

저 사양 안드로이드 기반 동영상 보안을 위한 워터마크 시스템 개발

황 선 철 *

Development of Video Watermark System for Low-specification System as Android Platforms

Seon-Cheol Hwang *

요 약

본 논문에서는 안드로이드 플랫폼과 같은 낮은 사양의 시스템에서 처리속도에 부담을 주지 않고 공간영역에서 워터마크의 삽입 및 검출 알고리즘을 제안하고자 한다. 워터마크는 사용자의 정보를 암호화하여 삽입하는 핑거프린트의 형태로 삽입되며 노출 위험을 피하기 위해 사용자 ID를 블록 암호화하여 의사 난수화하고 이를 토대로 워터마크의 삽입위치, 크기 및 삽입 프레임 번호에 변화를 주는 방법을 사용하였다. 또한 동영상의 특징상 많은 프레임이 존재하므로 이런 특징을 적극 활용하여 검출 결과를 안정화한 후 적분을 통하여 검출 가능성을 높였다.

본 연구에서 제시한 워터마크의 삽입 및 검출은 제한된 조건하에서도 높은 검출율을 얻을 수 있었다. 또한 검출 결과를 시각적으로 보여주기 위한 결과에서도 비교적 선명한 결과를 얻을 수 있었다. 상관계수 r 연산에 의한 결과도 움직임이 많은 경우 0.87, 중간의 경우 0.79, 움직임 적은 경우 0.84라는 높은 결과를 얻을 수 있어서 0.7 이상인 경우 확정적인 것을 참고하면 높은 검출 결과를 얻게 되었다. 워터마크 기호 영상 추출 단계에서 성공률 95.45%의 결과를 얻었으며 이 결과 영상에 상관계수 r 연산을 적용하면 99% 이상 검출 결과를 얻을 수 있었다. 향후 다양한 공격에 대비한 알고리즘의 보완을 통해 상품화의 단계를 지향하고 있으며 이를 통해 영상 관제 시스템과 지적 재산권 보호 시스템에 적용할 예정이다.

▶ Keywords : 워터마크, 안드로이드, 보안, 동영상, 핑거프린트

Abstract

This paper describes a method to insert and detect watermark or fingerprint to/from videos in low-computing powered system as Android platforms. Fingerprint, which is a kind of watermark, has features such as patterns that contain information. The inserting frame numbers in video-stream and the positions in a picture were chosen from the encrypted user ID to insert the

•제1저자 : 황선철

•투고일 : 2014. 6. 10, 심사일 : 2014. 6. 20, 게재확정일 : 2014. 7. 23.

* 인덕대학교 방송영상미디어과(Dept. of Broadcasting and media engineering, Induk University)

watermarks. The used encrypt algorithm is the HIGHT algorithm which was developed for low-computing powered systems by KISA(Korean Internet & Security Agency). Subtracting an inferred picture from the previous picture was used to extract a candidate feature. Median filtering was used to get rid of noise and stabilize the candidate feature. New algorithm that reduces calculating steps of the median filtering was developed and applied for low-specification systems. The stabilized features were accumulated over 150 times and calculated by correlation coefficient method to recognize the patterns. We examined 22 videos and successfully detected the patterns from 21 videos. The correlation coefficient r values that we examined through this study exceeded over 0.79 more than the threshold (0.7).

▶ Keywords : Watermark, Android, Video, Fingerprint

I. 서론

시스템의 성능이 급격히 증가하면서 동영상의 활용과 수요가 기하급수적으로 늘어나고 있지만 여전히 감시용 카메라인 CCTV나 차량용 블랙박스(Blackbox) 및 핸드폰은 연산능력이 낮은 저 사양 시스템이 주를 이루고 있다. 또한 기존의 동영상 보안개념은 드라마나 오락물이 대부분을 차지하고 있어 저작권 보호가 필요했지만, CCTV의 증가와 차량 블랙박스의 증가는 동영상의 사생활보호 필요성이 대두되고 있다. 이러한 동영상의 저작권 보호와 CCTV 등의 보안 통제를 위해서는 유통을 투명하게 하고 관제할 수 있는 개념이 필요하게 되는데 이를 위해서 동영상에 취급자의 정보를 삽입하는 기술이 필요하다. 이를 위해 워터마킹 기술을 기반으로 사용자의 정보를 삽입하는 핑거프린팅이란 응용기술이 필요하다. 즉, 워터마킹 시스템을 이용하여 저작권자의 정보를 입력하는 점이 핑거프린팅의 특징이다[1]. 워터마킹에 의한 콘텐츠 보안은 매우 오래전부터 시도되어 왔다[2]~[4]. 워터마킹은 콘텐츠에 눈에 띄게 또는 은밀하게 표식을 하여 향후 그 콘텐츠의 소유가 누구인가를 나타내주는 방식이다. 워터마크를 삽입하는 방식에는 다양한 방식이 존재한다. 본 연구에서는 낮은 사양의 시스템에서 처리속도에 부담을 주지 않고 상기의 장단점을 보완하기 위해 공간영역에서 워터마크를 삽입하고자 한다. 워터마크는 사용자의 정보를 암호화하여 삽입하는 핑거프린트의 형태로 삽입되며 이때 노출 위험을 피하기 위해 사용자 ID를 블록암호화하여 의사 난수화하고 이를 토대로 워터마크의 삽

