

특집논문-05-10-3-09

확률 기반의 블록 모드 결정 기법을 이용한 H.264에서의 고속 비트율 감축 트랜스코딩

김대연^{a)}, 이영렬^{a)*}

Fast Bitrate Reduction Transcoding using Probability-Based Block Mode Determination in H.264

Dae-Yeon Kim^{a)} and Yung-Lyul Lee^{a)*}

요 약

본 논문에서는 H.264로 부호화 된 비트스트림을 이전보다 낮은 비트율을 갖는 비트스트림으로 변환시키기 위한 고속 비트율 감축 트랜스코딩을 제안한다. 제안된 트랜스코딩 기법은, 복호기에서 나온 블록 모드 정보와 움직임 벡터를 이용하여, 확률 기반의 블록 모드 결정 기법과 움직임 벡터 재사용 및 정제 기법을 사용하였다. 그 결과, 직렬 화소 영역 트랜스코딩과 비교하였을 때 약 0.1 ~ 0.3 dB 정도의 화질 저하가 있지만, 부호화 시간을 비교하였을 때 약 40배 빠른 결과를 얻었다.

Abstract

In this paper, we propose a fast bitrate reduction transcoding method to convert a bitstream coded by H.264 into a lower bitrate H.264 bitstream. Block mode informations and motion vectors generated by H.264 decoder are used for probability-based block mode determination in the proposed transcoding method. And the motion vector reuse and motion vector refinement process are applied in the proposed transcoding. In the experiment results, the proposed methods achieves approximately 40 times improvement in computation complexity compared with the cascaded pixel domain transcoding, while the PSNR(Peak Signal to Noise Ratio) is degraded with only 0.1~ 0.3 dB.

Keywords : H.264, Transcoding, Block Mode Determination, Rate-Control

I. 서 론

최근 컴퓨터 및 통신 기술의 발달로 인하여 디지털 방송, 화상회의, 주문형 비디오(VOD), 원거리 학습 등 멀티미디어 서비스들이 보편화되고 있다. 그러나 디지털 방송과 같

은 고화질 멀티미디어 콘텐츠의 경우 약 4~20Mbps 정도의 넓은 네트워크 대역폭 및 실시간 전송을 요구하고 있어, 제한된 대역폭을 갖는 유무선 통신 시스템에 적용하기에 어려움이 있다. 이러한 문제점을 해결하기 위하여, 고화질 비디오와 같은 멀티미디어 데이터의 고비트율 비트스트림을 네트워크 환경 및 사용자 단말기의 성능에 따라서 최소한의 화질저하를 보장하면서 적응적으로 비트스트림의 대역폭을 조절하는 트랜스코딩 기술^{[1][2]}에 대한 연구들이 수행

a) 세종대학교 컴퓨터공학부 인터넷공학과
School of Computer Engineering, Dept. of Internet Engineering, DMS Lab., Sejong University

되어왔다. 트랜스코딩의 방법으로 여러 가지 고속의 구조들이 제안되었지만, 대부분의 트랜스코딩 구조들은 복호화와 부호화 과정을 거치면서 참조 영상의 차이로 인한 드리프트(drift) 오류가 발생하여 화질의 저하를 발생시킨다. 드리프트 오류를 억제하기 위해 부호화되어 있는 비트스트림을 복호화한 후 사용자에게 적합한 비트율로 재부호화하는 직렬 화소 영역 트랜스코더(Cascade Pixel-Domain Transcoder : CPDT) 구조^[2]와 직렬 DCT 영역 트랜스코더(Cascade DCT-Domain Transcoder : CDDT) 구조^[2]가 널리 쓰이고 있다. 하지만, H.264^{[3][5]} 기반의 DCT 영역 트랜스코딩은 루프 필터(Loop filter)로 인해 드리프트 오류가 생기게 된다. 따라서 H.264 기반의 트랜스코딩은 화소 영역에서의 트랜스코딩 기법을 사용하고 있다. 화소 영역에서의 트랜스코딩 기법 중 가장 기본적인 구조를 갖는 CPDT는 복호화 후에 재구성된 영상 데이터를 가지고 다시 부호화할 수 있기 때문에 가장 고화질의 영상 데이터를 제공할 수 있다. 하지만, 기본적인 복호화 과정과 부호화 과정을 모두 포함하고 있으므로 복잡도가 높고 계산량이 많으며 지연이 생기는 단점이 있다. 이런 단점을 보완하기 위하여 블록 모드 정보와 움직임 벡터를 재사용하는 수정된

CPDT(Modified CPDT : M-CPDT)^[6]가 제안되었다. 이 M-CPDT는 복호기에서 추출한 블록 모드 정보와 움직임 벡터를 부호기에서 그대로 사용하여 복잡도와 계산량을 상당히 줄일 수 있지만, 목표 비트율(Target Bitrate)을 맞추기 위해 높은 QP(Quantization Parameter)를 사용하여 화질 열화가 상당히 심하다. 따라서 본 논문에서는 M-CPDT의 화질 열화를 막기 위하여 확률 기반의 블록 모드 결정 방법을 이용한 H.264 간의 고속 비트율 감축 트랜스코딩을 제안한다.

