

압축 도메인에서 슬라이스를 기반으로 하는 H.264/AVC 비디오 스트림들의 혼합

준회원 장 승 민*, 정희원 유 흥 연*, 김 상 균*, 박 일 균**, 홍 성 훈*

A Mixing of H.264/AVC Video Streams Based on Slice in The Compressed Domain

Seung-Min Jang* *Associate Member,*

Hong-Yeon Yu*, Sang-Kyune Kim*, Il-Kyun Park**, Sung-Hoon Hong* *Regular Members*

요 약

본 논문에서는 저연산, 저지연성을 보장하는 다중 비디오 스트림 혼합 알고리즘을 제안한다. 제안된 알고리즘은 슬라이스기반 H.264/AVC 비디오 스트림들을 화소 도메인에서 혼합하지 않고 압축 도메인에서 선택스 비트스트림 변경에 의해 혼합한다. 혼합을 위한 선택스 비트스트림 변환은 혼합화면을 구성할 비디오 스트림들의 위치에 따라 변경한다. 실험결과 4개 이상의 비디오 스트림들을 고속으로 혼합할 수 있었다. 또한 표준 H.264/AVC를 지원하는 플레이어에서 화질열화 없이 재생할 수 있었다.

Key Words : NAL Unit, Slice, SPS, FMIS, Screen Layout

ABSTRACT

In this paper, we propose a novel mixing algorithm for multiple video streams, which can guarantee low complexity and low end-to-end delay. In the propose algorithm, we change the syntax bitstream in the compressed domain. And we mix the H.264/AVC video streams by coded slice unit. The syntax bitstream for a mixing changes according to the arrangement of videos streams in which it organizes the mixing screen layout. Experimental results show more than 4 video streams could be mixed at high speed. Moreover, it could display in the player which it supports the standard H.264/AVC without the picture quality loss.

I. 서 론

최근 통신, 방송, 컴퓨터 기술의 융합은 다양한 고부가가치 서비스를 창출 하였다. 그중에서 IP 네트워크를 이용한 비디오 기반 영상통신 시스템에 관한 연구가 활발하게 진행됨에 따라 다양한 사용자 요구에 상응하는 효율적인 서비스 방법에 대한 관심이 고조되고 있다. 일반적인 비디오 기반 영상통신 시스

템에서 지리적으로 떨어져 있는 단말간의 지점 대 지점(point to point) 나아가 다 지점 대 다 지점(multi-point to multi-point)으로 영상통신을 할 경우, 서로 전송될 비디오 스트림들을 혼합해서 단일 비디오 스트림으로 제공해 주어야 하는 요구사항이 발생할 수 있다. 예를 들어 네트워크기반 DVR 감시시스템에서 다채널을 갖는 입력영상을 압축하여 저장하고 원격지에 있는 관리자가 모니터링하고 있는 다

※ 이 논문은 2008년 정부(교육과학기술부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2008-521-D00299)

* 전남대학교 전자정보통신공학(bravesm@naver.com, keister@vip.chonnam.ac.kr, ksk3171@paran.com, hsh@chonnam.ac.kr)

** 한국전자통신연구원(ikpark@etri.re.kr)

논문번호 : KICS2009-06-234, 접수일자 : 2009년 6월 2일, 최종논문접수일자 : 2009년 10월 6일

지점 대 지점(multi-point to point)서비스의 경우, 다수의 채널을 통해 동시에 수신되는 비디오 스트림들을 복호화하고, 이들을 관리자 단말 환경에 맞게 비디오 스트림을 혼합하여 제공할 수 있는 기능이 필요하다. 또 다른 예로, IP 네트워크를 기반으로 하는 참여형 IPTV 방송 서비스^[1]의 경우, 원격지의 시청자가 방송 중인 서비스에 참여하기 위해서 시청자의 비디오 스트림을 방송 프로그램 공급자에게 전송하고, 방송 프로그램 공급자는 자신의 비디오 스트림과 실시간으로 혼합 및 재부호화하여 전송하는 기능이 필요하다.

다수의 비디오 스트림에 대한 혼합처리가 요구되는 가장 대표적인 서비스에는 화상회의 서비스가 있다. 화상회의 서비스에서는 회의가 둘 이상의 지리적으로 분리된 위치에 있는 다자간에 이루어짐으로써 유발되는 다 지점 신호방식(signaling)과 비디오, 오디오, 데이터와 같은 각종 미디어 처리기능 그리고 원활한 회의 진행을 위한 회의제어 및 관리기능을 처리할 MCU(Multi-point Control Unit)가 필요하다^[2].