입 위치, 크기 및 삽입 프레임 번호에 변화를 주는 방법을 사용하였다. 또한 동영상의 특징상 많은 프레임이 존재하므로 이런 특징을 적극 활용하여 검출 결과를 안정화한 후 적분을 하여 검출 가능성을 높이는 시도를 하였다.

II. 관련 연구

1. 관련연구

워터마크를 삽입하는 방식에는 다양한 방식을 크게 나뉘보면 첫 째 공간 영역(spatial domain)에 삽입 방식, 둘째 변환 영역(transform domain)에 삽입 방식으로 나뉘 수 있다 [5]. 공간 영역에 삽입하는 경우는 영상에 직접 워터마크를 덧씌워서 삽입하는 경우이며[6]~[8] 변환 영역에 삽입하는 경우는 영상을 다른 차원으로 변환한 후 변환 차원에서 워터마크를 삽입하는 경우이다. 변환된 영역은 다시 세분화될 수 있는데 이산코사인변환(Discrete Cosine Transform: DCT)[9]~[12], 푸리에변환(Discrete Fourier Transform: DFT)[2],[13]~[16]과 웨이블릿(Discrete Wavelet Transform: DWT)[17]가 대표적이다. 공간 영역에 삽입하는 경우 다양한 워터마크를 삽입할 수 있지만 노출의 위험이 있고, 변환된 영역에 삽입하는 경우 은밀하지만 동영상의 경우 연산 부담이 늘어난다[3]. 특히 공간 영역에서 노출의 위험을 낮추기 위해 의사 난수(pseudo-random)으로 핑거프린트의 모습을 감추는 시도도 있어 왔다[1][13]. 동영상 시스템에서는 그 특징상 연산 시간이나 연산력이 많이

필요한 변환 영역에서의 워터마킹이나 핑거프린팅은 사용이 어렵다.

III. 본 론

1. 워터마크 시스템 개요

본 연구에서 개발된 워터마크 시스템(Watermark System)은 워터마크에 일정 정보를 삽입하는 방식이다. 다른 용어로 핑거프린트 시스템이라고도 하는데 일반적인 워터마크는 그림이나 마크 등을 삽입하는 방식이지만 핑거프린트는 사용자의 ID 또는 제품의 일련번호 등 원하는 정보를 특정한 형태로 변형하여 워터마크처럼 미디어에 삽입하는 방식이다. 본 연구에서는 워터마크를 2가지 형태로 삽입하게 된다. 즉 문자 형태와 기호 형태 두 가지로 삽입하는 것이다. 이에 대한 설명은 다음 절에서 하게 된다.

워터마크 시스템은 삽입 방식에 따라 크게 주파수 영역과 영상 영역으로 나뉘는데, 본 연구에서는 영상 영역 삽입 방식을 택하였다. 영상 영역 삽입 방식 중에는 DCT 계수에 삽입하는 방식과 Bitmap 영상에 삽입하는 방식이 존재하는데 두 방식 모두 장단점을 가지고 있다. DCT 계수 삽입 방식은 기하학적 공격에 취약한 반면 노출이 잘 되지 않는 장점이 있고 Bitmap 영상 삽입 방식은 기하학적 공격 등에 강한 반면 노출 가능성이 다소 높은 단점이 있다.

본 연구에서 채택한 방식은 저 사양 시스템에서 손쉽게 삽입할 수 있는 Bitmap 영상에 삽입하는 방식으로 Bitmap은 +1과 0으로만 구성되며 노출의 위험을 줄이기 위해 암호화된 위치와 패턴을 사용하여 삽입하고자 한다. 본 연구에서 제안된 워터마크 삽입 및 검출 시스템의 개요는 다음 그림 1 및 그림 2와 같다. 그림에서 나타낸 바와 같이 삽입되는 워터마크

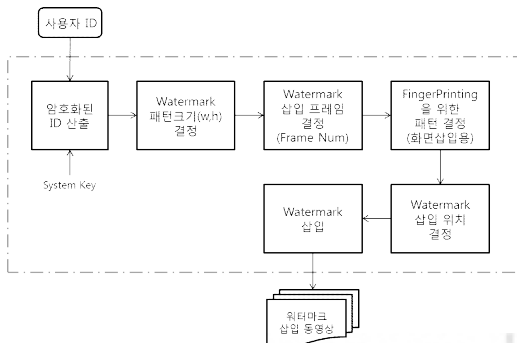


그림 1. 워터마크 삽입부 개요도
Fig.1. Watermark Insert System

크의 크기(가로x세로), 시기(프레임 번호) 및 위치(x, y) 등을 암호화된 사용자 아이디를 이용한 유사 난수 형태로 삽입함으로써 노출 가능성을 낮추는 방법을 개발하였다.

