

H.264에서 화소 변화량을 이용한 빠른 인트라 예측

이탁기[†], 김성민^{**}, 신광무^{***}, 정기동^{****}

요 약

H.264는 가장 최근에 제정된 동영상 압축 표준으로 다양한 기법 등을 도입하여 기존의 표준들에 비해 동일한 화질을 유지하면서도 높은 압축 효율을 보여준다. 하지만 이러한 기법들은 처리과정이 복잡해, 계산 과정을 간소화시킨 효율적인 기법들이 요구된다. 따라서 본 논문에서는 새롭게 도입된 기법 중에서 복잡한 처리가 요구되는 인트라 예측의 효율적인 처리를 위한 2단계의 빠른 인트라 예측 방법을 제안한다. 1단계에서는 매크로블록 내 작은 블록들(4x4, 8x8, 12x12 크기)의 경계 부분의 화소 변화량을 조사하고, 이를 통해서 매크로블록의 평탄 여부를 판단하여 인트라 예측을 위한 블록 크기를 빠르게 선택한다. 2단계에서는 매크로블록 내부의 대표성을 띄는 화소들을 이용하여 1단계에서 선택된 블록 크기의 여러 모드 중에서 최종 모드를 빠르게 결정한다. 제안한 인트라 예측 기법의 성능측정을 위해 다양한 테스트 동영상으로 화질, 비트율 및 처리시간을 확인한 결과, 관련기법 및 표준과 비교해서 동일한 화질과 비트율을 유지하면서 표준과 비교하여 41.5%, 관련기법과 비교하여 24.7%의 인트라 예측 처리 시간을 감소시켰다.

Fast Intra Prediction using Pixel Variation in H.264

Tak-gi Lee[†], Sung-Min Kim^{**}, Kwang-Mu Sin^{***}, Ki-Dong Chung^{****}

ABSTRACT

H.264/AVC is the newest video coding standard of ITU-T VCEG and the ISO/IEC MPEG, offering a significant performance improvement over previous video coding standards. However, the computational complexity of H.264/AVC is drastically increased because of new technologies such as intra prediction, variable block size, quarter-pels motion estimation/compensation, etc. In this paper, we propose a fast intra prediction scheme which has two step processing. The first step is a fast block size decision which can be calculated only in one block without considering all cases of 4x4 block and 16x16 block. The complexity of the intra prediction can be reduced by using boundary difference values of macroblock. After selecting the block size, we can make mode decision using the neighbouring reference pixels and representative pixels of the block in the second step. The experimental results show that the proposed algorithm saved on the average 41.5% encoding time without any significant PSNR losses.

Key words: H.264, Intra Prediction(인트라 예측), Pixel Variation(화소 변화량)

1. 서 론

최근에 ISO/IEC와 ITU-T의 JVT(Joint Video Team)에서 개발한 H.264가 각광받고 있다[1]. H.264

는 방송, 대화형 서비스, VoD, 멀티미디어 스트리밍 서비스 등 다양한 분야에서 활용되고 있으며, MPEG-2, MPEG-4, H.263과 같은 기존 표준과 비교하여 동일한 화질을 유지하면서 평균 비트율을 약

※ 교신저자(Corresponding Author) : 김성민, 주소 : 부산광역시 급정구 장전동(609-735), 전화 : 051)510-2877, FAX : 051)515-2208, E-mail : morethanow@pusan.ac.kr
접수일 : 2007년 12월 27일, 완료일 : 2008년 5월 14일
[†] 준회원, 부산대학교 컴퓨터공학과 졸업(석사)
(E-mail : zero@melon.cs.pusan.ac.kr)

^{**} 준회원, 부산대학교 컴퓨터공학과 박사과정
^{***} 준회원, 부산대학교 컴퓨터공학과 박사과정
(E-mail : sin@pusan.ac.kr)
^{****} 중신회원, 부산대학교 컴퓨터공학과 교수
(E-mail : kdchung@pusan.ac.kr)

40% 이상 절약할 수 있다[2,3]. H.264에서는 이와 같이 높은 압축 효율을 위해 가중치 예측(Weighted Prediction), 다중 참조 프레임(Multiple Reference Frame), 다양한 블록 크기(Variable Block Size), 인트라 예측(Intra Prediction) 등과 같은 새로운 기술들을 도입하였다[4].

H.264에서 새로 도입한 여러 기법들 중 인트라 예측은 기존의 영상 표준에서는 없었던 기술로 뛰어난 영상 화질과 높은 압축 효율을 제공한다. 인트라 예측을 통한 높은 압축 효율은 다양한 부호화 블록 크기와 블록 크기별 다양한 부호화 모드에 의해 가능하다. 매크로블록 단위로 인트라 예측을 수행할 때, 표준에서는 16×16 블록에 대해 4가지 모드, 4×4 블록에 대해서는 9가지 모드를 지원한다. 이 과정에서 16×16 블록과 4×4 블록의 모든 모드에서 오차값을 계산하는데, 그 계산 과정이 복잡하다.

