H.264 트랜스코딩과 비트스트림의 선택적 암호화 구현

이성연*, 조경연*, 김종남*
*부경대학교 전자컴퓨터정보통신공학부
e-mail : sylee9997@pknu.ac.kr

Implementation of H.264 Transcoding & Selective Encryption of bit stream

Seong-Yeon Lee*, Gyeong-Yeon Cho*, Jong-Nam Kim*
*Dept. of Computer Multimedia Engineering, PuKyong Nat'l Univ.

요 약

IPTV, VOD와 같은 스트리밍 서비스와 유료 케이블 TV 방송 채널, 유료 위성방송 채널 등에는 반드시 제한 수신 시스템(Conditional Access System, CAS)이 필요하다. CAS시스템은 인증 받은 사용자에게는 깨끗한 화면을 보여주어야 하고 인증 받지 않은 사용자에게는 정상적으로 화면을 즐길 수 없도록 하여야 한다. 이러한 환경을 만들기 위하여 추가비용이 적은 알고리즘이 필요한데 그 방법으로 암호화를 제안한다. 본 논문에서는 CAS 등의 인증시스템을 위하여 H.264 영상의 선택적 암호화를 구현하였다. 제안하는 방법은 여러 가지 포맷으로 된 콘텐츠를 트랜스코딩하여 YUV 형식으로 변환한 뒤, 이것을 H.264 코덱을 이용하여 압축한 다음 필요에 따라 특정한 부분을 암호화하는 것이다. 실험을 통하여 암호화 키가 없는 미 인증 사용자는 영상을 제대로 볼 수 없었고 암호화의 강도를 강하게 할 경우 재생은 되지만 영상의 확인이 불가능함을 확인하였다. 또한 300프레임의 영상을 암호화 하는데 평균 71.3초가 걸려 속도 역시 빠름을 확인하였다. 제안하는 내용은 IPTV, VOD와 같은 스트리밍 서비스에서의 사용자 인증 및 저작권 보호 등의 분야에 유용하게 사용될 것이다.

1. 서론

IPTV의 전국 실시와 유료 CATV 채널의 증가로 인하여 제한 수신 시스템(Conditional Access System, CAS)의 필요성이 늘어나고 있다. 현재의 CAS 시스템은 플래시 메모리 또는 스마트카드 등에 사용자 등록 정보를 저장하고 채널의 시청 가능 여부를 나누게 된다[1]. 이러한 방법은 추가 하드웨어가 필요하므로 비용에 있어서 불리할 뿐 아니라 데모 영상을 보여주기 위해서 시간제한이나 스크램블 신호를 섞어야 하므로 이에 따른 추가 비용이 발생하게 된다. 제안하는 방법은 암호화 알고리즘을 이용하여 영상을 암호화하므로 기존의 방법과는 다르게 별도의 하드웨어 장치가 필요하지 않고 영상의 식별이 불가능할 정도로 영상이 훼손되지 않으므로 비 인증 사용자는 스크램블 신호가 섞인 영상을 봄으로써 데모 영상으로 사용할 수 있는 방법이다. 그리고 암호화의 강도를 조절하여, 암호화 강도를 높일 경우 영상의 식별조차 불가능할 정도로 암호화가 되므로 저작권 보호 분야에도 적용이 가능하다

본 논문에서는 트랜스코딩 과정과 암호화 알고리즘을 이용한 선택적 영상 암호화를 통해 효과적인 CAS 시스템 및 비인증 사용자의 데모 화면 재생 방법을 제안한다. 이를 위해 다양한 영상 샘플에서 트랜스코딩 및 압축, 암호화 시스템을 구현하였다. 이 방법은 Key를 모르면 영상이 제대로 보기 힘들도록 훼손되지만 영상의 인식은 가능하여 효과적인 인증 방법임을 확인하였다. 압축 코덱으로는 H.264를 사용하였는데, H.264는 뛰어난 압축 효율과 품질로 광범위하게 사용되는 코덱이다.

