

H.264/AVC에서 빠른 요약영상 추출을 위한 역추적 방법

정희원 권용광*

Backtracking Method for Fast Thumbnail Extraction on H.264/AVC

Yong-kwang Kwon* *Regular Members*

요약

H.264/AVC (이하 H/264)는 인트라 예측 부호화 방법을 포함한 새로운 기술을 채용하여 기존의 MPEG-2/4에 비해 2배의 압축 성능을 제공한다. 그러나 인트라 예측 부호화는 기존의 표준에서 사용된 많은 영상처리 기법들의 적용을 매우 어렵게 만들었다. 따라서 본 논문은 영상처리 기법의 하나인 요약영상 추출을 목적으로, 인트라 예측 모드의 역추적 과정을 통해 선택적 복호를 위한 복호 선택 정보를 추출하였다. 그리고 복호 선택 정보를 이용한 요약 영상 추출 알고리즘(B-PAD)을 구현하여 기존의 방법과 비교하였다. 그 결과 연산량과 추출 시간이 각각 평균 40%과 30% 감소하였다.

Key Words : H.264/AVC, Thumbnail, Back-tracking Algorithm, Intra prediction mode

ABSTRACT

H.264/AVC video compression standard provides a twice of compression rate than existing MPEG-2/4 standard by adopting a new technologies including a intra prediction method. But the intra prediction can make it difficult for many image processing techniques used in existing standard to be adopted. So in this paper, I extracts the decoding selection information to decoding selectively via back-tracking process of intra prediction modes for thumbnail extraction. Than I implements a Back-tracking PAD using a decoding selection information, and it is compared with previous methods. In result, B-PAD can save the operation time and running time up to 40% and 30 % respectively.

I. 서론

최근 컴퓨터 기술의 급속한 발전으로 인해 기존의 텍스트 위주의 사용자 환경에서 벗어나 이미지, 그래픽, 오디오 및 비디오 데이터 등을 제공하는 멀티미디어 사용자 환경으로 변화하고 있다. 요약영상이란, 동영상의 일부 또는 전체를 복호하여 추출된, 내용을 파악하는데 도움을 주는 대표 영상을 의미한다^[1]. 추출된 요약영상은 디지털 방송 또는 컴퓨터 및 인터넷과 결합되어 다양한 사용자 인터페이스를 제공할 수 있다. 요약영상 추출 방법은 동영상 압축 비트열을 완전 복호하여 하향 표본화(down-sampling)하는

방법^[2]과, 압축 비트열의 특성을 이용하거나 부분 복호화를 통해 요약영상을 추출하는 방법^{[3]-[5]}이 있다.

MPEG-x (Motion Picture Experts Group)는 DVD나 DTB 등에 적용되었고, H.26x는 화상 전화 등에서 오랫동안 연구되었기 때문에 압축 구조를 이용한 다양한 요약영상 추출 알고리즘들이 개발되었다. 최근까지 제안된 (동)영상 압축 표준들은 DCT 변환 또는 웨이블릿 변환을 이용하여 공간적 중복성을 제거하는데, 웨이블릿 변환은 기본적으로 공간적 스케일리빌리티(scala-bility)를 제공하는데 비해, DCT 변환은 지원하지 않는다. 따라서 DCT 공간에서 요약영상을 추출하기 위해서는, DC 계수를 추출하거

* 동국대학교 전자공학과 디지털영상처리연구실 (fifthave@donguk.ac.kr)

논문번호 : 08060-0911, 접수일자 : 2008년 6월 8일

나, DCT 수식의 변형^{[3][4]}, 룩업 테이블을 이용, 또는 움직임의 추정을 통해 고속으로 정확한 요약 영상 추출 방법들이 사용되고 있다. 최근 정수배 하향 표본화가 가능한 H.264 기반의 SVC (Scalable Video Coding)가 표준화되었으나 구조적으로 매우 복잡하다.

