

논문 2014-51-9-27

선택적부호화를 위한 비트율 변환기

(A Bit-rate Converter for Selective Coding)

이 중 배*

(Jongbae Lee[©])

요 약

본 논문에서는 H.264/AVC 표준 비트 스트림을 전송영상이나 멀티미디어 환경에 응용가능한 선택적 부호화를 위한 비트율 변환방안을 제시하였다. 전송채널은 응용분야에 따라 다양한 채널 용량을 갖고 있고, 따라서 컴퓨터에 저장된 비트스트림은 전송채널용량을 초과하지 않도록 적절히 비트율 변환이 이루어져야한다. 기 제공된 영상압축 비트스트림을 다른 비트율로 재변환하는 문제가 바로 비트율 변환이며, 본 논문에서는 새로운 비트율 변환방식을 제안하고자 한다. MPEG이나 H.264/AVC 압축 비트스트림을 이용하여 비트율을 변환하는 다양한 방식들이 여러 학자들에 의해 제시되었으며 실제로 많이 응용되고 있지만, 본 논문의 선택적 부호화 응용 분야에는 제대로 적용되지 않는다. 즉 프레임 내의 특정한 영역이 다른 부분보다 정보의 가치가 크고 따라서 이들 부분은 다른 부분에 비해 고화질로 재생되어야 하지만 기존의 방식으로는 이를 구현하기가 힘들다. 따라서 본 논문에서는 관심영역과 배경 간의 정보의 중요성을 고려하여 화질을 차별화하는 새로운 비트율 변환기를 제안하고자 한다.

Abstract

This paper presents a video bitrate converter for baseline profile in H.264/AVC standards to control a selective coding scheme for several applications such as tactical scenes or multimedia area. Transmission channels have various capacities according to the application area, and the bitstream stored in computer should be converted in order not to exceed the capacities of a transmission channel. So the problem is how to convert compressed bitstreams of a given bit-rate into compressed bitstreams of other bit-rates. Such a specific transcoding problem in this paper is referred to as bit-rate conversion. Several researches have been done on bit-rate conversion for the bitstreams compressed by MPEG or H.264/AVC. But the existing schemes are not suitable for selective coding scheme because it needs to recover interest regions better image quality than background. So we propose a new bit-rate converter which considers the importance between interest regions and background.

Keywords : bitrate converter, selective coding, interest region, importance, H.264/AVC

I. 서 론

지금까지 통신업계나 학계, 컴퓨터 업계에서 MPEG1, MPEG2, MPEG4 등이 영상압축을 위해 사용되었고, 최

근에는 멀티미디어 환경에서의 영상압축을 위해서 H.264/AVC^[1]이 널리 사용되고 있다. 따라서 다양한 영상응용 서비스를 위해 영상을 부호화하여 비트스트림 형태로 저장매체에 저장해두고, 이를 응용에 맞도록 활용하는 방식(off-line 전송방식)이 각광을 받고 있으며 본 논문에서도 H.264/AVC^[1]으로 저장된 데이터를 선택적부호화 기법^[2]에 알맞도록 하는 비트율 변환 기법을 제안하고자 한다.

실제로 초저속 영상 부호화기법(very low bit rate

* 정회원, 명지전문대학
(Myongji College)

© Corresponding Author(E-mail: a2970297@naver.com)
접수일자: 2014년8월06일, 수정일자: 2014년8월24일
수정완료: 2014년8월27일

coding algorithm)이 요구되는 전술영상과 같은 환경하에서는 특정한 부분이 다른 부분에 비해 매우 중요하게 취급될 수 있다. 즉 비행기, 탱크, 선박 등이 하늘, 수풀, 들판, 바다 등 배경보다 정보의 가치면에서 보다 중요하게 취급될 수 있는데, 이들 특정한 영역을 본 논문에서는 관심영역으로 분류한다. 또, 이러한 전술영상에서는 관심 영역이 영상 전체에서 작은 부분에 위치하지만 그 부분의 정보의 가치는 대단히 크므로, 영상 부호화시 이들 영역을 고화질로 복원할 수 있도록 해야 한다. H.264/AVC^[1] 비디오코덱은 현재 MPEG-2 못지않게 널리 이용되고 있다. 부호화 과정을 거친 후에, 비디오 비트스트림은 다양한 통신채널과 통신 장치에 전달된다. 즉 전송과정에서 각 채널이나 디바이스들의 이질성 때문에 비트율 변환이 필요하다.

비트율 변환을 위해서 데이터포맷을 바꾸거나, 비트율, 공간해상도, 프레임율 등을 조절하게 된다. 비디오 비트스트림을 다시 부호화하기 위해서는 비트율변환기는 복호화된 정보를 부분적으로 이용하게 된다.

Dziri^[3], Al-Muscati^[4] 등은 시간확장성(temporal scalability) 개념을 도입하여 H.264/AVC 비트스트림에 대해 비트율 변환하는 방안을 제시하였다. Sachdeva 등^[5]은 공간확장성(spatial scalability) 개념을 도입하여 단일 계층의 H.264/AVC에 멀티계층으로 분할하여 부호화하는 방안으로 제시하였다. Yeh 등^[6]은 Bayesian 원리와 Markov 체인을 이용하여 H.264/AVC와 SVC 사이의 비트변환 모드를 확률적으로 모델링하였다.