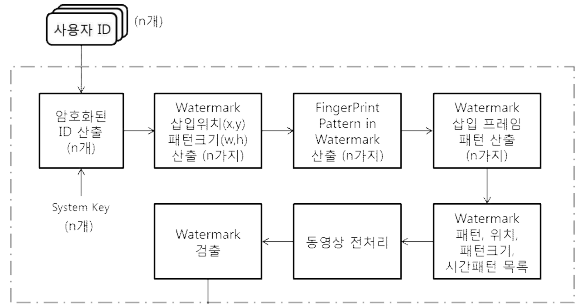


그림 2. 워터마크 검출부 개요도
Fig.2. Watermark Detect System

2. 패턴의 삽입

2.1 암호화된 ID 생성

사용자 아이디(user ID)는 다양한 형태로 존재할 수 있다. 영문, 한글, 숫자 또는 이들의 조합의 형태로 존재한다. 본 연구에서는 우선 영문과 숫자를 조합한 경우와 숫자만을 이용한 경우를 예제로 사용하고자 한다. 때에 따라서 더 길고 복잡한 ID를 사용할 수 있지만 실험의 편의성을 고려하여 영문 2자와 4비트 디지털 숫자 및 8비트의 디지털 숫자를 사용하였다. ID 암호화를 위해 사용된 알고리즘은 한국 인터넷 진흥원(KISA: Korean Internet & Security Agency)에서 개발하여 배포한 HIGHT(HIGH security and light weigHT) 블록 암호화 알고리즘으로 낮은 사양의 시스템 환경에서 사용하여 암호화하는데 적합하도록 개발된 경량 암호화 알고리즘이다. 본 연구에서는 이를 사용하여 사용자 ID를 64비트로 암호화하였다.

2.2 워터마크의 삽입 위치 및 패턴 크기의 결정

삽입된 패턴의 삽입 위치가 동일하면 삽입 위치가 노출될 가능성이 높기 때문에 위치와 패턴의 크기를 암호화하여 모든 미디어에 각각 다르게 적용하여야 한다. 이를 위해서 패턴의 삽입 위치와 삽입되는 패턴의 크기를 달리하였다. 삽입 위치 및 패턴 크기를 구하기 위해서는 암호화된 ID를 모듈로(Modulo) 연산으로 처리한다. 패턴의 삽입을 위해서 구하는 변수는 다음과 같다.

(1) 패턴의 크기 결정 : 삽입되는 패턴의 가로, 세로를 결

- 정한다. (width, height)
- (2) 프레임 번호 위치 : 패턴이 삽입될 프레임 간격을 결정한다. (Fr_num)
- (3) 프레임 내 위치 결정 : 삽입되는 패턴의 화면 내 위치를 정한다. (posX, posY)

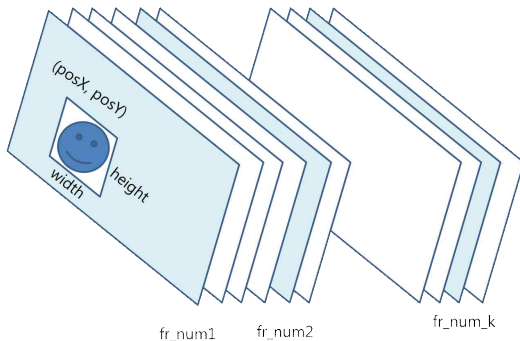


그림 3. 패턴 삽입 위치 및 간격 예제
Fig.3. An example of position and interval to insert patterns