가장 최근 표준화된 H.264^{[6]-[8]}는 많은 여러 신기술과 함께 특히 I 프레임에서 픽셀 기반의 새로운 인트라 예측 부호화(intra prediction) 기술이 사용되었다. 새로운 인트라 부호화 기술은 픽셀 영역에서 주변 블록을 참조하여 공간적 중복성을 제거하는 방법으로 원영상의 픽셀정보가 대부분 예측모드에 의해 생성된 예측블록에 포함되며, 변환 계수는 단지 원영상에서 예측 블록의 값을 뺀 오차이다. 따라서 H.264에서 DC 계수는 영상 블록의 평균이 아니라 $n \times n$ 잔여 데이터들의 평균이므로 DCT 기반의 요약영상 추출 알고리즘들의 적용이 불가능하다. 최근 H.264의 예측 정보를 DCT 공간으로 옮기는 방법으로 LUT로 빠른 요약영상 추출 방법이 제안되었으나^[9], H.264 구조 상 영상 크기와 에러 전파로 인한 급격한 화질 열화의 가능성이 비례하기 때문에 픽셀 공간에서 실수 연산으로 인한 참조 오차를 명확하게 고려하지 않는다면 픽셀 오차를 보상하는 것이 매우 어렵다. 그리고 [10]과 [11]에서 예측방향과 DC 계수만을 이용한 방법도 제안되었으나 역시 참조 픽셀의 부정확성으로 발생한 에러가 픽셀 공간에서 계속적으로 전파됨을 확인할 수 있다. 반면에, 이전의 연구^[12]에서는 선택 영역 복호 알고리즘(PAD; Partition Area Decoding algorithm)을 제안하여 에러 전파가 없는 요약영상을 추출할 수 있었다. 따라서, 본 연구에서는 인트라 부호화로 인한 픽셀 영역에서의 압축을 고려하여, 인트라 정보의 역추적을 통해 복호 선택 정보를 추출하였고, 이를 PDA에 적용함으로써 추가적인 연산량 감소를 통한 성능개선을 보인다.

논문의 구성은 다음과 같다. II장은 H.264의 인트라 예측을 설명하고, III에서 역추적 방법을 제안한다. IV장에서 요약영상 추출 시스템을 설명하고, V장에 제안 방법의 성능을 인간 시각적인 측면, PSNR 그리고 연산량 측면에서 비교/평가한다. 마지막으로 VI장에서 본 논문에 대한 결론을 맺는다.

II. H.264/AVC의 인트라 예측 부호화

인트라 예측 부호화는 프레임 내에서 그림 1, 2,

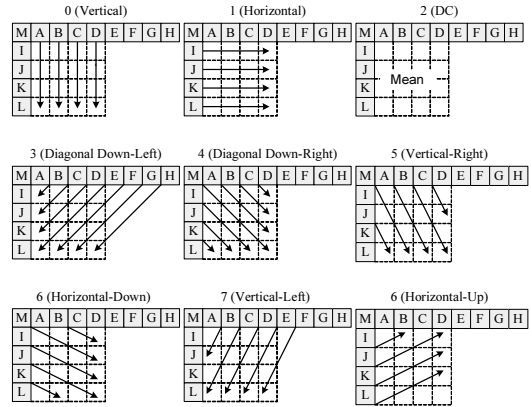


그림 1. 휘도성분 인트라 4x4 예측 모드

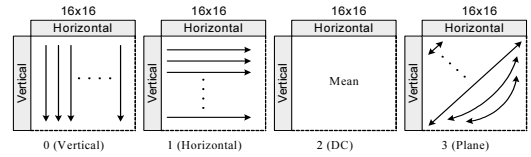


그림 2. 휘도성분 인트라 16x16 예측 모드

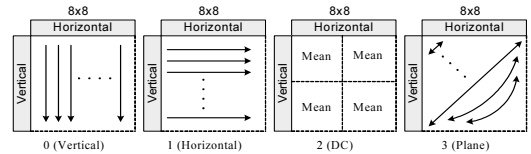


그림 3. 색차성분 인트라 8x8 예측 모드

3의 화살표 방향으로 이웃 블록의 픽셀을 참조하여 사전에 정의된 방법에 따라서 참조블록의 픽셀값들을 계산하고, 계산된 참조블록들과 원영상의 블록과의 SAE(Sum of Absolute Error)를 각각 계산하여 화소 상관도가 가장 높은 블록의 모드의 번호를 압축하는 방법이다^[7]. 그리고 선택된 화소 상관도가 가장 높은 예측 블록과 원영상과의 오차값들을 4x4 블록 단위로 DCT를 수행한다. 따라서 인코더는 선택된 예측 모드 정보 (1 또는 3 bits)와 오차 값에 대한 DCT 계수값에 대해서만 엔트로피 부호화를 수행하면 된다.

