

다중 쓰레딩 기법의 MPEG-1 디코더에서 공유버퍼를 이용한 오디오/비디오 스트림의 동기화

박태강, 이호석
호서대학교 컴퓨터공학과

Synchronization of audio and video streams on multi-threading MPEG-1 decoder using shared buffers

Tae-Kang Park, Ho Suk Lee
Department of Computer Engineering, Hoseo University

요약

소프트웨어로 MPEG-1 디코더의 구현이 가능하다. 소프트웨어 MPEG-1 디코더의 문제 중 하나는 MPEG-1 압축 알고리즘의 특징상 각각의 영상들이 서로 다른 압축율로 압축되기 때문에 재생 시에 디코더에 걸리는 부하가 매우 불규칙적이라는 점이다. 이 문제는 MPEG-1 디코더를 보다 작은 실행 단위인 쓰레드로 나누어 처리함으로써 효율적으로 해결할 수 있다. 이때 독립적인 실행 흐름을 가지는 쓰레드들간의 데이터 전달을 위하여 공유버퍼를 사용하게 된다. 본 논문에서는 다중 쓰레드로 구성된 소프트웨어 MPEG-1 디코더에서 쓰레드들 간의 데이터 전달에 사용되는 공유버퍼를 이용하여 오디오와 비디오 스트림의 동기화를 효과적으로 수행하는 기법을 소개한다.

1. 서론

MPEG(Moving Picture Experts Group)이란 멀티미디어 환경에서 가장 핵심이 되는 기법인 디지털 동영상의 압축 부호화를 위한 국제적인 표준안이다[1][2]. MPEG-1에 이어 최근 MPEG-2가 상용화되었으며 CPU 및 주변기기의 성능향상으로 소프트웨어 MPEG 디코더들이 등장하고 있다.

MPEG 압축알고리즘은 동영상의 각 프레임(Frame)을 I-프레임(Intra Frame), P-프레임(Predictive Directional Frame), B-프레임(Bidirectional Frame)으로 분류하여 각각의 프레임에 서로 다른 압축기법을 적용함으로써 고능률의 압축을 실현한다. 기준 프레임이 되는 I-프레임의 경우는 프레임내의 화소정보만으로 압축을 수행하기 때문에 압축 알고리즘이 비교적 단순하고 압축율이 떨어진다. P-프레임과 B-프레임은 프레임내의 화소 정보뿐만 아니라 인접 프레임을 참조하는 움직임 예측(Motion Estimation)과 움직임 보상(Motion Compensation)기법을 사용함으로써 보다 높은 압축율을 실현하는 반면에 매우 복잡한 처리과정을 거친다[1][2]. 이 때문에 이를 재생하는 디코더 측에서는 매 프레임을 디코딩할 때 처

리의 부하가 매우 가변적이다.

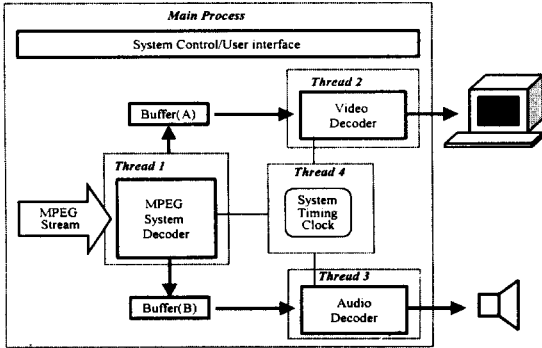
최근 보급되고 있는 가정용 PC(Personal Computer)는 MPEG 동영상을 디코딩하기에 충분한 성능을 가지고 있지만 이를 운용하는 운영체제의 멀티태스킹(Multi-Tasking)으로 인하여 CPU의 부하가 가변적이고, CPU를 특정 프로세스에 고정 할당하는 것이 불가능하다.

이런 이유들로 인해서 소프트웨어 MPEG 디코더를 순차처리구조로 제작한다면 유연한 영상을 얻기 어렵다. 이 문제를 해결하기 위해서 MPEG 디코더를 보다 작은 실행단위인 쓰레드로 나누어 각각의 쓰레드가 시스템의 부하에 효율적으로 대응할 수 있도록 구성할 수 있다. 이때 독립적인 실행흐름을 가지는 쓰레드들 간에 데이터 전달을 위하여 공유버퍼가 존재하게 된다[5][6].

본 논문에서는 다중 쓰레드로 구성된 소프트웨어 MPEG-1 디코더에서 쓰레드들 간의 데이터 전달을 위한 공유버퍼를 이용해서 오디오와 비디오 스트림의 동기화를 구현하는 효과적인 기법을 소개한다.

