

H.264/AVC에서 향상된 인트라/인터 예측을 위한 모드 추정 방법

박경석¹, 김민준², 전재현², 류상률³, 김승호^{4*}
¹경북대학교 모바일통신공학과, ²경북대학교 전자전기컴퓨터학과,
³청운대학교 컴퓨터학과, ⁴경북대학교 컴퓨터학부

Enhanced Mode Estimation Method for Intra/Inter Prediction in H.264/AVC

Kyung-Seok Park¹, Min-Jun Kim², Jae-Hyun Jun², Sang-Ryul, Ryu³ and
Sngng-Ho Kim^{4*}

¹Dept. of Information and Communication Engineering, Kyungpook National University

²Dept. of Electrical Engineering and Computer Science, Kyungpook National University

³Dept. of Computer Science, Chungwoon University

⁴Dept. of Computer Science and Engineering, Kyungpook National University

요 약 H.264/AVC에서는 인트라 예측과 인터 예측의 움직임 추정은 전체 압축 시간의 70 ~ 80%를 차지하게 된다. 다양한 부호화 기술을 이용하여 압축 효율은 높아지지만 복잡도가 증가하여 부호화 시간이 증가하게 되었다. 따라서 본 논문에서는 H.264 부호화 과정 중 화질 손실을 최소한으로 줄이면서 소모되는 시간을 단축시키는 인트라 예측의 블록크기 결정방법과 모드 결정방법을 제안하고자 한다. 추가적으로 계산량을 줄이기 위해 인터 예측의 움직임 추정을 위해 적응적으로 탐색방법을 결정하는 알고리즘을 제시하고자 한다. 실험을 통해 화질의 열화와 계산량을 측정하기 위해서 PSNR과 인트라 예측 및 인터 예측 계산 시 소모되는 시간을 알아보았다. 그 결과, 3가지 방법을 모두 사용했을 때, 기존의 H.264방법과 비교하여 화질은 거의 비슷하게 유지하면서 실험한 모든 영상의 경우 모든 프레임에서 평균 500 ~ 600ms정도의 부호화 시간이 단축됨을 보였다.

Abstract The intra prediction and the motion estimation of inter prediction occupy 70 ~ 80% of whole compression time in H.264/AVC. Compression efficiency has been higher, but complexity has increased and coding time has also increased much more. This paper proposes a block size decision method of the intra prediction and mode decision method which minimize the loss of video quality during the encoding and shorten the time spent. This paper, in addition, proposes an algorithm which determines the method of adaptive block mode for motion estimation of inter prediction. We investigated PSNR and the intra prediction and inter prediction of time-consuming calculations in order to measure video quality degradation and complexity through experiments. Consequently, when you use all three methods, these methods showed that average coding time is shortened as about 500 to 600ms in every frame in the case of all experimented videos, keeping video quality nearly similar, compared with existing methods of H.264.

Key Words : H.264/AVC, Intra prediction, Inter prediction, Block mode decision, PSNR, Complexity

1. 서론

H.264/AVC 표준은 ITU-T의 ISO/IEC의 MPEG (Moving Picture Experts Group)과 VCEG (Video Coding

Expert Group)의 공동 작업으로 제정되었다[1][2].

H.264/AVC의 부호화 방식은 이전의 동영상 부호화 표준들(MPEG-2, MPEG-4, H.263)에는 없는 새로운 부호화 방식들이 추가되었으며, 이러한 방식들로 인해 H.264/

*Corresponding Author : Sngng-Ho Kim

Tel: +82-16-834-0560 email: shkim@knu.ac.kr

접수일 12년 01월 20일

수정일 (1차 12년 02월 09일, 2차 12년 02월 27일)

게재확정일 12년 04월 12일

AVC표준은 이전의 표준들과 비교하여 보다 높은 압축효율을 가진다.

새롭게 추가된 부호화 방식에는 4×4 단위의 정수변환(integer transform), CAVLC(Context-based Adaptive Variable Length Coding) 또는 CABAC(Context-based Adaptive Binary Arithmetic Coding)을 통한 엔트로피 부호화, 루프필터(loop-filter), 인트라 예측(intra-prediction) 부호화, 다중 참조 영상의 사용, 다양한 블록 크기(VBS: Variable Block Size)등이 추가되었다. 이렇게 새롭게 추가된 부호화 방식들 중 인트라 예측은 기존의 영상 표준에서는 없었던 기술로 뛰어난 영상화질과 높은 압축효율을 제공한다. 인트라 예측을 통한 높은 압축 효율은 다양한 부호화 모드에 의해 가능하다. 매크로블록 단위로 인트라 예측을 수행할 때, 표준에서는 16×16블록에 대해 4가지 모드, 4×4블록에 대해서는 9가지 모드를 지원한다. 이 과정에서 16×16블록과 4×4블록의 모든 모드에서 유효곡 최적화 비용(RDcost: Rate Distortion cost)을 계산하는데, 그 과정으로 인해 부호화 시간이 길어지고 고성능의 하드웨어를 필요로 하는 단점을 가지고 있다[3][4].

