digital watermarking is the technique that embeds invisible signalsincluding owner identification information, specific code, or pattern into multimedia data such as image, video and audio. Watermarking techniques can be classified into two groups; robust watermarking and fragile(semi-fragile) watermarking. The main purpose of the robust watermarking is the protection of copyright, whereas fragile(semi-fragile) watermarking prevents image or video data from illegal modifications. To achieve this goal watermark should survive from unintentional modifications such as random noise or compression, but it should be fragile for malicious manipulations. In this paper, an invertible semi-fragile watermarkingalgorithm for authentication and detection of manipulated location in MPEG-2 bit-stream is proposed. The proposed algorithm embeds two kinds of watermarks, which are embedded into quantized DCT coefficients. So it can be applied directly to the compressed bit-stream. The first watermark is used for authentication of video data. The second one is used for detection of malicious manipulations. It can distinguish transcodingin bit-stream domain from malicious manipulation and detect the block-wise locations of manipulations in video data. Also, since the proposed algorithm has an invertible property, recovering original video data is possible if the watermarked video is authentic.

Keyword : watermarking, authentication, semi-fragile watermarking, invertible, transcoding

* 正會員, 서울通信技術(株)
(SEOUL COMMTECH CO., LTD.)
** 正會員, 東國大學校 電子工學科
(Dept. of Electronic Engineering, Dongguk University)

接受日字:2003年2月24日, 수정완료일:2003年9月16日

Ⅰ. 서 론

디지털 워터마킹[1-3]은 삽입되는 워터마크의 강인성에 따라서 크게 두 가지로 분류된다. 그 첫 번째는 강인성 워터마킹[1-5]으로써 저작권 보호, 불법 복제 추적 등의 용도에 사용되며 의도적, 비의도적 공격에 의해서 삽입된 워터마크가 제거되지 않는 강인한 워터마크를 삽입한다. 두 번째 분류로써 연성 워터마킹[6-7]이 있는데 이는 공격자의 의도적 공격에 강인하게 설계되어지는 강인성 워터마킹과는 반대로 삽입되어진 워터마크가 영상의 작은 변형에 의해서도 쉽게 소멸되어 워터마크가 검출되지 않는 상태를 유도하게 된다. 그러므로 워터마크가 검출되지 않거나, 변조된 부분에 대하여 워터마크가 소멸되어 변조된 위치까지 추적 할 수 있게 된다. 또한 이미지화 된 문서에 대하여 더 이상의 가공이 불가능하게 원천적으로 봉쇄하여 안전하게 공유할 수 있게 한다. 이러한 연성 워터마킹은 다시 준 연성 워터마킹[8-10]으로 구분할 수 있다. 준 연성 워터마크킹은 삽입된 워터마크가 사용 목적에 따라 일정한 공격에 대해서는 강인성을 갖으나 그 이외의 공격에 대해서는 쉽게 소멸되는 특성을 갖고 있다. 이러한 인증을 위한 워터마킹은 보다 효율적이고 확실한 인증을 위하여 삽입되는 워터마크를 원본 데이터로부터 생성하게 되며 일반적으로 해쉬 함수[11]를 사용한다. 보다 정확하고 기능적인 인증을 위한 워터마킹 기법이 되기 위해서는 몇 가지 요구사항을 만족해야 한다. 우선 원본 데이터의 변형 유/무와 함께 변형 위치 검출이 가능해야 하며 만일 원본 데이터에 변형이 없음이 확인되었다면 삽입된 워터마크를 제거하여 원본데이터를 얻을 수 있어야 한다. 또한 데이터의 전송 과정에서 흔히 발생하는 비의도적 변형과 공격자에 의한 의도적 변형을 구분 할 수 있어야 한다. 그러나 현재까지 비디오 데이터에 대한 인증을 위해 제시된 워터마킹 방법들은 위의 요구조건을 만족하지 못하고 있다. Yin[9]이 제시한 방법에서는 GOP내의 I 프레임에 연성 워터마크와 강인성 워터마크를 각기 삽입하도록 되어 있다. 연성 워터마크는 크게 GOP 워터마크와 I 프레임 워터마크로 구성되어 있다. GOP 워터마크는 비디오 데이터의 무결성을 확인하기 위해서 사용된다. GOP 워터마크의 생성은 GOP내 각 프레임의 프레임 데이터에 대한 해쉬 값, 각 프레임 데이터에 대한 해쉬 값의 m 비트를 연결한 데이터, 그리고 컨트롤 데이터를 사용하며 이들 세가지 데이터를 개인 키 암호화 과정을 통해 생성한다. I 프레임 워터마크는 I 프레임 데이터의 해쉬 값과 블록 데이터를 개인 키 암호화를 통해 생성하게 된다. 이때 컨트롤 데이터로는 GOP 인덱스, GOP내의 프레임 수, 타임 코드 등이 사용된다. 강인성 워터마크는 워터마크 검출 과정에서 연성워터마크가 파괴되었음이 확인 되었을 때, 비디오 데이터에 가해진 변형의 종류를 판별하는데 사용한다. 강인성 워터마크를 생성하기 위해, 트랜스 코딩에 의해 변하지 않는 특성을 갖는 I 프레임 내의 양자화된 DCT DC 계수를 강인성 워터마크 생성을 위한 피쳐(feature)로 선택하고 이를 랜덤 셔플링 후, 해쉬 함수를 사용하여 워터마크를 생성한다. 워터마크의 삽입을 위해 연성 워터마크에 대해서는 양자화된 DCT DC 계수의 LSB를 이용하였으며, 강인성 워터마크를 삽입하기 위해서 블럭 DCT based spread spectrum 을 이용하였다. 이 방법은 프레임 드롭과 같은 비디오 데이터에 대한 시간 축상의 변형을 검출 할 수 있고, 인위적인 변형과 트랜스 코딩에 의한 비디오 데이터 변형을 구별 할 수 있다. 그러나 각각의 워터마크가 I 프레임에 삽입 되도록 되어있기 때문에 워터마크가 삽입된 비디오 데이터에서 모든 I 프레임을 드롭하고 재 인코딩 할 경우 삽입된 워터마크를 전혀 검출 할 수 없게 되는 문제점을 갖고 있다. Du와 Fridrich[10]가 제시한 방법에서 비디오 데이터의 인증은 프레임과 GOP 단위로 이루어진다. 프레임 단위의 인증을 위해서 각 프레임의 양자화된 DCT 계수 중 0이 아닌 계수 값들에 대한 해쉬 값을 계산한다. 계산된 해쉬 값과 프레임 인덱스를 사용하여 삽입 될 워터마크를 생성하고 인트라, 넌 인트라(non-intra) 매크로 블록의 색차 성분 블록의 AC 계수를 선택하여 LSB에 삽입하게 된다. 이때 인버티블 한 특성을 얻기 위해 선택된 계수에 2를 곱하여 LSB가 0이 되도록 한 후 워터마크를 삽입하며, 해당 계수의 양자화 factor는 2로 나누어 준다. GOP 단위의 인증을 위해서 각 B 프레임과 참조 프레임의 양자화된 DCT 계수 중 0이 아닌 계수를 입력으로 사용하여 해쉬 값을 계산한다. 계산된 해쉬 값과 GOP의 인덱스를 이용하여 워터마크를 생성하고 각 B 프레임의 넌 인트라 매크로 블록의 색차 성분 블록에 프레임에 의한 인증 방법과 동일한 방법으로 삽입하게 된다. Du와 Fridrich가 제시한 워터마킹 방법은 해쉬 함수와 프레임 인덱스를 이용하여 비디오 데이터의 무결성과 프레

