

# H.264 to MPEG-2 Transcoding을 위한 효율적인 P-Frame 변환 방법

김기홍<sup>†</sup> · 손남례<sup>\*\*</sup> · 이귀상<sup>\*\*\*</sup>

## 요 약

MPEG-2가 발표된 이후 Digital-TV나 DVD등 멀티미디어 분야에서 폭넓게 사용되어 왔다. 그 후 2004년도에 H.264가 발표된 이후 MPEG-2를 대체할 차세대 비디오 압축 표준으로 각광받으며 IPTV나 DMB등의 방송 표준으로 채택되었다. 그러나 지금까지 MPEG-2를 가장 많이 사용해 왔기 때문에 현재는 MPEG-2관련 장비들이 많이 사용되고 있으며 이 장비들로는 H.264방송을 시청할 수 없게 되었다. 본 논문에서는 효율적인 H.264 to MPEG-2 트랜스코더를 제안한다. 또한 화질의 열화를 줄이기 위해 기본적으로 직렬 화소 영역기반 트랜스코더(CPDT: Cascaded Pixel Domain Transcoder)구조를 이용하였고 변환속도를 높이기 위해 복호기에서 사용된 SKIP블록이나 INTRA 블록, 움직임벡터 등의 정보들을 재사용 한다. H.264의 가변블록의 움직임벡터들중 최적의 값을 선택하기 위해 수정된 경계정합알고리즘(BMA: Boundary Matching Algorithm)을 사용한다. 실험결과, MPEG-2 인코더의 'Full Search'와 비교하였을 때 PSNR측면에서는 0.1dB정도 감소되었지만, 부호화 시간에서는 약 66% 개선되었다. 제안한 방법은 기존방법과 비교하였을 때, 우수한 화질을 확보함과 동시에 연산시간을 단축할 수 있음을 확인하였다.

키워드 : H.264, MPEG-2, 움직임 벡터, 트랜스코더, P-frame

## An Effective P-Frame Transcoding from H.264 to MPEG-2

Kim Gi Hong<sup>†</sup> · Nam Rye Son<sup>\*\*</sup> · Lee Guee Sang<sup>\*\*\*</sup>

## ABSTRACT

After the launch of MPEG-2, it is widely used in multimedia applications like a Digital-TV or a DVD. Then, After the launch of H.264 at 2004, it has been expected to replace MPEG-2 and services IPTV and DMB. As we have been used to MPEG-2 devices by this time, we can not access H.264 Broadcast with MPEG-2 device. So We propose a new approach to transcode H.264 video into MPEG-2 form which can facilitate to display H.264 video with MPEG-2 device. To reduce the quality loss by transcoding, we use CPDT(Cascaded Pixel Domain Transcoder) structure. And to minimize processing time, SKIP block, INTRA block and motion vectors obtain from decoding process is employed for transcoding. we use BMA(Boundary Matching Algorithm) to select only one from candidate motion vectors. Experimental results show a considerable improved PSNR with reduction in processing time compared with existing methods.

Keywords : H.264, MPEG-2, Motion Vector, Transcoder, P-frame

## 1. 서 론

1990년대 초반부터 H.261에서부터 MPEG-4에 이르기까지 다양한 비디오 압축 표준들이 발표 되었다. 그중에서도 MPEG-2는 Digital-TV와 DVD등의 표준으로 채택되는 등

기존의 비디오 표준들에 비해 넓은 영역에서 활발하게 이용되어 왔고, 현재 가장 많이 쓰이는 비디오 표준이다. 그 후 ITU-T와 ISO/IEC에서 JVT(Joint Video Team)라는 팀을 만들고 2001년부터 개발에 착수하여 2004년에 H.264라는 비디오 표준안을 발표하였다. H.264는 압축률을 높이기 위해 INTER프레임에서 여러 가지 방법들을 적용하였으며 I-프레임에서도 INTRA 예측을 통하여 부호화를 수행한다. 이로 인해 기존의 MPEG-2와 비교할 때 낮은 비트율에서도 MPEG-2와 거의 동등한 화질을 보인다[1]. H.264에는 Baseline, Main, Extended등 3종류의 프로파일이 존재한다. Baseline 프로파일은 네트워크 환경에서의 사용을 위해 최소의 복잡도와 높은 에러내성, 유연성을 강조하여 정의되었다. 우리나라

\* 이 논문은 2009년도 전남대학교 특성화분야 연구비지원 및 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(NIPA-2009-(C1090-0903-0008)).  
† 준 회원: 전남대학교 전산학과 석사과정  
\*\* 준 회원: 목포대학교 중점연구소 연구교수  
\*\*\* 종신회원: 전남대학교 전자컴퓨터공학부 교수  
논문접수: 2009년 7월 23일  
수정일: 1차 2009년 10월 29일  
심사완료: 2009년 10월 30일