H.264/AVC에서 제시된 인트라 예측은 화질은 우수하나 계산과정이 복잡해 처리 효율성이 떨어지므로, 처리 효율성을 개선하여 빠르게 인트라 예측이 가능한 기법들이 제안되었다. 하지만, 처리 효율성을 향상시키기 위해서는 화질의 손실이 발생되고, 화질의 손실을 줄이기 위해서는 처리 효율성의 향상을 가져오기 힘들다. 따라서 화질의 손실을 줄이면서 계산복잡도를 개선할 수 있는 인트라 예측 기법이 요구된다.

본 논문에서는 빠른 인트라 예측을 위해 2단계 인트라 예측방법을 제안한다. 2단계 인트라 예측 방법은 매크로블록 내의 공간 상관도에 따라 블록크기를 선택하는 과정과 선택된 블록의 모드 중에서 빠르게 최종 모드를 결정하는 과정을 제안하고자한다. 추가적으로 움직임 추정 과정에서의 계산량을 줄이는 방법을 제시하여 부호화 과정의 압축시간을 향상시키고자 한다.

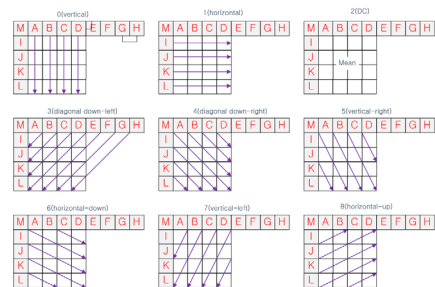
본 논문의 세부적 구성으로는 2장에서 H.264/AVC의 주요 인트라 예측에 대해 설명하고 3장에서는 고속 인트라 모드를 결정하는 알고리즘을 제안하고 4장에서는 기존의 방법과의 성능을 분석한다. 제안된 방법과 기존의 방법 간의 비교실험을 통해 결론을 맺도록 하겠다.

2. H.264/AVC에서의 인트라 예측과 인터 예측 방법

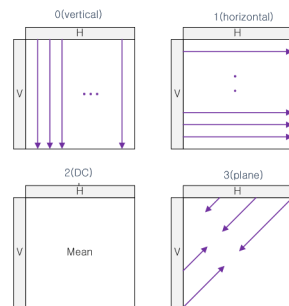
2.1 인트라 예측 방법

인트라 예측은 다른 프레임이나 블록을 참조하는 인터

예측과 달리 장면 내의 상관성을 공간적 정보를 통해 예측한다. H.264/AVC에서 인트라 프레임에 포함되어있는 모든 매크로블록들은 휘도(luminance)블록의 부호화를 위한 두 개의 인트라 모드(Intra16×16, Intra4×4)와 색차블록의 부호화를 위한 하나의 인트라 모드(Chroma8×8, 컬러포맷이 4:2:0인 경우)를 사용하여 부호화 한다. 각 인트라 모드는 서로 다른 예측 모드의 수를 가지는데, 그림 1은 Intra4×4 모드에서 사용하는 9가지의 예측 모드를 나타내며, 그림 2는 Intra16×16 모드에서 사용되는 4가지의 예측 모드를 나타낸다. 각 예측 방향에 따라 예측 번호가 할당되고, 예측방향의 발생빈도가 높을수록 낮은 번호가 할당된다. ChromaIntra8×8의 예측 모드는 Intra16×16과 동일하며 통계적 발생빈도에 따라 그 순서만을 달리한다.



[그림 1] 인트라 4×4 모드에서의 9가지 예측 방법
[Fig. 1] 9 Prediction method in Intra4×4



[그림 2] 그림 2. 인트라 16×16 모드에서의 4가지 예측 방법
[Fig. 2] 4 Prediction method in Intra16×16

유효곡 최적화기법이 사용되는 경우 참조소프트웨어는 색차블록의 각 예측 모드에 대해서 인트라 16×16과 인트라 4×4의 RDcost값을 비교하여 최소의 비용 값을 갖는 모드를 해당 매크로블록의 휘도 및 색차블록의 인트라 모드로서 선택한다. 이 과정에서 인트라 4×4는 각 4×4블록마다 RDcost 값을 계산하며 매크로블록 내에 포함되어 있는 16개의 모든 4×4블록에서의 RDcost 값의 합을

Intra16×16의 RDcost값과 비교한다[5][6]. RDcost 계산식은 식 1과 같다[7].

$$RDcost = Distortion + \lambda_{mode} \times Rates \quad (1)$$

$$\lambda_{mode} = 0.85 \times 2^{(QP-12)/3}$$

식 1에서 λ_{mode} 는 라그랑지안(Lagrangian) 상수이고, QP는 0에서 51까지의 값을 가질 수 있는 양자화 파라미터(quantization parameter)이다. Rates는 비트율을 의미한다. Distortion은 현재 영상과 복원된 영상의 에러 값이며 식 2와 같이 계산된다. 식 2에서 $c(i,j)$ 와 $r(i+x,j+y)$ 는 각각 현재 매크로블록과 복원된 매크로 블록의 (i,j)번째 화소 값을 나타낸다.

