

저 복잡도의 H.264/AVC를 위한 고속 인트라 모드 선택 기법

정회원 김 종 호*, 정 제 창**

Fast Intra-mode Selection Algorithm for H.264/AVC Video Coding with Low Complexity

Jongho Kim*, Jechang Jeong** *Regular Members*

요 약

H.264/AVC는 부호화 성능을 향상시키기 위해 많은 최신의 기술들을 채택하였지만, 그 결과 부호화기의 복잡도가 매우 증가하였다. 특히 화면내(intra) 예측 부호화에서는 주변블록과의 방향성 상관도에 따라 4×4 휘도블록은 9개, 16×16 휘도블록과 8×8 색차블록은 각각 4개의 부호화 모드가 있는데, 이들 중 최적의 모드는 윗-왜곡 최적화(RDO) 기법에 의해 선택한다. 따라서 하나의 매크로블록을 부호화하기 위해서는 592번의 RDO 계산을 수행해야 한다. 본 논문에서는 간단한 구조의 방향성 마스크와 주변블록의 모드정보를 이용하여 RDO 계산의 대상이 되는 모드의 개수를 줄임으로써 부호화 복잡도를 줄이는 알고리즘을 제안한다. 제안한 방법은 RDO 계산을 최대 132회로 줄이고 부호화 시간을 전역탐색(exhaustive search)에 비해 70%까지 줄이면서 PSNR 및 비트율 변화가 거의 없음을 실험을 통해 보이고 있다.

Key Words : intra-mode selection, intra prediction, H.264/AVC, RD optimization.

ABSTRACT

The emerging H.264/AVC video coding standard improves coding performance significantly by adopting many advanced techniques. This is achieved at the expense of great increasing encoder complexity. Specifically the intra prediction using RDO examines all possible combinations of coding modes, which depend on spatial directional correlation with adjacent blocks. For 4×4 luma blocks, there are 9 modes, and for 16×16 luma and 8×8 chroma blocks, there are 4 modes, respectively. Therefore the number of mode combinations for each macroblock is 592. This paper presents a method to reduce the RDO complexity using simple directional masks and neighboring modes. According to the proposed method, we reduce the number of mode combinations to 132 at the most. Experimental results show the proposed method reduces the encoding time up to 70% with negligible loss of PSNR and bitrate increase compared to the H.264/AVC exhaustive search.

I. 서 론

최근 다양한 네트워크 환경에서의 효율적인 멀티

미디어 서비스를 위하여 고성능의 비디오 압축기술에 대한 요구가 어느 때보다 커지게 되었다. 이에 MPEG과 ITU-T는 압축성을 극대화할 수 있는

* 한양대학교 전자통신전공학과 영상통신 및 신호처리 연구실(angel@ece.hanyang.ac.kr),

** 한양대학교 전자전기컴퓨터공학부(jjeong@ece.hanyang.ac.kr)

논문번호 : KICS2005-09-371, 접수일자 : 2005년 9월 13일

※본 연구는 한국과학재단 목적기초연구(R01-2003-000-11627-0) 지원으로 수행되었습니다.

표준을 공동으로 제정하기로 하고, 그 결과물로 최근 MPEG-4 part 10 AVC (Advanced Video Coding) 및 H.264의 표준안을 완성하였다¹¹. 또한 H.264/AVC 표준이 DMB (Digital Multimedia Broadcasting) 및 DVB-H에 포함됨에 따라 향후 다양한 응용제품의 개발에 있어 더욱 중요한 표준으로 자리잡을 전망이다.

H.264/AVC는 압축성능에 있어 MPEG-2보다 약 40%~50%, MPEG-4 ASP (Advanced Simple Profile)에 비해서는 약 30%~40% 향상된 것으로 보고되고 있다². 이러한 압축성능을 달성하기 위하여 H.264/AVC는 다양한 최신 기술들을 포함하고 있으며, 주요한 것들을 살펴보면 다음과 같다.

- 정수 기반의 변환부호화 및 정밀한 양자화
- 가변 블록크기 기반 및 1/4화소 단위의 화면간 예측 부호화
- 다중 참조 프레임 기반 움직임 예측/보상
- 향상된 루프내(in-loop) 디블록킹 필터
- 향상된 성능의 엔트로피 부호화(CAVLC, CABAC)
- 공간영역에서의 인트라 예측 부호화

H.264/AVC는 주어진 비트율에서 최상의 영상 품질을 얻기 위해 율-왜곡 최적화 (RDO: Rate-Distortion Optimization) 기법을 이용하여 다양한 부호화 모드 중 최적의 모드를 선택한다. 율-왜곡 최적화 과정을 위해서 부호화기에서는 가능한 모든 부호화 모드의 조합에 따라 율-왜곡 관계를 조사하고, 그 중 최적모드를 선택하는 방법을 사용한다. 이로 인해 부호화기의 복잡도가 극단적으로 증가하여 모바일 디바이스와 같이 저복잡도를 요구하는 응용분야에서 H.264/AVC를 사용하기 어렵게 하는 원인이 되어왔다.

H.264/AVC 부호화기의 복잡도를 줄이기 위하여 고속 움직임 예측/보상 방법, 고속 인트라 모드 선택 기법, 고속 인트라 모드 선택 기법 등에 대한 연구가 활발히 진행되고 있다. 고속 움직임 예측/보상 방법¹³, ⁴¹은 기존의 비디오 부호화 표준 등에서 꾸준히 연구되고 있는 분야이지만, 고속 인트라/인트라 모드 선택 기법은 H.264/AVC의 등장에 의해 새롭게 연구되고 있는 분야이다⁵, ⁶. 최적 인트라/인트라 모드를 선택하기 위해 매우 많은 부호화 모드의 조합에 대해 율-왜곡 계산을 해야 하므로 부호화기 전체의 복잡도를 감소시켜 다양한 응용분야에 적용하기 위해서는 고속 인트라/인트라 모드 선택 기법이 매우

중요하다. 그 중에서 본 논문에서는 인트라 모드 선택에서의 복잡도 감소 방법을 제안한다. 전체 비디오 시퀀스 관점에서 보면 화면내 예측 부호화되는 프레임의 개수는 적지만 블록당 부호화 모드가 매우 많고 화면내 프레임 뿐만 아니라 화면간 예측 프레임에서도 인트라 모드가 수행되기 때문에 고속 인트라 모드 선택 기법은 전체 부호화기의 성능을 향상시키는데 있어 중요한 부분이다.

