

멀티미디어 연출 QoS 향상을 위한 적응적 스트림 처리기 설계

최원희⁰ 김정훈 박대원 오미경 임영환
송실대학교 미디어 연구실
(wony12, jhkim, laputa, yujinn)@media.soongsil.ac.kr
yhlhm@computing.soongsil.ac.kr

The Design of Adaptive Stream Engine for Multimedia Presentation with Guaranteed QoS

Wonhui Choe⁰ Jeounghun Kim Daewon Park Mikyoung Oh
Younghwan Lim
Dept. of Computing, Soongsil University

요 약

Post PC 시대로 접어들면서 이동 단말기를 비롯한 임베디드 기기에서 멀티미디어 스트림 서비스가 요구되고 있지만 데스크탑 PC에 비해서 컴퓨팅 파워가 부족하기 때문에 스트림 처리가 쉽지 않다. 이동 단말기를 비롯한 임베디드 기기의 특징은 기종마다 멀티미디어 연출 QoS가 다르기 때문에 멀티미디어 스트림 처리가 어렵다. 본 논문에서 적응적 스트림 처리기를 제안한다. 적응적 스트림 처리기는 적응적 버퍼 관리 기법과 필터 파이프 라인으로 구성되어 있다. 적응적 버퍼 관리 기법으로 데이터의 양을 조절하고 필터 파이프 라인으로 데이터를 원하는 형태로 변형시켜서 다양한 멀티미디어 연출 QoS를 만족시킬 수 있다.

1. 서 론

현재 윈도우 기반 PC상에서 동작하는 멀티미디어 스트림 처리기 개발은 리얼플레이어나 윈도우 미디어 플레이어 처럼 상용화된 제품들이 많다. 하지만 무선 인터넷 망에서 멀티미디어 연출 QoS 향상을 위한 적응적 스트림 처리기의 개발과 멀티미디어와 관련된 서비스들의 구현이 늦은 편이다. 최근 컴퓨팅 환경의 변화는 윈도우 기반의 데스크탑 PC에서 이제는 이동환경, 무선 환경, 임베디드 시스템 등의 Post PC 시대로 접어들고 있다.

Post PC시대의 컴퓨팅 환경에서 이동 단말기에 탑재하여 동작하는 멀티미디어 스트리밍 처리기의 개발을 통해 인터넷 접속기능을 갖춘 제품인 인터넷 정보기기(웹TV, 웹폰 등)나 이동 단말기(인터넷 접속 PDA, Notebook, IMT2000 단말기 등) 그리고 다른 특정 목적의 임베디드 시스템에서 사용자가 원하는 기능들만을 가지는 많은 응용 프로그램에서 멀티미디어 서비스가 사용 가능할 것으로 예상된다.

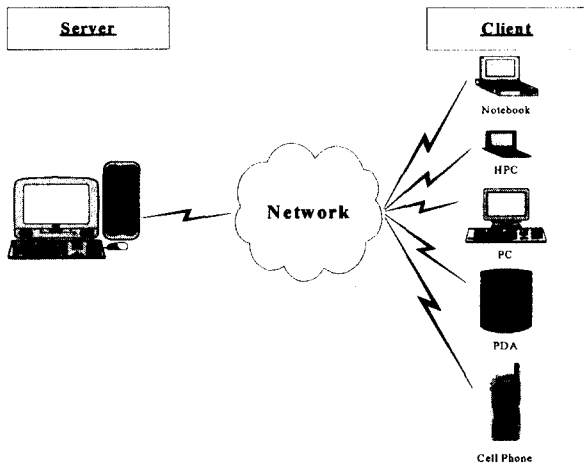
스트림 처리기가 이처럼 컴퓨팅 파워가 떨어지는 이들 시스템에서 멀티미디어 연출 QoS를 만족시키면서 동작하기 위해서는 멀티미디어 데이터 전송 시 효율적인 메모리 관리와 스트림 제어를 위해 적응적 버퍼관리 기법이 필요하다. 그리고 다양한 형태의 데이터 변환 필터로 구성되는 필터파이프 구성하여 실시간 데이터 전송이 가능한 최소한의 기능을 갖는 적응적 멀티미디어 스트림 처리기의 개발을 필요로 한다.

2. 문제 제기 해결 방안

이동환경에서 동영상상을 포함한 멀티미디어 연출 QoS를 만족시키며 다양한 단말기에서 재생하는 기술은 다음과 같은 문제점으로 인해 실용화되어 있지 못하다.

