

비디오 워터마크 삽입 및 검출 시스템 구현

권오준, 권순각
동의대학교 컴퓨터·소프트웨어공학부

An Implementation of Video Watermark Embedder and Detector

Oh-Jun Kwon, Soon-Kak Kwon
Division of Computer·Software Engineering, Dongeui University
E-mail : {ojkwon, skkwon}@dongeui.ac.kr

요 약

본 논문에서는 인터넷 기반 비디오 워터마킹 시스템에 대한 구현기법을 제시한다. 인터넷 기반 VOD 서비스등의 응용분야를 위하여 TCP/IP 기반으로 WinSock을 이용하여 압축정보를 전송, 수신하도록 구현하였으며, 또한 MPEG-4 동영상 부호화 알고리즘에 의하여 부호화된 압축영역에서 워터마크를 삽입하고, 검출하는 시스템을 구현하였다.

1. 서론

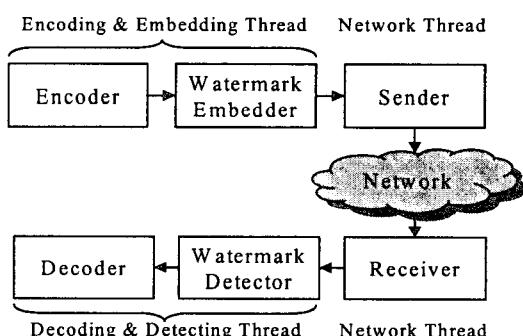
비디오 워터마킹은 이미지 워터마킹보다 워터마크 정보를 넣을 수 있는 비트의 수가 많다는 장점을 가지고 있지만 실시간 처리가 요구된다. 그리고 워터마크 삽입영역에 따라 비압축영역의 워터마킹과 압축영역의 워터마킹[1,2,3,4,5]으로 나눌 수 있다. 현재 이미지 워터마킹에서 사용되는 기법들이 조금 변형된 상태에서 비디오 워터마킹에 응용되고 있으며, 주로 압축영역에서의 워터마킹이 사용되고 있다.

본 논문에서는 비디오 워터마킹 기법중에서 성능이 우수한 DEW (Differential Energy Watermarking) 방법[2]을 MoMuSys에서 제공된 MPEG-4 비디오 codec에 적용하여 구현하였다.

2. 시스템 구성도

2.1 전체 구성도

전체 구성은 그림1과 같이 크게 3부분으로 나누어질 수 있다. 부호화와 워터마크 삽입을 맡고 있는 Encoding & Embedding Thread, 워터마크가 삽입된 비트열의 전송과 수신을 맡고 있는 Network Thread, 수신된 비트열에서 워터마크를 검출해서 복호화하는 Decoding & Detecting Thread로 구성된다.



[그림1] 전체시스템 구성도

2.2 구현환경

전체 시스템은 기존의 표준화된 부호화기로 압축된 비트열이 저장된 서버와 복호기가 설치된 클라이언트로 이루어진다. 인터넷 네트워크 환경을 위해서 게이트웨이에 연결된 로컬 환경에서 실험을 수행한다.

3. 구현 방법

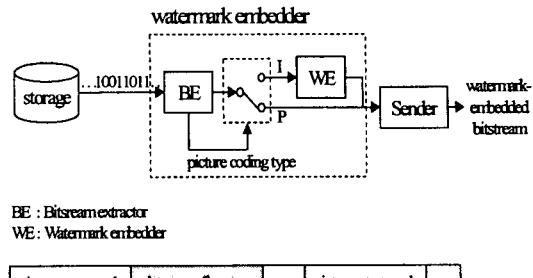
3.1 비디오 워터마크 삽입

비디오 워터마크의 삽입은 압축된 비트열을 읽어서 워터마크 라벨 정보를 삽입하는 방식과, 부호화 과정에서 직접 워터마크를 삽입하는 방식으로 나누어 생각해 볼 수 있다.