F. Pan et al.은 Sobel 마스크와 에지 히스토그램을 이용한 전처리 과정에 의해서 블록의 방향성을 검출하는 방법을 제안하였다⁶. 그러나 F. Pan et al.의 방법은 RDO 계산량은 줄이지만 H.264/AVC에서 정의하고 있는 예측방향 이외의 방향까지 정밀하게 찾음으로써 다시 한정된 개수의 방향으로 정리해야 하는 등의 과정이 상당히 복잡한 단점이 있다. 본 논문에서는 인트라 모드 선택의 복잡도를 줄이기 위해 매우 간단한 방향성 마스크를 이용하여 부호화 대상 블록의 방향성을 구하고, 더욱 정확한 예측 방향을 구하기 위해 주변 블록의 부호화 모드를 이용하는 방법을 제안한다. 제안한 방법은 RDO 계산량을 줄여 부호화 시간이 70%이상 감소하면서 발생 비트량 및 PSNR은 거의 유지되는 것으로 실험결과를 통해 보이고 있다.

본 논문의 구성은 다음과 같다. II장에서는 H.264/AVC에서의 인트라 예측 부호화 방법 및 필요한 계산량을 기술한다. III장에서는 제안하는 고속 인트라 예측 부호화 방법을 기술하고 제안한 방법에 의한 계산량 감소정도를 보인다. IV장에서는 실험결과 및 기존의 방법과의 비교결과를 보이고 V장에서 결론을 맺는다.

II. H.264/AVC에서의 인트라 예측 부호화

기존의 부호화 표준에서는 인트라 예측 부호화하기 위해 변환영역에서 DC 계수 또는 일부 AC 계수의 예측 부호화만을 수행하였으나, 영상의 특성상 변환영역의 계수보다는 공간영역의 화소들간의 상관도가 높기 때문에 H.264/AVC의 인트라 예측 부호화에서는 공간영역에서 화소간 상관성을 최대한 제거하여 부호화 효율을 높이는 방법을 도입했다. 또한 각 블록마다 국부적 방향성을 고려하여 효율적인 인트라 예측 부호화를 수행하기 위해 주변 블록으로부터 방향성 예측을 수행하여 부호화 대상이 되는 잔여신호(residual)를 최대한 줄이는데, 블록의 특성에 따라 4×4 휘도블록, 16×16 휘도블록, 8×8

색차블록에 대해서 각각 예측 부호화를 수행한다.

2.1 H.264/AVC의 인트라 모드 예측 방법

H.264/AVC는 주변 블록과의 방향성 상관도를 이용하여 인트라 예측 부호화를 수행한다. 이때 주변블록은 복호된 블록을 이용하며 현재블록의 위와 왼쪽 블록의 경계화소를 이용한다. 또한 블록의 형태는 휘도신호의 경우 4×4 또는 16×16 단위로 예측 부호화를 수행하고 색차신호의 경우에는 8×8 단위로 수행한다. 최적 예측 모드를 결정한 후 예측신호와 현재신호와의 차이, 즉 잔여신호를 변환, 양자화 및 엔트로피 부호화를 거쳐 화면내 부호화 과정을 수행한다. 4×4 휘도블록은 주로 영상의 복잡한 부분에서 선택되기 때문에 다양한 방향성을 반영하기 위해서 총 9가지의 예측방향이 있고, 16×16 휘도블록은 주로 영상의 평탄한 부분에서 선택되기 때문에 총 4가지의 예측방향이 존재한다. 또한 8×8 색차블록은 16×16 휘도블록과 같이 총 4가지의 예측방향이 있고 두 색차블록(C_r, C_b)을 같은 모드로 예측 부호화한다.

그림 1은 4×4 휘도블록에 대한 인트라 예측 방향을 나타낸다. 그림 1의 (a)에서 a~p는 현재블록의 화소를 나타내고 A~M은 복호된 주변블록의 화소를 나타낸다. 그림 1의 (b)에서 화살표는 각 예측 방향을 나타내고 숫자는 각 예측모드를 나타낸다. DC 예측(Mode 2)은 방향성을 갖지 않는 예측모드로서, 현재블록의 모든 화소를 A~M의 평균값으로 예측을 한다. Mode 0은 세로방향의 예측모드로서, 예를 들면 a, e, i, m은 A로 예측하고 b, f, j, n은 B로 예측하는 방식이다. 마찬가지로 Mode 1은 가로방향의 예측모드로서, 예를 들면 a, b, c, d를 I로 예측하는 방식이다. Mode 3~Mode 8에서는 현재블록의 화소들을 A~M의 방향성 가중 평균 (directional weighted average) 방식으로 예측한다. 예를 들면, 예측모드가 Mode 4일 경우 현재블록의 d는 round(B/4 + C/2 + D/4)로 예측한다.

