논문 2006-43CI-5-8

LDPC와 BIBD를 이용한 공모된 멀티미디어 핑거프린트의 검출

(Detection of Colluded Multimedia Fingerprint using LDPC and BIBD)

이 강 현*

(Kang Hyeon RHEE)

요 약

멀티미디어 핑거프린팅은 각각의 유저에게 배포되어지는 디지털 콘텐츠마다 고유한 정보를 가지게 만듦으로써 불법적으로 콘텐츠를 배포하는 사용자로부터 멀티미디어 콘텐츠를 보호한다. 또한, 핑거프린팅 기법은 대칭적이나 비대칭적인 기법과 달리 사용자만이 핑거프린트가 삽입된 데이터를 알 수 있고 데이터가 재배포되기 전에는 사용자의 익명성이 보장되는 기법이다. 본 논문에서는 공모자 검출과 에러 신호의 정정을 위하여 LDPC(Low Density Parity Check) 알고리즘을 이용한 멀티미디어 핑거프린트의 검출 알고리즘을 제안한다. 제안된 알고리즘은 LDPC 블록, 홉필드 망, 그리고 불법공모방지코드 생성 알고리즘으로 구성되어 있다. BIBD(Balanced Incomplete Block Design) 기반의 불법공모방지코드는 평균화 선형 공모공격(평균, AND, OR)에 대해 100% 공모코드 검출이 이루어졌으며, LDPC 블럭은 AWGN 0dB까지 에러비트를 정정할 수 있음을 확인하였다.

Abstract

Multimedia fingerprinting protects multimedia content from illegal redistribution by uniquely marking every copy of the content distributed to each user. Differ from a symmetric/asymmetric scheme, fingerprinting schemes, only regular user can know the inserted fingerprint data and the scheme guarantee an anonymous before recontributed data. In this paper, we present a scheme which is the algorithm using LDPC(Low Density Parity Check) for detection of colluded multimedia fingerprint and correcting errors. This proposed scheme is consists of the LDPC block, Hopfield Network and the algorithm of anti-collusion code generation. Anti-collusion code based on BIBD(Balanced Incomplete Block Design) was made 100% collusion code detection rate about the linear collusion attack(average, AND and OR) and LDPC block for the error bits correction confirmed that can correct error until AWGN 0dB.

Keywords: Multimedia Fingerprint, LDPC, BIBD, ACC, Digital copyright

I. 서 론

현재 사용되어지고 있는 데이터 기밀성 유지 메커니즘은 인터넷 및 방송 등의 대중매체를 통하여 재 가공된 후 분배되어질 경우 그 기밀성을 유지하기 힘들며 공모자들의 공모공격 등에 의해 손쉽게 디지털 콘텐츠의 기밀성이 파괴되어질 수 있다. 이러한 문제점들이 대두되어짐에 따라 디지털 콘텐츠 보호기술에 대한 연구가 활발히 진행되어지고 있다[1].

디지털 콘텐츠 보호기술은 콘텐츠 제작자의 저작권 관련 정보를 외부공격에 강인하도록 콘텐츠에 삽입하는 기술로 정의할 수 있으며 이는 크게 워터마킹 기술과 핑거프린팅 기술로 나누어진다. 워터마킹 기술은 저작권 정보를 워터마크로 변환시켜 비가시적으로 콘텐츠에 삽입하는 기술로써 콘텐츠 제작자의 소유권을 인증할 수 있는 기술이지만, 불법적인 유통과정과 공모 공격자를 알

* 평생회원, 조선대학교 전자공학과
(Dept. of Electronic Engineering, Chosun University)
접수일자: 2006년7월24일, 수정완료일: 2006년8월11일

수 없다는 단점이 있다. 이러한 워터마크의 문제점을 해결하기 위하여 핑거프린팅 기술에 대한 연구가 진행되어지고 있다. 디지털 핑거프린팅은 워터마킹의 확장 기술로 워터마크 삽입/추출 알고리즘을 이용하여 디지털 콘텐츠에 핑거프린트가 삽입/추출 되어지나, 워터마크의 단점을 보완하기 위하여 디지털 콘텐츠마다 고유사용자 정보(unique digital signature)가 삽입되어 원 저작자의 지적재산권리를 보호할 수 있다. 또한, 사용자들이 공모하여 복제 콘텐츠를 만드는 공모공격의 문제가 발생되었을 때, 공모공격자들을 추적하여 검출할 수 있는 디지털 콘텐츠 보호기술이다[2].

