

다중 참조 영상의 적응적 선택 및 선택적 인트라 모드를 이용한 H.264/AVC의 고속 모드 결정 방법

정희원 이 옹 호*, 이 정 호*, 조 익 환*, 정 동 석**

Fast Mode Decision in H.264/AVC Using Adaptive Selection of Reference Frame and Selective Intra Mode

Woong-ho Lee*, Jung-ho Lee*, Ik-Hwan Cho*, Dong-Seok Jeong** *Regular Members*

요 약

비트율-왜곡 최적화 기법은 H.264/AVC(Advance Video Coding)의 부호화 효율을 높이기 위한 방법이지만 하나 모드 결정 과정 중 부호화기의 복잡도를 높아지는 단점이 있다. 많은 고속화 모드결정 연구들이 모드결정의 복잡도를 줄이기 위하여 제안되어져 왔다. 본 논문에서는 H.264/AVC의 모드결정의 전체적인 복잡도를 줄이기 위하여 다중 참조 영상 선택 고속화 알고리즘과 선택적인 인트라 모드 선택 알고리즘의 두 가지 고속화 알고리즘을 제안한다. 참조영상 선택 고속화 알고리즘은 인터 모드 결정에 효과적이며, 선택적인 인트라 모드 선택 알고리즘은 과도한 인트라 모드 결정의 계산량을 효율적으로 감소시켰다. 제안된 알고리즘을 실험한 결과로 평균 44.63%의 부호화 시간 감소비를 보이면서 영상의 열화와 같은 부호화 효율 감소는 거의 눈에 띄지 않았다.

Key Words : H.264, mode decision, reference frame, encoder optimization, intra prediction

ABSTRACT

Rate-constrained coding is one of the many coding-efficiency oriented tools of H.264/AVC, but mode decision process of RDO(Rate distortion optimization) requires high computational complexity. Many fast mode decision algorithms have been proposed to reduce the computational complexity of mode decision. In this paper, we propose two algorithms for reduction of mode decision in H.264/AVC, which are the fast reference frame selection and selective intra prediction mode decision. Fast reference frame selection is efficient for inter prediction and selective intra prediction mode decision can effectively reduce excessive calculation load of intra prediction mode decision. The simulation results showed that the proposed methods could reduce the encoding time of the overall sequences by 44.63 % on average without any noticeable degradation of the coding efficiency.

I. 서 론

최근 ISO와 MPEG에서 표준이 확정된 국제 비디오 부호화 표준인 H.264/AVC(Advanced Video Coding)은 기존의 비디오 부호화 표준인 H.261,3과 MPEG-2,4에 비해서 화질 및 부호화 효율적인 측면

에서 모두 우수하다. 이러한 우수한 부호화 효율을 이루기위해서 H.264/AVC는 기존의 부호화 방법뿐만 아니라 효율적인 부호화 방법들을 채용하였다¹⁾.

여러 주요한 기술들 중 기존의 비디오 부호화 표준과 다른 기술들로는 4×4 블록단위 정수 변환, 움직임 벡터 예측을 위한 다중 참조 영상, 인트라

* 인하대학교 전자공학과 (Hlee, julian, teddydino}@inhaian.net),
논문번호 : KICS2006-01-054, 접수일자 : 2006년 1월 30일, 최종논문접수일자 : 2006년 3월 2일

** 인하대학교 전자공학과 교수 (dsjeong@inha.ac.kr)

예측 부호화, 1/4 화소 단위의 움직임 벡터 예측, Context-based Adaptive Binary Arithmetic Coding (CABAC) 등이 있다. 이러한 특징들로 인하여 기존의 비디오 부호화 표준인 MPEG-2, MPEG-4와 비교하였을 때 동일화질에서 약 60%의 효율적인 부호화를 이루었으며, H.263과 비교했을 때 동일 화질에서 약 30%의 비트량이 절감되었다²⁾. 이중에 인터 프레임 내에서 인트라 예측 부호화의 경우는 동일 비트량에서 JPEG2000과 MPEG-4보다 약 3-4dB의 화질의 개선을 보였다³⁾. 이런 우수한 부호화 효율에도 불구하고 기존의 비디오 부호화 표준과 비교 시 부호화기 및 복호화기의 복잡도가 이전 비디오 부호화 표준보다 훨씬 증가하는 단점이 있다. 이러한 복잡한 부호화 연산의 원인으로 가변블록 움직임 예측과 각 매크로 블록의 모드결정을 위한 연산과정 때문이다. 각 매크로 블록의 최종 모드 결정에 필요한 비트율-왜곡 최적화 기법(Rate-Distortion Optimization)이 연산량을 증가시키는 주요 요인 중 하나이다.