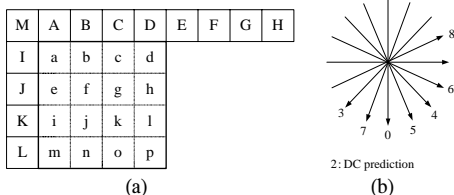


그림 1. 4×4 휘도블록에 대한 인트라 부호화 (a) 4×4 휘도블록 및 주변블록 화소의 인덱스 (b) 인트라 부호화 방향
Fig. 1. Intra prediction for a 4×4 luma block (a) indices of pixels for a 4×4 luma block and adjacent blocks (b) directions of intra prediction

16×16 휘도블록은 비교적 평탄한 영역에서 선택되기 때문에 적은 수의 예측모드가 정의되어 있다. 즉 vertical (Mode 0), horizontal (Mode 1), DC (Mode 2) plane (Mode 3)의 4가지 예측모드가 있다. 8×8 색차블록에 대해서도 마찬가지로 4개의 예측모드가 있지만, 표 1과 같이 모드가 발생하는 확률의 차이에 의해 순서가 다르다. 또한 색차블록의 경우 DC 모드는 4개의 4×4 블록으로 분할하여 그 분할된 블록의 위치에 따라 미리 정의된 주변 화소를 이용하여 예측 부호화를 수행한다. 4:2:0 포맷의 영상에서 하나의 매크로블록은 C_r과 C_b의 두 가지 8×8 색차블록을 포함하는데 H.264/AVC에서는 두 블록을 같은 예측방향으로 부호화한다. 이와 같이 각 블록의 형태 및 특성에 따라 다양한 예측모드가 있고 이들 중 최적의 부호화 효율을 갖는 모드를 선택하기 위해서 H.264/AVC에서는 울-왜곡 최적화 (RDO) 기법을 이용하여 모드를 선택한다.

표 1. 16×16 휘도블록과 8×8 색차블록에 대한 인트라 모드의 차이
Table 1. Comparison of Intra-modes for a 16×16 luma block and an 8×8 chroma block

모드번호	예측방향	
	16×16 휘도블록	8×8 색차블록
0	vertical	DC
1	horizontal	horizontal
2	DC	vertical
3	plane	plane

2.2 RDO 기반의 부호화 모드 선택 방법

하나의 매크로블록에 대해서 인트라 예측 부호화하기 위한 RDO 과정은 다음과 같다^{7, 8}.

초기화: 인트라 블록에 대한 RDO 계산을 위해 Lagrangian 승수 λ_{MODE}를 식 (1)과 같이 초기화 한다⁹.

$$\lambda_{MODE} = 0.85 \cdot 2^{(QP-12)/3} \quad (1)$$

여기서 QP는 매크로블록에 대한 양자화 파라미터이다.

Step 1: 4×4 휘도블록의 9가지 모드 중 식 (2)의 Cost를 최소로 하는 모드를 선택한다.

$$Cost = D + \lambda_{MODE} \cdot R \quad (2)$$

여기서 D와 R은 각각 주어진 QP에서의 왜곡과 비트율을 나타내고, MODE는 4×4 휘도블록의 9개 부호화 모드 중 하나를 의미한다. 왜곡은 원본 4×4 휘도블록 화소와 복호된 4×4 휘도블록 화소간의

SSD (Sum of Squared Difference)로 구하고 비트 율은 해당 모드와 4×4 휘도블록의 변환계수에 대한 비트를 포함한다. 이 과정을 16개의 4×4 휘도블록에 대해서 반복한다.

Step 2: 16×16 휘도블록에 대해서는 4가지 모드 중 SATD (Sum of Transformed Difference)를 최소화 하는 모드를 선택한다. 이때 변환은 하다마드 (Hadamard) 변환을 수행한다.

Step 3: 8×8 색차블록의 4가지 모드 중 식 (3)의 $Cost_C$ 를 최소화 하는 모드를 선택한다.

$$Cost_C = D + \lambda_{MODE} \cdot R \quad (3)$$

이때 왜곡 D 는 두 개의 원본 8×8 색차블록(C_r, C_b)과 복호된 8×8 색차블록간의 SSD로 구하고, 비트 율 R 은 8×8 색차블록의 변환계수에 대한 비트를 포함(해당 모드에 대한 비트는 제외)한다.

Step 4: Step 1에서 구해진 4×4 휘도블록에 해당하는 모드와 Step 2에서 구해진 16×16 휘도블록에 해당하는 모드의 $Cost$ 를 비교하여 작은 값에 해당하는 모드를 휘도블록에 대한 최적 모드로 선택한다.

N_8 을 8×8 색차블록의 예측모드 수, N_{16} 을 16×16 휘도블록의 예측모드 수, N_4 를 4×4 휘도블록의 예측모드의 수라고 할 때, 앞의 과정에 따라 인트라 예측 부호화를 수행하게 되면 RDO 계산을 위한 부호화 모드의 조합이 총 $N_8 \times (N_{16} + 16 \times N_4) = 592$ 임을 알 수 있다. 즉 하나의 매크로블록에 대해서 부호화 및 복호화 과정에 의한 RDO 계산을 592번 수행해야 하므로 부호화 복잡도가 극단적으로 증가하게 된다. 따라서 본 논문에서는 이러한 RDO 계

산회수를 줄이는 방법을 제안함으로써 고속 인트라 예측 부호화를 실현하는 방법을 제안한다.

III. 제안하는 H.264/AVC 고속 인트라 예측 부호화 기법

H.264/AVC의 인트라 예측 부호화에서 현재블록의 화소는 주변블록의 복호된 경계화소로부터 예측되고 최적 부호화 모드를 선택하기 위해서 RDO 기법을 이용한다. F. Pan et al.의 방법^[6]은 RDO 계산량을 감소시켜 고속 인트라 부호화를 수행하나, 이 방법은 후보모드를 얻기 위해 상당히 복잡한 전처리 과정을 거치므로 RDO 계산량 감소의 효과를 충분히 얻지 못하는 단점이 있다. 따라서 본 논문에서는 먼저 매우 간단한 방향성 마스크를 이용하여 블록의 방향성을 구하고, 모드 선택의 정확성을 높이기 위하여 주변 블록의 부호화 모드를 이용하여 후보 모드를 선택한다. 이러한 방법에 의해 RDO 계산의 대상이 되는 모드의 개수를 줄임으로써 복잡도를 감소시키는 알고리즘을 제안한다.