핑거프린팅은 디지털 콘텐츠마다 고유한 구매자 정보를 삽입하기 때문에 핑거프린팅 된 콘텐츠도 서로 조금씩 다르게 되는데, 이 차이점을 이용하여 핑거프린트 정보를 제거하려 하는 공모공격이 가능하게 된다. 대표적인 공모공격 방법에는 평균화 공모공격(Averaging Attack), 최대-최소공격(Max-Min Attack), 상관계수 음수화공격(Negative-Correlation Attack), 제로-상관공격(Zero-Correlation Attack) 그리고 모자이크 공격(Mosaic Attack) 등이 있으며[3] 핑거프린팅 기술은 이러한 공모공격에 강인하도록 개발되어야 한다. 대표적인 디지털 핑거프린팅 기술은 크게 듀얼 워터마킹/핑거프린팅 기법[4]과 공모보안코드 (collusion secure code)[5~10]로 나누어진다. 듀얼 워터마킹/핑거프린팅 기법은 현재 사용되어지는 워터마크 삽입/검출 알고리즘에 핑거프린트를 워터마크로 삽입하는 방법이며, 공모보안 코드 기법은 공모공격에 강인하도록 핑거프린팅 코드 자체를 공모가 어렵도록 설계한 코드로 Boneh와 Shaw가 제안한 c-secure와 c-frameproof 코드[5], Dittmann이 제안한 d-dtecting 코드[6], Domingo-Ferrer가 제안한 3-secure 코드[7,8] 그리고 Trappe가 제안한 Anti-Collusion 코드[11] 등이 있다.

본 논문에서는 [2]에서 제안된 멀티미디어 핑거프린트의 불법공모코드 검출 알고리즘에 LDPC 부호기의 패리티 체크 행렬[12]을 적용시켜 공모된 핑거프린트와 불법사용자를 검출 할 수 있는 알고리즘을 제안하였다.
실험을 통하여 공모보안 코드인 BIBD 코드의 불법 공모공격에 대한 강인성과 홉필드 망(Hopfield Network) 및 LDPC의 패리티 체크행렬이 적용되었을 때의 에러정정 성능을 측정하였다. 이를 위해 Ⅱ장에서는 BIBD, LDPC 그리고 홉필드 망의 이론적 배경을 설명하겠으며, Ⅲ장에서는 본 논문에서 제안된 공모된 핑거프린트의 검출 알고리즘을 설명하고, Ⅳ장에서 제안된 알고리즘의 성능을 [2]와 비교하여 측정 및 결과를 검토하겠다. 그리고 마지막으로 Ⅴ장에서 결론과 향후 연구방향에 대해 고찰하겠다.

Ⅱ. 이론적 배경

1. LDPC

LDPC 부호는 최근에 가장 주목 받는 오류정정부호로 1960년대 초 Gallager[12]에 의해 제안된 부호로, 패리티 검사 행렬의 0이 아닌 원소의 수가 부호의 길이에 비해 현저히 적게 존재하는 부호로 정의되며 섀논(Shannon) 한계에 가장 근접하는 오류정정부호로서, 터보부호와 더불어 제4세대 이동통신시스템에 활용될 수 있는 매우 우수한 오류정정부호로 평가되고 있다.

식 (1)은 [12]를 참조한 LDPC 부호화 과정을 나타낸다.

$$H = (A_p^{-1} \cdot A) \bmod 2 = [I \; A_2]$$
$$G = \left(\frac{A_2}{I}\right) \tag{1}$$
$$c = (G \cdot m) \bmod 2$$

A, H : 패리티 체크 행렬(parity check matrix)
A_p^{-1} : 역 피봇 행렬(inverse pivot matrix)
G : 생성 행렬(generator matrix)
m : 전송 메시지(transmission message)
c : 부호어(code word)

패리티 체크 행렬 A와 H를 이용하여 생성 행렬 G를 만든 다음 G와 메시지 m을 사용하여 부호어 c가 생성되어진다.

식 (2)는 LDPC 복호화 과정을 나타낸다.

$$q_n(x) = \alpha P(c_n = x|r_n) \Pi_{m \in u} P(z_m = 0|c_n = x, r) \tag{2}$$

$q_n(x)$: 의사 사후확률(pseudo posterior probability)
$\alpha P(c_n = x|r_n)$: 내부확률(intrinsic probability)
$\Pi_{m \in n} P(z_m = 0|c_n = x, r)$: 외부확률(extrinsic probability)
z_m : 패리티 체크 비트(parity check bits)
c_n : 부호어(code word)
n, m : 행과 열의 인덱스(row, column index)
u : 1의 위치 인덱스(1's position index)
r : 수신된 전체 신호(total received data)

전송 채널을 통과한 코드워드 c는 잡음 및 에러 성분이 첨가되어 r의 형태로 수신되고 수신된 신호는 식 (2)에서 채널의 사후확률 계산을 통하여 복호된다.