문제는 “멀티미디어 데이터를 어떻게 다양한 단말기에 맞추어 실시간 전송 및 재생하겠는가?”이다. 이를 극복하기 위한 기본 아이디어는 송신측과 수신측의 QoS 요소를 파악하여 비교한 후 이에 적합한 다양한 형태의 변환 필터들로 구성되는 스트림을 구성할 수 있는 적응적 스트림 처리기가 필요하다는 것이다. 이 문제를 해결하기 위하여 본 논문에서는 적응적 스트림 처리기를 제안하며 이동단말기의 QoS 정보에 맞추어 자동적으로 실시간 데이터를 변환하여 전송하는 방법을 제안한다.

다음의 그림과 같이 이동 단말기의 환경은 각 기종마다 다르고 그 QoS 정보 또한 다양하다. 이 문제를 해결하기 위해서 기존의 멀티미디어 스트림 처리기에 멀티미디어 데이터의 연출 QoS 관리를 위해 적응적 버퍼 관리 기법과 QoS Manager를 추가하고, 멀티미디어 데이터의 변환 기능을 필터 형태로 포함시켜 데이터를 전송하는 측과 데이터를 사용하는 측의 환경이 상이하더라도 수신측에 환경에 따라 데이터를 실시간 변환하여 멀티미디어 연출을 원활히 수행할 수 있도록 한다는 것이다.



(그림2.1) 다양한 형태의 단말기

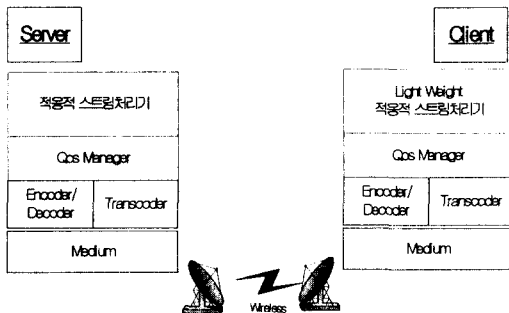
3. 적응적 스트림 처리기

적응적 스트림 처리기는 어떤 환경에서도 멀티미디어 스트리밍 서비스를 제공받을 수 있도록 멀티미디어 스트림을 연결하고, 컴퓨팅 능력이 떨어지는 저속 이동망에서 원활한 서비스를 위해 단말기에 따라 데이터를 변환하여 전송하도록 하는 멀티미디어 스트림 처리기이다.

이러한 적응적 스트림 처리기는 PC 기반의 운영체제에 탑재될 적응적 스트림 처리기와 이동 단말기의 Embedded 운영체제 기반에 탑재될 Light Weight 적응적 스트림 처리기를 생각할 수 있으며, 각각에 QoS 정보를 관리할 수 있는 QoS 관리자와 멀티미디어 코덱, 멀티미디어 데이터 변환기를 가지고 있다.

두 스트림 처리기는 기능면에서 비교해볼 때 이동 단말기의 Embedded 운영체제 기반에 탑재될 스트림 처리기는 컴퓨팅 파워가 떨어지는 이동단말기에서 사용할 수 있어야 함으로 최소한의 기능을 갖는 Light Weight 스트림 처리기를 말한다.

다음 그림은 서버(PC 기반의 운영체제에 탑재될 스트림 처리기)와 다양한 클라이언트(이동 단말기의 Embedded 운영체제 기반에 탑재될 스트림 처리기)가 네트워크상에 존재한다고 할 때 동영상을 포함한 다양한 멀티미디어 데이터 전송을 위한 시스템 구조도이다.



(그림3.1) 시스템 구조도

적응적 스트림 처리기의 가장 큰 특징은 크게 두 가지가 있는데 하나는 적응적 버퍼관리 기법을 통해 멀티미디어 연출 QoS를 관리하는 기능을 포함하고 있다는 것과 또 다른 특징은 다양한 형태의 데이터 변환이 가능하게 하는 데이터 변환 필터로 구성되는 필터 파이프라인이다.