첫번째 방식의 경우에는 미리 부호화되어 저장된 비트열에서 화면내부호화된 영상(I-vop)에 대해서만 비디오 워터마크를 삽입한다. 그림 2의 비트열 추출기(BE)에서는 저장된 비트열을 프레임 단위로 불러들여 어떻게 부호화되었는지를 파악한다. 비트열의 끝에 헤더를 검색하여 화면내부호화된 비트열에 대해서는 워터마크를 삽입하고 화면간 부호화(P-vop)된 비트열은 건드리지 않는다. 이 방식은 I-vop 내에 삽입된 워터마크 신호에 의해 발생된 열화가 I-vop를 기준 vop로 이용하는 P-vop와 B-vop에 전파되는 drift 문제가 발생한다.

두번째 방식은 비디오 부호화 과정에서 직접 I-vop마다 워터마크 정보를 삽입하는 방식이다. 본 구현 시스템에서는 drift 문제를 방지하기 위해서 이 방식을 이용하였으며, 이를 위해 MoMuSys 인코더 프로그램의 VopCodeShapeTextureIntraCom() 내에 CodeMBBeforeWE(), Embedding()와 CodeMBBeforeMB() 함수를 삽입하였다. 우선, CodeMBBeforeWE()는 MB 단위로 수행되며 워터마크 삽입을 위해 DCT, 양자화, 그리고 역양자화된 MB 신호를 picture에 복사하는 기능을 수행한다. Embedding() 함수는 DEW 알고리즘에 따라서 I-vop의 DCT 영역에 워터마크를 삽입하는 함수이다. 그리고, CodeMBAfterWE() 함수는 워터마크가 삽입된 picture에 대해 VLC를 위한 MB 단위의 양자화와 복호를 위한 MB 단위의 IDCT를 수행하는

함수이다.



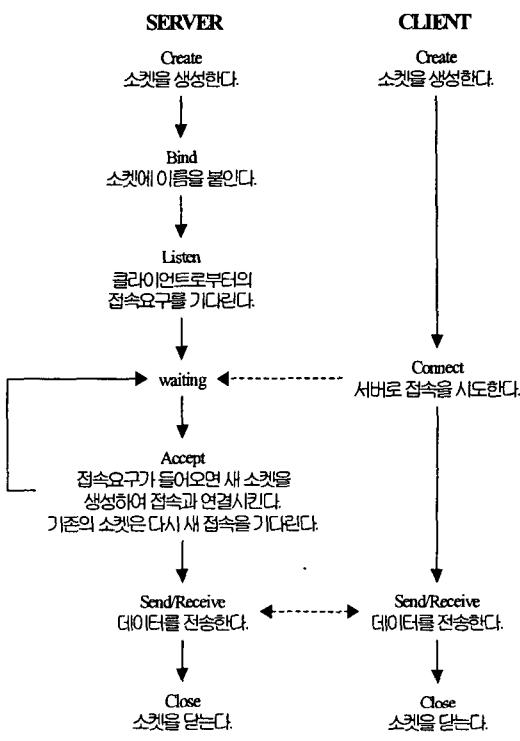
[그림 2] 비디오 워터마크 삽입기

3.2 네트워킹

인터넷상의 VOD 서비스는 항상 서비스를 제공하는 서버(server)측과 이를 요청하는 클라이언트(client)측으로 구성되는 클라이언트/서버 모형으로 동작한다. 이와 같이 인터넷을 이용하는 클라이언트/서버 환경에서 시간적 제약 조건이 비교적 느슨한 멀티미디어 정보를 전달하기 위한 통신 프로토콜로 실시간 스트리밍 프로토콜(RTSP:Real-Time Streaming Protocol)이 있지만 구현상의 편의를 위해 WinSock API를 이용한다.

Winsock은 유닉스 시절의 socket을 윈도우즈 운영체제에 맞도록 변형된 것이다. Winsock API (Application Programming Interface)는 TCP/IP 기반으로 소프트웨어 개발자로 하여금 하드웨어에 독립적인 인터페이스를 제공한다. Winsock 이전에는 TCP/IP stack vendor의 제품에 따라 소프트웨어 개발자가 일일이 인터페이스를 생각해주어야 하는 불편함이 있었지만 Winsock에 의해 이러한 불편함이 제거되었다.