화상회의에서 부호화된 비디오 신호의 혼합/절환 처리는 MCU에서 수행하거나 회의에 참가하는 각 단말에서 수행할 수 있는데, MCU에서 혼합 처리를 수행할 경우는 MCU가 복잡해지지만 단말은 비디오 혼합 처리를 위한 부가적인 동작을 수행하지 않으므로 표준 단말이면 된다. 반면에 회의에 참가하는 모든 참가 단말이 비디오 혼합 처리기능을 가지고 있으면 MCU가 간단해진다. MCU에서 비디오 혼합 처리를 수행할 경우, 다수의 비디오 데이터들을 복호화하고 부호화를 연속으로 수행하므로 지연이 길고 H/W가 복잡해지며 복호된 영상을 저장하기 위한 추가적인 버퍼를 사용해야 하므로 버퍼 사용량이 커지고 부호화/복호화 처리 시 발생하는 연산 오류로 인하여 화질 저하가 발생한다.

기존의 화상회의를 위한 다중 비디오 스트림들의 압축영역 혼합방식으로는 H.261기반의 혼합기와 JPEG기반의 혼합기가 있다^{[3],[4]}. [3]에서는 N-ISDN 환경에서 모든 단말의 전송률이 같은 경우 H.261로 부호화된 비디오 비트열을 혼합하는 방법에 대해 기술하고 있으며, [4]에서는 JPEG 방식으로 부호화된 비디오 비트열을 압축영역에서 혼합하는 방법에 대하여 기술하는데, JPEG 방식은 MC-DCT 방식으로 부호화된 비트열을 혼합하는 것에 비하여 혼합 처리의 유연성은 좋으나 부호화 효율이 떨어진다. 또한 H.261, JPEG을 이용한 방식들은 H.264/AVC^[5] 비디오 코덱을 사용하는 H.323^[6] 단말을 위한 비디오 혼합 방식으로는 적합하지 않다.

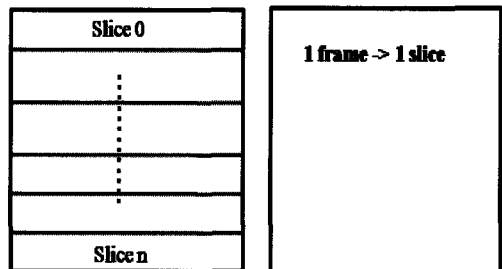
이와 같이 다중 비디오 스트림 처리는 계산량과 복잡도 및 버퍼 등의 크기 때문에 다수의 비디오를 동시에 다루는 영상통신 시스템의 성능에 가장 큰 변수로 작용한다. 특히 최근에 상용화된 WCDMA 화상전화, IP-phone, 영상회의 시스템 등에서 보다 많은 단말과 합성영상 포맷의 유연성에 대한 사용자의 요구를 지원하기 위해서는 저연산, 고품질, 화면 구성의 유연성을 갖는 영상혼합 방식이 필요하며, 무엇보다도 실시간성을 보장할 수 있는 압축 영역에서 혼합 방법이 필요하다. 따라서 본 논문에서는 다중 비디오 응용 서비스의 경우 수신단말 환경에 따라 적응적이고 유연성 있게 서비스를 제공하기 위해서 수신되는 다수의 비디오 스트림들의 비트열을 압축도메인에서 실시간으로 변환하여 고속으로 혼합할 수 있는 슬라이스 기반 H.264/AVC 다중 비디오 스트림들의 혼합 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 슬라이스 단위 H.264/AVC 압축 스트림의 구성에 대해 설명하고, 3장에서 제안된 다중 비디오 스트림들의 혼합 알고리즘을 기술한다. 그리고 4장에서 실험결과를 보이고 5장에서 결론을 맺는다.

II. H.264/AVC 스트림 분석

국제비디오압축 표준인 H.264/AVC는 다양한 유무선 네트워크 응용에 용이하도록 설계되어 기존의 비디오 코덱에 비해 높은 부호화 효율과 다양한 네트워크 환경에서 전송을 위한 적응성을 가지고 있다^[7].

기존의 비디오 부호화 표준 MPEG-2^[8] 등에서는 한 개의 프레임을 그림 1(a)와 같이 슬라이스 단위로 압축하는 것을 기본으로 하고 있다. 그러나 H.264/AVC는 그림 1(b)와 같이 한 개의 프레임을 한 개의 슬라이스로 취급하고 있어 혼합화면 구성의 유연성을 보장하기 어렵다. 따라서 본 논문에서는 혼



(a) MPEG-2 slice

(b) H.264 slice

그림 1. 슬라이스 단위 부호화

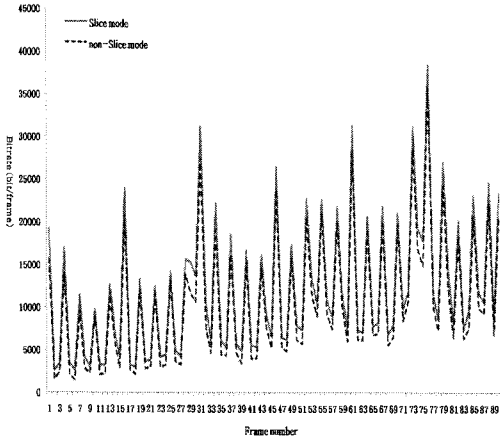


그림 2. H.264/AVC 슬라이스 모드 발생 비트량

합화면을 수신측 상황에 맞게 유연하게 구성할 수 있도록 비디오 프레임들을 그림 1(a)와 같이 H.264/AVC 슬라이스 모드를 이용하여 순차스캔순서(raster scan order)로 부호화한다.

