

MPEG-2 TS 스트림을 이용한 실시간 개인화면 방송 시스템 구현

남윤승*, 임동기*, 남지승*
*전남대학교 컴퓨터공학과
e-mail:yun_1@nate.com

Realtime Personal Screen Broadcasting System using MPEG-2 TS Stream : Design and Implementation

Yun-Seung Nam*, Dong-Gee Im*, Ji-Seung Nam *
*Dept of Computer Engineering, Chonnam National University

요 약

디지털 방송기술과 네트워크 통신기술의 발전은 방송 산업의 증가, 서비스의 다양화 그리고 방송 콘텐츠의 대용량화를 가져오게 되었다. 일반적으로 방송 시스템은 저장된 파일의 방송, 카메라나 캠코더와 같은 하드웨어를 통하여 인코딩된 데이터의 방송, MPEG-TS로 인코딩 및 패킷타이징된 데이터의 방송들이 대표적이다. 이러한 방송시스템에는 개인의 PC스크린을 실시간 방송하기 위한 기법이 필요할 것이며, 본 논문에서는 개인화면 실시간 방송을 위한 PC 스크린 캡처 모듈, 전송을 위한 인코딩 모듈, 다중 송수신을 위한 전송모듈 등의 시스템에 필요한 복합적인 기법을 구현하였다. 제안한 시스템은 이러한 모듈들을 통합하여 PC 스크린을 실시간으로 전송하여 시청할 수 있도록 하였다.

1. 서론

디지털 방송과 관련된 기술 및 네트워크 통신 기술의 발전으로 방송 산업 증가와 서비스의 다양화가 되고 있다. 콘텐츠들 역시 대용량화, 다양화되어지고 있다. 디지털 방송 업종은 디지털 TV방송, 인터넷 방송, 모바일방송이 대표적이며 본 논문에서 구현한 PC스크린 캡처 방송 시스템은 국내 인터넷 방송 회사에서 라이브 캡이란 서비스로 제공 중인 것으로 조사되었다.

일반적으로 방송 시스템은 저장된 파일의 방송, 카메라나 캠코더와 같은 하드웨어를 통하여 인코딩된 데이터의 방송, MPEG-TS로 이미 인코딩, 패킷타이징된 데이터의 방송들이 대표적이다.^[1] PC 스크린의 실시간 캡처 방송을 위해서는 실시간 스크린 캡처, 데이터 인코딩, 패킷 전송 등의 복합적인 기법들이 필요하다. 본 논문에서는 캡처, 인코딩, 전송 등의 모듈들을 통합하여 PC 스크린을 실시간으로 방송하고 시청할 수 있도록 구현하였다. 제안된 시스템은 윈도우 플랫폼상의 DirectX를 이용하였으며 사용자가 스크린 사이즈와 초당 프레임수를 조절하며 PC스크린을 캡처하도록 설계하였고 캡처된 영상은 MPEG-2 TS 비디오 스트림으로 압축되어지고 TCP/IP 멀티플 유니캐스트(Multiple Unicast)기법을 통하여 1:N 서버/클라이언트 개인화면 방송 시스템을 구현하였다.

본 논문의 구성은 다음과 같다. 2장에서는 구현을 위한 관련 연구를 기술하였고 3장에서는 제안된 시스템의 동작

절차 및 구현된 구조를 4장에서는 구현된 시스템의 실행 결과를 5장에서는 결론 및 향후발전으로 구성된다.