임 드롭 유무는 확인 가능하지만 인위적인 조작과 트랜스 코딩을 구분 할 수 없다. 따라서 본 논문에서는 위의 논문들이만족하지 못하는 요구 조건을 만족하는 MPEG-2 비트열에서의 인증 및 조작위치 검출을 위한 준 연성 워터마킹 알고리즘을 제안한다. 제안된 워터마킹 기법은 모든 프레임에 대해 비디오 데이터의 무결성에 대한 인증이 가능하며 만일 원본 비디오 데이터가 조작 되었다면 조작위치 검출이 가능하다. 또한 비디오 데이터의 전송 과정에서 흔히발생하는 비트스트림 영역에서의 트랜스 코딩[12-13]과 인위적인 조작의 구분이 가능하며 비디오 데이터에 변형이 없었을 경우 삽입된 워터마크의 검출 과정 종료 후 원본 데이터와 동일한 비디오 데이터를 얻을 수 있는 인버티블 특성을 갖고 있다.

본 논문은 우선 Ⅱ장에서 제안된 알고리즘의 워터마크 삽입과정을 설명하고 Ⅲ장에서는 워터마크 검출 과정을 설명한다. Ⅳ장에서는 실험 결과를 보여주고 Ⅴ장에서 결론 및 향후 과제를 제시한다.

Ⅱ. 워터마크 삽입

제안된 알고리즘에서는 <그림 1>에서의 점선으로 표시된 두 개의 블록과 같이 두 종류의 워터마크를 삽입한다. 첫 번째 워터마크는 비디오 데이터의 인증을 위하여 사용되며 <그림 1>의 좌측에 있는 점선으로 표시된 블록에서와 같이 해쉬 함수의 출력 값과 소유권자의 정보를 사용하여 워터마크를 생성하고 무손실 압축 기법을 이용하여 각 프레임내의 8×8 DCT 블록의 양자화 된 AC 계수에 삽입하게 된다. 두 번째 워터마크는 비디오 데이터에 대한 인위적인 조작위치 검출을 위하여 사용되는데 원본 비디오 데이터의 I 프레임의 양자화 된 DCT DC 계수와 비밀 키를 사용하여 발생시킨 랜덤 시퀀스를 이용하여 삽입하게 된다.

1. 인증을 위한 워터마크의 삽입

인증을 위해 삽입되는 워터마크는 각 프레임의 양자화 된 DCT 계수를 입력으로 하여 얻은 128 비트의 해쉬 출력 값과 소유권자의 정보를 결합하여 생성하며, 무손실 압축 기법을 사용하여 양자화 된 각 8×8 DCT 블록내의 고주파 성분에 위치한AC 계수의 LSB에 삽입하게 된다. <그림 1>은 워터마크 삽입과정에 대한 전체 흐름도를 나타내고 있다.

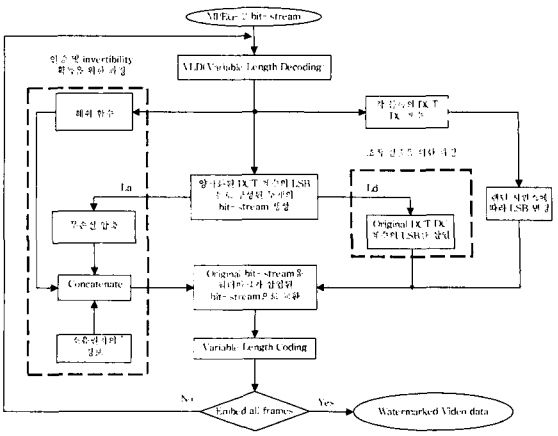


그림 1. 워터마크 삽입 전체 흐름도
Fig. 1. Flowchart of the proposed algorithm.