그림 2는 176x144 해상도를 가지는 “crew” 영상 90 프레임을 JM15.1 부호기를 이용하여 슬라이스 모드를 사용한 경우와 사용하지 않은 경우의 발생 비트량을 비교한 것이다. 그림 2에서 수평축은 프레임 번호를 나타내고, 수직축은 프레임당 발생 비트량을 의미한다. 그림 2와 같이 프레임당 발생 비트량이 H.264/AVC에서 한 개의 비디오 프레임의 슬라이스 단위로 부호화한 경우가 한 개의 슬라이스로 취급하는 것보다 전체 발생 비트량이 약 18KB 정도 늘어난다. 이는 그림 3과 같이 슬라이스 모드로 부호화한 경우 프레임당 슬라이스의 개수만큼 3바이트의 개시부호(start_prefix_code)가 한 프레임을 구성하는 슬라이스 시작점에 부가되기 때문이다. 그러나 혼합화면을 유연하게 구성하고, 오류내성을 가지는 고품질의 비디오 서비스를 수행할 수 있도록 본 논문에서는 H.264/AVC 슬라이스 모드를 이용하여 혼합 대상이 되는 영상들을 부호화 한다.

H.264/AVC는 다양한 응용 분야에 적용될 수 있게 Video Coding Layer(VCL)과 Network Abstraction Layer(NAL)로 구성된다. VCL은 순수한 동영상 부호화에 관련된 내용으로 구성되고, NAL은 VCL로 부호화된 비디오 프레임의 정보를 NALU(NAL Unit)이라고 불리는 단위로 구성한다.

NALU는 비디오 시퀀스의 프로파일, 레벨, 해상도 등의 정보를 포함한 전체 부호화 시퀀스에서 변하지 않는 파라미터 집합인 SPS(Sequence Paramet

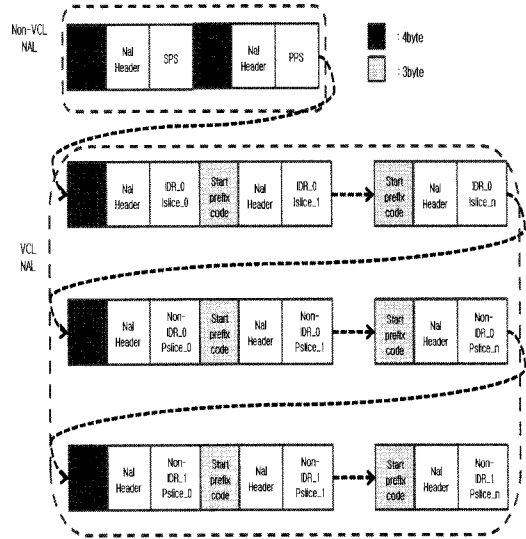


그림 3. 슬라이스 단위로 압축된 스트림 구성

er Set)과 엔트로피 코딩 모드, 슬라이스 그룹, 양자화 속성 등의 정보를 포함하는 프레임 단위에서 변하지 않는 PPS(Picture Parameter Set)으로 구성되며, 이들을 non-VCL NALU이라고 한다. 이와 반면, VCL NALU에서는 부호화된 비디오 데이터로 NALU를 구성하고, 매크로블록(Macroblock) 시작주소 등의 정보를 포함한 슬라이스 헤더(Slice header)와 매크로블록 타입 등의 정보를 포함한 슬라이스 데이터(Slice data)로 한 개의 비디오 프레임의 NALU로 구성한다.

그림 3은 프레임을 슬라이스 단위로 부호화한 경우 부호화 스트림의 선택 구성 예를 나타낸 것이다. 일반적인 H.264/AVC NALU의 경계는 4바이트 개시부호로 판단할 수 있지만, 그림 1(a)와 같이 슬라이스 단위로 부호화할 경우는 SPS와 PPS 및 한 프레임의 첫 번째 슬라이스의 NALU 앞에만 4바이트의 개시부호로 설정되고, 나머지 슬라이스로 부호화된 NALU들은 3바이트의 개시부호가 할당되기 때문에 슬라이스 단위로 비디오 스트림들을 혼합할 경우 3~4바이트의 개시부호를 고려하여 NALU의 경계를 판단해야 한다.

