

인접 블록의 움직임 벡터 및 인터모드간 상관관계를 이용한 고속 다중 참조 프레임 선택 기법

준회원 정 찬 영*, 정회원 김 명 진*, 준회원 주 원 희*, 종신회원 홍 민 철*

Fast Multiple Reference Frame Selection Method Using Neighboring Motion Vectors and Inter-mode Correlation

Chan-Young Jeong* Associate Member, Myoung-Jin Kim* Regular Member,
Won Hee Joo* Associate Member, Min-Cheol Hong* Lifelong Member

요 약

본 논문에서는 H.264 동영상 부호화 방식의 고속 움직임 추정을 위한 인접 블록의 움직임 벡터 및 인터모드간 상관관계를 이용한 고속 다중 참조 프레임 선택 기법에 대해 제안한다. 제안된 기법에서는 상위블록과 하위 블록의 선택된 참조 프레임들의 일정한 상관관계를 갖는 통계적 특성을 이용하여 상위블록의 움직임 추정 모드에서 선택된 최적의 참조 프레임 정보를 통해 하위블록의 움직임 추정 모드에서 사용할 참조 프레임을 예측한다. 또한 인접 블록의 움직임 벡터의 통계적 특성을 이용한 가변 탐색 스텝의 고속 움직임 추정 기법에 대해 제안한다. 실험결과를 통해 5개의 참조프레임을 사용 시 제안하는 기법의 움직임 추정에 소요되는 시간은 가용한 모든 참조프레임에 대해 움직임 추정을 사용하는 방식에 비해 평균 91% 감소되었으며, PSNR 및 발생 비트율은 거의 동일한 성능을 유지함을 확인 할 수 있었다.

Key Words : H.264/AVC, Multiple Reference Frame, Motion Estimation, Video Coding

ABSTRACT

In this paper, we propose a fast multiple reference frame selection method using neighboring motion vectors and the correlation of inter-mode blocks in H.264 video coding. Using the property that an optimal reference frame has the statistical characteristics, the motion estimation of small Inter-block size is estimated from the motion estimation of the larger Inter-block size. Simulation results show that the proposal method decreased the computations about 91%. Without the sacrifice of coding performance, comparing to the H.264 multiple reference picture coding.

I. 서 론

표준 동영상 부호화 방식인 H.264는 ITU-T의 VCEG과 ISO/IEC의 MPEG에 의해 제정되었다. H.264는 새로운 부호화 방식들을 채택함으로써 이전의 방식에 비해 50%이상의 높은 압축율을 가진

다. 추가된 기법으로는 화면내 예측 방식, 1/4 축소 단위의 움직임 추정, 16X16에서 4X4까지 다양한 블록 크기의 움직임 추정 방식, 다중 참조 프레임 방식 등이 있으며 이러한 방식은 잔여 신호를 최소화함으로써 부호화 효율을 높이는 주된 역할을 한다. 그러나 H.264 동영상 표준 부호화 방식의 움직임

* 본 연구는 서울시 산학협력 사업으로 구축된 서울 미래형콘텐츠 컨버전스 클러스터 지원에 의해 수행되었음.

* 숭실대학교 정보통신전자 공학부(jesuscy, webzealer, jimini, mhong}@ssu.ac.kr)

논문번호 : KICS2009-06-261, 접수일자 : 2009년 6월 26일, 최종논문접수일자 : 2009년 7월 16일

임 추정에는 가변 블록크기의 7가지 모드를 지원하기 때문에 각 모드에 대해 전역 탐색(Full Search)을 실시할 경우, 기존의 동영상 부호화 방식 대비 약 7배의 연산량이 요구된다^{[1]~[4]}. 또한 최대 16개까지 사용하는 다중 참조프레임 방식을 채택함으로써 움직임 추정에 필요한 연산량은 참조 프레임 개수에 비례하여 더욱 증가한다. 그러므로 움직임 추정을 수행할 참조 프레임의 개수를 줄이기 위하여 가능성 높은 참조 프레임을 사전에 예측하는 기법은 불필요한 움직임 추정 과정의 연산을 줄일 수 있으므로 실시간 시스템의 개발에 필수적이라 할 수 있으며, 최근에 이와 관련된 연구들이 활발히 진행되고 있다.