인간의 눈은 배경보다는 움직이는 영역에 대해 관심을 더 갖게 되며 R. Garrido-Contos^[7]은 이들 정보와 칼러 정보를 부분적으로 이용하여 배경과 ROI를 구분하였다. ROI 영역은 배경에 비해 좀 더 많은 비트를 할당하여 전체적으로 인간의 주관적 화질을 개선하는 방식을 제안하였다. 이들 이외에도 [8~9]에서 H.264 비트율 변환기법을 제시하였는데 이들 기존 방식들은 영상 내에 특별한 부분이 다른 부분에 비해 중요도가 큰 경우에 제대로 적용되지 않는 기법이다.

본 방식에서는 각 프레임을 관심영역과 배경영역으로 분할하고 H.264/AVC 비트스트림에 대한 비트율 변환시 영역별로 비트율을 차별화하여 부호화하는 방안을 제시한다.

II. 기존 비트율 변환기법들에 대한 고찰

기존의 비트율 변환기는 모두 텍스처 정보에 대한 비트율 변환을 고려했으므로 이들 관점에서 고찰하기로 한다. 가장 쉽게 생각할 수 있는 것으로 기부호화된 비트스트림을 화소 도메인에서 복호화를 행하고 양자화 스텝 사이즈를 변경하여 다시 부호화를 행하는 방식^[10~11]이다. 이는 복호화와 부호화 전과정을 다시 행해야 하므로 하드웨어 구현이 어렵고, 따라서 큰 시간적 지연을 요한다. 다음 DCT 도메인에서 부호화하는 방식^[10]으로 가변길이 부호화(variable length coding), 양자화(quantization) 과정 및 역양자화(dequantization) 과정이 요구된다. 이 방식은 화소도메인에서 부호화하는 방식에 비해서는 하드웨어 복잡도는 줄어들면서 시간적 지연도 줄어든다. 마지막으로 양자화된 DCT 도메인에서 비트율 변환을 행하는 방식^[11]으로 가변부호화 과정만이 필요하며, 하드웨어 복잡도도 작고 시간적 지연도 가장 작다. 그림 1은 전형적인 비트율 변환기의 일례를 나타낸 것으로 위치 (a)가 양자화된 DCT 도메인에서 비트율 변환이 이루어지는 기준점이 되며, 위치 (b), 위치 (c)는 각각 DCT 도메인과 양자화 DCT 도메인에서의 비트율 변환의 기준점이 된다. 결국 기존의 여러 방식들은 기본적으로 이 세 가지 형태중의 하나를 취한다.

Keesman 등^[10]의 경우에는 화소 도메인에서 비트율 변환기를 구현하였는데, 기존의 움직임 벡터와 기존 decision 값을 부호시에 그냥 이용함으로써 어느 정도 하드웨어 구현비용을 줄일 수 있도록 하였다. 다음에

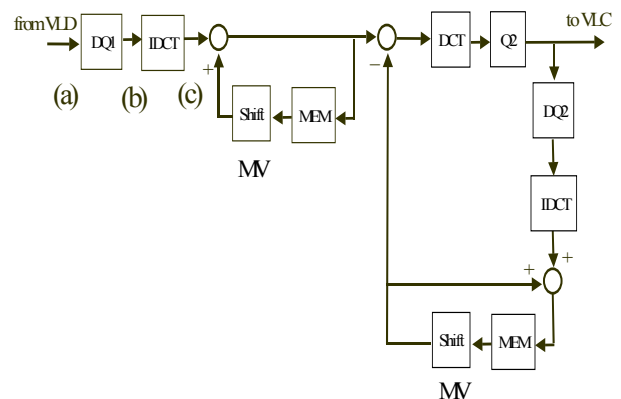


그림 1. 비트율 변환기의 블록도
Fig. 1. Block diagram of a bitrate converter.

DCT 도메인에서 재양자화를 행하는 방식의 일례로서 Nakajima^[12]는 양자화 스텝 사이즈를 키움으로서 전송 비트율을 줄이는 방식을 제안하였다. 이 경우는 움직임 벡터 예측이 필요하지 않고 DCT도 필요하지 않다. 즉 재양자화 시의 스텝사이즈 Q2를 원래의 비트스트림으로부터 복원한 신호 Q1에 비해서 크게 함으로써 비트율을 줄일 수 있다. 양자화 DCT 도메인에서 비트율 변환기를 구현하는 방식의 일례로서 H. sun 등^[13]은 AC 계수를 조절함으로써 비트율을 가변하는 방식을 제안하였다. 여기서는 움직임 예측, DCT 뿐만이 아니라 양자화도 불필요하게 되어 가장 구현이 간단하고 전송 지연 또한 작다. 이들 이외에도 이들 세 방식을 적절히 조절해서 사용하는 방식도 제안되고 있다. 또 서론에서 언급한 것처럼 [7]에서는 움직임정보와 칼리정보를 이용하여 배경과 ROI를 구분하고 ROI 영역에 대해 좀더 많은 비트를 할당하여 전체적으로 영상의 화질을 개선하는 방식을 제안하였다.

III. 제안한 방식

일반적으로 비트율 변환기는 그림 2와 같이 간략히 표현할 수 있다. 즉 고화질로 부호화되어 저장된 비트스트림 R1을 비트율 변환기에서 R2로 변환하여 전송하게 된다. 즉 비트율 변환기 구성에서 중요하게 고려되어야 할 사항은 망의 폭주를 방지할 수 있도록 전송 비트율을 적절히 조절하는 문제와 비용절감을 위한 하드웨어 단순화, 마지막으로 사용자의 요구에 맞도록 정보량을 자유롭게 조절할 수 있도록 하는 등의 문제이다.