<그림 1>에서와 같이 입력으로 들어온 MPEG-2 비트스트림에 대해서 각 프레임 단위로 VLD(Variable Length Decoding) 과정을 거쳐 양자화 된 계수 값으로 구성된 프레임을 생성하고 생성된 프레임을 서로 겹치지 않도록8×8 블록으로 분할한다. 양자화 된 DCT 계수들을 $QC_k(i,j)$라 하면 다음과 같은 수식으로 표현 할 수 있다.

$$QC_k(i,j) = \lfloor C_k(i,j) / Q(i,j) \rfloor \tag{1}$$

여기서 $QC_k(i,j)$, $C_k(i,j)$, $Q(i,j)$는 각각 k번째 블록에서의 $(i,j)\ 0 \le i,j \le 7$ 위치의 양자화 된 계수, DCT 계수, 양자화 스텝 사이즈를 나타낸다. 인증을 위해 삽입될 워터마크를 생성하기 위해서 <그림 1>의 좌측 점선으로 표시된 블록과 같이 프레임내의 양자화된 DCT DC 계수 값들을 해쉬 함수의 입력으로 사용하여 128비트 길이를 갖는 해쉬 함수 값을 구한다. 본 논문에서는 해쉬 함수로 MD5[11]를 사용하였으며 해쉬 함수 f_H의 출력을 구하는 관계식은 다음과 같이 표현된다.

$$H_{[0:127]} = f_H\left(\sum_{k=1}^{B}\sum_{j=0}^{7}\sum_{i=0}^{7} QC_k(i,j)\right) \tag{2}$$

여기서 $H_{[0:127]}$은 해쉬 함수에 의해 발생된 128비트 길이의 해쉬 값을 나타내며 B는 한 프레임내의 8×8 DCT 블록의 전체 개수를 나타내고 있다. 해쉬 함수의 출력을 구한 후에 프레임내의 각 8×8 DCT 블록에서 고주파 성분에 위치한 AC 계수를 선택한다. AC 계수의 선택은 고주파 성분에 위치한 계수를 임의로 선택

가능하며, 고주파 성분의 계수를 선택하는 이유는 선택된 계수의 LSB 변경에 의한 화질 열화를 최소로 하고, LSB 비트스트림의 무손실 압축 시에 압축 효율을 증대하기 위해서 이다. 한편 P, B 프레임에서는 스킵(skip)되는 매크로 블럭이 존재하며 스킵 되는 매크로 블록은 워터마크 삽입에 이용 할 수 없다. 따라서 매크로 블록의 스킵 여부를 체크하여 스킵 되지 않는 매크로 블록의 AC 계수만을 선택하여야 한다. 본 논문에서는 스킵 되지 않으면서 밝기 성분을 나타내는 88 DCT 블록의 고주파 성분에 위치한 AC 계수 중 56번째 계수를 선택하였다. 선택된 각 계수에서 LSB를 추출하고 추출된 LSB로 구성된 비트스트림을 만들어 낸다. 이렇게 생성한 비트스트림을 인증을 위한 LSB 비트스트림이라 하고, L_a로 나타내면 수식 (3)과 같이 표현된다.

$$L_a = \{L_k \in \{0,1\};\quad 1 \le k \le B\} \tag{3}$$

여기서 L_k는 k번째 블록에서 선택된 계수의 LSB를 나타낸다. 식 (3)과 같이 표현된 비트스트림을 생성하는 과정을 <그림 2>에서 보여주고 있다. <그림 2>가 나타내는 것과같이 인증을 위한 LSB 비트스트림은 8×8 DCT 블록에서, 고주파 영역에 위치한 AC 계수들의 LSB로 구성된 비트스트림이므로 대부분의 값들이 0을 갖게 된다. 따라서 무손실 압축 기법을 적용하기 용이하며 압축 기법에 따라 다소 차이가 있겠지만, 일반적인 경우 비트스트림의 길이가 1/2 이하로 감소하게 된다. 따라서 L_a 대한 무손실 압축을 통해서 감소한 공간에, 해쉬 함수의 출력 값과 소유권자의 정보를 결합

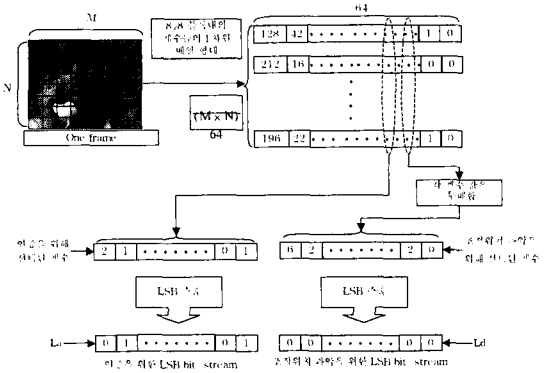


그림 2. 인증 및 조작위치 검출에 사용되는 LSB 비트스트림의 생성

Fig. 2. Bit-stream generation for authentication and detection of manipulated position.

하여 워터마크 비트스트림을 생성 할 수 있다.

본 논문의 실험에서도 L_a의 전체 bit 중 약 95% 이상이 0의 값을 갖고 있었으며 무손실 압축 후 1/3 이하로 길이가 감소하였다. 본 논문의 실험에서는 무손실 압축을 위해 간단한 런-레벨(Run-level) 4비트 픽스트(Fixed) 코딩 기법을 응용하여 사용하였다. 즉 0을 런으로 하고 1의 값을 레벨로 하여 코드를 할당 하였으며 대부분의 LSB 값이 0이므로 무손실 압축된 비트스트림이 원래의 비트스트림과 비슷한 데이터를 갖기 위해서 런이 13일 때는 0000을 할당하고 런이 0일 경우 1101을 할당하였다. 또한 1111은 엔드오브블록(End-of-블럭) 코드로 사용하였고 4비트의 표현을 넘어서는 런이 존재 할 경우 이를 처리하기 위해서 1110은 뒤이어 계속 0이 나온다는 의미의 코드, 즉 이스케이프(Escape) 코드로 사용하였다. <표 2>는 본 논문에서 사용한 런-레벨 코드에 대한 코드 할당을 나타낸다.

표 1. 런-레벨 코드 표

Table 1. Run-level code table.

