# IPv6 멀티캐스트를 이용한 고품질 화상회의 응용 개발*

이병욱**, 조호식**, 유태완**, 최양희**, 인민교***, 이승윤***,
**서울대학교 컴퓨터공학부
***한국전자통신연구원

# Development of High quality Video Conferencing Tool by using IPv6 multicast*

Hosik Cho**, Taewan You**, Byoungwook Lee**, Yanghee Choi**, Minkyo In***, Seungyun Lee***, Hyoungjun Kim***
**School of Computer Science and Engineering, Seoul National University
***Electronics and Telecommunications Research Institute

요 약

IPv6 기술은 IPv4의 문제점인 주소고갈, 보안문제 등을 해결하기 위해 제시된 차세대 인터넷 기술이다. IPv6 기술은 현재 프로토콜의 정의가 완료된 상황이지만 사업자들의 이해관계로 인해 아직까지 널리 보급되지 못하고 6Bone 과 같은 시험 망에서만 사용되고 있는 실정이다. 그러나 IPv6는 가까운 미래에 IPv4를 대체할 것으로 보이며 기존에 있던 IPv4를 기반으로 하는 응용 프로그램들도 IPv6 용으로 새로이 바뀌게 될 것이다. 이러한 배경 하에 본 연구팀은 IPv6 multicast를 기반으로 하는 화상회의 응용프로그램을 개발하였으며 본 논문은 그 개발에 대한 내용을 담고 있다. 본 논문에서는 화상회의 응용프로그램의 전체적인 구조와 구현 시 사용된 기술과 화상회의 플랫폼기술에 대해서 살펴보고 응용프로그램을 바탕으로 측정한 사용 bandwidth 와 delay를 분석해 보도록 하겠다.

## 1. 서론

향후 IPv6 기술은 IPv4를 대체하게 될 것이며 이에 따라 학계에서는 IPv6 기술에 대한 연구를 꾸준히 진행시켜 왔다. 최근 들어 많은 연구 기관들이 IPv4용 응용들을 IPv6망에서도 동작할 수 있도록 이식함으로서 IPv6 멀티캐스트에 기반 응용연구가 활발해질 것으로 예상된다.

이상을 배경으로 본 연구팀은 IPv6 멀티캐스트 응용기술을 개발하는데 있어 경험적인 지식을 쌓고 IPv6 기술이 널리 보급되는데 기여하고자 기존 MBONE 화상회의 프로그램보다 고품질 데이터를 안정적으로 제공하는 응용을 구현하게 되었다. 이후 본 논문에서는 2장에서 고품질 화상회의 응용에 대해 전체적인 개요와 구조를 살펴보고 제 3장에서는 본 응용 구현에 사용된 기술에 대해 살펴보도록 한다. 이어 제 4장에서는 본 연구팀에서 개발한 화상회의 응용프로그램으로 측정한 실험 데이터를 분석한다. 마지막으로 결론에서는 본 과제의 결과와 의미, 향후 적용방향을 제안함으로써 논문을 마친다.

## 2. 고품질 화상 회의 응용의 개요 및 구조

본 논문에서 구현한 화상회의 시스템은 GUI(Graphic User Interface)를 기반으로 한 윈도우용 응용 프로그램으로 IPv6 망 환경에서 동작하도록 만들어졌다. Microsoft Visual C++과 DirectX SDK를 이용하였으며 USB인터페이스의 화상 카메라와 DirectShow를 지원하는 사운드 카드 그리고 마이크가 필요하다 [3]

구현된 응용의 개요는 그림 1과 같다.