기존의 방식들은 한 프레임에 물체가 하나 있는 경우에는 제안한 방식과 큰 차이없이 응용될 수 있다. 그

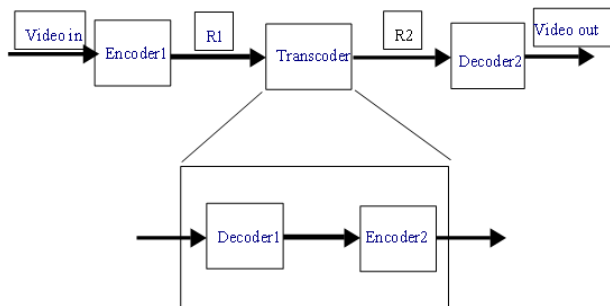


그림 2. 비트율 변환기의 구성
Fig. 2. Structure of bitrate converter.

런데 일반적으로 한 프레임에는 여러 물체가 존재하므로 경계정보를 함께 고려하는 것이 보다 타당하다. 또한 비트사용량이 제한되어 있는 상황에서 물체의 모형정보(Shape Information)를 고려한, 관심영역의 중요성을 고려한 새로운 비트율 변환기를 제안하기로 하였다. 본 방식은 텍스춰 정보와 모형정보를 모두 고려한 관심영역과 배경을 구분하는 차별화 제어를 행하며, 다양한 멀티미디어 데이터의 효율적 접근이 가능하도록 하였다. 그림 3은 제안한 방식의 간단한 블록도를 나타내고 있다. H.264/AVC로 부호화된 비트스트림은 메모리에 off-line 형태로 저장되어있게 된다. 이들 데이터는 모형정보, 움직임 정보, 텍스춰 정보로 구성되어 있는데 움직임 정보는 그 중요성에 비추어 전송비트를 감축하는 것이 실익이 없으므로 비트율 변환시에도 그대로 전송된다. 즉 움직임 정보를 감축하여 전송하면 각 VO 평면 사이의 텍스춰 정보가 증가하여(각 VOP 간 움직임 정보의 부정확성으로 인하여 움직임 보상 오차가 증가하게 됨) 급격한 화질 저하를 초래하므로 대부분의 경우에 전송시 비트 감축없이 그대로 보낸다. 본 방식에서도 움직임 정보는 디멀티플렉서에서 분리되어 비트 감축없이 그대로 전송되므로 그림 3에서는 생략하였다. 다음에 모형 비트스트림 정보는 모형 복호화부(Shape Decoding)에서 복호화 과정을 거쳐 원래의 모형이 복원된다. 복원된 모형정보는 모형개략화부(Shape Simplification)에서 모형단순화 과정을 거쳐 블록단위(블록크기는 응용분야에 따라 적절히 조절하게 됨)로 재부호화함으로써 모형 발생비트를 줄이게 된다. 텍스춰 정보는 본 응용에서는 화소 단위의 비트율 감축방식을 사용하였는데, 먼저 디멀티플렉서에서 분리

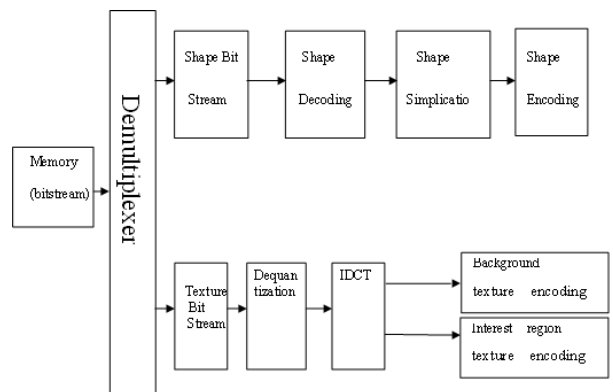


그림 3. 제안한 방식의 블록도
Fig. 3. A block diagram of proposed method.

된 텍스처 비트 스트림을 입력으로 하여 역양자화부(Dequantization)에서 역양자화를 행하고 IDCT부에서 IDCT를 행하여 텍스처를 복원한다. 복원된 텍스처 데이터는 관심영역과 배경으로 구분하여, 관심영역의 경우에는 비트할당을 충분히 하여 부호화하고 배경의 경우에는 나머지 여분의 비트를 이용하여 부호화한다.

관심영역과 배경을 분할하는 방법은 다양하게 제시될 수 있다. 예를 들어 데이터베이스에 관심영역의 feature를 저장해두고 패턴 정합을 통해 주어진 프레임으로부터 관심영역을 추출하는 방법, 반자동방식 등이 그 일례가 될 수 있다. 본 논문에서는 움직임 정보를 이용하여 관심영역을 제안하는 방법을 제안하고자한다. 실제로 많은 응용에서 움직이는 부분이 특별히 중요할 수 있고 특히 전술영상의 경우 적의 비행기나 선박, 탱크 등이 그 일례가 될 수 있다.

제 IV장의 관심영역 분할은 초기 배경영역 결정 및 관심영역 세분화 과정으로 크게 나뉜다.

IV. 관심영역 분할

1. 초기 배경영역 결정

먼저 과거영상 $U(x,y)$ 와 현재영상 $C(x,y)$ 에 대하여 full search BMA를 이용하여 각 블록의 속도벡터를 추출한다. 다음에 각 블록 속도벡터의 유사성(similarity)를 측정하고 이로부터 그룹을 정의한다. 즉 각 그룹(group)은 유사한 속도벡터를 갖는 블록들의 집합이다. 본 논문에서는 각 블록에 그룹들을 할당하기 위해서 agglomerative 클러스터링 방법^[14]에 기초한 계층적 클러스터링 기법을 사용하였다.