할당코드	0의 개수	런-레벨(Run-Level)
0000	13	(13,1)
0001	12	(12,1)
0010	11	(11,1)
0011	10	(10,1)
0100	9	(9,1)
0101	8	(8,1)
0110	7	(7,1)
0111	6	(6,1)
1000	5	(5,1)
1001	4	(4,1)
1010	3	(3,1)
1011	2	(2,1)
1100	1	(1,1)
1101	0	(0,1)
1110	14이상	
1111	엔드오브블록(EOB)	

무손실 압축에 의해 길이가 감소된 비트스트림을 CL이라 하면 다음 식 (4)과 같이 표현할 수 있다.

$$CL = \{CL_k \in \{0,1\};\quad 1 \le k \le B\} \tag{4}$$

<그림 3>과 같이 무손실 압축에 의해 감소한 공간에 128bit의 해쉬 함수 값과 소유권자의 정보를 결합하여 인증을 위한 워터마크 비트스트림 WL_a를 생성한다. 이렇게 생성된 WL_a는 L_a가 무손실 압축 된 비트스트

림인 CL, 128bit의 해쉬 함수 출력 값, 소유권자의 정보로 구성되며 삽입되는 소유권자의 정보량을 조절하여 WL_a의 길이가 L_a의 길이와 동일하게 만든다. 따라서 무손실 압축에 의해 감소한 공간에서 128bit의 해쉬 함수 값을 제외한 길이가 삽입 가능한 소유권자의 정보량이 되며 삽입하고자 하는 소유권자의 정보가 삽입 가능한 정보량보다 작을 경우에는 빈 공간에 0(zero)을 패딩(padding)을 하여 WL_a의 길이를 L_a와 동일하게 한다. 만일 소유권자의 정보량이 각 프레임별로 삽입 가능한 정보량을 초과할 경우 해당 프레임과 다음 1~2개의 프레임을 이용하여 WL_a를 생성한다. 위와 같은 방법으로 생성된 WL_a의 각 bit와 L_a 생성을 위해 선택된 AC 계수의 LSB를 교환하여 인증을 위한 워터마크를 생성하게 된다. 이때 삽입 가능한 소유권자의 정보량은 각 프레임별로 가변적이며 본 논문의 실험에서는 skip되는 매크로 블록이 없는 I 프레임일 경우 약 900~1100bit, P 프레임경우 약 400~550bit, B 프레임 경우 250~400bit 정도의 소유권자 정보를 삽입할 수 있다.

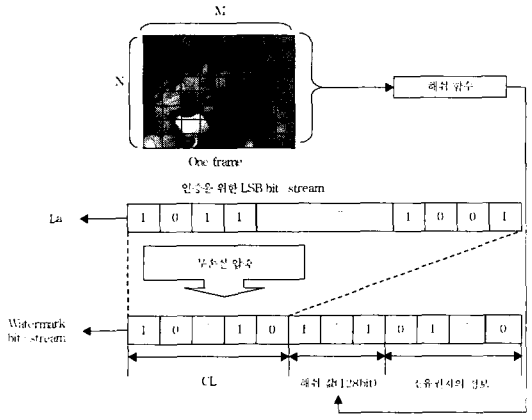


그림 3. 인증을 위한 워터마크의 생성
Fig. 3. Watermark generation for authentication.

2. 조작위치 검출을 위한 워터마크 삽입

조작위치 검출을 위한 워터마크는 8×8 DCT 블록의 양자화된 DC 계수와 고주파 성분의 AC 계수 및 랜덤 시퀀스를 이용하여 삽입하게 된다. 또한 MPEG-2 비디오 데이터에서 I 프레임의 8×8 블록내의 DC 계수는 이전 블록의 DC 계수와의 차분치를 양자화 하여 코딩하기 때문에 비디오 데이터의 비트 레이트가 변화하여도 코딩되는 I 프레임 내의 DC 계수의 값은 동일하다는 점을 이용하고 있다. 조작위치 검출을 위한 워터마크를 생성하기 위해서 우선 <그림 2>의 조작위치 검출을 위한 LSB 비트스트림 생성 과정과 같이 각 8×8 블록에서 AC 계수를 선택한다. 본 논문에서는 고주파 성분의 AC 계수 중 58번째 계수를 선택하였다. 선택된 계수에 2를 곱한 후, LSB를 추출하여 LSB로 구성된 비트스트림을 생성한다. 여기서 선택된 계수에 2를 곱하는 이유는 2를 곱함으로써 선택된 계수의 LSB가 모두 0으로 변하게 되므로2진 데이터의 삽입이 용이하며 쉽게 원래의 계수 값으로 복원 할 수 있는 인버터블 특성을 얻어 낼 수 있기 때문이다. 이렇게 생성된 LSB 비트스트림을 조작위치 파악을 위한 비트스트림이라 부르기로 하고 L_d로 표시 한다. 여기서 L_d는 프레임 내에서 스킵 되지 않은 8×8 블록 개수와 동일한 길이의 0으로 구성된 비트스트림이 된다. 다음으로 사용자의 비밀키를 사용하여 L_d와 동일하거나 더 큰 길이를 갖는 랜덤 시퀀스를 발생시킨다

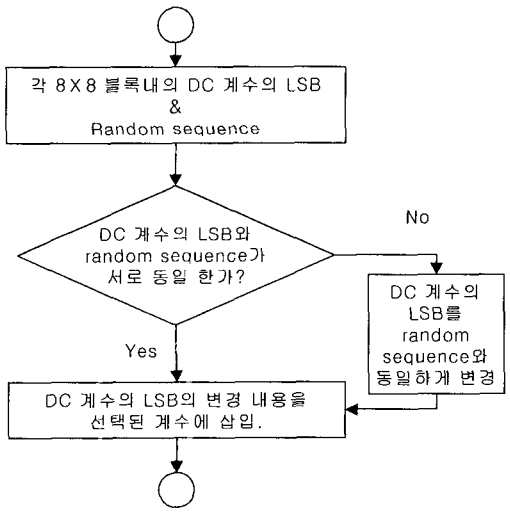


그림 4. 조작위치 검출을 위한 워터마크의 삽입
Fig. 4. Watermark embedding for detection of manipulated position.

