

논문 2009-46SP-2-6

# H.264/AVC 부호기에서 움직임 벡터 맵을 이용한 적응적인 부호화 모드 결정 방법

( Adaptive Coding Mode Decision Algorithm using Motion Vector Map  
in H.264/AVC Video Coding )

김태정\*, 고만근\*, 서재원\*\*

( Tae-Jung Kim, Man-Geun Ko, and Jae-Won Suh )

## 요약

우리는 H.264/AVC의 비디오 부호화에서 조기 인트라 모드 생략을 결정하기 위한 알고리즘을 제안한다. 새롭게 추가된 다양한 예측 방법들에 의한 매크로블록 부호화 방법은 압축의 효율의 증가를 가져오지만, 모든 부호화 가능한 모드에 대해 윌-왜곡 함수를 계산하여 가장 효율이 좋은 모드를 선택하기 때문에 상당한 계산량을 요구한다. 이 논문에서는 인트라 프레임에 대한 부호화 시간을 감소시키기 위해서, 적응적인 움직임 벡터 맵(AMVM)을 이용한 모드 결정 방법을 H.264/AVC 비디오 부호화기에서 제안한다. 제안한 알고리즘은 PSNR과 Bit rate 그리고 부호화 처리시간에 대해서 일반적으로 좋은 성능을 가진다.

## Abstract

We propose a fast intra mode skip decision algorithm for H.264/AVC video encoding. Although newly added MB encoding algorithms based on various prediction methods increase compression ratio, they require a significant increase in the computational complexity because we calculate rate-distortion(RD) cost for all possible MB coding modes and then choose the best one. In this paper, we propose a fast mode decision algorithm based on an adaptive motion vector map(AMVM) method for H.264/AVC video encoding to reduce the processing time for the inter frame. We verify that the proposed algorithm generates generally good performances in PSNR, bit rates, and processing time.

**Keywords :** H.264/AVC, Mode Decision, Fast Intra Skip, Motion Vector Map, Rate-Distortion Function

## I. 서론

효과적인 동영상 압축을 위해 ITU-T와 ISO/IEC에서는 공동으로 H.264/AVC(Advanced Video Coding)라는 고성능 압축 효율을 갖는 동영상 부호화 표준을 정의하였다<sup>[1-2]</sup>. H.264/AVC 동영상 표준의 새로운 기술은 이전에 정의되어진 H.263<sup>[3]</sup>, MPEG-2<sup>[4]</sup>, 그리고

MPEG-4<sup>[5]</sup> 등과 유사한 압축 구조이다. 하지만 압축 효율을 증가시키기 위해 향상된 움직임 추정 기법, 인트라 예측 모드, 윌-왜곡 함수를 이용한 최적 모드 결정 방법<sup>[7]</sup>, 문맥 기반 적응적 이진 산술 부호화(Context based Adaptive Binary Arithmetic Coding, CABAC)<sup>[6]</sup> 등을 추가하였다.

향상된 움직임 추정 기법은 다양한 크기의 블록에 대해서 움직임 추정을 하기 때문에 부호화 효율에 있어 많은 이득을 제공한다<sup>[2]</sup>. 게다가 다중 참조영상을 이용한 1/4화소 단위의 정밀한 움직임 벡터 예측은 부호화 효율을 한층 더 향상 시켰다. 화면내 공간의 상관성을 제거하는 방향성을 이용한 인트라 모드 예측 기법도 많

\* 학생회원, \*\* 정회원, 충북대학교 전자공학과  
(Department of Electronics Engineering, Chungbuk National University)

※ 이 논문은 2008년도 정부재원(교육인적자원부 학술 연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2008-331-D00379)

접수일자: 2008년7월23일, 수정완료일: 2009년2월27일

은 압축 효과를 이끌어냈다. 또한 주위의 정보를 기반으로 부호표 테이블을 바꾸는 CABAC방법<sup>[6]</sup>과 울-왜곡 최적화(Rate - Distortion Optimization, RDO) 기술<sup>[7]</sup> 기반으로 인트라 예측 모드와 인터 예측 모드 등의 모든 부호화 가능한 모드 중에서 최적의 모드를 선택하여 부호화하기 때문에 압축 효율을 최대한 향상시킬 수 있었다. 이러한 창의적인 부호화 기법으로 인해 현재 H.264/AVC의 압축 효율은 기존의 동영상 압축 방법과 비교할 때 동일한 PSNR(Peak Signal to Noise Ratio)에서 두 배 이상이며, 동일한 비트율에서는 2dB 이상의 화질이 개선되었다. 하지만 이런 고성능 압축 효율을 갖기 위해서 기존의 압축 방법보다 10배 이상의 복잡도를 갖는다<sup>[2]</sup>.

이런 H.264/AVC의 복잡성은 실시간 처리에 무리가 따르기 때문에 동영상 압축 방법의 성능 저하는 최소화 하면서 복잡도를 줄이기 위한 많은 연구가 진행되고 있다. 특히 부호화 모드 결정시 생기는 복잡도를 줄이기 위한 많은 연구가 진행되었다<sup>[8~16]</sup>. 인터 모드의 파티션을 조기에 결정하여 경우의 수를 제한하는 방법<sup>[8~11]</sup>, 인트라 모드 예측 중 부분적인 예측 모드를 생략하는 방법<sup>[14~16]</sup>, 모든 부호화 가능한 모드를 생략하는 조기 SKIP 모드 결정 방법<sup>[17]</sup>, 인트라 모드의 예측 방법 전체를 생략하는 방법<sup>[12~13]</sup>들이 연구되고 있다. 인트라 예측과 인터 예측에서 부분적으로 예측 모드를 생략하는 방법은 부호화 효율은 어느 정도 유지되나 복잡도가 많이 줄어들지 않는다. 또한, 조기에 SKIP 모드를 결정하는 방법은 복잡도의 감소에 뛰어나지만 부호화 효율이 감소되는 단점이 있다. 본 논문에서는 인터 프레임에서 10%내외의 비중을 차지하는 인트라 부호화 모드이지만 모두 예측 과정을 거치기 때문에 복잡도가 증가한다는 사실을 이용해 이 과정을 적절히 생략하는 방법을 제한한다. 기존의 선행 연구로 Lee<sup>[12]</sup>와 Kim<sup>[13]</sup>이 제안한 방법들이 있다.