라의 지상파 DMB(Digital Multimedia Broadcasting)의 표준으로도 사용되고 있다. Main 프로파일은 높은 압축능력을 강조하여 정의되었고, Extended 프로파일은 스트리밍 서비스를 목표로 하여 정의 되었다. 현재 H.264는 DMB와 IPTV(Internet Protocol Television)의 규격으로 채택되었으며, MPEG-2를 대체할 차세대 비디오 표준으로서 성장하고 있는 추세이다. 그러나 기존에 사용되고 있는 MPEG-2장비들이 교체되기까지는 상당한 시간이 걸릴 것으로 예상된다. 따라서 본 논문에서는 H.264방송을 기존의 MPEG-2장비에서 재생할 수 있는 트랜스코더를 제안한다.

트랜스코더는 크게 두 종류로 나눌 수 있다. 첫 번째는 동종간 트랜스코더로서, 서버에 저장된 영상스트림을 비트율이나 화면의 크기 등을 조절하여 성능이 낮은 클라이언트로 전송할 때 쓰인다. 두 번째는 이종 간 트랜스코더로서 서버에서 규격이 다른 디바이스로 영상 스트림을 전송하기 위해 비디오 표준을 다른 표준으로 변환할 때 사용한다.

이종 간 트랜스코더는 다시 몇 가지 방법으로 나눌 수 있다. 첫 번째 방법은 CPDT라고 불리는 방법이다. CPDT는 복호기와 부호기가 직렬로 연결되어 있으며 입력영상을 YUV형태로 완전히 복호한 뒤 다시 부호화하는 형태이다. 가장 간단한 형태이며 트랜스코더들 중 가장 화질이 좋다는 장점이 있다. 그러나 부호화하는 과정에서 다시 움직임 보상연산을 수행하기 때문에 연산량이 많아진다는 단점이 있다[2, 3].

두 번째 방법으로는 주파수 영역 기반 트랜스코더(DDT: DCT Domain Transcoder)방식이 있다. 이 방법은 움직임 추정 연산이 DCT영역에서 수행된다. 즉, 복호기의 IDCT부분부터 부호기의 DCT부분까지의 과정을 생략했기 때문에 CPDT방식에 비해 더 빠른 수행속도를 보인다. 그러나 IDCT와 DCT과정을 거치지 않으므로 참조프레임과의 오차가 발생하고, 이 오차가 누적되면서 화질의 열화가 발생한다[4].

H.264표준이 발표되면서 MPEG-2표준으로 부호화된 콘텐츠들을 재사용하기 위하여 MPEG-2 to H.264 트랜스코더가 이슈가 되었다. H.264와 MPEG-2표준의 부호화 차이로 움직임추정 및 보상, INTRA픽처 예측, 적응적인 필드 및 프레임코딩, 트랜스폼 코딩, 슬라이스 스트럭처, 디블러징 필터 등에 관한 연구가 진행중이다[5, 6]. 그러나 본 논문에서는 Baseline프로파일 H.264방송을 기존에 사용하던 MPEG-2 디바이스를 이용해서 시청이 가능하도록 하는 H.264에서 MPEG-2로의 트랜스코더를 제안한다. 이때 화질의 열화를 줄이기 위해서 CPDT방식을 사용하고, 중복된 움직임 보상연산으로 인한 CPDT의 느린 수행속도를 개선하기 위해 복호기측에서 사용한 움직임벡터 및 매크로블록 정보를 부호기에서 재사용하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 H.264 to MPEG-2 트랜스코더에 관한 연구를 살펴보고, 3장에서는 본 논문에서 제안한 수정된 경계정합알고리즘을 이용한 P-프레임 변환 방법에 대해서 소개한다. 4장에서는 제안된 방법의 실험결과 및 기존에 제안된 방법들과 비교, 분석한다. 마지막으로 5장에서는 본 논문의 결론을 맺는다.

## 2. 관련 연구

CPDT에서 가장 중요한 부분은 부호기에서 움직임 추정 시간을 최대로 줄이는 방법이 중요하다[7]. 그 이유는 부호기의 총 연산량 중 60%이상을 움직임 추정 연산이 차지하기 때문이다. 이 움직임 추정연산을 줄이는 방법은 복호기측에서 사용한 정보들을 부호기측에서 재사용하여 줄일 수 있다. 다음은 H.264 to MPEG-2 transcoding 관련된 논문들로 특히 움직임 벡터 재사용에 관련된 논문이다.