<그림 4>에서와 같이 발생된 랜덤 시퀀스의 각 비트에 따라 L_d 생성을 위해 선택된 AC 계수가 속하여 있는 각 8×8 DCT 블록의 양자화된 DCT DC 계수의 LSB를 변경하고 DCT DC 계수의 원 LSB 값은 L_d에 삽입하여 조작위치 검출을 위한 워터마크가 비트스트림 WL_d를 생성한다. 이후 생성된 워터마크 비트스트림 WL_d의 각 비트를 선택된 AC 계수의 각 LSB에 삽입

한다. 이때 선택된 계수는 모두 2가 곱해진 상태이므로 워터마크 비트스트림을 선택된 계수의 LSB에 삽입하여도 선택된 계수의 LSB를 잃어버리지 않는다.

Ⅲ. 워터마크 검출

1. 워터마크의 검출

삽입된 워터마크의 검출은 <그림 5>의 점선으로 표시된 위쪽 블록과 같이 인증 및 조작위치 검출을 위해 삽입된 워터마크를 검출하여 비디오 데이터의 인증을 실시하는 과정과 아래쪽 점선으로 표시된 블록과 같이 비디오 데이터의 변형이 있었음이 판단되면, 비디오 데이터에 가해진 변형이 인위적인 변형인지 혹은 비트스트림 영역에서의 트랜스 코딩에 의한 것인지를 조사하여 인위적 조작으로 판단되었을 경우 조작 위치를 검출하는 순서로 이루어진다.

인증 및 조작위치 검출을 위해 삽입된 워터마크 WL_a와 WL_d를 검출하기 위해서 워터마크가 삽입된 MPEG-2 비디오 비트스트림에 대해서 VLD를 실시하여 양자화된 DCT 계수들로 구성된 프레임을 생성하고 서로 겹치지 않도록 8×8 블록으로 분할 한다. 여기서 우선 WL_a를 검출하기 위해 각 8×8 블록마다 WL_a 삽입 과정에서 선택했던 위치의 AC계수를 선택하여 각 계수의 LSB를 추출한다. 추출된 LSB들로 비트스트림을 생성하면, <그림 6>에서와 같이 생성된 비트스트림은 CL과 해쉬 함수, 그리고 소유권자의 정보로 구성되어 있다. CL에 대해서 압축을 해제하여 L_a와 해쉬 값 및 소유권자의 정보를 각기 분리하고 L_a의 각 비트로 선택된 계수의 LSB를 변경한다.

다음으로 조작위치 검출을 위해 삽입된 워터마크 WL_d를 검출하기 위해서 WL_d 삽입 과정에서 선택한 계수와동일한 위치의 AC 계수를 선택한다. 선택된 계수는 워터마크 삽입 과정에서 2가 곱해지고 LSB에는 해당 블록의 양자화된 DCT DC 계수의 LSB 값이 삽입되어 있다. 따라서 <그림 7>과 같이 선택된 계수를 2로 나누어 주면 몫은 선택된 계수의 워터마크 삽입 이전 계수 값이 되고 나머지는 각 해당 블록의 양자화된 DCT DC 계수의 LSB 값이 된다.

<그림 7>에서와 같이 프레임내의 각 8×8 블록에서 워터마크 검출을 위해 선택된 계수의 값을 몫으로 변경하고, 양자화 된 DCT DC 계수의 LSB는 나머지 값으로 변경해 준다. 삽입된 두 가지 워터마크의 WL_a와 WL_d의 검출 과정을 통해 워터마크가 삽입 되었던 비

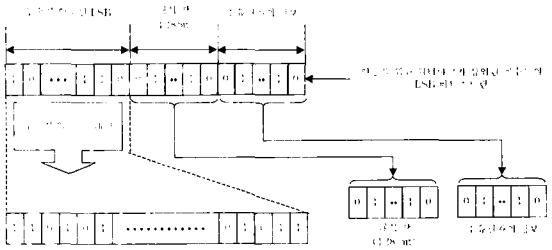


그림 6. 인증을 위한 워터마크 검출
Fig. 6. Watermark detection for authentication.

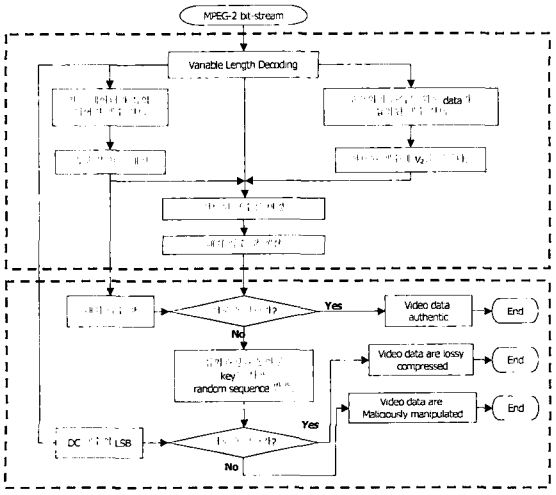


그림 5. 워터마크검출 전체 흐름도
Fig. 5. Flowchart of Watermark detection.

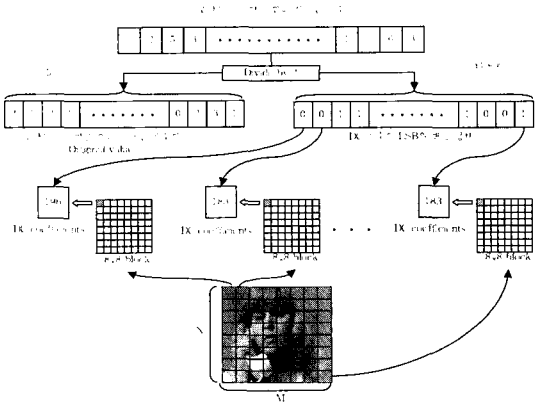


그림 7. 조작위치 파악을 위한 워터마크의 검출 및 계수 값 환원
Fig. 7. Watermark detection for detection of manipulated position and restoration of coefficients.