Lee<sup>[12]</sup>가 제안한 방법은 차분 이미지의 변환 계수와 이웃하는 블록(upper-row, left-column), 화소 값의 관계를 이용하여 인트라 모드 전체를 생략하는 방법을 제안하였다. 이 방법은 움직임이 적고 완만한 영상에서 효과적인 성능을 발휘하나 움직임이 빠르고 복잡한 영상에 부호화 처리 시간 감소율이 낮다. 또 다른 방법으로 Kim<sup>[13]</sup>은 이웃하는 매크로블록의 울-왜곡 값의 임계 값을 설계하여 인트라 예측 모드 전체를 생략하는 방법을 제안하였다. 이 방법은 비교적 간단한 알고리즘으로

구성되었고, 인트라 모드 예측 전체를 생략함으로써 부호화의 복잡도를 크게 줄여 주었다. 하지만 특정 영상에서 부호화율이 크게 변하는 단점이 있다.

이러한 문제점을 효율적으로 극복하기 위해 본 논문에는 움직임이 있는 영역에 대한 추적과 새롭게 등장하는 영역에 대한 검출을 통해 인트라 모드 예측을 효율적으로 생략하는 방법을 제시하였다. 본 논문의 구성은 다음과 같다. II장에서는 H.264/AVC의 울-왜곡 함수를 이용한 최적 모드 결정에 대해서 간단히 설명하고, III장에서는 알고리즘을 제안하게 된 동기와 제안한 알고리즘의 방법에 대해 설명하였다. 마지막으로 IV, V장에서는 본 논문에서 제안하는 모드 결정 방법에 의해 얻어진 결과를 분석하고 결론을 맺는다.

## II. H.264의 매크로블록 모드 결정 방법

매크로블록(Macroblock, MB)을 어떤 모드로 부호화 할지를 결정하는 것은 H.264/AVC 부호기의 핵심 방법 중 하나이다. 그림 1에 보인 흐름도처럼 울-왜곡 최적화 기법<sup>[7]</sup>을 이용하여 왜곡과 비트율면에서 가장 좋은 모드를 {SKIP, 16×16, 8×16, 16×8, SUB8×8, I4MB, I16MB}중에서 최적 모드를 선택한다.

먼저 그림 2처럼 인터 모드의 기본 블록인 매크로블

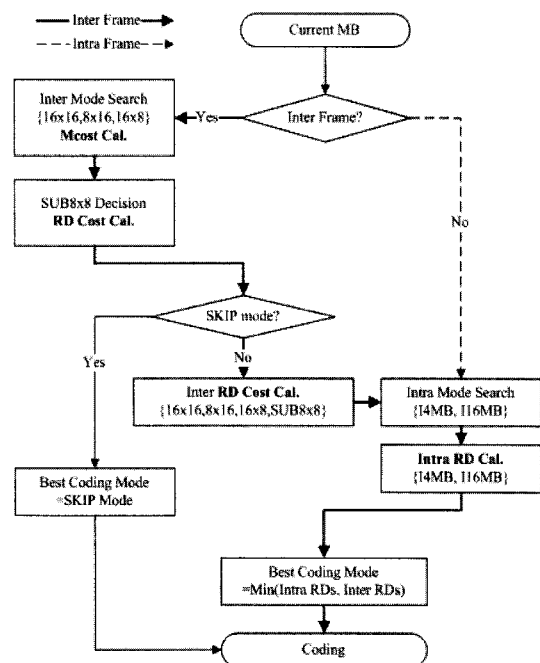


그림 1. JM S/W에서의 최적 모드 결정 방법  
Fig. 1. Best mode decision method for JM S/W.

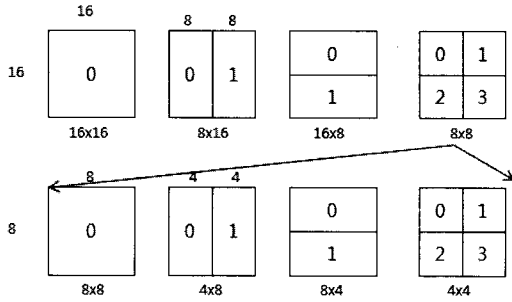


그림 2. 인터 모드의 파티션  
Fig. 2. The partition of inter mode.

록을 {16×16, 8×16, 16×8, SUB8×8} 파티션으로 분리한다. SUB8×8은 각 8×8 블록마다 독립적으로 {8×8, 8×4, 4×8, 4×4} 모드들로 더욱 세분화 될 수 있다. 분리된 파티션 블록 각각에 대하여 식(1)을 이용하여 최소값을 갖는 움직임 벡터와 참조 영상을 결정한다.