2. 균형불완비블록설계

BIBD 코드는 반공모(Anti-Collusion) 코드의 특징을 만족한다. 즉, 공모공격에 강인성을 가지는 코드로서, n개의 코드 벡터 중에서 $(n-1)$명의 공모자를 검출할 수 있다. BIBD 코드는 5개의 파라메터 (v, b, r, k, λ)로 생성되는데,

- v: 처리의 개수(number of treatments)
- b: 블록의 개수(number of blocks)
- r: 각 v의 반복 수(number of times each treatment is run, k<v)
- k: 하나의 블록에 포함된 v의 개수(number of treatments per block)
- λ: 각 처리 쌍이 나타나는 블록의 개수(number of blocks that processing pair appears)

5개의 파라미터는 식 (3)부터 (6)까지의 한정조건을 만족하며,

$$vr = bk \tag{3}$$

$$r(k-1) = \lambda(v-1) \tag{4}$$

$$b = \frac{v(v-1)\lambda}{k(k-1)} \tag{5}$$

$$r = \frac{\lambda(v-1)}{k-1} \tag{6}$$

$v \times b$의 크기를 갖는 BIBD코드 M은 식 (7)에 의해 내부 값이 결정되어진다.

$$M = [m_{ij}]$$
$$m_{ij} = \begin{cases} 1 \text{ if } j^{th} \; blocks \in i^{th} \; elements \\ 0 \; otherwise, \end{cases} \tag{7}$$

접속행렬(Incidence Matrix) M의 행벡터는 핑거프린트 코드가 되며 b명의 사용자들에게 부여되고, 이러한 M은 반공모 코드로 사용할 수 있다.

3. 홉필드 망

홉필드 망은 상호결합형 신경망 모델[13,14]로서 많은 수의 비동기적이고 국소적인 계산을 통하여 전역적 최적화(global optimization)를 이룰 수 있다. 특히 일정한 범용 패턴들을 특정 연결강도로 저장하였다가 미지의 입력패턴이 주어질 때 이와 가장 유사한 패턴을 찾아낼 수 있는데, 이는 인간의 기억방식과 유사한 방법으로 일부분의 정보를 가지고 그와 연관된 많은 부분을 기억해 내는 방법이다. 이와 같이 입력되는 데이터에 의해서 저장된 정보를 찾아내는 메모리를 내용지정 메모리(CAM:Content Addressable Memory) 또는 연상메모리라 하며 결과적으로 연상메모리 구조는 이진 데이터의 에러정정 회로에 적용되어질 수 있다.

홉필드 망은 자신을 제외한 모든 유니트(뉴론)들 간에 양방향으로 상호연결된 회로망으로 출력신호는 모든 유니트에 피드백 되어 식 (8)과 같이 각 유니트의 출력이 결정된다.

$$\sum_{i=0}^{n-1} \frac{\sigma_i \cdot \mu_i}{R_i} = \begin{cases} > 0 \; : \; V_{out} = 'high' \\ < 0 \; : \; V_{out} = 'low' \end{cases} \tag{8}$$

n : 유닛의 수(number of units)

σ_i : 입력벡터의 요소(elements of input vectors)

μ_i : 저장벡터의 요소(elements of storage vectors)

R_i : 연결저항(connection registration)

식 (8)의 연결저항은 식 (9)로 정의 되어지며 R_j는 유니트 j로부터 유니트 i로의 연결저항이고, x_i는 s번째 패턴의 i번째 요소이다.

$$R_{i,j} = \begin{cases} \sum_{s=0}^{M-1} x_i^s x_j^s & i \neq j \\ 0 & i = j \end{cases} \tag{9}$$
$$\text{for } 0 \leq i, \; j \leq M-1$$

결과적으로 유니트에 기억된 내용은 에러가 있는 유사한 벡터와 전역 최적화를 이룰 수 있기 때문에 에러정정 기능을 수행한다.

III. 핑거프린트의 불법공모 코드 검출

공모 공격자들은 콘텐츠에 삽입된 인식정보 즉 핑거프린트의 제거 및 검출 정보의 모호성을 증대하기위하여 평균화, 최대-최소공격, 상관계수 음수화, 제로-상관공격 그리고 모자이크 등의 공격을 콘텐츠에 가하며, 결과적으로 공모공격자에 대한 모든 추적을 제거하려고 한다.