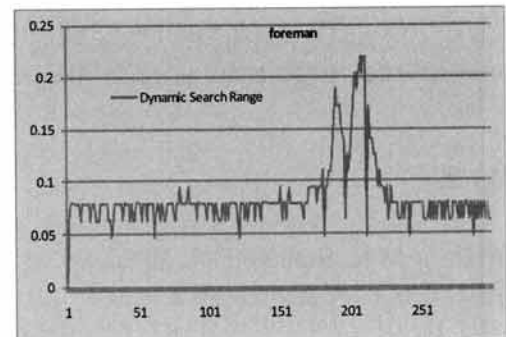
### 2.1 블록 크기에 따른 가중치 평균법[2]

J. Chu가 제안한 '평균법[8]'을 이용한 트랜스코딩은 변환 속도는 빠르나 PSNR이 약 3dB이상 떨어지는 등 단점이 존재한다. 이에 S. Sharma는 H.264의 매크로블록의 크기를 고려하지 않았던 '평균법[8]'의 단점을 보완하는 방법으로 '블록 크기에 따른 가중치 평균법'을 제안하였다. 이 방법은 H.264의 매크로블록들의 크기에 따라서 나누는 값을 달리하여 가중치 평균을 구하고, 이 위치를 기준으로 MPEG-2의 'Full Search'를 수행하는 방법이다. 이 방법은 기본적으로 평균법을 이용하였기 때문에 변환속도가 빠르다는 장점이 있으나 300장 정도의 프레임으로 테스트했을 때 PSNR이 2dB가까이 떨어지는 단점이 존재한다.

### 2.2 가변적인 탐색 영역법(Dynamic Search Range)[9]

이 방법은 한 매크로블록내의 움직임 벡터들 중 절대 값이 가장 큰 값을 'Full Search'의 범위로 움직임벡터를 추정하는 방법이다.

'가변적인 탐색 영역법'은 구현이 간단하고 기본적으로 MPEG-2의 'Full Search'를 이용하기 때문에 PSNR결과가 상당히 좋은 방법이다. 그러나 (그림 1)과 같이 동작이 큰 영상 즉, 움직임 벡터가 큰 영상에서는 탐색 범위가 넓어지므로 연산량이 증가한다는 단점이 있다.



(그림 1) '가변적인 탐색 영역법'을 이용한 프레임 별 변환 속도

### 2.3 경계정합알고리즘(BMA)

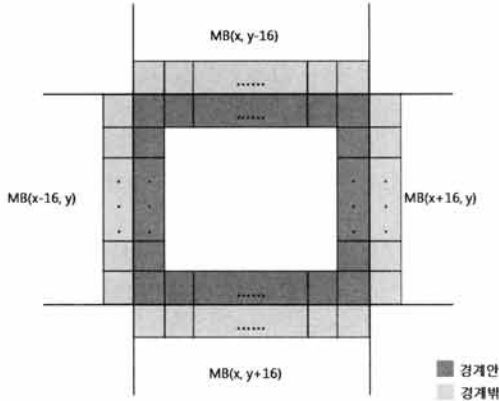
무선통신과 같은 환경은 주변 잡음에 민감하므로 에러가 많이 발생하게 된다. 이때 에러가 발생된 매크로 블록을 복원하기 위해서 사용되는 방법이 에러 은닉 방법이다. 그리

고 경계정합알고리즘은 에러은닉분야에서 많이 사용되는 방법이다. 경계정합알고리즘의 기본적인 방법은 (그림 2)에서 보는 바와 같이 손상된 매크로블록의 주변 화소를 미리 정의된 후보움직임벡터들에 따라 참조영상과 매칭을 수행한다. 그 후, 한 개의 움직임 벡터를 선택하고 이를 이용하여 손상된 매크로 블록을 최대한 비슷하게 복원하는 방법이다[10].

경계정합알고리즘에서 사용하는 후보 움직임벡터군은 아래와 같다.

- a) 손상된 블록과 같은 위치의 이전 프레임의 매크로블록의 움직임 벡터.
- b) 손상된 블록 주변의 움직임벡터들의 평균.
- c) 손상된 블록 주변의 움직임벡터들의 중간값.
- d) 손상된 블록 주변의 움직임벡터.
- e) Zero Vector

사용할 수 있는 후보 움직임벡터들이 존재하고 이 값들 중 한 개의 움직임 벡터를 선택한다는 점에서 본 논문에 적용할 수 있다고 판단하였다. 따라서 본 논문에서는 H.264에서 MPEG-2로의 P-프레임 변환에 이 방법을 수정하여 사용한다.