III. 역추적 알고리즘

이전 연구^[12]에서 제안된 PAD 알고리즘에서는 영상의 에러 전파 없이 요약영상을 생성하기 위해 4x4 블록의 오른쪽과 아래쪽에 위치한 7개의 픽셀을 모두 복호하고 이를 선택영역이라 하였다. 그런데 실

제 복호화 과정을 따라가 보면 13개의 참조픽셀(그림 1의 A~M) 중 일부만이 현재 블록의 예측블록 생성에 사용된다. 즉 참조되는 픽셀들만을 선별할 수 있다면 추가적인 연산량 감소에도 여러 전과가 일어나지 않을 것이다.

2.1 Back-Tracking Algorithm

인트라 부호화에서는 그림 1~3과 같이 주변 블록에서 경계에 위치한 픽셀들을 참조 픽셀로 활용하여 예측 블록이 생성된다. 이때 예측 모드 m 에 따라 이용되는 참조 픽셀들이 다르며, 또한 현재 블록에서 복호되는 픽셀 $\bar{x}_{i,j}$ 의 위치 (i,j) 에 따라 참조되는 픽셀들이 다르다. 따라서 제안하는 알고리즘은 각 픽셀에 대하여 참조픽셀의 위치를 파악하여 복호 여부에 대한 선택 정보로 저장하고 영상의 복호 과정에서 선택 정보를 바탕으로 복호화 과정을 수행한다.

이러한 접근에서, H.264의 압축 비트열로부터 예측 값을 추출하여 영역의 마지막에서 역방향으로 이동하면서 복호 시 사용될 복호 선택 정보를 추출하는 역추적 복호 알고리즘을 제안하고 그 기본 개념을 식 (1)과 같이 정의한다. 현재 블록의 선택 정보의 집합을 B_{curr} 라 하고 (i,j) 위치 $(i,j=0\sim3)$ 의 픽셀이 선택 영역에 포함된다면 복호 선택 정보는 다음과 같이 표시된다.

$$B_{curr}(i,j) = enable \quad (1)$$

그리고, 식 (2)에 의해 선택 정보 $B_{curr}(i,j)$ 가 예측 모드 m 에 따라 참조픽셀을 역추적하여 $(curr-k)$ 번째 블록의 (x,y) 픽셀에 대한 복호 선택 정보를 수정한다. 이때 l 은 가로 축으로 4×4 블록 개수이고, $k \in \{1, l+1, l, l-1\}$ 이다.

$$B_{curr-k}(x,y) = B_{curr}(i,j)|_m \quad (2)$$

그림 4에서 픽셀 0은 요약영상을 위해 복호되어

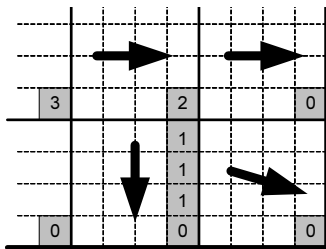


그림 4. 역추적의 예

야 하는 픽셀이다. 역추적은 영상의 오른쪽 하단에서 시작되며, 먼저 예측모드가 Horizontal Down (6)이므로 픽셀 0의 복호를 위해 픽셀 1들이 참조되어야 한다. 그리고 픽셀 1을 위해서는 예측모드를 고려할 때 픽셀 2가 필요하다. 같은 방법으로 영상 전체에 대해 역추적을 실시하여 복호될 픽셀 위치 정보를 추출한다. 그리고 복호과정에서는 역추적된 위치 정보를 바탕으로 선택 영역이 되는 0~3번 픽셀들만을 복호한다.

2.2 역추적 알고리즘의 구현

설명된 역추적 알고리즘은 영상의 전체 영역에서 수행하는 것이 효과를 극대화 할 수 있다. 그러나 제안된 알고리즘을 실제 구현함에 있어 다음의 두 가지 문제를 고려할 필요가 있다. 첫째, H.264에서 인트라 예측 모드 정보는 매크로블록 단위로 부호화되어 매크로블록의 헤더에 저장된다. 그러므로 메모리 사용과 복호 시간을 최소화하기 위해 매크로블록 단위로 역추적 알고리즘을 적용하는 것이 효율적이다. 둘째, 각각의 픽셀에 대해 역추적하는 것이 이상적이나 구현에서 너무 많은 경우의 수가 발생하므로 알고리즘의 단순화를 위해 선택 영역의 픽셀들을 몇 개의 조합으로 분할하는 것이 효과적이다.