3.1 4×4 휘도블록의 예측모드 결정

일반적으로 각 블록의 방향성 상관도는 에지의 방향과 대체로 일치하고, 각 블록의 예측모드는 주변블록의 예측모드와 상관도가 상당히 높다. 따라서 4×4 휘도블록에 대해서 먼저 그림 2와 같은 간단한 구조의 방향성 마스크를 이용하여 하나의 후보모드를 얻고, 그림 3과 같이 주변블록(위 블록, 좌측블록)의 예측모드를 추가 후보모드로 선택한다.

그림 2에서 검은 점은 방향성 검출을 위한 마스크를 의미하고 화살표는 해당하는 모드의 방향을 나타낸다. H.264/AVC의 인트라 예측 부호화에서는

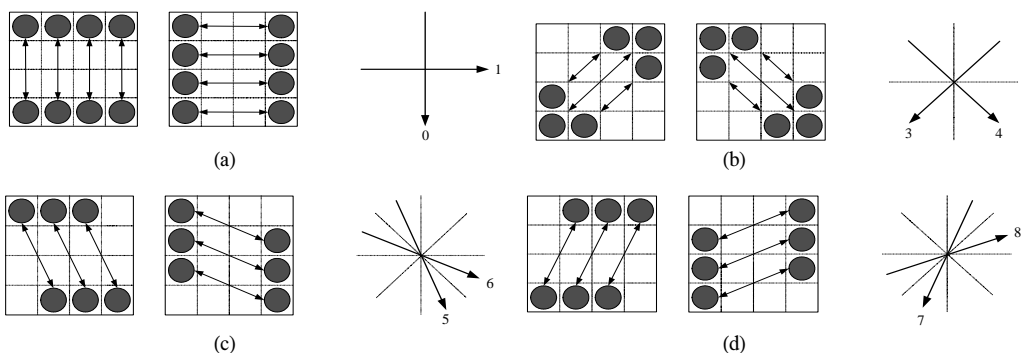


그림 2. 4×4 휘도블록의 방향성 검출을 위한 마스크 (a) vertical 및 horizontal 방향 (b) diagonal down left 및 diagonal down right 방향 (c) vertical right 및 horizontal down 방향 (d) vertical left 및 horizontal up 방향
 Fig. 2. Proposed directional masks for a 4×4 luma block (a) vertical and horizontal directions (b) diagonal down left and diagonal down right directions (c) vertical right and horizontal down directions (d) vertical left and horizontal up directions

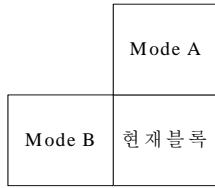


그림 3. 후보모드 선택을 위한 주변블록 및 모드
Fig. 3. Adjacent blocks and their modes for selecting candidates

DC 모드를 제외한 8개의 방향성 모드를 정의하고 있기 때문에 Sobel 마스크와 같이 정밀한 에지 방향을 검출하지 않고 각 방향에 해당하는 8개의 매우 간단하고 곱셈연산이 필요 없는 마스크를 이용하여 방향정보를 구한다. 그림 2의 마스크를 이용하여 구체적으로 식 (4)~(11)중 최소의 Diff를 갖는 모드를 후보모드로 선택한다.

$$Diff = |a-m| + |b-n| + |c-o| + |d-p|, \text{ for Mode 0} \quad (4)$$

$$Diff = |a-d| + |e-h| + |i-l| + |m-p|, \text{ for Mode 1} \quad (5)$$

$$Diff = |c-i| + 2 \cdot |d-m| + |h-n|, \text{ for Mode 3} \quad (6)$$

$$Diff = |b-l| + 2 \cdot |a-p| + |e-o|, \text{ for Mode 4} \quad (7)$$

$$Diff = |a-n| + 2 \cdot |b-o| + |c-p|, \text{ for Mode 5} \quad (8)$$

$$Diff = |a-h| + 2 \cdot |e-l| + |i-p|, \text{ for Mode 6} \quad (9)$$

$$Diff = |b-m| + 2 \cdot |c-n| + |d-o|, \text{ for Mode 7} \quad (10)$$

$$Diff = |e-d| + 2 \cdot |i-h| + |m-l|, \text{ for Mode 8} \quad (11)$$

식 (4)~(11)에서 $a \sim p$ 는 그림 1의 (a)에서 나타난 인덱스이다. 최소 Diff를 갖는 방향을 에지 또는 상관도가 최대인 방향으로 판단하여 식 (4)~(11)중에서 최소 Diff를 갖는 모드를 RDO를 위한 후보모

드로 선택한다.

실제 H.264/AVC의 인트라 예측을 수행할 때는 블록내부의 방향성이 아니라 주변블록과 현재블록의 방향성 상관관계를 고려하기 때문에 방향성 마스크를 이용하여 구한 후보모드는 다소 부정확할 가능성이 있다. 이러한 부정확성을 보완하기 위해 그림 3과 같이 주변블록의 모드, 즉 Mode A와 Mode B를 추가적인 후보모드로 이용한다. 이 경우에 두 모드가 같은 경우에는 하나의 추가 후보모드를, 두 모드가 다를 경우에는 두 개의 추가 후보모드를 얻을 수 있다.

또한 정확한 DC 모드의 선택을 위하여 식 (12)와 같이 S 를 정의한다.

$$S = \sum_{i=0}^{15} |avg - p_i| \quad (12)$$

여기서 $avg = \left(\sum_{i=0}^{15} p_i + 8 \right) \gg 4$ 이고, p_i 는 현재블록의 화소값이다.

