Journal of Korea Multimedia Society Vol. 12, No. 5, May 2009(pp. 653-663)

멀티미디어 통신을 위한 RTP 패킷 기반의 정밀한 오디오/비디오 동기화 기법

서광덕[†], 지원섭[††], 정순흥[†††]

요 약

미디어 간의 동기화 기능 제공은 멀티미디어 통신 시스템 디자인을 위해 중요한 사항이다. 본 논문에서는 IP 네트워크를 통해 비디오와 오디오를 전송할 때 미디어 간의 정밀한 동기화를 제공할 수 있는 새로운 메카니즘을 제안한다. IP 네트워크를 통해 전송된 비디오와 오디오 신호 사이에 동기화를 제공하기 위해서 일반적으로 RTP와 RTCP 프로토콜을 활용한다. 정밀한 미디어 동기화 제공을 위해 본 논문에서는 비디오와 오디오를 RTP 패킷화하여 전송할 때 RTP 패킷의 헤더에 기록될 타임스탬프 정보로부터 유도해 낼 수 있는 NPT (Normal Play Time)를 이용한다. 제안된 방법에서는 기존의 일반적인 동기화 기법에서 요구하는 RTCP SR (sender report) 패킷과 같은 별도의 제어 정보의 전송 및 처리가 필요 없기 때문에 RTCP 패킷 전송을 위해 필요한 UDP 포트의 개수를 줄일 수 있고 네트워크에 유입되는 제어 트래픽의 량을 경감시킬 수 있는 중요한 장점이 있다.

A Precise Audio/Video Synchronization Scheme Based on RTP Packet for Multimedia Communication

Kwang-deok Seo[†], Won Sup Chi[††], Soon-heung Jung[†††]

ABSTRACT

Synchronization between media is an important aspect in the design of multimedia communicationsystem. This paper proposes a precise media synchronization mechanism for video and audio transport over IP networks. To support synchronization between video and audio bitstreams transported over IP networks, RTP/RTCP protocol suite is usually employed. To provide a precise mechanism for media synchronization between video and audio, we suggest an efficient media synchronization algorithm based on NPT (Normal Play Time) which can be derivable from the timestamp information in the header part of RTP packet generated for the transport of video and audio. In the proposed method, we do not need to send and process any RTCP SR (sender report) packet which is required for conventional media synchronization scheme, and accordingly could reduce the number of required UDP ports and the amount of control traffic injected into the network.

Key words: precise media synchronization(정밀한 미디어 동기화), RTP(실시간 전송 프로토콜), RTCP (RTP 제어 프로토콜), multimedia communicatioin(멀티미디어 통신)

※ 교신저자(Corresponding Author) : 서광덕, 주소 : 강원도 원주시 흥업면 매지리 234(220-710), 전화 : 033)760-2788, FAX : 033)760-4323, E-mail : kdseo@yonsei.ac.kr
접수일 : 2008년 11월 5일, 완료일 : 2009년 4월 6일
[†] 정회원, 연세대학교 컴퓨터정보통신공학부 부교수
[††] 연세대학교 컴퓨터정보통신공학부
(E-mail : smilechi@naver.com)
[†††] 정회원, 한국전자통신연구원 방통미디어연구그룹 선임연구원
(E-mail : zeroone@etri.re.kr)
※ 본 연구는 지식경제부 및 정보통신연구진흥원의 IT 신성장동력 핵심기술개발 사업의 일환으로 수행하였음. [2008-S-006-01, 유무선 환경의 개방형 IPTV (IPTV2.0) 기술 개발]

1. 서 론

최근에 RTP (Real-time Transport Protocol)와 RTCP (RTP Control Protocol)는 시간정보와 각종 QoS (quality of service) 정보를 제공하여 실시간 데이터를 전송하기 위한 프로토콜로서 IETF (Internet Engineering Task Force)에 의해 표준화되었다[1]. RTP와 RTCP는 모두 UDP 상에서 동작하므로 그 하부 프로토콜의 특성상 품질보장이나 신뢰성 기능을 제공하지는 못하지만, 실시간 응용에서 필요로 하는 시간 정보와 정보 매체의 동기화 기능을 제공하기 때문에 현재 인터넷 상에서 실현되고 있는 대부분의 실시간 멀티미디어 애플리케이션 (VOD, AOD, 인터넷 방송, 영상 회의 등)들이 RTP 와 RTCP를 이용하고 있다[2,3].

IP (Internet Protocol) 네트워을 통한 멀티미디어 서비스를 위하여 비디오 및 오디오 비트스트림은 RTP 패킷화 과정을 거치게 되는데[2], RTP 패킷화 과정에서는 다른 종류의 미디어 정보와의 동기화 (synchronization)를 지원하기 위해서 RTP 헤더의 RTP 타임스탬프 (timestamp) 정보를 수신 측에 전송한다[3]. 예를 들면, MPEG-4 비디오를 AAC 등의 오디오와 함께 서비스할 경우 수신 측에서 비디오와 오디오 간에 입술동기화 (lip synchronization)를 지원하기 위해서 RTP 타임스탬프 정보의 전송은 필수적이다. 만약 파일 포맷 (file format) 기반의 MPEG-4 콘텐츠를 활용하여 스트리밍과 같은 비디오 서비스를 실시할 경우에는 파일 포맷 헤더에 기록되어 있는 CTS (composition time stamp) 정보를 기반으로 RTP 타임스탬프 값을 생성하여 RTP 패킷을 만들 수 있다. RTP와 RTCP 기반의 기존의 동기화 방법에서는 오디오와 비디오 간의 입술 동기화 지원을 위해서RTP 패킷의 타임스탬프 정보뿐만 아니라 RTCP 패킷에 의해 전달되는 NTP (Network Time Protocol) 정보도 필요하다. NTP는 서로 다른 미디어에 대해 독립적으로 발생되는 RTP 타임스탬프 값에 대한 공통 시간정보 (Wall Clock) 를 제공함으로써 미디어 간에 서로 동기화 지점이 되는 타임스탬프 값을 수신측에서 계산하게 하며 RTCP SR (sender report) 패킷에 의해 주기적으로 수신측으로 전송이 된다[1-5].

본 연구에서는 압축된 비디오와 오디오 비트스트림을 RTP 패킷을 통해 IP네트웍으로 전달할 때 RTCP 패킷에 대한 전송 및 처리가 필요없이 수신 측에서 비디오와 오디오 간의 정밀한 동기화를 달성할 수 있는 새로운 방법에 대해 제안한다. 수신 측에서는 수신된 비디오 및 오디오에 대한 RTP 타임스탬프 값으로부터 비디오 및 오디오 각각에 대한 NPT (Normal Play Time) 정보를 유도하고 이 NPT 정보를 활용하여 두 미디어 간에 정밀한 동기화를 제공할 수 있는 방법을 제안하게 된다.

