

실시간 MPEG-2 to MPEG-4 트랜스코더의 설계 및 구현

김 제 우, 김 용 환, 김 태 완, 최 병 호
*전자부품연구원 디지털미디어연구센터
jwkim@keti.re.kr

Design and Implementation of Realtime MPEG-2 to MPEG-4 Transcoder

Je Woo Kim, Yong-Hwan Kim, Tae-Wan Kim, and Beong-Ho Choi
Digital Media Research Center, Korea Electronics Technology Institute(KETI)

요 약

최근 디지털 방송과 이동통신 단말기의 대중화가 이루어짐에 따라 고화질 고해상도의 멀티미디어 콘텐츠의 이동통신 단말기에서의 재생 서비스에 대한 수요가 증가하고 있다. 이동통신 단말기에서 멀티미디어 콘텐츠 재생 서비스를 제공하기 위해서는 디지털 방송 콘텐츠를 단말기에 적합한 콘텐츠로 변환할 필요가 있다. 본 논문은 디지털 방송 규격인 MPEG-2 콘텐츠를 이동통신 단말기에서 지원하는 MPEG-4 SP(Simple Profile) 콘텐츠로 실시간으로 변환하는 트랜스코더에 대한 설계와 구현 기술을 제안한다. 구현된 트랜스코더는 화질 유지와 계산량 감소를 위한 적응적 움직임벡터 재추정, 매크로블록 모드 선택, 그리고 움직임벡터 scaling 등의 알고리즘을 포함하고, 인텔사에서 제공하는 SIMD(Single Instruction Multiple Data) 명령어를 이용하여 최적화되었다. 트랜스코더는 30fps, 8Mbps, 720×480 해상도의 멀티미디어 콘텐츠를 다양한 비트율의 30fps, 352×240 해상도의 MPEG-4 콘텐츠로 실시간 변환할 수 있다.

I. 서 론

디지털 방송 및 DVD의 대중화와 더불어 핸드폰, PDA등과 같은 개인 이동통신 단말기의 성능이 급속도로 발전함에 따라 사용자의 고화질·고해상도의 멀티미디어 콘텐츠에 대한 수요도 급증하고 있다. 디지털 방송 및 DVD와 같은 고화질·고해상도의 멀티미디어 콘텐츠를 이동통신 단말기에서 재생하기 위해서는 이동통신 단말기의 성능과 네트워크의 대역폭 및 특성에 맞게 적응적으로 콘텐츠를 변환할 필요가 있다. 비디오 트랜스코딩 기술은 하나의 포맷으로 인코딩된 비디오 신호를 다른 포맷으로 변환하는 기술로서, 고화질·고해상도의 멀티미디어 콘텐츠가 이동통신 단말기에서 고화질·저해상도의 멀티미디어 서비스로 제공될 수 있도록 한다.

MPEG-2 규격은 디지털 방송 및 DVD에서 채택되어 사용되고, MPEG-4 규격은 이동통신에서 사용될 수 있도록 error resilience 등과 같은 볼들을 제공하고 있다. 그러므로, 디지털 방송을 이동통신 단말기에서 재생하기 위해서는 MPEG-2 to MPEG-4 트랜스코딩 기술이 요구된다[1].

본 논문은 DVD 및 디지털 방송 규격인 MPEG-2[2]로 인코딩된 스트림을 이동통신 단말기 등에서 재생하기 위해서 MPEG-2 SP[3]로 실시간 변환하는 트랜스코더에 대한 설계와 구현 기술을 제안한다. 일반적으로 트랜스코더는 디코더와 인코더를 모두 포함하고 있기 때문에 복잡도가 아주 높다. 그래서, 복잡도를 감소시키기 위해서 기존 움직임벡터를 재사용하는 MVR(motion vector refinement) 기법[4], 고속 움직임벡터 재추정 기법[5] 등이 제안되었고, 또한 DCT 영역에서 동작하는 트랜스코더들도 제안되었다.

또한, MPEG-2와 MPEG-4 규격은 DCT, 양자화, VLC, 움직임 추정 및 보상 등 전체적인 코딩 방법은 비슷하지만, 세부적으로는 다른 기술들을 포함하고 있다. 특히, MPEG-2의 일반적인 프레임 구성은 I, P, B 픽처를 모두 포함하는 IBBPBBP... 또는 IBPBPBP...을 지원하지만, MPEG-4 SP 규격은 I-VOP와 P-VOP만을 지원하기 때문에 IPPPPPP... 또는 IIIII... 등의 프레임 구성을 갖는다. 그래서, MPEG-2에서 추출되는 움직임벡터를 직접적으로 재사용하여 MVR을 수행할 수 없다.

또한 각 프레임 형식에 따라 지원되는 매크로블록 모드들도 다르기 때문에 이에 대한 매핑도 고려되어야 한다.

그래서, 본 논문에서 제안된 트랜스코더는 실시간 트랜스코딩을 구현하기 위해서 계산량 감소 및 화질 유지를 동시에 만족하는 적응적 움직임벡터 재추정 기법, 매크로블록 모드 선택 기법, 그리고 움직임벡터 scaling 등의 알고리즘들을 포함한다. 또한 멀티미디어 콘텐츠를 처리하기 위해서 제공되는 MMX, SSE, 그리고 SSE2 등과 같은 SIMD 명령어를 사용하여 소프트웨어 최적화를 수행하였다.

본 논문의 구성은 본문에서 먼저 제안된 MPEG-2 to MPEG-4 트랜스코더의 구조와 기능에 대해서 기술하고, 그리고 트랜스코더에 포함된 움직임벡터 scaling, 매크로블록 모드 선택, 그리고 적응적 움직임벡터 재추정 기법 등을 기술한다. 그리고 트랜스코더의 최적 구현에 대해서 기술하고, 구현된 트랜스코더의 성능 결과를 기술한다. 그리고 결론을 맺는다.