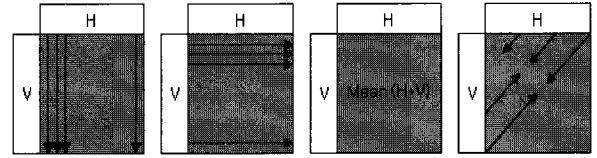
$$J_{motion} = SAD_{motion} + \lambda_{motion} \times R(MV, Residual) \quad (1)$$

$$SAD_{motion} = \sum_{x \in H, y \in V} |s(x, y) - r(x - m_x, y - m_y)| \quad (2)$$

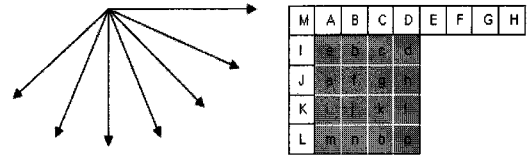
여기서  $J_{motion}$ 은 최적의 움직임 벡터와 참조 영상을 결정하기 위한 함수이며,  $SAD_{mode}$ 는 각 파티션 블록에 대해 현재 입력 영상 블록  $s$ 와 복호화 되어 프레임 메모리에 저장된 참조 영상 블록  $r$ 의 차를 식 (2)을 이용하여 구한다.  $H$ 와  $V$ 는 각각의 파티션 블록의 높이와 넓이이며,  $m_x$ 와  $m_y$ 는 움직임 벡터이다.  $R(MV, Residual)$ 은 움직임 벡터  $MV$ 와 변환된 현재 블록과 움직임 예측 블록간의 차를 부호화 하는데 필요한 비트 수이며,  $\lambda_{motion}$ 은 라그랑지안 계수로 양화자화 계수에 의존하면서 왜곡값과의 차수를 맞춰준다.

다음 순서로 조기 SKIP 모드 결정이 활성화된 경우 움직임 벡터가 예측 움직임 벡터와 같고 CBP (Coded Block Pattern)가 0인지를 확인하여 조건을 만족하면 SKIP 모드로 결정하고 SKIP 모드임을 알리는 1 비트만을 전송함으로써 비트량을 최소화 할 수 있다<sup>[17]</sup>.

다음으로 인트라 예측 모드에 대한 예측이 실행된다. 그림 3에서처럼 인트라 예측은 4×4크기의 블록을 가지고 9가지의 예측 방향을 가지는 I4MB 모드와 16×16크기의 블록에서 4가지의 방향성을 가지는 I16MB로 구분된다. I4MB와 I16MB에 대한 모드는 현재 매크로블록과 방향성에 따른 윌-왜곡 값이 최소가 되는 예측 모드를 최적 모드로 결정하게 된다. 차분치 값은 수식 (2)을



(a) I16MB and 8x8 chroma prediction modes



(b) I4MB prediction modes

그림 3. 인트라 매크로블록 예측 모드  
Fig. 3. The prediction mode of intra MB.

이용하여 계산한다.

인터 모드와 인트라 모드에 의해서 결정된 예측 모드들은 수식 (3)을 이용하여  $J_{mode}$ 값이 최소가 되는 모드를 매크로블록 부호화를 위한 최적의 모드로 결정하고 부호화한다.

$$J_{mode} = SSD_{mode} + \lambda_{mode} \times R(s, r_M, M) \quad (3)$$

$$SSD_{mode} = \sum_{x \in H, y \in V} \{s(x, y) - r_M(x, y)\}^2 \quad (4)$$

여기서  $\lambda_{mode}$ 는  $\lambda_{motion}$ 의 제곱값이다.  $H$ 와  $V$ 는 모드들에 대한 블록 사이즈이며,  $M$ 은 부호화 가능한 모든 예측 모드이다.  $R(s, r_M, M)$ 은  $M$ 에 해당하는 모드로 부호화했을 때 실제로 발생하는 비트수며,  $SSD_{mode}$ 는 현재 입력과 복호된 영상  $r_M$ 와의 차에 제곱으로 수식 (4)을 이용한다.

### III. AMVM(Adaptive MVM)을 이용한 모드 결정

H.264/AVC는 매크로블록마다 부호화 가능한 모든 모드의 예측을 통해서 윌-왜곡 값을 계산해야 하기 때문에 복잡도의 증가를 불러온다. 하지만 영상의 특성을 살펴보면 공간적인 상관도보다는 시간적 상관도가 높기 때문에, 인트라 예측 모드가 P-슬라이스와 같은 인터 프레임에서 선택될 확률은 적다. 표 1에서는 여러 시퀀스들에 대해서 bi-directional 움직임 예측을 위한 프레임 skip 수를 0으로 부호화 했을 때, 인트라 모드가 최적의 매크로블록으로 결정된 비율을 보여주고 있다. 표 1에서 보는 것처럼 인터 프레임에서 인트라 모드로 부

호화 되는 매크로블록은 10% 미만으로 매우 적다. 하지만 인트라 모드로 부호화 될 확률이 적다고해서 인트라 모드를 강제로 생략할 경우 부호화의 효율 이득이 감소하게 되는 문제가 발생한다. 그러므로 본 논문에서는 AMVM(Adaptive Motion Vector Map)을 이용하여 효율적으로 인트라 예측 모드를 생략하는 방법을 제안한다.

인트라 예측 모드를 효율적으로 생략하기 위하여 본 논문에서는 움직임 정보의 크기에 주목하였다. 그림 4

표 1. QP에 따른 인트라 모드 MB의 발생 빈도(%)  
Table 1. Occupation of intra mode MB for various QP values(%)

IPPP 영상 (100 Frame)	QP			
	24	28	32	36
News	0.51	0.35	0.22	0.3
Mobile	0.21	0.1	0.1	0.05
Coastguard	0.01	0.01	0.42	1.2
Paris	1.01	1.10	0.02	0.05
Soccer	0.81	0.86	1.01	1.15
Foreman	3.41	2.30	2.00	2.3
Bus	3.36	2.97	2.60	2.13
Stefan	5.61	3.56	2.44	2.13
Highway	8.44	6.08	3.50	2.57

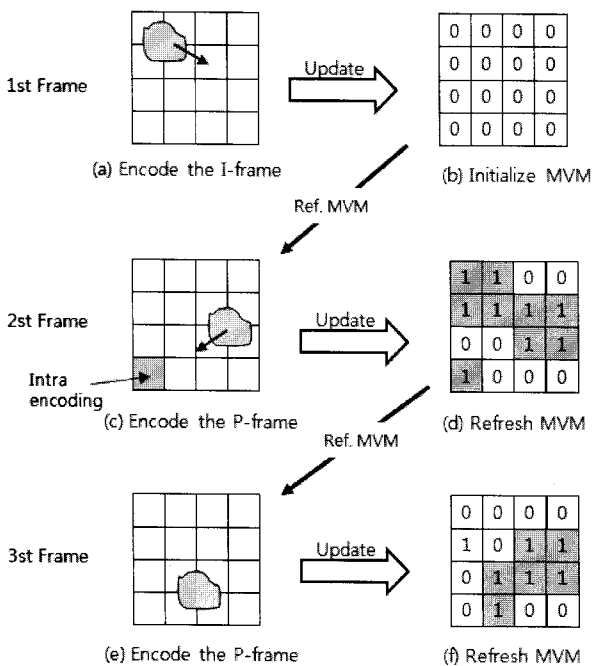


그림 4. MVM 업데이트 과정  
Fig. 4. The process for the MVM update.

