

소프트웨어 MPEG-1 재생 시스템 개발

박 태 강[†] · 이 호 석^{††}

요 약

기존의 하드웨어로 구현되었던 멀티미디어 시스템들의 다수가 CPU의 성능이 향상됨에 따라 소프트웨어로 구현이 가능하게 되었다. 또한, MPEG(Moving Picture Expert Group)은 디지털 동영상 압축 부호화의 표준화 기구로 자리잡고 있으며 MPEG-1과 MPEG-2가 상용화되어 있다. 본 논문에서는 윈도우 95 환경 하에서 전용 하드웨어의 도움 없이 MPEG-1 동영상을 소프트웨어로 재생할 수 있는 소프트웨어 MPEG-1 재생 시스템의 개발에 대하여 소개한다.

A Development of Software MPEG-1 Playback System

Tae-Kang Park[†] · Ho-Suk Lee^{††}

ABSTRACT

The MPEG is the standard of coding moving pictures, and MPEG-1 and MPEG-2 are widely in use. And as the performance of CPU has improved, it is now possible to implement MPEG-1 in software only. This paper introduces the techniques of developing MPEG-1 playback system in software without using hardware on Windows 95 operating system. The implemented MPEG-1 playback system shows an acceptable performance.

1. 서 론

MPEG(Moving Picture Expert Group)이란 멀티미디어 환경에서 가장 핵심이 되는 기법인 디지털 동영상의 압축 부호화를 위한 국제적인 표준안이다. 이미 저장미디어를 위한 MPEG-1과 저장미디어 및 차세대 방송표준을 위한 MPEG-2가 상용화되었고, 휴대용 통신 단말기를 위한 MPEG-4 이외에도 MPEG-7이 일부 국내업체를 포함하여 일본, 구미선진국에서 활발히 연구중이다[1][2][3][11][12].

이와 같이 MPEG이 멀티미디어의 핵심이 되어가고 있는 가장 큰 이유는 최소의 화질손실로 뛰어난 압축율을 보인다는 점이다. 이러한 고밀도 압축은 저장미

디어를 절약할 수 있다는 큰 장점을 가지고 있지만 이를 원래대로 복원하기 위해서는 많은 처리과정을 거친다. 따라서 MPEG 스트림을 재생하기 위해서는 전용의 MPEG 재생 하드웨어가 필요하였다.

그러나 본 연구에서는 최근 발표된 고성능의 CPU를 이용해 소프트웨어로 MPEG-1을 재생할 수 있는 기법을 제시한다. 이미 Xing, SoftPEG, Active Movie 등과 같은 외산 소프트웨어들이 상용화되어 있지만 국내에서는 필적할 만한 연구가 이루어지지 못하였다. 본 연구에서는 윈도우 95 환경 하에서 고성능의 CPU와 주변기기들의 기능을 최대한 활용하여 MPEG-1 스트림을 소프트웨어로 재생할 수 있는 방법을 제시한다.

2. MPEG-1의 소개

MPEG-1은 국제적인 표준화 기구인 ISO와 IEC가

† 준 회원 : 호서대학교 대학원 컴퓨터학부
 †† 정 회원 : 호서대학교 공과대학 컴퓨터학부 교수
 논문접수 : 1998년 10월 26일, 심사완료 : 1999년 5월 24일

결성한 ISO/IEC JTC(Joint Technical Committee)의 전문가 그룹이 만든 것으로 CD(Compact-Disk)와 같은 저장미디어를 기반으로 하여 약 70분 정도의 동영상 재생하는 것을 목표로 만들어졌다. 특징을 요약하면 다음과 같다[3].

- 부호화 단위를 MB(Macro Block)으로 한다.
- DCT(Discrete Cosine Transform)에 의한 공간적 정보 압축을 행한다.
- DPCM(Differential PCM)에 의한 시간적 정보 압축을 행한다.
- 부호화 정보 발생확률의 편중을 이용해서 허프만 코드를 바탕으로 VLC(Variable-Length Coding)를 행한다.
- DCT 계수의 양자화(quantization) 스텝 제어에 의해 전체부호 발생량을 제어한다.
- 영상타입은 I픽처(Intra-coded picture), P픽처(Predictive-coded picture), B픽처(Bidirectionally Predictive-coded picture)가 있다.
- 이들을 독립적으로 재생될 수 있도록 그룹화(GOP: Group of Picture)하여 랜덤 액세스를 가능하게 한다.
- 화면형식(SIF: Source Input Format)은 해상도에 자유도를 갖되 NTSC와 PAL방식 모두에 친화성이 좋도록 한다.
- 비트율은 1.5Mbps로 제한한다.