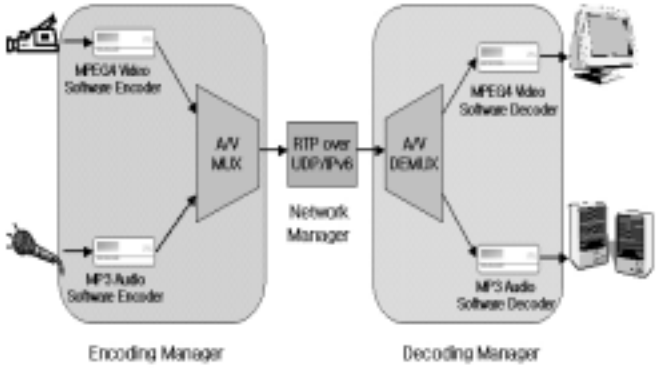


그림 1 고품질 화상회의 응용의 개요

### 2.1 개요

화상 카메라로부터 캡쳐된 영상은 MPEG4, MIPEG, MS Video1 등 비디오 코덱에 의해서 압축되며 마이크로부터 들어온 음성은 MP3 코덱에 의해서 압축되어 A/V MUX(Audio/Video Multiplexer)로 전달된다. A/V MUX에서는 비디오 스트림과 오디오 스트림을 하나의 스트림으로 만들어 네트워크 관리부로 보낸다.

네트워크 관리부에서는 A/V MUX로부터 전달된 데이터에 RTP 헤더를 덧붙여 IPv6 네트워크를 통해 전송하는 역할과 네트워크를 통해 들어온 RTP 패킷에서 데이터만 떼어내서 디코딩 관리부로 넘겨주는 역할을 한다. 또한 RTCP를 통해 사용자 정보와 사용하고 있는 대역폭, 오류에 대한 정보를 주고 받는다. [4]

A/V DEMUX(Audio/Video Demultiplexer)에서는 네트워크 관리부로부터 올라온 스트림을 비디오 스트림과 오디오 스트림으로 분리하여 각각 비디오 디코더와 MP3 디코더로

* 본 논문은 2004년도 한국전자통신연구원(ETRI)과 두뇌한국21, 국가지정연구실 프로젝트 지원을 받아 수행되었음

보낸다. 디코딩된 영상은 Video Renderer를 통해 화면에 보여지고 음성은 사운드 카드를 통해 스피커로 재생된다.

### 2.2 구조

구현된 화상회의 응용은 그림 2와 같은 구조를 가진다. RTP 데이터의 송신과 수신에 사용되는 소켓이 별개의 스레드로 동작하며 DirectShow의 필터로 구현된 캡쳐와 재생 모듈간에 공유버퍼를 사용한다. 전체적인 제어는 UI에서 담당하게 된다. 고품질 화상회의 응용에서 송신부는 화상과 음성을 캡쳐하여 네트워크로 전송한다. 화상의 경우 160x120~640x480 픽셀의 해상도를 가지고 5fps~30fps(frame/sec)로 캡쳐되며 음성의 경우 56Kbps로 인코딩 된다

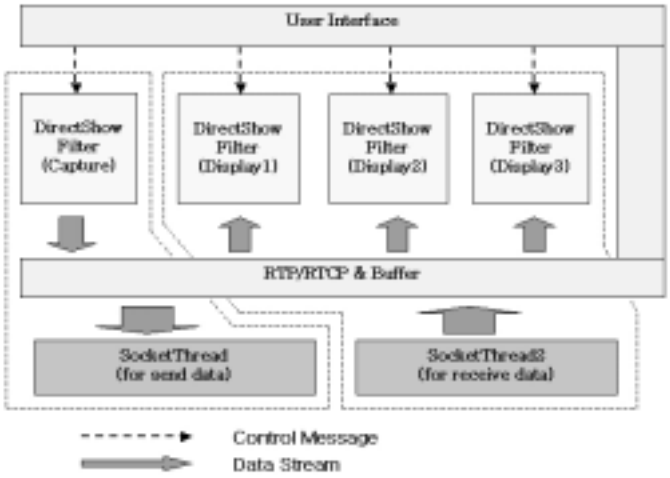


그림 2 고품질 화상회의 응용의 구조

Video와 Audio Tee에서 분리된 데이터 스트림에 RTP헤더가 덧붙여지고 IPv6 소켓에 의해 네트웍으로 전송된다. 네트워크를 통해 전송된 RTP 패킷들은 응용 계층에서 각 소스별로 구분되고 하나의 소스마다 필터 그래프가 생성되어 화상과 음성을 재생하게 된다