본 논문은 다음과 같이 구성된다. II장에서는 CPDT와 M-CPDT에 대해 소개하며, III장에서는 확률 기반의 블록 모드 결정 방법을 제안한다. IV장에서는 제안된 방식의 실험결과에 대하여 설명하고, 끝으로 V장에서 결론을 맺는다.

II. CPDT

그림 1은 CPDT의 블록 다이어그램이다. 그림 1에서 R1의 비트율을 갖는 비트스트림을 역엔트로피(Entropy

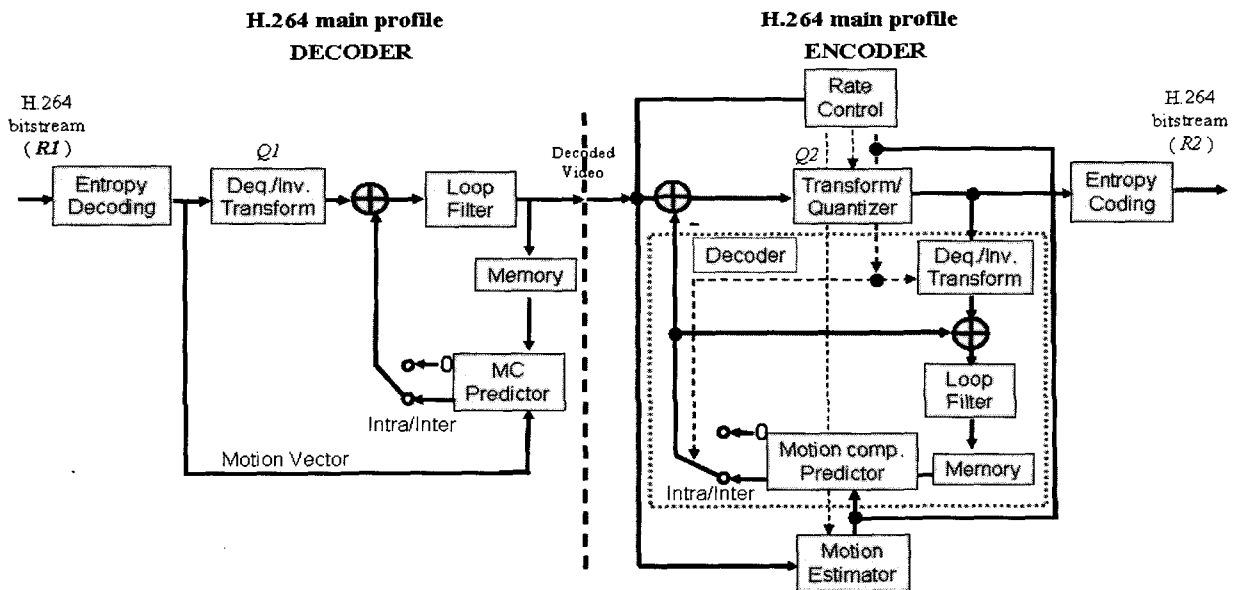


그림 1. CPDT의 블록 다이어그램
Fig. 1. Block diagram of CPDT(Cascade Pixel-Domain Transcoder)

Decoding) 과정을 통해 복호화한 후 $Q1$ 의 QP 값으로 역 양자화(Dequantization) 과정과 IDCT(Inverse Discrete Cosine Transform) 과정을 거쳐서 재구성된 화면을 얻게 된다. 이것이 부호화 과정으로 넘어가 DCT 과정과 $Q2$ 의 QP 값으로 양자화 과정을 거치고, 엔트로피 과정을 통해 $R2$ 의 비트율을 갖는 비트 스트림으로 만들어진다. 여기서 움직임 보상을 위한 예측 메모리가 부호화 과정과 복호화 과정에 각각 존재하는 분리형 구조로 되어서 디코딩 과정에서 복원된 영상을 부호화에서 참조 화면으로 사용하기 때문에 드리프트 오류가 발생하지 않는다. 이 때, H.264에서 사용하는 모든 블록 모드를 후보로 움직임 탐색 과정과 율-왜곡 최적화 과정(Rate Distortion Optimization)^[7]을 거쳤기 때문에 가장 이상적인 움직임 벡터와 블록 모드가 결정된다. 하지만 계산 복잡도가 가장 높은 움직임 탐색 과정과 블록 모드 결정 과정이 재수행됨으로써 부호화 시간이 상당히 증가하게 된다.

