

타일드-디스플레이 시스템에서 실시간 동영상 상영기의 설계 및 구현

최정훈[○] 최기석 유정수 낭종호

서강대학교 컴퓨터공학과

drumist@mlneptune.mlneptune.sogang.ac.kr brix@mlneptune.mlneptune.sogang.ac.kr

yjs@mlneptune.mlneptune.ac.kr jhnang@sogang.ac.kr

Design and Implementation of a Realtime Video Player on Tiled-Display System

Jeonghoon Choi[○] Gisoek Choe Jeongsoo Yoo Jongho Nang

Dept of Computer Science and Engineering, Sogang University

본 논문에서는 높은 해상도를 제공하기 위하여 여러 대의 PC와 모니터를 연결한 타일드-디스플레이(Tiled-Display) 시스템 상에서 동작하는 실시간 동영상 상영기를 설계 및 구현하였다. 제한한 동영상 상영기는 하나의 동영상을 기가비트(Giga bit) 폐쇄 이더넷으로 연결된 여러 PC에 UDP 멀티캐스트를 사용하여 전송하고 각 수신기는 받은 동영상 데이터의 압축을 푼 후 이미지를 자신의 영역에 분할하여 시간적인 동기화를 맞추어서 재생할 수 있도록 설계되었다.

먼저 동영상을 전송하는 방법은 모든 상영기가 각각 다른 영역을 디스플레이해야 하는 것을 고려해서 동영상 데이터를 압축을 푼 상태에서 각 상영기에 각기 다른 동영상 데이터를 전송할 경우 발생하는 네트워크 부하를 최대한 감소시키기 위해 그리고 압축을 푼 데이터를 각각의 나누어진 이미지로 다시 압축해서 전송할 경우 발생하는 송신측 PC의 시스템 부하를 적게 하기 위해, 또한 하나의 동영상 데이터 스트림을 네트워크에 연결된 모든 상영기에 쉽게 방송하기 위해 동영상의 압축된 데이터를 그대로 DUP 멀티캐스트를 사용해서 전송하는 방법을 택하였다. 압축된 동영상 데이터를 받은 수신기는 데이터를 디코딩한 후 자신의 영역만 클리핑해서 디스플레이 한다.

분할된 동영상을 동기화 하여 재생하는 방법으로는 어느 한 PC가 기준이 되어 나머지 PC에게 동기화 정보를 주기적으로 전송하는 방법이 있는데, 이러한 방식은 별도의 동기화 정보를 전송하는데 대한 오버헤드를 발생시키며, 특히 동기화 정보 전송의 딜레이차이로 인하여 정확한 동기화를 맞추기 어렵다. 여러 PC에서 동시에 재생하는 각 상영기들의 동기화는 스트림 타임(stream time)과 타임스탬프, 리퍼런스클럭을 통해서 이루어진다. 비디오의 재생은 스트림 타임과 동영상 프레임의 타임스탬프의 비교를 통해서 스케줄링 되며, 마스터로부터 전송되는 타임스탬프는 모든 상영기에서 동일하다. 따라서 각 상영기를 동기화하기 위해서는 스트림 타임을 동일하도록 하여야 하며, 이를 위해서는 각 상영기에서의 재생 시작 시간과 리퍼런스클럭의 속도를 동일하게 해주어야 한다. 재생 전 각 송신기는 글로벌 클럭과 각 상영기의 로컬 클럭을 Cristian's Algorithm[1] 방식에 의하여 동기화 한다. 송신기와 모든 상영기간의 클럭 동기화를 맞춘 후 방송을 시작하기 전 각 상영기가 재생을 시작해야 할 시간에 대한 글로벌 클럭의 값을 각 상영기에게 알려준다. 이와 같은 방식으로 각 상영기는 스트림 타임을 동일하게 유지함으로써 동기화를 맞추게 된다. 이러한 방법은 주기적으로 동기화 정보를 전송하는 방법에 비하여 오버헤드가 적으며 시작점에서만 동기화를 수행하므로 오차가 발생할 확률이 적다. 본 논문에서 제안한 동기화 방법은 각 상영기의 재생 시작 시간을 동기화한 후 각 상영기의 리퍼런스클럭의 속도가 동일하도록 하드웨어클럭을 사용하고 이에 맞추어서 동영상 프레임들을 상영하도록 하는 것이다. 추가적으로, 동영상 방송 시 각 PC의 상영기에서 발생하는 언더플로우 및 오버플로우를 방지하기 위하여 압축된 동영상의 비트레이트에 맞추어서 방송하는 방법을 설계하고 구현하였다. 본 시스템은 여러 압축 형식 및 전송 형식에 대한 종속성을 탈피하기 위하여 Microsoft DirectShow[2] 필터 구

조를 이용하여 구현하였다.