2.3 패턴 삽입

패턴의 삽입은 비트맵(Bitmap) 형태로 삽입된다. 삽입될 비트맵은 0과 +1의 형태로 구성되며 미리 계산된 위치에 오버랩(overlap)되어 삽입된다. 삽입된 패턴의 예제는 다음 그림 4.와 같다. 그림에서 검은 부분은 0으로 이루어졌으며 흰 부분은 1로 이루어진 비트맵이다. 이 패턴은 다양한 형태로 변형이 가능하다. 영문자를 삽입할 수 있으며 한글의 삽입도 가능하다. 하지만 한글의 경우 복잡한 글자 모양이 존재하므로 조합체로 사용하기 보다는 낱글자의 형태가 더 높은 검출율을 나타내기 때문에 이를 권고하고자 한다. 디지털 숫자의 경우 숫자와 패턴을 동시에 사용하면 검출율을 크게 향상시킬 수 있다. 패턴의 형태는 다양한 변형이 가능하다. 본 연구에서는 '0'의 경우 'ㄷ'의 형태로 '1'의 경우는 대칭 형태로 삽입하였다. 이 패턴의 경우 공모 공격과 평균화 공격 및 중간 값 공격에 내성을 갖도록 고려하였다. 패턴은 암호화된 ID로부터 얻어진 width × height의 크기로 만들어지는데, 이 크기는 동일하게 삽입되지 않고 ID에 따라 의사난수(Pseudo-random)에 의해 변형을 줄 수 있다. 이는 암호화된 ID가 나타내는 길이에 따라 달라지겠지만 64비트의 경우 8가지 128비트의 경우 16가지인 바이트 단위로 크기를 달리 할 수 있게 된다. 또한 화면 내에 삽입되는 위치도 암호화된 ID를 모듈로(modulo) 연산으로 계산하여 의사난수에 의한 위치를 얻어낼 수 있다. 삽입되는 프레임 간격도 이 방식

에 따라 변화하게 설계된다. 의사난수는 설계자는 그 위치를 알 수 있지만 암호화된 ID를 갖지 못하는 공격자에게는 실질적인 난수(real random)이므로 알아내기 매우 어려운 방식으로 CDMA 등에서 많이 사용되는 방식이다.



그림 4. 삽입 패턴 예제 (흑색:0, 백색:1)
Fig.4. An example of pattern to insert (black: 0, while: 1)

3. 워터마크 검출

패턴 검출 부분은 워터마크 시스템에서 가장 핵심이 되며 구현이 어려운 부분이다. 워터마크가 삽입된 영상에 공격이 가해지지 않는 경우는 매우 깨끗하게 패턴을 검출할 수 있지만 공격이 가해지면 패턴이 훼손되기 때문에 검출이 어렵게 되는 것이다. 본 연구에서는 이러한 문제에 대처하기 위해 패턴 후보의 추출, 전처리, 검출 단계에서 성능을 높이기 위한 노력을 하였다. 패턴의 후보를 추출하기 위해서 원본 영상을 대조하는 방법이 존재하는데 본 연구에서는 자가 영상에서 패턴 후보를 추출하고 추출된 패턴에서 훼손될 경우 잡음을 줄이기 위해 전처리(preprocessing)를 실시하였다. 끝으로 패턴 후보 검출을 위해 상관계수(correlation coefficient) r 연산을 사용하였으며 데이터의 집중도를 높이기 위해 150여 후보를 축적하여 사용하였다. 다음은 워터마크 검출 과정에 대한 구성도이다.

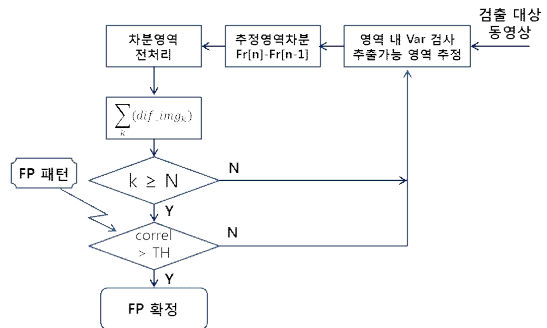


그림 5. 워터마크 검출 구성도
Fig.5. Block Diagram of Watermark Detection

3.1 패턴이 존재할 위치 추정

본 연구에서 개발된 시스템은 사전에 규정된 사용자 ID가 존재하고 이것을 시스템 Key에 의해 암호화되었다는 전제조건이 필요하다. 그러므로 검출 시스템은 이들 ID를 보유하고 있고 시스템 Key를 가지고 있다. 시스템은 우선 여러 ID를 시스템 Key를 사용하여 암호화하여 (1) 패턴의 가로 세로 크기 width, height를 산출하고 (2) 삽입되었을 프레임 간격 Fr_num을 산출하며 (3) 화면 내 위치 posX, posY를 산출하여 각각 구조체에 저장하여 가지고 있게 된다. 이들 구조체를 이용하여 검출할 프레임 번호와 위치 및 추출 영역 크기를 미리 정하여 검출에 사용된다.