2. 기존의 미디어 동기화 시스템

2.1 RTP 기반의 미디어 데이터 전송 시스템

최근 몇 년 사이에 인터넷은 다양한 멀티미디어 서비스를 수용하고 있는 가장 대중적인 네트웍으로 성장하였다. 가장 광범위하게 사용되고 있는 TCP (Transmission Control Protocol) 전송 프로토콜은 메세지 또는 파일 단위의 비실시간 특성을 갖는 데이터의 안정적인 전송에는 적합하게 설계되었지만, 오디오와 비디오 전송과 같은 실시간 멀티미디어 응용에는 적합하지 않다. 이러한 특징으로 인해 실시간 미디어 전송에서는 재전송 및 혼잡 제어 (congestion control)를 수행하지 않는 UDP (User Datagram Protocol)를 사용하게 되지만 UDP 자체로는 대역폭 변동, 패킷 전송 지연 및 손실에 대한 어떠한 대처 능력을 갖고 있지 못하다. 따라서 실시간 멀티미디어 서비스에 적합하도록 전송계층 상위의 애플리케이션 계층 (application layer)에서 보조적인 전송 기능을 담당하는 새로운 수송 프로토콜을 설계하게 되었고, 이러한 접근 방법의 결과로 IETF 에 의해 탄생된 수송 프로토콜이 RTP이다[1]. RTP는 하위의 전송 및 네트워크 계층에 무관하게 설계 되었으나 일반적으로 IP네트웍 기반에서는 IP 패킷의 재조립 (reassembly)과 검사합 (checksum) 기능을 활용하기 위해 그림 1과 같이 UDP를 하위 전송 프로토콜로 사용하는 프로토콜 스택 구조를 채택한다.

RTP는 VoD, AoD, VoIP, VT (Video Telephony)와 같은 멀티미디어 서비스를 위한 실시간 스트리밍 데이터를 수송하기에 적합하도록 설계되어 있어서, MPEG-2/MPEG-4/H.261/H.263/H.264 등의 비디오 데이터와 AAC/EVRC/QCELP/AMR/MP3 등의 오디오 데이터 전달을 위한 표준 수송 프로토콜로서

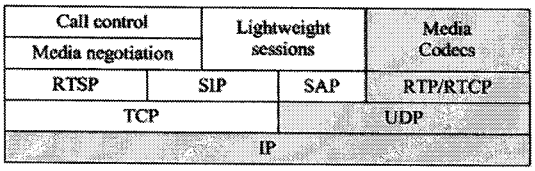


그림 1. IETF 멀티미디어 프로토콜 스택 구조

널리 활용이 되고 있다. 이종의 멀티미디어 서비스 간의 상호 호환성을 보장하기 위해서 다양한 형식의 비디오 및 오디오 데이터를 RTP 유료부하 (payload)에 싣기 위한 표준 방법이 제시되어 있는데, 이를 RTP 페이로드 포맷 (payload format)이라고 지칭한다. RTP 페이로드 포맷은 RTP에 대한 표준을 기술하고 있는 RFC 3550과는 별도의RFC (Request for Comment)로 제안되고 있는데, 표 1에 보이듯이 주요 비디오 및 오디오 압축 표준에 대한 RTP 페이로드 포맷이 IETF의 RFC 문서에 정의되어 있다.

표 1로부터 H.264에 대한 RTP 페이로드 포맷은 IETF에서 이미 RFC 3984로 제정되어 표준이 완료되었다. 슬라이스 또는 픽처 단위의 비트스트림 정보를 RTP 유료부하에 싣게 되는 기존의 H.263, MPEG-4 등의 비디오 압축 표준과 달리, H.264에서는 NAL (Network Abstraction Layer) 계층에서 생성된 NAL 단위 (unit) 별로 RTP 유료부하에 싣도록 되어 있다[6].

2.2 RTP/RTCP 기반의 미디어 동기화 시스템

그림 1에서 RTP와 함께 사용이 되는 RTCP는 데이터 전달 기능은 없지만, RTP에 대한 제어 프로토콜로서 서로 다른 미디어에 대한 동기화 정보를 제공하는 중요한 역할을 수행한다. 대부분의 실시간 멀티미디어 응용에서 필요로 하는 미디어 샘플링 시점에 관한 시간 정보와 이 시간 정보를 기반으로 하는 미디어 간의 동기화 기능을 RTP/RTCP의 협동 작용에 의해 제공할 수 있기 때문에 거의 대부분의 실시간 멀티미디어 응용들이 RTP와 RTCP를 동시에 활용한다.

RTP 패킷에 실려서 전송되는 미디어 간에 동기화를 맞추기 위해서는 RTP 타임스탬프 정보와 NTP 타임스탬프 정보가 동시에 필요하다. RTP 타임스탬프 정보는 그림 2에서 RTP 헤더 부분에 존재하는 32비트 필드로 RTP 페이로드에 실리는 미디어 데이터의 첫 번째 바이트의 샘플링 순간을 나타낸다[1]. 이 타임스탬프 정보는 오디오와 비디오 세션 각각에 대해서 독립적인 클럭 (clock)으로부터 발생하므로 오디오와 비디오 각각의 타임스탬프 시간정보만으로는 이 두가지 미디어 간의 시간적 연관성을 유도해 낼 수 없다. 따라서, 클라이언트에서 오디오와 비디오 간의 동기화를 제공하기 위해서는 오디오와 비디오 각각의 타임스탬프 간의 상호 연관성을 유도해 낼 수 있는 절대 기준시간 (absolute reference time) 정보가 서버로부터 제공이 되어야 한다. 이 역할을 하는 절대 기준시간 정보가 RTCP SR 패킷에 실려서 전송이 되는 NTP 타임스탬프이며 이 절대 기준시간에 해당하는 각 미디어의 상대적 시간인RTP 타임스탬프 값 또한 동일한 RTCP SR 패킷에 실려서 전송이 된다[3,5].

그림 3은 RTCP SR 패킷의 구조를 나타낸다. NTP 타임스탬프 정보는 64비트 필드로 전달되고 RTCP SR 패킷이 보내진 순간의 절대 기준시간을 나타내며 이 NTP 타임스탬프 시점에 해당하는 미디어의 RTP 타임스탬프 값도 그림에 보이듯이 함께 전달된다.