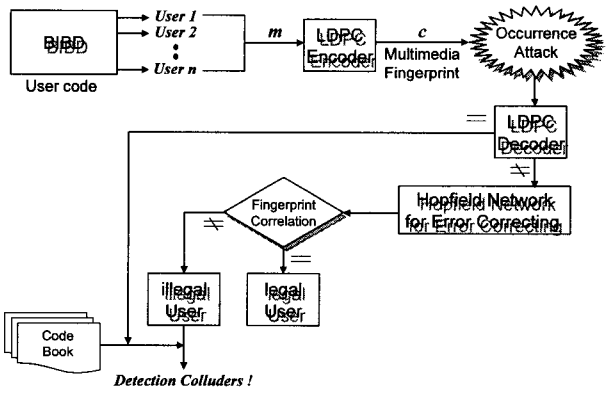

그림 1. 제안된 멀티미디어 핑거프린트의 불법공모 코드의 검출 흐름도

Fig. 1. The proposed multimedia fingerprint detection flow diagram.

본 논문에서 이러한 공모공격자 및 에러에 강인성을 가지는 멀티미디어 핑거프린팅 알고리즘을 제안하였으며 그림 1은 제안된 멀티미디어 핑거프린트의 불법공모 코드의 검출 흐름도이다.

1. 핑거프린트 생성 알고리즘

제안된 알고리즘에서는 사용자를 식별할 수 있는 코드로 BIBD 코드를 사용하였으며 LDPC 부호기에 사용자코드를 입력시켜 부호어를 생성시켰다. 생성된 부호어는 패리티체크비트와 BIBD 코드로 구성되어 있는데 이중 BIBD 코드를 핑거프린트 코드로 사용하였으며, 패리티 체크비트는 멀티미디어에 삽입된 핑거프린트의 외부 공격여부 및 에러 정정에 사용되어진다. 그림 2는 5비트의 핑거프린트 생성 과정을 설명하고 있으며 GF(2)상에서 계산된다.

그림 2에서 $n \times m$ 크기의 행렬 G는 식 (1)에서 유도된 생성 행렬로 BIBD의 한정 조건을 만족하며, n의 길이는 사용자 코드와 일치해야 된다.

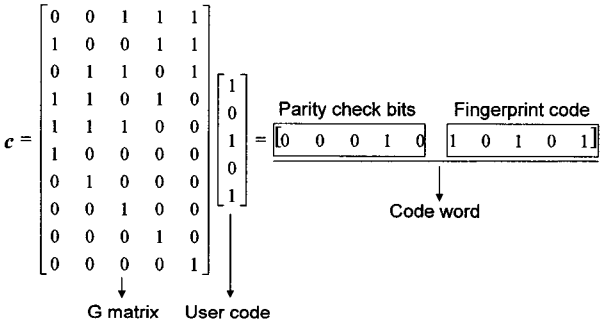

그림 2. 핑거프린트 생성

Fig. 2. Fingerprint generation.

2. 핑거프린트 검출 알고리즘

핑거프린트의 검출은 크게 두 가지 과정으로 이루어져 있다. 첫 번째 과정은 LDPC 복호기를 사용하여 핑거프린트의 공격 여부를 결정 및 AWGN 등의 채널 잡음 공격에서 발생되는 에러를 정정한다. 즉, 복호기의 반복횟수가 일정값 이상이 되어서도 패리티 체크 값이 0으로 수렴되지 않으면 공모자에 의한 공모 공격이 발생된 것으로 간주하고 두 번째 검출 과정으로 복호된 데이터를 넘겨준다. 두 번째 과정은 에러정정을 위한 홉필드 망이다. 공모공격이 발생되었을 때 복호된 코드를 홉필드망에 입력시켜 에러 신호를 정정하고, 정정된 핑거프린트는 코드북과의 상관관계를 계산하여 공모자를 검출한다.

핑거프린트 검출에 사용된 상관계수는 식 (10)를 사용하여 계산하였다.

$$k = \frac{\frac{1}{n}\sum_{m=1}^{n}(a_m - \bar{a})(b_m - \bar{b})}{\sigma_a \sigma_b} \tag{10}$$

k : 상관계수(coefficient of correlation), $-1 \le k \le 1$

$\bar{a}, \bar{b}$: 평균(average)

σ_a, σ_b : 표준편차(standard deviation)

검출된 핑거프린트는 식 (2)에 의해 복호된 후 식 (11)의 조건을 만족하지 않으면 외부 공격이 가해진 것으로 가정하고 홉필드 망을 이용하여 에러를 정정한다.

$$\begin{cases} Normal\ fingerprint & \text{if } G \cdot C = 0 \\ Attacked\ fingerprint & \text{if } G \cdot C \neq 0 \end{cases} \tag{11}$$

G : 생성 행렬(Generator matrix)

C : 검출된 핑거프린트(Detected fingerprint)