이러한 CPDT의 단점을 보완하기 위하여, 그림 2와 같이 복호기에서 추출한 블록 모드 정보와 움직임 벡터를 재사용하는 M-CPDT가 제안되었다. 이 논문에서의 트랜스코딩은 spatial resolution 변환이나 temporal resolution

변환이 아닌 동종 간의 비트율 감축을 위한 것이기 때문에 M-CPDT는 복호기에서 움직임 벡터와 블록 모드를 추출해 부호기에서 어떠한 변환 없이 그대로 사용할 수 있으며, 이로 인해 움직임 탐색 과정과 율-왜곡 최적화 과정이 생략되어 부호화 시간을 최대한으로 줄일 수 있다. 하지만, 이 방법을 실험한 결과 PSNR이 상당히 떨어지는 결과를 보였다. 그 이유는 율-제어 (Rate Control)^{[8][9]} 과정에서 목표 비트율(Target Bitrate)을 맞추기 위해 높은 QP를 사용하기 때문이다. 이러한 M-CPDT의 화질 열화를 줄이기 위하여 확률 기반의 블록 모드 결정 방법을 이용한 H.264 간의 고속 비트율 감축 트랜스코딩을 제안한다.

III. 확률 기반의 블록 모드 결정 방법

1. H.264의 블록 모드 결정 방법

H.264에서는 기존 영상 부호화 표준과는 달리 율-왜곡 최적화 과정을 사용하여 매크로블록 단위에서 SKIP 모드, Direct 모드, Intra16×16 모드, Intra4×4 모드, Inter16×16

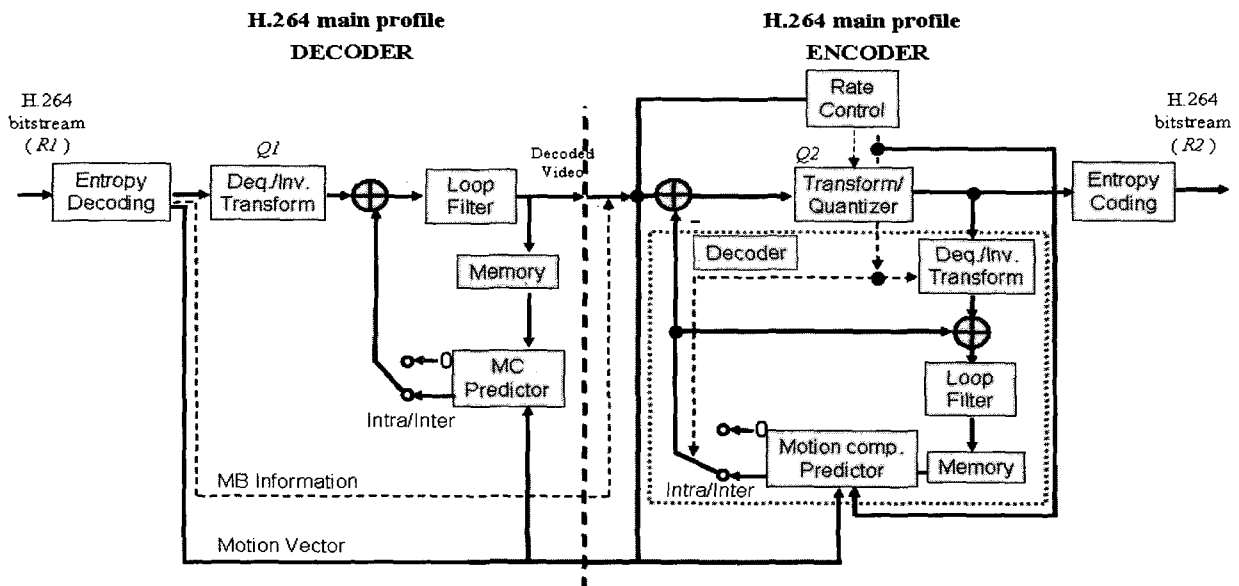


그림 2. M-CPDT의 블록 다이어그램
 Fig 2. Block diagram of M-CPDT

모드, Inter16×8 모드, Inter8×16 모드, P8×8 모드, 8×8 블록 단위에서 Direct 모드, Inter8×8 모드, Inter8×4 모드, Inter4×8 모드, Inter4×4 모드의 다양한 크기의 블록 모드들에 대해 *RDCost*를 계산하여 최소의 *RDCost*를 제공하는 블록 모드를 매크로블록의 부호화 모드로 결정한다. *RDCost*는 Lagrangian Cost Function을 이용하여 식 (1)과 같이 계산되어진다.

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

$$Distortion = \sum_{k=1}^{16} \sum_{l=1}^{16} |B(k, l) - B'(k, l)|^2 \quad (2)$$

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

*Distortion*은 SSD(Sum of Square Difference)를 이용하여, 원본 매크로블록과 비교하여 복원된 매크로블록의 왜곡정도를 나타내고, 식 (2)와 같이 계산되어진다. *B(k,l)*은 원본 매크로블록의 (*k,l*) 번째 픽셀 값을 나타내고 *B'(k,l)*은 복원된 매크로블록의 (*k,l*) 번째 픽셀 값을 나타낸다. λ 는 QP에 의해 결정 되는 상수이고, 식 (3)과 같이 계산된다. *Rates*는 매크로블록을 표현하는데 필요한 비트수를 나타내며 움직임 벡터, 블록 코딩 패턴(CBP), 블록 모드, 잔여 신호(Residual Signal)등의 매크로블록의 모든 정보에 의해서 결정 된다. 그러므로 *RDCost*는 원본 매크로블록과 복원된 매크로블록간의 왜곡정도와 부호화 된 매크로블록을 표현하는데 필요한 비트수에 의해서 결정된다고 할 수 있다. 그러나 모든 블록 모드들에 대해 *RDCost*를 계산해야함으로 각 모드들의 울-왜곡 최적화 계산 시 요구되는 계산량이 방대해 부호화 시간이 상당히 지연되는 단점이 있다.