(그림 2) 경계정합알고리즘

### 3. 제안하는 방법

본 논문에서는 H.264 스트림을 MPEG-2로 변환하는 과정에서 P-프레임을 효율적으로 변환하기 위한 방법을 제안한다. 기존연구와는 다르게 본 논문에서는 INTER 모드에서 주변 움직임벡터의 평균이나 Full Search 에 의한 탐색범위 조정은 달리 BMA를 사용하여 후보 벡터 중의 가장 좋은 벡터를 찾는 방법을 사용한다[10, 11]. BMA를 사용함으로써 Full Search보다 연산시간을 줄이면서 동시에 우수한 화질을 확보하고자 한다. 또한 기존연구에서는 H.264 INTRA 블록은 모두 MPEG-2 INTRA 블록으로 부호화하였지만, 본 논문에서는 H.264 INTRA 블록의 경우에도 MPEG-2에서 INTER 모드와 INTRA 모드 부호화량을 비교하여 최소 부호화량을 갖는 모드를 선택하게 함으로써 화질을 높게 하

였다. 그리고 SKIP 모드에 있어서도 H.264에서는 이전 프레임과 동일한 위치의 블록인 경우뿐 아니라 다른 위치의 블록에 대해서도 SKIP 모드를 적용하였으나 MPEG-2에서는 반드시 동일한 위치의 블록에 대한 SKIP 모드만을 인정한다. 따라서 본 논문에서는 이전 프레임의 다른 위치에 대한 H.264 SKIP 모드 블록의 경우에는 MPEG-2에서는 INTER모드로 전환하였다. 아래의 <표 1>은 H.264내 P슬라이스의 정보들을 나타내고 있다.

<표 1> 트랜스코딩에 사용할 H.264정보.

Frame(slice) Type	I Frame, P Frame
Macroblock Type	Inter, Intra
Macroblock Mode	16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4
SKIP Mode	고정 SKIP, 움직임 SKIP

#### 3.1 INTER 모드

BMA를 적용하기 위하여 H.264의 16x16 크기의 블록 안에 존재하는 각 움직임벡터마다 화소의 차분값을 산정해야 한다. MPEG-2의 매크로 블록 형태는 16x16크기의 한 가지 형태만을 이용한다. 그러나 H.264에는 매크로블록 형태가 총 7가지(16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4)가 존재한다. 그러므로 MPEG-2에서는 1개의 움직임 벡터만을 갖지만 H.264에는 최대 16개의 움직임 벡터를 갖게 된다. 본 논문에서는 H.264 INTER 매크로블록이 16x16인 경우와 그렇지 않은 경우로 구분하여 변환한다.

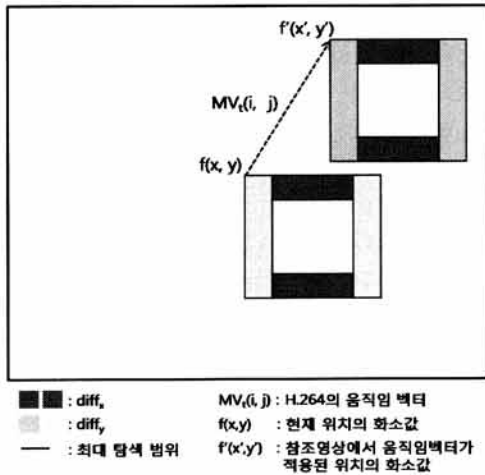
##### 3.1.1 16x16 매크로블록.

16x16크기의 매크로블록은 두 비디오 포맷 모두 사용하고 있으며 연산과정도 비슷하다. 이로 인해 H.264의 16x16 매크로블록의 움직임 벡터는 MPEG-2의 움직임 벡터와 유사한 경우가 상당한 많다. 따라서 본 논문에서는 H.264의 매크로블록이 16x16인 경우 별다른 연산을 하지 않고 그대로 적용하였다.

##### 3.1.2 16x16을 제외한 나머지 매크로블록

H.264에서 16x16이외의 매크로블록들은 2개~16개의 움직임벡터를 갖고 있고, 이 움직임 벡터들은 MPEG-2에서 얻은 움직임 벡터와 거의 유사한 움직임 벡터들이 존재한다. 따라서 본 논문에서는 이 움직임 벡터들을 찾기 위해 수정한 BMA를 이용하여 최적의 움직임벡터를 선택하는 방법을 제안한다. 이때 (그림 3)과 같이 현재 매크로블록내의 움직임 벡터들마다 매크로블록 내부의 경계면 영역과 참조영상과의 MAD를 통해 식(1)과 식(2)의 오차 값을 계산한다. 그리고 식(3)과 같이 오차 값이 최소인 움직임벡터를 MPEG-2의 움직임벡터로 적용한다.