홉필드 망은 식 (1)에 의해 생성된 핑거프린트에 공격이 가해졌을 때 피드백형 연상메모리방식에 의해 에러가 정정되는 회로로 최종적으로 코드북을 참조하여 공모자를 검출하게 된다. 그림 3은 본 논문에서 설계한 홉필드 망 정정 회로이며, 7비트의 핑거프린트 코드 중 1비트의 에러를 정정하여 불법공모의 여부를 확인할 수 있다. 전체회로는 N형과 P형의 MOSFET(Metal-Oxide Semiconductor Field Effect Transistor)으로 구현하였으며 MOSFET의 채널폭과 채널길이 및 게이트에 연결되는 입력값의 변화에 따라 MOSFET의 상태가 홍분과 억제상태로 제어되며, 이에 따라 입력 데이터의 에러가

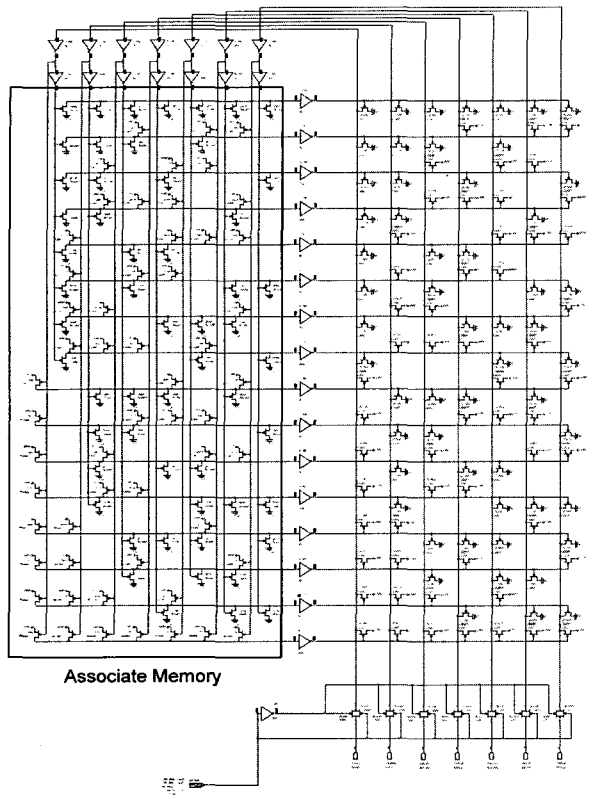

그림 3. 홉필드 망을 이용한 에러정정회로
Fig. 3. Error correction circuit using Hopfield network.

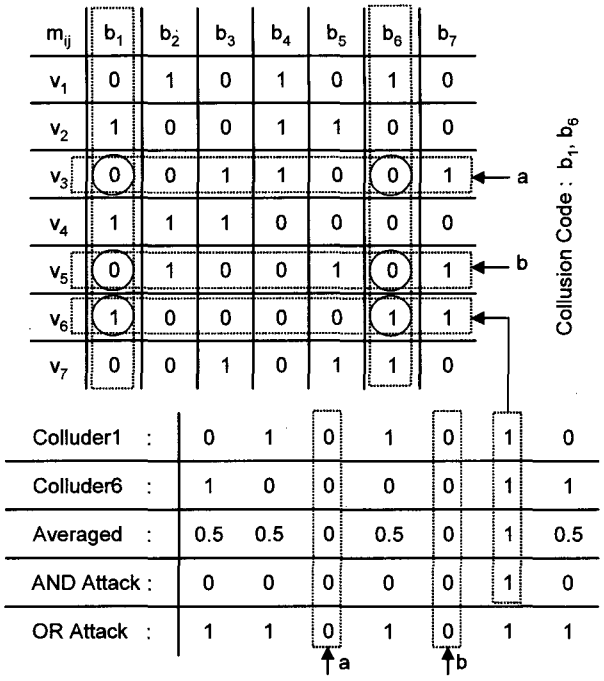

그림 4. 공모자 검출과정
Fig. 4. Colluders detection process.