표준에서 제시된 인트라 예측은 화질은 우수하나 계산과정이 복잡해 처리 효율성이 떨어지므로, 처리 효율성을 개선하여 빠르게 인트라 예측이 가능한 기법들이 소개되었다. 하지만, 처리 효율성을 극도로 향상시키기 위해서는 화질의 손실이 발생되고, 화질의 손실을 최대한 막기 위해서는 처리 효율성의 큰 향상을 가져오기 힘들다. 따라서 표준과 유사한 화질 및 압축 효율을 유지하면서 계산복잡도를 개선할 수 있는 인트라 예측 기법이 요구된다.

본 논문에서는 빠른 인트라 예측을 위한 2단계 인

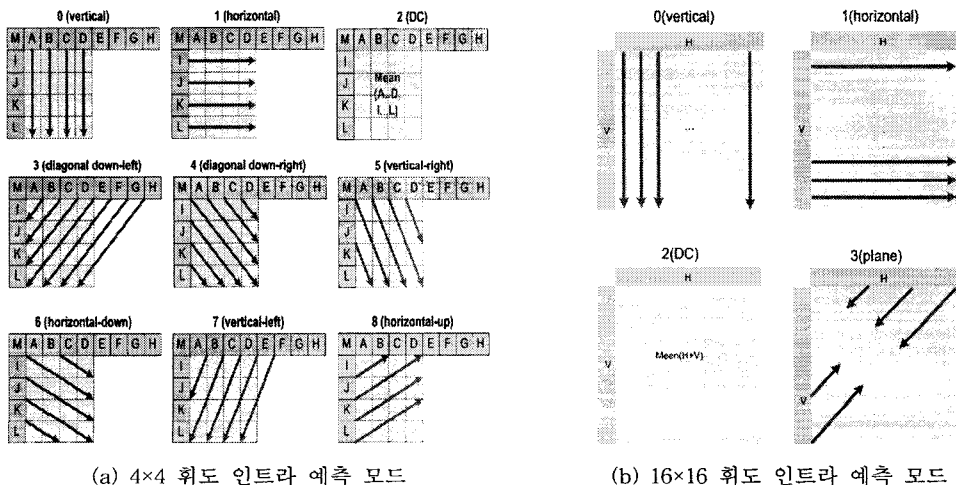
트라 예측 방법을 제안한다. 2단계 인트라 예측 방법은 매크로블록 내 경계 부분의 화소 변화량을 조사하고, 매크로블록의 평탄 여부를 판단하여 빠르게 블록 크기를 선택하는 1단계 블록 크기 선택 과정과, 선택된 블록의 모드 중에서 빠르게 최종 모드를 결정하는 2단계 최종 모드 선택 과정으로 구성된다.

본 논문의 구성은 다음과 같다. 2장에서 H.264 표준에서 채택하고 있는 인트라 예측 과정 및 관련 연구를 살펴보고, 3장에서는 제안하는 빠른 인트라 예측 기법에 대해서 알아본다. 4장에서는 제안한 기법의 성능을 분석 및 평가하고, 마지막으로 5장에서는 결론 및 향후 과제에 대해서 기술한다.

2. 관련 연구

본 장에서는 H.264에서 채택한 인트라 예측 방법에 대해서 알아보고, 인트라 예측 과정을 개선하기 위한 관련 연구에 대해서 살펴본다.

인트라 예측은 다른 프레임이나 블록을 참조하는 인터 예측과 달리 장면 내의 상관성을 공간적 정보를 통해 예측한다. H.264에서는 인트라 예측 모드를 위해 4×4 휘도(Luminance) 예측과 16×16 휘도 예측, 그리고 8×8 색차(Chrominance) 예측을 지원한다. 4×4 휘도 예측은 9가지 모드를 지원하는데, 그 9가지 모드는 그림 1에서 보는 바와 같이 수직(Vertical), 수평(Horizontal), 평균(DC), 대각 좌하(Diagonal



(a) 4×4 휘도 인트라 예측 모드

(b) 16×16 휘도 인트라 예측 모드

그림 1. H.264 인트라 예측 모드

Down-Left), 대각 우하(Diagonal Down-Right), 수직 우(Vertical-Right), 수평 하(Horizontal-Down), 수직 좌(Vertical-Left), 수평 상(Horizontal-Up) 방향이다. 16×16 휘도 예측은 4가지 방향을 지원하는데, 그림 1과 같이 수직(Vertical), 수평(Horizontal), 평균(DC), 플레인(Plane)이다. 그리고 8×8 색차 예측은 16×16 휘도 예측과 모드 번호만 다를 뿐 같은 방향성을 가진다.

블록 크기에 따른 인트라 예측의 특성을 살펴보면, 4×4 휘도 예측은 보다 세밀한 정보가 요구되는 블록의 경우에 더 적합하고, 16×16 휘도 예측은 블록의 전반적인 변화가 적은 경우에 더 적합하다[5,6]. 4×4 블록에서는 9개의 모드를 이용한 예측 블록과 현재 블록과의 차이를 구하고, 16×16 블록에서는 4개의 모드를 이용한 예측 블록과 현재 블록과의 차이를 구한다. 그리고 두 블록 크기의 예측 모드 중에서 부호화하는 매크로블록에 있어 가장 작은 오차값을 가지는 모드를 최종모드로 선택한다. 즉, 전체 모드에 대한 값을 계산하는 복잡한 처리과정 후에 최종 모드가 선택된다.