2. 제안된 확률 기반의 블록 모드 결정 방법

블록 모드 결정은 블록 내의 영상의 특성에 의해 영향을 많이 받기 때문에, 복호기에서 추출한 블록 모드(입력 블록

모드)와 재부호화 시에 결정되는 블록 모드(출력 블록 모드)간에는 상관관계가 매우 높다. 이것은 표 1을 통해서 증명된다. 표 1은 4Mbps의 비트율로 부호화 된 NTSC 방식의 30Hz 영상 8개 (Mobile, Football, Tempete, Bus, Flower, Popple, Susie, Tennis)와 PAL 방식의 25hz 영상 3개 (Canoa Valsesia, F1 car, Rugby)의 비트스트림에 대하여 복호기에서 추출한 블록 모드에 따른 부호기에서 결정된 블록 모드의 확률을 CPDT에서 측정 한 결과이다. 첫 번째 열은 복호기에서 추출한 블록 모드(A)이며, 첫 번째 행은 동일 매크로블록이 재부호화 과정에서 결정된 블록 모드(B)이다. 예를 들어, 복호기에서 Inter8×16 모드가 추출되었을 때 부호기에서 Inter16×16 모드로 결정 될 확률이 35.2% 임을 알 수 있다. 대부분의 경우에 복호기에서 추출한 블록 모드와 부호기에서 결정 된 블록 모드가 같은 경우 가장 높은 확률을 갖는 것을 볼 수 있다. 하지만, 이전 보다 더 큰 블록 모드로 결정되는 경우의 확률도 상당히 높은 편임을 알 수 있다. Inter16×8 모드, Inter8×16 모드, P8×8 블록은 Inter16×16 모드로 결정되는 확률이 높고, Inter8×4 모드, Inter4×8 모드, Inter4×4 모드는 Inter8×8 모드로 결정되는 확률이 높은 것이 그 예이다. 표 1에서 보는 바와 같이 복호기에서 추출된 블록 모드에 대하여 부호기에서 결정되는 블록 모드는 대부분 2~3 가지 모드로 집중되는 현상을 볼 수 있다. 이 같은 현상을 이용하여, 확률이 높은 블록 모드만 울-왜곡 최적화 과정의 후보 블록 모드로 사용하였다. 표 1에서 음영을 넣은 셀은 해당 모드가 부호기에서 선택된 확률이 10% 이상일 때이며, 그 이하의 확률을 갖는 블록 모드는 블록 모드 결정 과정에서 고려하지 않아도 전체적인 성능에 크게 문제를 주지 않을 것으로 판단하여 10%를 문턱값(Threshold)으로 정하여 10% 이상의 확률을 갖는 블록 모드만을 후보 블록으로 사용하였다. 예외적으로 Intra 블록의 중요성을 고려하여 Intra4×4 모드와 Intra16×16 모드 간에는 서로 선택 가능하게 하였다. 그리고 P picture에서 Inter8×8 모드와 Inter4×8 모드가 Inter8×4 모드로 결정되는 경우의 확률이 각각 10.5%와 10%이지만, P8x8에서의 부호화 시간을 줄이기 위해 고려하지 않았다.

표 1. 트랜스코더의 P,B picture에서 입력 블록 모드에 대한 출력 블록 모드의 확률

Table 1. In the P and B pictures of the transcoder, the probability of the output block modes for each input block mode

(a) P picture에서 각 입력 매크로블록 모드(A)에 대한 출력 매크로블록 모드(B)의 확률

(a) In P picture, the probability of determined MB modes(B) for each input MB mode(A)

A \ B	Skip	Inter16x16	Inter16x8	Inter8x16	P8x8	Intra4x4	Intra16x16
Skip	89.0%	8.4%	0.5%	0.4%	0.2%	0.5%	1.0%
Inter16x16	13.7%	67.3%	6.4%	4.7%	4.3%	1.0%	2.6%
Inter16x8	6.5%	28.5%	47.6%	4.0%	9.7%	1.2%	2.5%
Inter8x16	9.2%	35.2%	6.7%	36.3%	9.2%	1.2%	2.2%
P8x8	1.7%	12.0%	11.8%	6.4%	66.1%	1.0%	1.0%
Intra4x4	1.0%	3.0%	2.4%	1.1%	1.4%	71.8%	19.3%
Intra16x16	2.6%	5.1%	3.2%	1.2%	2.4%	18.9%	66.6%

(b) B picture에서 각 입력 매크로블록 모드(A)에 대한 출력 매크로블록 모드(B)의 확률

(b) In B picture, the probability of determined MB modes(B) for each input MB mode(A)