다중 참조 프레임 방식에 의해 증가된 연산량을 줄이기 위한 기존의 참조 프레임 선택 기법은 H.264 동영상 표준 부호화 방식의 움직임 추정 순서를 따르지 않는 이유로 움직임 추정 과정 중 일부를 반복하도록 구현해야 하고, 데이터의 통계적 특성에 기반 하여 여러 개의 임계치를 사용했기 때문에 입력 영상 시퀀스에 따라서 그 성능이 크게 좌우될 수 있으며, 추가적인 참조 프레임 코스트 계산 과정을 필요로 하여 많은 연산량을 낭비하는 등의 문제점이 있다[5-8]. 따라서 본 논문에서는 비교적 추가적인 연산량을 요구하지 않고, 입력 영상 시퀀스의 특성에 관계없이 안정적인 성능을 갖는 고속 다중 참조 프레임 선택 기법을 제안한다.

본 논문의 구성은 다음과 같다. II장에서 제안하는 기법을 설명하고, III장에서는 제안 알고리즘의 성능을 보이며, IV장에서는 결론을 맺는다.

II. 제안 방식

2.1 참조 프레임 선택의 통계적 특성

그림 1은 JM(Joint Model) 13.2[13]에 구현되어 있는 모드별 움직임 추정 과정을 나타낸 것이다. 각 모드의 서브 블록 별로 가용한 참조 프레임 모두에 대해 움직임 추정을 수행하며, Inter16×16 모드에서 Inter4×4 모드까지의 순서로 움직임을 추정을 수행한다. H.264 동영상 표준 부호화 방식의 가변 블록 크기의 움직임 추정은 동일 매크로블록을 각 모드에 따른 서브 매크로블록으로 나누어 움직임을 추정을 반복 수행하는 것이다. 따라서 각 움직임 추정 모드에서 선택되는 최적의 참조 프레임 간에는 일정한 상관관계가 존재할 수 있고, 표 1에서 이러한 통계적 특성을 확인 할 수 있다.

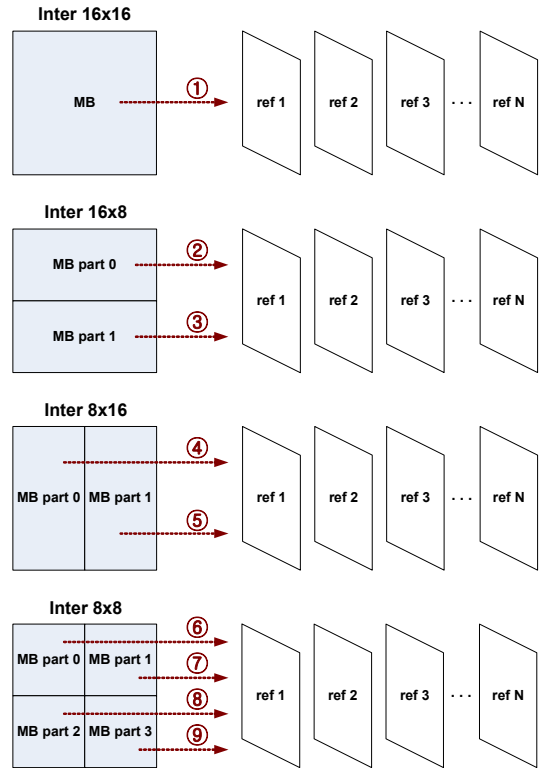


그림 1. 다중 프레임 참조 방식의 움직임 추정 과정 (JM13.2)

표 1은 참조프레임 5개를 수행한 경우의 통계적 특성을 나타낸다. Case A는 Inter16×16모드에서 선택된 최적의 참조 프레임이 Inter16×8모드나 Inter8×16모드에서도 최적의 참조 프레임으로 선택된 경우의 확률을 나타낸 것이고, Case B는 Inter16×8모드나 Inter8×16모드에서 선택된 최적의 참조 프레임이 Inter8×8모드에서도 최적의 참조 프레임으로 선택된 경우의 확률을 나타낸 것이다. 각각의 경우에 대한 수식은 식 (1)과 같다.

$$\begin{aligned}
 \text{Case A} &= P\left[\left(R_{16 \times 8}^m(i) \cup R_{8 \times 16}^n(i)\right) \mid R_{16 \times 16}^k(i)\right] \\
 \text{Case B} &= P\left[R_{8 \times 8}^k(i) \mid \left(R_{16 \times 8}^m(i) \cup R_{8 \times 16}^n(i)\right)\right] \\
 &\text{for } i = 1, \dots, N \\
 m: & 16 \times 8 \text{ mode sub-MB index} \\
 n: & 8 \times 16 \text{ mode sub-MB index} \\
 k: & 8 \times 8 \text{ mode sub-MB index}
 \end{aligned}
 \tag{1}$$