H.264에서는 RDO(Rate Distortion Optimization)를 통해서 최적의 모드를 선택하는 방법도 제공한다. 하지만, 각각의 모드에 대해서 RD 비용을 모두 계산해야 하므로, 비트율 대비 높은 화질에 비해 계산이 복잡하다는 단점을 가진다. 이와 같이 인트라 예측은 높은 압축 효율을 제공해 주는 반면, 복잡한 처리과정이 필요하므로 인트라 예측의 처리 효율성을 개선하기 위한 다양한 연구들이 진행되어 왔다[7-10].

[9]에서는 블록 크기를 빠르게 결정하기 위해 16×16 블록과 4×4 블록의 Edge Amplitude와 SATD(Sum of Absolute Transformed Difference) 사이의 임계값을 사용하고, 모드 결정을 위해서 [8]의 Sobel 연산자를 이용하여 빠르게 모드를 결정한다. 이 연구에서는 16×16 블록과 4×4 블록은 Edge Map Amplitude와 SATD 값에 의해서 확연하게 구분이 되는 특성을 확인하고, 16×16 블록과 4×4 블록을 임계값을 통해서 빠르게 결정할 수 있도록 하였다. 실험 결과, H.264의 참조 소프트웨어인 JM과 비교하여 유사한 화질과 비트율을 유지하면서도 상당량의 계산 시간을 줄였지만, 인트라 모드 결정 시 Edge Vector를 구하기 위한 많은 계산량과 4×4 블록 선택 시의 여전히 많은 RD 최적화 계산 과정으로 인해,

추가적인 계산 복잡도의 감소가 필요하다.

3. 빠른 인트라 예측 기법

본 장에서 제안하는 빠른 인트라 예측 기법은 매크로블록의 평탄 여부를 결정하여 인트라 예측 블록 크기를 선택하는 1단계 과정과, 선택된 16×16 블록 또는 4×4 블록의 대표 화소를 이용하여 최종 모드를 선택하는 2단계 과정으로 나뉜다. 다음은 각 단계별 처리과정에 대해서 설명하고 있다.

3.1 인트라 예측 블록 크기 결정 - 1단계

앞서 설명한 바와 같이 16×16 블록과 4×4 블록의 모든 모드를 계산해서 최종 모드를 선택하는 H.264 표준의 복잡함을 줄이기 위해서 16×16 블록과 4×4 블록 중 최종 모드가 포함된 블록 크기를 먼저 선택할 수 있다면 효율적인 인트라 예측이 가능하다. 본 논문에서 제안하는 빠른 인트라 예측 기법의 1단계 과정에서도 16×16 블록과 4×4 블록 중 최종 모드를 포함하는 블록 크기를 이른 단계에서 빠르게 결정한다.

일반적으로 인트라 부호화 시 결정되는 부호화 블록 크기는 블록의 평탄 정도와 연관성이 크기 때문에, 세부적이고 복잡한 정보를 가지고 있는 매크로블록은 4×4 블록 크기로 인트라 부호화되고, 움직임이 적고 정보의 변화량이 적은 매크로블록은 16×16 블록 크기로 인트라 부호화된다. 즉, 매크로블록 내 주변 화소들 간의 차이가 적어 블록이 평탄한 경우 16×16 블록 크기로 인트라 부호화되고, 그렇지 않은 경우 4×4 블록 크기로 인트라 부호화된다.

1단계에서는 현재 부호화하는 매크로블록의 인트라 예측 블록 크기를 빠르게 결정하기 위해서 매크로블록 내의 부분 블록들(4×4 블록, 8×8 블록, 12×12 블록)의 경계 화소 차이의 합을 이용한다. 블록 크기의 선택 기준은 그림 2에 있는 내부 블록들의 경계 화소 값들을 이용하여 수식 (1)을 통해서 이루어진다. 예를 들어, 가장 안쪽의 4×4 블록의 경계차를 구할 때, 그림 2의 오른쪽 그림처럼 4×4 블록의 윗부분의 경계차는 $|UA1-UB1| + |UA2-UB2| + |UA3-UB3| + |UA4-UB4|$ 와 같이 구할 수 있고, 나머지 세 부분의 경계차도 동일한 방법으로 구할 수 있다. 나머지 두 블록의 경계차도 동일한 방법으로 구할 수 있고, 이렇게 구해진 3개 블록의 경계차 합이 실험을

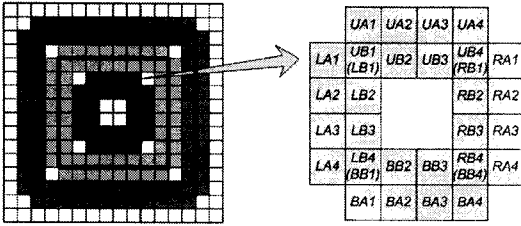


그림 2. 16×16 블록의 평탄 정도

통해 구해진 임계값보다 작게 되면 16×16 블록으로 선택되고, 그렇지 않으면 4×4 블록이 선택된다. 임계값 설정 및 자세한 설명은 참고문헌 [11]에 제시되어 있다.