와 같이 MVM은 부호화할 현재 프레임과 이전 프레임에서 움직임이 발생한 영역을 표시하는 정보를 담고 있다. 그림 4의 (a)와 (c)에서처럼 객체가 움직여 간다면 객체가 있던 자리에는 새로운 정보가 나타나는 것과 같기 때문에 인트라 모드로 부호화될 가능성이 높다. 또한 이동된 움직임 벡터의 크기가 크면 RD값이 클 수 있는데, 이 또한 인트라 모드로 부호화될 가능성이 높다. 이런 정보들을 표시하며 매 매크로블록마다 갱신되는 것이 MVM이다. 그러므로 MVM은 인트라 예측을 생략할 것인지를 판단할 때 이용할 수 있다.

MVM의 갱신 방법은 그림 4에서 보여주고 있다. 먼저 16x16모드의 움직임 탐색을 통해 최소의 SAD값을 가지는 위치의 16x16블록 크기의 움직임 벡터  $MV_{16x}$ ,  $MV_{16y}$ 를 구한다. 식 (5)처럼 두 벡터의 합( $MV_{val}$ )을 이용하여 MVM의 갱신을 결정한다.

$$MV_{val} = |MV_{16x}| + |MV_{16y}| \quad (5)$$

첫 번째 부호화 프레임은 모든 매크로블록이 인트라 모드로 부호화된다. MVM은 그림 4의 (b)처럼 모두 "0"으로 초기화한다. 두 번째 프레임 즉, 첫 번째 인터 프레임에서는 각 매크로블록 단위로 움직임 예측을 한다. 만약 현재 부호화할 매크로블록의  $MV_{val}$ 이 search range 보다 크다면 현재의 매크로블록은 움직임이 크기 때문에 인트라 모드로 부호화할 가능성이 높다. 이런 경우 현재 매크로블록의 영역에 대한 MVM을 "1"로 셋팅한다. 반면에 현재 매크로블록의  $MV_{val}$ 이 search range 보다 작다면 현재 매크로블록 영역은 움직임이 적은 영역으로 간주하여 MVM은 "0"으로 셋팅한다. 다음 프레임들부터는 현재 부호화 매크로블록의 MVM의 갱신 상태를 참조하여 부호화하게 된다. 현재 부호화할 매크로블록과 같은 영역의 참조 MVM이 "1"로 셋팅되었다면, 현재의 부호화 영역은 움직임이 많은 영역이므로 인트라 예측을 하게 된다. 반면에 MVM이 "0"인 경우에는, 움직임 적고 이전 프레임과 상관성이 매우 높은 영역이므로 인트라 예측을 생략하게 된다.

MV를 이용한 MVM만을 이용해서 부호화 하면 새로운 영상이 생기는 영역에서 인터 예측만 하는 예측 오류를 초래할 수 있다. 그림 5에서는 시퀀스 변화가 생길 때 MVM의 정보만을 이용했을 경우 오류가 발생하게 되는 예를 보여주고 있다. 원으로 표시된 영역은 부호화할 프레임에 변화가 생기는 경우이다. 이 경우 MVM만을 이용하여 현재 프레임을 부호화하게 되면,

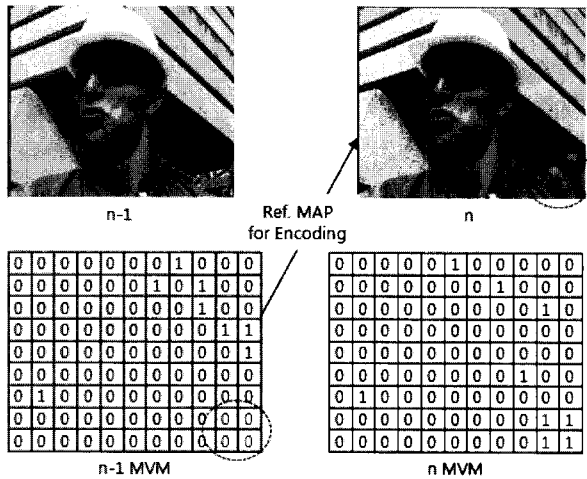


그림 5. 영상의 갑작스런 변화에 대한 고려  
Fig. 5. Consideration for abrupt change of image.

시퀀스 변화가 있는 영역에서는 인트라로 부호화해야 하는데 그렇지 못하기 때문에 문제가 발생한다. 즉, 참조될 이전 프레임에서의 MVM은 “0”으로 설정되어 있었기 때문에 인트라 예측을 생략한다. 하지만 현재 영상에서 새로운 변화가 생기는 영역에서는 인트라 모드로 결정될 확률이 높다. 그러므로 MVM에 의해서 생길 수 있는 예측 에러를 줄이기 위해 부호화된 이전 매크로블록들의 최적 모드 율-왜곡 함수 값을 사용하여 인트라 예측 모드를 생략할 것인지를 다시 한 번 예측한다.