RTCP SR 패킷을 통해 동일한 NTP 타임스탬프

표 1. 주요 오디오, 비디오 압축 표준에 대한 표준 RTP 페이로드 포맷

Payload Format	IETF Specification	Description
Audio/AAC	RFC 3016	MPEG-4 AAC
Audio/EVRC	RFC 3558	3GPP2 EVRC
Audio/QCELP	RFC 2658	PureVoice QCELP audio
Audio/AMR	RFC 3267	ETSI AMR and AMR-WB
Video/MPA	RFC 2250	MPEG audio (e.g., MP3)
Video/MPV	RFC 2250	MPEG-1, 2 video
Video/MP4V	RFC 3016	MPEG-4 video
Video/H261	RFC 2032	ITU H.261 video
Video/H263	RFC 2429	ITU H.263 video
Video/H264	RFC 3984	ITU H.264 video

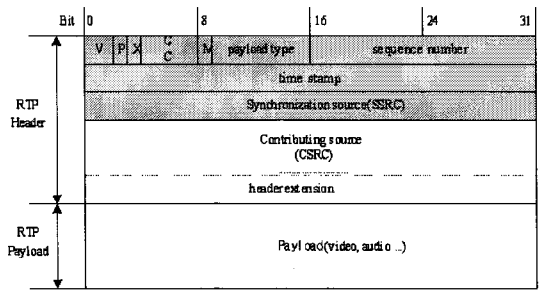


그림 2. RTP 패킷의 구조

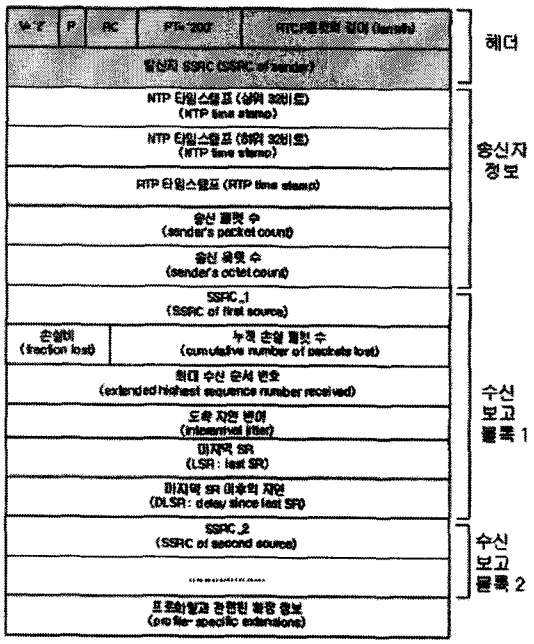


그림 3. RTCP SR (Sender Report) 패킷의 구조

에 해당하는 비디오와 오디오 각각의 RTP 타임스탬프를 알아 낼 수 있다. 같은 NTP 타임스탬프에 상응하는 비디오와 오디오의 RTP 타임스탬프가 각각 NTP_TSV, NTP_TSA이면, 이것을 토대로 비디오의 RTP 타임스탬프 TSV 에 해당하는 오디오의 RTP 타임스탬프 TSA를 다음 식을 통하여 얻어낼 수 있다 [3-5].

$$TS_A = (SR_A/SR_V) \times (TS_V - NTP\_TS_V) + NTP\_TS_A \quad (1)$$

이 식에서 SRA는 오디오의 샘플링율 (sampling rate)을 의미하고, SRV는 비디오의 샘플링율을 나타낸다. 그림 4는 식 (1)을 바탕으로 TSV=6000에 해당

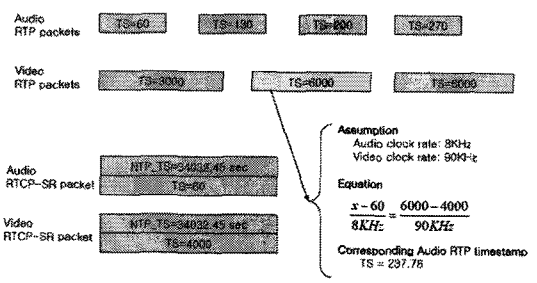


그림 4. RTP 타임스탬프와 RTP 타임스탬프를 이용한 오디오/비디오 동기화 예

하는 TSA를 계산하는 예를 나타낸다. 이 예에서 NTP 타임스탬프는 34032.45초이고 NTP_TSV 와 NTP_TSA는 각각 4000과 60이므로 TSV=6000에 해당하는 TSA는 237.78로 계산된다. 이 결과로부터 TSV=6000에 해당하는 비디오 데이터는 TSA=237.78에 가장 가까운 오디오 데이터와 동기화가 이루어져야 한다.

이처럼 RTCP SR 패킷의 NTP 타임스탬프와 RTP 타임스탬프를 이용하여 미디어 간의 동기화를 제공할 수 있지만, 64비트 필드로 전달되는 NTP 타임스탬프를 주기적으로 처리해 주어야 하고, 서버 단의 시스템 처리 지연, 네트워크 상에서 발생하는 패킷 지연 등으로 인하여 동일한 NTP 타임스탬프와 상응하는 NTP_TSV와 NTP_TSA를 구하기 위해서는 클라이언트 단의 로컬 시스템 클럭을 통한 별도의 계산 과정을 거쳐야 하는 문제점이 있다. 또한 RTCP SR 패킷 전송에 의한 제어 트래픽 발생으로 네트워크에 체증을 가중시킬 수 있다.

3. 제안된 오디오/비디오 동기화 알고리듬

오디오와 오디오 스트림 간의 미디어 동기화를 맞추기 위하여 본 논문에서는 RTP 타임스탬프로부터 얻어낼 수 있는 NPT정보를 이용한다. NPT는 비디오 및 오디오 데이터 각각의 RTP 타임스탬프만으로 그 값을 유도해 낼 수 있고 동일한 NPT 시점에서 비디오 및 오디오 데이터를 출력할 경우 비디오와 오디오 간에 동기화를 제공할 수 있다.

3.1 RTP 타임스탬프를 이용한 NPT 계산

오디오와 오디오 스트림 간의 미디어 동기화를 맞추기 위하여 본 논문에서는 RTP 타임스탬프로부터 얻어낼 수 있는 NPT정보를 이용한다. NPT는 비디오 및 오디오 데이터 각각의 RTP 타임스탬프만으로 그 값을 유도해 낼 수 있고 동일한 NPT 시점에서 비디오 및 오디오 데이터를 출력할 경우 비디오와 오디오 간에 동기화를 제공할 수 있다.

