MPEG-4 비디오 스트림의 디지털 저작권 관리를 위한 암호화 기법 연구

김 건 희[†] · 신 동 규[††] · 신 동 일[††]

요 약

본 논문에서는 MPEG-4 스트림의 디지털 저작권 관리(DRM)를 위한 암호화 방법을 제안한다. MPEG-4는 멀티미디어 스트리밍을 위한 표준이며 MPEG-4 파일 포맷에 따라 저장된다. 인가된 사용자들만 접근이 가능하도록 MPEG-4 파일 포맷으로부터 추출된 I-VOP, P-VOP의 매크로 블록(MB)들과 모션 벡터(MV)들을 암호화 하는 3가지 방법을 설계하였고, MPEG-4 데이터의 암호화에는 DES(Data Encryption Standard)를 사용하였다. 이러한 암호화 방법을 기반으로 MPEG-4 데이터 스트리밍을 위한 인터넷 방송 서비스용 DRM 솔루션을 구현하고, 최적의 암호화 방법을 선택하기 위해 복호화 속도 및 영상의 품질을 비교하였다.

A Study on Encryption Techniques for Digital Rights Management of MPEG-4 Video Streams

Gunhee Kim[†] · Dongkyoo Shin[††] · Dongil Shin[††]

ABSTRACT

This paper presents encryption techniques for digital right management solutions of MPEG-4 streams. MPEG-4 is a format for multimedia streaming and stored in the MPEG-4 file format. We designed three kinds of encryption methods, which encrypt macro blocks (MBs) or motion vectors (MVs) of I-, P-VOPs (Video Object Planes), extracted from the MPEG-4 file format. We used DES to encrypt MPEG-4 data. Based on theses three methods, we designed and implemented a DRM solution for an Internet broadcasting service, which enabled a MPEG-4 data streaming, and then compared the results of decryption speed and quality of rendered video sequences to get an optimal encryption method.

키워드 : 디지털 저작권 관리(Digital Right Management), 암호화(Encryption), MPEG(Moving Picture Experts Group)-4, 주문형 비디오(Video On Demand)

1. 서 론

오늘날의 통신 시스템은 유무선 환경을 통합하고, 디지털 컨텐츠(Contents) 배포를 위한 고품질의 서비스를 제공한다. MPEG-4를 사용한 스트리밍 서비스(Streaming Service)는 유무선 모두에 적합한 양질의 방송 솔루션 중의 하나이다. MPEG-4는 MPEG-1/MPEG-2와 같은 이미 널리 잘 알려진 멀티미디어 표준을 제정한 MPEG(Moving Picture Experts Group)에서 개발하였다. 이 표준은 CD-ROM, DVD, 디지털 텔레비전, 인터넷 방송을 위한 쌍방향 비디오 통신을 가능하게 한다[1].

MPEG-4 스트리밍의 저작권 관리는 주문형 비디오(Video On Demand)와 화상 회의(Video Conference) 솔루션과 같은 멀티미디어 서비스에 중요하며, 서비스에 대한 적절한 비용을 지불한 사람들만 디지털 컨텐츠에 접근할 수 있도록 하게 한다[2]. 디지털 저작권 관리(DRM, Digital Right Management) 시스템은 저작권자의 지적 재산권(Intellectual Properties)을 보호하기 위해서 디지털 컨텐츠의 사용을 제한하며, 디지털 컨텐츠의 재생 횟수, 재생 시간, 수정, 회람, 모사, 출력, 저장 등을 제한하게 된다. 이러한 기술은 재생기나 플러그인 또는 주변 장치 내의 프로그램 같은 소프트웨어에 포함된다. DRM 시스템은 컨텐츠를 보호하기 위해 두 가지 방법을 사용하는 데, 그 첫 번째는 인증된 사용자만 접근 가능하도록 디지털 컨텐츠를 암호화(Encryption) 하는 것이고, 두 번째는 미디어 복사를 막기 위해 디지털 컨텐츠에 워터마크(Watermark), 플래그, XrML(eXtensible Right Markup Language)을 삽입하는 것이다[3].

ISO의 MPEG에서는 이러한 저작권 관리 서비스를 위하여, 지적재산권관리보호(IPMP, Intellectual Property Management and Protection)라는 디지털 저작권관리기술을 표준화 하고 있다[16]. MPEG-4 IPMPX(IPMP Extension)는 MPEG-4 포

† 준 회 원 : 세종대학교 컴퓨터공학과 박사과정
†† 종신회원 : 세종대학교 컴퓨터공학과 부교수
논문접수 : 2004년 3월 18일, 심사완료 : 2004년 12월 3일

맷을 위한 컨텐츠 보호 시스템의 프레임워크를 정의하고 있으며, 다양한 DRM 기법들을 독립적인 도구(tool)의 개념으로 간주하고 이러한 보호 툴들이 단일의 단말에서 적용될 수 있도록 하는 목표를 추구한다[15].

본 논문에서는 MPEG-4를 사용하는 멀티미디어 서비스의 저작권 관리를 위하여 첫 번째 접근 방법을 적용하며, DES(Data Encryption Standard)[4, 5]를 사용하여 MPEG-4 데이터를 암호화 하였다. 이를 위하여 MPEG-4 파일 포맷에서 추출한 매크로 블록(MB, Macro Block), I-VOP(Video Object Plane), P-VOP의 모션 벡터(MV, Motion Vector)를 암호화하는 3가지 암호화 방법을 제안하였다. 또한 이 3가지 방법을 기반으로, MPEG-4 데이터 스트리밍을 위한 인터넷 방송 서비스용 DRM 솔루션을 설계하고 구현하였다.