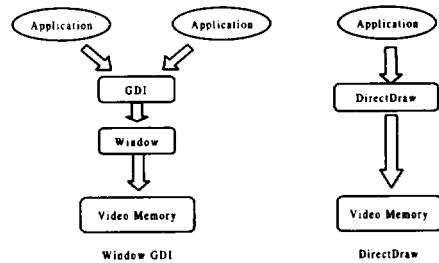
일반적으로 MPEG 영상의 화질은 동영상 전체를 디지털로 처리함으로써 일반 아날로그용 VTR과 동등, 혹은 그 이상의 화질을 볼 수 있는 것으로 알려져 있다.

3. 소프트웨어 MPEG 재생의 문제점

3.1 윈도우 95 GDI의 문제

일반적으로 윈도우 95 응용프로그램은 그래픽 함수를 처리하기 위해서 윈도우의 GDI(Graphic Device Interface)를 사용해야 한다. GDI는 비디오 출력 처리 전에 윈도우를 살펴봐야 하고, 또한 모든 응용프로그램이 공유하기 때문에 빠른 비디오 처리를 요구하는 응용프로그램에는 부적합하다. 특히, 동영상처럼 실시간 처리를 요구하는 작업에서는 더욱이 부적합하다. 이는 DirectDraw를[5][6][7] 적용하여 해결할 수 있다. DirectDraw는 GDI를 거치지 않고 비디오 메모리와 비디오 하드웨어에 직접 접근하므로 빠른 비디오 출력을

가능하게 한다[5][6][7][8].



(그림 1) GDI와 DirectDraw의 비교

3.2 YUV to RGB color space conversion

MPEG의 영상 데이터는 휘도(Y)와 색차(CbCr) 정보로 구성되어 있다. 이에 비해 윈도우 95는 RGB로 영상을 표시한다. 따라서 MPEG 영상을 윈도우 95에서 출력하기 위해서는 디코딩 한 후 얻어진 YCbCr (=YUV) 데이터를 RGB 데이터로 변환하여야 한다. 변환을 위해서는 각 픽셀마다 <표 1>의 공식을 적용해야 한다[1][2][4][14].

<표 1> YUV 컬러를 RGB 컬러로 변환하는 공식

$ \begin{aligned} R &= 1.164(Y - 16) + 1.596(Cr - 128) \\ G &= 1.164(Y - 16) - 0.813(Cr - 128) - 0.391(Cb - 128) \\ B &= 1.164(Y - 16) + 2.018(Cb - 128) \end{aligned} $
<p>MPEG-1 영상을 1초간 재생하기 위한 연산회수: $352 \times 240 \times 30(\text{frame/sec}) = 2534400 (\text{회/초})$</p>

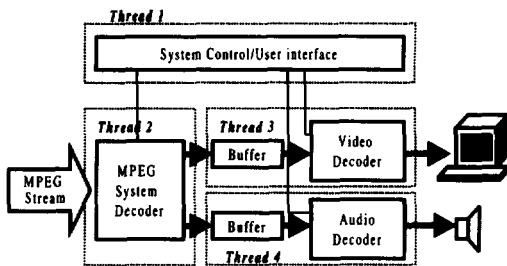
이러한 연산의 과부하는 VGA(Video Graphic Array) 하드웨어의 컬러공간 변환(color space conversion) 기능을 이용하여 해결할 수 있다. 최근 출시되는 VGA 하드웨어에는 제조업체마다 각기 다른 형식의 YUV 컬러공간을 RGB 컬러공간으로 변환하는 기능을 하드웨어로 지원한다. 이러한 기능은 FourCC(Four Char Code) 오버레이 영상으로 표준화되어 있으며, DirectDraw는 이를 사용할 수 있는 인터페이스를 제공한다. 즉, 컬러공간의 변환을 VGA 하드웨어에게 맡김으로써 연산량을 대폭 줄일 수 있고, 영상 확대에 따른 속도 저하 문제와 계단현상까지 해결할(VGA 하드웨어의 X/Y 보간 기능이 적용) 수 있다[9][10].