표 1에서 볼 수 있듯이 임의의 블록 크기의 움직임 추정 모드에서 선택된 최적의 참조 프레임은 그보다 작은 블록 크기의 움직임 추정 모드에서도 최적의 참조 프레임으로 선택될 확률이 평균 90%임을 확인할 수 있다. 따라서 상위 블록 크기의 움직임

표 1. 참조 프레임 선택의 통계적 특성

Sequences	Case A	Case B
Stefan	0.88	0.81
Table tennis	0.94	0.92
Foreman	0.89	0.86
Mobile	0.90	0.79
Claire	0.95	0.94
Average	0.91	0.86

추정 모드에서 계산된 가능한 참조 프레임들의 코스트 정보를 이용하여 하위 블록 크기의 움직임 추정 모드에서 사용될 참조 프레임을 선택할 수 있음을 알 수 있다.

2.2 적응적인 참조 프레임 선택

다중 참조프레임 움직임 추정 과정에서 프레임이 증가할수록 코스트의 값이 커진다면 영상내의 객체나 카메라의 움직임이 선형임을 예측 할 수 있다. 선형 움직임의 경우 참조 프레임이 증가 할수록 코스트의 값 또한 선형으로 증가 하게 되고 불필요한 연산을 계속해서 수행해야 한다.

이러한 문제점을 해결하기 위해 그림 2와 같이 Inter16X16의 코스트 값의 변화에 따라 선형 움직임 여부를 판별하여 플래그를 설정 하고 이에 따라 하위 모드에서 사용할 참조프레임의 개수를 적응적으로 조절하면 하위모드에 대한 움직임 추정에 대

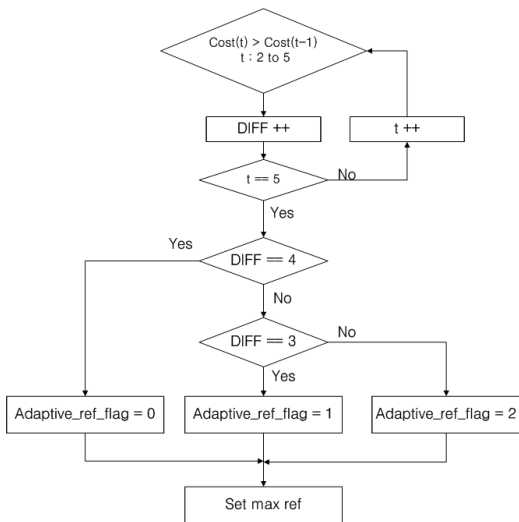


그림 2. 선형 움직임에 따른 참조프레임 개수 설정

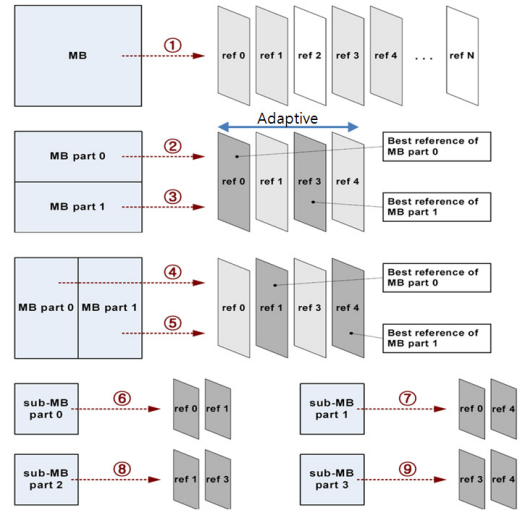


그림 3. 적응적인 고속 다중 참조 프레임 선택 기법

한 연산량을 감소시킬 수 있다. 앞에서 제시한 통계적 특성의 분석 결과와 적응적인 참조 프레임 개수 설정을 바탕으로, 각 모드별로 움직임 추정을 수행할 참조 프레임을 선택하는 방식을 제안한다.

그림 3은 제안하는 방식의 수행과정을 나타낸 것으로서, Inter16X16 모드에서 사용되어진 참조프레임이 4개 이상일 경우 코스트가 낮은 순으로 4개를 재정렬 하여 하위블록 크기의 움직임 추정 모드에서 사용할 참조 프레임으로 결정한다. 따라서 재정렬 되어진 최대 4개의 참조 프레임만이 하위 모드인 Inter8X16, Inter16X8 모드의 참조 프레임으로 사용되어진다.

이때 그림 2에서 설정한 Adaptive_ref_flag의 값에 따라 최소 2개까지 적응적으로 참조프레임 개수를 조절 하게 된다.

Inter16X8모드와 Inter8X16모드는 하나의 매크로 블록을 2개의 서브 매크로블록으로 나누어 움직임 추정을 수행하게 되므로, Inter16X8모드와 Inter8X16 모드에서 선택되는 최적의 참조 프레임은 각각 2개가 선택될 수 있다.