2. 배 경

디지털 저작권 관리(Digital Rights Management, DRM)는 컴퓨팅 환경에서 유통되고 저장, 전송될 수 있는 모든 형태의 디지털 콘텐츠 데이터를 보호한다. DRM의 기본 목적은 비용을 지불하고 인가된 권한을 갖는 사용자를 제외한 사용자들이 콘텐츠를 이용할 수 없도록 하는 것이며, 단순한 복사방지 시스템에서부터 워터마크(Watermark)를 삽입하는 방법, 내용을 암호화하는 방법, 핑거 프린팅(Fingerprinting)에 기반한 방법 등이 모두 여기에 속한다 [17].

MPEG-4 비디오 표준은 멀티미디어 환경에서 사용될 텍스처(Texture), 이미지, 비디오 데이터의 효율적인 저장, 전송, 조작 등을 가능하게 하는 표준 핵심 기술을 제공한다. MPEG-4 비디오 코딩 알고리즘은 다양한 수준의 입력 형식, 프레임 수, 비트 수의 이미지 시퀀스(Image Sequence)를 효과적으로 압축하는 것을 포함하여, 최종적으로 MPEG-1과 MPEG-2에서 지원되는 모든 기능을 지원하게 될 것이며, 추후에는 내용 기반 기능 또한 지원될 예정이다[1, 6]. 임의의 모양(Shape)의 VOP 이미지 영역(Image Region), 프레임(Frame)별로 유동적인 영역의 모양과 위치(Location)가 모두 입력이 될 수 있다. 하나의 장면(Scene) 내의 동일한 물리적 객체에 해당하는 연속적인 VOP들을 비디오 객체(VO, Video Object)라 부르며, 이것은 임의의 모양과 위치를 갖는 VOP들의 시퀀스(Sequence)이다. 동일한 비디오 객체에 해당하는 VOP들의 모양과 움직임, 텍스쳐 정보는 인코딩되고 전송되거나, 코드화된다. MPEG-1과 MPEG-2처럼, I-VOP, P-VOP, B-VOP은 VOP의 기본 타입이고, 각각의 VOP은 매크로 블럭들로 분해되고, 매크로 블럭은 다시 6개의 블럭들로 나누어지며, 그 블럭 내에서 DCT(Discrete Cosine Transform)가 적용된다[6, 7].

2.1 전통적인 MPEG 비디오 암호화 기법

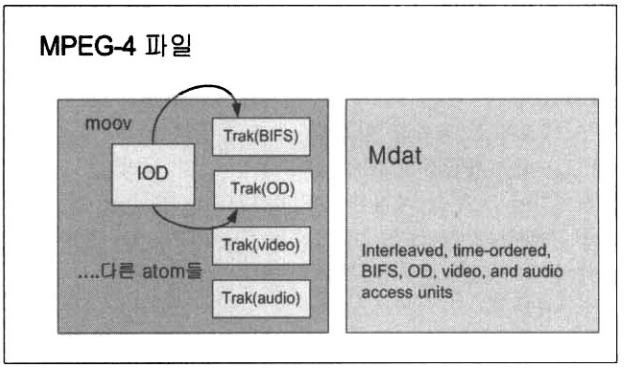
전통적으로 MPEG-1과 MPEG-2는 인증되지 않은 사용자가 스크램블하여 비디오 프로그램을 디코딩하지 못하도록 비디오 암호화 알고리즘을 사용한다[8, 9]. 일반적으로 평문으로 불리는 비디오 스트림 S에 비가역적 형 변환 E_{k1}을 적용하여, 비트스트림(Bitstream)인 암호문 C를 생성하는 방법이 사용된다($C = E_{k1}(S)$). 비밀키 $k2$를 가진 인증된 사용자는 $D_{k2} = E_{k1}^{-1}$을 이용해 암호화된 비디오 스트림을 복호화할 수 있다. 복호화 과정은 다음과 같으며, 인자 $k1$은 암호화 키, $k2$는 복호화 키이다.

$$D_{k2}(C) = E_{k1}^{-1}(C) = E_{k1}^{-1}(E_{k1}(S)) = S$$

몇몇 비디오 인코딩 알고리즘들은 MPEG으로 압축된 비디오의 DCT 계수들의 부호 비트에만 작용하는 선택적 암호화 알고리즘을 사용한다[8, 10]. VEA(Video Encryption Algorithm)에서는 암호화 키와 복호화 키가 동일하다. 짧은 키와 복잡한 계산(DES)을 이용하기 때문에, 텍스트 데이터를 보호하기 위해 개발된 블록 암호화 알고리즘이 사용된다. 외부의 공격을 예방하기 위해서는 길이가 긴 키를 사용해야 하고, 스트리밍 서비스와 같은 멀티미디어 어플리케이션을 위한 실시간 고성능을 모색하기 위한 간단한 계산 알고리즘을 사용해야 한다.