본 논문의 구성은 2장에서는 기존 연구에 대하여 기술하였고, 3장에서는 트랜스코딩 및 콘텐츠 암호화 구현 사항에 대하여 기술한다. 4장에서는 실험결과 및 분석을 기술하고 마지막으로 5장에서는 결론을 맺는다.

2. 관련연구

IPTV나 VOD, 케이블 TV 방송이나 위성 TV 방송 등의 유료 채널을 전송하기 위하여 사업자는 반드시 과금 체제를 갖추어야 한다. 이는 디지털 TV의 상업화에 있어서 반드시 필요한 기능이다. 이를 위해서 셋톱박스, 수신 프로그램 등에는 제한 수신 시스템이 포함된다. 제한 수신 시스템에는 크게 두 가지 종류가 있다. 유료 영상에 스크램블 신호(Scramble Signal)를 섞어 영상을 제대로 볼 수 없도록 망가뜨리는 방법이 있고, 특정 수신기나 수신 프로그램에서만 볼 수 있도록 제어워드(Control Word)키로 전달하는 기술 으로 나눌 수 있다[1]. 그런데 영화 채널 등을 비롯한 몇몇 채널을 제외한 나머지 채널은 시청자에게 데모 영상을 보여주어야 한다. 스크램블 신호를 섞어서 보낸다면 영상의 식별이 가능하므로 데모 영상으로 사용할 수 있다. 그러나 스크램블 신호를 추가하는 별도의 회로가 추가되고, 수신 단말기에서도 스크램블 신호를 처리하는

회로가 추가되어 비용이 발생한다. 제어워드를 삽입하는 방법은 수신기나 송신기의 설계에 있어서 스크램블 방식보다 간단하고 비용이 적으나 사용자 정보를 저장하는 하드웨어가 추가되고 데모 영상용 시간제한 코드 또는 스크램블 신호를 섞는 별도의 회로 또는 코드가 들어가야 하므로 비효율적이다.

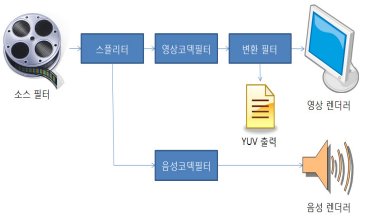
영상의 암호화를 통한 영상 훼손 방법은 기존에는 저작권 보호 분야에서만 사용되는 방법이었다. 그러나 영상에 암호화를 적용시키는 부분을 조절하여 영상을 훼손시키되 인식이 가능할 정도로 보여주고, 영상의 인덱스 이동이 불가능하도록 적용함으로써 데모 영상과 인증의 기능을 가지도록 하였다.

암호화 알고리즘에는 여러 종류가 있다. 암호화 평문을 블록 단위로 처리하는 블록 기반 알고리즘, 암호화키와 복호화 키가 동일한 대칭키 방식의 알고리즘, 암호화키와 복호화 키가 서로 다른 공개키 알고리즘, 단순 암호 방식인 치환 암호화 알고리즘 등이 있다[2]. 블록 기반 대칭키 알고리즘에는 AES, DES, SEED 등이 있고, 공개키 알고리즘에는 RSA, SHA 등이 있다. 공개키 알고리즘은 암호화와 복호화에 시간이 오래 걸리지만, 키가 서로 다르므로 전자서명 등의 분야에 사용된다. 블록 기반 알고리즘, 대칭키 알고리즘은 빠른 속도를 가지고 있으므로 데이터 암호화에 많이 쓰인다.