최근에 H.264/AVC의 부호화시 최종 모드결정에 필요한 연산량을 감소시키고 최적화하려는 연구들이 활발히 진행되었다⁴⁻⁸⁾. 그 중에서 [4], [5]에서는 움직임 예측을 위한 다중 참조 영상의 선택을 고속화함으로써 전체적인 부호화의 속도를 증가시켰다. 또 인트라 모드 결정의 고속화를 위하여 F. Pan은 인트라 예측모드를 블록의 방향성 히스토그램(Directional histogram)을 사용하여 비트율-왜곡 최적화 연산 대상의 수를 줄임으로써 고속화를 이루었으며, 약 5%의 비트량이 증가에도 불구하고 I 슬라이스의 경우 55-65%의 연산량의 절감을 보였다⁶⁾. P-슬라이스 내의 시간적인 연관성과 공간적인 연관성을 고려하여 선택적인 인트라 모드 결정방법으로 연산량을 줄이려는 연구가 [7]에서 보고되었다. [8]에서는 주위블록의 모드에 대한 확률을 미리 조사하여 발생 확률이 우세한 모드에 대하여만 비트율-왜곡 최적화를 적용하는 방법이 제안되었다.

본 논문에서는 H.264/AVC의 표준을 유지하면서 움직임 벡터를 위한 다중 참조 영상의 선택을 고속화하고 영상의 인트라 모드와 인트라 모드와의 관계를 살펴서 선택적인 인트라 모드를 비트율-왜곡 최적화 계산을 위한 후보 모드로 선택함으로써 계산량을 감소시키는 알고리즘을 제안한다. 2장에서는 움직임 벡터의 참조영상들 내의 발생 확률 및 고속화 알고리즘을 제안하고 3장에서는 비트율-왜곡 최적화를 위한 선택적인 인트라 모드 결정 고속화를 제안한다. 4장의 실험과 성능 검증을 통하여 제안된

알고리즘들이 기존의 표준을 해치지 않고 실험 표준 영상들에서의 화질 저하 및 부호화 효율을 유지하면서 계산량이 감소함을 보인다.

II. H.264/AVC의 일반적인 모드 결정 방법

H.264/AVC의 일반적인 모드 결정은 I-슬라이스와 P 또는 B-슬라이스에서 구분된다. I-슬라이스에서의 부호화는 인트라 모드로만 부호화를 진행하게 된다. 기존의 비디오 부호화 표준에서 적용되었던 인트라 코딩은 대상 매크로블록의 DCT(Discrete Cosine Transform)변환 후 부호화하는 것과는 다르게 주위 블록과의 공간적인 상관성을 이용한 인트라 예측 후 예측된 값과의 차만을 정수 변화 후 부호화에 적용한다. P또는 B-슬라이스에서는 기존의 비디오 표준에서의 다른 점은 인터 코딩을 위한 움직임 벡터의 예측시 이전 프레임뿐만 아니라 그전의 다중 참조 프레임에서 움직임예측을 하고 인터 코딩의 시간적인 중복성을 제거함과 동시에 해당 I-슬라이스에서의 인트라 예측 부호화처럼 공간적인 중복성을 제거하기 위하여 인트라 예측 코딩을 하게 된다. 다중 참조 프레임 움직임 벡터 예측은 반복되는 움직임이 우세한 영상에서 보다 효율적인 움직임 벡터를 찾고 부호화 효율을 높일 수 있다. P 또는 B-슬라이스에서 인트라 예측 부호화는 휘도성분의 경우 주변 16×16 블록 및 4×4 서브블록으로 나누어서 수행하고 16×16의 경우 이웃한 블록의 공간적인 상관성을 이용하여 4개의 방향성에 대하여 인트라 예측 블록을 만들고 비트율-왜곡 최적화를 적용한다. 4×4 서브블록의 경우 9개의 방향에 대하여 인트라 예측을 수행한다. 컬러 성분의 경우는 16×16의 휘도와 같은 4개의 인트라 예측을 수행한다. 그림 1은 가변 블록 인트라 모드 및 인트라 모드를 나타낸다.