2.2 MPEG-4 파일 포맷

MPEG-2는 엔터테인먼트용 비디오와 오디오의 표준이며, DVD(Digital Versatile Disc)와 DVB(Digital Video Broadcasting)를 위해 제정된 형식인 반면에, MPEG-4는 현재 ISO의 MPEG(Moving Picture Experts Group)에 의해 정의된 디지털 미디어 표준이며, 사용자들이 디지털 컨텐츠의 오디오, 비디오 및 다른 형식의 디지털 컨텐츠들을 선택하고 관람하고 조작할 수 있게 한다. MPEG-4 파일 포맷 표준을 채용함으로써, 모든 디지털 미디어 컨텐츠가 실시간 비디오와 오디오 스트리밍까지 지원하는 하나의 공통 파일 포맷으로 저작될 수 있게 된다[1]. 이러한 디지털 스트림은 인터넷과 기업 네트워크를 통해 전송되거나, 가정으로 직접 방송되게 된다.

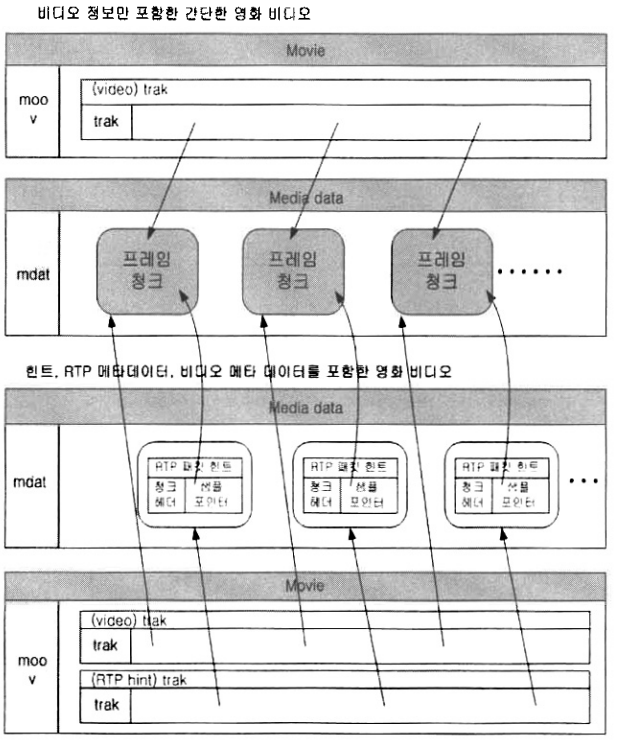


(그림 1) MPEG-4 파일 포맷의 예

MPEG-4 파일 포맷은 MPEG-4 표현의 미디어 정보를 유연하고 확장 가능한 형태로 포함할 수 있도록 설계되었고, 이것은 상호 교환, 관리, 편집, 그리고 미디어의 표현을 용이

하게 한다. 이러한 설계는 Apple의 QuickTime 포맷[11]을 기반으로 한다. MPEG-4 파일 포맷은 atom이라 불리는 객체 기반 구조로 구성되어 있다. 유일한 태그와 길이는 각각의 atom을 식별해주며, 대부분의 atom들은 미디어 데이터에서 색인점, 시간, 미디어 데이터를 가리키는 포인터 같은 정보를 제공하는 메타데이터의 계층구조를 기술한다. 이러한 atom들은 movie(MOOV) atom에 포함되어 있으며, 미디어 데이터의 위치는 어디든 가능하다. 미디어 데이터는 MPEG-4 파일 내부 혹은 외부에도 위치할 수 있고, URL을 통해 지정될 수도 있다. (그림 1)은 3 개의 스트림을 포함한 간단한 교환 파일의 예이다[1].

파일 포맷은 스트리밍(streaming)포맷이 아니고 스트림 가능한(streamable) 포맷이다. 즉, 파일 포맷이 실제로 전송 매체를 통해 스트림되지 않으며, 힌트 트랙(hint track)인 메타데이터가 특정한 전송 프로토콜을 통해 미디어 데이터가 전달되는 방법을 서버 어플리케이션에게 알려주는 등의 지시사항을 제공한다. 하나의 프리젠테이션에는 다양한 전송 프로토콜을 통해 전달할 방법을 기술하는 여러 개의 힌트 트랙이 존재할 수 있다. 이런 방식으로 파일 포맷은 직접 전송되지는 않고 스트리밍을 용이하게 해준다[12, 13].

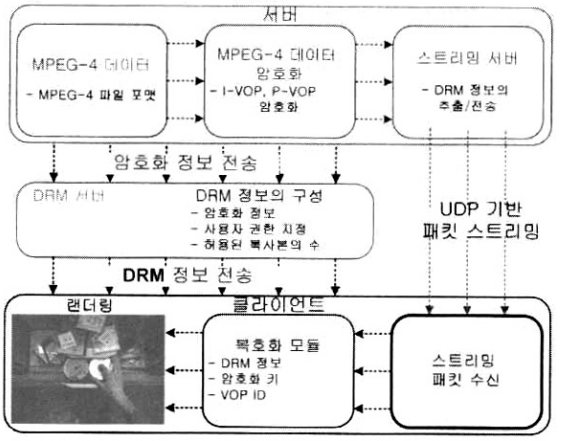


(그림 2) 간단한 비디오 스트림용 RTP 프로토콜 힌트 트랙들의 관계

(그림 2)는 간단한 비디오 스트림용 RTP 프로토콜 힌트 트랙들의 포함관계를 보여준다. 파일내의 메타데이터는 미디어 데이터의 유연한 저장성과 결합하여, MPEG-4 파일 포맷이 컨텐츠의 스트리밍, 편집, 재생, 및 교환을 지원하게 한다[1, 14].