디오 데이터는 인위적인 조작이 가해지지 않았을 경우 워터마크 삽입 이전의 원본 데이터로 환원된다.

2. 인증 및 조작위치 검출

WL_a 와 WL_d 의 검출 과정을 통해서 워터마크 삽입으로 인해 변경 되었던 각 프레임의 계수들은 워터마크 삽입 이전의 계수 값들로 복원 되었다. 따라서 워터마크 삽입과정에서 사용한 동일한 해쉬 함수에 복원된 각 프레임의 양자화된 DCT 계수들을 입력으로 사용하여 해쉬 값을 구했을 때, 해쉬 함수의 출력 값과 WL_a 의 검출 과정에서 얻은 해쉬 값이 서로 동일 할 경우 입력된 비디오에 대해 무결성을 입증 할 수 있다. 만일 두 해쉬 함수 값이 다를 경우 입력된 비디오 데이터는 조작이 있었던 것으로 판단할 수 있으며 비디오 데이터에 가해진 조작이 비트스트림 영역에서의 트랜스 코딩에 의한 변형인지 인위적인 조작으로 인한 변형인지에 대한 판단이 필요하다. 비디오 데이터에 가해진 변형이 어떤 것인지를 판단하기 위해서는 워터마크가 삽입되어 있는 비디오 데이터의 양자화된 DCT DC 계수의 LSB와 워터마크 삽입 과정에서 사용했던 랜덤 시퀀스와의 관계를 살펴보아야 한다. 이때 랜덤시퀀스와 비교하게 되는 양자화 된 DCT DC 계수는 WL_d 검출 과정을 거치지 않은 DC 계수 이어야 한다. WL_d 삽입 과정에서 I 프레임내 각 8×8 DCT 블록의 양자화된 DCT DC 계수의 LSB는 랜덤 시퀀스와 동일하게 변경하였다. 따라서 <그림 8>에서와 같이 랜덤시퀀스와 양자화된 DCT DC 계수의 LSB를 서로 비교 하였을 때 두 값이 동일하다면 비디오 데이터는 비트스트림 영역에서의 트랜스 코딩에 의해 비트 레이트가 변화 되었다고 판단할 수 있다.

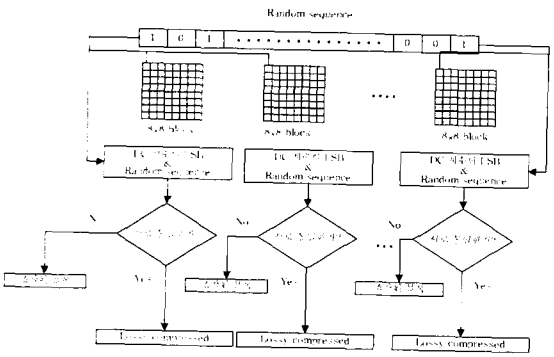


그림 8. 조작의 형태 판단 및 위치 검출
Fig. 8. Decision of manipulation type and detection of manipulated position.

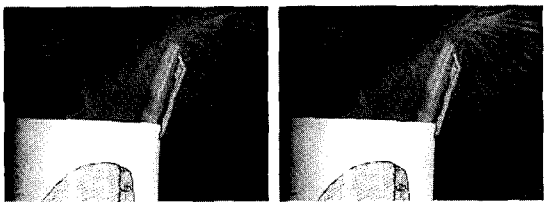
만일 DC 계수의 LSB와 래덤 시퀀스가 서로 동일하지 않은 블록이 존재 한다면 그 블록은 인위적으로 조작된 블록으로 판단할 수 있다. 이러한 판단을 할 수 있는 근거는 MPEG-2 비디오 데이터의 인코딩 알고리즘을 살펴보면 알 수 있다. MPEG-2 비디오 데이터는 인코딩 과정에서 프레임의 각 픽셀 값들을 8×8 블록 단위로 DCT 변환을 하고 각 계수에 할당되어 있는 양자와 스텝과 파라미터에 따라 양자화 하여 VLC (Variable Length Coding)를 거쳐 비트스트림을 생성한다. 이때 인트라 프레임(I 프레임)의 각 8×8 DCT 블록내의 DC 계수는 이전 블록의 DC 계수 값과의 차분치를 이미 설정되어 있는 양자화 매트릭스와 인코딩 과정에서 사용자의 설정에 의해 선택되는 intra_dc_precision이라는 파라미터를 사용하여 양자화 하게된다. 따라서 비트스트림 영역에서의 트랜스 코딩에 대해서는 양자화 매트릭스와 intra_dc_precision 값은 변화하지 않게 되므로 비록 비디오 데이터의 비트 레이트가 변화 할 지라도 인트라프레임, 즉 I 프레임의 양자화된 DCT DC 계수 값은 변화하지 않는다. 따라서 워터마크가 삽입된 인트라 프레임의 각 8×8 블록의 양자화된 DCT DC 계수 값과 랜덤 시퀀스를 비교하여 조작의 형태와 조작위치를 블록 단위로 검출 가능하게 된다.

Ⅳ. 실험결과

제안된 알고리즘의 컴퓨터 시뮬레이션을 위해 본 논문에서는 352×240 화면 크기와 120 프레임 길이의 MPEG-2 비디오 비트스트림을 사용하였다. <그림 9>는 본 논문에서 사용한 원본 비디오 비트스트림의 첫 번째 프레임과 워터마크가 삽입된 비디오 비트스트림의 첫 번째 프레임을 나타내고 있다. <그림 9>에서 확인 할 수 있는 바와 같이 워터마크 삽입에 의한 화질 열화는 눈으로 확인 할 수 없을 정도이다.