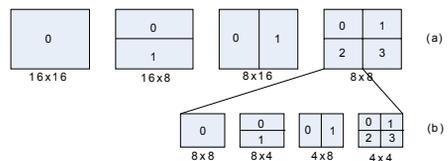


그림 1. 매크로 블록 파티션과 서브 매크로블록 파티션 (a) 매크로블록 파티션 : 모드 1, 모드 2, 모드 3(b)부-매크로블록 파티션: 모드 4, 모드 5, 모드 6, 모드 7

Fig. 1. MB partition and Sub-MB partition (a) MB partition : mode 1(16×16), mode 2(16×8), mode 3(8×16), (b) Sub-MB partition: mode 4(8×8), mode 5(8×4), mode 6(4×8), mode 7(4×4)

H.264/AVC에서는 최적의 모드를 위하여 인트라 예측 모드와 인터 모드의 결정을 위한 방법으로 Low-complexity mode와 High-complexity mode로 나누어진다. Low-complexity mode는 단순히 각 모드간의 Sum of Squared Difference(SSD)를 비교하여 최소의 에러를 갖는 모드를 최적 모드로 결정하는 반면 High-complexity mode는 비트율-왜곡 최적화 기법을 통해 최적 모드를 결정한다.

$$J(s, c, MODE|QP, \lambda_{mode}) = SSD(s, c, MODE|QP) + \lambda_{mode} \cdot R(s, c, MODE|QP) \quad (1)$$

모드결정을 위한 비트율-왜곡 최적화는 식(1)에 의해서 결정되며 $J(\bullet)$ 값을 최소로 하는 모드가 최적의 모드로 선택이 되어 부호화 된다. 식 (1)에서 $s, c, MODE, QP$ 와 λ_{mode} 는 각각 원 영상, 부호화-복호화 복원된 영상, 모드, 양자화 계수 그리고 각 모드에 따른 ramda 계수값을 나타낸다. $SSD(\bullet)$ 는 Sum of Squared Difference로써 현재 모드의 distortion을 구할 수 있고 $R(\bullet)$ 에서는 그 모드에 해당하는 압축된 비트열의 길이를 나타낸다. 이때 비트율-왜곡 최적화한 모드를 구하기 위한 계산량은 인터모드의 경우는 매크로블록의 각 파티션 모드별 연산이 필요하고 인트라모드의 경우는 각각의 휘도 성분, 컬러성분과 그 블록의 사이즈에 따른 모드의 수만큼 계산해야 한다. H.264/AVC에서는(휘도성분의 I4MB 모드수×부-블록의 수+I16MB 모드수)×컬러성분의 모드수의 비트율-왜곡 함수의 계산이 필요하다. 인터모드의 경우 최대 비트율-왜곡 계산을 128번과 인트라 모드의 경우 640번의 연산을 수행하게 된다. 비트율-왜곡 최적화를 적용한 High-complexity mode는 최적의 모드 결정을 위하여 계산량이 증가 하는 단점이 있다. 위에서 기술한 이유로 인하여 모드 결정에 필요한 계산량을 줄이기 위한 연구의 필요성이 대두되며 이때 최종 부호화 효율의 저하를 최소화하는 제한사항이 따르게 된다.

III. 움직임 벡터 예측을 위한 다중 참조 영상 선택 고속화

직입 벡터 예측은 비디오 부호화의 과정 중 가장 연산량을 많이 차지하는 부분으로 많은 고속화 알고리즘들이 제안되었다. H.264/AVC에서는 움직임 예측시 이전 프레임에 대한 움직임 예측뿐만 아니

라 과거의 프레임들을 버퍼에 저장하고 여러 참조 프레임에서 움직임 예측을 하는 다중 참조 프레임을 적용한다. 다중 참조 프레임을 적용할 때는 영상 내의 눈의 깜박임 같은 단시간의 반복되는 움직임 벡터의 발생에 매우 효율적으로 이전 참조 프레임 예측 보다 정확한 예측을 통하여 부호화 효율을 증가시킬 수 있다^[1]. 그러나 다중 참조 프레임을 적용할 때 부호화기에서 움직임 벡터 예측을 위한 계산량은 참조 프레임의 숫자만큼 선형적으로 증가하게 되는 단점을 갖게 된다. 가장 최근의 참조 영상에 대하여 가변 블록 움직임 벡터 예측을 수행하게 된다. 참조 소프트웨어에서는 이때 16×16 블록 파티션(Partition)을 먼저 수행하고 이후에 4×4 부-블록 파티션에 대하여 수행한 후 각 움직임 벡터를 각 모드의 RDcost(Rate-Distortion cost)를 비교하기 위하여 저장해 놓는다. 일반적으로 최종적으로 16×16 블록 파티션 모드 : 모드 0, 모드 1, 모드 2, 모드 3 로 결정될 때는 현재의 블록의 움직임이 없거나 (motionless) 블록내의 특성이 균질한(homogeneous) 특성을 지닐 때이다. 그러므로 현재의 블록이 만약 16×16 매크로블록 파티션 모드들이 동일 참조 프레임에서 발행했다면 그 참조 프레임에서 4×4 부-매크로블록 파티션 모드들의 참조 프레임으로 쓰일 확률이 높다.