3. DRM을 적용한 인터넷 방송 시스템의 설계 및 구현

본 논문에서는 MPEG-4 데이터 스트리밍과 다운로드가 가능한 인터넷 방송 서비스용 DRM 솔루션을 설계하고 구현하였다. 시스템의 전반적인 구조는 (그림 3)에 나타내었다.



(그림 3) 인터넷 방송을 위한 DRM 솔루션의 전반적인 구조

먼저 MPEG-4 파일에서 비디오 데이터를 추출하고, 여기에 3가지 종류의 암호화 방법을 적용하였다.

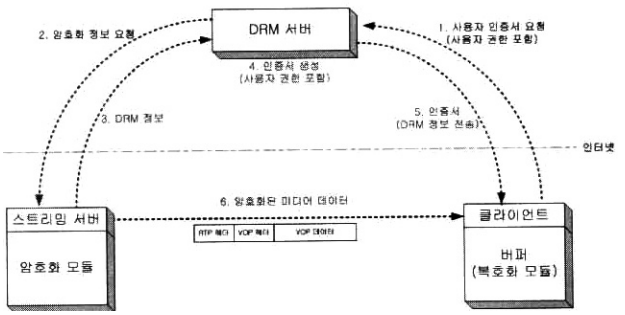
1) I-VOP 내의 매크로 블록(MB)들의 암호화 (*방법-1*)
2) P-VOP 내의 MB들과 모션 벡터(MV)들의 암호화 (*방법-2*)
3) I-VOP 내의 MB들과 P-VOP 내의 MB와 MV를 모두 암호화 (*방법-3*)

암호화 알고리즘으로는 DES(Data Encryption Standard)[4, 5]를 사용했으며, 본 연구의 목표는 이러한 3가지 방법을 구현하고 테스트하여 DRM 솔루션을 위한 최적의 암호화 기법을 찾는 것이다.

이전의 암호화 방법[8, 9]은 비디오 데이터를 얻기 위해서 MPEG 암호화 프로그램 소스를 사용하기 때문에 일반적인 접근 방법이 되기 어렵다. 본 연구에서는 각각의 VOP에 대한 MB와 MV를 추출하기 위해 MPEG-4 파일 포맷 내의 MPEG 데이터에 직접 접근을 시도하고, 그렇게 해서 이미 인코딩된 데이터에 접근하였다. 덧붙여, 이미 암호화된 MPEG-4 파일을 효율적으로 관리 할 수 있도록 MPEG 데이터 파일에 DRM 정보를 삽입하게 된다.

암호화된 MPEG-4 데이터는 DRM 서버가 제공한 DRM 정보와 함께 클라이언트에 전송된다. DRM 정보는 복호화 키와 사용자 인증 정보 등을 포함한다. 클라이언트는 복호화 모듈을 사용해서 실시간으로 전송된 미디어 데이터를 복호화하고, 복호화된 미디어 데이터를 렌더링(rendering)한다. (그림 4)는 클라이언트와 스트리밍 서버 사이에서 DRM 서버가 DRM 정보를 전송하는 구조를 보여준다. 이와 같은 구조에서는 버퍼(buffer)에서 전처리하는 방식으로 복호화 모듈이 각각의 VOP을 처리하기 때문에, 데이터를 복호화하기 위해

비디오 코덱(codec)이나 오디오 코덱을 수정할 필요가 없다.

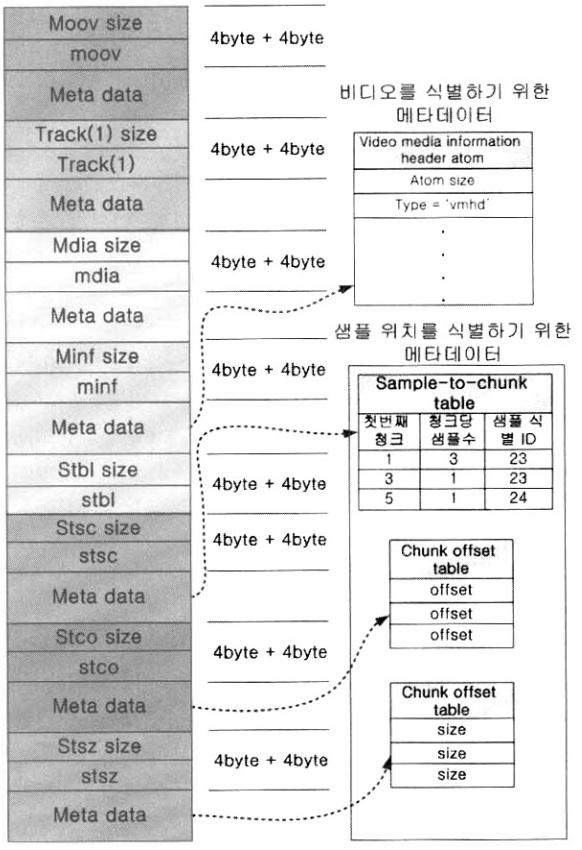


(그림 4) 클라이언트와 서버 간의 DRM 정보 전송

3.1 비디오 프레임 추출 알고리즘

2.2에서 전술한 것처럼, MPEG-4 파일 포맷은 atom들로 구성되어 있으며, 크게 메타데이터와 실제 데이터의 두 부분으로 나뉘어 진다. 메타데이터는 Movie(MOOV) atom에 포함되어 있고, 실제 미디어 데이터는 mdat atom에 포함되어 있다[1, 11].