결과적으로, 제안되는 역추적 알고리즘은 그림 5(a)와 같이 매크로블록 단위로 적용되며, 화살표 방향으로 0번에서 15번 블록으로 이동하면서 위치 정보를 수집한다. 또한 구현되는 알고리즘을 단순화하기 위해 선택 영역의 픽셀들을 그림 5(b)와 같이 분류하여 각각 Right, Bottom, Corner로 지정한다. 그리고 현재 블록의 복호 선택 정보 B_{curr} 에 따라 그림 5(c)~(g) 중에서 한 형태로 복호된다. 단, Right 또는 Bottom이 enable이라면 Corner가 disable로 지정되고 반대로 Corner가 enable이면 Right와 Bottom이 disable이다.

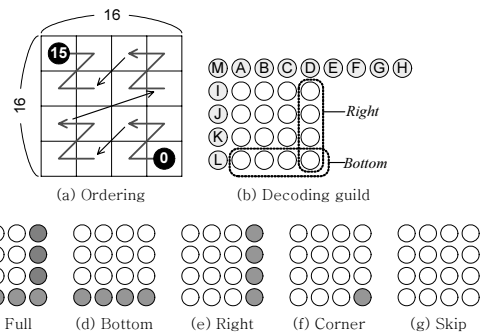


그림 5. 역추적 알고리즘의 복호 조합

복호 선택 정보를 획득하는 과정은 다음과 같다. 먼저 식 (3)과 같이 초기화 하여 매크로블록의 오른쪽과 아래쪽 픽셀들은 선택 영역에 우선 포함시켜 이가 이웃 블록으로 전파되는 것을 차단한다($\alpha = 0 \sim 15$).

$$B_{\alpha}(\text{Right}) = \begin{cases} \text{enable} & , \alpha \% 4 = 3 \\ \text{disable} & , \text{otherwise} \end{cases} \quad (3.a)$$

$$B_{\alpha}(\text{Bottom}) = \begin{cases} \text{enable} & , \alpha \geq 12 \\ \text{disable} & , \text{otherwise} \end{cases} \quad (3.b)$$

$$B_{\alpha}(\text{Corner}) = \text{disable} \quad , \text{for all block} \quad (3.c)$$

역추적 알고리즘은 매크로블록 단위로 그림 5(a)의 순서에 따라 식 (4)을 적용하여 이웃 블록의 복호 선택 정보를 수정한다($k = \{1, 3, 4, 5\}$).

$$B_{curr-k} = T(B_{curr}, m) \quad (4)$$

where, $B = [\text{Right}, \text{Bottom}, \text{Corner}]^T$

식 (4)에서 함수 $T(\cdot)$ 는 현재 블록 선택 정보 별로 확인하여, *enable*이면 예측 모드 m 에 따라 참조 픽셀들의 위치를 그림 1에서 확인하여 B_{curr-k} 의 선택변수를 *enable*로 지정한다.

제안된 방법은 선택 판단만을 하는 매우 단순한 구조이므로 구현이 매우 간단하며, 다음 사항에 주의한다. 일반적인 디코더 설계에서, 5번째 또는 13번째 블록의 예측 모드가 3 또는 7 경우(그림 1 참조)에는 그림 5의 참조픽셀 E~H가 존재하지 않는다. 이는 인코더에서 화소간의 상관성을 최대화 하는 지그재그 스캔과 같은 개념에서, 픽셀 공간에서의 블록의 부호화 순서가 블록 간 상관성을 최대화 하기 위해 그림 5(a)의 화살표와 반대 방향(15→0)으로 수행되므로 발생된다. 이 경우 복호 과정에서 그림 5의 E~F 대신 D를 사용하게 되며, 따라서 $B_{curr-l+1}(\text{Bottom})$ 대신 $B_{curr-l}(\text{Corner})$ 의 선택을 *enable* 한다.