제안하는 참조 프레임 선택 고속화는 다음과 같은 순서로 진행한다.

단계 1: 16×16 매크로블록 파티션의 각 모드 즉 모드 1, 2, 3에 대하여 모든 참조 프레임에 대하여 움직임 예측을 수행한다. 이때 각 모드에 대하여 각각의 최적의 참조 프레임을 얻게 된다.

단계 2: 각 16×16 매크로블록 파티션의 최적의 참조 프레임을 $best-ref_{mode}$ 라 하고 각 모드 1, 2, 3의 최적의 참조 프레임을 찾는다.

$$best-ref_{mode1} = best-ref_{mode2} = best-ref_{mode3} \quad (2)$$

식(2)와 같이 최적의 프레임이 각 모드에 대하여 동일 프레임에서 발생한다면 이 참조 프레임을 4×4 부-매크로블록의 추천 참조프레임으로 정하고 단계 3으로 넘어 간다. 참조프레임이 동일하지 않다면 단계 4로 진행한다.

단계 3: 4×4 부-매크로블록 파티션 모드들을 추천 참조 프레임에서만 움직임 예측을 적용 움직임

벡터를 찾아내고 단계 5로 간다.

단계 4: 4×4 부-매크로블록 파티션에 대한 움직임 예측을 모든 참조 프레임에 대하여 수행하고 단계 5로 간다.

단계 5: 각 인터 모드의 RDcost를 계산하여 최적의 인터 모드를 찾아낸다.

위와 같은 단계로 다중프레임 참조 움직임 예측 시 이전 16×16 매크로블록 모드에서 찾은 동일 참조프레임만 4×4 부-매크로블록에 적용하여 연산량을 줄일 수 있고 참조 프레임내의 인터 모드를 최종적으로 확정하게 된다.

IV. 선택적인 인터라 모드 결정 방법

매크로 블록의 인터라 모드를 결정하기 위하여 각 매크로블록의 휘도 성분을 16×16 매크로블록 파티션(Partition)에 대하여 네 가지의 방향성에 대하여 예측을 수행하고 4×4 부-블록 파티션에 대하여 여덟 개의 방향성에 대하여 예측을 수행 후 각 예측된 모드를 식(1)의 RDcost가 최소가 되는 모드를 최종 인터라 모드로 결정하게 된다. 일반적으로 균질한 블록(homogeneous block)인 경우 16×16 매크로블록 파티션에 대한 모드로 결정되고 비 균질 블록(non-homogeneous block)인 경우는 보다 적은 4×4 부-매크로로 블록 파티션 모드로 세밀하게 예측모드가 결정된다. 그러므로 적용될 인터라 모드를 두 개의 군으로 나눌 수 있다. 첫 번째 인터라 모드군(Intra Mode Group 1)은 모드 1(16×16), 2(16×8), 3(8×16)을 포함하고 나머지 인터라 모드군(Intra Mode Group 2)은 모드 4(8×8), 5(8×4), 6(4×8), 7(4×4)을 포함한다.

인터라 모드 결정의 복잡도를 감소시키기 위한 고속화 알고리즘은 다음과 같다.

단계 1: 인터 모드 결정후의 최적화 인터 모드에 대하여 16×16 매크로블록 파티션 모드에 대하여 제 1 인터라 모드 군안에 있는 인터라 모드들에 대하여만 비트율-왜곡 최적화를 적용한다.

단계 2: 만약 인터 모드가 IMG 1 에 속한다면 I16MB를 수행한다.

단계 3: 만약 인터 모드가 IMG 2 에 속한다면 I4MB를 수행한다.

단계 4: 3장에서 구한 최적 인터 모드와 4장에서 구한 최적 인터라 모드 중 RDcost가 가장 적은 모드를 최적 모드로 결정한다.