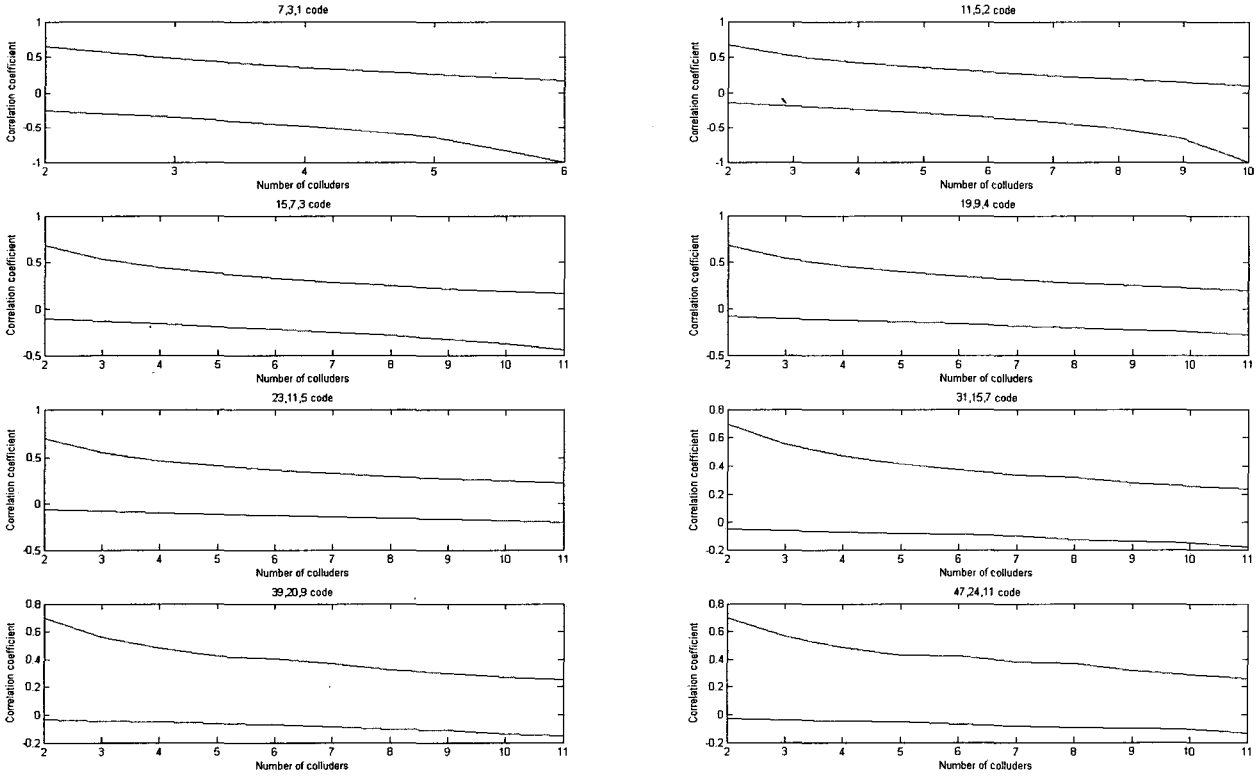

그림 5. 공모코드와 비공모코드의 상관계수
Fig. 5. Correlation coefficient of collusion and anti-collusion code.

표 1. 공모 공격에 대한 검출된 공모자수
Table 1. Number of the detected colluders by collusion attack.

	Number of Colluders							
BIBD Code	6	10	14	18	22	30	38	46
{7,3,2}	6	-	-	-	-	-	-	-
{11,5,2}	6	10	-	-	-	-	-	-
{15,7,3}	6	10	14	-	-	-	-	-
{19,9,4}	6	10	14	18	-	-	-	-
{23,11,5}	6	10	14	18	22	-	-	-
{31,15,7}	6	10	14	18	22	30	-	-
{39,20,9}	6	10	14	18	22	30	38	-
{47,24,11}	6	10	14	18	22	30	38	46

표 2. AWGN 변화에 따른 검출된 공모자수
Table 2. Number of the detected colluders by changing AWGN.

BIBD코드	AWGN[dB]						
	4	2	0	-1	-2	-3	-4
7,3,2	0	0	0	0.2	3.1	4.7	13.7
11,5,2	0	0	0	0.7	3.2	6.8	14.4
15,7,3	0	0	0	0.9	3.2	9.3	18.7
19,9,4	0	0	0	1.3	4.3	11.2	20.2
23,11,5	0	0	0	1.4	4.6	11.6	21.3
31,15,7	0	0	0	1.4	4.9	12.5	23.9
39,20,9	0	0	0	1.8	5.4	13.5	25.2
47,24,11	0	0	0	2.1	7.1	15.9	25.9

표 3. 홉필드망을 통과한 핑거프린트 코드에서 검출된 공모자수
Table 3. Number of the detected colluders in fingerprint code that passed Hopfield network.