$$Distortion = \sum_{i=0}^{15} \sum_{j=0}^{15} (c(i,j) - r(i+x,j+y))^2 \quad (2)$$

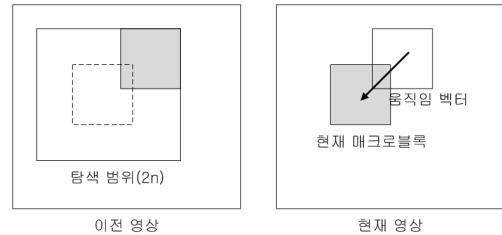
앞서 언급한 바와 같이 각 인트라 모드는 서로 다른 예측 모드의 방법을 가지기 때문에, 하나의 인트라 매크로블록에 대한 최적의 인트라 모드를 선택하기 위해서는 모든 예측 모드들의 조합에 대한 RDcost값을 계산하여야 하며, 최종적으로 부호화에 사용된 해당 매크로블록의 예측 모드의 정보는 부호화된 비트스트림에 포함하게 된다.

인트라 모드 선택과정은 4개의 색차성분의 예측 모드에 대해서 먼저 인트라 16×16의 4개의 예측 모드들 중 최소의 비용 값을 가지는 예측 모드를 선택하고, 이후 인트라 4×4의 비용값을 계산한다. 이러한 과정은 4가지의 색차성분에 대해서 수행되며 최종적으로 최소의 비용 값을 갖는 모드를 휘도블록 및 색차블록의 매크로블록 모드로서 선택되어진다. 따라서, 요구되어지는 많은 계산량을 줄이기 위해서 본 논문에서는 인트라 16×16과 인트라 4×4를 결정하는 블록 크기 결정방법과 예측 모드를 줄이기 위한 방법을 제안하고자 한다.

2.2 인트라 예측 방법

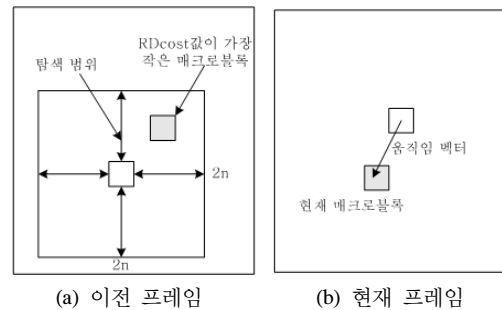
시간적인 중복성은 인트라 예측에 있어서 영상 부호화의 압축효율에 중요한 역할을 한다. 그러므로 대부분의 부호화 방법들이 압축방식에 있어서 복호기로 이미 전송된 영상들을 활용하고 있다. 움직임은 연속된 영상들 사이의 차이를 만드는 주요 원인이 되기 때문에 움직임 보상 기술은 다음 영상을 예측하는데 적용되고 따라서 움직임 추정이 이루어져야 한다. 때때로 움직임 추정은 부호기에서 주로 이루어지고 복호기로는 움직임 정보가 주로 전송된다. 적절한 움직임 모델의 선택은 응용 모델에 따라

다르기 때문에 그 다양성이 고려되어야 한다. 그러나 오늘날 비디오 압축 표준에서는 블록마다 하나의 움직임 벡터가 전송되고 블록의 분할에 따라서 적응적인 움직임 모델을 사용하고 있다[8]. 그림 3은 움직임 추정에 있어서 각 영역마다 적절한 움직임 벡터를 찾는 과정이다.



[그림 3] 움직임 추정을 위한 블록정합 방법
[Fig. 3] Block matching method for motion estimation

블록 정합 방식에는 크게 고정 블록 정합과 가변 블록 정합이 있다. 전자는 고정된 블록 크기로 움직임 추정이 이루어지기 때문에 부호화 효율이 낮은 반면 후자의 경우에는 영상 내 객체와 배경의 크기 및 움직임의 변화에 따라서 가변 블록 크기로 움직임 추정이 이루어지기 때문에 높은 부호화 효율을 제공한다. 움직임 추정에서 연산량의 대부분은 RDcost의 계산 과정에서 나온다.



(a) 이전 프레임 (b) 현재 프레임
[그림 4] 전역탐색기법
[Fig. 4] Full search algorithm

그림 4에서와 같이 전역탐색 방법은 RDcost 값을 이용하여 현재 매크로블록의 두 배(2n)의 탐색범위를 가지며, 탐색 영역 내에 있는 모든 탐색점들에 대한 RDcost를 계산하고 그 중 가장 작은 정합 오차를 가지는 지점에 대한 이동 변위를 움직임 벡터로 추정한다. 이 방법으로 구한 벡터를 이용하여 움직임 추정을 수행하면 가장 좋은 화질을 제공할 수 있지만 현재 프레임과 이전 프레임간의 탐색 범위 이내의 모든 블록간의 정합 오차를 계산해야 하므로 많은 계산량으로 인해 실시간 부호화에 적합

하지 않다는 단점이 있다. 따라서, 많은 고속 인터 모드 예측방법이 제안되었다[9-11]. 하지만, 부호화 시간은 향상된 반면 모든 영상에 대해서 고정된 탐색 패턴을 적용하기 때문에 움직임이 많은 블록의 경우 첫 번째 탐색이 잘못되었을 경우 국부 최소점(local minimum point)에 빠지게 되는 단점이 있다. 따라서, 본 논문에서는 고속 움직임 탐색기법의 단점을 보완하고 계산량을 줄이는 알고리즘을 제안하고자 한다.