4. 적응적 버퍼 관리 기법

Windows 기반의 PC에서 멀티미디어 데이터를 캡처하여 압축한 후 무선 인터넷 이동망을 통해 전송하고, 이동 단말기에서 이를 수신하여 복원한 후 재생하는 경우를 살펴보자. 이 경우 구성되는 송신측 스트림은 카메라미디움, H.263인코더필터, Transcoder 필터, WirelessWrite미디움으로 구성되며, 수신측은 WirelessRead미디움, 디코더필터, 모니터미디움으로 구성된다.

여기서 각각의 미디움 객체와 필터 객체는 하나의 버퍼를 가지고 동작하지만, 송신측에서는 실제 데이터 캡처 및 전송 시 Capture 속도와 인코딩 속도의 차이로 인해 인코더 객체가 가지고 있는 버퍼는 계속 증가하여 메모리 사용량이 증가하는 문제가 발생된다. 또 수신측에서도 이렇듯 스트림을 형성하고 있는 각 객체간의 처리속도의 차이로 인해 버퍼의 크기가 증가하는 문제가 발생할 수 있다.

이에 대한 해결 방법으로 본 논문에서는 적응적 버퍼 관리 기법을 제안한다.

적응적 버퍼 관리 기법이란 버퍼에 데이터가 입력될 때 마다 버퍼 크기를 체크하여 일정 수준의 임계치를 설정하여 임계치 이상이나 임계치 이하가 되면 이벤트를 발생하여 데이터의 전송이 조절될 수 있도록 한다. 즉 Capture속도와 인코딩 속도의 차이로 인해 발생하는 문제는 인코더 객체의 버퍼를 체크하여 일정 수준의 임계치에 도달하면 이벤트를 발생하여 Input Medium 객체로 하여금 캡처 속도를 떨어뜨리고 반대의 경우엔 캡처 속도를 증가시킴으로써 적응적으로 대처할 수 있도록 한다.

이러한 적응적 버퍼 관리 기법을 통해 전송되는 데이터 양을 조절함으로써 멀티미디어 연출 QoS를 관리하게 한다.

적응적 스트림 처리기는 일반적인 스트림 처리기법에 이러한 버퍼 관리 기법과 함께 다양한 환경에서 사용자 QoS를 만족시킬 수 있도록 QoS 관리가능 기술을 추가한 것을 말한다.

QoS 관리자에 의해 추출되는 QoS 정보로는 File format, Resolution, Frame rate, Color 정보, Buffer size등이 있다. QoS 관리자에 의해서 추출된 전송측과 수신측의 QoS 정보를 사용하여 단말기의 환경에 맞도록 데이터를 실시간 변환하여 전송함으로써 기존의 다양한 멀티미디어 콘텐츠를 이동단말기에서 사용할 수 있다.

5. 필터 파이프 라인 구성 및 필터 스케줄링

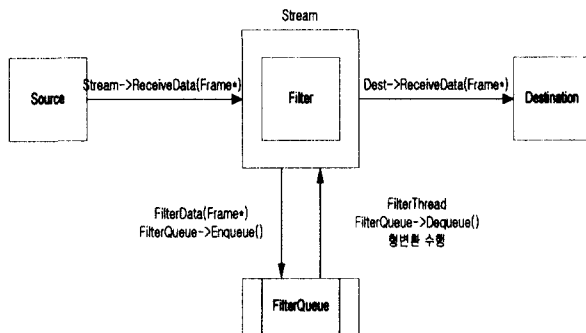
적응적 스트림 처리기의 가장 큰 특징 중 다른 하나는 다양한 형태의 데이터 변환이 가능하게 하는 데

이더 변환 필터로 구성되는 필터 파이프라인이다. 데이터 변환 필터는 입력된 데이터에 대해 사용자가 원하는 또 다른 데이터 형으로 바꾸는 필터이다. 이 필터는 입력된 데이터에 대해 데이터의 변형을 수행하는 필터이다.

데이터 변환 필터로는 다음과 같은 종류가 있다.
 Encoding Filter : SWH263Encoder, MPEGEncoder 등
 Decoding Filter : SWH263Decoder, MPEGDecoder 등
 Transcoding Filter :
 FormatTranscoder, ResolutionTranscoder 등

5.1 필터의 동작 모델

필터의 동작 과정은 아래 그림과 같다.