BIBD코드	AWGN[dB]			
	-1	-2	-3	-4
7,3,2	0.00	1.55	4.28	13.19
11,5,2	0.02	1.60	6.52	14.12
15,7,3	0.05	1.62	8.69	18.41
19,9,4	0.11	2.16	10.96	19.17
23,11,5	0.13	2.37	11.10	20.99
31,15,7	0.17	2.49	11.97	22.98
39,20,9	0.20	3.58	12.91	24.06
47,24,11	0.24	5.80	15.15	25.44

정정되어진다. 또한, 본 논문에서는 핑거프린트의 코드 길이 및 정정 가능한 비트의 수를 증가시키기 위하여 Matlab을 이용하여 홉필드 망 회로를 설계하여 시뮬레이션에 사용하였다.

Ⅳ. 실험 및 결과 검토

제안된 알고리즘의 성능 측정을 위하여 Matlab으로 시뮬레이션 환경을 구현하였으며, 사용된 컴퓨터는 인텔 펜티엄Ⅳ 3.0GHz CPU와 4.0GB RAM을 가진다. 본 논문에서는 [2]와의 성능 비교를 위하여 BIBD 코드의 파라미터 $\{v,k,\lambda\}$가 {7,3,1}, {11,5,2}, {15,7,3}, {19,9,4}, {23,11,5}, {31,15,7}, {39,20,9} 그리고 {47,24,11}의 조건을 가지는 코드를 생성하여 실험하였다. 공모자의 수는 $(v-1)$명으로 제한하여 공모자 검출 실험을 진행하였으며 실험 방법은 크게 AND, OR 공격 등의 평균화 공격에 대한 강인성과 외부 잡음공격에 의해 변형되는 비트에러에 대한 강인성을 실험하였다.

1. 공모 공격에 대한 강인성

그림 4는 {7,3,1} BIBD 코드를 사용하여 7명의 사용자 중 2명의 공모공격자를 구분하는 과정을 설명하고 있다.

OR, AND 그리고 평균화 공격에 대해서 생성된 부호의 '0'과 '1'의 상관관계 및 각 코드간의 상관계수를 계산하여 공모자를 검출할 수 있다.

그림 5는 공모코드와 코드북과의 상관계수를 나타내며 상관계수(r)가 0보다 큰 경우에 공모자, 0보다 작은 경우에는 비공모자로 판별하였다.

표 1은 공모 공격에 대한 공모자 검출결과로 공모평균화, AND, OR 공격에 대하여 제안된 알고리즘은 공모자를 100% 검출하였다.

2. LDPC와 신경망 회로를 이용한 비트 에러정정

공모공격이외에 핑거프린트코드에 잡음 및 고의적인 비트 조작의 공격을 가할 수 있는데 본 논문에서는 이러한 공모공격에 대한 강인성을 가지기 위하여 LDPC 알고리즘을 적용하였으며, 연상메모리 구조를 사용하여 이진 데이터의 에러를 정정할 수 있는 홉필드 망을 설계하였다.

표 2는 각각의 코드를 1,000개씩 생성하여 AWGN의 변화에 따른 공모자 검출 결과표이다. AWGN을 4dB부터 -4dB까지 변화시키며 실험한 결과 0dB까지는 공모자수를 정확히 검출할 수 있지만 -1dB 이하부터는 코드길이에 비례하여 검출할 수 있는 공모자의 수가 감소함을 알 수 있다.

표 3은 LDPC 복호기를 통과한 핑거프린트 코드 중 공모가 발견된 코드를 홉필드 망을 통하여 재 정정한 코드에서 검출된 공모자의 수이다. 홉필드망 사용 시 표 2에 비해 -1dB와 -2dB에서는 약 40%정도의 성능이 향상되었음을 알 수 있다.

결과적으로 본 논문에서 제안된 신경회로망에 의한 핑거프린트 검출 알고리즘은 설계된 BIBD 기반의 코드에 의해 평균화 공모공격에 대해서는 100% 공모자 검출이 가능하며, LDPC와 홉필드 신경회로망에 의해 공모코드의 비트 변환 공격에 대해서 AWGN 0dB까지에 대해서 공모자를 정확히 검출할 수 있다.

V. 결 론

본 논문에서는 최근에 활발히 연구가 진행되고 있는 멀티미디어 핑거프린팅 알고리즘을 제안하였다. 핑거프린팅 기술은 멀티미디어 콘텐츠에 구매자마다 고유한 정보를 입력시키므로 콘텐츠가 불법 복제되었을 때 원 구매자를 추적할 수 있으나, 각각의 콘텐츠마다 내용이 약간씩 다름을 이용한 공모공격이 존재하게 된다. 따라서 워터마킹 기술과는 달리 공모공격에 강인성을 가져야한다.