여기서 각 블록에 그룹 number를 할당하기 위하여 먼저 몇 가지 파라미터를 정의한다. 먼저 C_l 을 클러스터 l, P_l 을 C_l 의 부모 클러스터(parent cluster), th 를 속도벡터의 유사성을 판단하기 위한 문턱값, W_{ij} 를 블록 B_{ij} 의 속도벡터, U_i, Z_i 을 각각 같은 클러스터에 속하는 블록들의 평균속도벡터 및 블록의 개수로 놓자. 또, 주어진 영상에 대하여 다음과 같이 가정한다.

가정 1. 주어진 영상이 $M \times N$ 블록으로 구성되어 있다.

가정 2. $i = 0$ 일 경우 $U_{(i-1)j}$ 를 U_{ij} 로, $V_{(i-1)j}$ 를 V_{ij} 로 놓고, $j = 0$ 일 경우 $U_{i(j-1)}$ 를 U_{ij} 로, $V_{i(j-1)}$ 를 V_{ij} 로 놓는다.

가정 3. $B_{\{ij\}}$ 가 클러스터 C_l , $B_{(i-1)j}$ 이 클러스터 $B_{(i-1)j}$, $B_{i(j-1)}$ 이 클러스터 C_m 에 속한다.

위에서 정의한 파라미터와 가정에 따라 아래와 같이 그룹 알고리즘을 제안한다.

< 그룹 number 할당 >

Step 1. 각종 파라미터를 초기화 한다. 먼저 $I=0, j=0, k=0, m=0$ 으로 놓는다. 다음에 블록 b_{ij} 가 클러스터 C_l 에 속한다고 가정한다. 즉 $C_k = b_{ij}$. 또 P_l 과 C_l 을 같게 두고 U_m, U_n 의 초기값을 U_{ij} 로 놓는다.

Step 2. 먼저 i 를 증가시킨다. 만약 i 가 M 일 경우에는 j 를 증가시키고, $i=0$ 으로 놓는다. 만약 $j=N$ 이면 step 4로 간다.

Step 3. 아래의 네 가지 경우에 따라 현재 블록이 어느 클러스터에 속하는 지를 결정한다.

Case 1 : $\|U_{ij} - \overline{U}_i\| < th$ 이고, $\|U_{ij} - \overline{U}_m\| < th$ 면 $(Z_l \overline{U}_i + U_{ij}) / (Z_l + 1)$ 을 새로운 \overline{U}_i 로, P_l 을 P_m 으로 놓는다. 또, Z_l 을 1 증가시키며, 블록 B_{ij} 를 클러스터 C_l 에 포함시킨다.

Case 2 : $\|U_{ij} - \overline{U}_i\| < th$ 이고, $\|U_{ij} - \overline{U}_m\| > th$ 면 $(Z_l \overline{U}_i + U_{ij}) / (Z_l + 1)$ 을 새로운 \overline{U}_i 로 놓는다. 또, Z_l 을 1 증가시키며 블록 B_{ij} 를 클러스터 C_l 에 포함시킨다.

Case 3 : $\|U_{ij} - \overline{U}_i\| > th$ 이고, $\|U_{ij} - \overline{U}_m\| < th$ 면 $(Z_m \overline{U}_m + U_{ij}) / (Z_m + 1)$ 을 새로운 \overline{U}_m 로 놓는다. 또, Z_m 을 1 증가시키며 블록 B_{ij} 를 클러스터 C_m 에 포함시킨다.

Case 4 : $\|U_{ij} - \bar{U}_l\| > th$ 이고, $\|U_{ij} - \bar{U}_m\| > th$ 면 $(Z_m \bar{U}_m + U_{ij}) / (Z_m + 1)$ 을 새로운 \bar{U}_m 로 놓는다. k 를 1 증가시키고 B_{ij} 를 C_k 로 포함시킨다. 또 Z_k 를 1로 두고 P_k 에 k 를 할당하며, U_{ij} 를 새로운 \bar{U}_l 으로 놓는다.

Step 4. 부모 클러스터의 값이 같은 블록을 같은 클러스터로 놓고 클러스터 개수 및 속도벡터를 갱신한다. 즉 $k_1 < k_2$ 이고 $P_{k_1} = P_{k_2}, Z_{k_1} = a_1, Z_{k_2} = a_2$ 인 경우 새로운 클러스터 C_{k_3} 는 C_{k_1} 과 C_{k_2} 를 합친 값이 되며, $Z_{k_3} = Z_{k_1} + Z_{k_2}$ 이 된다.

Step 5. 클러스터 개수가 큰 순서대로 재배열(sorting)을 행하여 각 블록에 대한 최종 클러스터의 값을 결정한다.

이와 같이 각 블록에 대하여 클러스터 그룹이 결정되면, 초기 배경영역을 구하게 된다. 이는 가정에 의해서 목표물 영역이 배경영역에 비해 충분히 작으므로 가장 큰 그룹이 초기 배경영역이 된다.