NALU에는 개시부호 이후에 1바이트의 NALU의 형태를 구분할 수 있는 정보를 나타내주는 헤더가 필수로 부가된다. NALU의 헤더에는 고정비트인 forbidden_zero_bit 플래그, 참조픽처 여부를 표시하는 nal_ref_idc 플래그, NALU의 종류를 표시하는 5비트의 nal_unit_type 플래그를 포함하고 있다. 5비트의 nal_unit_type 플래그에는 IDR(Instantaneous De

coding Refresh) 프레임, non-IDR 프레임, SPS, PPS 등의 32개의 정보를 표현 할 수 있다. 따라서 nal_unit_type 플래그 정보를 분석하여 혼합에 필요한 NALU를 분류할 수 있다.

III. 비트열 변환에 의한 비디오 혼합

그림 4는 제안된 다중 H.264/AVC 비디오 스트림들의 혼합 알고리즘을 도시한 블록도 이다. 도시한 블록도와 같이 다수의 압축 비디오 스트림 혼합 알고리즘은 입력되는 다수의 비디오 스트림들의 비트열을 저장하기 위한 입력버퍼와, 각 비디오 스트림들의 비트열을 분석하는 분석기, 최종 혼합화면 구성(Screen Layout)에 따라 각 비디오 스트림들을 배치하는 혼합기, 혼합된 비트열들이 표준 H.264/AVC 복호기에서 재생할 수 있는 형태로 구성 출력하는 출력버퍼로 구성된다.

제안된 비디오 혼합기에 입력되는 비디오 비트열의 수(n)는 최종 혼합화면 구성에 따른 부화면의 수와 같다. 그리고 혼합화면의 프레임 주기는 동일하다고 가정한다면 그림 4의 다수 비디오 스트림들을 혼합하는 알고리즘의 처리 과정은 다음과 같다.

입력버퍼는 비디오 스트림들의 비트열을 저장하고, 아직 혼합되지 않은 비트열들 중에서 버퍼 충만도(buffer fullness)가 가장 큰 버퍼의 비트열을 선택적으로 분석기로 보낸다.

분석기는 NALU로 입력되는 비트열에서 4바이트 혹은 3바이트로 구성된 개시부호로부터 NALU의 헤더를 찾고, NALU 헤더에 포함되어 있는 nal_unit_type 플래그 정보로부터 non-VCL NALU와 VCL NALU의 정보를 분류하여 혼합기에 보낸다. 이때 분석기는 혼합하고자 하는 다중 비디오 스트림들 중에서 최종 혼합화면 구성에 따라 기준 비디오 스트림의 비트열(즉, 화상회의 서비스에서는 회의 참가자중 현재 발언권을 가진 참여자의 비디오 스트

림)을 설정한다. 기준 비트열을 설정하는 이유는 혼합 비트열을 구성하는데 필수적으로 필요한 non-VCL 비트열을 가져오기 위해서이다.

혼합기는 기준 비트열로부터 가져온 non-VCL NALU의 SPS에서 영상 포맷을 위한 파라미터의 변환을 수행하고, PPS에서 프레임의 구성을 조절하기 위한 파라미터를 변경하여 출력버퍼로 보낸다. 혼합할 비디오 스트림이 n개의 스트림이 입력으로 들어와도 각 슬라이스에 대한 n개의 SPS가 필요하지 않으므로 기준 비트열의 단일 non-VCL NALU를 출력버퍼로 보낸다. 또한 기준 비트열로부터 개시부호 값을 출력버퍼로 보낸다. 그리고 혼합화면을 구성할 혼합 대상 스트림들을 화면 배치에 따라 슬라이스 헤더의 FMIS(first_mb_in_slice) 플래그 정보를 변경하고, 슬라이스 데이터는 입력 스트림의 데이터를 변형 하지 않고 출력버퍼로 보낸다. 슬라이스 데이터는 실제 영상에 대한 데이터를 가지고 있으므로 슬라이스 헤더의 일부만 변형되게 된다. 따라서 최종 혼합 비디오 스트림은 한 개의 SPS, PPS를 구성되고, n개의 혼합된 슬라이스 헤더와 슬라이스 데이터로 구성된다.

3.1 non-VCL NALU 변환

표 1은 기준 비디오 스트림의 non-VCL NALU 중에서 혼합 스트림을 구성하는데 변경해야할 플래그 정보를 보여 준다.

표 1에서 profile_idc는 기준 비트열의 값을 그대로 사용하고, level_idc는 최종 화면구성의 전체 해상도에 따라서 값을 변경한다.

최종 화면 구성에 따른 전체 해상도의 수평축 매크로블록의 수를 나타내는 pic_width_in_mbs_minus1 플래그, 수직축 매크로블록의 수를 나타내는 pic_height_in_map_units_minus1 플래그들을 식 1과 같이 변경하고 변환된 SPS 값을 출력버퍼로 보낸다.