미디어 데이터 전송 시 수신 버퍼의 언더플로우 및 오버플로우를 방지하기 위해서는 전송 중 흐름 제어를 통해 데이터양을 조절하여야 한다. 이러한 방식의 전송 시 최대 버스트 크기는 최대 미디어 샘플 사이즈와 동일하며, 대부분의 미디어 포맷에서 최대 미디어 샘플 사이즈는 스트림을 끝까지 읽어보기 전에는 알 수 없는 경우가 많으므로 초당 최대 비트레이트 / 평균 프레임레이트를 통해 근삿값을 사용한다. 그리고 소켓으로부터 데이터를 읽는 일을 담당하는 쓰레드를 두어 소켓 인터페이스에서 발생하는 이벤트에 빠르게 반응할 수 있도록 하여서 디코딩 및 렌더링 작업에 블러킹되지 않고 안정적으로 소켓 버퍼의 데이터를 소비할 수 있도록 한다. 또한 버스트 모드의 전송으로 인한 지터 발생과 미디어 스트림의 비트레이트 변동으로 인한 수신율의 불규칙성은 상영기의 일시적인 언더플로우 및 오버플로우를 발생시킬 수 있다. 이 문제를 해결하기 위해 버퍼를 두어 디코딩 및 렌더링작업이 언더플로우시 블러킹되는 현상을 최소화할 수 있도록 한다. 버퍼링 양은 본 시스템 상에 적합한 초기 버퍼링시간을 실험을 통하여 구한 후 초기 버퍼링 시간 \times 평균 비트레이트 만큼의 크기로 초기화 하며, 재생 중 버퍼 량을 모니터링 하여 크기를 조절한다. 재생 미디어의 특성은 끝까지 전부 재생하기 전까지는 완전히 알 수 없으며, 미래의 네트워크 혼잡도는 정확히 예측할 수 없다. 하지만 실험을 통해 버퍼가 없거나 0.25초 이하의 버퍼링을 수행할 경우 재생품질이 좋지 않다는 것을 알 수 있다. 이러한 예측할 수 없는 상황과 실험 결과를 고려하면, 안정적인 재생 품질을 보장하려면 적어도 0.25초를 초과하는 버퍼링이 필요하다는 것을 알 수 있다. 본 논문에서는 안정적인 재생 품질과 사용자가 느끼는 버퍼링 딜레이를 고려하여 1초의 버퍼링 초기 값을 선택하였다.

전체 시스템 구조는 마스터(Master), 슬레이브(Slave), 관리 PC로 구성된다. 관리 PC 는 인터페이스를 통해 동영상 세션 컨트롤을 본 시스템에서 설계한 컨트롤 프로토콜 형식으로 마스터에게 요청한다. 동영상 세션의 소스로는 캠코더, 동영상파일, 스트리밍 URL, 브로드캐스팅 등이 될 수 있다. 마스터는 관리 PC 의 요청에 따라 하나의 동영상 소스에 해당하는 세션을 생성하고, 동영상 소스로부터 스트림을 읽어서 필요시 역다중화 또는 파일 포맷 파싱 등을 거친 후 오디오와 비디오 스트림을 UDP로 전송할 수 있도록 패킷화 하여 멀티캐스트로 전송한다. 슬레이브는 마스터가 전송하는 스트림을 받아서 필요시 디코딩을 거친 후 이미지를 자신이 담당하는 영역으로 클리핑하거나 사이즈를 조절하여 재생한다. 마스터와 슬레이브는 데몬에 의하여 동적으로 생성되고 여러 전송 포맷 및 압축 포맷을 지원하기 위하여 Microsoft DirectShow 필터 구조로 구성된다. 슬레이브는 크게 리시버 필터, 디코더 필터(Decoder Filter), 오버레이 필터(Overlay Filter) 와 렌더러 필터(Renderer Filter) 로 구성된다. 리시버 필터는 버퍼링을 담당하는 쓰레드가 UDP 소켓 인터페이스를 통해 데이터를 읽어서 역 패킷화 한 후 버퍼링을 한다. 디코딩과 렌더링을 담당하는 쓰레드는 리시버 필터로부터 비동기적으로 미디어 데이터를 읽어간다. 각 미디어의 디코딩 및 렌더링을 담당하는 쓰레드들은 렌더러 필터에서 리퍼런스클럭(reference clock)을 기준으로 하여 미디어 데이터의 타임 스탬프까지 블러킹되는 방식으로 스케줄링 된다. 모든 미디어가 동기화되기 위해서는 각 상영기의 리퍼런스클럭의 속도가 동일해야 하는데, 오디오 스트리밍 속도는 각 상영기간의 차이가 존재하므로 리퍼런스클럭의 어긋남이 발생하게 된다. 따라서 오디오 렌더러 필터에서 제공한 클럭보다 더 정확한 RTC를 기준으로 하는 클럭을 사용하도록 설정한다. 그밖에, 디코더 필터는 압축된 데이터의 디코딩 작업을 하며, 오버레이 필터는 마스터로부터 세팅된 이미지 좌표 정보를 참조하여 디스플레이 할 이미지에 대한 클리핑, 리사이징 및 블렌딩을 한다. 관리 PC는 타일드-디스플레이 상에서 동영상의 이미지 배치 및 비율 설정과 각 PC사이에서 마스터와 슬레이브들의 지정을 위한 인터페이스를 GUI를 통해 제공한다. 또한 동영상 세션에 참가하는 모든 슬레이브들은 재생 중 QoS 통계 데이터를 생성하여 마스터에게 주기적으로 보고하며, 마스터는 보고 된 데이터들을 멀티플렉싱 하여 관리 PC로 전송하고, 관리 PC 는 이를 그래프 형태로 사용자가 모니터링 할 수 있도록 한다.

참고문헌

- [1] A. Tanenbaum and M. Steen, 2002, *Distributed Systems Principles and Paradigms*, Prentice Hall.
- [2] Microsoft MSDN, *DirectShow*, <http://msdn2.microsoft.com/en-us/library/ms783323.aspx>, 2007.