$$diff_y = \sum_{n=0}^{15} |f(x, y+n) - f'(x', y'+n)| + |f(x+15, y+n) - f'(x'+15, y'+n)| \tag{1}$$



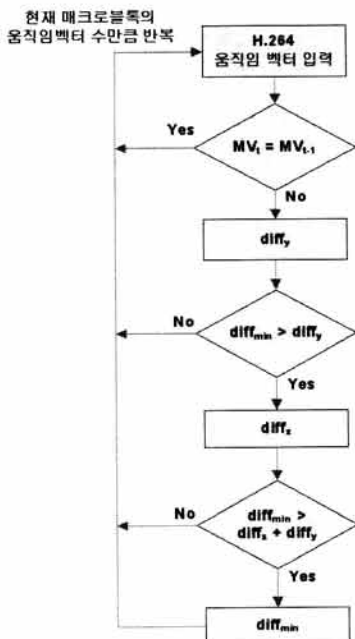
(그림 3) 제안한 경계정합알고리즘

$$diff_x = \sum_{n=1}^{14} |f(x+n, y) - f'(x'+n, y')| + |f(x+n, y+15) - f'(x'+n, y'+15)| \quad (2)$$

$$diff_{min} = MIN(diff_x + diff_y) \quad (3)$$

또한 본 논문에서 연산량을 최소화하기 위하여 아래와 같은 방법들이 수행된다. 자세한 흐름도는 (그림 4)에 나타나 있다.

- (1) H.264의 움직임벡터들 중에는 중복되는 값들이 상당수 존재한다. 그러므로 이 중복되는 값들은 연산을 하지 않고 다음 움직임벡터로 넘어감으로서 수행시간을 단축시킬 수 있다. 그러나 중복되는 값들을 찾기 위해 정렬 알고리즘을 사용할 경우, 모든 움직임벡터



(그림 4) 제안한 경계정합알고리즘의 연산 과정

들을 계산하는 방법과 수행시간의 차이가 거의 없다. 따라서 바로 앞의 움직임 벡터와 현재 값만을 비교하여 연산을 수행한다.

- (2) 경계정합알고리즘은 매크로블록의 외곽영역간의 오차를 구해서 이 오차가 가장 적은 움직임벡터를 찾아내는 방법이다. 그러므로 이 오차를 구하는 과정에서 오차 값이 앞에서 계산된 가장 작은 값보다 크면 연산을 수행할 필요가 없다. 본 논문에서는 좌, 우 수직 화소들을 먼저 계산한 후 즉, 식(1)을 먼저 계산하고 이 값을 현재 계산된 최소값과 비교한다. 식(1)에서 계산된 오차 값이 최소값 보다 더 클 경우 연산을 중단하고 다음 움직임 벡터를 실행한다. 반대로 작을 경우 식(2)를 계산한 후, 최소값과 비교하는 방법을 사용한다.

### 3.2 INTRA 모드.

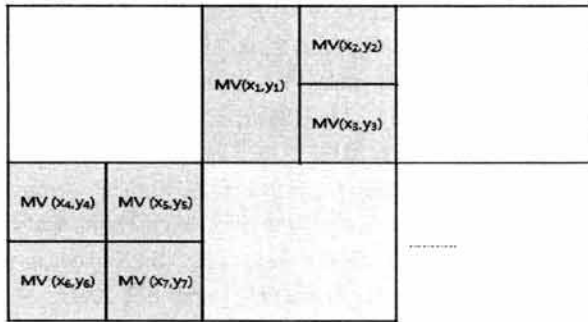
H.264와 MPEG-2는 부호화방법이 달라서 H.264의 INTRA 블록이 반드시 MPEG-2에서도 INTRA 모드로 설정되어야 하는 것은 아니다. 기존 방법에서는 이러한 구분을 두지 않고 모든 H.264 INTRA 블록은 MPEG-2 INTRA 블록으로 변환하였다. <표 2>에서 볼 수 있듯이 영상에 따라 차이는 있지만 매크로블록이 H.264에서 INTRA모드일 때 MPEG-2에서도 INTRA모드인 경우는 드물다는 것을 알 수 있다. 실제로 같은 영상의 H.264 버전과 MPEG-2 버전을 비교해보면 H.264 INTRA 블록이 MPEG-2 INTRA 블록으로 변환되는 것은 아니라는 것을 알 수 있다. <표 2>는 H.264 INTRA 블록 중에서 MPEG-2 INTRA 모드로 부호화된 블록의 비율을 보인 것이다. 여기서 H.264의 16x16 크기를 갖는 INTRA 블록만을 고려한 것이다.

