



정보기술의 최근동향(V)

데이터 압축 기술

원광일

멀티미디어 컨설턴트 / 서울사스팀 상무

목 차

1. 미디어 최신 동향	7월호
2. 대화형 TV 및 인터넷	8월호
3. 컴퓨터 및 소프트웨어	9월호
4. 초고속 및 이동통신	10월호
5. 데이터 압축기술	이번호
6. 표준화	다음호
7. 분야별 사례연구	내년 1월호

멀티미디어 데이터 압축 기술 중에서 동화상 압축표준인 MPEG은 오늘날 가장 중요한 기술로 인식되고 있다. MPEG은 컴퓨터뿐만 아니라 가전, 통신, 방송 등 우리 일상 생활과 관련된 전 부분에 걸쳐 영향을 끼치고 있다. 오늘날 디지털 기술은 MPEG의 발전과 매우 연관되어 있다고 하겠다.

MPEG(Moving Picture Experts Group) 즉 오디오-비디오 정보의 디지털 압축에 관한 국제협약은 원래 이들 표준을 개발한 전문가 그룹에 부여한 이름이며 MPEG-1, MPEG-2 및 곧 정립될 MPEG-4로 구성되어 있다.

MPEG-1과 MPEG-2는 각각 컴퓨터와 디지털 방송에 적용되어 현실화되고 있으나 멀티미디어 통신은 아직도 가설에 머무르고 있는 실정이다. “멀티미디어”란 단어가 나온 지 10년이 지났고 “융합(Convergence)”이란 단어는 5년, “디

지털 비디오”가 실용화된 지 2년 반이 지났지만 우리는 아직도 멀티미디어 통신을 실현시키기 위하여 몸부림치고 있다. 진정으로 초고속 정보화 시대가 도래하기 위해선 지금 한참 논의 중인 MPEG-4가 조속히 완료되어야 한다. 따라서 본 기사는 MPEG-4를 중심으로 기술하기로 한다.

MPEG-1

MPEG은 1988년 1월에 동화상, 오디오 및 이들의 동기에 관한 표준을 위하여 설립되었다. 이 그룹은 정보기술에 관한 ISO/IEC 합동 기술 위원회(Joint Technical Committee; JTC 1) 산하에서 활동하였고 곧이어 정식으로 SC29, WG11이라 명칭이 부여되었다.

이 그룹에서 제정한 최초의 표준은 MPEG-1으로 1.5 Mbit/s의 전송 속도에서 오디오-비디오 신호 코딩에 관한 표준이다. 이것은 VHS 카세트 수준의 화질로 콤팩트 디스크에 비디오 신호를 저장하는 것을 염두에 두고 정한 것이다.

그런데 이렇게 낮은 속도에서 디지털 코딩을 실현시키기 위해서 수십 년에 걸친 최신의 코딩 기술이 적용되긴 했어도 특히 비디오 부분에선 물리적으로 손해를 감수해야만 했다. 즉 비디오 부분은 수직과 수평으로 각각 절반으로 줄어들어 정상 화면의 1/4이 되며, 이렇게 줄어든 화면을 잡아 늘려 정상 크기의 화면으로 만들면 화질이 떨어지는 것은 당연하다. 그리고 오디오부분에서



정보기술의 최근동향(V)

는 샘플링 오디오 정보(PCM)를 1/6 가량 압축하여 스테레오 신호를 256 kbit/s에 실현되도록 하였다. 이들 비디오 신호와 오디오 신호는 동일한 시간 속성(time base)하에서 MPEG 시스템 계층에서 단일 스트림으로 결합된다.

공식적으로 ISO/IEC 11172인 MPEG-1 표준은 5 부분으로 되어 있다. 처음 3 부분은 각각 시스템, 비디오, 오디오 부분이고 나머지 두 부분은 성능 시험과 소프트웨어 시뮬레이션을 규정하고 있다.

MPEG-1은 여러 가지 변형된 모습으로 적용되고 있다. PC에서 실시간으로 동작하는 소프트웨어 형태로, 카드 형태로, 또 비디오 CD 형태로 구체화하고 있다. 특히 비디오 CD 디코더는 일부 국가에서 큰 성공을 거두고 있는데, 가령 중국에서는 금년에 2백만 개가 판매되었으며 내년에는 두배 가량 증가될 것이라고 한다.

MPEG-2

MPEG그룹에서 1990년 6월에 착공하여 서둘러서 두 번째의 표준은 MPEG-2이다. 이렇게 서두르게 된 것은 MPEG-1의 비디오부분에서 화질의 손실이 막대하여 이를 조속히 TV 수준으로 회복시킬 필요성이 있었고, 그 당시의 VLSI 기술 수준이 이미 10 Mbit/s정도의 디지털 신호를 처리할 수 있었기 때문이었다.