IV. B-PAD 알고리즘의 구현

구현된 역추적 알고리즘은 매크로블록 헤더에 저장된 인트라 예측 정보를 사용하므로 헤더 정보 파싱과 요약영상 추출 모듈 중간에 삽입되어 별도의 독립 모듈로써 동작한다. 구현되는 B-PAD는 JM 12.2를 기반으로 PAD [12]에 선택 영역을 최소화 하기 위한 역추적 모듈이 삽입되었다. 그림 6은 구현된 요약영상 추출을 위한 복호기의 블록도이며, 그림에서 압축열에서 추출된 예측모드는 역추적 모

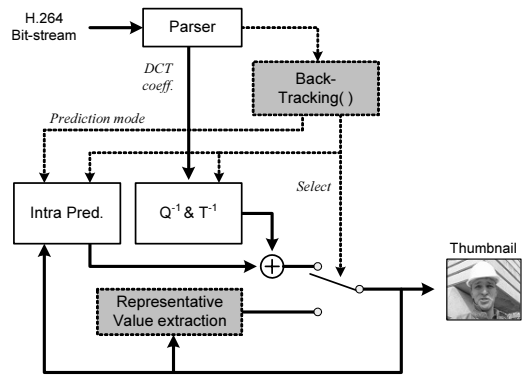


그림 6. 제안된 요약영상 추출 시스템

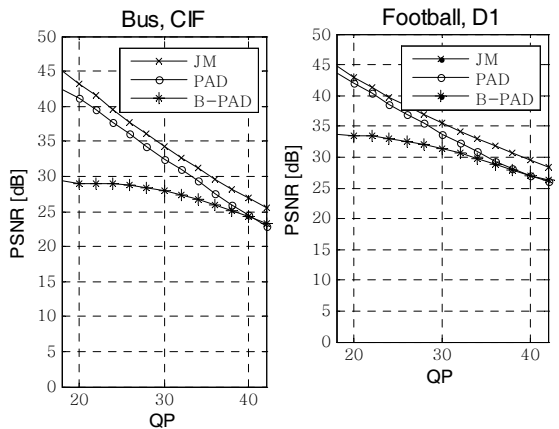


그림 7. PSNR 비교

들의 입력이 되어 역추적이 수행된다. 그리고 역추적된 정보(=복호 선택 정보)와 예측 모드는 인트라 부호화 모듈과 변환 및 양자화 모듈로 입력되어 변환 계수와 함께 선택적인 복호화 과정을 수행한다.

V. 실험

실험영상은 베이스라인 프로파일에서 CIF 영상은 Level 2.0, D1 영상은 Level 4.0으로 압축되었다.

먼저 제안된 B-PAD 알고리즘의 산술적 평가를 위해 PSNR 비교를 그림 7에 도식하였다. B-PAD는 PAD나 JM의 결과에 비해 높은 QP에서는 거의 유사한 결과를 보이지만 낮은 QP에서는 3~6dB의 차이를 보인다. 그 이유는 B-PAD를 구현하는 과정에서 더 많은 연산량 감소를 위해 역추적 과정에서 그림 5(g)가 선택된 경우 복호 과정을 생략하고 블록의 대표값을 이웃 대표값에 의해 단순 추정하는 방법을 사용함으로써 발생하는 오차 때문이다. 즉,

표 1. QP30에서의 PSNR 비교 [dB]

PSNR		JM	PAD	B-PAD
CIF	Bus	34.3	32.9/-1.4	28.1/-6.2
	Crew	36.6	35.3/-1.3	34.1/-2.5
	Foreman	35.8	34.7/-1.1	32.4/-3.4
D1	Football	35.5	34.2/-1.3	32.2/-3.3
	Garden	34.7	34.3/-0.4	28.4/-6.3

표 2. QP30에서 프레임당 평균 연산량 [회]

Adder		JM	PAD	B-PAD
CIF	Bus	992,627	636,250	451,980
	Crew	1,036,433	643,167	464,419
	Foreman	1,010,866	632,161	449,281
D1	Football	3,059,719	2,193,899	1,536,430
	Garden	3,695,176	2,308,043	1,608,696
Shift		JM	PAD	B-PAD
CIF	Bus	599,921	374,646	273,380
	Crew	591,862	350,911	249,492
	Foreman	583,090	352,851	253,218
D1	Football	2,030,249	1,221,070	868,908
	Garden	2,185,759	1,349,192	967,871

H.264는 낮은 QP에서 영상의 세밀도가 높기 때문에 추정된 값이 실제 값과 차이를 보일 수 있다. 그럼에도 매크로블록 단위로 에러 전파가 차단되며, 그림 5(g)가 픽셀 단위로 나타나므로 비교 영상과 시각적인 차이가 거의 없다.