일반적인 WinSock의 흐름도는 그림 3과 같다. 먼저 서버가 소켓을 생성하여 listen 상태에 들어가서 클라이언트의 접속 시도를 기다린다. 클라이언트가 서버 쪽으로 접속을 원할 때는 역시 소켓을 생성하고 접속시도를 한다. 제대로 접속이 이루어지는 경우 서버는 accept를 하게 되고 이 순간 새로운 소켓을 생성하여 현재 요구되고 있는 접속과 연결시켜 준다. 최초로 listen을 하던 소켓은 계속 listen 상



태에 있어 또 다른 접속 요구에 응답하게 된다. 서버에는 listen을 위한 소켓과 실제 통신을 하는 소켓이 따로 있음에 유의하라. 여기서 서버와 클라이언트는 기능적인 의미에서 나누어진 것으로 두개의 동일한 단말의 경우 어느 쪽이나 서버가 될 수 있다. 다시 말해 접속의 중심이 되는 쪽이 서버일 뿐이다.

네트워크 계층에서 Winsock을 이용한 통신은 UDP를 기반으로 하였고 고정 크기로 전송과 수신을 하게 하였다. 전송 부분과 수신 부분에서의 입력, 출력간의 rate 차이가 발생하기 때문에 버퍼를 두어 underflow/overflow의 문제점을 완화시킨다.

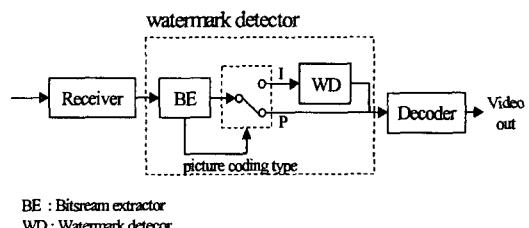
구현된 시스템에서 클라이언트는 복호기의 역할을 수행하며, 수신 받고자 하는 비디오 데이터를 선택하여 서버에 전송한 후 수신대기 상태에 들어간다. 클라이언트의 함수 중에서 OnReceive() 함수는 서버에서 PACKET_BUFFER_SIZE(8kB)의 데이터를 전송해 오면 이를 수신하여 네트워크 버퍼에 저장하는 기능을 수행한다. 그리고, BitstreamFillBuffer() 함수

는 네트워크 버퍼에서 디코더 버퍼로 BFR_SIZE(2kB)의 bitstream을 읽어오는 기능을 수행한다. 이때 네트워크 버퍼가 BFR_SIZE 만큼 채워져 있지 않다면 서버에 신호를 보내어 전송하게 하고 waiting 한다.

서버는 클라이언트에서 선택된 비디오 데이터의 bitstream을 전송하는 역할을 담당한다. OnReceive() 함수가 클라이언트에서 보내온 신호를 해석하여 다음 동작을 수행하게 되는데, 클라이언트에서 video source name을 받으면 전송을 시작한다. SendingSomething() 함수는 ACK 신호를 받을 때마다 비디오 데이터를 PACKET_BUFFER_SIZE로 전송한다.

3.3 비디오 복호 및 워터마크 검출

네트워크 계층에서 수신한 워터마크가 삽입된 비트열은 우선 비트열 검출기에서 어떻게 부호화되었는지 파악이 되어야 한다. 일단 화면내부호화된 것으로 결정이 나면 영상 하나에 해당하는 비트열이 수신될 때까지 버퍼링하여야 한다. 이 비트열에서 워터마크를 검출하고 나면 복호화기를 거쳐 영상을 복원한다. 반면에 화면간부호화로 결정이 된 비트열에 대해서는 추가적인 버퍼링의 과정 없이 실시간으로 바로 복호화하여 영상을 출력시킨다.