3.3 MPEG 영상의 가변적인 bit 발생

MPEG의 특징중의 하나가 매 프레임 마다 압축율이

다르다는 것이다. 또한, 압축기법으로 VLC(Variable-Length Coding) 기법이 적용되어 각 프레임을 디코딩하기 위한 연산량이 가변적이다[1][2][3]. 따라서 소프트웨어 MPEG 재생시스템을 순차 처리구조로 구현하면 유연한 영상을 볼 수 없다.

이는 다중 쓰레딩 기법을 이용하여 각 모듈이 비동기적으로 실행되도록 하면 해결할 수 있다. (그림 2)는 시스템 디코더, 비디오 디코더, 오디오 디코더, 사용자 인터페이스등 총 4개의 쓰레드로 구성된 소프트웨어 MPEG-1 재생시스템의 구조를 보여준다.



(그림 2) 다중 쓰레딩 기법을 이용한 소프트웨어 MPEG-1 재생시스템의 구성도

각각의 쓰레드는 시스템 전체를 제어하는 Thread 1에 의해 동기화되며 Thread 2, Thread 3, Thread 4는 이들 사이에 존재하는 버퍼를 이용해서 서로 비동기적으로 수행될 수 있다. Thread 1은 각 쓰레드들의 부하를 결정하는 파라미터를 설정하고, 이에 따라 이들의 우선 순위를 스케줄링 하여 시스템 자원을 효율적으로 사용하게 된다.

4. DirectDraw의 소개

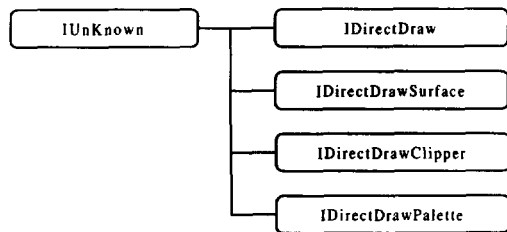
DirectDraw는 마이크로소프트 윈도우 환경 하에서 그래픽 성능 향상을 위해 제작된 Microsoft Game SDK인 DirectX의 2D 비디오 출력 요소이다[5][6][7]. DirectDraw는 응용프로그램이 윈도우의 GDI를 거치지 않고 비디오 메모리를 직접 접근하게 하여 고속의 비디오 출력이 가능하게 한다.

게임과 같은 빠른 처리를 위해 DirectDraw는 응용프로그램이 비디오 하드웨어를 독점할 수 있는 전체화면모드(full screen mode)를 지원하며, 사용자 인터페이스의 편의를 위해 일반 응용프로그램과 같은 창모드

(window mode)를 지원한다. 또한 VGA 하드웨어에서 지원하는 FourCC 오버레이 영상을 사용할 수 있는 인터페이스를 제공하므로 멀티미디어 응용프로그램에서는 필수적인 요소라 할 수 있다.

4.1 DirectDraw의 구조

DirectDraw는 COM(Component Object Model)객체[5]를 기반으로 만들어지며, IUnknown의 기본 클래스에서 파생되었다.



(그림 3) DirectDraw의 구조

IDirectDraw는 DirectDraw 객체이며, 디스플레이 장치를 표현한다. IDirectDrawSurface 객체는 화면에 디스플레이 하는데 사용되는 비디오 메모리 영역을 표현한다. IDirectDrawClipper 객체는 응용프로그램에서 특정 영역의 바깥 부분을 그리지 못하게 한다. 즉, 창모드로 수행되는 응용프로그램에 사용된다. IDirectDraw-Palette 객체는 색상표(palette)를 표현한다[6][7][8].