A \ B	Direct	Inter16x16	Inter16x8	Inter8x16	P8x8	Intra4x4	Intra16x16
Direct	86.3%	12.8%	0.3%	0.2%	0.4%	0.0%	0.0%
Inter16x16	13.0%	80.5%	2.6%	1.8%	1.3%	0.4%	0.4%
Inter16x8	7.2%	39.6%	43.7%	2.5%	5.6%	0.7%	0.7%
Inter8x16	8.0%	46.4%	4.1%	35.3%	5.2%	0.5%	0.5%
P8x8	4.1%	20.1%	11.4%	5.8%	57.6%	0.6%	0.4%
Intra4x4	3.4%	13.8%	7.9%	2.6%	6.3%	57.1%	8.9%
Intra16x16	7.2%	25.4%	5.0%	1.4%	1.8%	14.1%	45.1%

(c) P picture에서 각 입력 서브 매크로블록 모드(A)에 대한 출력 서브 매크로블록 모드(B)의 확률

(c) In P picture, the probability of determined sub MB modes(B) for each input sub MB mode(A)

A \ B	Inter8x8	Inter8x4	Inter4x8	Inter4x4
Inter8x8	80.1%	10.5%	6.8%	2.6%
Inter8x4	26.3%	61.8%	5.0%	4.9%
Inter4x8	37.4%	10.0%	47.1%	5.5%
Inter4x4	23.0%	26.1%	14.5%	36.4%

(d) B picture에서 각 입력 서브 매크로블록 모드(A)에 대한 출력 서브 매크로블록 모드(B)의 확률

(d) In B picture, the probability of determined sub MB modes(B) for each input sub MB mode(A)

A \ B	Direct8x8	Inter8x8	Inter8x4	Inter4x8	Inter4x4
Direct8x8	77.3%	19.8%	1.6%	0.8%	0.5%
Inter8x8	4.7%	88.7%	4.0%	1.9%	0.7%
Inter8x4	2.6%	42.6%	50.7%	1.9%	2.2%
Inter4x8	3.3%	42.7%	4.5%	46.2%	3.3%
Inter4x4	2.1%	33.5%	20.0%	12.8%	31.7%

3. 움직임 벡터 재사용 및 정제

위에서 언급한 확률 기반의 블록 모드 결정 기법을 사용할 때, 복호기에서 추출한 블록 모드가 현재 부호화하려는 블록 모드와 같은 경우에는 복호기에서 추출한 움직임 벡터를 직접 사용할 수 있다. 하지만, 부호화하려는 블록 모드가 복호기에서 추출한 블록 모드와 다를 경우 블록의 크기에 맞게 움직임 벡터를 변환시켜 줄 필요가 있다. 표 1에서 보는 바와 같이 확률이 10% 이상일 경우는 복호기에서 추출한 블록 모드와 부호기에서 결정된 블록 모드가 같거나, 부호기에서 결정된 블록 모드의 크기가 더 클 경우이다. 전자의 경우는 움직임 벡터를 그대로 사용하고, 후자의 경우는 그림 3과 같이 입력 블록들의 움직임 벡터의 평균을 구하여 출력 블록의 움직임 벡터로 사용하였다. 그림 3에서 결정되는 움직임 벡터 M은 식 (4)와 같다.

$$M(a,b) = \left\{ \sum_{i=a}^b MV_i + (b-a+1)/2 \right\} \gg (b-a+1)/2 \quad (4)$$

그리고 복호기에서 추출한 블록 모드가 Intra이고 부호기에서 후보 블록 모드가 Inter일 경우, 움직임 벡터가 없기 때문에 이때는 PMV(Predicted Motion Vector)를 사용하였다. 이와 같은 방법으로 구한 움직임 벡터를 탐색 중심(Search Center)으로 정하고 이를 정제^[10]하였다. 그림 4는 30Hz의 비월주사 방식의 720× 480 크기의 영상 3가지(Bus, Flower, Tennis)를 제안된 트랜스코더를 이용하여 탐색 범위를 ±1~±5까지 바꿔주면서 움직임 탐색을 하지 않은 것과의 평균 PSNR의 향상과 평균 부호화 시간의 증가율을 나타낸 것이다. PSNR의 향상은 탐색 범위가 ±1일 때, 약 0.05dB 향상 되지만, 탐색 범위가 증가할수록 부호