<표 2>는 각기 다른 비트 레이트에서 워터마크 삽입에 의한 비디오 비트스트림 파일 크기의 변화를 나타내고 있으며, <표 4>는 워터마크 삽입에 따른 화질 열화를 평균 PSNR을 이용하여 나타내고 있다. <표 2>과 <표 3>에서 알 수 있듯이 워터마크 삽입으로 인한 비디오 데이터의 파일 크기의 변화는 0.5% 미만이며 시각적 왜곡은 일반적인 사람의 시각으로는 감지 할 수 없을 정도로 미약하다.

Susi : 원본 비디오 비트스트림의 첫 프레임 Susi : 워터마크가 삽입된 비트스트림의 첫 프레임



Cact : 원본 비디오 비트스트림의 첫 프레임 Cact : 워터마크가 삽입된 비트스트림의 첫 프레임

그림 9. 원본 비디오 비트스트림과 워터마크가 삽입된 비트스트림

Fig. 9. Original and Watermarked frames.

표 2. 파일 크기 변화

Table 2. File size variation.

Bit rate	Video data	PSNR(dB)
2Mbps	Susi	43.61771
	Cact	43.57458
4Mbps	Susi	43.87521
	Cact	43.59456
8Mbps	Susi	43.95865
	Cact	43.62569

표 3. 평균 PNSR(#1~120 프레임)

Table 3. Average value of PSNR.

		Bit-Stream file size(Bits)		Variation(%)
Bit rate	Video data	Before watermarking	After watermarking	
2Mbps	Susi	1,001,120	1,006,335	0.521
	Cact	1,004,961	1,009,292	0.431
4Mbps	Susi	1,998,980	2,009,459	0.516
	Cact	2,006,803	2,015,412	0.429
8Mbps	Susi	3,999,961	4,019,960	0.509
	Cact	1,017,601	1,021,602	0.425

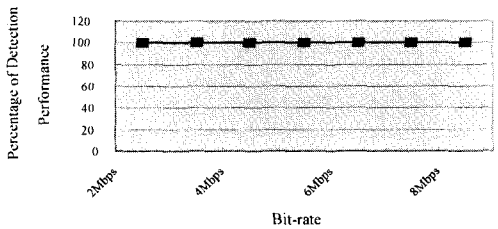
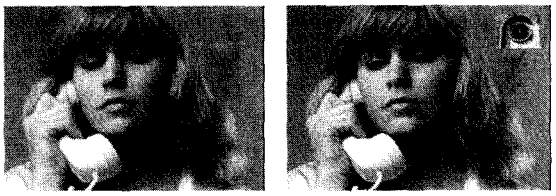


그림 10. 트랜스 코딩에 의한 Bit-rate 변화에 따른 조작위치 검출을 위한워터마크 검출 성능

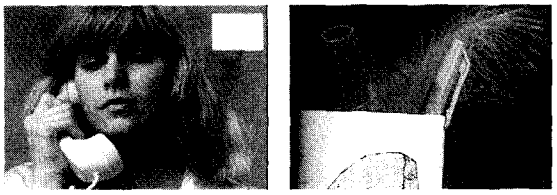
Fig. 10. Detection performance of Watermarking for detection of manipulated position according to bit-rate changing by transcoding.

비트스트림 영역에서의 트랜스 코딩에 대한 워터마크의 검출 결과는 <그림 10>에 표시 되어있다. <그림 10>에서 볼 수 있듯이 제안된 알고리즘은 트랜스 코딩에 의한 비트 레이트 변화에 대해 I 프레임에 삽입된 조작위치 검출을 위한 워터마크를 완벽히 검출 할 수 있다. 즉 인증을 위해 삽입된 워터마크는 8×8 DCT 블록내의 AC 계수에 삽입되므로 트랜스 코딩 과정에서 손상되어 검출되지 않음으로써 비디오 데이터의 조작이 있었음을 알려주고 비디오 데이터의 변형형태 및 조작위치 검출을 위해 I 프레임의 양자화된 DCT DC 계수에 삽입된 워터마크는 손상되지 않고 검출되어 비디오 데이터에 가해진 변형이 비트스트림 영역에서의 트랜스 코딩에 의한 비트 레이트의 변화임을 알려주고 있다. Invertible 특성을 확인하기 위하여 워터마크 삽입 후 변형을 가하지 않고 검출 과정을 수행하여 검출 과정이 종료된 비디오 비트스트림과 원본 비디오 비트스트림간의 PSNR을 비교한 결과 두 비트스트림이 서로 일치하고 있음을 확인하였다.

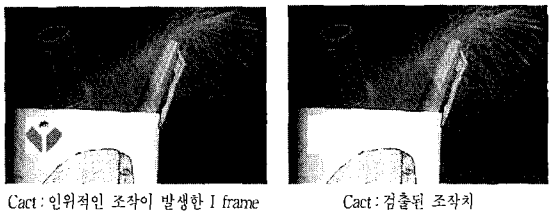
인위적 조작에 대한 검출을 위하여 <그림 11>의 좌측에 있는 원 비디오 프레임은 중앙에 위치한 프레임과 같이 변경 되었으며 조작된 위치는 우측과 같이 흰



Susi : 원본 비디오 비트스트림의 I frame Susi : 인위적인 조작이 발생한 I frame



Susi : 검출된 조작치 Cact : 원본 비디오 비트스트림의 I frame



Cact : 인위적인 조작이 발생한 I frame Cact : 검출된 조작치

그림 11. 인위적 조작과 검출

Fig. 11. Manipulation and detection

사각형으로 검출되었다. 8×8 블록 단위로 조작위치를 검출하므로 검출된 위치에서의 흰 사각형의 크기는 미묘하지만 실제 조작된 영역의 크기보다 조금 크게 나타난다. 제안된 알고리즘에서 조작위치 검출을 위한 워터마크는 각 GOP의 I 프레임에만 삽입되어 있다. 따라서 조작위치 검출은 I 프레임에서만 가능하지만 인증을 위한 워터마크가 모든 프레임에 삽입 되어 있으므로 비디오 데이터의 무결성 여부는 모든 프레임에서 입증 가능하다.