$$\text{평탄정도} = \sum_{n=4,8,12,16}^n (|UA_n - UB_n| + |LA_n - LB_n| + |RA_n - RB_n| + |BA_n - BB_n|) \quad (1)$$

3.2 최종 모드 결정 - 2단계

1단계를 통해서 부호화 블록 크기가 결정되면, 필요한 블록 크기의 예측 모드에 대한 계산과정을 없앨 수 있기 때문에, 인트라 예측의 효율성을 상당 부분 개선시킬 수 있다. 하지만, 4×4 블록 및 16×16 블록은 각각 9개와 4개의 모드를 가지고 있어, 최종 모드를 선택하기 위해서는 선택된 블록 크기의 모든 모드를 조사해야 하므로, 이 과정에서 모든 모드를 조사하지 않고 최종 모드를 결정할 수 있다면 추가적인 효율성의 개선이 가능하다. 따라서 본 절에서는 1단계를 통해서 4×4 블록 및 16×16 블록으로 선택되었을 경우 해당 블록의 최종 모드를 빠르게 결정할 수 있는 방법을 소개한다.

3.2.1 4×4 휘도 예측 모드 결정

1단계에서 선택된 블록이 4×4 블록일 경우, 최종 모드를 결정하기 위해서 블록 내와 블록 주변의 대표

픽셀들을 이용하여 후보 모드를 우선적으로 선택하고, 모드 선택의 정확성을 높이기 위해서 후보 모드의 인접 방향성을 고려하여 후보 모드 그룹을 구성한다. 후보 모드 그룹은 4개의 모드로 구성되어 있어 9개의 모드에 비해 계산 과정을 감소시킬 수 있다.

가. 대표 화소를 이용한 후보 모드 선택

4×4 인트라 모드를 선택하기 위해 그림 3에 나타난 4가지 방향(Vertical, Horizontal, Diagonal Down-Right, Diagonal Down-Left)을 조합하여 만든 총 8가지의 비교 그룹을 이용한다. H.264에서 이용하는 8개의 방향 대신 그림 3과 같이 4개의 방향을 이용하고, 현재 블록 안의 대표 화소를 선정하여 인접한 블록의 참조 화소와의 비교를 통해 빠르게 모드를 선택한다.

그림 4는 참조 화소 중 대표 화소인 A, D, F, I, L과 현재 블록의 대표 화소인 a, b, c, d, e, h, i, l, m, n, o, p의 차이 값을 구하는 방법을 보여준다. 예를 들어, 그림 4에서 Diagonal Down-Right의 비교 값은 $|A-b|+|A-l|$ 와 $|I-e|+|I-o|$ 로 구해지게 되고, 이 비교 값이 다른 비교 그룹에 비해 가장 작은 값이면 A와 b, A와 l, I와 e, I와 o의 차이 값이 가장 작다는 의미로 그림 1에서 모드 4,5,6으로 나타날 확률이 높게 된다. 여기서, 두 번째로 작은 비교 값에 따라서 최종 후보 모드를 선택하게 되는데, 만일 두 번째로 작은 비교 값이 Horizontal1 또는 Horizontal2라면 수평방향에 가까운 6번 모드가 선택된다.

제안하는 기법에서 나머지 후보 모드의 선택도 앞서 설명한 방법과 유사한 과정을 통해서 처리되고, 후보 모드를 선택하기 위한 비교 수식은 그림 5와 그림 6에 나타나 있다. 그림 6의 비교 관계식 II에서 $a_1 \sim a_8$ 의 값을 비교하여 후보 모드를 선택하는 알고리즘은 그림 7과 같다.

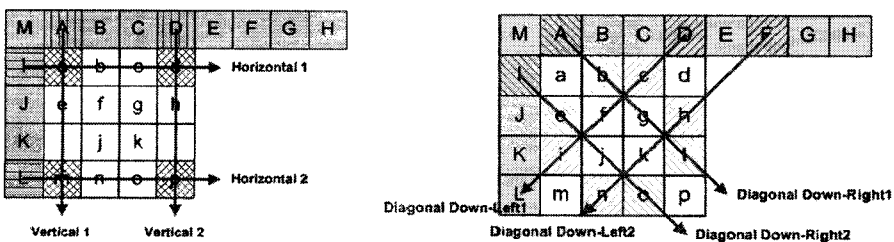


그림 3. 4×4 인트라 예측을 위한 제안 모드

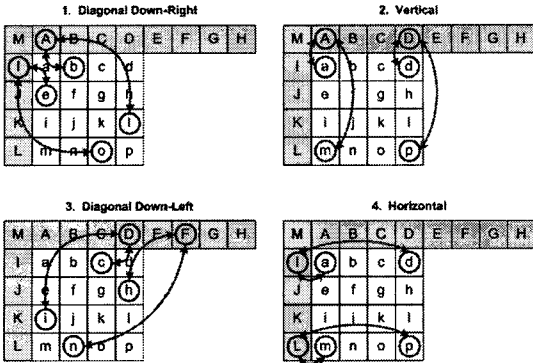


그림 4. 4×4 인트라 예측을 위한 대표 화소

방향	관계식
Diagonal Down-Right 1	$D_1 = A-b, D_2 = A-l$
Diagonal Down-Right 2	$D_3 = I-e, D_4 = I-o$
Vertical 1	$D_5 = A-a, D_6 = A-m$
Vertical 2	$D_7 = D-d, D_8 = D-p$
Diagonal Down-Left 1	$D_9 = D-c, D_{10} = D-i$
Diagonal Down-Left 2	$D_{11} = F-h, D_{12} = F-n$
Horizontal 1	$D_{13} = I-a, D_{14} = I-d$
Horizontal 2	$D_{15} = L-m, D_{16} = L-p$