사용자가 캠코더 등을 이용하여 찍은 영상은 일정한 코덱을 통하여 압축되어 저장된다. 이를 다른 코덱으로 다시 압축하려면 압축 코덱에 맞도록 영상을 트랜스코딩 해주어야 한다. 본 논문에서는 트랜스코딩을 통하여 다양한 포맷의 영상을 변환 및 압축할 수 있도록 하였다. 트랜스코딩이란 압축되거나 서로 다른 방법으로 만들어진 영상을 사용자가 원하는 포맷으로 변경하는 방법을 뜻한다. 트랜스코딩 기법을 사용하면 입력 포맷과 출력 포맷이 같지 않아도 되므로 적용 가능 분야가 넓어지고 사용상의 편의성도 증가한다. 트랜스코딩을 위하여 원본 영상의 압축해제가 선행되어야 하는데, 여기에 많이 사용하는 기술은 Microsoft의 VFW(Video For Windows)와 Direct Show가 있다. VFW가 사용이 더 간단하고 속도가 빠르지만 지원하는 파일이 2GByte 미만의 AVI파일밖에 없다[3]. Direct Show는 윈도우즈용 영상처리 라이브러리로 미디어 재생 및 영상처리용으로 널리 사용된다. 하나의 기능을 가진 모듈 단위로 프로그램이 작성되며 각 기능을 하는 모듈을 필터라 한다. 필터의 조합으로 프로그램이 완성되며 각 조합을 해주는 것을 그래프빌더라 한다[4].

3. 트랜스코더 및 암호화 모듈 구현

본 논문에서는 여러 포맷, 여러 코덱으로 압축된 원 영상을 트랜스코딩하여 압축하는 시스템과 허가받지 않은 사용자에게 영상을 스크램블링(Scrambling)하여 보여주거나 허가받은 사용자에게 스크램블이 제거된 영상을 보여주기 위한 암호화 및 복호화 시스템을 제안한다. 그림1은 트랜스코딩 시스템의 블록도를 나타낸다.



(그림 1) 트랜스코딩 시스템의 블록도

트랜스코딩 시스템은 Microsoft사의 Direct show 필터를 이용하여 작성하였으며 원리는 다음과 같다. 소스 필터를 통과한 데이터는 스플리터를 통해 영상 데이터와 음성 데이터로 나뉘게 되고 각 데이터는 코덱 필터를 통하여 압축이 해제된다. 압축이 해제된 영상은 32비트의 RGBA 포맷을 가지며, 이 데이터는 Direct Show 라이브러리를 이용하여 작성된 변환 필터 내에서 변환 알고리즘을 이용하여 YV12(YUV420P)포맷의 YUV 변환 및 저장을 하게 된다. 저장된 YUV 파일은 후에 인코더를 통해 압축을 하게 된다. 변환 작업이 끝나면 렌더러를 통해 영상과 음성 신호를 출력하게 된다. 위와 같이 필터들의 조합을 해 주는 기술을 널 렌더링(Null Rendering)이라 하며, 널 렌더링에서는 사용할 필터를 정해주면 그래프빌더에서 효과적으로 필터를 연결해준다.

트랜스코딩 시스템에서 Direct show를 사용한 이유는 다음과 같다. 시스템에 설치되어 있는 여러 코덱 필터를 간단히 사용할 수 있고, 원본 영상의 크기에 제한이 없으며, 실험에 사용하기 전에 그래프 에디터 프로그램을 이용하여 간단히 실험해 볼 수 있으므로 사용의 편의성이 높다.

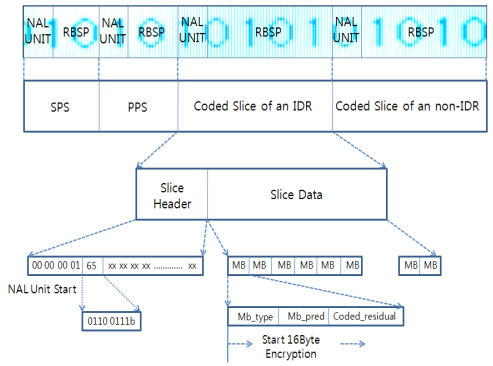
압축 알고리즘으로는 H.264를 사용하였으며 사용한 인코더는 X.264이다. X.264는 H,264 Main 프로파일을 이용하는 인코더로 무료 배포가 되며 속도가 빠르기 때문에 범용으로 사용된다.