2. Quadtree를 이용한 관심영역의 세분화

앞에서 언급한 것처럼 BMA 수행시 블록 크기를 고정시킬 경우 여러 가지 문제점이 있다. 따라서 quadtree를 이용하여 블록 크기를 가변적으로 조정하므로써 계산량이 증대되는 단점은 있지만 부가 정보량과 예측이득 등의 부호화 효율이 우수하기 때문에 여러 연구가 행해졌다^[15~20]. 일례로서 P.Stobach^[16]는 움직임 보상 영상에 대하여 quadtree 기법을 이용하여 영상분할을 행하였다. 그는 먼저 이동보상 오차영상에서 16×16 DCT 집중효율이 30~40 % 정도에서는 오히려 DCT하기 전보다 떨어짐을 보였다. 이에 따라 DCT 변환영상 대신에 원래의 이동보상영상(motion compensated image)에 대하여 오차블록의 균질성(homogeneity)을 조사하고, 균질성을 만족하는 경우 그 블록에 대한 quadtree 과정을 끝낸다. 그렇지 않으면 네 개의 서브블록으로 나눈 다음, 각 서브블록에 대하여 같은 과정을 반복한다. 이렇게 가변블록으로 분할되면 DCPM을 이용하여 영상 부호화를 행하였다.

본 논문에서는 영상의 오차신호와 속도정보를 동시

에 고려하여 초기 배경영역으로 정의되지 않는 부분을 quadtree 방식으로 분할하였다. 본 quadtree 방식은 top-down 방식으로, 주어진 블록이 하나의 leaf로 정의되거나 네 개의 서브블록(sub block)으로 나누어진다. 각 블록이 분할되면, 마찬가지로 과정으로 해당블록의 서브블록에 대해 보다 세밀한 영역 분할이 필요한 지를 결정하게 된다. 여기서 가장 효과적인 영상압축 알고리즘은 비트율(bit rate)과 왜곡(distortion)을 어떻게 효과적으로 줄이느냐 하는데 있다. 이는 R-D 평면에서의 최적점을 선택하는 문제이다. 즉 Lagrange multiplier ($\lambda \geq 0$)를 이용하여 각 영역 k 에 대하여 아래와 같은 목적함수 (objective function)를 최소화함으로써 R-D 평면에서의 최적점(optimal point)을 구할 수 있다.

$$\min\{d_k + \lambda b_k\} \quad (1)$$

여기서 b_k 는 각 영역에 할당된 비트의 수, d_k 는 이와 같이 비트를 할당하였을 경우 발생하는 왜곡, λ 는 Lagrange multiplier를 나타낸다.

이를 ‘principle of separate minimization’이라 부르는데, 그 이유는 Shoham과 Gersho^[19]가 각각의 분리된 영역에서의 목적함수를 최소화함으로써 전체영역의 최적화가 가능하다는 것을 보였기 때문이다. 일례로써 크기 $2^{n+1} \times 2^{n+1}$ 의 블록 X_{n+1} 의 최적화 quadtree 구조를 구하는 방법에 대하여 생각해보자. 먼저 $2^n \times 2^n$ 의 네 서브블록에 대한 최적 quadtree구조를 알고있다고 가정하고, 각 서브블록의 비트율과 왜곡을 각각 $b_n^*, d_n^*, (i = 1, \dots, 4)$ 로 놓자. 또 이 네개의 서브블록을 주어진 블록의 leaf 들로 나타낼 경우 비트율과 왜곡을 각각 b_{n+1}, d_{n+1} 로 놓자. 식 (2)로부터 $C_r \leq 0$ 를 만족하면 네 개의 서브블록은 하나의 leaf로 합쳐진다.

$$C_r = \Delta d - \lambda \Delta b$$

$$\Delta d = d_{n+1} - \sum_{i=1}^4 d_{n,i}^* \quad \text{and} \quad (2)$$

$$\Delta b = \sum_{i=1}^4 b_{n,i}^* - b_{n+1},$$

여기서 Δd 와 Δb 는 각각 네 개의 서브트리를 하나의 leaf node로 합침으로써 발생하는 왜곡의 증가량과 비트율 감소량을 나타낸다. Quadtree 구조가 그림 4와 같은 경우에 서브트리 X_{n+1} 을 나타내는데 필요한 비트 수는

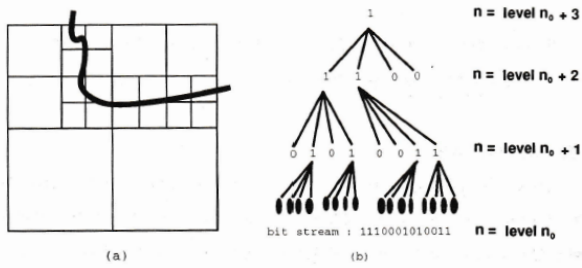


그림 4. Quadtree 구조
Fig. 4. Quadtree structure.

아래와 같이 표시된다.

$$b_{n+1}^* = \begin{pmatrix} '1' + b_{n+1} & \text{if } \Delta d \leq \lambda \Delta b \\ '1' + \sum_{i=1}^4 b_{n,i}^* & \text{otherwise} \end{pmatrix} \quad (3)$$

그리고

$$d_{n+1}^* = \begin{pmatrix} '1' + d_{n+1} & \text{if } \Delta d \leq \lambda \Delta b \\ '1' + \sum_{i=1}^4 d_{n,i}^* & \text{otherwise} \end{pmatrix} \quad (4)$$

여기서 '1'의 의미는 주어진 블럭을 네 개의 서브블럭으로 분할할 지의 여부를 나타내는 비트이다.

위에서 정의한 왜곡과 비트율을 기본으로 다음과 같은 quadtree 방식을 제안한다.