그림 5. 4×4 인트라 예측을 위한 비교 관계식 I

$a_1 = D_1 + D_2 $	$a_2 = D_3 + D_4 $
$a_3 = D_5 + D_6 $	$a_4 = D_7 + D_8 $
$a_5 = D_9 + D_{10} $	$a_6 = D_{11} + D_{12} $
$a_7 = D_{13} + D_{14} $	$a_8 = D_{15} + D_{16} $

그림 6. 4×4 인트라 예측을 위한 비교 관계식 II

Step1 : $a_1 \sim a_8$ 중에서 가장 작은 두 개를 선택
 선택된 두 값의 조합이

- (a_3, a_1) 혹은 (a_3, a_2) 혹은 (a_4, a_1) 혹은 (a_4, a_2) 이면 모드 5 선택
- (a_7, a_1) 혹은 (a_7, a_2) 혹은 (a_8, a_1) 혹은 (a_8, a_2) 이면 모드 6 선택
- (a_3, a_5) 혹은 (a_3, a_6) 혹은 (a_4, a_5) 혹은 (a_4, a_6) 이면 모드 7 선택
- 선택된 조합이 i ~ iii에 해당하지 않으면 Step2로 이동

Step2 : $a_1 \sim a_8$ 의 값 중에서 가장 작은 값을 선택

- a_3 또는 a_4 일 경우 모드 0 선택
- a_7 또는 a_8 일 경우 모드 1 선택
- a_5 또는 a_6 일 경우 모드 3 선택
- a_1 또는 a_2 일 경우 모드 4 선택

그림 7. 대표 화소를 이용한 후보 모드 선택 알고리즘

나. 인접 방향성을 이용한 후보 모드 그룹

대표 화소를 이용한 후보 모드 선택 시, 모드 2번과 모드 8번이 선택되지 않는다. 따라서 최종 모드를 선택할 때 정확도를 높이기 위해 후보 모드의 인접된 모드와 방향성을 가지지 않는 2번 모드를 추가적으로 후보 모드로 선택하여 후보 모드 그룹을 만들고, 그 중에서 최종 모드를 선택한다.

예를 들어, 그림 8에서 보는 바와 같이 이전 단계에서 후보 모드가 5로 선택되었다면 모드 5에 인접한 방향성을 가지는 모드 0과 모드 4를 추가적으로 후보 모드로 선택하고, 방향성이 없는 모드 2를 포함하여 최종 후보 모드 그룹이 구성된다. 최종적인 후보 모드 그룹은 표 1에 나타나 있다.

3.2.2 16×16 휘도 예측 모드 결정

1단계에서 선택된 블록이 16×16 블록일 경우, 최종 모드의 결정은 4×4 블록에서 후보 모드를 선택하기 위해 대표 화소를 이용한 방법과 유사하게 이루어진다. 그림 9는 전체 대표 화소들을 나타내고, 그림 10은 방향성 별로 조사되는 대표 화소들을 나타내고 있다.

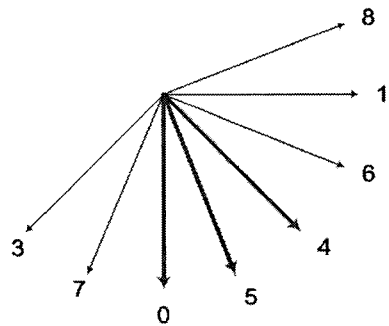


그림 8. 선택된 후보 모드가 5인 경우의 인접된 후보 모드들

표 1. 후보 모드 그룹

선택된 후보 모드	후보 모드 그룹
0	모드 7, 0, 5, 2
1	모드 8, 1, 6, 2
3	모드 7, 3, 8, 2
4	모드 6, 4, 5, 2
5	모드 4, 5, 0, 2
6	모드 1, 6, 4, 2
7	모드 0, 7, 3, 2