3. 제안한 인터 예측과 인트라 예측 방법

H.264는 16×16 블록과 4×4 블록의 모든 모드를 계산해서 최종 모드를 선택하기 때문에 계산 복잡도가 많이 증가하게 된다. 따라서 본 장에서 제안하는 인트라 예측 방법은 공간적 상관도를 고려하여 블록의 크기를 선택하는 1단계 과정과 선택된 16×16과 4×4 블록의 주변픽셀 값의 차이를 이용하여 최종 모드를 선택하는 2단계 과정으로 나뉜다. 다음은 각 단계별 처리과정에 대해서 알아보도록 하겠다.

3.1 인트라 예측 블록 크기 결정 방법

인터 예측은 시간적인 상관성을 고려한 반면, 인트라 예측은 공간적인 상관성을 이용한다. 블록안의 픽셀 값이 주변 픽셀 값과 유사한 화소인 경우 영상 압축 기술에서 상관성이 높게 된다. 이와 반대로 블록 안의 픽셀 값이 주변 픽셀 값과 차이가 큰 경우 영상 압축 기술에서 상관성이 낮게 된다. 따라서 본 논문에서는 공간적인 상관성을 예측하기 위해 매크로블록의 분산 값을 이용하여 인트라 4×4 모드와 16×16 모드 중에서 한 모드를 선택하는 방법을 제안한다. 본 논문에서는 공간적인 상관도를 이용하였다. 매크로블록 안에 분산 값이 크면 픽셀화소의 차이가 크므로 공간적인 상관도가 낮게 된다. 이와는 반대로 분산 값이 작으면 픽셀 화소의 차이가 작은 경우이므로 공간적인 상관도가 높게 된다. 식 (3)은 분산 값의 계산식을 나타낸다. 즉, 매크로블록 내의 분산 값이 본 논문에서 실험한 임계치보다 작은 경우에는 픽셀화소 값의 차이가 작은 경우이므로 인트라 16×16 모드만 이용하고 매크로블록 내의 분산 값이 임계치보다 큰 경우에는 픽셀화소 값의 차이가 큰 경우이므로 인트라 16×16 모드와 인트라 4×4 모두 둘 다의 RDcost 값을 구한다.

$$\sqrt{Var} = \sqrt{\frac{1}{256} \left(\sum_{i=1}^{16} \sum_{j=1}^{16} p_{i,j}^2 - 256 \bar{p}^2 \right)} \quad (3)$$

3.2 인트라 예측 모드 결정 방법

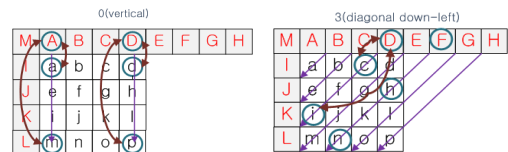
1단계를 통해서 블록 크기를 결정되면, 불필요한 블록 크기의 예측 모드에 대한 계산과정을 없앨 수 있기 때문에, 인트라 예측의 계산량을 줄일 수 있다. 하지만, 4×4 블록 및 16×16 블록은 각각 9개와 4개의 모드를 가지기 때문에 최종 모드를 선택하기 위해서는 선택된 블록 크기의 모든 모드의 RDcost를 계산해야하므로 이 과정에서 모든 모드를 조사하지 않고 최종 모드를 결정할 수 있다면 추가적인 계산량을 줄일 수 있다. 따라서 본 장에서는 1단계를 통해서 4×4블록 및 16×16 블록으로 선택되었을 경우 블록의 최종 모드의 결정을 위한 계산량을 줄이기 위한 방법을 제안하고자 한다.

블록 크기 결정 후, 4×4블록의 경우 블록 내와 블록 주변의 대표 픽셀들을 이용하여 후보 모드를 우선적으로 선택하고, 모드 선택의 정확성을 높이기 위해서 주변 픽셀의 방향성을 고려하여 후보 모드 3개를 선택한다. DC 모드는 방향성을 고려하지 않기 때문에 후보 모드로 선택하게 되고, 나머지 8개 모드 중에서 2개를 선택하게 된다.