3장의 고속 다중 참조 영상 선택 방법과 4장의 선택적인 인터라 모드 결정 방법을 혼합하여 그림 2의 블록다이어그램과 같은 방법으로 H.264/AVC의 전체적인 모드 결정의 복잡도를 감소시킨다.

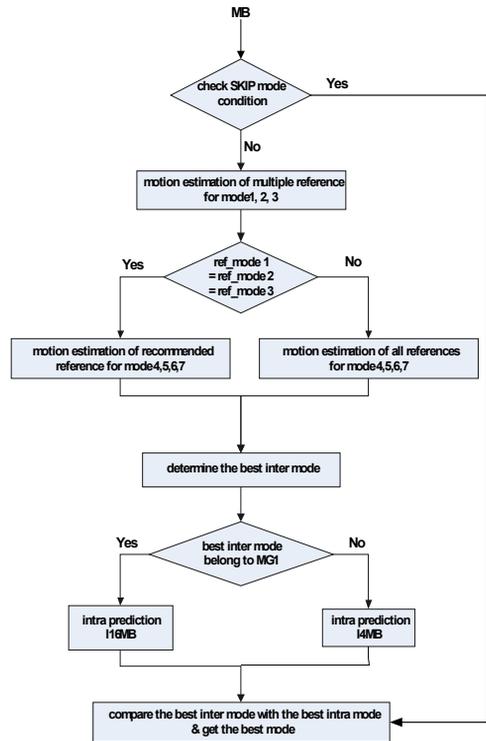


그림 2. 제안하는 모드결정 방법의 블록다이어그램
Fig. 2. Block diagram of the proposed mode decision algorithm

V. 실험 및 성능

제안된 알고리즘은 HHI(Heinrich-Hertz-Institute)에서 제공하는 H.264/AVC의 참조 소프트웨어인 JM 10.1에서 구현하였다^[9]. 또 JVT(Joint Video Team)에서 제시하는 공통 실험 조건(Common Test Condition)을 맞추기 위하여 실험 구현을 위한 조건을 맞추었다^[10].

표 1. 실험 테스트 비디오 영상들
Table 1. Test video sequences

Size	Video	Frames
QCIF (176×144)	container	200
	carphone	150
	coastguard	100
CIF (352×288)	foreman	200
	football	150
	crew	100

실험의 테스트 영상들은 표 1과 같이 다양한 크기 및 프레임레이트의 비디오 표준영상에 대하여 실험을 하였다.

부호화기의 부호화 옵션들은 다음과 같다.

- # of reference frames = 5
- fast search motion estimation
- context-based adaptive length coding (CAVLC)
- Rate Distortion Optimization (RDO)
- GOP structure = IPPP
- QP = 24, 28, 32, 36, 40

위의 부호화기의 옵션들을 사용하여 다음과 같이 원 JM 부호화 알고리즘과 [7]에서 제안한 모드 고속화 알고리즘 및 4장에서 제안한 선택적인 인트라 모드 알고리즘, 3장에서 제안한 다중 참조 영상 선택 고속화 알고리즘 그리고 마지막으로 제안한 알고리즘의 조합을 실험하였다.

- ① JM original with RDO
- ② JM with early termination and selective intra mode coding
- ③ JM with proposed selective intra mode coding algorithm
- ④ JM with proposed fast reference frame selection algorithm
- ⑤ JM with ③ +④

위의 알고리즘 중 ①의 알고리즘은 [7]에서 제안되어서 JM10.1의 부호화기의 기본 옵션으로 구현되어 있다. 위의 각 알고리즘의 부호화 효율의 변화를 측정하고자 다음과 같은 식들을 사용하였다.

1) PSNR change[dB]

$$CH_PSNR = PSNR_{algorithm} - PSNR_{original} \quad (3)$$

2) Bitrate change[%]

$$CH_Bits = \frac{(Bits_{algorithm} - Bits_{original})}{Bits_{original}} \times 100 \quad (4)$$

3) Encoding time change[%]

$$CH_Time = \frac{(Time_{algorithm} - Time_{original})}{Time_{original}} \times 100 \quad (5)$$

일반적으로 고속화 알고리즘들은 고속화에 따른 화질의 열화를 수반하게 된다. 비트율-왜곡측면에서는 부호화 효율의 저하는 제안된 알고리즘에서는 동일 비트율에서의 PSNR의 저하 및 동일 화질에서의 비트량의 증가를 뜻한다. 우선 각 알고리즘의 원 JM 알고리즘과 비교한 PSNR의 저하를 표 2에서 나타내었다.