3.2 패턴 후보 추출

패턴은 영상의 상태와 상관없이 삽입된다. 프레임 내에서 패턴 후보가 가장 선명하게 추출될 가능성이 높은 영역은 영역 내의 변화가 아주 적은 부분이다. 즉 원하는 영역의 $Var < \epsilon$ ($\epsilon \approx 0$) 이라는 조건을 만족하는 부분을 찾아내야 한다. 앞에서 서술한 3.1 절에서 구한 프레임 간격 Fr_num과 화면 내 위치 및 영역 정보를 이용하여 삽입 가능성이 있는 프레임에 대해 해당 영역에 대해 분산을 구한다. 분산은 식 (1)과 같다.

$$Var = \frac{\sum_{i=영역} (Fr[k] - Fr[k-1])^2}{M_{영역}} \quad (M=영역\ 평균) \quad (1)$$

이때 $Var < TH$ (TH : 실험에 의한 문턱값) 이면 그 영역에서 패턴을 추출할 가능성이 높은 부분이므로 이때 이 영역에 대해 앞 프레임과 현재 프레임 간의 차분 영상을 패턴 후보로 정한다.

3.3 전처리 과정

패턴 후보를 추출한 후 이 영역에 대해 전처리 과정을 실시하는데 다음과 같은 순서에 의해 처리한다.

- (1) if dif_img ≤ 0, then dif_img = 0
- (2) if dif_img ≥ 1, then dif_img = 1
- (3) Median Filtering

이때 미디언 필터링은 잡음을 없애주는 역할도 하지만 신

호의 밀도를 높여서 안정화시켜주는 역할도 하게 된다. 이때 필터링은 연산량이 매우 많아 연산속도를 현격히 저하시킨다. 본 연구에서 사용된 미디언 필터는 11 × 11 에서 17 × 17 등의 윈도우를 사용하는데 이 경우 1 픽셀을 필터링하기 위해서 121개 ~ 289개의 픽셀을 내림차순 연산하여 중간 값을 골라내는 많은 연산이 필요하게 된다. 이렇게 되면 고화질 영상의 경우 매우 많은 연산이 필요하므로 검출시간이 오래 걸리는 단점을 지닌다. 그러므로 본 연구에서는 본 시스템에 특화하여 다음과 같은 단순한 방식으로 미디언 필터의 시간을 줄일 수 있었다.

- (1) 모든 픽셀은 0 또는 1의 값을 갖는다
- (2) 윈도우 가로를 W, 세로를 H라 할때 윈도우 내의 픽셀 값의 합은 W×H가 된다
- (3) 내림차순으로 정렬하여 중간 값을 구하면 0 또는 1이 될 것이다
- (4) TH = ROUND(W×H/2) (이때, ROUND는 반올림)
- (5) if $\sum(pixel) < TH$, dif_img = 0
- (6) if $\sum(pixel) \geq TH$, dif_img = 1

이 연산은 내림차순 연산이 아니라 단순한 합산만 하므로 연산속도가 빠르다. 본 연구에서는 다음과 같은 합산 알고리즘을 한 번 더 적용하여 연산속도를 훨씬 빠르게 할 수 있었다.

- (1) 최초 블록에 대해 $\sum(pixel)$ 을 구한다
- (2) 오른쪽 1 픽셀에 대한 연산에 대해서는 이전 윈도우의 좌측 끝 y방향 1 line 합산을 빼고 새로운 윈도우 우측 끝 y방향 1 line 합산을 더해준다
- (3) x 방향 1 line에 대해 연산이 끝난 후 최초 블록을 1 픽셀 아래로 위치시키는데 이때 이전 블록의 최상단 x방향 1 line에 대한 합을 빼고, 새 블록 최하단 x방향 1 line 합을 더해준다

위의 두 알고리즘을 적용하면 연산 단계를 상당 부분 줄일 수 있게 된다.

이 단계에서 얻어진 패턴 후보 영역 영상을 150~200 프레임 합산 영상을 얻어내어 다음 단계인 상관계수 연산식에 의한 패턴 매칭을 실시한다.

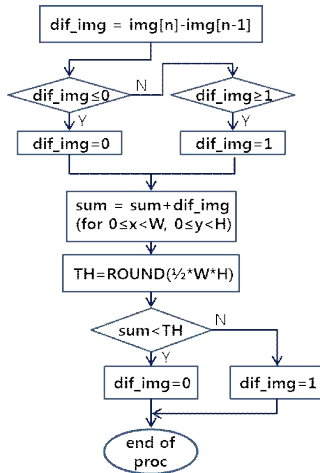


그림 6. 전처리 과정
Fig.6. Preprocessing

3.4 상관계수 r 연산

상관계수(correlation coefficient) r의 연산은 후보 패턴 중 글자와 기호에 대해 수행한다. 이는 상호 검출 결과의 동일성을 검증하기 위해 수행된다. 즉 상단의 숫자와 하단의 기호를 각각 연산하여 얻은 결과를 교차 확인하는 절차로 수행된다. 본 연구에서 사용된 상관계수 r에 대한 식은 다음 식(2)와 같다.

$$r = \sqrt{\frac{\sum_{k=0}^{n-1} (x_k - m)^2}{n} - m^2} \quad (\because m = avg(x_k)) \quad (2)$$