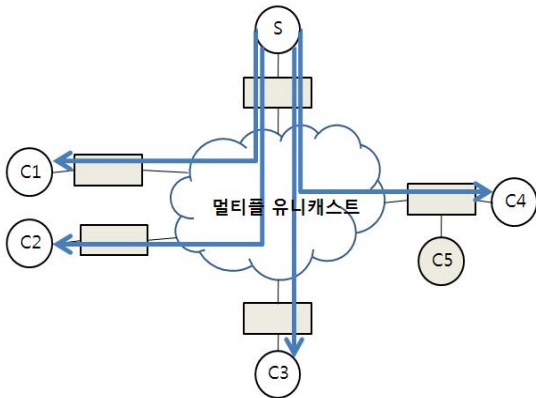
2. 관련연구

스크린 캡처란 윈도우 상에서 보이는 모니터 화면을 스크린 샷 하여 일정기간동안 비디오 형식의 파일로 저장하는 것을 말한다. 스크린 샷(Screen shot)은 컴퓨터 모니터에 보이는 그대로의 화면을 찍어서 출력한 그림을 말하며 스크린 캡처를 위해서는 화면 디스플레이의 일정 영역을 스크린 샷 하여 추출되는 RGB나 YCbCr형태의 데이터를 초당 스크린 프레임 비율로 리코딩 하게 된다. 스크린의 사이즈와 초당 프레임 비율에 따라서 스크린 캡처된 비디오의 품질이 결정되고 캡처된 파일은 동영상이라는 특성상 파일의 크기가 커지게 된다.^[2]

ActiveMovie에서 발전한 DirectShow는 Microsoft Windows에서 구동되는 스트리밍 미디어를 위한 아키텍처이며 고화질 캡처와 멀티미디어 스트리밍 재생을 할 수 있도록 제공한다. ASF, MPEG, AVI, MP3, WAV 파일과 같이 폭넓고 다양한 포맷을 지원하며 이 API들은 Windows Driver Model(WDM)장치 또는 Windows용 이전 버전의 비디오 장치들을 이용하여 캡처할 수 있도록 지원한다. DirectShow는 다른 DirectX 기반 기술과 통합되었다. 이는 비디오와 오디오의 하드웨어 가속기능이 사용가능 할 경우 이를 사용할 수 있도록 자동적으로 검출

할 수 있을 뿐 아니라 하드웨어 가속기능이 없는 경우에도 사용이 가능하다. 다양한 소스와 포맷 그리고 하드웨어 장치를 지원하기 위해 DirectShow는 컴포넌트로 모듈화된 필터를 사용하며^[3] 제안하는 시스템에서는 원본 데이터의 추출과 송/수신 및 파싱을 위하여 모듈화된 필터들을 설계 및 구현하였다.

유니캐스트(Unicast)기법은 네트워크에서 단일 송신자와 단일 수신자간의 일대일 데이터 전송 방식을 말한다. 멀티플 유니캐스트(Multiple Unicast)란 한명의 송신자와 여러 수신자간의 연결 상태에서 송신자는 연결된 수신자에게 유니캐스트 기법으로 데이터를 전송하는 방식이다. 요즘 대부분의 라우터가 유니캐스트 방식을 지원하기 때문에 가장 많이 사용되고 있다.



(그림 1) 멀티플 유니캐스트

그림 1은 멀티플 유니캐스트 전송 모습을 표현한 그림이다. 서버 S는 클라이언트 C1, C2, C3, C4에 독립된 각각의 데이터를 전송하고 있다.

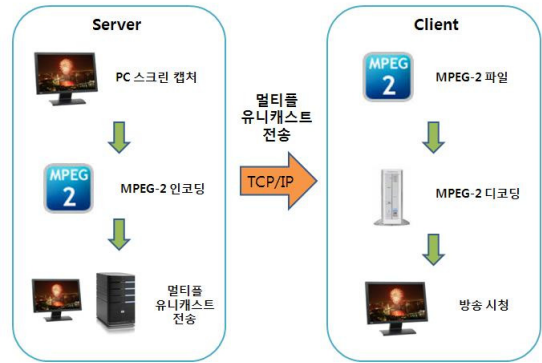
MPEG-2 비디오 인코딩은 오디오 신호를 비롯한 동영상의 일반적인 인코딩을 위한 규약이다. 부호화된 비디오 스트림은 인트라(Intra), 전방 예측(Predictive), 양방향 예측(Bidirectional)의 I프레임, B프레임, P프레임 세 가지 프레임들이 GOP(Group of Picture)구조라는 특별한 구조를 이루어 만들어진다. 일반적으로 원본 데이터는 720*480 픽셀에 초당 29.97 프레임으로 인코딩된다. MPEG-2 TS는 MPEG-2 시스템의 일부로 전송 오류를 유연하게 대처할 수 있는 다중화 방식 및 동기화 관련 정보 등을 제공함으로써 방송 환경이나 네트워크 전송에 적합한 방식으로 전송오류를 대비하여 짧고 고정된 길이의 패킷으로 구성된다.^[4]