본 논문에서는 H.264 INTRA 블록에 대해서도 INTER모드 부호화 가능성을 비교하여 INTER/INTRA 중에서 보다 효율적인 모드를 결정한다. INTRA모드는 움직임 보상을 적용한 오차값이 움직임 보상을 적용하지 않은 오차값 보다 클 때 발생한다. H.264 INTRA 블록에 대해 MPEG-2의 INTER/INTRA 모드결정을 위해 움직임벡터를 찾아 부호화량을 비교하여야 하며, 이를 위해서는 적절한 움직임벡터를 빠른 시간 내에 찾아야 하며 이를 위해 BMA를 수행한다. 본 논문에서 제안한 주변매크로블록의 움직임 벡터는 (그림 5)와 같이 MPEG-2에서 부호화할 매크로블록의 상단과 좌측 블록들의 움직임 벡터를 사용한다. 만약 주변에 인접한 상단 및 좌측 매크로블록이 존재하지 않거나 또는 INTRA 블록인 경우, 하단 및 우측의 매크로블록의 움직임벡터를 이용하여 제안한 BMA를 수행한다. 그리고 최종적으로 움직임 벡터가 결정되면, 이 위치를 기준으로 MSD(Mean Square Difference)를 이용하여 반화소 탐색을 실행하고, 이 오차값

<표 2> H.264와 MPEG-2의 INTRA 모드 일치율(%)

	FOREMAN	SOCCER	COASTGUARD	STEFAN	BUS
INTRA 모드 일치율	38.97	57.11	58.17	24.12	10.48





(그림 5) INTRA모드를 변환하기 위한 후보 움직임벡터의 위치

을 기준으로 INTRA모드로 할 것인지 INTER모드로 할 것인지 선택하게 한다.

### 3.3 SKIP 모드

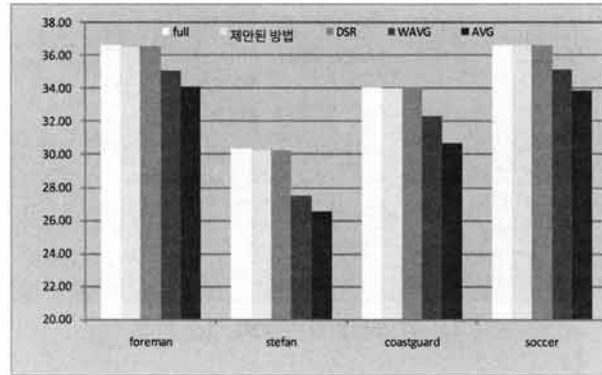
SKIP 모드란 현재 매크로블록과 참조 영상의 화소 값이 같을 경우, 이 매크로블록의 정보를 보내지 않고 움직임 벡터를 '0'으로 하여 이 화소 값을 그대로 사용하는 방법으로 말한다. 그러나 H.264에는 기존의 비디오 표준들과는 달리 SKIP 모드가 고정 위치 SKIP 형태와 움직임 SKIP 형태의 두 가지 방법을 사용한다[12]. 첫 번째 방법은 다른 비디오 표준들과 마찬가지로 움직임 벡터를 '0'으로 하여 현재 위치의 화소를 그대로 사용하는 방법이다. 두 번째 방법은 주변의 움직임 벡터를 적용하여 그 위치의 화소를 그대로 사용하는 방법이다. 이 방법을 함께 사용하면 고정 위치 SKIP 모드만을 사용했을 때보다 더 나은 효율을 갖게 된다[1]. 본 논문에서는 연산량을 줄이기 위해 H.264의 매크로블록이 고정 위치 SKIP 모드인 경우에는 MPEG-2에서도 움직임벡터를 '0'으로 하였다. 그리고 움직임 SKIP모드인 경우에는 16x16 매크로블록으로 취급하여 움직임 벡터를 그대로 적용한다.