본 논문에서 제안된 알고리즘은 LDPC 블록, 홉필드 망, 그리고 불법공모방지코드 생성 알고리즘으로 구성되어 있다. BIBD 기반의 불법공모방지코드는 선형 공격(평균, AND, OR)에 강인성을 가지며, LDPC 블록과 피드백형 연상메모리 방식의 홉필드 망은 외부공격에 의해 발생한 에러비트를 정정한다. 실험 결과 BIBD 기반의 불법공모방지코드는 평균화 선형 공모공격에 대해 100% 공모코드 검출이 이루어졌으며, LDPC 블록과 홉필드 망을 사용함으로써 AWGN의 변화에 따른 외부 잡음 첨가에 대해서는 0dB까지 공모자수를 정확히 검출할 수 있었다.

앞으로의 연구는 제안된 멀티미디어 핑거프린트를 삽입할 수 있는 알고리즘 개발과 제한된 크기의 콘텐츠에 효율적으로 삽입하기 위한 코드 개발 및 다양한 핑거프린팅 정보를 수용할 수 있는 대용량 핑거프린팅 코드체계에 대한 연구가 진행되어야겠다.

참 고 문 헌

[1] Zang Li, Wade Trappe, "collusion-resistant Fingerprints from WBE Sequence Sets," ICC 2005 IEEE International Conf., Vol 2, pp. 1336-1340, May 2005.

[2] J.S. Noh, K.H. Rhee, "Detection of Colluded Multimedia Fingerprint by Neural Network," Journal of The Institute of Electronics Engineers of Korea, Vol. 43-CI, NO. 4, July 2006.

[3] H. Stone, "Analysis of Attacks on Image Watermarks with Randomized Coefficients," NEC Technical Report, 1996.

[4] D. Kirovski, H.S. Malvar, and Y. Yacobi. "Multimedia Content Screening using a Dual Watermarking and Fingerprinting System," in Proc. of ACM Conf. on Multimedia, pp. 372-381 , France, 2002.

[5] D. Boneh and J. Shaw, "Collusion-Secure Fingerprinting for Digital Data," IEEE Trans. Inf. Theory, Vol. 44, No. 5, pp. 1897-1905, Sep. 1998.

[6] J. Dittmann, "Combining Digital Watermarks and Collusion Secure Fingerprints for Customer Copy Monitoring," Proc. IEE Seminar Sec. Image & Image Auth., pp. 128-132, Mar. 2000.

[7] J. Domingo-Ferrer and J. Herrera-Joancomarti, "Simple Collusion-secure Fingerprinting Schemes for Images," in IEEE International Conference on Information Technology: Coding and Computing,

ITCC'2000, ISBN 0-7695-0540-6, pp. 128-132.
[8] F. Sebe and J. Domingo-Ferrer, "Short 3-Secure Fingerprinting Codes for Copyright Protection," Lecture Notes in Computer Science, Vol. 2384, pp. 316- 327, 2002.
[9] Yiwei Wang, John F. Doherty, and Robert E. Van Dyck, "A Watermarking Algorithm for Fingerprinting Intelligence Images," 2001 Conference on Information Sciences and Systems, The Johns Hopkins University, March 21-23, 2001.
[10] W. Trappe, M. Wu, and K. J. R. Liu, "Collusion-Resistant Fingerprinting for Multimedia," Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'02), Vol. IV, pp. 3309-3312, Orlando, FL, May 2002.
[11] W. Trappe, M. Wu, Z. Jane Wang, and K.J.R. Liu, "Anti-Collusion Fingerprinting for Multimedia," IEEE Trans..on Signal Processing, Vol. 51, No. 4, pp. 1069-1087, Apr. 2003.
[12] R.G. Gallager, "Low-density parity-check codes," IRE Trans. Inform. Theory, vol. IT-8, pp. 21-28, Jan. 1962.
[13] J. J. Hopfield and D. W. Tank, "Neural Computation of Decision in Optimization Problem," Biol. Cybern. Vol. 52, 1985.
[14] J. Freeman and D. Skapura, "Neural Networks," Addison-Wesley Publishing Company, 1991

저 자 소 개

이 강 현(평생회원)
1979년, 1981년 조선대학교 전자공학과 공학사 및 석사
1991년 아주대학교 대학원 공학박사
1977년～현재 조선대학교 교수
1991년, 1994년 미 스탠포드대 CRC 협동연구원.
1996년 호주 시드니대 SEDAL 객원교수
2000년～현재 한국 멀티미디어기술사협회 이사
2002년 영국 런던대 객원 교수
2002년 대한전자공학회 멀티미디어연구회 전문 위원장
2003년 한국 인터넷 방송/TV 학회 부회장
2003년～현재 대한전자공학회 상임이사
2005년～현재 조선대학교 RIS 사업단장
<주관심분야 : 멀티미디어 시스템설계, Ubiquitous convergence>