상업적 목적으로 많이 적용되는 QP30에서의 PSNR을 표 1에 비교하였다. JM 모델을 기준으로 B-PAD가 평균 3dB 낮은 결과를 보이며, Bus나 Garden과 같이 영상의 복잡도가 높은 영상에서는 약간 낮은 결과를 낸다.

다음으로 JM과 PAD, B-PAD 모델의 프레임 당 평균 연산량을 비교하여 그림 8에 도식하였다. 그림에서 수치화된 연산 횟수의 범위는 역양자화 및 역변환과 예측모드 계산, 그리고 역추적 모듈 및 대표값 추출에 따른 연산량을 모두 포함한다. 그래프에서 영상에 따른 차이는 있지만 model 별로 연산 횟

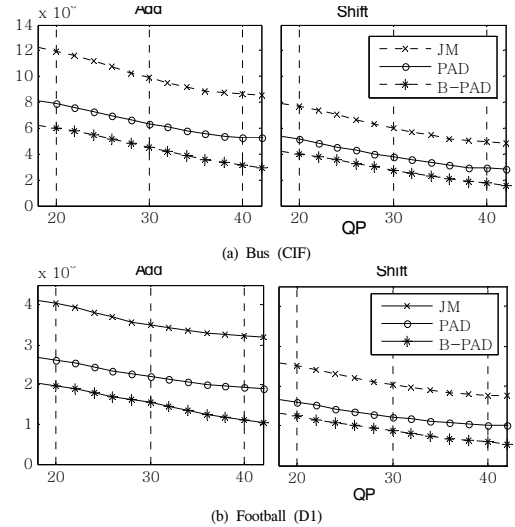


그림 8. 프레임 당 평균 연산량 비교

수의 순위는 일정한 모습을 보이는데, 덧셈과 쉬프트 연산에서 JM과 비교하여 B-PAD의 경우 평균 50%가 감소되었다. 표 2는 상업적 용도로 많이 사용되는 QP30에서의 프레임당 평균 연산량을 비교한다.

요약영상의 응용에서는 무엇보다 인간 시각적으로 영상의 내용이나 특징을 얼마나 분명하게 판단할 수 있는가가 관건이다. 따라서 그림 9에서는 Football 영상(720x480)에서 B-PAD에 의해 추출된 요약영상을 기존의 연구와 비교하였다.

결과적으로 B-PAD는 PSNR 측면에서는 평균 35dB의 결과를 보이고 있다. 그러나 QP가 높을수록 JM 모델과 유사한 성능을 보이고 있는데 평균적으로 QP25이상에서 5dB이하의 차이를 보이고 35dB 이상에서는 JM과 유사한 PSNR 결과를 보이고 있다. 반면에 연산량 비교에서는 JM과 비교하여 덧셈과 쉬프트 연산에서 평균 40%의 연산량만을 필요로 한다. 또한 시각적인 측면에서는 JM 모델의 결과와 차이가 없다. 단, 역추적을 위한 연산과정이 추가적으로 요구되고 있으나 역추적 알고리즘은 모

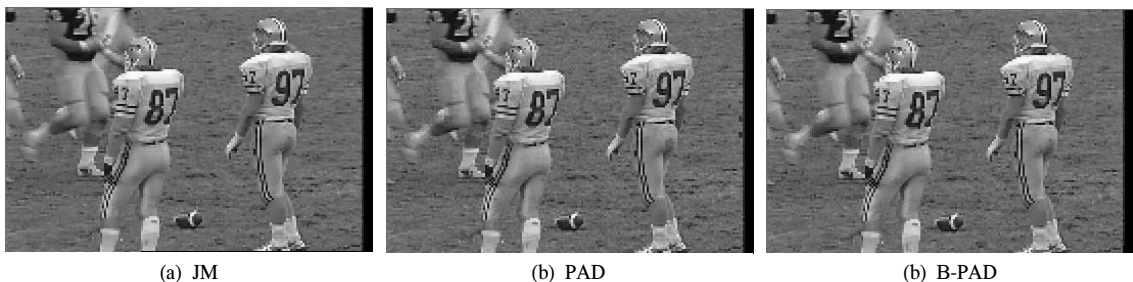


그림 9. 십네일 영상의 대한 인간 시각적인 비교 (Football)

두 조건문으로 구성되고 반면에 메모리 사용량과 참조횟수가 감소되므로, S/W 또는 H/W로 설계에서 메모리 참조 최적화에 의해 제안된 방법을 적용하는 경우 요약영상 생성에서 추가적인 시간 절약이 기대된다.