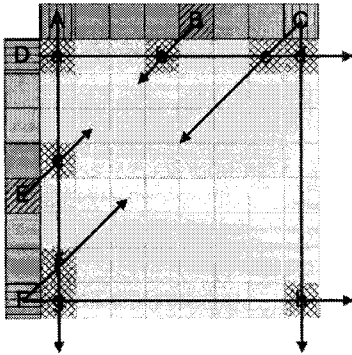


그림 9. 16×16 인트라 예측을 위한 제안 모드

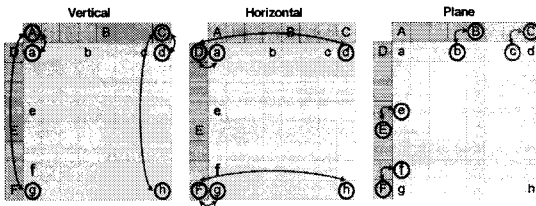


그림 10. 16×16 인트라 예측을 위한 대표 화소

대표 화소를 통해서 후보 모드를 선택하기 위한 방법은 그림 11에 있는 비교 관계식 값 중에서 D1~D4의 합, D5~D8의 합, D9~D12의 합 가운데 가장 작은 값에 해당되는 방향성의 모드를 후보 모드로 선택한다. 즉 D1~D4의 값이 가장 작은 경우 후보 모드는 모드 0(Vertical)이 선택된다. 모드 2의 경우 4×4에서와 마찬가지로 방향성이 없기 때문에 대표 화소를 통해서 선택된 모드 0, 1, 3과 함께 후보 그룹에 포함되어, 최종 모드를 선택하기 위해서 사용된다. 최종적으로 구성된 후보 모드 그룹은 표 2와 같다.

3.3 빠른 인트라 예측 기법 - 전체 알고리즘

본 논문에서 제안하는 빠른 인트라 예측은 앞서

방 향	관 계 식
Vertical 1	$D_1 = A-a, D_2 = A-g$
Vertical 2	$D_3 = C-d, D_4 = C-h$
Horizontal 1	$D_5 = D-a, D_6 = D-d$
Horizontal 2	$D_7 = F-g, D_8 = F-h$
Plane 1	$D_9 = C-c, D_{10} = F-f$
Plane 2	$D_{11} = B-b, D_{12} = E-e$

그림 11. 16×16 인트라 예측을 위한 비교 관계식

표 2. 후보 모드 그룹

선택된 후보 모드	후보 모드 그룹
0	모드 0, 2
1	모드 1, 2
3	모드 3, 2

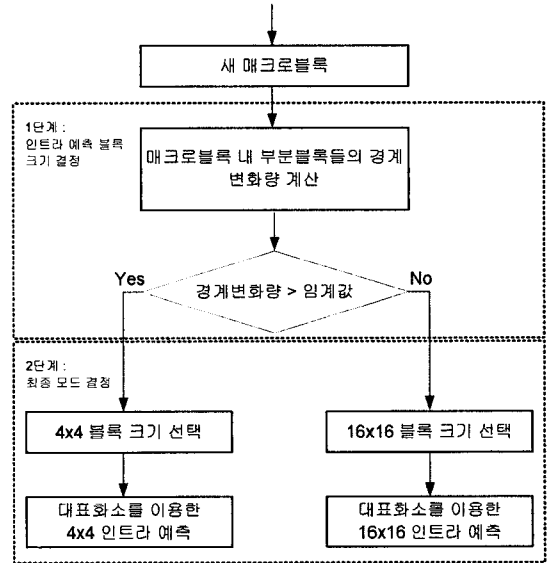


그림 12. 전체 알고리즘

설명했던 바와 같이 4×4 블록 크기와 16×16 블록 크기를 빠르게 선택하는 1단계 과정과 선택된 블록 크기 안에서 대표 화소를 이용하여 최종 인트라 모드를 선택하는 2단계 과정으로 이루어지며, 빠른 단계에서의 블록 크기 선택 및 대표 화소를 이용한 최종 모드의 선택을 통해서 많은 계산과정을 제거할 수 있다. 그림 12는 빠른 인트라 예측을 위한 전체 과정을 나타내고 있다.

4. 실험 및 성능 평가

4.1 실험 환경

본 논문에서 제안하는 빠른 인트라 예측 기법의 성능을 평가하기 위해 움직임의 특성이 다른 6가지 동영상을 이용하여 실험하였다. 표 3은 성능 평가를 위한 실험 환경과 측정 인자를 나타낸다. 실험을 위해 사용된 테스트 동영상은 움직임이 거의 없는 Akiyo와 Suzie, 움직임이 중간 정도인 Carphone 그

표 3. 성능 평가를 위한 실험 환경

CPU	Core 2 Dou 2.2GHz
메모리	2GB
파일 포맷	QCIF(176×144)
동영상 종류	Akiyo, Suzie, Carphone, Foreman, Football, Tempete
프레임 구성	모두 I 프레임으로 구성
프레임 수	100
양자화 계수	20, 24, 28, 32, 36, 40
참조 소프트웨어	JM 10.2
측정 인자	화질(PSNR), 계산 시간(sec), 비트율(Kbit/sec)

리고 움직임이 많은 Foreman과 Football로 움직임 정도에 따라 구분하였고, 화면 줌인과 같은 특별한 움직임을 가지는 Tempete을 추가로 실험하였다. 본 논문의 제안 기법은 H.264 참조 소프트웨어인 JM 10.2를 수정하여 구현하였다[12].