표 2. 테스트영상들의 각 알고리즘의 PSNR 변화
Table 2. PSNR difference of each test video

Video		CH_PSNR [dB]			
		②	③	④	⑤
Q	container	0.000	-0.002	-0.002	0.002
C	carphone	0.000	0.011	-0.057	-0.056
I	coastguard	0.000	-0.008	-0.060	-0.056
F					
C	foreman	0.000	-0.018	-0.052	-0.066
I	football	0.000	-0.087	-0.043	-0.165
F	crew	0.000	-0.024	-0.041	-0.075

원 JM 부호화와 비교하여 ② 알고리즘은 PSNR의 변화가 없으나 제안한 알고리즘 ③, ④, ⑤에서는 각각 평균 0.0217, 0.047, 0.069 dB의 PSNR이 떨어진 것을 알 수 있다. 알고리즘 ③의 PSNR의 변화가 알고리즘 ④의 PSNR의 변화보다 적게 나왔음을 확인할 수 있었는데 이는 일반적으로 P-슬라이스내의 인트라 모드의 선택 비율이 매우 낮으므로 고속화에 따른 화질의 저하에 영향을 거의 주지 않았음을 알 수 있다. 또 비록 알고리즘 ⑤에서 0.069 dB의 화질저하를 보였으나 이는 육안으로는 구별할 수 없을 정도의 아주 미미한 화질의 열화를 뜻하므로 고속화에 따른 화질의 열화는 무시할 수 있다.

비트량의 변화를 살펴보면 표 3과 같이 원 JM 부호화기에서의 비트량과 각 알고리즘을 적용한 부호화기에서의 비트량의 변화를 볼 수 있다. 알고리즘 ③, ④, ⑤ 에서 각각 0.63%, 0.53%, 0.22%의

표 3. 테스트영상들의 각 알고리즘별 비트량의 변화
Table 3. Bits difference of each test video

Video		CH_Bits [%]			
		②	③	④	⑤
Q	container	0.00	0.00	0.92	1.05
C	carphone	0.00	0.22	1.06	1.08
I	coastguard	0.00	0.06	1.31	1.44
F					
C	foreman	0.00	0.47	1.37	1.87
I	football	0.00	1.72	-0.74	1.26
F	crew	0.00	1.30	-0.72	0.61

비트량이 증가했다. ②의 알고리즘 외의 다른 고속화 알고리즘들에서 부호화 비트량이 늘어난 것을 알 수 있었다. 전반적으로는 ③의 알고리즘에 의한 비트량의 증가가 ④의 알고리즘에 따른 비트량의 증가보다 적은 것이 확인되었으나 CIF 비디오 중 football 과 crew에서는 ④의 알고리즘에 의한 비트량이 오히려 감소했다. 이들 비디오의 특성은 전반적으로 영상내의 크고 많은 움직임 벡터들이 존재하고 P-슬라이스내의 다수의 인트라 모드로 결정된 매크로블록이 존재하는 경우로 알고리즘 ③에서 선택적인 인트라 모드로 부호화하는 과정 중 최적의 모드로 선택을 못한 경우가 발생했기 때문이다.

제안된 고속화 알고리즘들의 전체적인 화질 열화를 종합적으로 보기위하여 비트율-왜곡 곡선을 그림 3와 같이 나타낼 수 있다.

그림 3의 (a)는 QCIF container 비디오의 비트율-왜곡곡선으로서 제안된 고속화 알고리즘간의 곡선들의 원JM 부호화된 비트율-왜곡 곡선과 거의 겹쳐지는 것을 확인할 수 있으므로 고속화 알고리즘에

따른 화질의 열화는 거의 없다고 봐도 무방하다. 그림 3의 (b)는 CIF foreman 비디오의 비트율-왜곡 곡선으로 낮은 비트율에서는 곡선들이 겹쳐지고 있었으나 500 kbps이상의 높은 비트율에서는 동일 비트율에서 약 0.5 dB의 화질의 열화가 보였다.

표 4는 각 알고리즘의 고속화의 효과인 부호화 시간의 감소비를 나타낸다.