현재 디스플레이 장치에 출력되는 k번째 비디오 화면의 NPT인, $NPT_{V_o}^k$은 RTP 타임스탬프 정보를 이용하여 다음 식으로부터 유도해 낼 수 있다.

$$NPT_{V_o}^k = \left(RTPT_{V_o}^k - RTPT_{V_o}^1\right) / SR_V . \quad (2)$$

여기서 $RTPT_{V_o}^1$는 첫번째 출력화면 (I-픽처)의 RTP 타임스탬프이고, $RTPT_{V_o}^k$는 k번째 출력화면에 해당되는 RTP 타임스탬프를 나타내며 SR_V은 전송단에서 비디오의 접근단위(Access Unit)에 대한 샘플링율을 의미한다.

비디오의 경우 출력되는 단위가 불연속적인 개별적 화면이므로 각 출력화면 단위로 NPT를 손쉽게 구해낼 수 있다. 그러나, 오디오는 출력 단위가 연속적인 PCM (pulse code modulation) 데이터 블록이므로 출력단위를 구분하여 NPT를 직접적으로 얻어낼 수가 없다. 이 문제를 해결하기 위하여 PCM 데이터가 출력되기 이전에 버무르게 되는 웨이브 출력버퍼 (wave-out buffer)의 고정된 크기를 이용하여 오디오에 대한 NPT를 구해 내게 된다. 그림 5는 하나의 오디오 프레임 (frame)이 복호화 후 PCM 데이터로 재생되어 웨이브 출력버퍼에 입력 및 출력되는 과정을 보인다. RTP 패킷으로부터 프레임 단위로 추출된 오디오 압축 데이터는 주기적으로 복호화되어 PCM데이터로 재생이 되고, 재생된 PCM 데이터는 연속적으로 웨이브 출력버퍼에 저장된다. 웨이브 출력버퍼에 저장된 PCM 데이터 블록은 출력 디바이스로 전달이 되고, 디바이스 드라이버에 의해 스피커로 출력된다. 끊임없이 일정한 속도의 오디오 출력을 위하여 웨이브 출력버퍼의 크기는 항상 일정한 값 (t_{buff}) 으로 설정이 된다.

이상의 오디오 데이터 처리 과정을 바탕으로 웨이브 출력버퍼에서 현재 출력될 예정인 s번째 PCM 데이터 블록에 대한 RTP 타임스탬프인 $RTPT_{A_o}^s$를 다음 식을 통해 추정해 낼 수 있다.

$$RTPT_{A_o}^s = RTPT_{A_i}^n - \left(t_{buff} \times SR_A\right). \tag{3}$$

여기서 $RTPT_{A_i}^n$는 $RTPT_{A_o}^s$를 계산할 순간에 웨이브 출력버퍼로 입력되는 n번째 PCM 데이터의 RTP 타임스탬프 값을 나타내며, SR_A는 오디오의 기본 접근단위인 프레임에 대한 샘플링율을 의미한다. 따라서 곧바로 스피커로 출력 예정인 s번째 PCM 데이터 블록의 NPT인, $NPT_{A_o}^s$는 식 (3)의 결과를 이용하여 다음과 같이 계산할 수 있다.

$$NPT_{A_o}^s = \left(RTPT_{A_o}^s - RTPT_{A_o}^1\right) / SR_A. \tag{4}$$

여기서 $RTPT_{A_o}^1$는 첫 번째로 출력된 PCM 데이터 블록의 타임스탬프 값을 나타낸다.

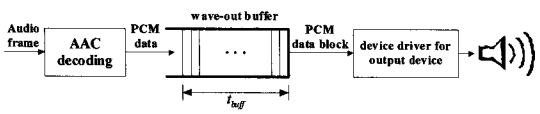


그림 5. PCM 데이터의 웨이브출력버퍼 입출력 과정

3.2 NPT 기반의 오디오/비디오 동기화 알고리듬

동기화의 기본 원리는 출력 예정인 비디오 화면의 NPT와 이 화면과 동시에 출력 예정인 오디오 PCM 데이터의 NPT를 비교하여 비디오 화면의 출력 간격을 조절한다. 즉, 오디오 신호가 비디오 신호보다 그 중요성이 상대적으로 높기 때문에 비디오와 상관없이 오디오를 끊김없이 출력 시키게 되며 출력되는 오디오와 동기가 이루어지도록 비디오의 NPT를 오디오의 NPT와 비교하여 비디오의 디스플레이 속도를 조절하게 된다.

제안된 알고리듬의 전체적인 동작 블록도는 그림 6에 나타나 있다. 비디오의 경우 수신된 RTP 패킷으로부터 각 화면별 타임스탬프인 $RTPT_{V_i}^m$를 추출하며, B-픽처를 고려한 화면 순서 재정렬 (reordering) 에 의한 화면 디스플레이 순서를 고려하여 출력 화면별로 $RTPT_{V_o}^k$를 구한다. $RTPT_{V_o}^k$를 바탕으로 식 (2)를 이용하여 k번째 출력 화면의 NPT인, $NPT_{V_o}^k$를 계산하게 된다. 오디오의 경우 RTP에 실려서 도착하는 오디오 프레임 단위로 오디오 복호화를 수행하여 PCM 데이터를 복원한다. 이와 동시에 RTP 패킷 헤더로부터 순서대로 도착하는 오디오 프레임의 RTP 타임스탬프인 $RTPT_{A_i}^n$를 추출해 낸다. $RTPT_{A_i}^n$를 바탕으로 식 (3)을 통해 현재 출력 예정인 PCM 데이터 블록의 타임스탬프인 $RTPT_{A_o}^s$를 추정해 낼 수 있다. 또한 식

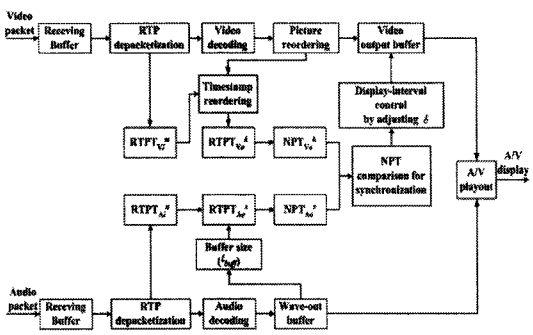


그림 6. PCM 데이터의 웨이브출력버퍼 입출력 과정

(4)를 통해 출력 직전인 PCM 데이터 블록의 NPT인 $NPT^{s}_{A_o}$를 계산해 낼 수 있다. 출력 직전인 화면이 k번째라고 가정하고 이 화면과 동기화를 이룰 예정인 오디오의 PCM 데이터 블록이 s번째라고 가정하면 $NPT^{k}_{V_o}$와 $NPT^{s}_{A_o}$를 비교하여 그 차이 값을 바탕으로 비디오 화면의 출력 시간 간격을 조절하여 동기화를 맞추게 된다.