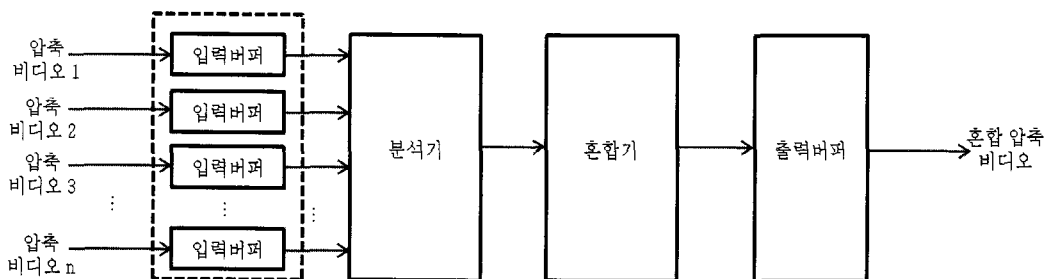


그림 4. 제안된 혼합 알고리즘

표 1. SPS 변환 플래그

변환 플래그	처리방식
profile_idc	입력 유지
level_idc	합성 영상 크기에 따라 변형
pic_width_in_mbs_minus1	합성영상 가로 크기에 따라 변형
pic_height_in_map_units_minus1	합성영상 세로 크기에 따라 변형

$$\begin{aligned} width_{mb} &= (Rwidth_{mb} + Swidth_{mb}) - 1 \\ height_{mb} &= (Rheight_{mb} + Sheight_{mb}) - 1 \end{aligned} \quad (1)$$

여기서 $width_{mb}$, $height_{mb}$ 는 변경할 각 수평, 수직 매크로블록의 수를 나타내고, $Rwidth_{mb}$, $Rheight_{mb}$ 는 기준 비트열의 매크로블록의 수이고, $Swidth_{mb}$, $Sheight_{mb}$ 는 혼합화면을 구성할 배치에 따른 부화면의 매크로블록의 수이다.

3.2 슬라이스 헤더 변환

H.264/AVC 신덱스에는 한 화면을 구성하는 각 슬라이스 마다 첫 매크로블록 어드레스만을 지시하는 플래그 값을 가지고 있다. 그러므로 다수의 비디오 스트림을 혼합할 때 화면 구성을 유연하게 정의할 수 있고, 기존에 매크로블록 어드레스 정보를 추출하기 위한 역 VLC와 매크로블록 어드레스 증가에 따른 차이 값을 VLC로 부호화하는 과정을 수행할 필요가 없다. 또한 재부호화 과정이 없으므로 다수의 비디오 스트림을 화질 저하 없이 고속으로 혼합할 수 있는 장점을 가지고 있다.

본 논문의 슬라이스 헤더 변환기는 혼합화면 구성에 따라 각 비디오 스트림의 시작 매크로블록 어드레스를 지시하는 FMIS 플래그 정보만을 변경한다.

그림 5는 176x144 해상도를 가지는 4개의 비디오 스트림들을 혼합할 때 FMIS를 변경하는 방법을 도시한 것이다. 그림 5에서 굵은 점선은 각 비디오 스트림들의 혼합된 경계를 나타낸 것이고, 화살표 실선은 혼합된 슬라이스 헤더의 FMIS 플래그 값을 변경하기 위한 매크로블록 어드레스 증가 방향을 나타낸 것이다.

본 논문에서 제안하는 혼합화면 구성에 따른 각 슬라이스의 FMIS 변경은 식 2를 이용하여 변경한다.

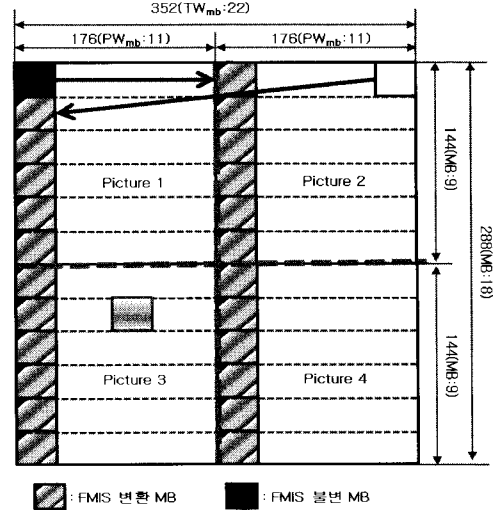


그림 5. 화면 구성에 따른 FMIS 변환

$$\begin{aligned} FMIS &= OFMIS(i) + SN(i)(TW_{mb} - PW_{mb}(i)) \\ &\quad + offset_{mb}(i) \end{aligned} \quad (2)$$