4.2 성능 평가

표 4는 실험환경에 따라 JM, Zhang의 기법[9], 제

표 4. 동영상별 성능 비교 (양자화 계수 : 28)

동영상	비교기법	PSNR (dB)	Bits (kbit)	Time (sec)
Akiyo	JM10.2	38.84	560.36	4.157
	Zhang의 기법	38.80	562.15	2.892
	제안기법	38.77	564.73	2.315
Suzie	JM10.2	38.24	455.13	4.079
	Zhang의 기법	38.18	456.38	3.561
	제안기법	38.15	457.21	2.371
Carphone	JM10.2	38.10	603.87	4.189
	Zhang의 기법	38.07	604.65	3.069
	제안기법	38.04	605.01	2.374
Foreman	JM10.2	36.64	749.44	4.111
	Zhang의 기법	36.60	749.97	3.10
	제안기법	36.59	750.19	2.49
Football	JM10.2	36.86	769.24	4.249
	Zhang의 기법	36.85	770.62	3.275
	제안기법	36.84	771.08	2.328
Tempete	JM10.2	35.53	1254.93	4.359
	Zhang의 기법	36.50	1255.24	3.635
	제안기법	36.47	1256.17	2.817

안하는 기법의 PSNR과 비트율 및 계산 시간을 비교한 결과를 보여준다. 표 4를 통해 제안하는 기법이 PSNR 값과 비트율에서는 JM, Zhang의 기법과 비교하여 거의 비슷한 결과를 보여주고, 계산 시간에서는 큰 감소를 보이고 있음을 확인할 수 있다. 즉, 제안하는 기법이 화질과 압축 효율을 기존의 표준과 동일하게 유지하면서도, 표준과 비교하여 41.5%, Zhang의 기법과 비교하여 24.7% 이상 계산 복잡도를 감소시켰다.

그림 13~그림 15는 Suzie, Foreman, Tempete 동영상의 비트율에 대한 PSNR 변화를 보여준다. 제안하는 기법은 JM과 비교하여 약간의 화질 저하와 비트율 증가를 보이고 있지만 거의 무시할 정도의 값을 보여주고 있다.

그림 16~그림 18은 JM10.2와 Zhang의 기법 및 제안기법을 통해 생성된 Akiyo, Carphone, Football 동영상의 특정장면에 대한 주관적 화질의 비교이다. 객관적인 화질을 대표하는 PSNR에서 세 가지 기법

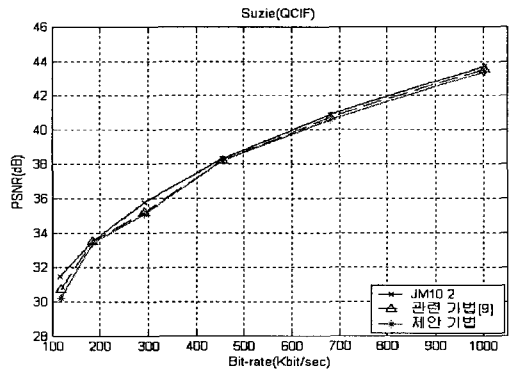


그림 13. 비트율에 따른 PSNR 변화 (Suzie)

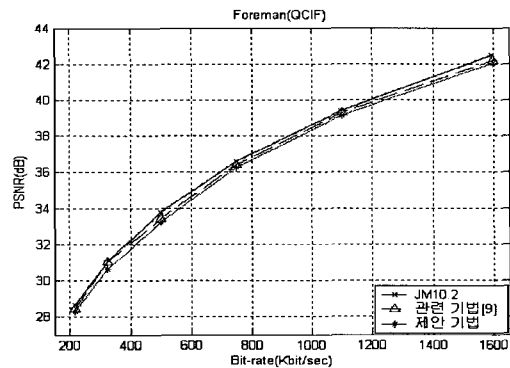


그림 14. 비트율에 따른 PSNR 변화 (Foreman)

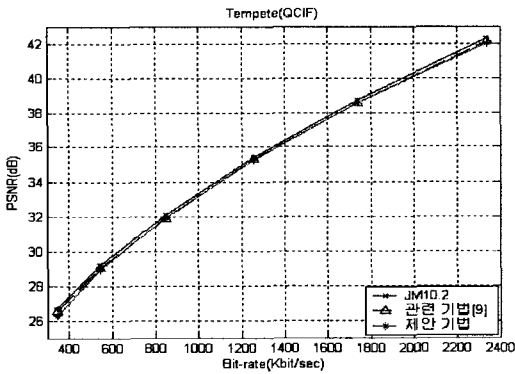


그림 15. 비트율에 따른 PSNR 변화 (Tempete)

이 유사한 값을 가진 것과 마찬가지로, 주관적 화질도 큰 차이가 없음을 확인할 수 있다.

5. 결론 및 향후 과제

H.264는 기존 표준에 없던 새로운 기법들로 압축 효율의 뛰어난 향상을 가져왔다. 새롭게 도입된 기법 중에서 다양한 모드를 통한 인트라 예측 기법은 H.264의 높은 압축 효율에 크게 기여하고 있지만, 복잡한 처리가 필요하다는 단점을 가지고 있다. 따라서 본 논문에서는 기존의 복잡한 인트라 예측 방법을 매크로블록의 평탄 정도를 이용한 빠른 블록 크기 결정과 결정된 블록 크기 안에서 대표 화소를 이용한 빠른 모드 결정을 통해 개선된 처리시간을 보여주는 빠른 인트라 예측 기법을 제안하였다. 실험 결과, 제안한 기법은 화질과 압축 효율을 기존의 표준과 동일하게 유지하면서도, 표준 및 관련기법과 비교하여 각