따라서 Inter8X8서브 블록모드에서 사용될 참조 프레임의 최대 개수는 4개가 된다. 8X8 서브 매크로블록에서 사용될 참조 프레임은 서브 매크로블록의 위치를 포함하는 Inter16X8모드와 Inter8X16모드의 최적의 참조 프레임이 된다.

2.3 탐색 영역 결정 기법

일반적인 동영상은 인접 영상과 현재 부호화 영

상의 블록 사이에 높은 상관성을 보인다^[10]. 이러한 특성으로 인해 움직임 벡터의 예측이나 움직임 추정 등이 이루어진다. 그러나 기존의 움직임 추정 방식은 미리 지정된 정방향의 움직임 탐색 영역 내에서 움직임 추정을 하게 되며 실제 최적의 움직임 벡터가 위치하는 곳이 원점에서 멀리 떨어진 곳이 아닐 때에도 불필요한 위치를 탐색하여 연산량이 증가하는 결과를 나타낸다^[9]. 이와 같이 불필요한 연산을 감소시키기 위해 부호화 블록과 인접한 블록들과의 국부 움직임 벡터의 통계적 특성을 이용하여 수평 수직 각각의 방향에 대해 움직임 추정 탐색 영역을 결정한다. 본 논문에서는 참고 문헌 [13]에 제안된 방식을 고속 참조프레임 선택 기법에 적합하도록 수정된 방식을 제안하였다.

움직임 추정 대상 블록과 인접 블록, 그리고 국부 움직임 벡터의 통계적 특성을 알아보기 위해 실시한 실험에서 인접 블록의 움직임 벡터 크기가 모두 0이거나 방향이 같은 경우, 또 인접 블록의 움직임 벡터 합이 작은 경우는 기타 경우 보다 움직임 벡터가 작게 결정되는 특징을 볼 수 있었다.

그림 4는 Container 영상에서 인접 블록의 상이한 통계적 특성에 따른 부호화 블록의 움직임 벡터가 인접 블록의 움직임 벡터의 median 값과 차이인 MVD (Motion Vector Difference)의 분포를 나타낸다.

본 논문에서는 다음과 같은 방식을 제안 하고자 한다. 인접 블록들의 위치는 그림 5와 같이 나타난다. 가변크기 블록을 사용하는 H.264 동영상 표준 부호화 방식에 따라 각 모드(mode)별 인접 블록을 결정하였으며, 부호화 블록의 위치에 따라 존재하지 않는 인접 블록이 발생하는 경우 존재하지 않는 블록의 움직임 벡터 크기 값은 주어진 탐색 영역 크기로 대체 하였다. 이와 같이 인접 블록의 움직임 벡터를 모두 정의한 후에 탐색 영역을 결정하기 위한 연산을 수행한다.

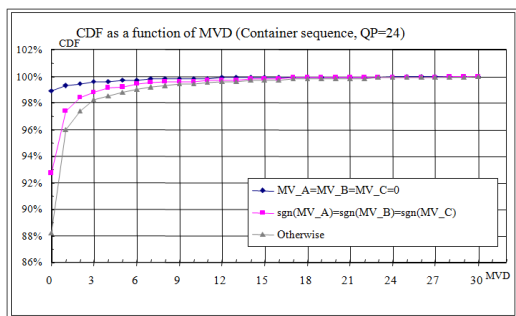


그림 4. 인접한 블록의 통계적 특성에 대한 MVD

그림 5에서 블록 E는 현재 부호화 대상 블록이며 좌상측 좌표 위치를 (i,j), 우하측 좌표 위치를 (i+block_size_h, j+block_size_v)로 표기 하였다. 이때 block_size_h는 해당 블록의 수평 방향의 크기를 나타내고 block_size_v는 블록의 수직 방향의 크기를 나타낸다.

블록 A, B, C는 블록 E와 상관관계가 가장 높은 인접 크기 블록으로 각 블록은 이미 수평 수직 방향의 움직임 벡터가 결정된 상태이다. 상기 인접 블록의 움직임 벡터를 이용하기 위해 먼저 주어진 탐색 영역 크기를 식 (2)와 같이 정의 한다.

$$SearchRange(i) \leq |w|, i = x, y \quad (2)$$

식 (2)는 부호화 조건으로 정의된 최대 움직임 탐색 영역의 값을 의미하며, x 및 y는 각각 수평 및 수직 방향을 의미한다. 그리고 인접한 세 개 블록의 움직임 벡터를 이용하여 식 (3)을 정의 할 수 있다.