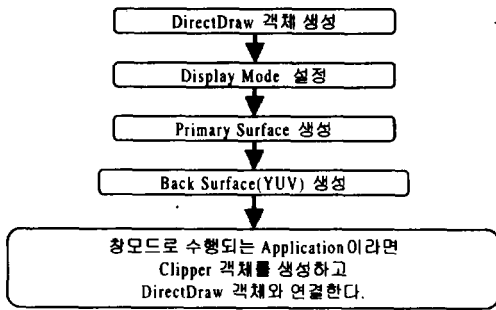
4.2 DirectDraw HAL(Hardware Abstraction Layer)과 HEL(Hardware Emulation Layer)

DirectDraw HAL은 디스플레이 장치의 하드웨어 가속기능을 제어한다. HEL은 하드웨어에서 지원하지 않는 기능을 소프트웨어로 제공한다. HAL과 HEL을 함께 사용하므로써 DirectDraw 응용프로그램은 필요한 함수를 하드웨어의 지원여부에 관계없이 사용할 수 있다.

4.3 DirectDraw Application의 생성

DirectDraw를 사용하기 위해서는 일련의 절차가 필요하며 그 절차는 (그림 4)와 같다.

그리고 이를 MPEG 재생기에서 사용하기 위해서는 VGA 하드웨어의 컬러공간 변환기능을 이용해야 하므로 약간의 추가적인 코딩이 필요하다[9][10].



(그림 4) DirectDraw 응용프로그램의 생성 과정

5. Video와 Audio의 동기화

소프트웨어 MPEG 재생시스템에서 가장 중요시되는 부분이 비디오와 오디오의 동기화이다. CPU 처리속도의 한계로 소프트웨어로 완벽한 동기화를 구현하는 것은 불가능하다. 그러나, msec(millisecond) 정도의 오차는 사람의 감각으로 판별할 수 없으므로 적절한 오차 한계를 설정하여 동기화를 구현할 수 있다.

5.1 동기화를 위한 기본 전략

CPU의 성능에는 한계가 있다. 또한, 윈도우 95와 같은 다중 쓰레딩 운영체제에서는 시분할(time sharing)로 인하여 시스템의 부하가 수시로 바뀔 수 있다. 이 때문에 CPU를 특정 응용프로그램에 고정 할당하는 것은 불가능하며, CPU가 충분히 빠르다고 해도 소프트웨어 기반의 MPEG 재생시스템에서는 디코딩 중에 특정 시점에서 디코딩을 위한 CPU 성능부족 현상이 발생할 수 있다. 이러한 문제는 특정 프레임(오디오 또는 비디오)을 디코드하지 않고 drop하는 것에 의해 해결할 수 있다[12].

B 프레임의 경우 다른 프레임에 영향을 주지 않으므로 쉽게 drop될 수 있다. P 프레임이 drop되면 이 프레임의 영향을 받는 B 프레임이 함께 drop되어야 한다. 이것은 부자연스러운 영상을 만들어 내며, I 프레임이 drop된 경우는 더욱 손상된 영상을 보게 될 것이다. 따라서 drop의 우선 순위는 B, P 프레임 순서가 되며 최악의 경우 I 프레임을 drop 시킬 것이다. 일반적으로 음성의 끊김에 의한 정보전달의 상실보다 화면의 끊김에 의한 정보전달의 상실이 적다는 점을 감안하여 본 연구에서는 끊김이 없는 오디오 재생을 기본으로 비디오 프레임의 drop 전략을 제시한다.

Drop 전략을 위한 시간 파라미터는 다음과 같다.

- **Tpts** : 현재 Display될 프레임의 Presentation Time Stamp
- **Tstc** : 현재의 System Timing Clock Count 값
- **Tacc** : n개의 B 프레임이 연속으로 Drop되어 누적된 Tstc와 Tpts의 차.
- **T_DIFF** : 실제 디스플레이될 시간과의 허용오차(상수)
- **T_DROP_P** : P 프레임 drop을 위한 허용한계 시간(상수)
- **T_DROP_I** : I 프레임 drop을 위한 허용한계 시간(상수)

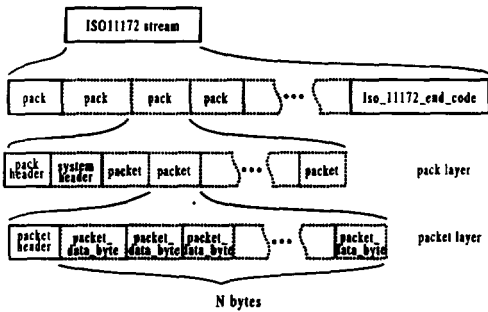
Drop 전략을 위한 의사코드는 다음과 같다.