본 논문에서 구현한 암호화는 경량화와 영상의 적당한 훼손을 위해 Elementary Stream(ES) 데이터 중 I프레임의 첫 16바이트(128bit)만을 암호화한다. 여기서 헤더 부분은 암호화 하지 않는데 헤더를 암호화하면 영상의 전체 정보가 암호화되어서 비인증 영상 재생 시 플레이어에 문제가 생길 수 있기 때문이다. 이는 포맷 규정을 지켜서 영상의 인식을 가능하게 해 준다.

제안하는 알고리즘에서 암호화 하는 데이터는 영상의 극히 일부에 지나지 않는다. 하지만 이것만으로도 전체 프레임을 정상적으로 출력되지 않게 하는 것은 충분하다. 첫 부분의 작은 데이터 변화는 디코딩시의 허프만 코딩, RLE, DCT, Quantization이 역으로 적용되는 디코딩 과정에서 큰 왜곡을 발생시키기 때문이다[5]. 또한 뒤이어 오는 P프레임은 I프레임을 기반으로 하여 디코딩을 하므로

영상을 효율적으로 망가뜨리는데 부족함이 없다.

그림 2는 암호화를 적용하는 부분을 나타낸다.



(그림 2) ES스트림에서 암호화 영역

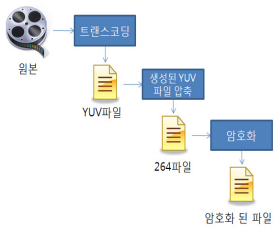
암호화를 적용하는 방법은 다음과 같다. 우선 H.264 스트림에서 헤더 정보를 제외한 매크로블록(MB) 부분만 암호화가 적용되므로 매크로블록을 찾아야 한다. 매크로블록은 Slice 데이터가 포함된 Raw Byte Sequence Payload (RBSP)에 위치하므로 우선 NAL Unit 의 시작패턴을 찾고 뒤이어 나오는 nal_unit_type정보를 이용하여 RBSP에 담긴 데이터가 어떠한 슬라이스인지 알 수 있다. 이를 이용하여 첫 번째 MB데이터를 찾아 암호화를 적용한다.

암호화 알고리즘은 AES[2]와 SEED[6]를 적용하였다. SEED암호화 및 복호화, AES암호화 및 복호화를 사용자의 선택에 의하여 적용할 수 있도록 하였으며, 빠른 I/O 처리를 위하여 압축된 파일의 일부분을 읽어 암호화 한 뒤 수정하는 방법을 사용하였다. 암호화를 적용하는 부분을 프레임 단위가 아닌 슬라이스 단위로 내려간다면, 암호화하는 부분을 늘려서 영상의 인식 자체를 불가능하도록 만들 수 있는데, 이 방법은 저작권 보호 분야에 적용이 가능하다.

4. 실험 및 결과

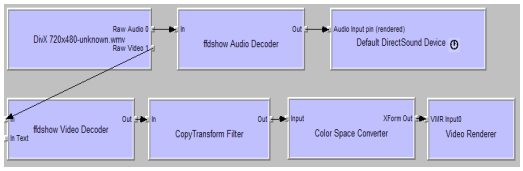
구현 및 실험에 사용된 시스템은 다음과 같다. PC는 펜티엄4 2.6GHz, 2GB Ram, Windows XP SP3 사양이며 프로그램을 작성하기 위해 Microsoft Visual Studio 2008과 Windows Platform SDK를 설치하여 Direct Show 개발 환경을 맞추었다. 트랜스코딩 후 압축 프로그램으로는 VideoLan의 X.264 프로그램을 이용하였으며 암호화 결과 확인을 위한 플레이어로 VideoLan의 VLC Player를 사용하였다. 프로그램에서는 각각 중간 결과 확인을 위하여 중간 결과 파일을 생성하도록 제작하였다.