3. 방송 시스템 모델

1. 개요

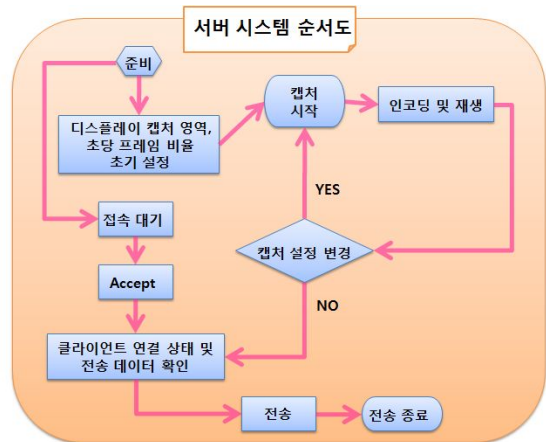
제안된 시스템에서 방송 서버의 설계 구조는 다음과 같다. DirectShow를 사용 스크린 화면의 영역과 초당 프레임수를 설정하여 실시간 캡처하게 된다.^[3,5,6] 캡처된 영상은 MPEG-2 TS로 인코딩하고 이 패킷은 서버에 연결된 클라이언트들에게 TCP/IP를 통하여 전송하게 된다. 각 클

라이언트들은 전송된 MPEG-2 TS 데이터를 받아 저장하거나 디코딩하여 재생하게 된다. HDTV의 전송표준으로 채택된 MPEG-2 TS는 압축된 오디오나 비디오 스트림뿐만 아니라 필요에 따라 사용자 데이터를 다중화 하여 전송에 적합하도록 포장된 비트 스트림이다. 전송을 위한 스트림이기 때문에 전송 시 발생할 수 있는 오류에 대비하여 188바이트의 짧고 고정된 길이의 패킷으로 구성된다.^[4]



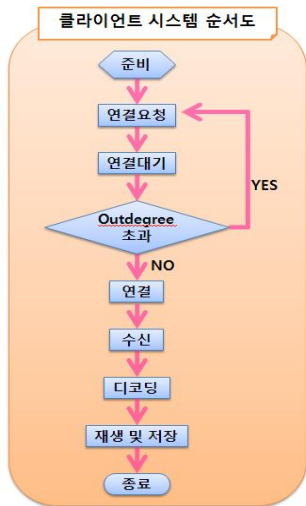
(그림 2) 시스템 구조도

그림 2는 앞에서 설명한 방송 서버와 클라이언트의 전체적인 시스템 모습을 표현한 그림이다. 방송 서버는 캡처, 인코딩, 전송 및 재생을 하게 되고 각각의 클라이언트들은 전송된 데이터를 받아 저장하거나 디코딩하여 재생하게 된다.



(그림 3) 서버 시스템 순서도

그림 3은 서버와 시스템 각각의 실행 순서도를 표현한 그림이다. 그림을 보면 서버는 프로그램 실행 후 준비과정에서 사용자가 초기 디스플레이의 캡처 영역과 초당 프레임 비율을 설정하게 된다. 캡처 시작을 하게 되면 초기 설정 값으로 캡처가 시작되고 인코딩 후 재생된다. 만약 재생도중 캡처 설정을 변경하게 되면 변경된 값으로 캡처가 다시 시작된다. 시스템은 전송대기 상태에서 클라이언트의 접속을 기다리게 되고 클라이언트들이 접속하면 시스템은 Count를 증가시키며 클라이언트의 연결 상태를 확인하게 된다. 이 후 인코딩된 MPEG-2 TS 패킷 파일은 각각의 클라이언트에게 전송한다. 만약 전송 도중 사용자가 캡처 설정을 변경하게 되면 변경된 캡처 영상이 전송된다.