MPEG-4 파일 포맷의 비디오 데이터를 암호화하기 위해서 MPEG-4 파일의 메타데이터를 파싱(parsing)해야만 하고, I-VOP, P-VOP, B-VOP과 같은 비디오 프레임들을 추출해야 한다. (그림 5)는 비디오 데이터를 추출하기 위해서 MPEG-4 파일 포맷의 메타데이터를 해석하는 방법을 보여준다.

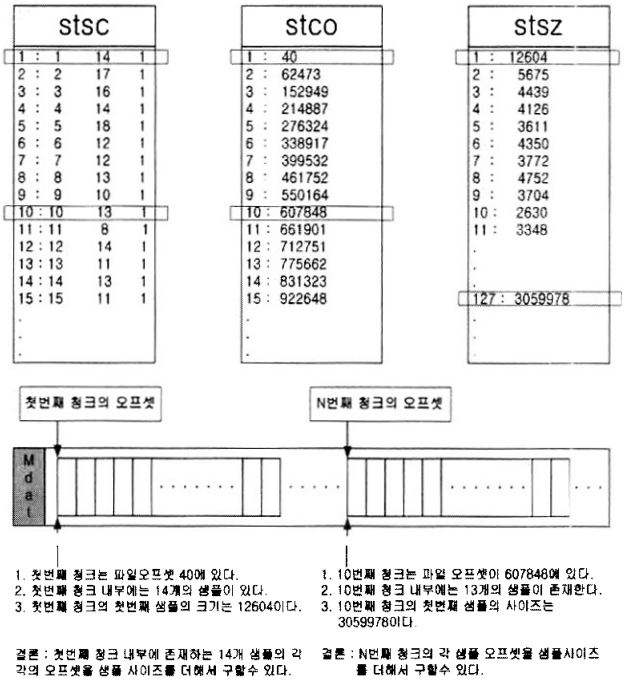


(그림 5) MPEG-4 파일 포맷의 메타데이터 해석

비디오 샘플을 추출하기 위한 절차는 다음과 같다.

1) 비디오 미디어 정보 헤더 atom의 존재를 검사하여 비디오 트랙을 검색한다.
2) 비디오 트랙 내의 stbl(sample table) atom들을 획득한다. stbl atom들은 실제 미디어 데이터와 기술(description) 정보 등의 위치를 포함한다.
3) stco(sample to chunk), stsc(sample per chunk), stsz (sample size) atom들로부터 테이블 정보(table information)를 획득한다.

위의 기술된 절차에 따라, 비디오 트랙 내의 sample 테이블과 각 샘플의 위치 정보를 얻는다. (그림 6)은 각 샘플의 위치정보를 이용하여 오프셋(offset)을 계산하여 mdat atom으로부터 실제 데이터를 획득하는 방법을 보여준다[1, 11].



(그림 6) 오프셋을 계산하여 비디오 데이터를 획득

3.2 I-VOP의 매크로 블록 추출 및 암호화 (방법-1)

I-VOP을 비디오 데이터에서 추출하고, DES를 I-VOP에 적용하여 비디오를 암호화할 수 있다. DES는 블록 암호화 알고리즘이며, 입력으로 64비트 데이터를 갖고, 암호화된 64비트 데이터를 출력하므로 입출력 데이터의 크기에는 변화를 주지 않는다. 만약 입력과 출력 데이터의 크기가 다른 암호화 알고리즘을 사용한다면, MPEG-4 파일에 덧붙임(padding) 비트를 추가해야 하거나, 오프셋이 바뀌게 되고, 결국 MPEG-4 파일 자체의 크기와 구조에 변화가 생기게 된다. 그러므로 스트림 암호화를 위해서는 DES와 같은 대칭형 알고리즘을 사용해야 하고, 파일 크기가 변하지 않도록 하기 위해 DES의 입력 값으로 64의 배수를 적용해야하며, 이렇게 하여 DES 함수의 덧붙임이 일어나는 것을 방지한다.

MPEG-4 비디오 표준[6]을 따라 비디오 데이터에서 I-VOP을 추출하였다. (그림 7)에서 보인 것처럼, 각 VOP은

하나의 video_start_code를 가진다. video_start_code의 값은 16진수로 00 00 01 B6이며, vop_coding_type는 I-VOP, P-VOP, B-VOP들의 VOP타입을 식별한다.

```
VideoObjectPlane() {          No. of bits
Mnemonic
        vop_start_code        32        bslbf
        vop_coding_type       2         uimsbf
        do {
                modulo_time_base  1       bslbf
        } while (modulo_time_base != '0')

        marker_bit1           bslbf
        vop_time_increment    1-16      uimsbf
        marker_bit1           bslbf
.....
}
```