< Quadtree알고리즘을 이용한 영역분할 >

- Step 1.** 주어진 영상 $M \times N$ 블럭에 대하여 배경으로 정의되지 않는 영역에 대하여 level을 0으로 할당한다.
- Step 2.** 각 블럭을 서브블럭으로 나누고, 분할된 영역의 속도벡터를 측정한다.
그리고 level을 1 증가시킨다.
- Step 3.** 4 분할 영역에 대해서 $C_r \leq 0$ 을 만족하면 각 서브블럭을 합쳐서 하나의 leaf로 놓고 그 블럭에 대한 quadtree 과정을 종료한다. 그렇지 않을 경우 step 4로 간다.
- Step 4.** 분할된 영역의 그룹 number를 측정한다. 만약

분할된 영역의 그룹 number가 인접한 영역의 그룹 number로부터 구할 수 없으면 이 영역에 새로운 그룹 number를 할당하고 step 2로 가서 본 과정을 반복한다.

그림 5는 움직임 보상된 과거 영상과 현재 영상 간에 quadtree방식을 이용한 영상 분할 방식의 예이다. 영역 I,II,III은 배경을 나타내고 있고 IV, V는 관심 영역을 나타내고 있다.

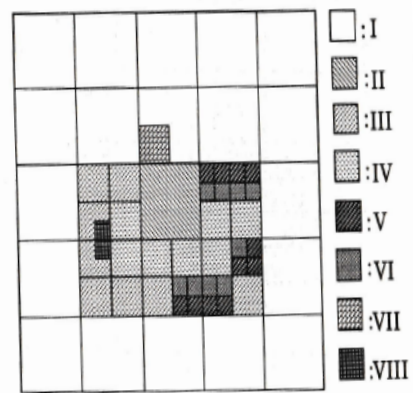


그림 5. Quadtree 영역분할의 예
Fig. 5. Example of quadtree segmentation.

V. 관심영역의 중요도를 고려한 비트율 차별화

전송 영상과 같이 전송 채널의 용량이 한정되어 있고, 영상 내의 특정부분이 다른 부분에 비해 정보의 가치가 훨씬 중요한 경우 기존의 영상압축이나 비트율 변환방식을 그대로 적용하는 경우, 제대로 된 성능을 구현할 수 없다. 배경과 관심영역 간의 정보의 가치를 고려하지 않고 영상을 압축·전송하는 경우에는 제한된 채널용량으로서 관심영역을 고화질(high quality)로 복원할 수 없다. 이를 해결하기 위해서는 배경과 관심영역 간의 양자화기 스텝크기(step size)를 다르게 하여야 한다. 먼저 E_{max} 를 관심영역에 대한 최대 허용오차로 놓고, 관심영역에 속하는 블럭 b_i 에 B_i 비트를 할당하였을 경우 $\theta_T(b_i(B_i))$ 를 관심영역에 속하는 블럭 b_i 의 양자화 오차신호라 가정하자. 이 경우 재생된 관심영역의 화질을 보장하기 위하여 식 (5)가 만족되도록 양자

화 비트가 할당되어야 한다.

$$\sum_{b_i \in R_t} \Theta_T(b_i(B_i)) \leq E_{\max} \quad (5)$$

여기서 R_t 는 목표물 영역에 속하는 블록의 수이다.

또 R 이 전체 시스템에서 허용하는 최대 비트율이라고 하면 식 (6)을 만족하도록 비트를 할당하여야 한다.

$$B_s \leq \frac{R}{\text{frame rate}} - \sum_{b_i \in R_t} B_i - P_a \quad (6)$$

여기서 B_s 는 배경신호를 전송하기 위하여 사용될 수 있는 비트율이고, P_a 는 전역이동 파라미터, 그룹 속도벡터, 위치정보 등을 포함하는 파라미터 정보이다.

본 시스템의 성능을 평가하는 척도로서 기존의 PSNR은 관심영역의 중요성을 고려하지 않으므로 제안한 알고리즘의 성능을 평가하는데 적합하다고 볼 수 없다. 따라서 본 논문에서는 아래와 같은, 정보의 중요성을 고려한 새로운 N-PSNR을 제안하고 이에 따라 본 시스템의 성능을 평가하고자 한다. 여기서 M 이 1이면 N-PSNR 은 기존의 PSNR과 동일하게 된다.

$$N - PSNR = \frac{(MI + B) \times 255^2}{10 \log(M \sum_{i \in I} \sigma_i^2 + \sum_{i \in B} \sigma_i^2)} \quad (7)$$

M 은 중요도, I 는 관심영역, B 는 배경영역, σ_i^2 : 원 화소와 부호화화소 값의 차의 평균제곱

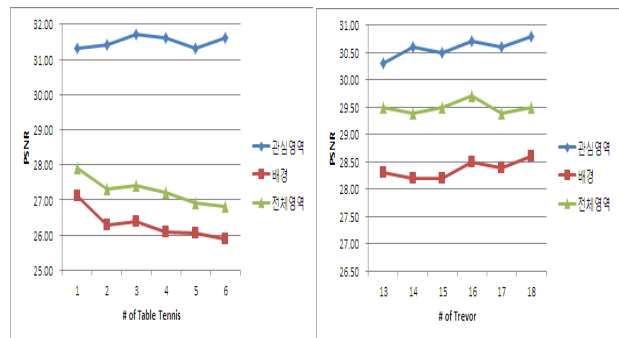
즉 관심영역 부분이 우리의 주된 관심분야이므로 배경 부분은 상대적으로 정보의 중요성이 떨어진다. 따라서 본 연구에서는 비트율 변환과정에 있어서 관심영역의 화질을 항상 고정도로 유지할 수 있도록 충분한 비트를 할당하고, 나머지 비트로 배경 부분에 대한 압축·전송을 행하였다.