$$MV_{max}(i) = \max(|MV_A(i)|, |MV_B(i)|, |MV_C(i)|) \quad (3)$$

식 (3)의 값은 움직임 벡터 추정 오류를 줄이기 위한 방안으로 인접 블록 움직임 벡터 최대치를 최소 탐색 영역 크기로 정의한다. 이 과정을 통해 부호화블록은 가능성 있는 움직임 벡터 지점을 탐색 영역 내부에 포함할 수 있게 된다.

$$\alpha(i) = |MV_A(i)| + |MV_B(i)| + |MV_C(i)| \quad (4)$$

식 (4)는 인접 영역의 상관도와 움직임 벡터의 크기를 추정할 수 있는 척도로 사용된다. 즉, $\alpha(i)$ 가 작은 경우 부호화블록은 전반적으로 적은 움직

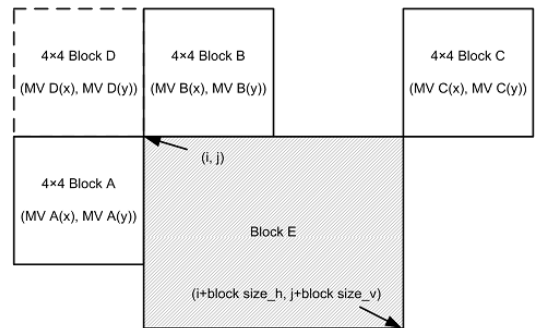


그림 5. 움직임 추정을 위한 인접 블록의 위치

임을 가지는 영역에 속하므로 현재 블록 또한 적은 움직임 벡터를 가질 확률이 높으며, 반대로 $\alpha(i)$ 가 큰 경우 많은 움직임을 가지는 영역에 속하므로 현재 부호화 대상 블록의 움직임 또한 큰 움직임 벡터일 가능성이 높다. 특히 $\alpha(i)$ 가 0이 되는 경우는 인접한 세 블록의 움직임 벡터가 모두 0이므로 현재 부호화 대상 블록 또한 0에 가까운 적은 값의 움직임 벡터를 가질 확률이 높아진다.

움직임 탐색 영역을 추정할 수 있는 또 다른 척도로 인접 블록의 움직임 벡터 크기와 방향의 동일함을 고려할 수 있다. 앞의 그림 4에서 나타난 바와 같이 부호화 블록의 움직임 벡터는 인접 블록의 움직임 벡터 방향이 모두 같은 경우 적은 값을 갖게 되며, 이와 같은 경우는 부호화 블록이 인접 블록과 동일 객체 내부에 존재할 확률이 높다. 본 논문에서는 인접 블록과의 상관성을 확인하기 위하여 움직임 벡터의 방향성과 더불어 크기의 동일함을 고려하였다. 인접 블록 움직임 벡터의 방향과 크기의 동일한 여부를 판별하기 위해 다음과 같은 식을 정의하였다.

$$Flag(i) = \begin{cases} 1, & \text{for } MV_A(i) = MV_B(i) = MV_C(i) \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

식 (4)에 의한 $\alpha(i)$ 값과 식 (5)에 의한 $Flag(i)$ 에 따라 주어진 탐색 영역을 국부 통계 특성을 반영하는 탐색 영역 중 하나로 결정한다. 이때 식 (5)에 의한 $Flag(i)$ 는 식 (4)에 $\alpha(i)$ 의한 보다 우선시 되는 조건으로 $Flag(i)$ 값이 1인 경우는 탐색 영역의 크기를 가장 작게 하고 그렇지 않은 경우 $\alpha(i)$ 에 의해 탐색 영역의 크기를 적응적으로 결정하게 된다.

$$\beta(i) = \begin{cases} \frac{\omega+4}{8}, & \text{for } Flag(i) = 1 \\ \frac{\omega+2}{4}, & \text{for } Flag(i) \neq 1 \alpha(i) \leq 2, i = x, y \\ \frac{\omega+1}{2}, & \text{for } Flag(i) \neq 1 \alpha(i) > 2 \end{cases} \quad (6)$$

식 (6)에서 구한 값은 인접 블록의 상관성에 따른 최소한의 탐색 영역 값이므로 너무 적은 탐색 영역으로 인한 움직임 추정 오류를 방지할 필요가 있다. 따라서 본 논문에서는 이를 위하여 다음 식 (7)을 이용하여 움직임 추정 오류를 방지한다.