## 4. 실험환경 및 결과

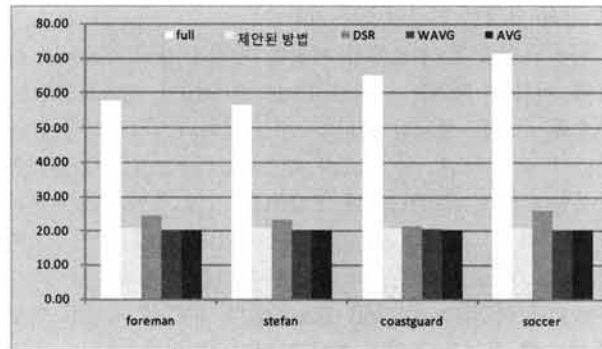
### 4.1 실험 환경

본 논문에서는 Baseline 프로파일 H.264방송을 기존에 사용하던 MPEG-2 디바이스를 이용해서 시청이 가능하도록 하는 H.264에서 MPEG-2로의 트랜스코더를 제안하였다. 실험환경은 AMD 4600 CPU, 1G RAM의 PC에서 VC++ 2005를 이용하였고, 테스트 영상은 CIF크기의 영상들을 가지고 각 300프레임씩 테스트하였다. H.264스트림은 H.264 JM 10.2 인코더를 이용하여 메인 프로파일, 1Mbits/sec의 비트율, B프레임을 제외한 15프레임의 GOP 크기의 환경으로 인코딩 하였다. 트랜스코더의 구성은 JM 10.2 H.264 디코더와 MPEG-2 표준 인코더('Full Search')를 사용하였고, Full Search의 범위는 -15 ~ 15 pixel, GOP 크기와 비트율은 H.264 인코딩환경과 동일하다.

(그림 6)과 (그림 7)은 'Full Search', '가중치평균법(WAVG)[2]', '평균법(AVG)[8]', '가변적인 탐색영역법(DSR)[9]', 제안



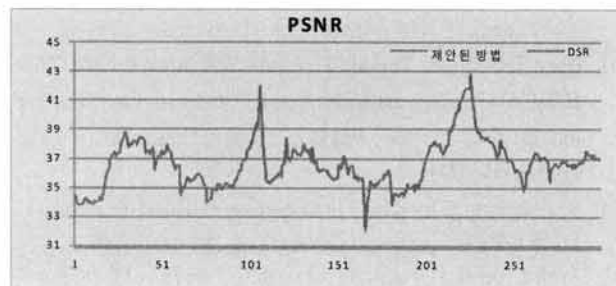
(그림 6) PSNR 비교



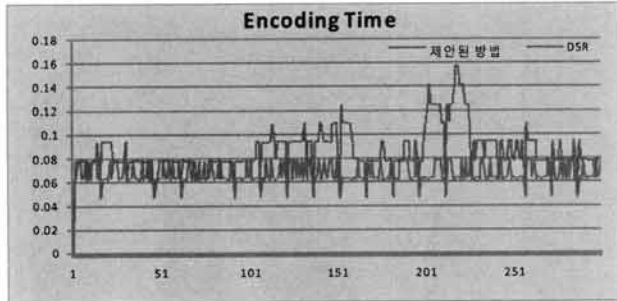
(그림 7) 부호화 시간 비교

한 방법을 비교한 PSNR 및 계산시간 결과이다. PSNR측면에서 기존 방법인 '가변적인 탐색영역법', '가중치평균법', '평균법'은 'Full Search'에 비해 평균적으로 각각 0.08dB, 1.95dB, 3.11dB 정도 감소하였다. 그리고 부호화 시간 측면에서는 'Full Search'에 비해 평균적으로 각각 61.91%, 67.43%, 67.49% 정도 감소하였다. 기존 방법 중 성능이 우수한 '가변적인 탐색영역법'은 'Full Search'와 거의 비슷하나, 이 방법은 움직임이 큰 영상 즉, 움직임벡터가 큰 값을 가진 매크로블록은 탐색 범위가 커지므로 연산량이 증가하는 경우가 발생한다.

다음 (그림 8)과 (그림 9)는 'SOCCER' 영상에 대한 PSNR 및 부호화 시간을 '가변적인 탐색영역법'과 비교한 것이다. PSNR 측면에서는 거의 차이가 없으나 부호화 시간은 제안한 방법이 약 6%이상 감소되었다. 또한 'SOCCER' 영상은 100~150번 프레임과 200~250번 프레임에서 카메라가 급격하게 이동하므로 '가변적인 탐색영역법'은 움직임 벡터가 커



(그림 8) 프레임별 PSNR 비교



(그림 9) 프레임별 부호화 시간 비교

지면서 탐색영역이 넓어지게 되고 연산량이 증가된다. 반면에 제안된 방법은 움직임 벡터의 수에 따라 영향을 받으므로 거의 일정한 연산량을 유지한다.

마지막으로 본 논문에서 제안한 방법은 'Full Search' 방법에 비하여 PSNR는 평균적으로 0.09dB가 감소하였지만, 부호화 시간은 움직임 벡터가 큰 영상이나 작은 영상에 상관없이 평균적으로 약 66%가 감소되었다. 그리고 '가변적인 탐색영역법'과 비교했을 때, PSNR는 거의 차이가 없으면서도 부호화 시간은 약 5%정도 감소되었다. 그리고 '가중치 평균법' 및 '평균법'의 부호화 시간은 제안한 방법과 거의 비슷하였으나 PSNR는 2~3dB 감소하였다. 따라서 제안한 방법이 우수한 성능을 가지고 있다고 말할 수 있다.