본 알고리즘이 잘 적용되기 위해서는 관심영역이 배경영역에 비하여 충분히 작아야한다. 그 이유는 관심영역이 커짐에 따라 관심영역의 움직임 보상 오차 신호의 양이 많아지고, 부가적으로 각 관심영역에 대한 위치정보, tree level, 그룹 number등의 보조정보가 많이 요구되기 때문이다.

본 논문에서는 “Table Tennis” 연속영상, “Trevor” 연속영상을 이용하여 시뮬레이션을 행하였다. 그림 6은 “Table Tennis” 영상에서 영상분할에 의해 추출된 관



그림 6. 영상분할에 의해 추출된 관심영역(Table Tennis)
Fig. 6. Interest region extracted by image sementation.



(a) Table Tennis 연속영상 (b) Trevor 연속영상

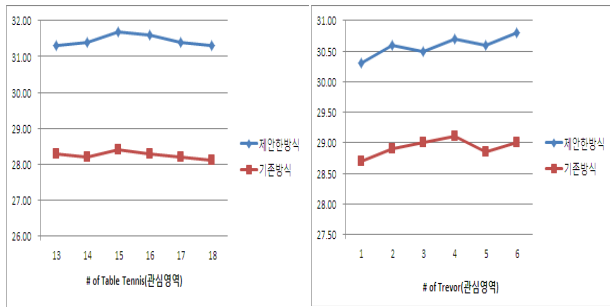
그림 7. 제안한 방식의 실험결과(PSNR)
Fig. 7. Simulation result of suggested scheme.

심영역을 표시하고 있다.

그림 7은 “Table Tennis” 연속영상을 48k bit/s, 10Hz로, “Trevor” 연속영상을 64k bit/s, 10Hz로 부호화했을 경우의 실험 결과를 제시하고 있다.

특히 목표물 영역으로 설정한 탁구공(ping-pong ball)의 경우, 재생된 영상이 원영상을 고화질로 복원하고 있다. 즉 비트율 변환시 관심영역에 상대적으로 보다 많은 비트를 할당하여 복원하므로써 관심영역의 화질이 배경에 비해 훨씬 우수함을 알 수 있다. 관심영역의 크기가 ‘Table tennis’ 연속영상에 비해 상대적으로 큰 경우의 일례로서, ‘Trevor’ 연속영상의 경우에는 배경영역과 관심영역의 화질 차이가 줄어들을 알 수 있다. 이는 관심영역의 크기가 큰 경우, 고압축을 구현해야하는 환경에서 프레임단위로 사용할 수 있는 비트량이 정해져 있어 이들 부분의 화질을 개선하는데 한계가 있기 때문이다.

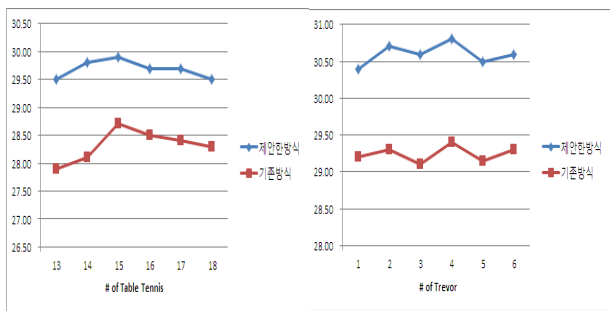
그림 8은 관심영역에 대해 비트율 변환시 관심영역을 차별화하여 부호화한 제안한 방식과 그렇지 않은 기존의 방식을 비교하고 있다. 제안한 방식이 기존의 방식에 비해 예상한 것처럼 관심영역의 성능이 좋게 나온 것을 확인할 수 있다. 다만 ‘Table tennis’ 연속영상에



(a) Table Tennis 연속영상 (b) Trevor 연속영상

그림 8. 기존방식과 제안한 방식의 관심영역 성능비교 (PSNR)

Fig. 8. Performance comparison for existing scheme and suggested scheme(PSNR).



(a) Table Tennis 연속영상 (b) Trevor 연속영상

그림 9. 기존방식과 제안한 방식의 성능비교(N-PSNR)

Fig. 9. Performance comparison for existing scheme and suggested scheme(N-PSNR).

비해 관심영역의 크기가 큰 ‘Trevor’ 연속영상은 제안한 기법과 기존방식 사이에 관심영역에 대한 화질 차이가 상대적으로 줄어들음을 알 수 있다. 그러나 이 경우에도 선택적 부호화를 행하지 않은 방식에 비해서 선택적 부호화 방식이 관심영역의 화질을 전반적으로 개선시키며, 연속영상 사이에서 관심영역의 화질을 보다 일정하게 유지해 줌을 알 수 있다.

그림 9는 “Table Tennis” 연속영상과 “Trevor” 연속영상에 대해 N-PSNR을 이용하여 영상 전체의 성능을 비교한 것이다. 중요도 M을 10으로 했을 경우의 일례로서 제안한 방식의 성능이 기존방식에 비해 개선됨을 알 수 있다. 만약 M=1로 했을 경우는 본 방식과 제안한 방식의 성능차이는 거의 없게 된다. 즉 본 방식은 관심영역과 배경 사이의 중요도의 차이가 클수록, 관심영역의 크기가 영상 내에서 차지하는 비중이 작을수록 큰 효과를 발휘함을 알 수 있다.