표 4. 테스트영상들의 각 알고리즘별 부호화 시간의 변화
Table 4. Encoding time change of each test video

Video		CH_Time [%]			
		②	③	④	⑤
Q	container	0.04	-30.47	-17.56	-50.09
C	carphone	-0.34	-25.46	-16.22	-44.22
I	coastguard	-0.13	-20.79	-21.83	-46.09
F					
C	foreman	-0.21	-23.32	-17.95	-43.69
I	football	0.03	-18.28	-20.60	-41.55
F	crew	-0.08	-19.87	-20.02	-42.10

알고리즘 ②의 부호화 시간 감소비는 평균 0.12% 이내로 전체적인 부호화 시간에 비하면 미미한 반면 알고리즘 ③과 ④를 적용한 부호화 시간은 각각 평균 23.03%, 19.03%의 감소를 보였다. 알고리즘 ③이 알고리즘 ④ 보다 전반적으로 부호화기의 복잡도를 감소시켰으며 이는 전체 부호화 과정 중 인트라 모드결정에 많은 계산량이 차지하고 있기 때문이다. 알고리즘 ③과 ④를 결합한 알고리즘 ⑤에서는 평균 44.63% 부호화 시간의 감소비를 나타냈다. 이전의 고속화에 따른 화질의 열화를 고려하더라도 매우 큰 부호화 시간의 감소를 의미한다.

그림 4는 부호화기의 양자화 계수의 변화에 따른 각 알고리즘의 부호화 시간의 감소비를 나타내는 것으로서 제안된 알고리즘은 양자화 계수의 변화에

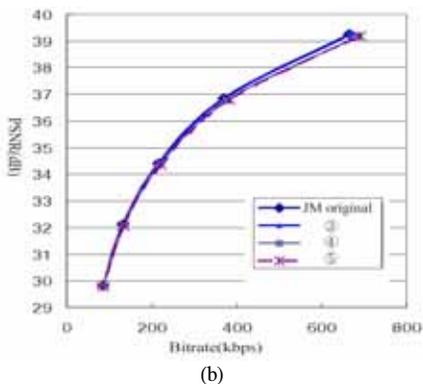
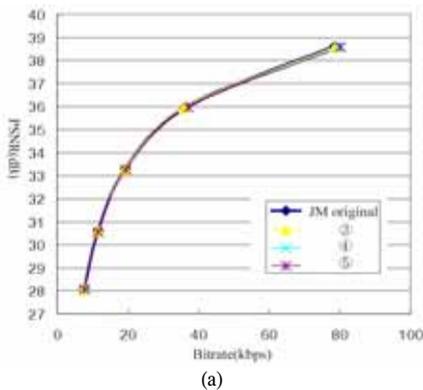


그림 3. 각 알고리즘별 비트율-왜곡 곡선 (a) QCIF container의 비트율-왜곡 곡선, (b) CIF foreman의 비트율-왜곡 곡선
Fig. 3. RD-curve of the test video sequences (a) RD-curve of QCIF container, (b) RD-curve of CIF foreman

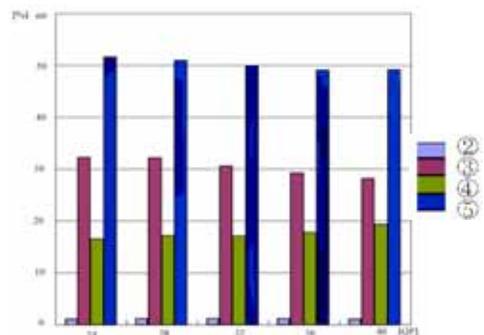


그림 4. 양자화 계수에 따른 알고리즘별 부호화시간 감소비
Fig. 4. Encoding time ratio of each algorithm with variation of quantization

도 부호화 시간의 감소비가 거의 일정하다. 이는 제안된 알고리즘이 비트율과 상관없이 일반화될 수 있다는 것을 의미한다.

VI. 결론

H.264/AVC는 기존의 비디오 부호화 표준보다 월등한 부호화 효율과 성능을 지니고 있으며 최근에 기존의 비디오 부호화 표준을 대체하면서 그 응용 분야가 확대되고 있다. 월등한 부호화 효율의 장점에도 불구하고 H.264/AVC는 부호화 과정의 복잡도가 증가한다. 최근 이러한 복잡도를 줄이는 연구들이 많이 진행되었는데 부호화 과정의 복잡도가 증가하는 주된 이유 중에서 매크로 블록 부호화시 비트율-왜곡 최적화에 따른 인트라 예측 모드 결정과 움직임 벡터 예측 시 다중의 참조 영상을 사용하는 것이다.