움직임이 많고 복잡한 영역에서는 SUB8×8 모드와 I4MB 모드로 결정될 확률이 높다. 또한 SUB8×8과 I4MB는 최적 모드로 결정되었을 때 다른 모드들보다 율-왜곡 값이 평균적으로 높은 값을 갖는다. 그래서 본 논문에서는 이전에 부호화된 매크로블록들이 SUB8×8 또는 I4MB로 결정되었을 때의 율-왜곡값을 임계값으로 이용하였다. 인트라를 예측하기 위한 임계값 설계는 다음 식과 같다.

$$J_{\min\{inter\}} \leq Avg(J_{\{I4MB\}}, J_{\{SUB8 \times 8\}}) \quad (6)$$

여기서  $J_{\min\{inter\}}$ 은 현재 부호할 매크로블록의 인터 모드들 중 최소값을 가지는 율-왜곡 값을 의미하며,  $Avg(J_{\{I4MB\}}, J_{\{SUB8 \times 8\}})$ 는 이전에 부호화된 매크로블록 중 최적의 모드가 SUB8×8 또는 I4MB로 결정되었던 율-왜곡 값의 평균이다. 수식 (6)을 만족할 때 인트라 예측 모드를 생략한다.

최종적으로 본 논문에서 제안한 방법은 2단계를 거쳐 모드를 생략할 것인지를 판단한다. 먼저 복호된 이

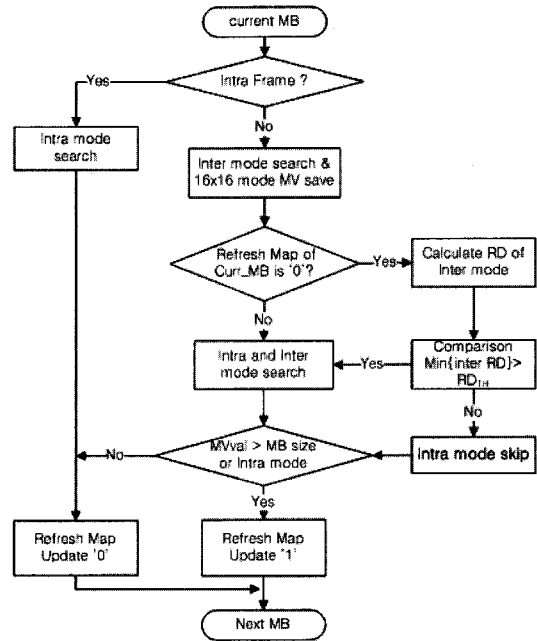


그림 6. 제안된 AMVM의 알고리즘 순서도  
Fig. 6. Flowchart of the proposed AMVM algorithm.

전 프레임에서 생성된 MVM의 정보를 이용하여 인트라 예측 모드 생략을 결정한다. 그리고 새롭게 등장하는 객체나 복잡한 화면구성의 매크로블록에 대한 보상을 위해 인터 모드에 대해서 최소 율-왜곡치를 구하여 이전 매크로블록들에 의해 계산된 SUB8×8과 I4MB의 율-왜곡치의 평균값과 비교하여 인트라 모드를 예측할 것인지에 대해 결정한다. 다음으로 다음 프레임 부호화에 사용할 MVM을 최종적으로 갱신한다.

본 논문에서는 영상의 움직임과 새로운 객체들에 대한 정보를 이용하여 인트라 모드를 생략하는 방법을 제안하였고, 이러한 방법을 AMVM이라 칭한다. 본 논문에서 제안한 AMVM 방법을 그림 6과 같이 구현하였고, 다음과 같이 모드를 결정하기 위한 과정을 요약할 수 있다.

- Step1. 인트라 프레임에서 MVM을 “0”으로 초기화
- Step2. 인터 프레임인 경우, 16×16 움직임 예측을 통한  $MV_{val}$  계산
- Step3. 인터 프레임인 경우, 현재 프레임의 MVM 갱신
  - 만약  $MV_{val} > Search\ range$ , MVM은 “1”로 셋팅
  - 만약  $MV_{val} \leq Search\ range$  이면 MVM은 “0”으로 셋팅
- Step4. 두 번째 인터 프레임부터 이전 프레임 MVM을 이용한 모드 결정

- 현재 매크로블록 위치에 해당하는 MVM값이 “1”이면 부호화 가능한 모든 모드 예측
- MVM이 “0”이고 식 (6)을 만족하면 인트라 예측 생략

Step5. 현재 부호화된 MB가 인트라 모드이면, 현재 MVM을 “1”로 갱신

Step6. Step 3로 이동

### III. 실험

제안한 AMVM을 이용한 빠른 모드 결정 알고리즘을 MPEG에서 제공한 다양한 시퀀스(CIF)를 사용하여 컴퓨터 시뮬레이션을 통해 성능 평가하였다. 부호화기의 환경 설정은 Main Profile을 기본으로 하였으며, JM12.2의 참조 소프트웨어<sup>[19]</sup>에 제안한 알고리즘을 구

현하여 비교하였다. 인코더의 부호화 파라미터 (Parameter) 설정은 다음과 같다.

- IPPP Sequence Type, 100 Frame
- QP: 20, 24, 28, 32, 36
- 4×4 Transform, on Hadamard
- Search range 16, reference frame 1
- enabled CABAC

실행 결과는 기존의 선행 연구 방법인 Lee 방법<sup>[12]</sup>과 Kim 방법<sup>[13]</sup>을 구현하여, ΔPSNR, ΔBits, ΔTime에 대해 비교 설명하였다. ΔPSNR, ΔBits, ΔTime의 계산 방법은 다음과 같다.

- ΔPSNR은 제안한 알고리즘과 참조 소프트웨어에서의 전역 모드 탐색간의 복호화 시 발생하는 화질 변화율이다.

표 2 빠른 인트라 생략 결정 알고리즘들의 실행 비교

Table 2. A performance comparison of the fast intra mode skip decision algorithms.