VI. 결 론

본 논문에서는 각 프레임의 관심영역과 배경영역으로 분할하고 H.264/AVC 비트스트림에 대한 비트율 변환시 영역별로 비트율을 차별화하여 부호화하는 방안을 제안하였다. 이를 위하여 본 논문에서는 먼저 프레임 클러스터링 기법과 quadtree 기법을 이용하여 배경과 관심영역으로 나누고, 비트율 변환시 관심영역에 대해 비트를 상대적으로 많이 할당하고 배경에 나머지 비트를 할당하였다.

실험 결과를 통해 관심영역을 기존의 방식에 비해 제대로 복원하였음을 보였고 기존의 PSNR이 아닌 중요도를 고려한 N-PSNR을 제시하여 이를 통해 기존 방식과 본 방식의 성능을 비교하였다. 예상대로 본 방식이 기존 방식에 비해 우수한 성능을 보임을 확인할 수 있었다. 만약 배경과 관심영역을 구별하지 않고 압축을 행했을 경우는 기존의 PSNR을 이용하여 영상 전체의 화질을 평가해보면 기존 방식이 제안한 알고리즘과 비슷하게 나온다. 그러나 우리가 중요하게 생각하는 관심영역에 대해서는 본 알고리즘이 훨씬 우수한 화질을 나타낸다. 그 이유는 관심영역에 대해서 본 알고리즘은 각종 파라미터 및 움직임 보상오차신호를 충분한 비트를 할당하여 보냄으로써 원하는 화질로 복원할 수 있도록 하기 때문이다.

REFERENCES

- [1] ITU-T, ISO/IEC JTC 1, “Advanced Video Coding for Generic Audiovisual Services”, *H.264/AVC and ISO/IEC 14496*, ITU 2009.
- [2] J. B. Lee and S. D. Kim, “A selective coding method based on global/local motion information”, *Signal Processing*, pp. 339-350, 1996.
- [3] A Dziri, A Diallo, M Kieffer, P Duhamel, “P-picture based H.264 AVC to H.264 SVC temporal transcoding”, *the proceedings of international wireless communications and mobile computing conference*, Greece, 2008.
- [4] H Al-Muscatti, F Labeau, “Temporal transcoding of H.264/AVC video to the scalable format”, *at the proceedings of 2nd international conference on image processing theory tools and applications*, France, 2010.

- [5] R Sachdeva, S Johar, E Piccinelli, "Adding SVC spatial scalability to existing H.264/AVC video", *the proceedings of 8th IEEE/ACIS international conference on computer and information science*, China, 2009.
- [6] C-H Yeh, W-Y Tseng, S-T Wu, "Mode decision acceleration for H.264/AVC to SVC temporal video transcoding", *EURASIP J. Adv. Signal Process*, 2012.
- [7] Rosario Garrido-Cantos¹, et al., "Low-complexity transcoding algorithm from H.264/AVC to SVC using data mining", *EURASIP Journal on Advances in Signal Processing 2013*.
- [8] u-Hyun Seo, et al., "An efficient ME scheme for H.264 to SVC transcoder", *Journal of the Institute of Electronics and Information Engineers ITC-CSCC*, July 2007, 665-666.
- [9] Y. T. Hong, et al., "MPEG-2/H.264 Transcoder Structure", the Paper collection of 2006's Fall Conference, the Institute of Electronics and Information Engineers, Nov. 2006, pp. 699-702.
- [10] G. Keesman, R. Hellinghuizen, Fokke Hoeksema, and Geert Heideman, "Transcoding of MPEG bitstreams", *Signal Processing: Image Communication* 8, pp. 481-500, 1996.
- [11] G. Morrison, "Video Transcoders with Low Delay", *IEICE. Trans. Communi.*, Vol. E80-B, No. 6, Jun. 1997, pp. 963-969.
- [12] Yasuyuki Nakajima, Hironao Hori, and Tamotsu Kanoh, "Rate conversion of MPEG coded video by requantization process", *Proc. ICIP*, 1995, pp. 408-411.
- [13] H. Sun, W. Kwok, and J. W. Zdepski, "Architectures for MPEG compressed bitstream scaling", *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 6, No. 2, April 1996, pp. 191-199.
- [14] R.O.Duda and P.E.Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, 1973.
- [15] D.H.Ballard and C.M. Brown, *Computer Vision*, Prentice Hall. 1982.
- [16] P. Strobach, "Tree-structured Scene Adaptive Coder", *IEEE Tr. on Comm.* 38. 1990.
- [17] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of image and video", *IEEE Int. Conf. on ASSP*, May 1991, pp. 2661 - 2664.
- [18] D. J. Vaisey and A. Gersho, "Variable block size image coding", *IEEE International Conf. on ASSP*, Apr. 1987, pp.25.1.1-4.
- [19] Y. Shoham and A. Gersho, "Efficient bit

allocation for an arbitrary set of quantizers," *IEEE Trans. on ASSP*, Vol 36, Sep. 1988, pp. 1445-1453.

- [20] J.Vaisey and A. Gersho, "Image compression with variable block size segmentation", *IEEE Trans. on Signal Processing*, Vol. 40, Aug. 1992, pp. 2040-2060.

저 자 소 개



이 종 배(정회원)

1985년 서울대학교 제어계측학과
학사 졸업.

1987년 카이스트 전기 및
전자공학과 석사졸업.

1996년 카이스트 전기 및
전자공학과 박사졸업.

1997년~현재 명지전문대학 컴퓨터정보과 교수
<주관심분야 : 통신, 정보보호, 신호처리>