본 논문은 다중 참조 영상들 중 매크로블록 파티션의 움직임 벡터 예측 후 부-매크로블록의 움직임 예측을 위한 참조 영상의 선택을 고속화하고 최적의 인트라 모드와 인트라 모드와의 관계의 연관성을 이용하여 비트율-왜곡 최적화의 계산 대상인 인트라 예측 모드의 수를 줄임으로써 전체적인 부호화의 고속화를 이루었다. 제안된 고속화 알고리즘은 약 45%의 부호화 시간의 감소를 보였으며, 특히 표준 부호화 과정을 해치지 않고 고속화를 이루었다. 또 고속화에 따른 화질의 열화와 비트량의 증가는 고속화의 정도에 비하여 매우 미미하였다.

참고 문헌

[1] Wiegand, T., Sullivan, G.J.; Bjntegaard, G.; Luthra, A., "Overview of the H.264/AVC video coding standard", IEEE Trans. on Circuits and Systems for Video Tech., Vol. 13, no. 7, pp. 560-576, 2003.

[2] Ostermann, J., Bormans, J., List, P., Marpe, D., Narroschke, M., Pereira, F. Stockhammer, T., Wedi, T., "Video coding with H.264/AVC: tools, performance, and complexity", IEEE Circuits and Systems Magazine, Vol. 4, no. 1, pp. 7-28, 2004.

[3] Helbacg T., "Performance comparison : H. 26L intra coding vs. JPEG2000", JVT of ISO/IEC MPEG and ITU-T VCEG, Klagenfurt,

Austria, July 2002.

[4] Ozbek, N.; Tekalp, A.M., "Fast H. 264/AVC Video Encoding with Multiple Frame References", IEEE ICIP 2005, Vol. 1, pp. 597-600, 2005.

[5] Hung-Ju Li, Ching-Ting Hsu, Mei-Juan Chen, "Fast multiple reference frame selection method for motion estimation in JVT/H.264", The 2004 Proc. of IEEE Asia-Pacific Conf. on Circuits and Systems, vol. 1, pp. 605-608, 2004.

[6] F. Pan, X. Lin, R. Susanto, K. P. Lim, Z. G. Li, G. N. Feng, D. J. Wu, and S. Wu, "Fast Mode Decision for Intra Prediction", JVT, G013, 2003

[7] Jeon, B., Lee, J, "Fast mode decision for H.264", JVT-J033, Hawaii, USA, 2003.

[8] C. Kim, H. Shih, C.C.J. Kuo, "Feature-based intra-prediction mode decision for H.264", ICIP 2004 Image Processing, Vol. 2, pp. 769-772, 2004.

[9] JM reference software : <http://iphome.hhi.de/suehring/tml/>

[10] G. Bjontegaard, "Recommended Simulation Conditions for H.26L", VCEG-M75, April 2001.

[11] Y. Shen, C. Huang, L. Yu, D. Zhang, J. Li, "Fast multiframe motion estimation algorithm in H.264", ICSP '04, vol.2, pp. 1187-1190, 2004.

이 응 호 (Woong-ho Lee)

정회원



1996년 2월 인하대학교 전자공학과 졸업

2002년 2월 인하대학교 전자공학과 공학석사

2002년 3월~현재 인하대학교 전자공학과 박사과정

<관심분야> 영상처리, Video

Coding, MPEG-4, H.264/AVC, SVC, 3DAV, MVC(Multi-View Coding)

이 정 호 (Jung-Ho Lee)

준회원



1999년 2월 인하대학교 전자공
학과 졸업
2001년 2월 인하대학교 전자공
학과 공학석사
2001년 3월~현재 인하대학교
전자공학과 박사과정
<관심분야>Multi-view Coding,

Panoramic image

정 동 석 (Dong-Seok Jeong)

정회원



1977년 2월 서울대학교 전기공
학과 졸업
1985년 Virginia 주립 공과대학
전자공학과 공학석사
1988년 Virginia 주립 공과대학
전자공학과 공학박사
1988년 3월~현재 인하대학교

전자공학과 교수

<관심분야> 영상처리, 컴퓨터 비전, 멀티미디어 정보
처리

조 익 환 (Ik-Hwan Cho)

준회원



2000년 2월 인하대학교 전자공
학과 졸업
2002년 2월 인하대학교 전자공
학과 공학석사
2002년 3월~현재 인하대학교
전자공학과 박사과정
<관심분야>영상처리, MPEG-4,

MPEG-7, SVC(Scalable Video Coding), 의료영상
처리, 이미지 워터마킹