S 가 문턱값 T_1 보다 작을 경우, 평탄한 블록이라 판단하여 후보모드에 DC를 포함한다. 반면 S 가 문턱값 T_1 보다 클 경우, 식 (4)~(11)에서 최소 Diff와 두 번째 최소 Diff를 갖는 모드를 후보모드로 선택한다. 따라서 S 가 T_1 보다 작을 경우에는 {최소 Diff 모드, Mode A, Mode B, DC 모드}를 후보모드로 선택하고, S 가 T_1 보다 클 경우에는 {최소 Diff 모드, 두 번째 최소 Diff 모드, Mode A, Mode B}를 후보모드로 선택한다. 각 경우 총 4개의 모드를 후보모드로 선택하지만, 각 후보모드가 같을 경우에는 더 적은 수의 후보모드만을 선택하므로 통상 4개보다 적은 수의 후보모드에 대해서 RDO 과정을 수행하게 된다. 그림 4에 4×4 휘도블록에 대한 후보모드 선택과정에 대한 순서도를 나타내었다.

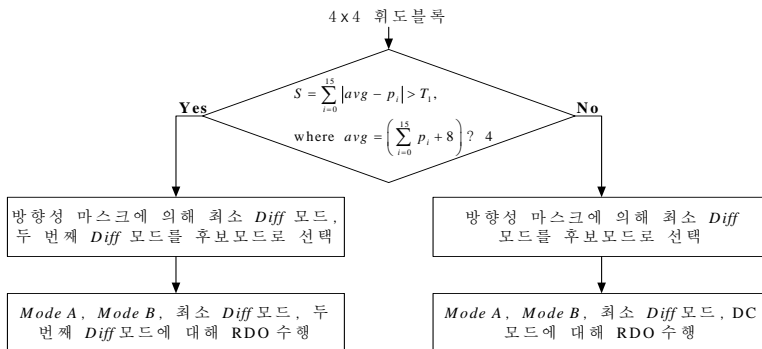


그림 4. 제안하는 4×4 휘도블록에 대한 인트라 모드 선택 알고리즘
Fig. 4. Flow chart of the proposed algorithm for 4×4 luma blocks

3.2 16×16 휘도블록과 8×8 색차블록의 예측 모드 결정

H.264/AVC 부호화 과정에서 16×16 휘도블록은 대체로 평탄한 영역에서 선택된다. 또한 4:2:0 포맷의 영상에서 색차신호는 다운 샘플링되기 때문에 대체로 평탄하다. 따라서 이들 블록들은 4×4 휘도블록과는 달리 4가지의 방향성 예측만을 수행하게 된다. 평탄한 영역에서는 현재블록의 부호화 모드와 주변블록의 부호화 모드의 상관도가 매우 크기 때문에 본 논문에서는 16×16 휘도블록과 8×8 색차블록의 후보모드를 결정하는데 있어 주변 블록의 모드를 이용하는 방법을 제안한다. 먼저 16×16 휘도블록에 대해서 예측모드를 결정하는 방법은 다음과 같다.

Step 1: 주변블록의 크기가 모두 16×16인지 검사한다. 모두 16×16이면 Step 2를, 그렇지 않으면 Step 4를 수행한다.

Step 2: 주변블록의 모드가 같은지를 검사한다. 두 모드가 같으면 Step 3을 수행한다. 두 모드가 같지 않으면 Mode A(위 블록의 모드)와 Mode B(왼쪽블록의 모드)를 후보모드로 선택하고, 이들 중 SATD가 작은 모드를 최적모드로 선택한다.

Step 3: 두 모드가 모두 DC 모드일 경우, Step 4를 수행한다. DC 모드가 아닐 경우, 주변블록의 모드(두 모드가 같기 때문에 한 모드임)와 DC 모드를 후보모드로 선택하고, 이들 중 SATD가 작은 모드를 최적 모드로 선택한다.

Step 4: 현재블록의 경계화소와 주변블록의 경계화소간의 차이를 다음과 같이 정의한다.

$$\Delta_V = \sum_{i=0}^{15} |u_i - p_i|, \Delta_H = \sum_{i=0}^{15} |l_i - q_i| \quad (13)$$

여기서 u_i 와 p_i 는 각각 위쪽블록의 아래 경계화소와 현재블록의 위 경계화소를 나타내고, l_i 와 q_i 는 각각 왼쪽블록의 오른쪽 경계화소와 현재블록의 왼쪽 경계화소를 나타낸다. Δ_V 와 Δ_H 의 의미를 그림 5에 나타내었다. 그림 6과 같이 문턱값 $T_2 > 0$ 이고, (i) $|\Delta_V - \Delta_H| < 2T_2$ 일 경우에는 DC 모드와 plane 모드를 후보모드로 선택하고, (ii) $(\Delta_V - \Delta_H) > T_2$ 일 경우에는 가로방향으로 상관도가 높기 때문에 horizontal 모드와 DC 모드를 후보모드로 선택하고, (iii) $(\Delta_V - \Delta_H) < -T_2$ 일 경우에는 세로방향으로 상관도가 높기 때문에 vertical 모드와 DC 모드를 후보모드로 선택한다. 각 경우에 대해서 모두 후보모드는 2개이고 이들 중 SATD가 작은 모드를 최적 모드로 선택한다.