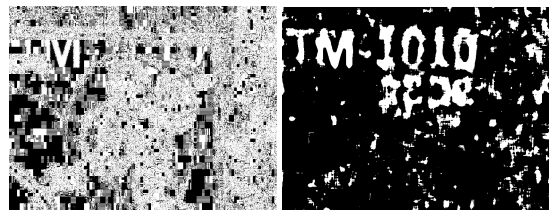
r > TH (TH : 실험에 의한 문턱치) 인 경우 원하는 숫자로 결정하고 그렇지 않으면 미확정 글자로 처리한다. 본 시스템은 동영상의 길이가 긴 경우 많은 데이터를 확보할 수 있기 때문에 검출율을 증대시킬 수 있게 된다.

4. 결과 및 고찰

실험은 지상파 TV방송을 TS 형태로 캡처받은 동영상을 이용하여 실시하였다. 동영상은 드라마, 토크쇼 등을 대상으로 하였으며 드라마의 경우 움직임이 적은 경우, 중간 정도의 움직임이 있는 경우와 움직임이 큰 경우로 나누어 실험했으며 여행 다큐멘터리는 화면 변화가 매우 큰 경우가 많다.

4.1 후보 패턴 추출 및 전처리 결과

패턴이 삽입되었을 것으로 추정되는 프레임과 1 장 이전 프레임을 단순히 차분한 결과와 미디언 필터링 한 결과는 다음 그림과 같다. 그림 7. (가)는 차분한 영상에서 1 이상을 모두 1로 제한한 영상으로 편의상 0을 검은색, 1을 흰색으로 표시하였다. 그림 7. (나)는 이것을 미디언 필터링하여 전처리한 결과로 마찬가지로 0을 검은색, 1을 흰색으로 표시하였다. 결과에서 볼 수 있듯이 패턴이 들어가 있는 부분은 미디언 필터링하면 남게 되고 잡음 형태의 부분은 사라지게 된다. 패턴이 비교적 잘 추출되었고 영상의 변화에 따른 차분도 일부 나타나있다.

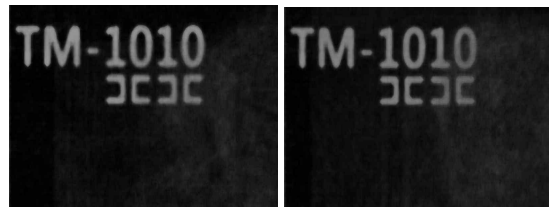


(가) 차분한 영상 (검은색:0, 흰색:1) (나) (가)의 전처리 영상 (검은색:0, 흰색:1)

그림 7. 앞 장면과 뒷 장면을 차분 한 후 전처리한 영상 결과
Fig.7. Results of median filtering after pre and current image difference

4.2 차분 영상을 150장 축적 후 결과

다음 그림은 상기 4.1절에서 얻어진 차분 후 전처리한 영상을 150장 축적한 결과이다. 150장을 축적하였기 때문에 영상에 큰 수를 곱하지 않아도 데이터가 상존하는 부분은 뚜렷하게 그렇지 않은 부분은 흐릿하게 표현되었다.



(가) 움직임 중간 동영상 검출 결과 (나) 움직임 많은 동영상 검출 결과

그림 8. 차분 후 전처리한 결과를 150장 축적한 결과
Fig.8. Accumulating results of 150 frames after preprocessing the differentiated image

본 연구에서는 시각적인 판단을 위해 150장 축적 결과를 영상으로 출력하여 나타내었다. 결과가 좋아서 시각적으로도 판단이 용이할 정도로 뚜렷한 결과를 얻을 수 있었다. 이 결

과는 삽입된 영상 중 변화량이 문턱치를 넘지 않는 영상을 대상으로 150장 축적하여 얻어진 결과이다. 따라서 변화량이 많은 영상의 경우 150장을 축적하기 위해서는 변화량이 적은 영상보다 많은 데이터를 필요로 한다. 움직임이 중간 정도의 경우 2분 영상 (3600 frame)에서 280 프레임만을 얻을 수 있었다.

4.3 동일한 동영상에서 다른 형태의 워터마크 삽입 및 검출 결과

본 연구에서는 동일한 동영상에서 다른 형태의 워터마크를 삽입하고 검출하여 유사한 결과를 얻을 수 있었다. 이 실험을 통해 다양한 워터마크의 삽입 및 검출이 가능한 결과를 얻을 수 있었고 삽입 정보의 다양화를 기할 수 있었다. 다음 그림은 드라마 중 몸싸움 장면으로 움직임이 많은 동영상에서의 결과를 보여 주고 있다. 이때 심하게 움직이는 장면이 계속되지 않기 때문에 검출 결과에 심각하게 영향을 주지 않는다.