NPT 비교에 사용될 $NPT^{k}_{V_o}$와 $NPT^{s}_{A_o}$의 차이값인 T_s는 다음과 같이 구한다.

$$T_s = NPT^{k}_{V_o} - NPT^{s}_{A_o}. \quad (5)$$

$|T_s|$가 설정된 동기화 영역 (in-sync region)의 한계값인 동기화 문턱 값 $\eta(>0)$ 이내일 경우 동기화가 맞추어진 것으로 판단하여 현재 화면율에 의한 화면 출력 간격으로 비디오 화면을 디스플레이 하게 된다. 그러나, $|T_s|$가 η를 초과할 경우 비디오가 오디오보다 빠른 출력상태인지 또는 느린 출력상태인지를 판단하여 비디오의 화면간 출력 시간 간격을 조절하게 된다. 비디오 화면의 출력 시간 간격 (display-interval) f_I는 주어진 화면율 f_R에 의해 다음과 같이 계산된다.

$$f_I = 1000/f_R (\text{ms}). \quad (6)$$

$|T_s|$가 η를 초과할 경우 화면 간격 크기 조절 파라미터인 δ는 스케일 팩터 (scale factor) s_f에 의해 크기가 결정되며 다음 식과 같이 표현된다.

$$\delta = T_s \cdot s_f (\text{ms}). \quad (7)$$

s_f는 동기화가 이루어지지 않은 경우 다시 동기화를 맞추기 위한 수렴속도를 조절하는 역할을 하며 0.05~0.1 정도의 값이 적당함을 실험을 통하여 확인하였다. δ에 의해 조정된 새로운 화면 출력 간격 f_I'은 다음과 같이 계산 된다.

$$f_I' = f_I + \delta. \quad (8)$$

식 (5)-(8)을 기반으로 비디오와 오디오 간의 NPT값의 비교를 통해 동기화를 맞추기 위한 NPT 비교 순서도는 그림 7과 같다. 그림 7에서 T_s가 설정된 동기화 문턱 값 η이내일 경우 미디어 동기화가 이루어 졌다고 판단하지만, 그렇지 않을 경우 T_s값이 양수인지 또는 음수인지에 따라서 비디오 화면의 출력 시간 간격을 조절하게 된다. 즉, 값이 음수일 경우

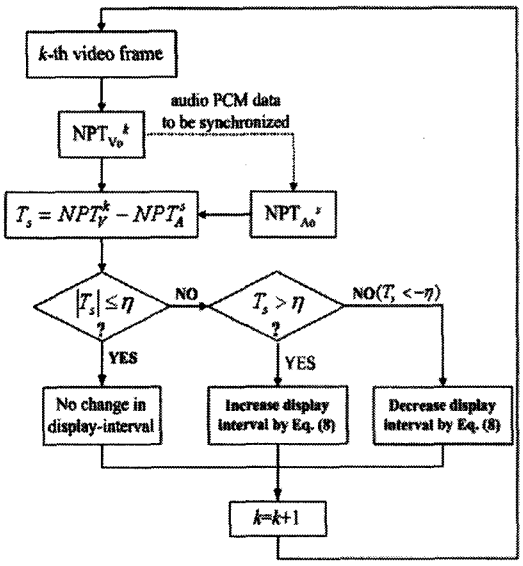


그림 7. 동기화 지원을 위한 NPT 비교 순서도

비디오의 출력이 오디오에 비해 지연되고 있으므로 식 (8)에 의해 비디오의 출력 시간 간격을 줄이게 되고 ($\delta < 0$), T_s 값이 양수일 경우 반대로 비디오의 출력 시간 간격을 늘이게 된다 ($\delta > 0$). 이때, T_s값을 η 이내로 수렴시키기 위한 속도를 제어하기 위하여 식 (7)에 사용된 s_f값을 활용하게 된다.

미디어 동기화를 위한 제안된 방법은 RTP 타임스탬프 정보만을 이용하기 때문에 기존의 일반적인 동기화 방법[3-5]이 요구하는 RTCP SR 패킷 전송 및 처리가 별도로 필요 없으므로 멀티미디어 서비스 시스템의 동작 구조를 훨씬 간소하게 설계할 수 있고, 원하는 동기화 정밀도를 η값의 제어함으로써 멀티미디어 서비스의 동기화 품질을 조절할 수있다. 또한, RTCP 패킷 전송을 위해 필요한 UDP 포트의 개수를 줄일 수 있고 RTCP 패킷 전송이 필요 없기 때문에 네트워크에 유입되는 제어 트래픽의 량을 경감시킬 수 있는 장점이 있다.

4. 실험 결과

제안된 동기화 기법은 비디오 및 오디오 압축 표준 기법의 종류에 관계없이 전송 프로토콜로서 RTP를 사용하는 모든 멀티미디어 응용 서비스에 포괄적으로 적용이 가능하다. 특히, 오디오와 비디오 간에 정밀한 입술동기화 (lip synchronization)를 요구하

는 미디어 응용 서비스에 효과적으로 활용될 수 있다. 제안된 기법의 동기화 성능을 검증하기 위하여 H.264 비디오와 AAC 오디오를 인터넷을 통하여 스트리밍 서버에서 클라이언트로 전송한다. 스트리밍 서버로는 Apple 사의 QuickTime 스트리밍 서버인 Darwin Streaming Server (DSS)[8]를 활용하였고 클라이언트는 DSS와 호환이 가능하도록 자체 구현한 프로그램을 사용하였다. DSS는 그림 8에 보이는 스트리밍 프로토콜 스택 구조를 기반으로 구현되어 있고 RTSP (Real-time Streaming Protocol) 를 기반으로 미디어 세션에 대한 개시를 시작하는데 클라이언트와의 RTSP 메쏘드 (method) 통신을 통해 RTP/RTCP의 채널을 설정하고 비디오 및 오디오 스트림을 RTP를 통해 전송하게 된다.

먼저, 미디어 동기화의 필요성을 확인하기 위하여 동기화 기능이 지원되지 않을 경우 비디오와 오디오의 출력 데이터 간의 시간적 어긋남(temporal skew)을 측정하였다. 이 실험에서는 시간적 동기화 기능이 제공되지 않기 때문에 비디오와 오디오가 서로 독립적으로 복호화 및 출력이 이루어지게 된다. 미디어 동기화 기법의 적용에 의한 제어가 이루어지지 않을 경우 그림 9에 보이듯이 시간이 지날수록 미디어 간의 시간적 어긋남은 더욱더 커지게 됨을 알 수 있다.