### 5. 결 론

본 논문에서는 H.264 to MPEG-2 Transcoder에 적용할 수 있는 P프레임 변환 방법을 제안하였다. H.264에서 SKIP 블록인 경우에는 고정 위치 SKIP형태와 움직임 SKIP형태로 구분하여 MPEG-2에 적용하였다. 매크로블록의 모드가 16x16인 경우에는 움직임 벡터를 그대로 적용하였고, 그 외 16x8, 8x16, 8x8, 8x4, 4x8, 4x4 모드와 INTRA모드인 경우에는 경계정합알고리즘을 적용하였다. 실험 결과, 'Full Search'와 비교하였을 때 PSNR에서는 약 0.1dB정도 감소되었으나 부호화 시간에서는 약 66%이상 감소되었다. 또한 '가변적인 탐색영역법'과 비교 하였을 때에도 PSNR는 거의 차이가 나지 않으면서도 부호화 시간은 약 5%감소되었고, '가중치 평균법'이나 '평균법'을 이용한 방법과도 비교하였을 때 크게 개선되었음을 확인할 수 있었다.

### 참 고 문 헌

[1] Gary J. Sullivan, Pankaj Topiwala, and Ajay Luthra, "The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions," SPIE Conference, 2004.  
 [2] S. Sharmat, K.R. Rao, "Transcoding of H.264 bitstream to MPEG-2 bitstream," APCC 2007, pp.391-396, 2007.  
 [3] 신익경, "비디오 트랜스코더에서 움직임벡터를 이용한 효율적 매크로블록 모드 결정 기법 연구," 석사학위 논문, 2003.

[4] 김동형, 정제창, "MPEG-2에서 H.264로의 트랜스코딩 기법," 정보통신학회논문지, 30권, 7호, pp.706-715, 2005.  
 [5] Qiang Tang; Nasiopoulos, P.; Ward, R.; "Fast block size prediction for MPEG-2 to H.264/AVC transcoding," ICASSP 2008, pp.1029-1032, 2008.  
 [6] Fernandez-Escribano, G.; Kalva, H.; Cuenca, P.; Orozco-Barbosa, L.; "Speeding-Up the Macroblock Partition Mode Decision in MPEG-2/H.264 Transcoding," IEEE International Conference on Image Processing, pp.869-872, 2006.  
 [7] Hari Kalva, "Issues in H.264/MPEG-2 Video Transcoding," CCNC, pp.657 - 659, 2004  
 [8] J. Chu et. al., "H.264/MPEG-2 Transcoding based on Personal Video Recorder Platform," ISCE 2005, pp.438-440, 2005.  
 [9] P. Kunzelmann, H. Kalva, "Reduced Complexity H.264 to MPEG-2 Transcoder," ICCE 2007, pp.1-2, 2007.  
 [10] 손남례, 이귀상, "움직임벡터의 거리와 방향성을 고려한 H.264 에러 은닉 방법," 한국통신학회논문지, 34권, 1호, pp.37-47, 2009.  
 [11] 손남례, 이귀상, "후보벡터 분류에 의한 영상 에러 복원," 정보처리학회논문지, 10권, 2호, pp.163-168, 2009.  
 [12] Gary J. Sullivan, "The H.264/MPEG4 Advanced Video Coding Standard and its Applications," IEEE Communications Magazine, 2006.



### 김 기 홍

e-mail : calri99@nate.com

2006년 2월 목포대학교 전자공학과 학사  
 2008년 3월~현 재 전남대학교 전산학과 석사과정  
 관심 분야: 영상처리, 동영상 압축



### 손 남 례

e-mail : nrson72@gmail.com

1996년 2월 호남대학교 컴퓨터공학과(이학사)  
 2000년 2월 전남대학교 전산학과(이학석사)  
 2005년 2월 전남대학교 전산학과(이학박사)  
 2007년 3월~2009년 2월 호남대학교 인터넷 소프트웨어학과 전임강사  
 2009년 11월~현 재 목포대학교 중점연구소 연구교수  
 관심분야: 비디오 코딩, IPTV, e-learning, 방송통신융합기술, 임베디드시스템



### 이 귀 상

e-mail : gslee@jnu.ac.kr

1980년 서울대학교 전기공학과(학사)  
 1982년 서울대학교 전기계산기공학과(석사)  
 1991년 Pennsylvania 주립대학교 전산학(박사)  
 1984년~현 재 전남대학교 전자컴퓨터공학부 교수  
 관심분야: 멀티미디어 통신, 영상 처리 및 컴퓨터비전