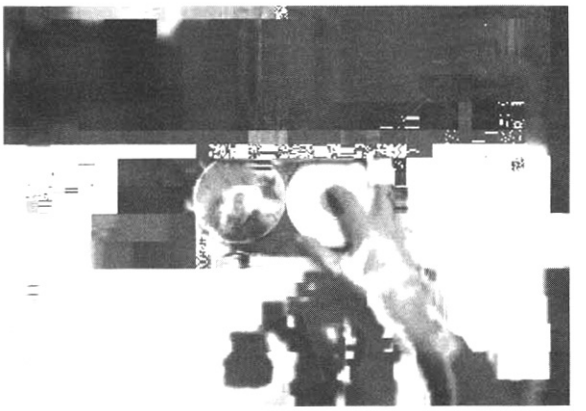
(그림 7) MPEG-4 비디오 표준의 Video Object Plane 구조

<표 1>은 vop_coding_type을 보여준다.

〈표 1〉 VOP 코딩 타입

vop_coding_type	코딩 방법
00	Intra-coded (I-VOP)
01	Predictive-coded (P-VOP)
10	Bidirectionally Predictive-coded (B VOP)
11	Sprite

(그림 8)은 I-VOP 내의 매크로 블록들을 암호화한 결과를 보여준다. 왼쪽에 있는 그림은 원본 데이터이고, 오른쪽에 있는 그림은 복호화 과정을 거치지 않고 암호화된 데이터를 그대로 재생한 결과이다. P-VOP의 인트라 코딩된 매크로 블록들이 중요한 재생 정보를 보존하고 있기 때문에, 오른쪽과 같은 깨진 그림 내의 몇몇 인지 가능한 작은 그림 조각들이 나타난다.



(그림 8) I-VOP 내의 매크로 블록을 암호화한 결과

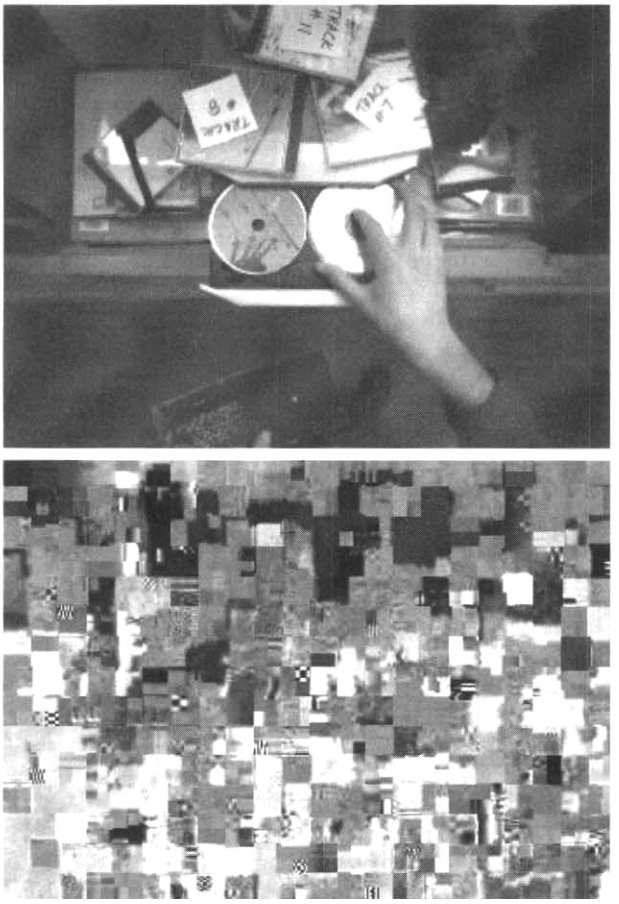
3.3 P-VOP 내의 매크로 블록들과 모션 벡터들의 암호화 *(방법-2)*

MPEG-4 파일 포맷과 헤더구조를 분석하여, P-VOP들을 추출하고 P-VOP 내의 인트라 코딩된 매크로 블록(MB)들과 모션 벡터(MV)들을 역시 *방법-1*처럼 DES를 이용하여 암호화할 수 있으며, 결과는 (그림 9)와 같다. I-VOP의 인트라 코딩된 매크로 블럭들이 많은 정보를 가지고 있기 때문에, 테스트 결과가 스트리밍되는 동안 주기적으로 몇몇 완전한 정지 영상들이 나타나게 된다. 만약, 비디오 시퀀스 내에서 I-VOP의 점유 비율이 크지 않다면, 이러한 P-VOP의 암호화는 효율적인 암호화 방법이 될 수 있다.

(그림 9) P-VOP 내의 매크로 블록과 모션벡터 암호화 결과

3.4 I-VOP 내의 매크로 블럭과 P-VOP 내의 모션 벡터 동시 암호화 *(방법-3)*

인트라 코딩된 매크로 블록들이 많은 코딩 정보들을 지니고 있기 때문에 *방법-1*과 *방법-2*는 모두 완전히 만족할 만한 결과를 제공해 주지는 않는다. 이 문제를 해결하기 위해서 I-VOP 내의 매크로 블록들과, P-VOP 내의 모션 벡터들을 모두 암호화하는 방법을 이용할 수 있다(*방법-3*). (그림 10)은 그 *방법-3*의 결과를 보여주며, 이것은 3가지 방법 중에 가장 강력한 암호화 방법이라는 장점을 갖지만, 암호화를 위해 처리해야할 데이터의 양이 *방법-1*과 *방법-2*에 비하여 가장 많다는 단점을 갖는다.