여기서 i 는 FMIS를 변경하려는 슬라이스가 속한 화면 번호로 $0 < i < n$ 의 범위이다. $OFMIS(i)$ 는 현재 변경하고자 하는 슬라이스의 원본 비트열의 FMIS값이 되고, $offset_{mb}(i)$ 는 i 번째 화면 앞에 있는 매크로블록의 수로 최종 혼합화면으로 구성될 각 화면의 좌측 상단 값으로 항상 고정된다. 즉, 그림 5에서는 각 화면의 $offset_{mb}(i)$ 값은 $offset_{mb}(0) = 0$, $offset_{mb}(1) = 11$, $offset_{mb}(2) = 198$, $offset_{mb}(3) = 209$ 가 된다. 그리고 TW_{mb} 는 전체 혼합화면의 수평축 매크로블록의 수를 의미하고, 그림 4에서 $TW_{mb} = 22$ 이다. $PW_{mb}(i)$ 는 i 번째 화면을 구성하는 한 개의 슬라이스에 포함된 수평축 매크로블록의 수를 의미한다. 그림 5에서 $PW_{mb}(0) = 11$, $PW_{mb}(1) = 11$, $PW_{mb}(2) = 11$, $PW_{mb}(3) = 11$ 이다. 또한 $SN(i)$ 는 i 번째 화면의 변경하고자 하는 슬라이스의 번호로 $0 < SN(i) < n$ 의 범위를 가지고 식 3으로 구할 수 있다. 즉, $SN(i)$ 는 한 개의 화면을 구성하는 슬라이스의 수이다.

$$SN(i) = \lfloor OFMIS(i) / PW_{mb}(i) \rfloor \quad (3)$$

따라서 4개의 비디오 스트림들을 혼합할 경우, FMIS의 변경 값은 그림 6과 같이 구할 수 있다. 그림 6에서 OFMIS는 입력 비트열의 실제 FMIS를

OFMS	Convert	FMIS	OFMS	Convert	FMIS
0	0+0(22-11)*0	0	0	0+0(22-11)*11	11
11	11+1(22-11)*0	22	11	11+1(22-11)*11	33
22	22+2(22-11)*0	44	22	22+2(22-11)*11	55
33	33+3(22-11)*0	66	33	33+3(22-11)*11	77
44	44+4(22-11)*0	88	44	44+4(22-11)*11	99
55	55+5(22-11)*0	110	55	55+5(22-11)*11	121
66	66+6(22-11)*0	132	66	66+6(22-11)*11	143
77	77+7(22-11)*0	154	77	77+7(22-11)*11	165
88	88+8(22-11)*0	176	88	88+8(22-11)*11	187
0	0+0(22-11)*198	198	0	0+0(22-11)*209	209
11	11+1(22-11)*198	220	11	11+1(22-11)*209	231
22	22+2(22-11)*198	242	22	22+2(22-11)*209	253
33	33+3(22-11)*198	264	33	33+3(22-11)*209	275
44	44+4(22-11)*198	286	44	44+4(22-11)*209	297
55	55+5(22-11)*198	308	55	55+5(22-11)*209	319
66	66+6(22-11)*198	330	66	66+6(22-11)*209	341
77	77+7(22-11)*198	352	77	77+7(22-11)*209	363
88	88+8(22-11)*198	374	88	88+8(22-11)*209	385

그림 6. 혼합된 슬라이스 FMIS 변환 예

나타내고, 가운데 열 'convert'는 식 2를 의미하며, 혼합화면 구성에 따라 변경된 FMIS를 나타낸다.

H.264/AVC 신택스 표준은 한 화면에서 한 개의 슬라이스를 구성하는 비트열이 슬라이스 헤더와 슬라이스 데이터를 포함하여 바이트 단위로 정렬되어 구성되어 있다. 따라서 본 논문에서는 혼합된 슬라이스 FMIS 어드레스가 값이 변화 되는 것에 따라 슬라이스 데이터의 최종 바이트를 그림 7과 같이 정렬한다. 그림 7에서 상단의 실선 영역의 a는 입력되는 한 개의 슬라이스 헤더와 슬라이스 데이터를 포함한 슬라이스 비트열의 크기이고, 하단점선 영역의 g는 최종 변환된 슬라이스 헤더와 슬라이스 데이터를 포함한 한 슬라이스를 구성하는 비트열의 크기를 나타낸 것이다.

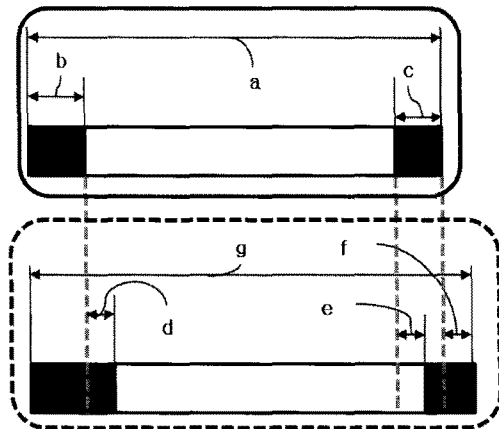


그림 7. 혼합된 비트스트림의 바이트 정렬

그림 7에서 b는 입력 슬라이스 헤더의 비트열 크기, c는 입력 비트열의 마지막 바이트 크기를 나타낸다. d는 슬라이스 헤더의 매크로블록 어드레스가 변경되어 상단 b의 크기가 늘어난 길이를 나타내고, d만큼 늘어난 비트에 의해 슬라이스 데이터의 마지막 바이트가 e에 해당하는 크기만큼 비트가 밀린다. 따라서 f만큼 밀려난 비트 수를 계산하여 바이트 정렬에 필요한 임의의 '0' 값을 삽입하여 바이트 정렬을 수행 한다. 그리고 슬라이스 단위로 처리된 비트 스트림들은 한 프레임을 구성할 수 있도록 선택적으로 출력버퍼로 보내고 최종 혼합화면을 구성하는 비트스트림을 출력한다.