기본적으로 낮은 속도로 동화상을 디스크에 저장시키기 위한 MPEG-1과는 달리 MPEG-2는 훨씬 광범위한 응용 분야에 적용시키도록 MPEG 그룹에 강요하였다. 그 결과 특히 비디오 부분에서는 여러 응용 분야에 각기 다른 코딩 도구(툴)를 적용하도록 하여 매우 복잡한 모습을 띠게 되었다. “프로파일”이라 불리는 상이한 코딩 툴은 각기 다른 분야에 적용하도록 하였으며, 각각의 프로파일들은 상이한 “레벨”(이러테면 화

면의 크기)을 갖고 있다. 다음 표는 MPEG-2 비디오의 프로파일과 레벨을 요약한 것이다.

〈표 -1〉 MPEG-2 Profile/Level 도표

	Simple Profile	Main Profile	SNR Scalable Profile	Spatially Scalable Profile	High Profile	4:2:2 Profile
High Level		X			X	
High-1440 Level		X		X	X	
Main Level	X	X	X		X	X
Low Level		X	X			

MPEG-2 오디오는 단지 MPEG-1 오디오를 멀티채널로 확장한 것이다. MPEG-2 시스템 부분은 저장과 전송에 맞도록 각각 프로그램(program) 스트림과 전송(transport) 스트림 두가지로 나뉘어져 있다. 프로그램 스트림은 MPEG-1 시스템과 대응된다. 이것은 비교적 애러가 없고 소프트웨어 처리가 포함된 환경에서 사용되도록 설계되었다. 전송 스트림은 스트림 단위가 작게 나뉘어져 있어 각기 다른 통신의 전송속도에 적응하도록 하였고 잡음이 비교적 많은 환경에서 작동되도록 설계되었다.

MPEG-2는 매우 성공적인 표준안이다. MPEG-2를 적용한 제품들이 수백 만 개나 생산되고 있다. 디지털 위성 방송 수신기는 대표적인 제품이다. 그리고 장차 이 표준을 장착한 CATV, DVD 기기가 크게 보급될 것이다. 통신 표준을 담당하고 있는 ITU-T는 MPEG-2 시스템과 비디오 개발 과정에 참여하였다. 그 결과 광대역 화상통신에서 MPEG-2는 표준안으로 크게 확약할 것이다.

MPEG-2의 확장

MPEG-2는 매우 성공적인 표준안이기도 하

지만 기술의 발전에 따라 좀더 기능을 확장할 필요성이 대두되었다. 이것은 디지털 비디오의 저장 부문과 좀 더 향상된 음질에 대한 소비자의 욕구를 반영하는 것이다.

MPEG-2는 MPEG-1과 같이 5개 부분(시스템, 비디오, 오디오, 테스트 및 소프트웨어 시뮬레이션)으로 구성되어 있다. 그런데 MPEG-2는 정확하게 말해서 서버에서 사용자의 디코더까지를 표준화 대상으로 삼고 있다. 즉 통신기능에 초점을 맞춘 것이다. 그런데 Video on Demand나 홈쇼핑 등과 같이 초기에 예측하지 않던 분야가 크게 대두됨에 따라 10개 부분으로 확장하고 있다.

Part 6: DSM-CC(Digital Storage Media Command and Control) : 비디오 스트림을 관리하고 통제할 기능을 정의하는 이 규격은 원래 초기 MPEG 규격 제정시 작업 목록에 포함 된 것이었다. 그러나 당시 케이블업계와 위성방송업계의 조속한 MPEG-2 규정 완료 압력에 밀려 후일로 미루어졌다. MPEG-2 규격이 안정되자 다시 시작되고 있다. 1996년 6월에 공식적으로 표준화가 시작된 이 규격은 독립 환경(stand alone) 및 이기종 네트워크 환경 모두에서 어플리케이션을 지원하도록 되어 있다. DSM-CC는 서버에서 수집되는 정보와 클라이언트에서 전달되는 모든 정보를 DSM-CC 네트워크의 유저로서 관리한다.

Part 7: 비가역 (Non-Backward Compatible; NBC) 오디오 코딩 표준: MPEG-2는 MPEG-1의 오디오 표준을 사실상 그대로 인용하는 데 그쳐 그간의 기술 발전을 수용하지 못했다. 더구나 MPEG-2의 비디오 표준이 HDTV 화질까지 지원하는 데도 오디오 음질은 그에 상응하지 못하여 디지털 오디오 표준에서 Dolby 사의 AC-3 규격에 밀리는 수모를 당하게 되었다.

ITU는 음악에 대한 PCM 규격을 32 kHz, 48 kHz로 권고하여 대부분의 방송 기기들은 이 권고안을 따르고 있다. 그런데 필립스와 소니사는 콤팩트 디스크 규격을 44.1 KHz로 정하여 여러 문제를 야기하고 있다. MPEG-2 오디오는 ITU의 권고안에 맞추기 위하여 1997년 4월까지 새로운 표준안 완성을 위해 노력하고 있다.