그림 3에서는 트랜스코딩 및 압축, 암호화를 하기 위한 프로그램 도식을 나타내었다.



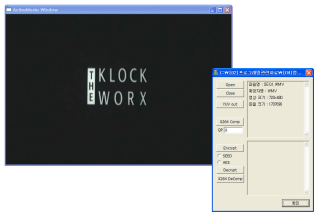
(그림 3) 트랜스코딩 및 암호화 절차

프로그램 제작에 앞서 필터가 정상적으로 작동하는지 실험해보기 위하여 그래프 에디터 프로그램을 이용해 필터 테스트를 진행하였다. 그림 4는 그래프 에디터 상에서 보이는 렌더링 모습이다.



(그림 4) 그래프에디터 실행화면

그래프 에디터를 통하여 트랜스코딩이 제대로 작동함을 확인하였다. 각 상자들은 필터를 나타낸다. 실험에 사용한 영상은 720x480 크기로 WMV9 코덱을 이용하여 압축하였으며, 디코딩에는 ffdshow의 디코더를 이용하였다. 변환 필터로 인한 색공간 변화를 바로잡기 위하여 Elecard의 색공간 변환 필터를 사용하여 화면에 출력하였다. 필터의 정상 작동을 확인하였으므로 프로그램으로 옮겨 적용하였다. 그림 5는 실험용 프로그램의 실행 모습을 나타낸다.



(그림 5) 구현된 트랜스코딩과 암호화 프로그램

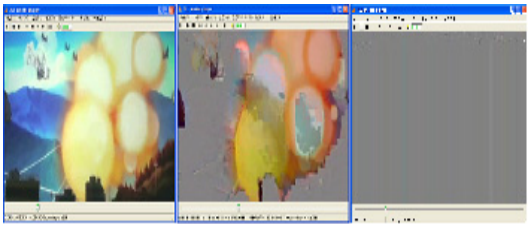
트랜스코딩 과정에서 시간의 소요가 가장 많다. 300프레임의 WMV 영상을 YUV로 트랜스코딩하는데 평균 258초 정도 소요된다.

실험은 QP를 15로 적용하여 압축하였다. 표 1은 압축된 파일과 원본 파일의 비교를 나타낸다.

<표 1 > 압축 파일과 원본 파일의 비교

	원본파일	YUV생성	압축파일
코덱	WMV9	-	H.264
용량(Byte)	1,707,696	155,520,000	16,819
PSNR	-	-	26.94

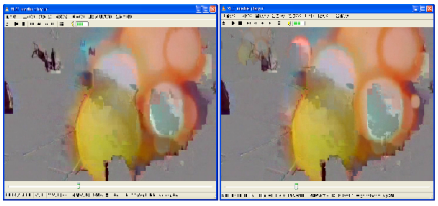
암호화는 SEED 알고리즘과 AES알고리즘을 적용하여 작성하였다. 또한 선택적 암호화를 위하여 P슬라이스의 암호화를 적용하여 암호화의 강도를 조절할 수 있다. 그림 6은 실험의 결과를 나타낸다. 6.(a)는 암호화가 적용된 파일, 그림 6.(b)는 암호화가 적용되지 않은 파일, 그림 6.(c)는 P슬라이스까지 암호화 한 파일을 각각 VLC 플레이어로 재생한 모습이다. 암호화가 적용된 파일은 영상이 제대로 보이지 않고, P슬라이스까지 암호화 한 파일은 영상의 확인조차 불가능한 것을 알 수 있다. P슬라이스까지 암호화 하면 저작권 보호의 목적에도 사용이 가능함을 확인할 수 있다.