16×16 휘도블록의 모드선택 과정을 순서도로 나타내면 그림 7과 같다. 8×8 색차블록에 대해서는

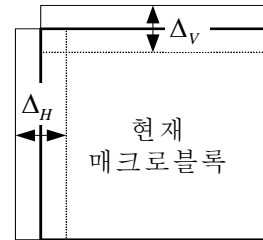


그림 5. Δ_V와 Δ_H의 의미
Fig. 5. Meaning of Δ_V and Δ_H

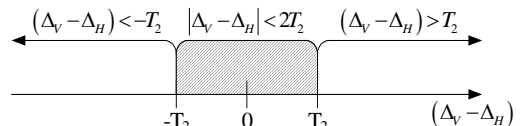


그림 6. Δ_V와 Δ_H를 이용한 후보모드의 선택($T_2 > 0$)
Fig. 6. Selection of candidates using Δ_V와 Δ_H의($T_2 > 0$)

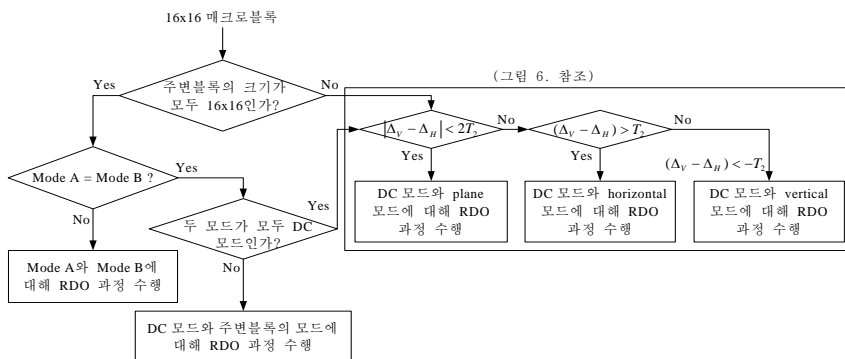


그림 7. 제안하는 16×16 휘도블록 및 8×8 색차블록에 대한 인트라 모드 선택 알고리즘
Fig. 7. Flow chart of the proposed algorithm for 16×16 luma blocks and 8×8 chroma blocks

주변블록의 크기가 모두 8×8로 같기 때문에 Step 1의 과정을 생략하고 Step 2~Step 4의 과정을 16×16 휘도블록의 경우와 동일하게 수행한다. 단, 8×8 색차블록에서 각 단계별 후보모드에 의해 최적모드를 선택할 때는 RDO 과정을 거쳐 선택한다.

3.3 각 블록형태에 따른 RDO 계산량의 비교

표 2에 H.264/AVC에서 인트라 예측 부호화를 수행하기 위한 RDO 계산량과 제안하는 방법에 의한 RDO 계산량을 비교하였다. H.264/AVC에서는 한 매크로블록당 592번의 RDO 계산을 수행하는데 비해 제안하는 방법은 최대 132번의 RDO 계산만을 수행한다.

표 2. 제안하는 방법에 의한 RDO 계산량의 감소
Table 2. Reduction of complexity using the proposed method

블록형태	H.264/AVC 방법	제안한 방법
4×4 휘도블록	9	최대 4
16×16 휘도블록	4	2
8×8 색차블록	4	2

IV. 실험 및 결과

4.1 실험방법

제안한 방법의 성능평가를 위하여 JM8.4를 이용하였고 테스트 영상과 테스트 방법은 JVT에서 권고한 바를 따랐다^{10, 11}. 테스트 영상은 QCIF(176×144) 크기의 *Akiyo*, *Foreman*, *Carphone*, *Hall Monitor*, *Silent*, *News*, *Container*, *Coastguard*를 사용하였다. 양자화 파라미터를 28, 32, 40으로 달리 적용하여 결과를 비교하였고 IPPP..., 즉 첫 번째 프레임만 화면내 부호화하고 나머지는 모두 화면간 예측 부호화한 형태와 All-I, 즉 모든 프레임을 화면내 부호화한 형태에 대해서 성능을 평가하였다. 구체적인 실험환경은 다음과 같다.

- CPU: Intel Pentium IV 2.0GHz
- 메모리: 512MB
- 운영체제: MS Windows XP
- 음-왜곡 최적화: 사용
- 움직임 추정: ±32 범위, 전역탐색(full search)
- 움직임 벡터 해상도: 1/4화소 단위
- 참조영상: 1 프레임

PSNR을 비교하기 위하여 휘도신호의 PSNR과

색차신호의 PSNR을 식 (14)와 같이 평균한 값을 사용하였다.

$$PSNR_{avg} = 10 \log_{10} \left(\frac{255^2}{MSE_{avg}} \right) \quad (14)$$

여기서 $MSE_{avg} = \frac{4 \cdot MSE_Y + MSE_U + MSE_V}{6}$ 에 의해서 구하고, MSE_Y 는 휘도블록, MSE_U 와 MSE_V 는 색차신호의 평균지승오차(mean squared error)를 각각 나타낸다.

실험결과와의 비교는 세 가지 관점에서 수행하였는데, H.264/AVC 부호화기와 제안된 방법에 의한 부호화 시간차이인 $\Delta Time$, PSNR 차이인 $\Delta PSNR$, 그리고 비트율 차이인 ΔBit 가 그것이다. $\Delta PSNR$ 은 식 (14)의 차이로 계산하였고, $\Delta Time$ 은 식 (15)와 같은 방식으로 계산하였다.

$$\Delta Time = \frac{T_p - T_{JM}}{T_{JM}} \times 100 (\%) \quad (15)$$

여기서 T_p 는 제안된 방법에 의한 부호화 시간이고 T_{JM} 은 H.264/AVC의 방법에 의한 부호화 시간을 의미한다.

4.2 실험결과 및 고찰

화면간 부호화 화면에서도 인트라 부호화되는 매크로블록이 존재하기 때문에 본 논문에서는 IPPP... 형태의 영상에 대해서도 성능평가를 수행하였다. 실험에 사용된 테스트 영상은 총 300 프레임을 대상으로 하였고 100 프레임마다 한 프레임씩 화면내 부호화 프레임을 삽입하였다. 또한 전술한 바와 같이 4×4 휘도블록의 모드결정을 위한 문턱값 T_1 은 32, 16×16 휘도블록과 8×8 색차블록의 모드결정을 위한 문턱값 T_2 는 8로 설정하였다. IPPP... 형태의 영상에 대한 실험결과는 표 3에 정리하였다.