그림 5는 주변 픽셀 중 대표 픽셀과 현재 블록의 대표 픽셀의 차이 값을 구하는 방법을 보여준다. 예를 들어, 그림 5(a)에서와 같이 vertical 모드의 비교값은 |A-a|+|A-m|+|D-d|+|D-p|로 구해지게 되고, 이 비교 값이 다른 비교 그룹에 비해 가장 작은 값이면 주변 픽셀값과의 차이가 적기 때문에 공간적인 상관성이 높아서 최종 모드로 선택될 확률이 높아지게 된다. 그림 5(b)의 경우도 마찬가지로 Diagonal Down-Left의 비교값은 |D-c|+|D-i|+|F-h|+|F-n|로 구해지게 된다. 따라서, 주변 픽셀과의 차이가 작은 2개의 모드를 후보 모드로 결정하여, DC 모드와 함께 3가지 모드에 대해서 RDcost를 계산해 주게 된다.

H.264/AVC에서 가장 많이 모드로 선택되는 vertical과 horizontal의 계산을 줄일 수 있기 때문에 계산량이 많이 줄어든다.

16×16 블록의 경우, DC 모드를 제외한 3가지 모드에 대해 수행하기 때문에 제안한 알고리즘에 따른 계산량이 많이 향상시키지 못한다. 따라서, 본 논문에서는 화질의 손실을 최소화 하기 위해서 4×4 모드에 대해서만 모드를 선택해 준다. 다음은 인트라 예측의 최종 단계를 나타내고 있다.



[그림 5] 후보 모드 결정 방법
[Fig. 5] Candidate mode decision method

[표 1] 인트라 16×16 모드를 선택한 매크로블록의 분산 값
 [Table 1] Macroblock variance value selected Intra16×16

파일명	인트라16×16선택 매크로블록 수	분산<1		분산<2		분산<3		분산<4		분산<5		분산≥5	
		개수	백분율	개수	백분율	개수	백분율	개수	백분율	개수	백분율	개수	백분율
foreman	1813	779	42.9%	1033	56.9%	1178	64.9%	1396	76.9%	1604	88.4%	218	12.1%
news	4721	2408	51.0%	2903	61.5%	3305	70.0%	3679	77.9%	4421	93.6%	300	6.4%
city	2317	756	32.6%	1094	47.2%	1340	57.8%	1458	62.9%	2006	86.5%	311	13.5%
container	4328	1992	46.0%	2510	57.9%	3076	71.1%	3673	84.8%	3991	92.2%	337	7.8%

[표 2] 인트라 4×4모드를 선택한 매크로블록의 분산 값
 [Table 2] Macroblock variance value selected Intra4×4

파일명	인트라4×4선택 매크로블록 수	분산<1		분산<2		분산<3		분산<4		분산<5		분산≥5	
		개수	백분율	개수	백분율	개수	백분율	개수	백분율	개수	백분율	개수	백분율
foreman	8087	81	1%	242	2.9%	299	3.6%	440	5.4%	607	7.5%	7480	92.5%
news	5179	36	0.7%	41	0.8%	67	1.3%	78	1.5%	161	3.1%	5018	96.9%
city	7583	229	3%	461	6.1%	603	7.9%	910	12%	1012	13.3%	6571	86.7%
container	5572	62	1.1%	170	3.0%	401	7.2%	440	7.9%	546	9.8%	5026	91.2%

case1 : 분산 < Th

- i)인트라 16×16모드 선택
- ii)4가지 모드에 대한 RDcost 계산
- iii)4개의 모드 중 가장 작은 RDcost값을 가지는 모드를 최종적으로 선택

case2 : 분산 >= Th

- i)인트라 16×16과 4×4 모드 둘 다 선택
- ii)인트라 16×16의 경우 case1을 따름
- iii)인트라 4×4 모드에 대해서 주변 픽셀 값과 현재 픽셀 값의 차이를 구함
- iii)DC 모드를 제외한 최소의 픽셀 차이값을 가지는 2개의 모드만 선택
- iV)최소의 모드 2개와 DC모드의 RDcost 계산
- V)3개의 모드 중 작은 RDcost값을 가지는 모드를 최종적으로 선택
- Vi)인트라 16×16모드와 16개의 인트라 4×4모드의 합 중 작은 RDcost값 가지는 블록 사이즈의 모드 선택

전에 프레임 간에 픽셀 값 차이에 따라 임계치를 구해서 움직임이 많은 영역에 대해 사물이 이동한 거리를 예측하여 탐색범위를 다르게 적용한다. 임계치 구하는 방법은 참고문헌 [12]에 나타나 있다.

또한, 움직임 추정 시 계산량을 줄이기 위해서 공간적으로 이웃한 블록들의 움직임벡터와 그 비용 사이에는 높은 상관관계를 가지는 특성을 이용하여 모드 선택 시 적응적으로 계산하고자 하였다. H.264/AVC에서는 현재 움직임벡터의 예측치(PMV : Predictive Motion Vector)를 계산하여 그 점이 초기 탐색점으로 사용되며 탐색 영역의 중심이 된다. 초기 탐색점을 중심으로 탐색영역 범위 안에서 미리 계산되어진 RDcost값의 중간 값, 최대 값을 계산한다. 계산되어진 중간 값과 최대 값을 임계치 T_1 , T_2 로 정한다. 현재 블록의 RDcost값이 임계치 T_1 보다 작으면 움직임이 적은 블록이므로 16×16 블록의 RDcost값만 계산해주게 된다. 또한, 현재 블록의 RDcost값이 임계치 T_2 보다 크다면 움직임이 많은 블록이므로 4×4 블록부터 16×16 블록의 RDcost값만 계산해주게 된다.