Contents(IPP)		QP=20			QP=28			QP=36			Average		
		ΔP	ΔB	ΔT	ΔP	ΔB	ΔT	ΔP	ΔB	ΔT	ΔP	ΔB	ΔT
News	Lee	-0.01	0.95	45.01	-0.01	0.19	40.45	0.00	0.03	38.20	0.00	0.48	41.01
	Kim	0.02	0.37	21.80	0.00	1.01	24.31	-0.04	2.39	21.57	-0.01	1.35	23.09
	AMVM	-0.01	0.68	46.49	-0.01	0.39	42.33	-0.03	0.85	38.01	-0.01	0.71	42.25
Akiyo	Lee	-0.07	-0.25	42.88	0.00	-0.01	39.41	-0.01	-0.79	38.33	-0.02	-0.39	33.23
	Kim	-0.05	-0.06	27.08	0.00	-0.40	29.76	-0.01	-0.80	28.53	-0.02	-0.32	28.90
	AMVM	0.00	-0.07	44.51	0.00	-0.09	39.89	-0.01	-0.95	33.87	-0.01	-0.33	39.12
Mobile	Lee	0.00	0.25	32.00	-0.01	0.07	13.39	-0.01	0.24	25.89	0.00	0.17	21.97
	Kim	0.01	0.05	32.73	0.00	0.06	38.49	0.00	0.37	33.72	0.00	0.04	35.96
	AMVM	0.00	0.16	34.61	0.00	0.15	28.93	-0.01	-0.30	26.92	0.00	0.04	30.05
Coastguard	Lee	0.00	0.09	21.49	0.00	-0.06	12.18	-0.03	0.14	23.06	-0.01	0.14	19.41
	Kim	0.00	0.12	35.02	0.00	0.81	34.99	-0.02	1.99	27.46	0.00	1.33	33.28
	AMVM	0.00	0.07	36.02	0.00	0.14	32.22	-0.01	0.56	25.46	0.00	0.16	31.23
Paris	Lee	-0.01	0.53	44.79	-0.02	0.20	38.11	-0.04	0.23	38.13	-0.02	0.27	39.92
	Kim	-0.01	0.08	32.67	0.01	0.22	35.17	-0.05	0.52	29.68	-0.02	0.28	33.16
	AMVM	-0.01	0.21	48.20	0.00	0.17	44.40	-0.01	-0.08	38.99	0.00	0.12	43.94
Soccer	Lee	-0.01	0.31	28.05	-0.01	0.18	29.98	-0.07	0.23	29.03	-0.02	0.16	28.64
	Kim	0.00	-0.03	32.48	0.03	0.25	31.21	-0.06	1.72	23.45	-0.01	0.59	29.56
	AMVM	0.00	0.13	31.85	0.01	0.03	28.79	-0.02	0.29	23.81	0.00	0.07	28.23
Foreman	Lee	-0.01	0.52	32.57	-0.01	0.16	28.44	-0.09	0.43	31.86	-0.03	0.30	30.46
	Kim	0.00	0.31	32.05	0.00	1.53	28.91	-0.07	1.97	24.26	-0.02	1.26	28.61
	AMVM	-0.01	0.10	37.98	0.00	0.20	35.52	-0.06	0.22	30.82	-0.02	0.17	34.93
Bus	Lee	0.01	0.24	31.33	0.00	0.14	21.11	-0.01	-0.27	26.36	0.00	0.07	25.33
	Kim	0.02	0.07	34.06	0.04	0.75	38.09	0.00	0.76	30.49	0.02	0.55	35.19
	AMVM	0.01	0.16	38.83	0.01	0.17	19.51	0.02	-0.32	16.25	0.01	0.00	21.59
Stefan	Lee	-0.01	-0.75	33.76	0.00	0.44	22.55	-0.03	0.58	31.92	-0.02	0.17	26.56
	Kim	-0.02	-1.19	33.60	-0.02	4.20	35.04	-0.06	1.98	30.46	-0.03	2.35	33.52
	AMVM	-0.01	-1.18	27.72	0.00	-0.05	21.03	0.00	0.51	29.47	0.00	-0.18	25.21
Highway	Lee	-0.01	1.00	23.07	-0.05	0.50	31.31	-0.04	-1.09	35.84	-0.03	0.75	30.05
	Kim	0.03	3.80	29.50	-0.06	2.52	24.55	-0.03	-0.52	21.41	-0.03	2.46	24.97
	AMVM	0.00	1.08	27.35	-0.05	1.64	33.48	-0.04	-1.13	30.97	-0.03	1.02	31.48

$$\Delta PSNR = PSNR_{new} - PSNR_{org} \quad (7)$$

- $\Delta Bits$ 은 부호화되어진 전체 비트율의 비교 값이다.

$$\Delta Bit = \frac{Bit_{new} - Bit_{org}}{Bit_{org}} \times 100(\%) \quad (8)$$

- $\Delta Time$ 은 전체 부호화 처리 시간을 가리키는 요소이다.

$$\Delta Time = \frac{Time_{org} - Time_{new}}{Time_{org}} \times 100(\%) \quad (9)$$

표 2는 양자화 값 20, 28, 36로 변화시켜가며 각 알고리즘에 의해 도출된 결과를 정리한 데이터이다. 또한, Average 데이터는 양자화 값 20, 24, 28, 32, 36에 대해 모두 실험한 결과의 평균을 의미한다. 그림 7, 8, 9는 비교된 알고리즘의 울-왜곡 그래프<sup>[18]</sup>이다.

그림 7은 움직임이 매우 적은 Akyio 영상의 결과 그래프이다. 결과 그래프에서 보는 것과 같이 제안한 AMVM 알고리즘은 QP가 20일 때 화질의 열화는 거의 생기지 않고 평균 비트가 감소되었다. 또한, 표 2에서 보는 것처럼 PSNR은 거의 비슷하지만, Kim 방법의 부호화 처리 시간보다 평균 10%정도 더 감소되었고, Lee 방법보다는 평균 6% 더 감소되었다.