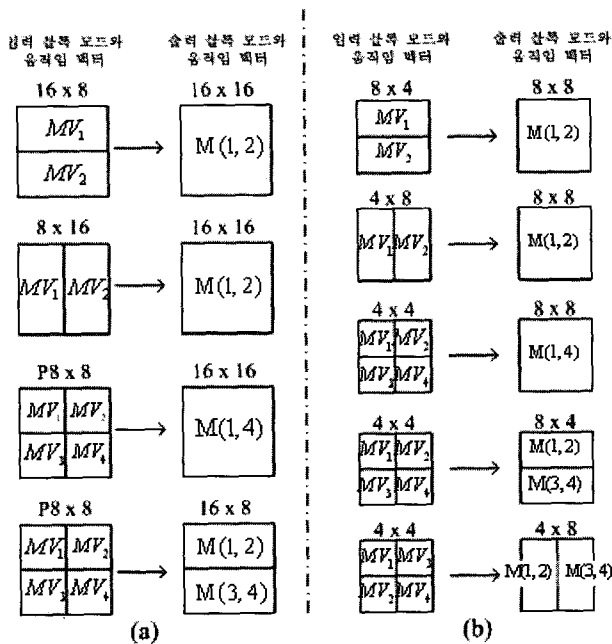


그림 3. 부호기에서 사용할 움직임 벡터의 결정
(a) 매크로블록 단위 (b) P8×8 블록 단위
Fig 3. determination of motion vector to use in encoder
(a) Macroblock unit (b) P8×8 block unit

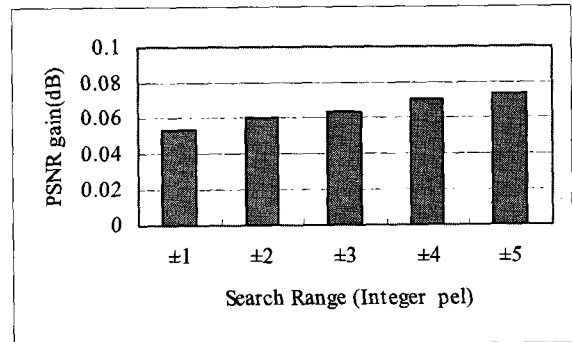


그림 4 (a). 탐색범위를 ±1~±5 변화시켰을 때의 PSNR 향상
Fig 4 (a). PSNR gains when the search ranges are ±1~±5

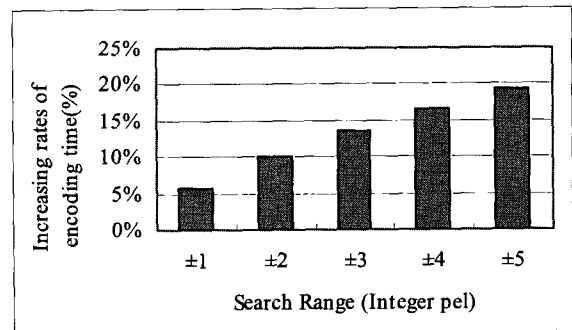


그림 4 (b). 탐색범위를 ±1~±5 변화시켰을 때의 부호화 시간 증가율
Fig 4 (b). Increasing rates of encoding time when the search ranges are ±1~±5

화 시간 증가율에 비해 PSNR 향상은 거의 없는 것을 알 수 있다. 따라서 제안된 트랜스코더에서는 탐색 범위를 ± 1 로 정하여 움직임 벡터 정제를 하였다.

IV. 실험 결과

실험을 위해 H.264 Reference software인 JM(Joint Model) 8.2 버전을 이용한 CPDT, M-CPDT, 제안한 트랜스코더의 성능을 비교 분석하였다. M-CPDT는 제안한 트랜스코더와의 정확한 비교를 위하여 복호기에서 추출한 움직임 벡터를 탐색 중심으로 정하고 탐색 범위를 ± 1 로 주어 움직임 벡터 정제를 하였다. 3개의 트랜스코더는 모두 JM8.2의 율-제어 (Rate-Control)를 이용하여 4Mbps의 비트스트림을 1.5Mbps, 2Mbps, 2.5Mbps의 비트스트림으로 변환하였을 때의 성능을 비교하였다. 실험 조건은 표 2와 같고, 실험 환경은 Pentium-4 3.00 GHz의 PC, 1Byte 메모리이며, 운영체제는 Windows XP 이다.

표 2. 실험 조건

Table 2. Experimental Conditions

Sequence	Bus (720x 480) Interlaced	Flower (720x 480) Interlaced	Tennis (720x 480) Interlaced
Total Frames	148	148	148
Bitrate	1.5 Mbps, 2Mbps, 2.5Mbps		
Coding Options	가변 블록 움직임 추정, 율-왜곡 최적화, 하다마드 변환, 30장 주기의 I프레임 사용, IBBPBBP...구조, CABAC, 에러틀 사용안함, Reference Frame 5장, Rate Control 사용, 1/4 화소 움직임 정확도 지원, CPDT의 움직임 탐색 영역 (± 16) M-CPDT의 움직임 검색 영역 (± 1) 제안된 트랜스코딩의 검색영역 (± 1)		

실험 결과, 제안된 트랜스코더는 낮은 비트율의 환경일 수록 M-CPDT에 비하여 더욱 좋은 결과를 보였다. 그림 5에서 보는 바와 같이, 1.5Mbps의 비트율에서, CPDT에 비하여 Bus 영상은 0.4dB, Flower 영상은 0.4dB, Tennis 영