3.3 인터 예측 모드 결정 방법

추가적으로 계산량을 줄이기 위해 인트라 예측과 달리 인터 예측에서의 시간적인 상관도를 고려한 알고리즘을 제안하고자 한다. 일반적인 영상의 경우 프레임간 사물은 빠르게 변하고 배경은 거의 변하지 않는다. 그러나 H.264의 경우, 움직임이 많거나 적거나 상관없이 움직임 추정을 위한 동일한 탐색 범위를 적용시키기 때문에 시간복잡도가 증가하는 문제점이 있었다. 따라서, 시간 복잡도를 줄이기 위해서 본 논문에서는 부호화과정을 거치지

4. 실험 결과

실험은 일반적으로 영상 압축 표준안에서 많이 사용되는 QCIF 크기의 실험 영상 중 움직임이 많은 foreman, city 영상과 움직임이 적은 news, container 등의 4개를 사용하였다. 그림 6은 실험에 사용된 실험영상 4개의 첫 화면이다.



[그림 6] 4가지 QCIF 크기의 실험 영상

[Fig. 6] 4 QCIF size experimental video

본 논문에서 제안한 블록 크기 결정 알고리즘을 적용하기 위해서 공간적인 상관도를 알아보기 위해 분산의 임계값을 두어서 조사하였다. 그래서 적합한 임계치를 조사하기 위해서 인트라 16×16과 인트라 4×4 모드에 대한 매크로블록의 분산 값을 계산하였다. 다음 표 1과 표 2는 영상 100프레임(frame)에서 매크로블록의 분산 값을 나타낸 것이다.

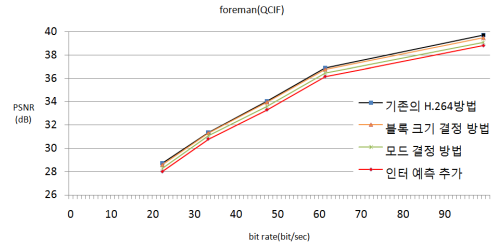
표 1과 표 2에서 알 수 있듯이, 인트라 모드를 결정할 때 분산 값이 중요한 결정요소가 될 수 있다는 것을 보인다. 분산 값이 5미만을 가질 때 인트라 모드 16×16 모드를 선택할 확률이 높다는 것을 표 1과 표 2를 통해 알 수 있고, 분산 값이 5이상을 가질 때 인트라 4×4 모드를 가질 확률이 높다는 것을 알 수 있다. 따라서 본 논문에서는 임계치를 5로 두어서 매크로블록 안의 분산 값이 5미만일 경우에는 인트라 16×16 모드만 수행하고 분산 값이 5 이상일 경우에는 16×16 모드와 4×4 모드까지 모두 수행하게 된다.

지금까지 제안한 알고리즘을 적용하여 실험하기 위해서 총 3가지로 나누어 실험 하였다.

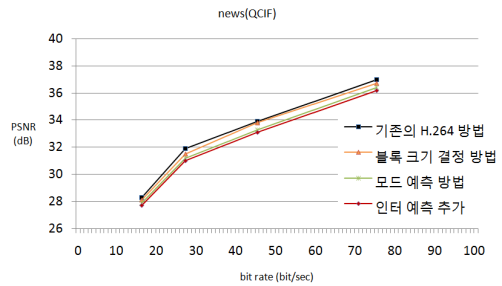
- 기존의 방법에서 인트라 예측의 블록 크기 결정 방법만 사용했을 때
- 인트라 예측의 블록 크기 결정 방법과 모드 예측 방법 사용했을 때
- 인터 예측까지 적용 했을 때

실험결과를 통한 성능 개선은 PSNR과 인코딩 시간 그래프를 통해서 나타내었다. 그림 7은 본 논문에서 제안한 인트라 예측의 2가지 방법과 인터 예측 방법을 사용 했을 때의 영상의 PSNR비교 그래프이다. 영상의 순서는 foreman, news, city, container이고, 각 영상의 Rate Distortion Curve 그래프로 나타내었다. 그림 7에서와 같이 제안한 방법의 PSNR이 H.264 방법의 결과와 비교해

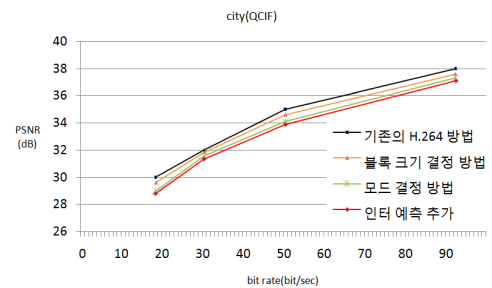
서 PSNR손실이 대략 0.5 ~ 1 dB 정도에 불과하다는 것을 알 수 있다. 이는 본 논문에서 제안하는 방법을 사용 했을 때, 화질의 차이는 약간 있지만 눈으로 구별할 수 없는 차이이다.