그림 8은 완만한 움직임을 가지는 Foreman 영상의 결과를 나타낸다. 그래프에서 보는 것과 같이 제안한 알고리즘은 Foreman 영상에서는 화질의 열화는 거의

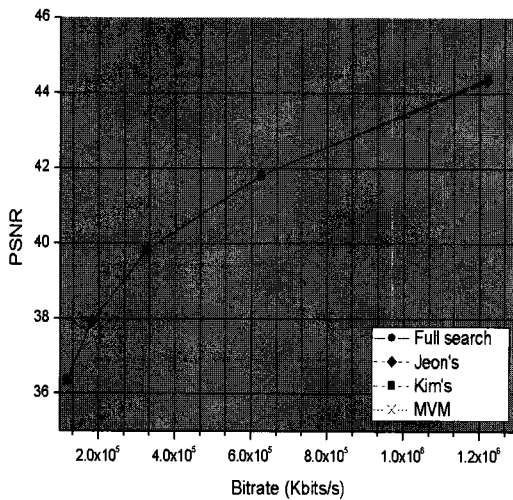


그림 7. Akyio 영상의 울-왜곡 그래프 QP= 20:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=-0.07$ ,  $\Delta Time=44.51$   
 Fig. 7. Akyio CIF of RD covers for QP= 20:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=-0.07$ ,  $\Delta Time=44.51$ .

없이 실행되었다. 하지만 Kim 방식과 비교하여 평균 비트가 감소됨을 표 2에서 확인할 수 있다. QP가 28일 때 Kim 방법은  $\Delta Bit$ 이 1.5%증가한 반면 제안된 MVM 방법은 0.2%증가되었다. 또한 Lee와 Kim 방법보다 부호화 처리 시간이 7%정도 감소되었다.

그림 9는 움직임이 빠르고 복잡한 Stefan 영상의 결과 그래프이다. 그림에서처럼 QP가 28일 때 화질 손실이 생기지 않고 비트수를 0.05% 감소시켰다. 하지만 Kim 방법은 부호율이 4%증가 되었다. Stefan 영상에서는 부호화 처리 시간에 있어서는 Lee 방법과 비슷하다.

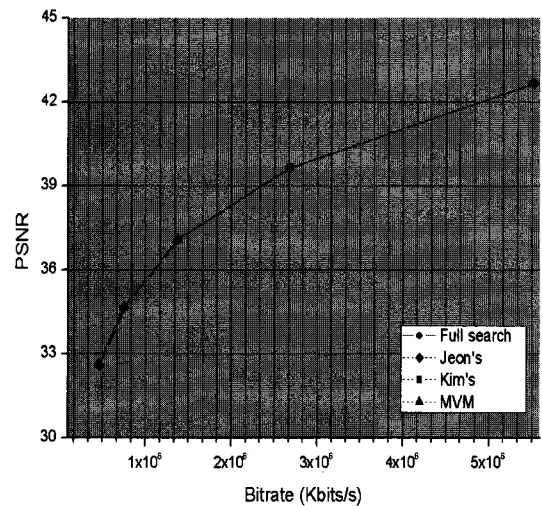


그림 8. Foreman 영상의 울-왜곡 그래프 QP= 28:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=0.2$ ,  $\Delta Time=35.52$   
 Fig. 8. Foreman CIF of RD covers for QP= 28:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=0.2$ ,  $\Delta Time=35.52$ .

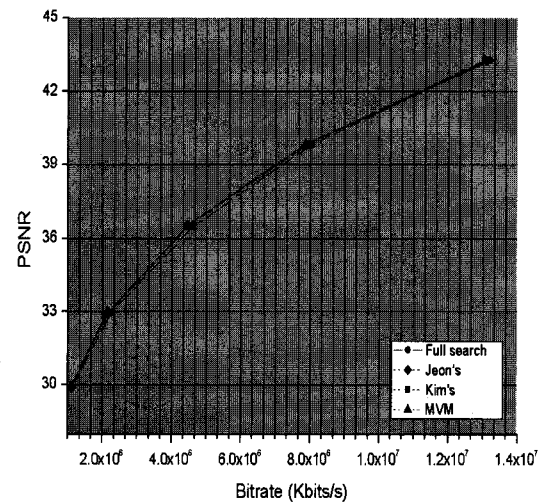


그림 9. Stefan 영상의 울-왜곡 그래프 QP= 28:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=-0.05$ ,  $\Delta Time=21.03$   
 Fig. 9. Stefan CIF of RD covers for QP= 28:  
 $\Delta PSNR=0.00$ ,  $\Delta Bits=-0.05$ ,  $\Delta Time=21.03$ .

Lee와 Kim 방법에 비해 PSNR이 가장 좋으며, Full search에 의한 부호화 방법보다 Average 비트율이 0.18% 감소되고 화질 열화는 생기지 않는다.

전체적으로 평균 부호화 성능은 움직임이 빠르고 복잡한 영상에서는 Lee 방법과 비슷한 성능을 보이고 있다. 또한 움직임이 적은 영상에 대해서는 화질 열화와 부호화 효율의 손실이 거의 없이 Lee 방식보다 부호화 시간이 5%정도 감소되었다. 반면, Kim 방법보다는 비트율이 감소되었고, 움직임이 적고 느린 영상에 대해서는 부호화 시간이 감소되었다. 제안한 알고리즘은 평균적으로 Lee와 Kim보다는 PSNR과 비트율의 변화가 Full search 모드와 비슷하게 부호화되었다. 하지만 Full search 모드의 부호화 시간보다 평균 32% 감소되었다.