(가) 'TM-1010' 삽입 후 검출결과 (나) '10110100' 삽입 후 검출 결과

그림 9. 움직임이 많은 동영상에서의 삽입/검출 결과 (움직인 대상도 흐릿하게 검출됨)

Fig.9. Results of insertion/detection for the video which object is moving very much (the silhouette is the moving object)

4.4 연산 시간 결과

본 실험에서 개발된 시스템에서 삽입 시간은 0.219 ± 0.094 (mS)로 실시간 연산에 적합한 연산시간이 소요되었다. 하지만 워터마크 검출은 사후의 문제이므로 실시간 검출이 불가능하여도 무방하다. 그러나 그 시간을 줄일 수 있는 알고리즘이 개발되어 119.33초 짜리 동영상의 처리 경우 870.58 ± 99.21 (초)로 실시간의 약 7.30배 오래 걸렸다. 이때 시간의 워터마크 후보를 검사하는 시간은 약 87.87 ± 7.73 (mS) 이고 전체 처리 시간은 약 394.00 ± 23.82 (mS) 였다.

4.5 워터마크 검출 결과

총 22개 동영상 실험 중 1개에서 검출 실패를 하였고 21개 동영상에서 검출에 성공하여 95.45%의 성공률을 나타내었다. 이 중 1개의 동영상은 화면전환이 매우 큰 해수욕 및 선상 파티 장면으로 다른 실험과 동일하게 검출 길이를 3580 프레임으로 제한하였기 때문에 실패한 것으로 판단된다. 즉 모든 동영상이 처음부터 끝까지 화면 변화가 많은 경우는 드물기 때문에 검출 확률이 높아질 것으로 판단된다.

표 1. 동영상 특성에 따른 검출 결과
Table 1. Results of detection according to features of videos

동영상 특성	검출율	비고
움직임 많음 (8 cases)	7 검출 (87.5%)	zoom & dancing
움직임 중간 (7 cases)	7 검출 (100%)	action
움직임 적음 (7 cases)	7 검출 (100%)	talk show

4.6 결과 고찰

본 논문에서 제시한 워터마크의 삽입 및 검출은 제한된 조건하에서도 높은 검출율을 얻을 수 있었다. 또한 검출 결과를 시각적으로 보여주기 위한 결과에서도 비교적 선명한 결과를 얻을 수 있었다. 상관계수 연산에 의한 결과도 움직임이 많은 경우 0.87, 중간인 경우 0.79, 움직임 적은 경우 0.84라는 높은 결과를 얻을 수 있어서 0.7 이상인 경우 확정적인 것을 참고하면 높은 검출 결과를 얻게 되었다. 워터마크 기호 영상 추출 단계에서 성공율 95.45%의 결과를 얻었으며 이 결과 영상에 상관계수 r 연산을 적용하면 99% 이상 검출 결과를 얻을 수 있었다.

본 연구는 향후 다양한 공격에 대비한 알고리즘의 보안을 통해 상품화의 단계를 지향하고 있으며 이를 통해 영상 관계 시스템과 지적 재산권 보호 시스템에 적용할 예정이다.

IV. 결 론

본 연구에서는 안드로이드와 같은 저 사양 임베디드 시스템에서 동영상을 대상으로 사용자 ID에 대한 일정한 정보를 삽입하고 검출하는 워터마크 응용 시스템을 개발하여 다음과 같은 결과를 얻을 수 있었다.

- (1) 본 연구에서 적용한 비트맵 오버랩 워터마크 기술을 이용

하면 삽입 시간이 0.219 ± 0.094 (mS)로 실시간 삽입 연산이 가능하여 저 사양 시스템에 적합하였다.

- (2) 삽입 시간과 위치를 사용자 ID를 암호화하여 서로 다르게 삽입함으로써 예측 가능성을 낮추었다.
- (3) 검출 시에 원본 동영상의 대조 없이 자체 영상을 이용하여 추출함으로써 혹시 있을 기하학적 공격에 대비할 수 있었다.
- (4) 차분 영상에 대해 0과 +1로 제한한 후 중간 값 필터링을 사용하여 검출 결과를 안정화시킬 수 있었다.
- (5) 중간 값 필터링의 속도를 높이기 위해 내림차순을 사용하지 않고 본 연구에서 개발한 좌우 한 줄 연산에 의해 연산 시간을 대폭 줄일 수 있었다.
- (6) 검출 결과를 적분함으로써 상관계수 r 연산 값 문턱치를 크게 상회하는 결과를 얻을 수 있었다.

본 연구의 결과는 상관계수 연산에 의한 패턴 인식으로도 결과를 얻을 수 있을 뿐만 아니라 영상으로 출력함으로써 시각에 의해 직관적인 검출도 가능하도록 하고 있다. 향후 본 연구를 바탕으로 다양한 공격에 대비한 실험을 통해 시스템 안정화를 이루어 영상관계 시스템에 탑재하기 위한 연구를 계속할 예정이다.