VI. 결론

본 연구에서는 H.264 동영상 압축 비트열에서 역추적 방법이 적용된 효율적인 요약영상 추출방법을 제안하고 JM 모델을 이용한 요약영상과 비교하였다. 제안된 B-PAD 방법은 간단한 역추적 알고리즘의 적용으로 연산량을 50% 줄일 수 있다. 실험으로 역추적 과정으로 인한 추가적인 시간 증가에도 불구하고 해상도에 관계없이 평균 40%의 연산량으로 에러 전파가 없는 인간 시각적으로 매우 우수한 요약영상이 추출됨을 확인하였다. 또한 최근에 발표된 방법¹⁹⁾은 영상의 모든 주파수 성분을 별도 저장하기 위해 JM보다 약 2배의 메모리가 요구되는데 비해 B-PAD는 15~44%의 메모리만을 사용한다.

B-PAD는 표준 H.264 디코더에 따르므로 기존 응용 H/W 또는 S/W에 쉽게 적용 가능한 장점을 가지고 있다. 그러므로 제안된 B-PAD는 DMB에서의 다양한 응용이 기대된다. 최근 보편화 되고 있는 DMB 장비에서 고려되어야 할 문제 중 하나가 전력과 속도 문제이다. 따라서 일반적인 DMB 시청 장비는 디블록킹 필터를 제거하는 등의 방법으로 연산량을 줄이고 있다. 그러나 요약영상이 필요한 경우 모든 복호 과정을 수행하고 있기 때문에 전력 낭비가 발생한다. 특히 휴대용 시청 장비의 경우 요약영상을 요구하는 여러 응용 분야에서 최소한의 연산으로 요약영상의 추출이 가능한 방법이 제시될 필요가 있다.

참 고 문 헌

[1] J. Jiang, Y. Weng, "Video Extraction for Fast Content Access to MPEG Compressed Video," IEEE Trans. CSVT, vol.14, no.5, pp.595-605, May 2004.
 [2] H.A. Cohen, "Retrieval and Browsing of Images Using Image Thumbnails," Journal of Visual Comm. and Image Representation, vol.8, no. 2, pp.226-234, Jun. 1997
 [3] C.H. Yim, M. Isnardi, "An Efficient Method

for DCT-Domain Image Resizing with Mixed Field/Frame-Mode Macroblocks," IEEE Trans. CSVT, vol.9, no.5, pp.696-700, Aug. 1999.
 [4] S. Martucci, "Image resizing in the discrete cosine transform domain," Image Processing ICIP, vol.2, pp.244-247, Oct. 1995.
 [5] E. Dugad, M. Ahuja, "A Fast Scheme for Image Size Change in the Compressed Domain," IEEE Trans, CSVT, vol.11, no.4, pp.461-474, Apr. 2001.
 [6] ITU-T Recommendation H.264/AVC 264 and ISO/IEC 1496-10, Advanced Video Coding for Generic Audiovisual Services, May 2003.
 [7] C. Gomila, P. Yin, "New features and applications of the H.264 video coding," Int. Conf. on Information Tech. 2003, pp.6-10, 11-13 Aug. 2003
 [8] G. Sullivan, T. Wiegand, "Video compression -from concepts to the H.264/AVC standard," Proc. IEEE, no.1, pp.18-31, Jan. 2005.
 [9] 유상준 외 5, "H.264/AVC 비트스트림에서 효율적으로 축소영상을 추출하는 방법", 방송공학회논문지, 13권 2호, pp 222-235, 2008.
 [10] 권용광, 김성민, 원치선, "H.264의 인트라 예측모드와 DC를 이용한 요약영상 추출 방법", 통신학회학계기술대회, vol.35, pp. 215, 2007.
 [11] 권용광, 원치선, "H.264/AVC의 인트라 예측 모드를 이용한 바이오 동영상의 고속 Thumbnail 추출 시스템", 화상학회지, 13권 3호, 2007.
 [12] 권용광, 원치선, "DMB를 위한 요약영상 추출 알고리즘", 방송공학회 논문지, 12권 5호, pp 525-534, 2007

권용광 (Yong-kwang Kwon)

중신회원



2002년 2월 동국대학교 전자공학
과 공학석사
2008년 2월 동국대학교 전자공학
과 공학박사
<관심분야> H.264/AVC, 영상처리,
디지털 통신