#### IV. 결 론

본 논문에서는 빠른 모드 결정을 위해 이전 프레임의 적응적인 움직임 벡터 맵(AMVM)을 이용하였다. AMVM을 통해 영상의 움직임 성향과 정보를 이용하여 인트라 모드를 생략함으로써 화질 열화 및 비트율 증가를 최소로 하면서 평균 32%의 부호화 처리 시간을 감소시켰다. 제안된 알고리즘은 영상의 변화가 적은 영역에 대해서 평균적으로 부호화 처리 시간의 감소율이 크게 증가하였고, 움직임이 복잡하고 빠른 영상에 대해서는 기존의 선행 연구 되어진 알고리즘에 비해 평균적으로 화질 열화와 비트율의 증가를 최소화하고 부호화 시간을 감소시켰다.

#### 참 고 문 헌

- [1] ISO/IEC 14496-10, "Information Technology Coding of Audio-Visual Objects-Part 10: Advanced Video Coding", Dec. 2003.
- [2] T. Wiegand, G. J. Sullivan, G. Bjontegard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard", IEEE Trans. Circuit Syst. Video Technol., vol. 13, pp. 506-576, July 2003.
- [3] "Video Coding for Low Bit-rate Communications, Version 1", ITU-T Recommendation H.263, 1995
- [4] "Generic Coding of Moving Pictures and Associated Audio Information-Part 2: Video", ITU-T and ISO/IEC JTC1, ITU-T Recommendation H.262-ISO/IEC 13 818-2 (MPEG-2), 1994.
- [5] "Information Technology-coding of Audio-Visual Objects Part2: Visual Amendment 1: Visual Extensions", ISO/ICE JTC1/SG29/WG11 N3056, Dec. 1999.
- [6] D. Marpe, G. Blattermann, G. Heising, and T. Wiegand, "Further Results for CABAC Entropy Coding Scheme", Austin, TX, ITU-T SG16/Q.6 Doc. VCEG-M59, 2001.
- [7] H. Schwarz and T. Wiegand, "An Improved H.26L Coder Using Lagrangian Coder Control", ITU-T SG16/VCEG-D146, Porto Seguro, Brasil, June 2001.
- [8] Lee. J. Y, and Jeon. B. W, "Fast Mode Decision for H.264 with Variable Motion Block Sizes", in ISICIS, vol. 2869/2003, pp. 723-730, 2003.
- [9] L. Yang, K. Yu, J. Li, and S. Li, "An Effective variable Block-Size Early Termination Algorithm for H.264 Video Coding", IEEE Trans. Circuits Syst. Video Technol, vol. 15, pp. 784-788, June 2005.
- [10] T. Y. Kuo and C. H. Chan, "Fast Variable Block Size Motion Estimation for H.264 Using Likelihood and Correlation of Motion Field", IEEE Trans. Circuits Syst. Video Technol, vol. 16, pp. 1185-1195, Oct. 2006.
- [11] P. L. Tai, S. Y. Huang, C. T. Liu, and J. S. Wang "Computation-Aware Scheme for Software-Based Block Motion Estimation", IEEE Trans. Circuits Syst. Video Technol., vol. 13, pp. 901-913, Sept. 2003.
- [12] J. Y. Lee and B. W. Jeon, "Fast mode Decision for h.264", Proc. of IEEE Int'l Conf. on Multimedia and Expo(ICME), vol. 1, pp. 1131-1134, 2004.
- [13] B. Y. Kim, J. H. Kim, and C. S. Cho, "A Fast Intra Skip Detection Algorithm for H.264/AVC Video Encoding", ETRI Journal, vol. 28, pp. 721-731, Dec. 2006.
- [14] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode Decision Algorithm for Intra Prediction in H.264/AVC Video Coding", IEEE Trans. Circuits Syst. Video Technol, vol. 15, pp. 813-822, July 2005.
- [15] F. Fu, X. Lin and L. Xu, "Fast Intra Prediction Algorithm in H.264/AVC", Proc. of the 7th Int'l Conf. on Signal Processing, vol. 2, pp. 769-772, 2004.
- [16] C. C. Cheng and T. S. Chang, "Fast Three Step Intra Prediction Algorithm for 4x4 Blocks in



- H.264”, Proc. of IEEE Int’l Symp. on Circuits and Systems, vol. 2, pp. 1509-1512, 2005.
- [17] 이제윤, 전병우, “H.264 동영상 압축을 위한 고속 부호화 모드 결정 방법”, 전자공학회 논문지 제 41 권 SP편 제6호, pp. 165-173, 2004년 11월
- [18] G. Bjontegaard, “Calculation of Average PSNR Differences between RD-curves”, Austin, TX, ITU-T SG16/Q.6 Doc.VCEG-M33, 2001.
- [19] [http://iphome.hhi.de/suehring/ttml/download/old\\_jm, JM12.2.zip](http://iphome.hhi.de/suehring/ttml/download/old_jm JM12.2.zip)

---

 저 자 소 개
 

---



김 태 정(학생회원)  
 2006년 2월 충북대학교 전기전자  
 컴퓨터공학부 공학사  
 2008년 2월 충북대학교  
 전자공학과 석사  
 2008년 3월~ 현재 충북대학교  
 전자공학과 박사과정

<주관심분야 : 신호처리, 영상 압축, 멀티미디어 응용 >



고 만 근(학생회원)  
 2007년 2월 충북대학교 전기전자  
 컴퓨터공학부 공학사  
 2009년 2월 충북대학교  
 전자공학과 석사  
 2009년 3월~ 현재 충북대학교  
 전자공학과 박사과정

<주관심분야 : 신호처리, 영상 압축, 멀티미디어 응용 >



서 재 원(정회원)  
 1995년 2월 충북대학교  
 전자공학과 졸업  
 1997년 2월 광주과학기술원 정보  
 통신 공학과 석사  
 2003년 2월 광주과학기술원 정보  
 통신공학과 박사

2003년 4월~2004년 8월 LG전자 기술원  
 MM연구소 선임 연구원  
 2004년 9월~현재 충북대학교 전기전자컴퓨터  
 공학부 조교수, 컴퓨터정보통신연구소  
 비전임 연구원

<주관심분야 : 디지털 영상신호처리, 동영상 압축>