그림 10은 기존의 RTP/RTCP 기반의 동기화 기법을 적용했을 때 1600장의 비디오 화면 별로 측정된

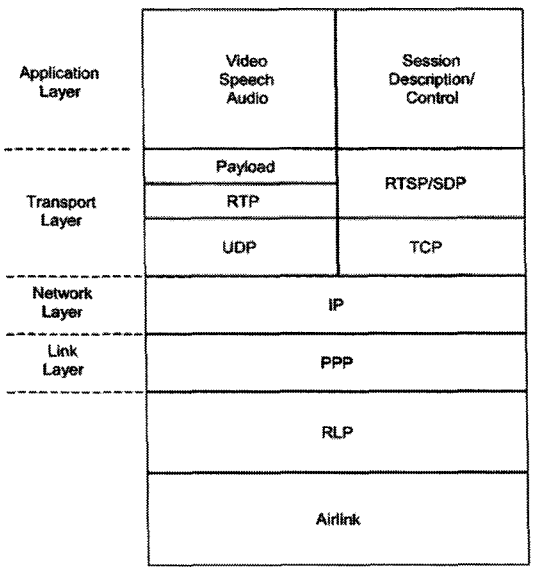


그림 8. 미디어 스트리밍을 위한 DSS 프로토콜 스택 구조

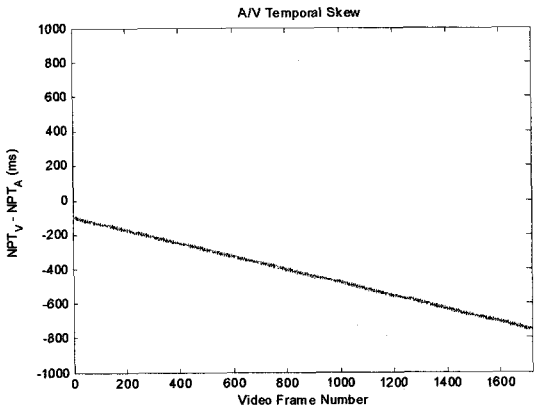


그림 9. 미디어 동기화 기법이 적용되지 않은 경우 비디오와 오디오 간의 화면별 NPT 차이값

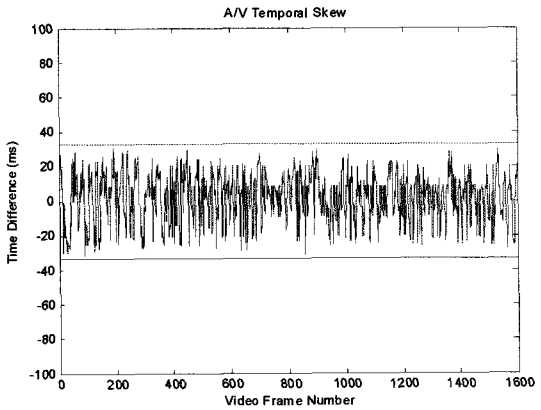
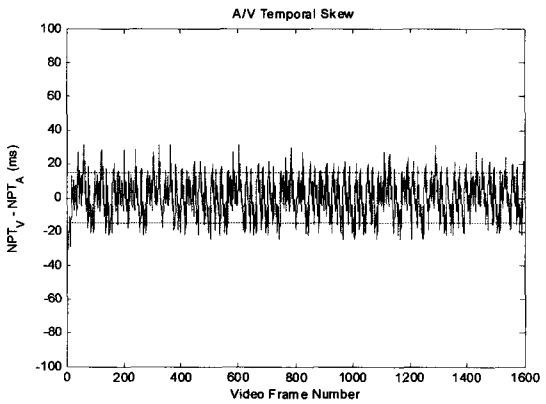


그림 10. 기존의 RTP/RTCP 기반의 동기화 기법이 적용된 경우 비디오오와 오디오 간의 시간적 차이값

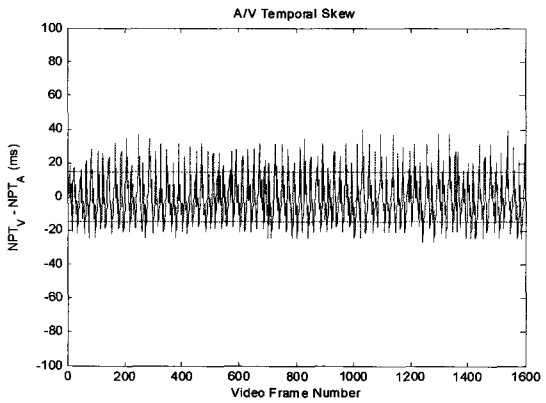
비디오와 오디오 간의 동기화 성능을 나타낸다. 이 실험을 위하여 2.2절에서 설명한 일반적인RTP/RTCP 기반의 동기화 기법을 적용하였다. 비디오와 오디오를 위한 RTCP SR 패킷의 전송 빈도는 RTCP 표준 권고안[1]에 의해 전체 트래픽의 5% 이내가 되도록 각각 3초마다 전송이 되도록 설정하였다. 기존의 RTP/RTCP 기반의 동기화 기법에서는 실제 오디오와 비디오가 디스플레이 되는 시점인 NPT 정보를 전혀 고려하지 않고 오직 오디오 및 비디오 패킷에 실려서 전송되는 RTP 타임 스탬프와 RTCP SR 패킷이 제공하는 NTP 타임스탬프 정보만을 이용하여 동기화를 맞추기 때문에 동기화의 정확도는 RTP 타임스탬프 간격의 정밀도 즉, RTP 패킷 사이의 RTP 타임스탬프 크기의 간격에 큰 영향을 받게 된다. 실험에서는 초당 30장의 비디오를 전송하였기 때문에

각 화면별 RTP 타임스탬프의 간격이 시간적으로 33ms 에 해당이 되고, 2.2절의 동기화 원리를 적용하여 각 비디오 화면의 RTP 타임스탬프에 가장 가까운 RTP 타임스탬프를 갖는 오디오 패킷에 동기를 맞출 경우 최대 ±33ms 의 동기화 오차가 발생할 수 있다.

그림 11은 제안된 동기화 기법의 정밀한 동기화 성능을 검증하기 위하여 오디오와 비디오 간의 η 값을 매우 작은 값인 15ms 로 설정하고 각각에 대한 s_f 값을 0.06과 0.1로 설정하여 제안된 동기화 알고리듬을 적용하여 실험한 결과이다. 제안된 알고리듬은 그림 7에 보이듯이 비디오 화면마다 NPT를 비교하기 때문에 실험결과도 비디오 화면 단위로 비디오와 오디오의 NPT 값의 차이인 T_s를 측정하였다. 제안된 알고리듬의 적용으로 ± 15 ms인 동기화 한계 영역 내부로 T_s 값이 정밀하게 제어됨을 확인할 수 있다.