$$\gamma(i) = \max(\beta(i), MV_{\max}(i)) \quad (7)$$

식 (7)에서 구한 $\gamma(i)$ 값이 최종 탐색 영역으로 결정되기 전에 이 값이 식 (2)의 주어진 탐색 영역 안에 있는지 식 (8)을 통해 확인한다. 이는 원래의 탐색 영역 보다 더 큰 영역에 대한 움직임 추정이 더 많은 연산을 소요하기 때문이다. 따라서 ω 와 $\gamma(i)$ 사이에서 작은 값이 선택 된다.

$$\delta(i) = \min(\omega, \gamma(i)) \quad (8)$$

x, y 방향에 대한 최종적인 탐색영역인 dx, dy 는 식 (8)에서 구한 $\delta(i)$ 값에 의해 다음과 같이 결정된다.

$$\begin{cases} -\delta(x) \leq dx \leq \delta(x) \\ -\delta(y) \leq dy \leq \delta(y) \end{cases} \quad (9)$$

이와 같이 결정된 영역 dx, dy 내에서 움직임 추정을 수행하게 된다.

III. 실험 결과

본 실험에서는 QCIF와 CIF 크기의 Akiyo, Foreman, Stefan, Mobile 영상을 4가지의 양자화 파라미터(QP)에 대해 실험 하였다. 부호화기의 환경은 JM13.2에서 Baseline Profile을 사용하였으며, RD Optimization은 사용하지 않았다. 움직임 벡터 탐색영역(Search range)은 QCIF 동영상에 대해 16×16 을 CIF 동영상에 대해 32×32 를 사용하였다. 움직임 벡터 추정을 위한 최대 참조 프레임 수는 최대 5로 설정하였으며, 전체 부호화 프레임 개수는 100프레임 (10 frames/sec)으로 설정 하였다. 제안한 알고리즘의 성능 비교를 위하여 참고문헌 [11] 및 참고문헌 [13], 그리고 참조프레임 1장 및 5장을 사용한 전역 탐색(Full Search) 기법과 비교 하였다. 성능 비교를 위해 100프레임을 부호화시 인터 예측 과정의 소요시간 TMET(Total Motion Estimation Time)(sec)과 PSNR(Peak Signal to Noise Ratio)을 사용하였다. PSNR은 $M \times N$ 크기의 8bits 영상에 대하여 원 영상 대비 복호화 된 영상의 잡음 비율을 나타낸 것으로 다음 식 (10)으로 정의될 수 없다.

$$PSNR = 10 \log \frac{MN \times 255^2}{\|f - \hat{f}\|^2} \quad (10)$$

식 (10) 에서 $\|\cdot\|$ 는 유클리드 노름(Euclidean norm)을 나타내고, f 는 원영상, \hat{f} 은 복호화 된 영상을 나타낸다. 실험 결과에서 MR1과 MR5는 참조 프레임을 1장 사용하였을 때 및 참조프레임을 5장 사용하였을 때의 실험결과이며, 기타 방식을 최대 참조 영상이 5장인 경우의 결과이다.

표 2 와 표 3을 통해 움직임 추정에 소요되는 연산량이 기존 방식 (MR5) 대비 평균 90%이상 감소되었음을 알 수 있다. 이는 참조영상을 1장 사용하는 MR1의 연산량 보다도 약 50%이상 절감된 값이다. 더불어, 비트율 함수에 대한 PSNR 성능 비교를 그림 6~7에 도시하였다. 실험 결과에서 알 수 있듯이 동일한 비트율(Bitrates)에서 PSNR은 모든 실험 영상에 대해 평균 0.05dB의 화질 저하만을 보였다. 비트율(Bitrate)의 경우 Akiyo, Foreman 과 같은 움직임이 비교적 적은 영상의 경우 기존 방식 (MR5) 대비 평균 0.1kbps/s의 증가만을 보였고 Stefan, Mobile과 같이 움직임이 비교적 많은 영상의 경우 평균 6.5kbps/s 정도 증가하였다.

상기 실험 결과에서 제안 방식은 H.264/AVC 부

표 2. TMET(sec) 성능 비교(QCIF)

Sequence	QP	16	24	32	40
Akiyo	MR1	41.28	40.72	34.70	27.51
	MR5	223.85	213.50	177.79	137.29
	ref[12]	106.69	108.75	99.80	81.63
	ref[11]	61.66	61.54	52.72	42.24
	Prop.	22.61	24.57	19.38	18.02
Foreman	MR1	89.67	77.77	59.16	43.55
	MR5	479.32	417.77	321.63	239.04
	ref[12]	192.40	177.23	150.40	128.16
	ref[11]	115.55	102.91	82.16	63.88
	Prop.	31.68	31.83	29.30	27.50
Stefan	MR1	94.92	87.44	75.37	61.32
	MR5	513.05	475.02	416.72	346.04
	ref[12]	214.98	205.44	184.81	162.64
	ref[11]	129.71	119.20	101.68	83.72
	Prop.	30.80	30.93	29.73	28.38
Mobile	MR1	112.00	106.07	91.92	72.37
	MR5	572.19	543.72	470.56	373.23
	ref[12]	239.50	232.06	211.49	182.19
	ref[11]	142.27	133.99	117.27	96.86
	Prop.	30.95	31.26	32.30	31.82