(그림 10) I-VOP 내의 매크로 블럭과 P-VOP 내의 모션벡터를 모두 암호화한 결과

4. 암호화 속도에 관한 모의 실험

실험에 사용된 데이터 파일은 Apple사의 홈페이지에서 내려 받은 CDR_Dinner_800k-997.mp4 샘플 파일을 사용하였으며, 이 데이터 파일의 특성을 <표 2>에 기술하였다.

본 연구에서는 MPEG-4 데이터를 위한 3 가지 암호화 방법을 제안 하였다. 그 중 *방법-3*이 가장 뛰어난 암호화 결과를 보여주는 반면, 암호화 과정에서 처리해야하는 데이터의 양이 가장 크다. 이러한 단점을 극복하기 위한 속도

〈표 2〉 CDR_Dinner_800k-997.mp4 파일의 특성

Size	2,965,043 bytes
Number Of Samples	902
Video Type	MPEG-4 Advanced Simple @ L3
Bitrate	713 kbps
Duration	0:30.063 minutes
Framerate	30.004 fps
Width x Height	348 x 240

증진을 위해 I-VOP의 빈도수를 조정하여 암호화에 사용될 데이터의 양을 조절할 수도 있으나, 일반적으로 I-VOP의 매크로 블럭들만 암호화하는 작업에 필요한 데이터양은 그리 크지 않다. VOP을 암호화하는 시간은 아래와 같다.

$$E(t) = DES(t) + M(t)$$

$E(t)$는 VOP의 암호화 작업을 처리하는 시간이고, $DES(t)$가 DES 암호화 작업을 처리하는 시간, $M(t)$는 전처리 시간이다.

<표 3>는 위 공식에 의하여 "CDR_Dinner_800k-997. mp4" ((그림 8, 9, 10)의 암호화 결과에서 사용된 파일)파일을 이용해 얻은 결과이다.

〈표 3〉 제안된 3 가지 방법의 암호화 속도 측정 결과

(시간 단위 : 밀리 초, millisecond)

데이터 파일	이용된 방법	VOP 수	VOP 당 평균 암호화 시간	전체 VOP 암호화 시간
CDR_Dinner_800k 997.mp4	방법-1	23	1.5294389	35.1770958
	방법-2	879	0.2557534	224.8072710
	방법-3	902	0.2664301	240.3199226

<표 3>은 미디어 스트림을 위한 암호화의 속도는 암호화를 위한 데이터의 양에 좌우된다는 것을 보여준다. 비록 *방법-3*이 *방법-1*보다 암호화 및 복호화 작업에 7배 정도의 시간소요를 요구하지만, 일반적인 MPEG 클라이언트는 복호화, 디코딩, 렌더링 작업 모두를 메모리나 스왑영역의 전처리 버퍼에서 처리한 후 재생하기 때문에 실제 재생 시간 그 자체에는 큰 영향을 주지 않는다.

5. 성 능

본 연구에서 제안한 *방법-1,2,3*은 MPEG-4 파일 포맷 및 헤더 구조를 파악하여 직접 I-VOP 및 P-VOP을 추출하고, 여기서 얻어진 VOP들의 매크로블럭 및 모션 벡터만을 암호화하므로, 전통적으로 MPEG 파일을 디코딩한 후 특정한

암호화 알고리즘을 적용하여 다시 인코딩하는 방법들에 비해 매우 빠르고 효과적인 암호화 작업을 수행하도록 할 수 있다. 이러한 방법은 최소한의 데이터만을 대상으로 암호화 작업을 수행하므로, 비디오 데이터의 암호화에 소요되는 컴퓨팅 파워 및 소비 시간을 최소한으로 줄일 수 있다. 또한, 파일 사이즈 및 데이터 파일의 구조에 변화를 발생시키지 않으므로, 암호화된 데이터의 관리 효율도 증대 시킬 수 있다. 그리고, 암호화 때문에 특정한 디코더나 인코더를 따로 구성할 필요도 없으며, 추가적인 암호화/복호화 모듈만을 이용하여 일반적인 클라이언트와 디코더를 그대로 사용할 수 있는 장점도 갖는다.

6. 결 론

본 연구에서는 MPEG-4파일 포맷에서 비디오 데이터를 추출하고 DES를 사용하여 VOP들을 암호화하는 DRM 솔루션을 설계하고 구현하였다. VOP에서 매크로 블록(MB)과 모션 벡터(MV)를 추출하여 암호화하는 3가지의 암호화 방법을 사용하였으며, *방법-3*이 가장 좋은 암호화 결과를 보여주는 반면, 가장 많은 데이터양과 처리 시간이 요구 하였다. 이러한 각각의 암호화 방법들은 특정한 조건 하에서 각각 최적의 성능을 나타낼 수 있다.

본 논문에서 제시한 방법은 전통적인 MPEG 비디오 영상의 암호화 기법처럼, 영상을 보호하기 위해 기존의 MPEG 영상을 디코딩하여 특정한 암호화 기법을 적용한 후 다시 MPEG방식으로 인코딩하거나 두 작업을 동시에 진행하는 일반적인 암호화 기법에 비하여 부가적인 계산 오버헤드가 들어가지 않는다는 장점을 가진다. 이러한 특징은 무선 멀티미디어 스트리밍 서비스와 같은 저 대역의 통신 환경 하에서 구축되는 상업용 멀티미디어 컨텐츠 배달 서비스와 같은 응용 프로그램이나 서비스에 매우 큰 강점을 가지게 할 수 있다.