## 3. 구현 시 사용된 기술

이 절에서는 본 화상회의 응용의 개발에 사용된 기술들에 대하여 간단하게 살펴보도록 한다. 화상의 캡쳐와 재생을 위해 채택한 DirectShow 기술과 DirectShow filter로부터 캡쳐된 데이터를 RTP/RTCP쪽으로 전달하기 위한 버퍼로 사용된 named shared memory기술, 버퍼에 데이터가 들어오는 속도에 따라 가변적으로 프레임의 재생속도를 조절하는 버퍼관리 기법이 등이 사용되었다.

### 3.1 Named shared memory buffer

프로세스 사이에 데이터를 주고 받는 Inter Process Communication (IPC) 에는 크게 세 가지 방법이 주로 쓰인다. 하나는 프로세스 사이에 Pipe를 두고 두 프로세스를 연결하는 방법 Pipe 기법. 두 번째는 운영체제에서 제공하는 공유 메모리를 사용하는 shared memory 기법. 마지막으로 소켓을 사용하여 두 프로세스가 하나의 네트워크 노드처럼 동작하도록 하는 socket 기법을 들 수 있다. 각각의 기법은 장단점이 존재하지만 본 응용에서는 많은 양의 데이터를 빠른 속도로 처리해줄 필요가 있기 때문에 shared memory 기법을 사용하였으며 Windows API에서는 shared memory를 파일처럼 손쉽게 사용할 수 있도록 named shared memory 라는 접근 방식을 제공하고 있다.

### 3.2 버퍼 관리 기법

버퍼의 크기와 화상/음성의 안정성은 tradeoff 관계에 있다. 버퍼의 크기를 크게 하면 네트워크의 delay 필터에서의 pull 타이밍이 약간 어긋나더라도 버퍼링 되어있는 데이터가 많이 있기 때문에 안정적인 화상과 음성이 재생되게 된다. 그러나 버퍼가 크다는 뜻은 이미 시간이 많이 경과된 데이터를 재생하고 있다는 뜻이므로 end-to-end delay가 커지는 문제가 있었다. 이러한 문제를 해결하기 위해 현재 버퍼의 수준에 따라 프레임의 재생 속도를 가변적으로 하는 접근 방식을 취하게 되었다. 그 동작방식은 다음과 같다.

버퍼에 많은 양의 데이터가 있을 경우 재생 속도를 빠르게 하여 현재의 데이터에 빠른 속도로 근접하게 하며 버퍼의 수준에 따라 재생 속도를 가변적으로 조절하게 함으로써 가능한 한 최신의 데이터를 재생하도록 하는 기법이다. 코드상의 구현은 아래와 같다.

```
static int nFrameRate = 666665;

        if (nRead+30<nWrite)
                nFrameRate=222222;
        else if (nRead+15<nWrite)
                nFrameRate=333333;
        else if (nRead+7<nWrite)
                nFrameRate=444444;
        else if (nRead+2<nWrite)
                nFrameRate=555555;
        else if (nRead < nWrite)
                nFrameRate=666665;
```

그림 3 버퍼 관리 구현

## 4. 실험 망 구축과 성능평가

고품질 화상회의 응용을 위한 서울대학교(SNU)와 한국전자통신연구원(ETRI) 간의 실험 망 구성은 그림 5와 같다 [6] 이러한 망에서의 간단한 실험을 통해 bandwidth 사용량과 end to end delay 같은 값들을 측정할 수 있었다.