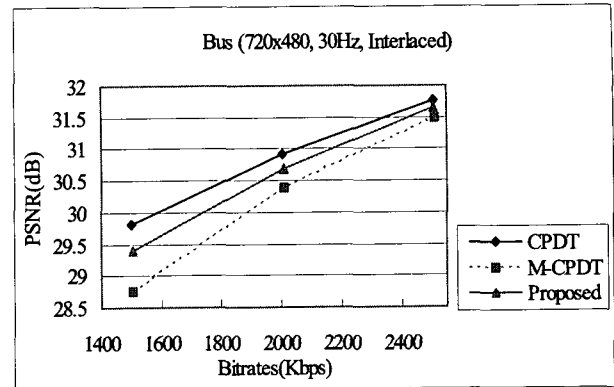


그림 5 (a). Bus 영상의 R-D 곡선
Fig 5 (a). R-D curves of the Bus sequence

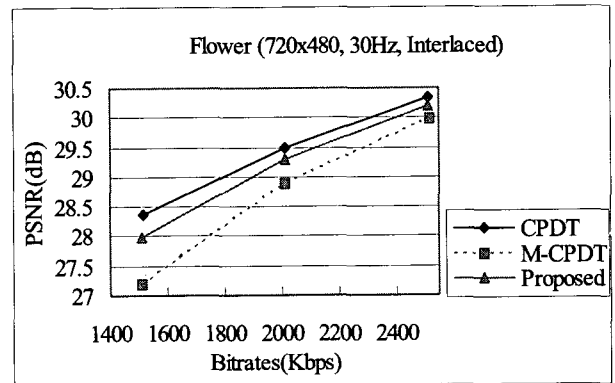


그림 5 (b). Flower 영상의 R-D 곡선
Fig 5 (b). R-D curves of the Flower sequence

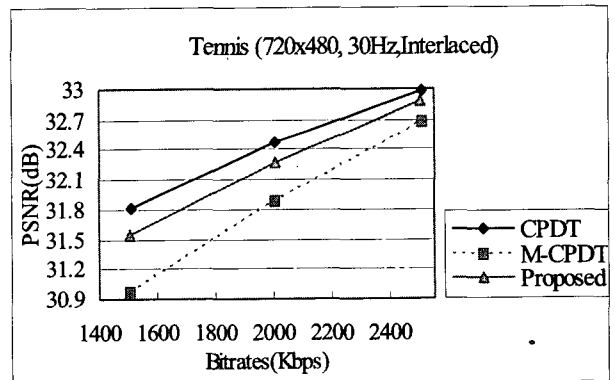


그림 5 (c). Tennis 영상의 영상의 R-D 곡선
Fig 5 (c). R-D curves of the Tennis sequence

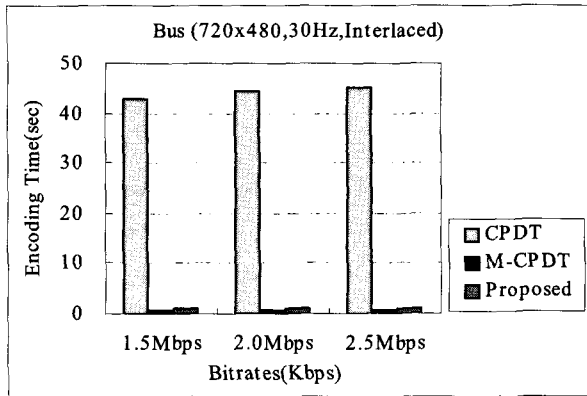


그림 6 (a). Bus 영상의 부호화 시간
Fig 6 (a). Encoding times of the Bus sequence

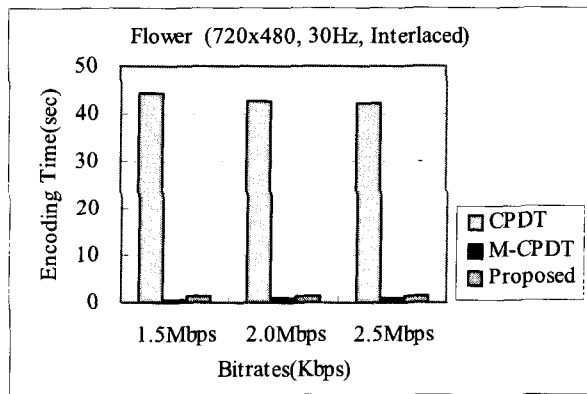


그림 6 (b). Flower 영상의 부호화 시간
Fig 6 (b). Encoding times of the Flower sequence

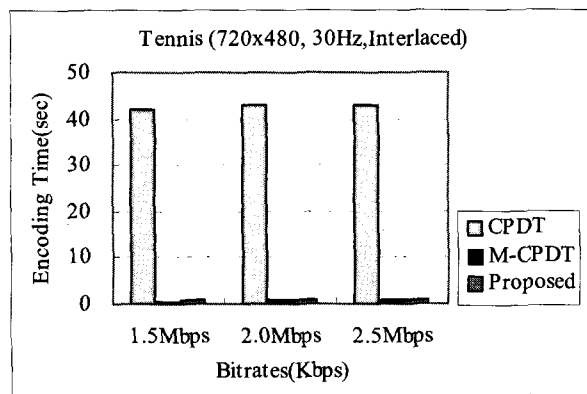


그림 6 (c). Tennis 영상의 부호화 시간
Fig 6 (c). Encoding times of the Tennis sequence

상은 0.3dB 감소를 보였으나, M-CPDT와의 비교에서는 Bus 영상은 0.7dB, Flower 영상은 0.8dB, Tennis 영상은 0.6dB 높은 결과를 보였다. 2Mbps의 비트율에서는, CPDT에 비하여 Bus 영상은 0.3dB, Flower 영상은 0.2dB, Tennis 영상은 0.2dB 감소를 보였으나, M-CPDT와의 비교에서는 Bus 영상은 0.3dB, Flower 영상은 0.4dB, Tennis 영상은 0.4dB 높은 결과를 보였다. 부호화 시간을 비교하였을 때, 그림 6에서 보는 바와 같이 CPDT는 한 장의 영상을 부호화하는데 모든 비트율에서 평균적으로 약 43초의 부호화 시간이 소요되었으나, 제안된 트랜스코더는 약 1.1초가 소요되어 약 40 배의 속도 증가를 보였다. 그리고 M-CPDT와 비교하여 Bus 영상은 약 0.3초, Flower 영상은 약 0.4초, Tennis 영상은 약 0.3초의 미미한 부호화 시간이 증가하였다.

V. 결론

본 논문은 H.264로 부호화 된 비트스트림을 이전보다 낮은 비트율을 갖는 비트스트림으로 변환시키기 위한 고속 비트율 감축 트랜스코딩을 제안하였다. 그 방법으로 확률 기반의 블록 모드 결정 기법과 움직임 벡터 재사용 및 정제 기법을 사용하였다. 그 결과, 낮은 비트율에서도 CPDT와 비교하여 화질 저하가 거의 없었고, 부호화 시간은 상당히 줄어들었다. 따라서 제안된 트랜스코더는 고화질 비디오와 같은 멀티미디어 데이터의 고비트율 비트스트림을 네트워크 환경 및 사용자 단말기의 성능에 따라 최소한의 화질저하를 보장하면서, 실시간을 필요로 하는 곳에서 유용하게 쓰일 수 있을 것이다.

참고 문헌

- [1] A. Vetro, C. Christopoulos, and H. Sun, "Video Transcoding Architectures and Techniques: An Overview", IEEE Signal Processing Magazine, March 2003.
- [2] J. Xin, C. W. Lin, M. T. Sun "Digital Video Transcoding", Proceedings of the IEEE, Vol.93 No.1, January 2005.