지금까지의 실험결과에서 볼 수 있듯이 본 논문에서 제안된 알고리즘은 기존의 비디오 데이터에 대한 인증 알고리즘이 제시한 비트스트림 영역에서의 워터마크 삽입 및 검출, 프레임 단위의 무결성 검증, 인버티블 특성, 트랜스 코딩 여부의 판단이 가능 할 뿐만 아니라, 비디오 데이터에 인위적인 조작이 가해졌을 경우 I 프레임에 대하여 조작위치를 블록 단위로 파악 할 수 있는 특성을 갖고 있다.

V 결 론

본 논문에서는 MPEG-2 비디오 데이터에 대한 인증 및 조작 형태의 파악과 조작위치 검출이 가능한 가역적 준 연성 워터마킹 기법을 MPEG-2 비트스트림 영역을 기반으로 하여 제시하였다. 제안된 알고리즘은 두 가지 워터마크를 삽입하는데 첫 번째 워터마크는 해쉬 함수와 소유권자의 정보를 무손실 압축 기법을 사용하여 워터마크 데이터를 생성 및 삽입하였으며 비디오 데이터의 인증을 위해 사용된다. 두 번째 삽입되는 워터마크는 비밀 키에 의해 발생된 랜덤 시퀀스와 양자화된 DCT DC 계수 값을 이용하여 조작 형태 및 위치 검출을 위해 사용된다. 본 논문에서 제안된 알고리즘은 통신 채널을 통한 비디오 데이터의 전송시 자주 발생하는 비트스트림 영역에서의 트랜스 코딩에 의한 비디오 데이터의 비트 레이트 변화와 인위적인 조작을 구분함으로써 비디오 데이터에 대한 인증 알고리즘으로서의 실제 활용도를 높였으며 비디오 데이터에 대한 변형이 없었을 경우에 대해서는 워터마크가 삽입되기 이전의 비디오 데이터로의 복원이 가능하다. 따라서 제안된 알고리즘은 비트스트림 영역에서의 트랜스 코딩에 의한 비디오 시퀀스의 비트레이트 가 변화되거나 군사용, 의료용 등과 같은 미세한 화질 열화에도 민감한 분야에 사용할 수 있다.

참 고 문 헌

[1] C. I Podilchuk, "Digital image Watermarking using visual models" Proc. Electronic Imaging, Vol. 3016, pp. 234-242, 1996.

[2] G. C Langelaar, "Watermarking digital image video data. A state-of the art overview", IEEE Signal Processing Magazine, Vol. 17 Issue 5, pp. 20-46, 2000.

[3] F. Hartung, M. Kutter, "Multimedia watermarking techniques", Proc. IEEE, Vol. 87 Issue 7, pp. 1097-1107, 1999.

[4] Chun Shien Lu; Hong-Yuan Liao; Kutter, M. "Denoising and copy attacks resilient watermarking by exploiting prior knowledge at detector", Image Processing, IEEE Transactions on, Vol. 11 Issue 3, pp. 280-292, March 2002.

[5] S. Voloshynovskiy, F. Deguillaume, T. Pun, "Multibit digital watermarking robustagainst local nonlinear geometrical distortions", Image Processing Proc. International Conference on, Vol. 2, pp. 999-1002, 2001.

[6] Ping Wah Wong, Memon, N, "Secret and public key image watermarking schemes for image authentication and ownership verification", Image Proc. IEEE Transactions on, Vol. 10 Issue 10, pp. 1593-1601, 2001.

[7] Min Wu, Bede Liw, "Watermarking for image authentication" Proc. 1998 international conference on, Vol. 2, pp. 437-441, 1998.

[8] J Fridrich, M Goljan, R Du, "Invertible authentication watermark for JPEG images", Proc. SPIE Vol. 4314, pp. 197-208, 2001.

[9] Peng Yin; Yu, H.H., "A semi-fragile watermarking system for MPEG video authentication", Acoustics, Speech, and Signal Processing, 2002 IEEE International Conference on, Vol. 4, pp. 3461-3464, 2002.

[10] Rui Du; Fridrich, J., "Lossless authentication of mpeg-2 video", Image Processing. 2002. Proceedings. 2002 International Conference on, Vol.

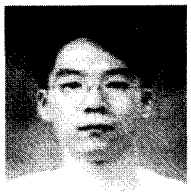
2, pp. 893-896, 2002.

[11] J. Fridrich, M Goljan, "Robust hash functions for ditiatal watermarking", Information Technology : Coding and computing, Proc. International conference on, pp.178-183, 2000.

[12] Lancini, R. Mapelli, F., Tubaro, S., "A robust video watermarking technique for compression and transcoding processing", Multimedia and Expo Proc. IEEE International Conference on, Vol. 1, pp. 549-552, 2002.

[13] A. Vetro, T. Hata, N. Kuwahara, H. Kalva, "Complexity-quality analysis of MPEG-2 to MPEG-4 transcoding airchitectures", ICCE 2002 Digest of technical papers, pp. 130-132, 2002.

저 자 소 개



朴 宰 延(學生會員)
2001년 2월 : 동국대학교 학사 졸업. 2003년 2월 : 동국대학교 석사 졸업. 2003년~현재 : 서울통신기술㈜ 재직 중. <주관심분야 : 디지털 워터마킹, MPRG-2>

林 載 爀(正會員) 第36卷 S編 第1號 參照
현재 : 동국대학교 전자공학과 박사과정 재학 중

元 致 善(正會員) 第34倦 S編 第12號 參照
현재 : 동국대학교 전자공학과 교수