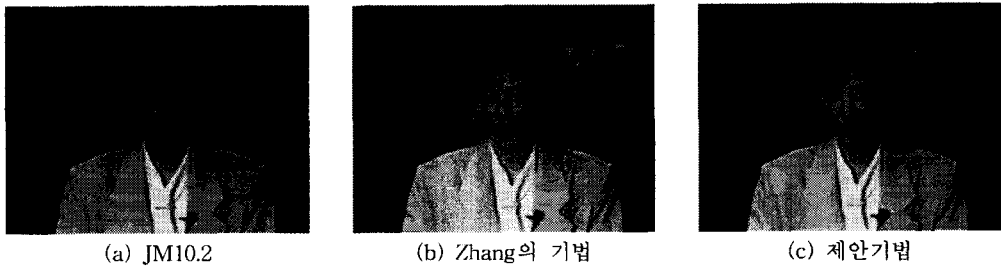


그림 16. Akiyo 동영상의 주관적 화질 비교 (프레임 번호 : 15)



그림 17. Carphone 동영상의 주관적 화질 비교 (프레임 번호 : 55)



그림 18. Football 동영상의 주관적 화질 비교 (프레임 번호 : 64)

각 41.5%, 24.7%의 처리시간을 감소시켰다. 현재, 휴대 이동기기를 포함하여 다양한 형태의 멀티미디어 기기 활용이 높아지고 있으므로, 향후에는 최적화된 계산량으로 높은 화질과 압축 효율이 가능한 연구가 필요할 것이다.

참 고 문 헌

[1] Joint Video Team(JVT) of ISO/IEC MPEG and ITU-T VCEG, "Draft ITU-T recommendation and final draft of Joint Video Specification (ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC)," *Document JVT-G050*, Mar. 2003.

[2] 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 Syst. Magazine*, Vol.4, No.1, pp. 7-28, Apr. 2004.

[3] P. Topiwala, G. Sullivan, A. Joch, and F. Kossentini, "Performance evaluation of H.26L TML 8 vs. H.263++ and MPEG4," *Document VCEG-042*.

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

[5] Q. Liu, R. Hu, L. Zhu, X. Zhang, and Z. Han, "Improved fast intra prediction algorithm of H.264/AVC," *Journal of Zhejiang University SCIENCE A*, Vol.7, pp. 101-105, July 2006.

[6] Y. K. Lin and T. S. Chang, "Fast Block Type Decision Algorithm for Intra Prediction in H.264 FRext," *ICIP 2005. IEEE International Conference of Image Processing*, Vol.1, pp. 585-8, Sept. 2005.

[7] J. S. Park and H. J. Song, "Fast selective intra mode decision in H.264/AVC," *IEEE CCNC 2006*, Vol.2, pp. 1068-1072, Jan. 2006.

[8] F. Pan, X. Lin, S. Rahardja, K. P. Lim, Z. G. Li, D. Wu, and S. Wu, "Fast mode decision

algorithm for intraprediction in H.264/AVC video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol.15, No.7, 2005.

[9] K. Zhang, C. Yuan, Q. Li, and Y. Zhong, "A Fast Block Type Decision Method for H.264/AVC Intra Prediction," *The 9th International Conference on Advanced Communication Technology*, Vol.1, pp. 673-676, Feb. 2007.

[10] C. Kim, H. H. Shih, and C. J. Kuo, "Feature based intra prediction mode decision for H.264," *International conference on Image Processing*, Vol.2, pp. 769-772, Oct. 2004.

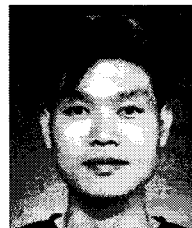
[11] S. M. Kim, T. G. Lee, K. M. Sin, J. M. Kang, and K. D. Chung, "A Fast Block Size Decision Using Edge Information of Sub-blocks for H.264/AVC Intra Prediction," *Proceeding of the 7th APIS(Application and Principles of Information Science)*, pp.703-706, Jan. 2008.

[12] JM Reference Software Version 10.2, <http://iphome.hhi.de/suehring/tml/download/>



이 탁 기

2006년 경성대학교 컴퓨터학과 졸업(학사)
 2008년 부산대학교 컴퓨터공학과 졸업(석사)
 관심분야 : 멀티미디어 압축, 멀티미디어 통신



김 성 민

2001년 부산대학교 전자계산학과 졸업(학사)
 2003년 부산대학교 전자계산학과 졸업(석사)
 2003년~현재 부산대학교 컴퓨터공학과 박사과정

관심분야 : 멀티미디어 압축 및 통신, 멀티미디어 스트리밍



신 광 무

- 2005년 부산대학교 전자전기정보컴퓨터공학부 졸업(학사)
- 2005년 부산대학교 컴퓨터공학과 졸업(석사)
- 2007년~현재 부산대학교 컴퓨터공학과 박사과정

관심분야 : 멀티미디어 압축, 다시점 비디오 부호화, 임베디드 리눅스



정 기 동

- 1973년 서울대학교 졸업(학사)
- 1975년 서울대학교 대학원 졸업(석사)
- 1986년 서울대학교 대학원 계산통계학과 졸업(이학박사)
- 1978년~현재 부산대학교 컴퓨터공학과 교수

관심분야 : 멀티미디어 시스템, 멀티미디어 통신, 병렬처리