(a) 원본 (b) 부분 암호화 (c)전체 암호화

(그림 6) 원본영상과 데모용 영상, 저작권 보호 영상

암호화 알고리즘은 AES와 SEED를 적용하였다. 프로파일링 결과 AES보다 SEED 가 근소하게 빠름을 확인하였으며 암호화의 결과는 두 알고리즘이 비슷한 모습을 보였다. 결과 비교는 그림 7에 나타내었다.



(a) AES 암호화 (b) SEED 암호화

(그림 7) AES와 SEED 암호화 알고리즘의 결과

그림 7.(a)는 AES 알고리즘을 사용한 결과이고 7.(b)는 SEED 알고리즘을 사용한 결과이다. 암호화와 복호화에 평균 71초가량 소요되었다. 실험에서 소요된 시간을 표2에 나타내었다.

<표 2> 실험에 소요된 시간

실험	소요 시간(3회)			평균
트랜스코딩	261	259	256	258.67
압축	64.24	63.44	63.79	63.82
암호화 및 복호화(SEED)	72.4	72.4	70.8	71.87
암호화 및 복호화(AES)	71.3	71.2	70.7	71.07
총계	397.64	394.84	390.59	394.36

5. 결론

본 논문에서는 Direct Show를 이용한 트랜스코딩 시스템과 적당한 강도의 암호화 시스템, 복호화 시스템을 제안하고 구현한다. 제안한 시스템은 다양한 포맷으로 제작된 원본 영상을 트랜스코딩하여 압축하고, 암호화를 적용한다. 그 결과 별도의 해석 모듈과 키 없이는 영상을 정상적으로 볼 수 없었다. 그러나 영상의 재생이 불가능하거나 영상을 전혀 식별하지 못할 정도로 훼손되지 않아 데모 영상으로서 사용할 수 있음을 확인하였다. 본 논문에서 제안한 시스템은 VOD, IPTV등의 유료 채널 뿐 아니라 케이블 TV 방송과 위성TV의 유료 채널 등에도 사용이 가능하고 추후 DMB 방송 등의 유료화 시 제한 수신 장치로서의 역할을 할 수 있으며 그 외 다양한 용도의 인증 장치로 적용 가능하다.

감사의 글

본 논문은 중소기업청의 산학연공동기술개발지원사업(선도형), 한국산업기술재단의 지역혁신인력양성사업의 지원으로 수행되었음.

참고문헌

[1] Baofeng Liu; Wenjun Zhang; Tianpu Jiang “A scalable key distribution scheme for conditional access system in digital pay-TV system,” *IEEE Transactions on* Volume 50, Issue 2, 2004 pp. 632 - 637

[2] W. Stalling, "Cryptography and network security : principles and practice" 3rd Ed. Prentice Hall

[3] James D. Murray, William Vanryper, "Encyclopedia of Graphics File Formats," 2nd Ed. O'Reily

[4] Mark D.Pesce, "Programming Microsoft Direct Show For Digital Video," Microsoft

[5] “128비트 블록 암호알고리즘(SEED) 개발 및 분석 보고서,” 한국정보보호진흥원, 2003.

[6] 김건희, 신동규, 신동일, “효율적인 MPEG-4 비디오 파일의 암호화에 관한 연구,” 한국정보과학회 추계학술발표논문집, 제31권, 제2호, pp. 631-633, 2004.

[7] Jeonghyun Kim, Dowon Nam, Seongoun Hwang, Kisong Yoon, “Protection of MPEG-2 multicast streaming in IP-TV,” *ICCE '06. Digest of Technical Papers,* pp. 45-46, 2006.

[8] W. Zeng, “Format-Compliant Selective Scrambling for Multimedia Access Control," *in Proceedings of IEEE ICASSP*, pp. 77-80, 2002.

[9] Prasertsatid, N. “Implementation conditional access system for pay TV based on Java card,” *Proceedings of IEEE. ICCEA,* pp 533 - 536 2004

[10] Noore, A, "A secure conditional access system using digital signature and encryption," *Consumer Electronics, 2003. ICCE. 2003 IEEE International Conference on* pp. 220 - 221 2003.