(그림 5.1) 필터의 동작 모델

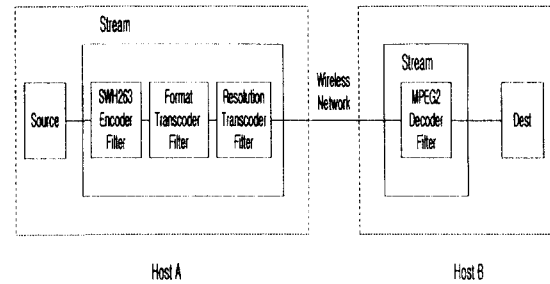
먼저 Source 객체는 Stream객체의 ReceiveData() 함수를 호출하여 입력 미디어로부터 전달 받은 멀티미디어 데이터를 Stream 객체로 전달한다.

Stream 객체는 Source객체로부터 전달 받은 데이터를 필터파이프가 있으면 필터파이프의 첫 번째 필터에 FilterData() 함수를 호출하여 그 필터가 가지고 있는 버퍼인 큐에 데이터를 저장한 후 이벤트를 발생한다. 필터 파이프가 구성되어 있지 않으면 Destination객체의 ReceiveData() 함수를 호출하여 데이터를 Destination 객체로 전송한다.

필터 파이프를 구성하고 있는 필터객체는 이벤트가 발생하면 큐로부터 데이터를 가져와 필터 고유의 데이터 변환기능을 수행한 후 변환된 데이터를 다음 필터가 있으면 다음 필터에 전달하고 그렇지 않으면 Destination 객체의 ReceiveData() 함수를 호출하여 데이터를 Destination 객체로 전송한다.

마지막으로 Destination 객체는 전달 받은 데이터를 출력 미디어로 전달하여 재생한다.

다음 그림에서 Host A는 Windows 기반의 PC이고 Host B는 PDA라고 가정하자. 전송측인 Host A에서 352*288(CIF)크기의 비디오 데이터를 카메라로 캡처하여 H.263으로 인코딩한 후 PDA인 Host B에서 연출 QoS를 만족시키기 위해 H.263에서 Mpeg2로 파일 포맷을 변환시키고, 변환된 데이터에 대해 해상도를 176*144(QCIF)로 변환한 후 전송한다. 이렇게 변환함으로써 저속의 PDA로 전달되는 데이터의 크기도 줄어들게 된다. 수신측인 Host B에서는 전송 받은 Mpeg2 포맷의 비디오 데이터를 Decoding한 후 재생하는 그림이다.



(그림 5.2) 필터 파이프 구성

6. 결론

본 논문에서 다양한 환경에서 멀티미디어 연출 QoS를 향상시키기 위해 적응적 스트림 처리기를 제안했다. 적응적 스트림 처리기의 적응적 버퍼 관리 기법과 필터 파이프 라인을 이용해서 컴퓨팅 파워가 부족한 환경에서도 멀티미디어 연출 QoS를 보장할 수 있다. 현재 윈도우 기반의 적응적 스트림 처리기가 구현이 되어있고 향후 멀티미디어 연출 QoS 보장을 위해 적응적 스트림 처리기를 더욱 발전 시킬 예정이다.

7. Reference

[1] A. Dan and D. Sitarma, "Buffer Management Policy for an On-Demand Video Server," IBM Research Report, RC 19347, Yorktown Heights, NY, 1993.
 [2] B. Ozden and R. S. Yu, "Consumption-Based Buffer Management for Maximizing System Throughput of a Continuous Media Data," in Proc. of IEEE International Conference on Multimedia Computing and Systems, Jun 1996.
 [3] David P. Anderson and George Homsy "A Continuous Media I/O Server and Its Synchronization Mechanism," IEEE Computer, Special Issue on Multimedia Information System, October 1991, pp. 51 ~ 57.
 [4] Y. S. Ryu and K. Koh, "A Dynamic Buffer Management Technique for a Video-on-Demand Server," in Proc of IPSJ International Symposium on Multimedia Systems, Feb. 1996, pp. 216 ~223.
 [5] M. Salmony and D. Shepherd, Extending OSI to Support Synchronization Required by Multimedia Applications, Computer Communications, 13:399-406, September 1990.
 [6] Younghwan Lim, Doohyun Kim, Snagwan Kung, "An Integrated Synchronization Method for a Hyperpresentation in a Distributed Computing Environment," The Journal of KIPS, No. 5, Vol.6, 1998.6
 [7] Younghwan Lim, Myungsu Lim, SunHye Lee, and Seeyun Woo, "An Iconic Programming Tool for the Hyperpresentation," The Proceedings of KIPS, No 2, Vol 4, 1997