1. Display : Tpts가 허용오차 내에 있다면 디스플레이 한다.
 if ((Tstc-T_DIFF)<=Tpts && Tpts<=(Tstc+T_DIFF)) {
 display();
 Tacc = 0;
 }
2. B 프레임 Drop : Tpts가 허용오차를 넘어서면 B 프레임을 drop하고 Tstc와 Tpts의 차를 누적한다.
 else if((Tstc+T_DIFF)>=Tpts) {
 drop_B_frame();
 Tacc += (Tstc-Tpts);
 }
3. P 프레임 Drop : Tacc가 T_DROP_P 보다 크다면 P 프레임을 drop한다.
 if(Tacc > T_DROP_P) {
 drop_P_frame();
 Tacc = 0;
 }
4. I 프레임 Drop : Tacc가 T_DROP_I 보다 크다면 I 프레임을 drop한다.
 if(Tacc > T_DROP_I) {
 drop_P_frame();
 Tacc = 0;
 }

5.2 동기화를 위한 정보

MPEG-1 스트림은 데이터의 일관성 및 정보유지, 전송 및 저장매체의 특징 등을 고려하여 (그림 5)와 같은 계층 구조로 이루어져 있다. Pack 계층 내에는 비디오 패킷과 오디오 패킷이 불규칙적으로 섞여 있으며, 동기화를 위한 시간정보는 각각의 패킷 헤더에 포함된 DTS(Decoding Time Stamp)와 PTS(Presentation Time Stamp) 파라미터이다. 이들 시간 정보는 부호기측의 STC(System Timing Clock)라 불리는 90kHz의

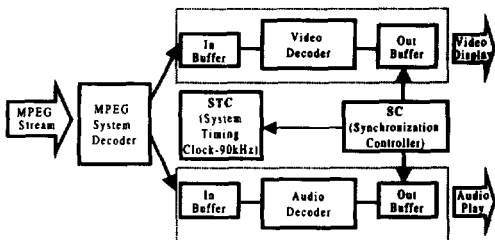
system clock에 의해 결정된 값을 가지고 있으며, 각각의 패킷이 디코딩 되어야 할 시간과 화면출력되어야 할 시간을 나타낸다. DTS와 PTS 파라미터는 일반적으로 그 값이 같을 경우 PTS만 기록되어 있고, 다를 경우 DTS와 PTS가 함께 기록된다. 이들 정보는 정확한 디코딩 타이밍과 화면출력 타이밍을 결정하기 위해 필요하지만 DTS의 경우는 실시간 처리가 불가능한 소프트웨어 MPEG-1 재생시스템에서는 부적합하다[1][3].



(그림 5) MPEG-1 시스템의 계층구조

5.3 동기화의 실현

(그림 6)의 기본 구조는 (그림 2)과 같으며, 동기화를 위한 STC(System Timing Clock)와 PBC(Playback Controller)를 가시화 하여 보여주고 있다. 비디오 디코더와 오디오 디코더는 (그림 2)의 Thread 1의 스케줄링에 의해 비동기적으로 동작하여 디코딩된 프레임출력버퍼에 저장한다. 출력버퍼에 저장된 프레임은 PTS를 포함하고 있다. 이때 PBC는 STC를 참조하여 (5.1)절에서 설명한 drop 전략에 의해 각 프레임을 출력하도록 제어한다.



(그림 6) 비디오/오디오 동기화를 위한 시스템 구조

정확한 STC의 구현을 위해서는 약 0.01msec(90kHz)의 정확도가 필요하지만 CPU의 속도제한으로 사실상

불가능하다. 현실적으로 기존의 PC환경하에서 10msec 이하의 제어는 어렵다. STC를 참조하여 출력을 제어하는 PBC 또한 많은 오차를 유발한다. 그러나, 사람의 감각으로는 10msec 단위의 오차는 감지하기 어렵다. 실제 구현된 실험 시스템에서 10~20msec 범위의 오차한계 파라미터(TDIFF)를 설정하여 비교 실험한 결과 하드웨어 MPEG-1 재생시스템과 화면 품질의 차이를 구분하기 어려웠다.