- [3] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra,

- "Overview of the H.264/AVC Video Coding Standard", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, July 2003
- [4] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Study of Final Comitee Draft of Joint Video Specication (ITU-T Rec. H.264; ISO/IEC 14496-10 AVC) (JVT F-100), ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Dec. 2002.
- [5] Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Reference Software to Commitee Draft JVT-F100 JM82, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6.
- [6] Zhang, Jijun;Perkis, Andrew;Georganas, Nicolas. "H.264/AVC and Transcoding for Multimedia Adaptation", In Proceedings of the 6th COST 276 WORKSHOP, Thessaloniki, Greece, 6-7 May 2004.
- [7] G.J. Sullivan and T. Wiegand: Rate-Distortion Optimization for Video Compression IEEE Signal Processing Magazine, Vol. 15, November 1998, pp. 74 - 90.
- [8] Z. Li, F. Pan, G. Feng, K. Lim, X. Lin, S. Rahardja, New Rate Control Algorithm, JVT-E069, 5th Meeting: Geneva, Oct. 2002
- [9] Z. He, Y. K. Kim, Sanjit K. Mitra "Low-Delay Rate Control for DCT Video Coding via λ -domain Source Modeling", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 8, pag. 928-940, Aug. 2001.
- [10] K. D. Seo and J. K. Kim, "Motion vector refinement for video downsampling in the DCT domain," IEEE Signal processing letters, vol.9, pp.356-359, November 2002.

저 자 소 개



김 대 연

- 2005년 2월 : 세종대학교 인터넷학과 졸업
- 2005년 3월 ~ 현재 : 세종대학교 컴퓨터공학부 인터넷공학과 석사과정
- 주관심분야 : H.264, 3차원 비디오 코딩, 트랜스코딩



이 영 렬

- 1985년 2월 : 서강대학교 전자공학과 졸업
- 1987년 2월 : 서강대학교 전자공학과 석사
- 1994년 2월 : 삼성전자 중앙연구소 DMS Lab. 선임연구원
- 1999년 2월 : 한국 과학 기술원 전기 및 전자공학과 박사
- 2001년 8월 : 삼성전자 중앙연구소 DMS Lab. 수석연구원
- 2001년 9월 ~ 현재 : 세종대학교 컴퓨터공학부 인터넷공학과 부교수
- 주관심분야 : 영상처리(압축, 복원), 영상전송, 멀티미디어 시스템, 3차원 비디오 코딩, 트랜스코딩, Scalable 비디오 코딩