감사의 글

본 연구는 2011학년도 인덕대학교 학술연구비 일부 지원에 의하여 이루어진 연구로서, 관계부처에 감사드립니다.

참고문헌

- [1] A. Kejariwal, "Watermarking," IEEE POTENTIALS, pp.37-40, Oct/Nov. 2003
- [2] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," IEEE Trans. on IMAGE PROCESSING, Vol. 6, No. 12, pp.1673-1687, Dec. 1997
- [3] I.J. Cox, M.L. Miller and J.A. Bloom, Digital Watermarking, Morgan Kaufmann Publishers, 2002
- [4] I.J. Cox, T. Kalker, H. Lee et. al, Digital watermarking: third international workshop, IWDW 2004, Seoul, Korea, Oct. 30 - Nov. 1, 2004 : revised selected papers, 2004
- [5] Sarabjeet Singh, "Digital Watermarking Trends," International Journal of Research in Computer Science, Vol.1, No.1, pp.55-61, Sep. 2011
- [6] W. Lu, H. Lu, F. Chung, "Feature based watermarking using watermark template match," Applied Mathematics and Computation, Vol.177, pp.377-386, June 2006
- [7] A. Giannoula, N. V. Boulgouris, D. Hatzinakos, and K. N. Plataniotis, "Watermark Detection for Noisy Interpolated Images," IEEE Trans. on Circuits and Systems-II: Express Briefs, Vol.53, No.5, pp.359-363, May 2006
- [8] A.Z. Tirkel, C.F. Osborne, T.E. Hall, "Image and watermark registration," Signal Processing, Vol.66, pp.373-383, May 1998
- [9] Jiwu Huang, Yun Q. Shi, and Yi Shi, "Embedding Image Watermarks in DC Components," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 10, No. 6, pp.974-979, Sep. 2000
- [10] Seon-Cheol Hwang, "Coded Watermark Detecting Algorithm using a Blind Detecting Method Without Original Videos," The Transactions of the Korean Institute of Electrical Engineers. P, Vol.59P No.4, pp. 395-400, Dec. 2010
- [11] D. Tsolis, S. Nikolopoulos, L. Drossos, S. Sioutas, T. Papatheodorou, "Applying robust multibit watermarks to digital images," Journal of Computational and Applied Mathematics, Vol.227, pp.213-220, May 2009
- [12] C. Hsu and J. Wu, "Hidden Digital Watermarks in Images," IEEE Trans. on IMAGE PROCESSING, Vol. 8, No. 1, pp.58-68, Jan. 1999
- [13] X. Guihua and Y. Shenglin, "Spread Spectrum Watermark Detection in DRT-Domain," Journal of Electronics, Vol.24, No.6, pp.782-786, Nov. 2007
- [14] P. Kumar Dhar and I. Echizen, "Robust FFT Based Watermarking Scheme for Copyright Protection of Digital Audio Data," 2011 Seventh International Conference on Intelligent

- Information Hiding and Multimedia Signal Processing, pp. 181-184, Oct. 14-16, 2011
- [15] S. Pereira and T. Pun, "Robust Template Matching for Affine Resistant Image Watermarks," IEEE Trans. on IMAGE PROCESSING, Vol. 9, No. 6, pp.1123-1129, Jun. 2000
- [16] D. Kirovski, H. Malvar, and Y. Yacobi, "A Dual Watermark-Fingerprint System," IEEE Multimedia, pp.59-73, July-September 2004
- [17] M. Hsieh, D. Tseng, and Y. Huang, "Hiding Digital Watermarks Using Multiresolution Wavelet Transform," IEEE Trans. on INDUSTRIAL ELECTRONICS, Vol. 48, No. 5, pp.875-882, Oct. 2001
- [18] Christine I. Podilchuk and Edward J. Delp, "Digital Watermarking: Algorithms and Applications," IEEE Signal Processing Magazine, pp.33-46, July 2001
- [19] F. Hartung, and M. Kutter, "Multimedia Watermarking Techniques," Proceedings of the IEEE, Vol. 87, No. 7, pp.1079-1107, Jul. 1999
- [20] M. Hasnaoui, M. Mitrea, "Multi-symbol QIM video watermarking," Signal Processing: Image Communication, Vol.29, pp.107-127, Jan. 2014

저 자 소 개



황 선 철

1986: 연세대학교
전기공학과 공학사.

1988: 연세대학교
전기공학과 공학석사.

1998: LG전자
선임연구원.

1999: 연세대학교
전기전자공학과 공학박사

현 재: 인덕대학교
방송영상미디어과 교수

관심분야: 영상처리

Email : sthwang@induk.ac.kr