그림 4 SNU-ETRI 간 native IPv6 망

### 4.1 bandwidth 사용량

본 응용에서는 비디오를 MPEG4, MJPEG, MS Video1등의 codec으로 인코딩하고 오디오를 MP3 codec으로 인코딩 하였다. 비디오의 경우 320x240 pixel 크기의 비디오를 전송할 때 결과로 나오는 데이터는 350~400Kbps 정도가

되었으며 오디오의 경우 16bit 샘플링, 스테레오, 샘플링 속도 44.1kHz 로 하여 전송을 했을 때 결과 데이터는 56Kbps의 정도의 bandwidth를 사용하였다. 종합해 보면 0.5Mbps보다 작은 bandwidth를 이용하여 화상과 음성의 전송이 가능하였다.

### 4.2 UCL vic와의 비교

기존에도 IPv6 프로토콜 상에서 멀티캐스트를 이용한 화상회의 어플리케이션을 구현하려는 움직임이 있었다. 그 대표적인 예가 University College London (UCL) 의 vic인데[7] 이번 절에서는 UCL 의 vic과 본 연구팀이 구현한 어플리케이션(HVIC)을 bandwidth 측면에서 비교해 보도록 하겠다. 아래 그림4 와 그림5는 UCL의 vic과 HVIC의 시간에 따른 bandwidth를 비교한 그림이다.

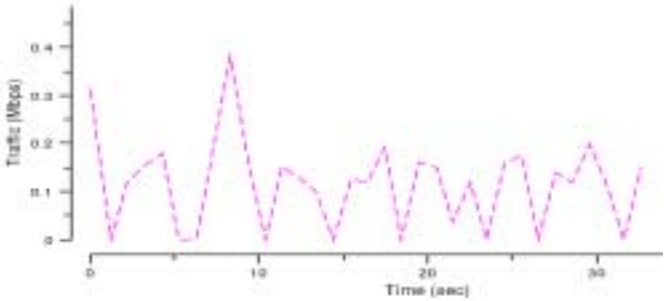


그림 4 UCL vic 의 시간에 따른 bandwidth 변화

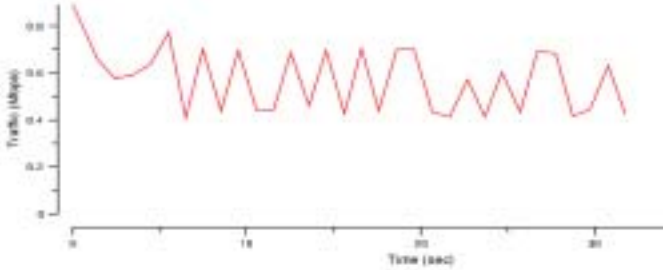

그림 5 본 응용프로그램의 시간에 따른 bandwidth 변화

UCL의 vic 의 경우는 H.261을 사용하였고 초당 10프레임을 전송했을 때의 그래프이며 HVIC의 경우는 MPEG4 코덱을 사용 초당 15프레임을 전송했을 때의 그래프이다. 평균적으로는 HVIC이 좀더 많은 bandwidth를 사용하는 것을 볼 수 있으며, 시간에 따른 bandwidth의 변화폭은 HVIC이 UCL의 변화폭보다 작은 것이 관찰되었다.

### 4.3 delay 분석

본 화상 회의 소프트웨어의 어느 부분에서 얼마만큼의 delay를 차지하는가 아는 것은 향후 본 응용의 성능 개선을 위해 필요한 작업이다. 이에 본 연구팀은 소프트웨어의 각 모듈에서 얼마만큼의 delay를 차지하는 지를 간단하게 측정해보았다. 아래의 그림은 각각 비디오와 오디오의 delay를 분석한 결과이다.