IV. 실험결과

본 논문에서 표준 JM15.1 부호기를 이용하여 베이스라인 프로파일(baseline profile)을 만족하는 비디오 스트림들을 슬라이스 단위로 압축하여 실험영상으로 사용하였다.

제안된 알고리즘은 Intel Core 2 Duo 2.66, 2G RAM으로 구성된 시스템에서 구현 하였으며, 표 2와 같이 4가지의 경우에 대한 다중 비디오 스트림들의 혼합 실험을 수행하였다.

표 2에서 최종 혼합된 화면을 구성하는 화면 사이즈는 각 타입 순번으로 704x576, 880x576, 704x720, 704x720 화면으로 혼합을 수행하였다. 각 혼합 대상 비디오는 다음과 같은 형태를 가지고 있다.

Type 1은 352x288 해상도의 4개 비디오 혼합이고, Type 2는 1개의 704x576 비디오와 4개의 176x144 해상도를 갖는다. 그리고 Type 3은 1개의 704x576과 4개의 176x144 비디오, Type 4는 1개

표 2. 혼합 형태별 수행시간 비교

혼합 형태	수행시간(fps)
Type 1 (704x576)	150
Type 2 (880x576)	130
Type 3 (704x720)	130
Type 4 (704x576)	90



그림 8. 다중 스트림 혼합 시스템 사용자 환경

의 528x432 비디오와 7개의 176x144 비디오에 대한 혼합을 수행하였다.

그림 8은 제안된 혼합기의 사용자 환경을 보여주고 있다. 그림에서 좌측 상단은 사용자가 요구하는 4가지 형태로 최종 혼합 화면을 구성하는 것을 선택할 수 있고, 왼쪽 중간 부분은 사용자가 선택한 혼합모드에 따라 각각의 비디오 스트림들을 입력하

는 부분이다. 그리고 오른쪽은 현재 혼합되고 있는 결과를 표준 H.264/AVC 복호기로 재생하는 결과 화면을 나타낸다.

그림 9는 표 2의 혼합 구성에 따른 실험결과를 보여주고 있으며, 상단부터 차례로 type 1, type 2, type 4로 혼합한 결과이다. 그림 8에서 상단의 결과는 표준 H.264/AVC 복호기를 이용해서 재생한 혼합 스트림을 재생한 결과이고, 중앙과 하단은 표준 H.264/AVC를 지원하는 일반 플레이어로 재생한 결과이다.

실험결과 다수의 비디오 스트림들을 혼합하기 위하여 부호화된 비디오 스트림들을 복호화하고 혼합 화면 구성에 따라 배치 후 다시 부호화 하지 않으므로 화질열화가 없고, 부호화된 비트열을 변경함으로써 실시간으로 합성이 가능하다. 따라서 제안된 알고리즘은 실시간성을 요구하는 화상회의 시스템이나 IPTV 참여형 방송 서비스 시스템에 적용할 수 있다.

V. 결론

본 논문에서는 슬라이스기반 H.264/AVC로 부호화된 다수의 비디오 스트림들을 실시간으로 혼합하는 알고리즘을 제안했다. 제안한 알고리즘은 다중 비디오 스트림들을 혼합하기 위해 혼합화면 구성에 따라 슬라이스의 헤더 정보와 혼합된 비트열로 생성되는 파라미터 정보만을 변경함으로써 재부호기 연산이 필요하지 않으며, 해상도 변환을 수행하지 않을 경우는 비트열을 분석하는 연산만으로 다중 비디오 스트림들에 대한 절환/혼합 처리가 가능하다. 특히 혼합 처리가 H.264/AVC 비트열중 일반적인 헤더에 해당하는 영역을 변환하여 이루어지므로 H/W로 구현할 경우 복잡도가 크게 감소할 수 있다. 그러나 제안된 알고리즘은 이중 대역폭, 이중 단말 등의 서로 다른 복호화 능력을 가지고 있는 단말간에 서비스를 수행할 경우 해상도를 실시간으로 줄이거나, 대역폭에 맞게 발생 비트량을 줄이는 연구가 필요하다.

참고 문헌

- [1] Il kyun Park, Seung Hun Oh, Sim Kwon Yoon, Ho Young Song, "An Implementation of User-Participated Interactive IPTV Service System," in *Proc. IEEE ISCE*, April. 2008.