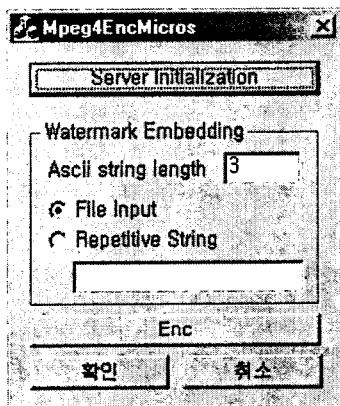
[그림 4] 비디오 복호기 및 워터마크 검출기

워터마크 검출을 수행하는 Detection() 함수는 MoMuSys 디코더 프로그램 중 I-vop마다 picture 단위로 수행되는 DecodeVopCombinedShapeTextureIntra() 함수 내에 들어있다. GetMBblockdata() 함수를 이용하여 워터마크 검출을 위해 MB단위로 역 양자화된 계수를 picture에 복사하고, 하나의 I-vop에 대한 계수가 모두 구성되면 Detection() 함수에서 picture 단위로 워터마크를 추출한다.

4. 구현결과

인터넷상의 VOD 서비스 환경을 구축하기 위해 학내망에 연결된 PC 2대를 서버와 클라이언트로 구분하였다. 서버에 저장된 비디오 신호에 대해 I-vop마다 워터마크를 차별적으로 삽입하여 winsock API를 통해 네트워크 계층으로 전송한다. 전송과 수신은 TCP/IP 스택을 통해 이루어지며 클라이언트는 영상 단위로 워터마크를 검출하여 요청한 영상을 재생한다.

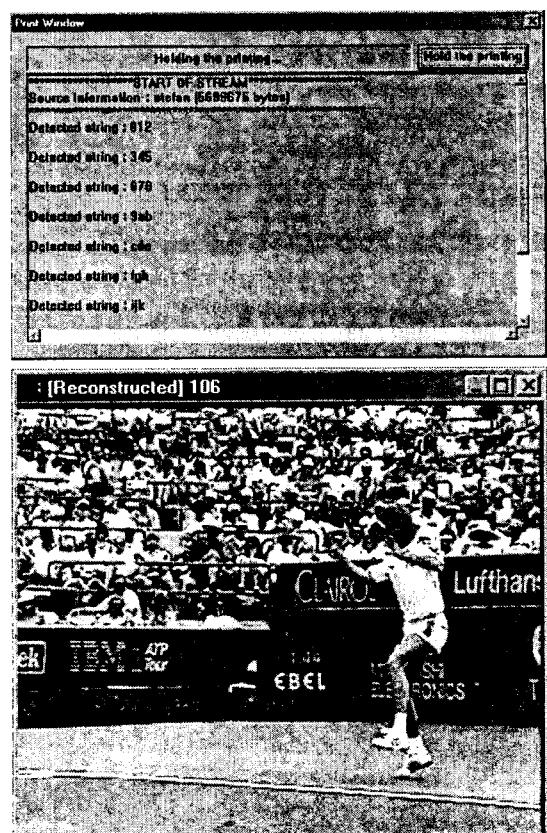
인코더 및 디코데에 대한 유저 인터페이스 화면은 다음 그림과 같다.



[그림 5] 인코더 유저인터페이스 화면

[참 고 문 헌]

- [1] F.Hartung and B.Girod, Watermarking of uncompressed and compressed video, Signal Processing, vol.66, no.3, pp. 283-301, 1998.
- [2] G.C.Langelaar, R.L.Lagendijk and J.Biemond, Real-time labeling of MPEG-2 compressed video, Journal of Visual Communication and Image Representation, vol.9, no.4, pp. 256-270, Dec., 1998.
- [3] F.Hartung and B.Girod, Digital watermarking of raw and compressed video, Proc. SPIE Digital Compression Technologies and Systems for Video Commun., vol. 2952, Oct. 1996, pp. 206-213.



[그림 6] 디코더 화면

- [4] J.-P. Linnartz. (1998). MPEG PTY marking. [Online]. Available [WWW:](http://diva.eecs.berkeley.edu/linnartz/pty).
- [5] F.Jordan, M.Kutter and T.Ebrahimi, Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video, ISO/IEC Doc. JTC1/SC29/WG11 MPEG97/M2281, July, 1997