표 3. TMET(sec) 성능 비교(CIF)

Sequence	QP	16	24	32	40
Akiyo	MR1	536.67	446.57	301.80	218.53
	MR5	2869.73	2345.26	1548.57	1107.98
	ref[12]	1477.73	1330.66	894.02	637.05
	ref[11]	652.04	560.03	416.16	335.07
	Prop.	183.06	179.79	121.31	99.37
Foreman	MR1	1150.63	880.92	587.01	387.73
	MR5	6237.34	4894.35	3316.05	2192.16
	ref[12]	2489.92	2133.51	1698.63	1266.55
	ref[11]	1279.31	996.67	697.74	500.21
	Prop.	315.43	293.63	244.95	194.15
Stefan	MR1	1274.43	1105.07	827.77	582.24
	MR5	6792.81	5955.99	4708.72	3565.03
	ref[12]	2855.64	2572.92	2093.25	1675.56
	ref[11]	1434.04	1264.07	958.87	699.84
	Prop.	302.65	280.14	239.67	208.54
Mobile	MR1	1471.66	1339.44	1074.28	772.68
	MR5	7531.97	6862.81	5479.48	4007.13
	ref[12]	3077.04	2893.94	2505.21	2034.96
	ref[11]	1625.80	1478.79	1196.98	877.71
	Prop.	289.35	284.83	274.13	237.18

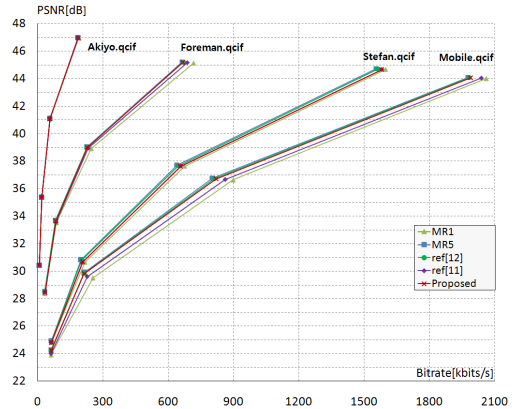


그림 6. 비트율 대비 PSNR 성능 비교 (QCIF)

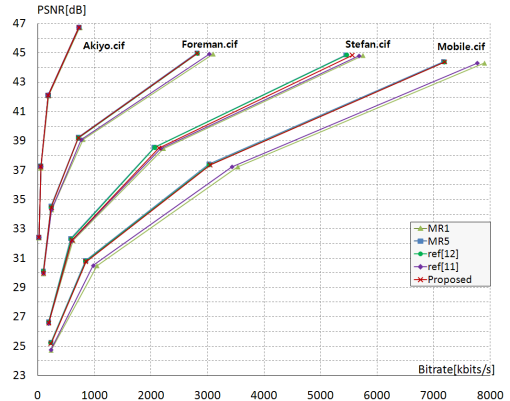


그림 7. 비트율 대비 PSNR 성능 비교 (CIF)

호화의 성능 저하 없이 다중 참조 영상을 이용한 움직임 추정 방식의 연산량을 현저히 저하할 수 있음을 확인할 수 있었다.

IV. 결 론

본 논문에서는 H.264 동영상 부호화 방식의 고속 움직임 추정을 위한 인접 블록의 움직임 벡터 및 인터모드간 상관관계를 이용한 고속 다중 참조 프레임 선택 기법에 대해 제안 하였다. 제안 방식이 각각의 움직임 추정 모드에서 선택된 최적의 참조 프레임간의 상관관계에 기반 하여, 상위 블록 크기의 움직임 추정 모드에 대해 미리 계산된 가용한 참조 프레임들의 코스트를 이용하여 다음 하위 블록 크기의 움직임 추정 모드에서 사용할 참조 프레임을 선택하였기 때문에 많은 연산량을 줄일 수 있었다. 실험 결과로부터 제안 방식을 사용하는 경우, 평균 0.05dB의 PSNR 감소와 평균 1%의 비트량이 증가하는 반면, 평균 90%이상의 연산량 절감이 있음을 확인할 수 있었다.