(그림 4) 클라이언트 시스템 순서도

그림 4는 클라이언트 시스템의 전체적인 순서도를 표현한 그림이다. 클라이언트는 서버와의 연결을 준비하게 된다. 이후 연결 요청을 하게 되면 서버는 Count 상태를 확인하여 Outdegree를 초과되지 않을 경우 연결이 허락된다. 만약 초과되어 서버와의 연결이 허락되지 않을 경우엔 연결요청 상태로 돌아가게 된다. 연결이 허락되어 서버에 연결되면 클라이언트는 서버로부터 데이터를 전송받게 되고 전송 받은 MPEG-2 데이터는 파일로 저장되거나 디코딩하여 실시간 재생된다.

2. Windows 플랫폼 상에서 시스템 구현

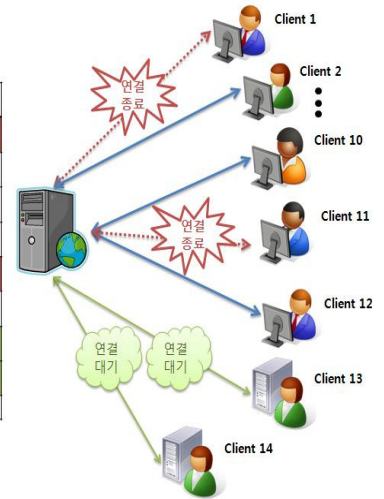
제안된 모델에서 스크린 캡처는 DirectShow를 이용하는데 방송자가 화면 크기와 영역, 초당 프레임수를 설정 및 변경이 가능토록 하기 위하여 소스 필터의 Interface 헤더를 외부 노출하였다.^[5,6] 추출되어진 스크린 캡처 원본 파일은 크기가 너무 크기 때문에 전송에 알맞은 MPEG-2 TS로 인코딩하게 되며 제한된 시스템에서 사용된 MPEG-2 TS 인코더 필터는 인터넷상에서 오픈소스로 제공된 필터를 사용하였다.

제안된 시스템은 하나의 서버가 다수의 클라이언트들에게 실시간 전송하는 멀티플 유니캐스트를 사용하므로 서버 네트워크 연결은 클라이언트 접속 시 쓰레드를 사용 클라이언트의 동시 접속이 가능토록 하였고 방송의 질을 높이기 위하여 OutDegree를 설정 클라이언트 접속 수를 제한 가능토록 하였다.

그림 5는 서버 시스템의 클라이언트 관리 모습을 표현한 그림이다. 서버는 쓰레드를 사용하여 멀티플 유니캐스트가 가능토록 구현하였고 Outdegree를 초과하지 않을 경우 클라이언트의 연결을 허락하고 Count를 추가하게 된다. 만약 연결 되었던 클라이언트가 연결을 종료하게 되면 Count는 감소하게 된다. 클라이언트로부터 연결 요청이 들어오면 서버 시스템은 Count값과 Outdegree값을 비교하여 설정된 Outdegree보다 클 경우 연결이 허락지 않고 클라이언트는 연결 대기하며 서버의 연결허락을 기다리게 된다.