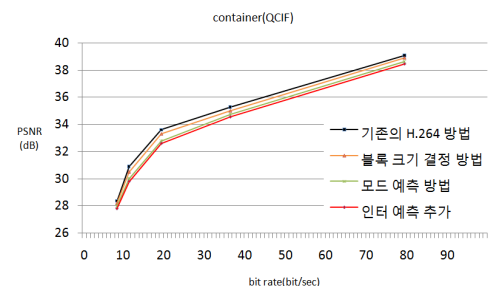
(a) foreman(QCIF) 영상의 PSNR 비교



(b) news(QCIF) 영상의 PSNR 비교



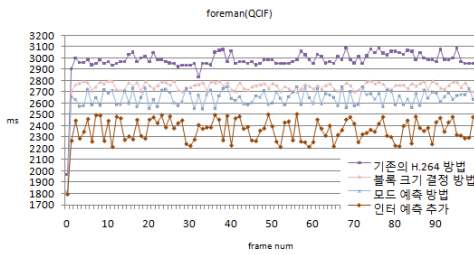
(c) city(QCIF) 영상의 PSNR 비교



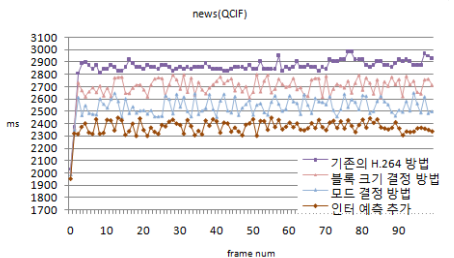
(d) container(QCIF) 영상의 PSNR 비교

[그림 7] 기존의 방법과의 PSNR 비교 그래프

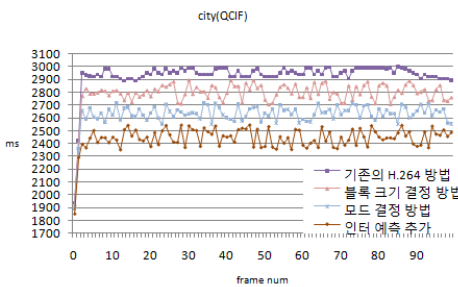
[Fig. 7] PSNR comparison graph with traditional method



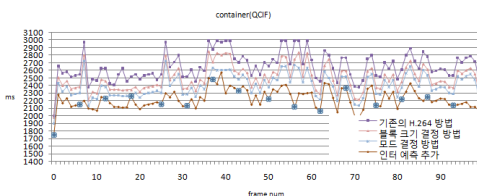
(a)foreman(QCIF)영상의 시간 비교



(b)news(QCIF)영상의 시간 비교



(c)city(QCIF)영상의 시간 비교



(d)container(QCIF)영상의 시간 비교

[그림 8] 기존의 방법과의 인코딩 시간 비교 그래프
[Fig. 8] Encoding time comparison graph with traditional method

그림 8은 기존의 방법과 제안하는 방법을 사용했을 때 각 프레임의 코딩 시간(ms)을 비교해서 나타낸 그래프이다. 네 종류의 영상을 통해 앞의 실험에 적용된 기존의 H.264 방법과 제안한 인트라 예측의 2가지 방법과 본 논

문에서 제시한 인트라 예측 방법을 추가하여 비교하였다. 결과적으로 제안한 방법이 각 프레임마다 최소 480ms에서 최대 620ms정도 빠르게 부호화됨을 확인 할 수 있었다.

5. 결론

H.264/AVC에서는 인트라 예측과 인트라 예측을 위해 16×16에서 4가지 모드와 4×4에서 9가지 모드에 대해서 RDcost를 수행해 주게 된다. 그래서 압축효율은 많이 높아졌지만 인트라 예측의 여러 가지 모드들로 인해 계산이 매우 복잡해 졌고, 부호화 압축시간도 많이 증가하게 되었다. 이와 달리 본 논문에서는 공간적인 상관성을 이용하여 공간적인 상관성이 높은 범위에서는 16×16 모드만 수행하고 9가지 모드 중에서도 예측픽셀 값과의 차이를 비교하여 부호화 계산량을 줄이고자 하였다 추가적으로 인트라 예측에서의 움직임 추정 방법에서 계산량을 줄이는 방법을 제안하였다.

실험 결과 제안하는 방법이 기존의 H.264의 인트라 예측 및 인트라 예측 방법과 비교하여 실험 영상들의 경우 모든 프레임이 평균적으로 화질은 01 ~ 0.2 dB 정도 떨어지면서 평균적으로 500 ~ 600ms 정도의 부호화 시간이 단축되었다. 제안된 방법은 실시간 영상 코딩 응용 및 소프트웨어 구현에 도움이 될 것으로 예상된다.