Part 8: 이것은 원래 10 bit 비디오 코딩을 다루려고 하였으나 이 규격의 제정을 요청한 프로페셔널 비디오 편집 장비 제조사가 요청을 반려해 중단되고 있다.

Part 9: 실시간 인터페이스(Real-time Interface; RTI) 전송 스트림 디코더의 실시간 인터페이스에 대한 규격을 1996년 6월에 착수하였다. RTI는 전송 스트림 디코더를 이용하는 가전제품, 컴퓨터, 및 다른 제품에서 실시간 인터페이스를 제공하도록 장비 레벨에서 지원한다. 이 규격은 네트워크 적응 계층에서 버퍼 및 타이밍 회복 메커니즘을 제공함으로써 이를 달성할 것이다.

Part 10: DSM-CC의 테스트에 관한 규약이며 아직 진행이다.

MPEG-4

MPEG-4는 멀티미디어 통신을 위한 규격으로서 1993년 7월에 시작되었으며 1996년 11월 초안이 나오고 1998년 11월 국제 규격이 나올 예정이다. MPEG-4 표준화 작업 시작 당시에는 오늘날의 인터넷을 예측하지는 못했지만 MPEG-4는 오늘날 우리가 당면하고 있는 대부분의 문제를 해결해줄 것으로 기대하고 있다. MPEG-4는 다양한 스토리지와 전송 선로를 지원하기 위해서 가장 세련되고 정교한 기술이 모두 동원되었다.

MPEG-4는, 첫째, 인터넷 프로토콜인



정보기술의 최근동향(V)

TCP/IP 혹은 UDP/IP(데이터 전송을 위한 TCP/IP를 개량하여 연속적이고 동기적인 멀티미디어 데이터 전송에 적합한 전송 규약)를 지원하며 e-mail의 SMTP, 웹의 HTTP, 파일전송의 FTP 등과 같은 모드를 지원한다. 또 전화선 동축 케이블, 광섬유, 마이크로 웨이브 등의 전송 선로를 지원한다. 둘째, 오디오 비디오를 실시간 검색할 때 이를 충분히 지원하며 셋째, 네트워크 컴퓨터 등에서 필요한 기간만 소프트웨어를 다운로드할 시 이를 충분히 지원한다.

MPEG-4 비디오

MPEG-1과 MPEG-2에서는 화면 크기가 고정된 것을 전제로 하였다. MPEG-4에서는 유저의 디코더 성능에 따라 다양한 화면을 연출할 수 있도록 하였다. 즉 각기 다른 프로세서와 각기 다른 전송 선로를 사용하고 있다. 객체 지향형 기술을 도입하여 비디오 오브젝트(VO), 비디오 오브젝트 계층(VOL), 비디오 오브젝트 플레인(VOP)의 개념을 도입하였다. VOP는 VO의 인스턴스이며 VO와 VOP는 사용자가 액세스하고 조작할 수 있는 비트 스트림 내의 엔티티에 해당한다. 다시 말하면 사용자의 디코더 측에서 허용하는 값을 송신자에게 전송하면 송신자 측에서는 그에 대응하는 VO를 전송하고 VOP가 화면에 표시된다.

이와 같이 MPEG-4는 사용자 환경에 따라 프레임 수와 화면 크기를 임의적으로 조절할 수 있으며, 이 점이 이전 MPEG과의 가장 큰 차이점이자 장점이다. 또한 무전 전송과 같이 노이즈가 많은 환경에서도 안정적으로 오디오 비디오 정보를 전송할 수 있다.

MPEG-4 오디오

MPEG-4에서는 MPEG-2의 7번째 모드에서 채용한 비가역(Non-backward Compatible;

NBC) 오디오 모드를 채용하고 있다. 이것을 채용함으로써 MPEG-1에서 한 음악 채널 128 kbit/s로 셋팅한 것을 64 kbit/s 혹은 그 이하로 줄이도록 하였다. 더구나 매우 낮은 전송 속도에서도 오디오 정보를 전송하도록 하였는데, 이로써 인터넷, 셀룰러 통신, 위성, ISDN과 같이 다양한 통신 선로에서도 원활하게 전송된다.

기타

MPEG-4는 SNHC(Synthetic-Natural Hybrid Coding)이라고 하는 새로운 개념을 도입하였는데 이는 VRML을 원활하게 지원하기 위함이다. VRML은 인터넷상에서 가상 현실을 체험하도록 한 것으로서 미래의 인터넷 환경에서 매우 중요한 역할을 할 것으로 기대되는 언어이다.

결론

진정한 멀티미디어 통신은 MPEG-4의 등장으로 새로운 환경을 맞이할 것이다. Video-on-Demand, 홈쇼핑을 원활히 지원하고 멀티미디어 데이터베이스를 위한 DSM-CC기능이 MPEG-2에서 보강되었으며 CD의 오디오 규격에서 벗어나 새로운 오디오 표준(NBC)을 채택됨으로써 MPEG 표준은 21세기 디지털 정보 통신 환경을 위한 초석을 다지게 된다. **DC**