6. 성능 측정

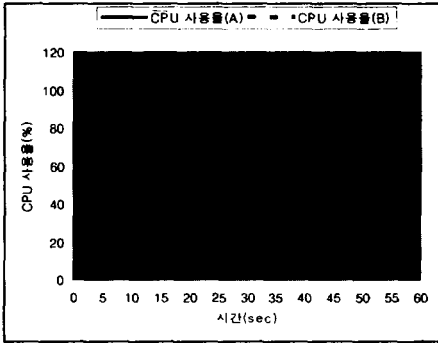
성능 측정은, 비디오와 오디오의 동기화를 구현한 완벽한 MPEG-1 재생 시스템이 공개되어 있지 않기 때문에, 버클리 소프트웨어 MPEG-1 비디오 재생시스템의 Win32 포팅버전[13]을 대상으로 하였다. 이 소프트웨어는 단일 쓰레드로 구성되어 있으며, 디스플레이 부분에서 윈도우의 GDI를 사용한다.

다음은 펜티엄 MMX 200MHz의 시스템에서 MPEG-1 비디오 샘플을 1분간 재생하면서 5초 간격으로 CPU 사용율과 프레임율을 관찰한 결과이다. 그림에서 (A)는 본 연구에서 개발한 재생 시스템을 나타내고, (B)는 버클리 MPEG-1 재생 시스템을 나타낸다.

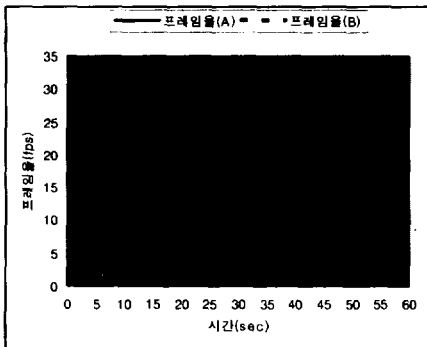
(그림 7)과 (그림 8)에서 볼 수 있듯이 멀티스레딩 기법과 DirectDraw를 사용한 시스템 (A)의 경우 평균 63.4%의 CPU 사용율과 29.2fps(frame per sec)의 프레임율을 유지한다. 하지만 단일 쓰레드에 DirectDraw를 사용하지 않은 시스템 (B)의 경우에는 평균 99.8%의 CPU 사용율과 23.8fps의 비교적 불규칙한 프레임율을 보이고 있다. 더욱이 전체화면 모드로 실행시킨 경우에는 시스템 (A)가 5%정도의 CPU 사용율만 증가하고 평균 29.0 fps의 프레임율을 보여주는데 반하여, 시스템 (B)는 거의 100%의 CPU 사용율과 함께 프레임율은 7fps 이하로 떨어졌다.

실험에 사용된 MPEG-1 재생 시스템은 기존에 상용화된 MPEG-1 재생 시스템에 비해, 다소의 성능차이를 보인다. 이것은 본 실험에 사용된 MPEG-1 비디오 재생 소스[13] 화면 출력을 위한 모든 계산을 그대로 수행하는 반면에 상용화된 MPEG-1 재생 시스템은 화질에 영향을 끼치지 않는 범위 내에서 내부 계산 과정을 생략하는 방법을 택하고 있기 때문이다. 즉, 예를 들면 IDCT(Inverse Discrete Cosine Transform)에서 계산양에 비하여 영상의 변화에 미세한 영향을 미치는

데이터를 무시하거나 혹은 양자화 스케일의 제어를 통하여 눈으로 쉽게 감지하기 어려운 정도로 화질을 삭감하여 유연한 재생을 얻는 것이다.