References

- [1] J. R. Smith, "The H.264 Video Coding Standard", Multimedia, IEEE, Vol. 13, pp. 86-90, 2006.
- [2] A.Ahmad, N. Khan, S. Masud, M. A. M명, "Efficient block size selection in H.264 video coding standard", Electronics Letters, Vol. 40, pp. 19-21, January, 2004.
- [3] H. Zeng, C. Cai and K.K. Ma, "Fast Mode Decision for H.264/AVC Based on Macroblock Motion Activity," Circuits and Systems for Video Technology, IEEE Trans. Vol. 19, pp. 491-499, April, 2009.
- [4] L. J. Pan, and Y. S. Ho, "Fast mode decision algorithm for H.264 inter-prediction," Electronics Letters, Vol. 43, pp.1351-1353, Nov.2007.
- [5] F. Pan, X. Lin, S. Rahardja, K. P. Lin, Z. G. Li, D. Wu, and S. Wu, "Fast Mode Decision Algorithm for intra prediction in H.264/AVC Video Coding", Trans. on Circuit and Systems for Video Technology, vol. 15 no.7 pp.813-822, July 2005.
- [6] B. D. La, M. Y. Eom, and Y. S. Choe, "Fast Mode

Decision for Intra Prediction in H.264/AVC Encoder", Proc of ICIP, Vol.5, pp.321-324, 2007.

[7] Xuanxing Xiong, Yang Song and Ali Akoglu, "Architecture design of variable block size motion estimation for full and fast search algorithms in H.264/AVC," In Elsevier. Vol. 4, Feb, 2011.

[8] T. Wiegand, J. Sullivan, "Overview of the H.264/AVC Video Coding Standard", IEEE Trans.Circuits and Systems for Video Technology, Vol 13, No. 7, pp.560-576, Ju;y, 2003.

[9] J. N. Kim, S. C. Byun, Y. H. Kim and B. H. Ahn, "Fast full search motion estimation algorithm using various matching scan in video coding," Systems, Man, and Cybernetics, Part C, Applications and Reviews, IEEE Trans. Vol. 31, pp. 540-548, 2001.

[10] X. Xiong, Y. Song and A. Akoglu, "Architecture design of variable block size motion estimation for full and fast search algorithms in H.264/AVC," In Elsevier. Vol. 4, Feb, 2011.

[11] Parlak, M. Adibelli, Y. and Hamzaoglu, I. "A novel computational complexity and power reduction technique for H.264 intra prediction," Consumer Electronics, IEEE Trans. Vol. 54, pp. 2006-2014, Nov, 2008.

[12] K. S. Park, M. J Kim, J. H. Jung, J. H. Cho, S. H. Kim "Enhanced H.264 Coding Method Applied to Adaptive Block Search Algorithm", Journal of Computing Science and Engineering, software and application, Vol 12, No. 38, pp.663-670, December, 2011.

박 경 석(Kyung-Seok Park)

[정회원]



- 2006년 2월 : 대구대학교 전산통계학과 졸업(학사)
- 2008년 2월 : 경북대학교 컴퓨터공학과 졸업(석사)
- 2008년 3월 ~ 현재 : 경북대학교 모바일통신공학과 박사과정

<관심분야>
멀티미디어, MPEG, H.264, IPTV 등

김 민 준(Min-Jun Kim)

[정회원]



- 2005년 2월 : 경북대학교 컴퓨터공학과 졸업(학사)
- 2007년 2월 : 경북대학교 컴퓨터공학과 졸업(석사)
- 2007년 2월 ~ 현재 : 경북대학교 전자전기컴퓨터학부 박사과정

<관심분야>
멀티미디어, 다시점 동영상, MPEG, IPTV 등

전 재 현(Jae-Hyun Jun)

[정회원]



- 2009년 2월 : 대구가톨릭대학교 전자공학과 졸업(학사)
- 2011년 2월 : 경북대학교 컴퓨터공학과 졸업(석사)
- 2011년 3월 ~ 현재 : 경북대학교 전자전기컴퓨터학부 박사과정

<관심분야>
멀티미디어, 다시점 동영상, 동기식 이더넷, DDoS 탐지, 트래픽 분류

류 상 료(Sang-Ryul, Ryu)

[정회원]



- 1988년 2월 : 경북대학교 전자공학과 졸업(학사)
- 1990년 2월 : 경북대학교 컴퓨터공학과 졸업(석사)
- 1987년 8월 : 경북대학교 컴퓨터공학과 졸업(박사)
- 1998년 3월 ~ 현재 : 청운대학교 컴퓨터학과 교수

<관심분야>
알고리즘 분석 및 설계, 이미지 프로세싱

김 승 호(Sung-Ho Kim)

[정회원]



- 1981년 2월 : 경북대학교 전자공학과 졸업(학사)
- 1983년 2월 : 한국과학기술원 전산학과 졸업(석사)
- 1994년 2월 : 한국과학기술원 전산학과 졸업(박사)
- 1985년 3월 ~ 현재 : 경북대학교 컴퓨터학부 교수

<관심분야>

알고리즘, 감시시스템, 멀티미디어, 다시점동영상