서버 시스템 클라이언트 관리

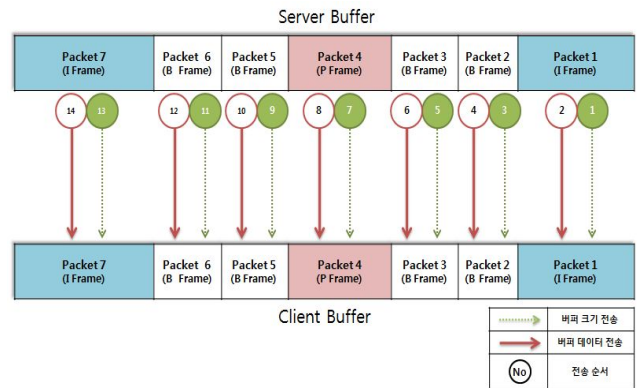
No	State	Count	ADDR
1	OUT		121.121.121.121
2	JOIN	1	192.168.17.22.
3	JOIN	2	192.17.12.23
...
11	OUT		155.131.203.65
12	JOIN	10	168.121.10.36
13	WAIT	11	210.133.67.17
14	WAIT	12	121.169.49.128
Outdegree : 10			



(그림 5) 서버 시스템의 클라이언트 연결 관리

MPEG-2 데이터 패킷은 쓰레드를 통해 연결된 모든 클라이언트에게 전송된다. 서버에서 데이터 전송 시 클라이언트에게 MPEG-2 포맷임을 알려주어야 하기 때문에 각각의 시스템과 패킷들은 데이터 설정 및 시퀀스 헤더 층을 초기화 하여야 한다. 또한 MPEG-2 영상의 프레임은 I 프레임 1개와 n개의 B프레임 m개의 P프레임과 같은 형태의 구조로 이루어져 있기 때문에^[4] 서버는 버퍼에 있는 패킷 전송 전에 전송할 패킷의 버퍼 사이드를 얻어서 클라이언트에게 보내준 다음 버퍼에 있는 패킷을 전송하여야 패킷의 손실 없이 송수신 가능하다. 클라이언트는 서버가 전송하려는 패킷의 버퍼 사이드를 받아서 전송받을 패킷 크기만큼의 버퍼 공간을 마련해가며 데이터를 받게 된다. 이후 클라이언트는 버퍼에 받은 패킷을 파일로 저장하거나 디코딩하여 재생하게 된다. 이러한 과정은 데이터 전송을 시작하여 전송 종료될 때까지 서버에 접속된 모든 클라이언트들에게 같은 방식으로 처리된다.

그림 6은 제안한 시스템의 서버와 클라이언트 데이터 전송과정을 버퍼상태와 함께 보여주는 그림이다. 그림을 보면 서버의 버퍼에는 패킷들이 채워진다. 패킷은 I프레임과 B프레임 P프레임으로 구성되어 있고 프레임들은 각각

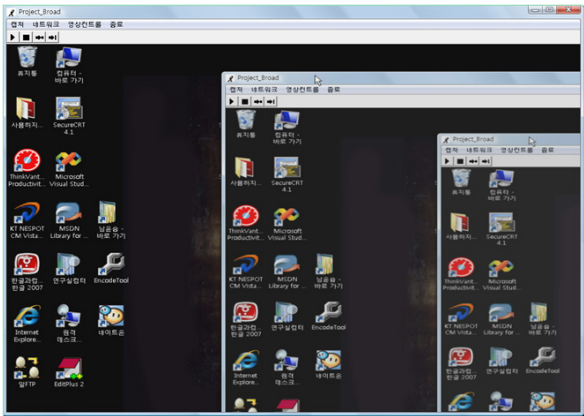


(그림 6) 서버/클라이언트 데이터 전송 과정

사이즈가 다르므로 각각의 패킷과 버퍼도 사이즈가 일정하지 않게 된다. 버퍼의 크기는 캡처되는 영상 이미지에 따라서 변하게 되고 프레임의 구성 순서는 인코더마다 다른 것으로 조사되었다. 전송 과정은 다음과 같다. 우선 서버 버퍼의 가장 오른쪽에 있는 Packet1이 전송되게 되는데 서버는 Packet1의 데이터를 전송하기 전에 Packet1의 버퍼 사이즈를 알아내고 이를 클라이언트에게 전송하게 된다. 클라이언트는 전송받은 사이즈만큼의 버퍼공간을 마련하게 되고 이후 서버는 Packet1의 실제 데이터를 전송하게 된다. 클라이언트는 마련된 버퍼에 Packet1을 받게 되고 Packet1의 전송이 끝나면 서버는 다시 Packet2의 버퍼 사이즈를 체크하고 클라이언트에게 보내는 방법으로 데이터의 송/수신이 이뤄진다.