참 고 문 헌

[1] *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264|ISO/IEC 14496-10 AVC) JVT-G050*, Geneva, Switzerland, 23-27 May, 2003.

[2] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthar, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuit and Systems for Video Technology*, vol. 13, pp.560-576, July, 2003.

[3] Iain E.G. Richardson, *H.264 and MPEG-4 Video Compression*, Wiley, 2003.

[4] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," *IEEE Circuits and Systems Magazine*, vol. 4, pp. 7-28, first quarter. 2004.

[5] T. Wiegand, X. Zhang, and B. Girod, "Motion-compensating long-term memory prediction," *Proc. IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Oct. 1997.

[6] T. Wiegand, X. Zhang, and B. Girod, "Block-based hybrid video coding using motion-compensated long-term memory prediction," *Proc. Picture Coding Symp.* Berlin, Germany, Sept. 1997.

[7] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Trans. Circuit and Systems for Video Technology*, vol. 9, pp. 70-84, Feb. 1999.

[8] T. Wiegand and B. Girod, *Multi-Frame Motion-Compensated Prediction for Video Transmission*. Norwell, MA: Kluwer, 2001.

[9] Y. W. Huang, B. Y. Hsieh, T. C. Wang, S. Y. Chen, S. Y. Ma, C. F. Shen, and L. G. Chen, "Analysis and reduction of reference frames for motion estimation in MPEG-4 AVC/JVT/H.264," *Proc. Int. Conf. Multimedia and Expo (ICME)*, vol. 2, pp. II - 809-812, July. 2003.

[10] C. W. Ting, L. M. Po, and C. H. Cheung, "Center-biased frame selection algorithms for fast multi-frame motion estimation in H.264," *Proc. Int. Conf. Neural Networks and Signal Processing*, vol. 2, pp. 1258-1261, Dec. 2003.

[11] A. Chang, O. C. Au, Y. M. Yeung "A novel approach to fast multi-frame selection for H.264 video coding," *Proc. Int. Symp. Circuits and Systems (ISCAS '03)*, vol. 2, pp. II-704-II-707, May. 2003.

[12] H. J. Li, C. T. Hsu, M. J. Chen, "Fast multiple reference frame selection method for motion estimation in JVT/H.264," *Proc. IEEE Asia-Pacific Conf. Circuit and Systems*, vol. 1, pp. 605-608, Dec. 2004.

[13] Y. Shen, C. Hunag, L. Yu, D. Zhang, J. Li, "Fast multiframe motion estimation algorithm in H.264," *Proc. Int. Conf. Signal Processing (ICSP '04)*, vol. 2, pp. 1187-1190, Sept. 2004.

[14] Yu-Wen Huang, Bing-Yu Hsieh, Shao-Yi Chien, Shyh-Yih Ma, Liang-Gee Chen, "Analysis and Complexity Reduction of Multiple Reference Frames Motion Estimation in H.264/AVC," *IEEE Trans. Circuit and Systems for Video Technology*, vol. 16, No. 4, pp.507-522, April 2006.

정 찬 영 (Chan-Young Jeong)

준회원



2007년 2월 숭실대학교 정보통신전자공학부 학사
2007년 3월~현재 숭실대학교 정보통신전자공학부 석사과정
<관심분야> H.264 동영상부호화 및 영상신호처리

주 원 희 (Won Hee Joo)

준회원



2008년 2월 수원대학교 정보통신전자공학과 학사
2008년 3월~현재 숭실대학교 정보통신전자공학부 석사과정
<관심분야> 동영상부호화 및 Rate control, 영상신호처리

김 명 진 (Myoung-Jin Kim)

정회원



2002년 2월 한국방송대학교 컴퓨터과학과 학사
2005년 2월 한국방송대학교 정보과학과 석사
2005년 3월~현재 숭실대학교 정보통신전자 공학부 박사과정
<관심분야> 동영상 부호화 및

영상통신, Rate Control, 영상 복원 및 Enhancement

홍 민 철 (Min-Cheol Hong)

종신회원



1988년 2월 연세대학교 전자공학과 학사
1990년 8월 연세대학교 전자공학과 석사
1990년 7월~1991년 8월 LG 정보통신 연구원
1997년 9월 Northwestern

University, 전기및컴퓨터공학과 박사
1997년 9월~1998년 8월 Northwestern University, Research Fellow
1998년 9월~2000년 2월 LG 전자, 선임연구원
2000년 3월~현재 숭실대학교, 부교수
<관심분야> 영상 복원 및 enhancement, 정지 및 동영상 필터링, 동영상 부호화, 비선형 필터링, Image Deconvolution, Rate Control