(a) s_f=0.06인 경우



(a) s_f=0.1인 경우

그림 11. 제안된 동기화 방법이 적용된 경우 비디오와 오디오 간의 화면별 NPT 차이값 (η = 15ms)

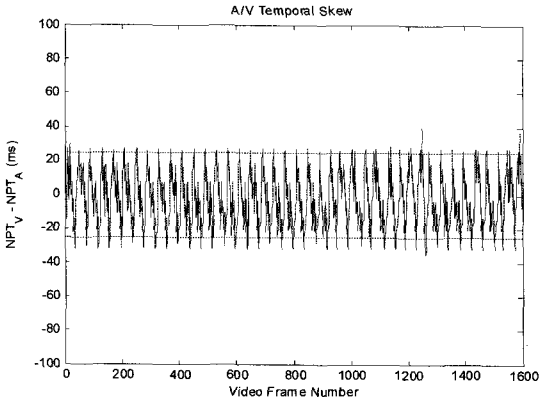
표 2. s_f 값에 따른 동기화 영역을 벗어나는 빈도수와 수렴속도 비교 (η = 15ms)

s_f	동기화 영역을 벗어나는 빈도수	동기화 영역으로 재진입한 후 측정된 모든 T_s 값의 절대치의 평균 (ms)
0.06	189	9.4
0.1	247	6.3

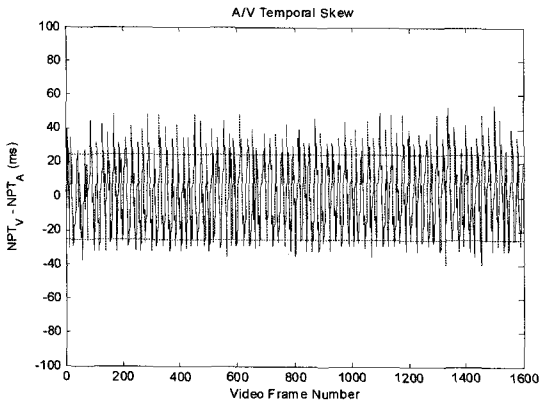
T_s 값이 순간적으로 동기화 한계 영역의 범위인 ±15 ms를 벗어날 경우 한계 영역 이내로 수렴하는 속도는 s_f 값의 차이로 인하여 s_f 값이 큰 그림 11(b)의 결과가 그림 11(a) 보다 훨씬 큼을 알 수 있다. 그런데, 그림 11(b)의 경우 수렴 속도가 빠른 반면에 화면간 출력 시간 간격의 변동량도 커지기 때문에 순간적으로 동기화 한계 영역을 벗어나는 빈도는 그림 11(a) 보다 많음을 알 수 있다.

표 2는 그림 11의 결과에서 s_f 값의 차이로 인해 동기화 영역을 벗어나는 빈도와 수렴 속도를 비교하고 있다. 동기화 영역을 벗어나는 빈도는 전체 1600장의 화면에 대해 ±15 ms를 벗어난 회수를 측정하였고, 수렴 속도는 ±15 ms 범위를 벗어난 직후에 다시 동기화 영역으로 재진입한 후 측정된 모든 T_s 값의 절대치의 평균을 측정하였다. T_s 값의 절대치의 평균이 작은 경우가 완벽한 동기화를 의미하는 시간 차이 값 0에 대한 수렴 속도가 빠른 경우이다. 표 2의 결과로부터 s_f 값이 0.1인 경우가 0.06인 경우보다 동기화 영역을 벗어나는 빈도가 많지만 0으로 수렴하는 속도는 더 빠름을 확인할 수 있다. 따라서 s_f 값의 설정에 있어서 수렴 속도와 동기화 한계 영역을 벗어나는 빈도 사이에는 서로 상충관계(trade-off)가 존재함을 알 수 있고, 제공되는 서비스의 만족도를 고려하여 적절한 값으로 조절할 수 있다.

그림 12는 그림 11의 경우보다 η의 크기를 크게 하여 25ms로 설정한 경우의 실험 결과이다. s_f 값 또한 마찬가지로 0.06과 0.1로 설정하였다. 그림 11과 마찬가지로 설정된 한계값인 25ms 이내로 비디오와 오디오 간의 시간적 어긋남을 제한하여 제어하고 있음을 확인할 수 있다. 또한 s_f 값의 차이로 인해 그림 12(a)와 12(b) 간에 동기화 한계 영역을 벗어나는 빈도와 벗어났을 때 0으로 수렴하는 속도에 차이가 있음을 확인할 수 있다. 표 3은 그림 12의 결과에서 s_f 값의 차이로 인해 동기화 영역을 벗어나는 빈도와



(a) s_f=0.06인 경우



(a) s_f=0.1인 경우

그림 12. 제안된 동기화 방법이 적용된 경우 비디오와 오디오 간의 화면별 NPT 차이값 (η=25ms)

표 3. s_f 값에 따른 동기화 영역을 벗어나는 빈도수와 수렴속도 비교 (η=25ms)

s_f	동기화 영역을 벗어나는 빈도수	동기화 영역으로 재진입한 후 측정된 모든 T_s 값의 절대치의 평균 (ms)
0.06	102	15.8
0.1	145	11.2

수렴 속도를 비교하고 있다. 동기화 영역을 벗어나는 빈도는 전체 1600장의 화면에 대해 ±25 ms를 벗어난 회수를 측정하였고, 수렴 속도는 ±25 ms 범위를 벗어난 직후에 다시 동기화 영역으로 재진입한 후 측정된 모든 T_s 값의 절대치의 평균을 측정하였다. T_s 값의 절대치의 평균이 작은 경우가 완벽한 동기화를 의미하는 시간 차이값 0에 대한 수렴 속도가 빠른 경우이다. 표 3의 결과로부터 s_f 값이 0.1인 경우가 0.06인 경우보다 동기화 영역을 벗어나는 빈도가 많지만 수렴 속도는 더 빠름을 확인할 수 있다.

이상의 실험 결과로부터 제안된 기법을 적용할 경우 원하는 동기화 정밀도를 동기화 문턱 값인 η를 조절함으로써 제어할 수 있고, 설정된 η값을 충실히 만족시키도록 정확한 동기화가 수행이 됨을 확인할 수 있다.