4. 실행결과

수신품질의 척도에는 일반적으로 손실, 지연 그리고 지터 등이 있다. 제안 시스템은 TCP/IP의 신뢰적인 전송을 통하여 최소한의 손실을 보장하였다. 하지만 스크린 캡처에서 서버 컴퓨터에 많은 과부하와 전송 시 지연이 발생하는 것을 확인하였고 클라이언트가 추가될 때마다 전송 품질이 떨어지고 딜레이가 발생하는 것으로 확인되었다. 테스트는 LAN에서 하였으며 초기 1대1 방송을 테스트 하였고 다음에는 Outdegree를 5로 하여 클라이언트 5개를 차례로 연결하며 테스트 하였다. 처음 실행에서 서버와 클라이언트 컴퓨터의 성능은 CPU 코어2 듀오 2GHz , RAM 2GB이고 두 번째 테스트에서는 서버 CPU 코어2 듀오 2GHz, RAM 2GB 5대의 클라이언트의 성능 펜티엄 4, RAM 512MB~1GB 에서 하였다. 캡처 품질은 720*480 크기에 rate는 30 Frame/Sec와 1024*768 크기에 rate는 30Frame/Sec로 각각 테스트 하였다.



(그림 3) 방송자 실행 모습

그림3 은 방송자 어플리케이션 실행 화면이다. 현재 스크린의 일부 영역을 720*480 사이즈에 초당 30 프레임 비율로 실시간 캡처하여 인코딩 후 재생한 모습이다.



(그림 4) 클라이언트 실행 화면

그림 4는 클라이언트에 실행 화면이다. 서버로부터 전송받는 영상의 사이즈는 1024*768 이고 초당 30프레임 비율의 캡처 영상을 전송받아 재생하고 있는 모습이다.

5. 결론 및 향후 연구

본 논문에서는 MPEG-2 TS 스트림을 이용한 실시간 개인화면 방송 시스템을 설계 및 구현하였다. 테스트는 한 LAN에서 하였으며 최대 접속은 5명까지 하였는데 접속자가 증가할 때마다 전송 지연이 발생하는 것으로 확인 되었다. 이는 서버에서 스크린 캡처 및 전송 시 CPU에 많은 부하가 걸리는 것으로 보이며 또한 클라이언트들의 성능에도 의존하는 것으로 보인다. 패킷 전송 시 좋은 품질을 얻기 위해서는 오버레이 기법 등을 사용하여 CPU의 지연을 낮추고 버퍼 협상과 패킷 전송 기법에 보다 많은 연구가 필요할 것으로 보이며 서버와 클라이언트 관리에도 보다 효율적인 알고리즘연구가 필요할 것으로 보인다.

참 고 문 헌

- [1] J Lee and K. Chon, "High-Definition Digital Video Over IP: System Design, Experiments and Applications," in Proc. APAN Network Research Workshop 2004, Cairns, Australia, July. 2004.
- [2] Screen shot, Video capture mechanism <http://www.wikipedia.org>
- [3] DirectX Standard, http://www.microsoft.com/win_dows/directx/
- [4] ISO 13818-1,2:2000, "Information technology: Generic coding of moving picture and associated audio information: Systems," 2000.
- [5] 신화선, "Directshow 멀티미디어 프로그래밍," 한빛미디어, 2004.
- [6] Kris Gray, "The Microsoft DirectX 9 Programmable Graphics Pipeline," Corporation Microsoft, July. 2003.
- [7] 손승철, 박용완, 이형욱, 허권, 남지승 "오버레이 멀티캐스트를 위한 HDTV 중계전송 시스템 설계 및 구현"