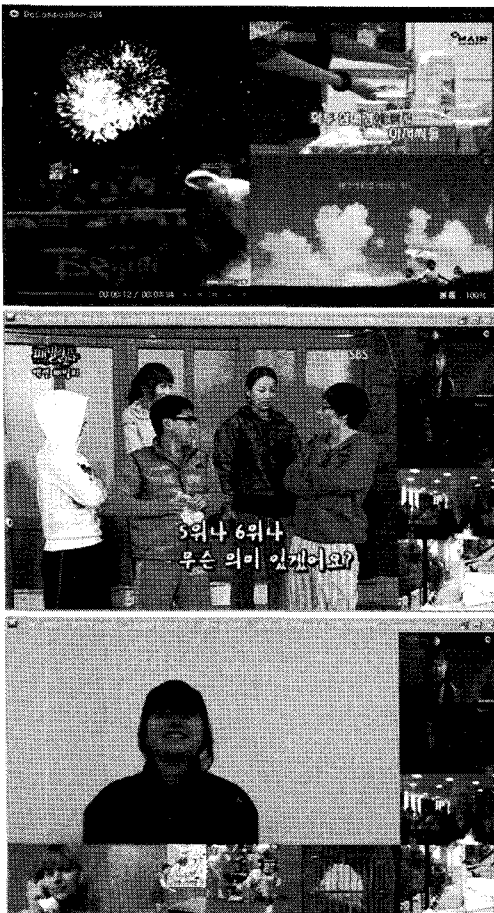
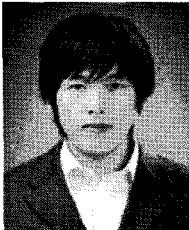


그림 9. 혼합 결과

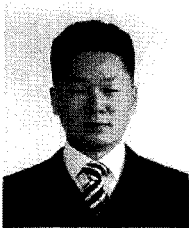
- [2] W.J. Clark and J. Boucher, "Multipoint Communication - the key to groupworking," *BT Technical Journal* Vol. 12, No. 3, July 1994.
- [3] S.M. Lei, T.C. Chen, and M.T. Sun, "Video Bridging Based on H.261 Standard," *IEEE Trans. on Circ. and Syst. for Video Technology*, Vol. 4, No. 4, Aug. 1994
- [4] R.D. Gaglianello and G.L. Cash, "Montage: Continuous Presence Teleconferencing Utilizing Compressed Domain Video Bridging," *ACM Multimedia* 1995.
- [5] ITU-T Draft Recommendation H.264, "Advanced video coding for generic audiovisual services," Nov. 2007.
- [6] ITU-T Draft Recommendation H.323, "Packet-based multimedia communications systems," June. 2006.
- [7] Thomas Wiegand, Gary J. Sullivan, Gisle Bjontegaard, Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. On Circ. and Syst. for Video Technology*, Vol. 13, No. 7, July 2003.
- [8] Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video, ISO/IEC 13818-2, 1995.

장 승 민 (Seung-Min Jang) 준회원



2008년 2월 전남대학교 전자컴퓨터공학부 공학사
 2008년 3월~현재 전남대학교 전자정보통신공학과 석사과정
 <관심분야> 영상통신시스템, 영상신호처리

유 흥 연 (Hong-Yeon Yu) 정회원



2001년 2월 동신대학교 전자전기공학과 공학사
 2004년 2월 전남대학교 전자정보통신공학과 공학석사
 2009년 8월 전남대학교 전자정보통신공학과 공학박사
 2008년 6월~현재 한국전자통신연구원 위촉연구원

<관심분야> 영상통신시스템, 객체 추출 및 합성, 의료영상처리, 영상 부호화

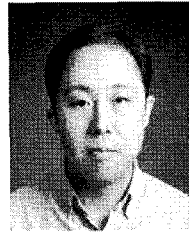
김 상 균 (Sang-Kyune Kim) 준회원



2006년 8월 조선대학교 정보통신공학과 공학사
 2009년 2월 전남대학교 전자컴퓨터공학과 공학석사
 2009년 3월~현재 전남대학교 전자컴퓨터공학과 박사과정
 2007년~현재 한국전자통신연구원 위촉연구원

<관심분야> MANET, P2P, IPTV

박 일 균 (Il-Kyun Park) 정회원



1997년 2월 숭실대학교 정보통신공학과 공학사
 1999년 2월 숭실대학교 정보통신공학과 공학석사
 2006년 8월 숭실대학교 전자공학과 공학박사
 2006년~현재 한국전자통신연구원 연구원

<관심분야> MANET, IMS, IPTV

홍 성 훈 (Sung-Hoon Hong) 정회원



1988년 2월 영남대학교 전자공학과 공학사
 1991년 2월 한국과학기술원 전기및전자공학과 공학석사
 1999년 2월 한국과학기술원 전기및전자공학과 공학 박사
 1991년~2000년 7월 LG전자 DTV 연구소 책임연구원

2000년 7월~현재 전남대학교 전자컴퓨터공학부 교수

<관심분야> 의료영상 처리, 영상 부호화, 멀티미디어 시스템,