2. 실험에 사용된 소프트웨어 MPEG-1 디코더의 구성도

(그림 1)은 시스템 클럭, 시스템 디코더, 비디오 디코더, 오디오 디코더의 5개의 부가적인 쓰레드로 구성된 소프트웨어 MPEG-1 디코더의 구조를 보여준다. 각각의 쓰레드는 시스템 전체를 제어하는 메인 프로세스에 의해 동기화되어 수행되며 Thread 1, Thread 2, Thread 3은 이들 사이에 존재하는 공유 버퍼를 이용해서 MPEG-1 오디오 스트림과 비디오 스트림의 동기화를 수행하고 서로 독립적으로 수행될 수 있다. Thread 4는 시스템 클럭(System Timing Clock: STC)으로써 오디오와 비디오 데이터의 디코딩 시점과 출력 시점을 결정하는 역할을 한다[1][4][5].

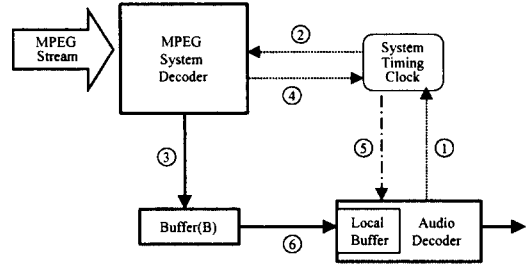


(그림 1) 다중 쓰레딩 기법을 이용한 소프트웨어 MPEG-1 디코더의 구성도

3. 공유버퍼의 동시접근 문제 해결

(그림 1)에서 Buffer(A)와 Buffer(B)는 시스템 디코더 그리고 오디오/비디오 디코더에 의하여 동시 접근된다. Buffer(B)를 예를 들면, 시스템 디코더는 분리된 오디오 패킷을 Buffer(B)에 쓰려는(Write) 동작을 반복할 것이고 오디오 디코더는 Buffer(B)에서 패킷을 읽으려는(Read) 동작을 반복할 것이다. 이 과정에서 문제가 발생할 수 있는 한가지 경우는 시스템 디코더가 Buffer(B)에 데이터를 쓰고 있는 동안 오디오 디코더가 읽으려고 시도하는데서 발생할 수 있다.

이 문제를 해결하기 위한 가장 쉬운 방법은 이들 버퍼를 임계구역(Critical Section)으로 설정하고 각각의 디코딩 쓰레드들이 서로 경쟁하도록 하는 것이다. 하지만 이 방법은 쓰레드간의 경쟁으로 인한 과부하 때문에 분리된 패킷 데이터가 각각의 디코더로 정해진 시간(Display Time Stamp : DTS) 내에 전달되지 못할 수 있다. 다른 효율적인 해결방안은 데이터 패킷(Data Packet)의 흐름 방향이 일정하다는 점을 감안하여 디코딩 쓰레드들의 공유버퍼에 대한 접근을 순서화 시키는(Serialize) 것이다.



(그림 2) 오디오 데이터 공유버퍼의 제어구조

(그림 2)는 오디오 데이터 버퍼에 대한 동시접근 문제를 순서화 시킨(Serialize) 제어구조를 보여주고 있다. 최초 오디오 디코더는 STC(System Timing Clock)의 데이터 요청 루틴을 활성화시키고 자신은 수면(Sleep)상태로 된다①. STC의 데이터 요청 루틴은 시스템 디코더에게 데이터를 요청하게 되고 ② 시스템 디코더는 MPEG-1 스트림에서 하나의 패킷을 분리해내어 Buffer(B)에 쓰고(Write)③ STC에게 버퍼가 채워졌음을 알린④ 후 자신은 유휴(Suspend)상태가 된다. 시스템 디코더로부터 버퍼가 채워졌음을 통보 받은 STC는 DTS에 맞추어 수면상태에 있는 오디오 디코더를 깨운다(Wakeup)⑤. 수면에서 깨어난 오디오 디코더는 Buffer(B)의 데이터를 자신의 지역(Local)버퍼에 복사한⑥ 후 다시 STC의 데이터 요청 루틴을 활성화시킨다①. 오디오 디코더는 자신의 지역버퍼에 읽어들이는 데이터를 디코딩하며, 데이터를 모두 소모하면 수면상태가 된다. 오디오 디코더가 디코딩을 수행하는 동안 STC는 시스템 디코더에게 데이터를 요청하게 되고② 시스템 디코더는 또 다른 패킷을 분리해내어 Buffer(B)에 쓰고(Write)③ STC에게 버퍼가 채워졌음을 알린 후 자신은 유휴상태가 된다④.

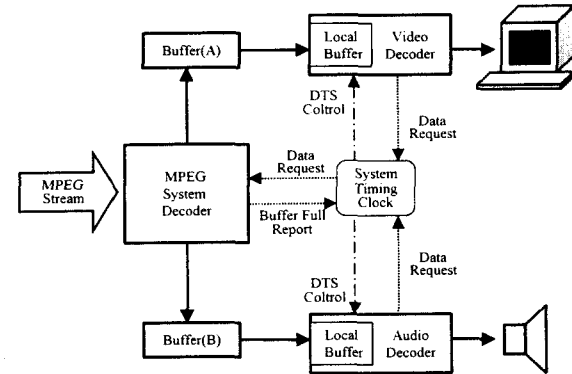
이러한 제어구조는 오디오 디코더와 시스템 디코더가 동시에(concurrent) 동작하면서도 Buffer(B)에 대한 접근을 경쟁 없이 순서화 시키는(Serialize) 역할을 한다. 비디오 디코더도 동일한 제어구조를 가진다[6].