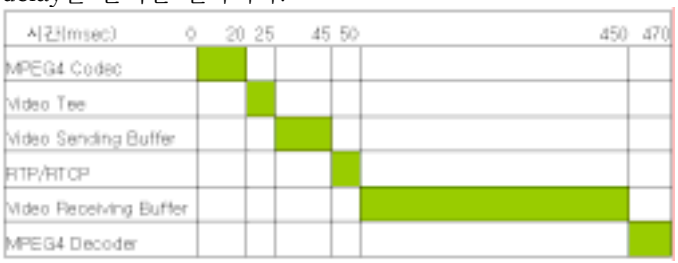


그림 6 비디오의 delay 측정결과

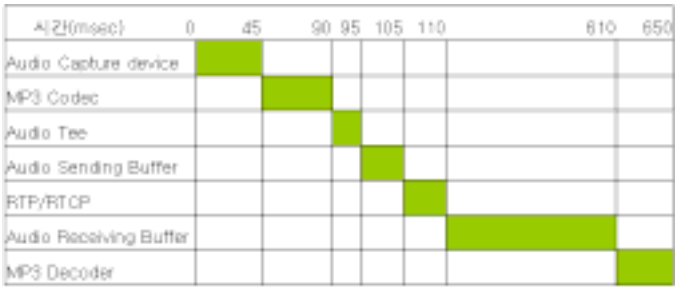


그림 7 오디오의 delay 측정결과

비디오나 오디오의 경우 모두 수신 측 버퍼에서의 delay가 가장 크다는 것을 확인할 수 있다. 이는 실시간 멀티미디어 응용의 특성상 jitter에 의한 끊김 현상을 막기 위해 일정량의 data를 버퍼링 한 후 재생시키기 때문이다. 향후 좀 더 효율적인 수신 측 버퍼관리 알고리즘을 개발하게 된다면 본 응용의 성능이 상당부분 개선될 것으로 보인다.

### 4.4 CPU 점유율 측정

멀티캐스트를 이용한 화상회의의 특징은 사용자의 수가 $N$이고 화상을 전송하는데 필요한 CPU가 $C$, 전송된 화상을 재생하는데 필요한 CPU가 $c$라 했을 때 총 필요한 CPU 프로세싱 파워는 $C + N*c$ 로 참여한 사용자의 수가 증가함에 따라 선형적으로 CPU 프로세싱 파워를 요구한다는 것이다. 때문에 필요한 CPU 점유율을 낮추는 노력이 필요하다. CPU점유율 측정결과 펜티엄4 1.5GHz에서 3자간 회의 시 약 70%의 CPU 점유율, 펜티엄4 2.4GHz에서 3자간 회의 시 약 60%의 CPU 점유율을 보이고 있음을 확인할 수 있었다.

## 5. 결론

개발된 응용은 한국전자통신연구원과 서울대학교 간 IPv6 망에서 원활하게 동작하며, 만족스러운 비디오와 오디오 품질을 제공하면서도 CPU의 효율을 최대화 시켰다. 다만 오디오와 비디오의 동기화(synchronization)를 하지 않아 비디오와 음성이 약간 다른 지연을 가지고 실행되며 약 900ms의 높은 지연을 갖는다. 이러한 문제점들은 향후 개선해나갈 계획이며 본 과제의 결과물은 손쉽게 보급이 가능하여 IPv6 활성화에 크게 기여함과 동시에 다양한 트래픽 통계 데이터를 제공하고 있어서 IPv6 망에서의 전송 성능 측정 및 개선에 기여할 수 있을 것으로 기대된다.

### 참고 문헌

[1] 신명기, 김용진, “IPv6 도입을 위한 기술개발 동향”, IPv6 포럼 코리아 기술문서 2000-002, http://www.ipv6.or.kr
[2] Linda S. Cline, John Du, Bernie Keany, K. Lakshman, Christian Maciocco, David M. Putzolu , “DirectShow RTP Support for Adaptivity in Networked Multimedia Applications”, IEEE multimedia systems, 1998
[4] RFC 1889, “RTP: A Transport Protocol for Real-Time Applications.”
[5] “RTP Payload Format for MPEG-4 Streams”, draft-ietf-avt-mpeg4-multisl-04.txt
[6] 정재훈, 이승윤, 김용진, “IPv6 PC 라우터 및 호스트 설치 및 설정 방법”, IPv6 포럼 코리아 기술문서 2001-003 , http://www.ipv6.or.kr
[7] http://www-mice.cs.ucl.ac.uk/multimedia/