(그림 7) CPU 사용율의 비교



(그림 8) 프레임율의 비교

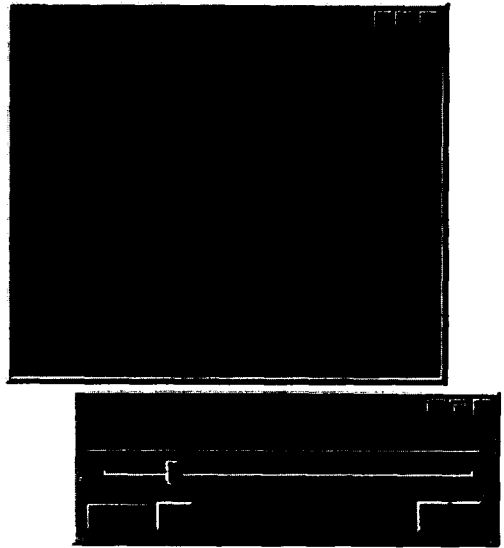
7. 결 론

하드웨어의 성능향상에 힘입어 제작상의 저렴한 비용, 간편한 설치의 이점으로 멀티미디어 도구들이 소프트웨어로 구현되고 있다. 그러나 범용 시스템에서 구현하기에는 많은 제약이 따른다. 본 논문에서는 멀티미디어에서 동영상 압축 부호화의 표준인 MPEG-1의 재생을 소프트웨어로 개발하기 위한 기법들을 제시하였다.

현재 이러한 기법들을 적용한 소프트웨어 MPEG-1 재생시스템의 실험 모델이 구현되었으며, 펜티엄 MMX-200MHz 시스템에서 평균 24fps의 프레임율을 나타내었다. (그림 9)는 개발된 MPEG-1 실험 모델의 실행모

습이다.

현재 개발된 실험 모델은 MPEG-1 재생을 위한 기본적인 기능을 제공한다. 추후에 random access, 재생 속도 선택, 영상 조절 등과 같은 사용자 편의를 위한 기능이 추가될 것이다.



(그림 9) 개발된 실험 모델의 재생모습

참 고 문 헌

- [1] Joan L. Mitchell, William B. Pennebaker, Chad E. Fogg & Didier J. LeGall "MPEG Video Compression Standard," Chapman & Hall, 1997.
- [2] Prabhat K. Andleigh, Kiran Thakrar, "Multi-media systems design," Prentice-Hall, Inc., 1996.
- [3] 정제창 번역, "최신 MPEG", (주)교보문고, 1995.
- [4] Phillip E. Mattison, "Practical Digital VIDEO with Programming Example in C," John Wiley & Sons, Inc., 1994.
- [5] Michael Stein, Eric Bowman, Gregory Pierce, "Direct3D Professional Reference," New Riders Publishing, 1997.
- [6] Jason Kolb, "Win32 Game Developer's Guide with DirectX 3," Waite Group Press™, 1997.
- [7] Clayton Walnum, "Windows 95 Game SDK Strategy Guide," QUE Corporation, 1995.

- [8] Microsoft Corp, "Microsoft DirectX," <http://www.microsoft.com/directx>.
- [9] Microsoft Corp, "FOURCC for Video Compression," <http://www.microsoft.com/hwdev/devdes/fourcc.htm>.
- [10] Web Art, "The (Almost Definitive) FOURCC Definition List," <http://www.webartz.com/fourcc>.
- [11] MPEG.ORG, "MPEG Pointers and Resources," <http://www.mpeg.org>.
- [12] Gabriel Zuniga and Ephraim Feig, "Synchronization Issues on Software MPEG Playback System," IBM T.J. Watson Research Center Yorktown Heights, NY 10598, 1995.
- [13] Thanassis Tsiodras, "Programming Multimedia and 3D in realtime," <http://www.image.ece.ntua.gr/~ttsiod>.
- [14] 박태강, 이호석, "DirectDraw를 이용한 Software MPEG1 Video Player 개발", 한국정보과학회 '98 봄 학술발표대회 논문집(B), pp.527-529, 1998. 4.



이 호 석

e-mail : hslee@dogsuri.hoseo.ac.kr

1983년 서울대학교 전자계산학과 졸업(공학 학사)

1985년 서울대학교 대학원 전자계산학과 졸업(공학 석사)

1993년 서울대학교 대학원 컴퓨터공학과(공학 박사)

1994년~현재 호서대학교 컴퓨터공학부 조교수

관심분야 : 멀티미디어, 이미지처리, 데이터베이스, 컴퓨터그래픽스, 프로그래밍 언어 등



박 태 강

e-mail : carypark@mmlab.hoseo.ac.kr

1997년 독학학위과정 전자계산학과 학위 취득(학사)

1999년 6월 현재 호서대학교 대학원 컴퓨터공학과 석사과정 재학중

관심분야 : 멀티미디어, 컴퓨터그래픽스, 컴퓨터네트워크, HCI 등