표 3에서 $\Delta Time$, $\Delta PSNR$ 항목의 음수는 JM 결과에 비해 부호화 시간과 PSNR이 감소했음을 나타낸다. 실험결과에서 알 수 있는 바와 같이 제안한 방법에 의해 부호화 시간은 약 35%까지 줄어들었으나 PSNR과 비트율은 미미한 변화만을 나타낼 수 있다. 또한 F. Pan et al.의 방법과의 비교결과도 표 3에 보이고 있는데, 결과에서 보는 바와 같이 제안하는 알고리즘이 우수함을 알 수 있다. 이는 주로 F. Pan et al.의 방법이 RDO 계산량을 줄이기 위해서 상당히 복잡한 전처리 과정을 수행하고 주변블록의 모드정보를 이용하지 않는 점에 기인한다.

표 3. IPPP... 형태의 시퀀스에 대한 실험 결과
Table 3. Results for IPPP... type sequences

실험영상 (QCIF)	[6]의 방법			제안한 방법								
				QP=28			QP=32			QP=40		
	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)
Akiyo	-22.72	-0.053	1.17	-28.05	-0.013	0.16	-32.04	-0.008	0.21	-34.51	-0.002	0.31
Foreman	-21.80	-0.077	1.54	-30.52	-0.016	0.21	-33.76	-0.010	0.28	-35.73	-0.001	0.34
Carphone	-20.51	-0.082	1.80	-30.49	-0.018	0.17	-33.72	-0.012	0.31	-35.48	-0.008	0.33
Hall Monitor	-23.38	-0.065	1.23	-32.53	-0.018	0.19	-34.61	-0.013	0.26	-36.09	-0.005	0.37
Silent	-21.94	-0.033	0.86	-31.72	-0.014	0.18	-32.83	-0.008	0.25	-35.26	-0.004	0.29
News	-23.11	-0.067	1.23	-33.27	-0.017	0.19	-35.31	-0.011	0.25	-36.83	-0.003	0.30
Container	-20.78	-0.081	1.80	-29.51	-0.018	0.16	-32.55	-0.011	0.22	-34.61	-0.003	0.27
Coastguard	-21.20	-0.017	0.50	-29.83	-0.010	0.13	-32.82	-0.007	0.18	-35.02	-0.001	0.23

표 4. All-I 형태의 시퀀스에 대한 실험 결과
Table 4. Results for All-I type sequences

실험영상 (QCIF)	[6]의 방법			제안한 방법								
				QP=28			QP=32			QP=40		
	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)	ΔTime (%)	ΔPSNR (dB)	ΔBit (%)
Akiyo	-64.32	-0.210	3.21	-61.84	-0.10	0.47	-66.49	-0.12	0.92	-68.12	-0.07	1.62
Foreman	-65.38	-0.285	4.44	-62.27	-0.08	0.14	-67.03	-0.06	1.06	-69.83	-0.03	1.79
Carphone	-65.93	-0.276	3.91	-64.73	-0.15	0.92	-66.93	-0.13	1.52	-68.27	-0.09	1.72
Hall Monitor	-66.51	-0.252	3.73	-65.25	-0.13	0.36	-68.72	-0.11	1.91	-70.06	-0.10	2.94
Silent	-65.17	-0.183	3.54	-62.16	-0.12	0.51	-66.48	-0.08	1.35	-69.36	-0.05	2.63
News	-55.34	-0.294	3.90	-60.38	-0.11	0.86	-65.56	-0.10	1.26	-68.28	-0.06	1.85
Container	-56.36	-0.234	3.70	-61.06	-0.09	0.90	-66.07	-0.08	1.07	-69.21	-0.05	1.71
Coastguard	-55.03	-0.106	2.36	-59.72	-0.08	0.73	-64.81	-0.06	0.82	-68.03	-0.03	1.57

표 4에 All-I 형태의 영상에 대한 실험결과를 정리하였다. All-I 형태의 실험에서는 총 300 프레임을 모두 화면내 부호화하였고 다른 부호화 파라미터 및 문턱값은 IPPP... 형태의 영상에서와 동일하게 설정하였다.

표 4에서 보는 바와 같이 모든 프레임이 화면내 부호화되었기 때문에 부호화 시간은 약 70%까지 감소하였다. 반면, IPPP... 형태에서는 인트라 모드가 선택되는 매크로블록의 개수가 제한되어 있기 때문에 PSNR 감소 및 비트율 증가 정도가 매우 미미했지만 All-I 형태에서는 모든 프레임에서 인트라 모드가 선택되기 때문에 PSNR 감소 및 비트율 증가 정도가 IPPP... 형태보다는 약간 커지게 된다. 그러나 부호화 속도의 감소율에 비하면 상당히 적은 정도이고 실제 응용에서 All-I 형태의 응용은 매우 제한적이기 때문에 본 논문의 결과는 충분히 용인될 수 있는 정도이다. All-I 형태의 영상에 대해서도 F. Pan et al.의 방법과 비교하였고, IPPP... 형태에서와 같은 이유로 제안된 방법이 더 우수함을 알

수 있다. 그림 8과 그림 9는 News 시퀀스의 IPPP... 형태와 All-I 형태에 대해 제안한 방법과 F. Pan et al.의 R-D 곡선을 나타내고 있다. 그림의 R-D 곡선에서도 표 3과 표 4의 결과와 마찬가지로 제안한 방법이 F. Pan et al.의 방법에 비해 우수함을 확인할 수 있다.