주관적 관점에서 제안된 동기화 방법의 효과를 검증하기 위해 10명의 관찰자가 오디오/비디오 서비스 기반의 멀티미디어 스트리밍 서비스의 품질을 평가하였다. 10명의 관찰자는 표 4에 나타나 있는 7 단계의 오디오/비디오 신호 간의 동기화 품질을 선택한다. 선입관에 의한 편견이 개입되는 것을 배제하기 위하여 각 관찰자에 대한 실험은 블라인드 (blind) 테스트 방식을 통해 관찰자가 테스트 대상인 멀티미디어 스트리밍 서비스에 적용된 동기화 방법에 대해 알지 못하게 했다. 10명의 관찰자에 의한 표 4를 활용한 평가 결과의 모든 점수는 평균화 되어 표 5에 나타나 있다. 동기화 문턱 값인 η 값을 낮게 설정할 수록 동기화 품질이 우수하게 평가됨을 알 수 있다. 특히, 제안된 방법의 경우 동기화를 적용하지 않은 경우보다 매우 우수한 품질을 나타낼 뿐만 아니라 기존의 RTP/RTCP기반의 동기화를 적용한 결과와도 거의

표 4. 주관적 동기화 품질 평가를 위한 7 단계 점수

Impairment Class	Score
Not noticeable	1
Just noticeable	2
Definitely noticeable but only slight impairment	3
Impairment not objectionable	4
Somewhat objectionable	5
Definitely objectionable	6
Extremely objectionable	7

표 5. 평균적인 동기화 품질 평가 결과 비교

	동기화 문턱값 (η)			
	20 ms	30 ms	50 ms	100 ms
동기화 기법을 적용 안한 경우	5.94			
RTP/RTCP 기반의 동기화 적용	2.12			
제안된 동기화 적용	1.31	1.52	2.19	2.61

유사한 품질을 나타낸다. 제안된 기법의 경우 임의로 η 값을 설정하여 제어할 수 있기 때문에 η 값 20ms, 30 ms, 50ms, 100ms 등으로 임의로 설정하여 동기화 품질을 제어할 수 있다. 반면에, 기존의 RTP/RTCP 기반의 동기화 기법에서는 단순히 RTP 타임스탬프 값을 비교하여 동기화를 설정하기 때문에, 비디오 전송 시스템의 설계자가 서비스에 적용될 동기화의 정밀도를 임의로 조절하기가 거의 불가능하다. 표 4에 나타나 있는 RTP/RTCP기반의 동기화 품질은 30 fps 의 화면율을 기준으로 비디오 서비스를 제공했기 때문에 동기화 정확도는 ±33ms 이내에서 변동하게 된다.

5. 결 론

본 논문에서는 RTP 패킷에 실려서 전송되는 비디오와 오디오 스트림의 NPT를 유도하여 수신 측에서 미디어 간의 동기화를 제공할 수 있는 새로운 동기화 기법을 제안하였다. 제안된 방법에서는 비디오와 오디오의 RTP 패킷으로부터 NPT 정보를 유도하여 동기화를 제공하기 때문에 기존에 요구되었던 RTCP SR 패킷과 같은 별도의 제어 정보의 전송 및 처리가 필요 없다. 따라서 멀티미디어 통신 서비스 시스템의 동작 구조를 훨씬 간소하게 설계할 수 있고, 제어 정보 전달에 필요한 UDP 포트 생성과 과도한 네트워크의 대역폭 점유를 피할 수 있다. 제안된 방법을 적용할 경우 원하는 동기화 정밀도를 동기화 문턱값을 조절함으로써 제어할 수 있고, 설정된 동기화 문턱값 이내로 비디오와 오디오의 NPT 차이 값이 유지되도록 제어할 수 있다. 제안된 방법은 동기화를 위해 별도의 제어 데이터 생성이 필요 없고, RTP 패킷만을 이용하여 미디어 간의 동기화를 제공한다는 측면에서 실용성과 효율성이 매우 높은 방법이라고 할 수 있다.

참 고 문 헌

[1] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, "Real-time transport protocol," IETF *RFC 3550*, July 2003.

[2] D. Wu, Y. Hou and Y. Zhang, "Transporting real-time video over the Internet: Challenges and approaches," *Proceedings of the IEEE*, Vol.88, No.12, pp. 1855-1877, Dec. 2000.

[3] L. Bertoglio and P. Migliorati, "Intermedia synchronization for video conference over IP," *Signal processing: Image Communication*, Vol.15, No.1, pp. 149-164, 1999.

[4] A. Boukerche and H. Owens, "Media synchronization and QoS packet scheduling algorithms for wireless systems," *Mobile Networks and Applications*, Vol.10, No.1, pp. 233-249, Feb. 2005.

[5] F. Segui, J. Cebollada and J. Mauri, "Multimedia group synchronization algorithm based on RTP/RTCP," *IEEE Int. Symp. on Multimedia*, pp. 754-757, San Diego, USA, Dec. 2006.

[6] S. Wenger, M. Hannuksela, M. Westerlund and D. Singer, "RTP payload format for H.264 video," IETF *RFC 3984*, Feb. 2005.

[7] T. Wiegand, G. Sullivan, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, pp. 560-576, July 2003.

[8] Apple Darwin Streaming Server (DSS)- http://developer.apple.com/darwin/projects/streaming.

서 광 덕

1996년 2월 KAIST 전기및전자공학과 공학사

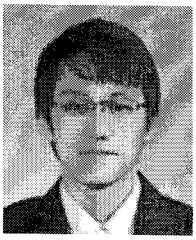
1998년 2월 KAIST 전기및전자공학과 공학석사

2002년 8월 KAIST 전기및전자공학과 공학박사

2002년 8월~2005년 2월 LG전자 단말연구소 선임연구원

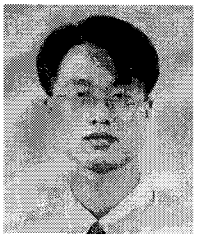
2005년 3월~현재 연세대학교 컴퓨터정보통신공학부 부교수

관심분야 : 영상부호화, 영상통신, 멀티미디어 통신시스템 구현



지 원 섭

2007년 8월 연세대학교 컴퓨터정보통신공학부 학사
2008년 3월~현재 연세대학교 컴퓨터정보통신공학부 석사과정
관심분야 : 영상부호화, 영상통신, 멀티미디어 통신시스템 구현



정 순 흥

2001년 2월 부산대학교 전자공학과 학사
2003년 2월 KAIST 전기및전자공학과 석사
2003년 3월~2005년 3월 LG전자 단말연구소 주임연구원
2005년 4월~현재 ETRI 방통미디어연구그룹 선임연구원
관심분야 : 디지털방송, 영상압축, IPTV, 영상통신