4 동기화의 구현

(그림 3)에 본 연구에서 실험을 위해 개발한 소프트웨어 MPEG-1 재생시스템의 전체적인 구조가 표시되어 있다. 각각의 디코더와 STC(System Timing Clock)는 서로 다른 쓰레드로 수행되며 최초 이들은 수면(Sleep)상태 또는 대기(Suspend)상태이다. 사용자의 재생(Play) 요청에 의해 STC의 데이터 요청 루틴이 동작되고 이것을 신호로 앞 3절에서 설명한 제어구조로 비디오 디코더와 오디오 디코더가 연속적으로 동작한다.

STC는 지속적으로 각각의 쓰레드들의 요청에 반응하여 적절한 제어신호를 전달해 주며 오디오와 비디오 데이터를 패킷 단위로 적절한 디코딩 시간(DTS)에 각각의 디코더에게

전달될 수 있도록 제어한다. 오디오 디코더와 비디오 디코더가 패킷 데이터를 지역버퍼(Local Buffer)에 복사해와서 디코딩을 수행하기 때문에 이들 디코더에 유희시간이 생긴다면 시스템 디코더는 패킷을 분리하여 Buffer(A) 또는 Buffer(B)를 채울 수 있다.



(그림 3) 동기화가 적용된 소프트웨어 MPEG-1 디코더의 구조

만일 시스템의 과부하로 패킷이 적절한 시간 내에 디코딩되지 못한다면 영상의 끊김과 음향의 끊김을 느끼게 될 것이다. 이 경우 일반적으로 인간의 감각 중 청각보다는 시각이 둔감하다는 것을 감안하여 비디오 디코더 측에서 특정 비디오 프레임을 디코딩하지 않고 건너뛰는 프레임 드롭(Frame Drop) 기법이 사용된다[3]. 프레임 드롭은 STC에 의해 결정되며 프레임 드롭 상황이 발생할 경우 STC는 비디오 디코더에게 알려주게 된다.

5. 결론

범용 PC의 성능향상에 힘입어 소프트웨어 MPEG-1 디코더들이 등장하고 있다. 하지만 압축된 MPEG-1 스트림의 가변적인 비트 할당과 범용 운영체제의 멀티태스킹으로 인하여 소프트웨어 MPEG-1 디코더를 순차처리 구조로 설계하는 것은 비효율적이다. 이러한 구조는 아주 강력한 성능의 시스템이 아니라면 시스템 부하가 과도한 경우에는 오디오와 비디오 스트림의 동기화를 위해 많은 비디오 프레임을 드롭해야 하고 반대의 경우에는 시스템이 유희상태가 되기 때문에 비효율적이다.

쓰래드는 이러한 문제를 해결할 수 있는 좋은 방안이며 MPEG-1 디코더를 다수의 쓰래드로 분리하여 시스템 부하를 분산함으로써 보다 유연한 동작을 얻을 수 있다. 하지만 자신만의 실행흐름을 가진 쓰래드들은 데이터 전달을 위해서 공유버퍼를 사용해야 하며 이것은 임계구역(Critical Section)을 만들어 낸다. 본 논문에서는 쓰래드간의 경쟁이 될 수 있는 공유버퍼에 대한 접근을 순서화(Serialize)시켜 쓰래드간의 경쟁으로 인한 부하를 없앤 공유버퍼 제어구조를 제안하였다.

그리고 이러한 제어구조를 이용하여 오디오와 비디오 스트림의 동기화를 효과적으로 구현할 수 있다.

참고문헌

- [1] Joan L. Mitchell, William B. Pennebaker, Chad E. Fogg & Didier J. LeGall "MPEG Video Compression Standard", Chapman & Hall, 1997.
- [2] 정제창 번역, "최신 MPEG", (주)교보문고, 1995.
- [3] Gabriel Zuniga and Ephraim Feig, "Synchronization Issues on Software MPEG Playback System", IBM T.J. Watson Research Center Yorktown Heights, NY 10598.
- [4] 박태강, 이호석, "DirectDraw를 이용한 Software MPEG-1 Video Player 개발", 한국정보과학회 '98 봄 학술발표대회 논문집(B), pp. 527-529, 1998.4.
- [5] 박태강, 이호석, "Software MPEG-1 재생 시스템을 위한 Audio/Video 스트림의 동기화", 한국정보과학회 '98 가을 학술발표대회 논문집(II), pp. 303-305, 1998.10.
- [6] 박태강, Software MPEG-1 재생 시스템 개발, 호서대학교 컴퓨터공학부 석사학위논문, 1999.8.