향후에는 멀티미디어 스트리밍 서비스를 위해 암호화된 미디어 데이터와 결합된 DRM 정보의 관리 방법이 중점 연구될 것이다. 일례로 DRM 정보는 효율적인 서비스를 지원하기 위해 MPEG-4 파일 포맷에 삽입되어야 할 필요성이 있다.

참 고 문 헌

[1] "Information technology-Coding of audio-visual objects-Part 1 : Systems ISO/IEC 14496-1 : 2001", ISO/IEC/SC29/WG11, 2001.

[2] Hartung Frank and Ramme Friedhelm, "Digital Rights Management and Watermarking of Multimedia Content for M-Commerce Applications", IEEE Communications Magazine, pp.78-84, Nov., 2000.

[3] "EPIC(Electronic Provacy Information Center) Digital Right Management and Privacy Page", http://www.epic.org/privacy/drm/.

[4] "Data Encryption Standard (DES)", FIPS PUB 46-3, Oct., 25., 1999.

[5] Stallings William, "Cryptography and Network Security : Principles and Practice", 3rd Edition, Prentice Hall, 2002.

[6] "Information Technology Coding of Audio-Visual Objects Part 2 : Visual, ISO/IEC 14496-2", ISO/IEC/SC29/WG11, Nov., 1998.

[7] Sikora Thomas, "The MPEG-4 video standard verification model", IEEE Transactions on Circuits and Systems for Video Technology, pp.19-31, Feb., 1997.

[8] C. Shi and B. Bhargava, "An Efficient MPEG Video Encryption Algorithm", Proceedings of Seventeenth IEEE Symposium on Reliable Distributed Systems, pp.381-386, Oct., 1998.

[9] A. M. Adnan, G.I. Al-Regib and S.A. Al-Semari, "Improved Selective Encryption Techniques for Secure Transmission of MPEG Video Bit-Streams", Proceedings of 1999 International Conference on Image Processing, Vol.4, pp. 256-260, Oct., 1999.

[10] C. Shi and B. Bhargava, "A Fast MPEG Video Encryption Algorithm", proceedings of the ACM Multimedia, Sep., 1998.

[11] "QuickTime File Format", Apple Computer, June., 2000.

[12] "QuickTime Streaming Server Modules", Apple computer, 2002.

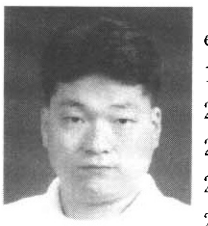
[13] Gringeri Sieven, Iren Sami, "Transmission of MPEG-4 video over the Internet", IEEE International Conference on Multimedia and Expo, pp.1767-1770, Jul., 30-Aug., 2. 2000.

[14] "RTP Payload Format for MPEG-4 Audio/Visual Stream", RFC 3016, Nov., 2000.

[15] ISO/IEC JTC-1/SC29/WG11 N2614 MPEG-4 IPMP Overview & Applications Document

[16] M8141, Proposed new text of IPMP FAQ, 2002

[17] W. Jonker and J.-P. Linnartz, "Digital rights management in consumer electronics products," Signal Processing Magazine, IEEE, Vol.21, Issue 2, pp.82-91, Mar., 2004.



김 건 희

e-mail : ghkim@gce.sejong.ac.kr

1997년 한림대학교 컴퓨터공학과(공학사)

2000년 세종대학교 전산과학과(이학석사)

2000년~2001년(주)에피온

2001년~2004년Motorola Korea Inc.

2004년~현재 세종대학교 컴퓨터공학과 박사과정

관심분야 : 멀티미디어 디지털 저작권 관리, 상황인식 컴퓨팅 미들웨어 기술, 인간-컴퓨터 상호작용, 유비쿼터스 컴퓨팅, 무선 인터넷 플랫폼, MPEG-4

신 동 규

e-mail : shindk@sejong.ac.kr

1986년 서울대학교 계산통계학과(이학사)

1992년 Illinois Institute of Technology 전산학과(공학석사)

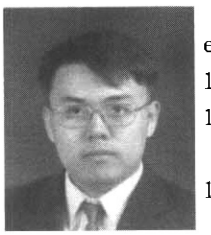
1997년 Texas A&M University 전산학과 (공학박사)

1986년~1991년 한국국방연구원 연구원

1997년~1998년 현대전자 멀티미디어연구소 책임연구원

1998년~현재 세종대학교 컴퓨터공학과 부교수

관심분야 : 웹기반 멀티미디어, 멀티미디어 저작권 보호기술, XML 보안 및 응용, 영상압축



신 동 일

e-mail : dshin@sejong.ac.kr

1988년 연세대학교 전산과학과(이학사)

1993년 M.S. in Computer Science, Washington State University

1997년 Ph.D in Computer Science, University of North Texas

1997년~1998년 시스템공학연구소 선임연구원

1998년~현재 세종대학교 컴퓨터공학과 조교수

관심분야 : 무선인터넷, 게임, XML 보안 및 응용, 지능형 에이전트, HCI