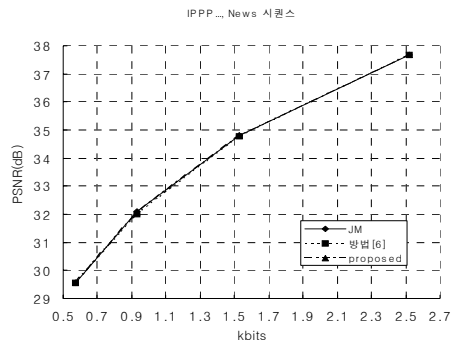


그림 8. IPPP... 형태의 시퀀스에 대한 R-D 곡선(News, QCIF)
Fig. 8. R-D curve for IPPP... type sequence (News, QCIF)

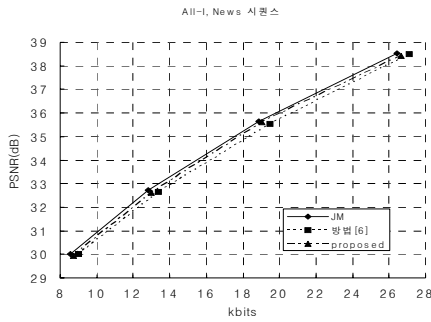


그림 9. All-I 형태의 시퀀스에 대한 R-D 곡선(News, QCIF)
Fig. 9. R-D curve for All-I type sequence (News, QCIF)

V. 결론

본 논문에서는 간단한 형태의 방향성 마스크 및 주변블록의 부호화 모드의 상관성을 이용하여 H.264/AVC를 위한 고속 인트라 부호화 방법을 제안하였다. H.264/AVC는 뛰어난 압축성능으로 인해 최근 모바일 기기 및 차량용 기기에서의 응용으로 각광 받고 있어 복잡도 감소 및 최적화 기술이 매우 중요하다. 제안한 고속 인트라 부호화 방법은 각 블록의 형태 및 특성에 따라 적은 수의 후보모드를 선택하여 RDO 계산량을 크게 감소시킨다. 또한 제안하는 방향성 검출 마스크는 그 구조가 매우 간단하고 곱셈연산이 필요 없어 후보모드를 선택하는데 추가적인 계산량을 요구하지 않는다. 제안한 알고리즘은 모든 프레임은 화면내 부호화할 경우 부호화 시간을 70%까지 감소시키면서 미미한 PSNR 감소 및 비트율 증가를 보였다. 또한 화면간 예측 프레임을 삽입한 실험에서도 부호화 시간을 35%까지 감소하면서 PSNR 및 비트율은 변화가 거의 없었다.

참고 문헌

[1] ITU-T Rec. H.264 | ISO/IEC 14496-10, "Information Technology-Coding of Audio-Visual Objects, Part 10: Advanced Video Coding," 2002.

[2] Thomas Wiegand, Gary Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol.13, no.7, pp.560-576, Jul. 2003.

[3] Zibo Chen, Peng Zhou, and Yun He, "Fast

integer pel and fractional pel motion estimation for JVT," JVT-F017, 6th JVT Meeting, Awaji, Japan, Dec. 2002.

[4] Bing-Yu Hsieh, Yu-Wen Huang, Tu-Chih Wang, Shao-Yi Chien, and Liang-Gee Chen, "Fast motion estimation algorithm for H.264/MPEG-4 AVC by using multiple reference frame skipping criteria," in *Proc. SPIE, VCIP 2003*, vol.5150, pp.1551-1560.

[5] K. P. Lim, S. Wu, D. J. Wu, S. Rahardja, X. Lin, F. Pan, and Z. G. Li, "Fast inter mode decision," JVT-I020, 9th JVT Meeting, SanDiego, USA, Sept. 2003.

[6] 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 and Syst. for Video Technol.*, vol.15, no.7, pp.813-822, Jul. 2005.

[7] Changsung Kim, Hsuan-Huei Shih, and C.-C. Jay Kuo, "Multistage mode decision for intra prediction in H.264 codec," in *Proc. SPIE, VCIP 2004*, vol.5308, pp.355-363.

[8] Keng-Pang Lim, Gary Sullivan, and Thomas Wiegand, "Text description of joint model reference encoding methods and decoding concealment methods," JVT-N046, 14th JVT Meeting, HongKong, China, Jan. 2005.

[9] Thomas Stockhammer, Dimitros Kontopodis, and Thomas Wiegand, "Rate-distortion optimization for JVT/H.26L video coding in packet loss environment," in *Proc. Int. Packet Video Workshop*. 2002.

[10] Gary Sullivan, "Recommended simulation common conditions for H.26L coding efficiency experiments on low resolution progressive scan source material," VCEG-N81, 14th Meeting, SantaBarbara, USA, Sept. 2001.

[11] JVT Test Model AdHoc Group, "Evaluation sheet for motion estimation," Draft ver.4, Feb. 2003.

김 종 호 (Jongho Kim)

정회원



1998년 2월 한양대학교 전자통신공학과 졸업
2000년 2월 한양대학교 전자통신공학과 석사
2000년 3월~현재 한양대학교 전자통신전파공학과 박사과정
<관심분야> 영상처리, 영상압축,

비디오 코딩

정 제 창 (Jechang Jeong)

정회원



1980년 2월 서울대학교 전자공학과 졸업
1982년 2월 KAIST 전기 및 전자공학과 석사
1990년 미국 미시간대학 전기공학과 공학박사
1980년~1986년 KBS 기술연구소 연구원(디지털 TV 및 뉴미디어 연구)
1990년~1991년 미국 미시간대학 전기공학과 연구교수 (영상 및 신호처리 연구)
1991년~1995년 삼성전자 멀티미디어 연구소(MPEG, HDTV, 멀티미디어 연구)
1995년~현재 한양대학교 전자전기컴퓨터공학부 교수 (영상통신 및 신호처리 연구실)
1998년 11월 이달의 과학기술자상 수상
1998년 12월 정보통신부장관상 수상
<관심분야> 신호처리, 영상처리, 영상압축