II. 본 론

1. 제안된 MPEG-2 to MPEG-4 트랜스코더

본 연구에서 MPEG-2 비트스트림을 MPEG-4 비트스트림으로 변환하는 트랜스코더는 그림 1과 같이 공간(픽셀) 영역에서 동작한다. 그러므로, MPEG-2 디코더와 MPEG-4 인코더를 포함하고, 또한 해상도 변환을 수행하기 위한 공간영역 DS(down sampling) 필터를 포함한다.

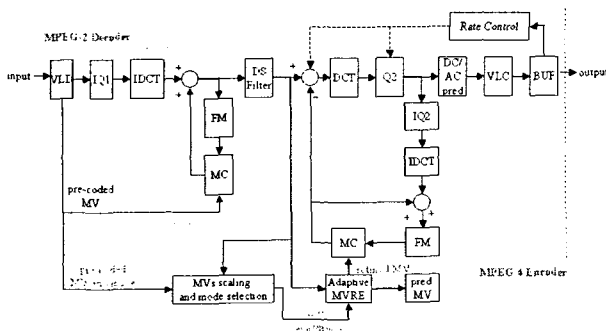


그림 1. MPEG-2 to MPEG-4 트랜스코더의 블록도

제안된 트랜스코더는 계산량 감소 및 화질 유지를 위해서 다음의 블록들을 포함한다. MPEG-2와 MPEG-4의 각 매크로블록은 지원되는 모드가 다르고 그 용법이 틀리기 때문에 기존의 MPEG-2 매크로블록 모드를 효율적으로 MPEG-4 매크로블록 모드로 매핑하는 mode decision 블록이 필요하다. 또한 MPEG-2 규격과 MPEG-4 규격은 지원하는 프레임 구조, 프레임 형식 등의 차이에 의해 기존 움직임벡터를 그대로 재사용할 수 없

으므로, 이에 대한 보정을 수행하는 움직임벡터 scaling 블록이 필요하다. 그리고 보정된 움직임벡터를 그대로 사용하는 경우 영상의 종류에 따라서 화질의 열화가 심해지므로 이러한 화질의 열화를 막기 위해서 최적의 움직임벡터를 찾는 적응적인 움직임벡터 재추정 블록이 요구된다.

트랜스코더에서 DCT, IDCT, MVRE(motion vector re-estimation), MC(motion compensation) 등의 블록들은 루프를 통한 반복 계산이 요구되므로 SIMD 명령어를 사용하여 최적화한다.

2. 움직임벡터 Scaling

MPEG-2와 MPEG-4 SP는 지원되는 코딩 형식의 차이로 인하여 현재 코딩되는 프레임에 대한 참조 영상의 시간적인 불일치가 존재한다. 그래서 MPEG-2의 움직임벡터 및 매크로블록 정보를 MPEG-4 SP로의 트랜스코딩에 직접 사용하기 어렵다. 그러므로 참조영상에 대한 시간적 불일치를 보정하기 위해서 움직임벡터를 scaling 하는 것이 필요하다. scaling 방법은 MPEG-2의 M 파라미터 값과 선형 예측을 이용하여 수행된다. P 픽처를 P-VOP로 코딩하는 경우에는 P 픽처의 참조영상은 이전의 I 또는 P 픽처인 반면 P-VOP는 바로 이전 프레임이 되므로, M 만큼의 참조 영상에 대한 시간적인 간격이 존재하게 된다. B 픽처를 P-VOP로 코딩하는 경우는 B 픽처가 이전 P(또는 I) 픽처와 이후 P(또는 I) 픽처를 참조영상으로 사용하므로, M 값을 현재 B 픽처의 위치를 감소한 만큼의 시간적인 간격이 존재하게 된다. 표 1에 M이 3일 때의 움직임벡터의 scaling을 수행하는 방법을 보이고 있다. MVc는 scaling된 움직임벡터를 의미하고 이 움직임벡터는 움직임벡터 재추정에서 사용하는 움직임벡터 후보군에 해당한다. 그리고 MVf와 MVb는 MPEG-2의 순방향, 역방향 MV를 의미한다.

표 1. 움직임벡터의 scaling 방법(M=3)

picture	scaling process
P → P	$MV_c = (1/M)MV_f$
B1 → P	$MV_c = MV_f$ $MV_c = -(1/(M-1))MV_b$
B2 → P	$MV_c = (1/(M-1))MV_f$ $MV_c = -MV_b$

scaling된 움직임벡터들은 트랜스코더의 움직임벡터 재추정 블록에서 움직임벡터 후보로서 탐색 영역의 중심점이 된다. 움직임벡터 후보가 다수인 경우 median filter를 사용하여 하나의 후보를 추출한다. 만약 해상도 변환을 수행하는 경우 움직임벡터 scaling에서도 MVc에 1/2을 곱한다. 매크로블록 모드가 필드인 경우에는

motion_vertical_field_select 값에 따라 1/2을 더한다.

3. 매크로블록 모드 선택 기법

최적 매크로블록 모드를 찾기 위해서는 움직임벡터 추정과 연동하여 각 매크로블록 모드에 대한 계산이 수행되어야 한다. 그래서 미리 정의된 기존의 MPEG-2 매크로블록 모드를 효율적으로 MPEG-4 매크로블록 모드로 매핑하는 것은 계산량을 감소시키는 방법 중의 하나이다. MPEG-2의 매크로블록이 intra인 경우는 MPEG-4 SP의 intra로 매핑을 하고, MPEG-2의 나머지 모든 모드들은 일단 MPEG-4 SP의 inter로 매핑을 한다. 단지, P 픽처의 no_mc의 경우에는 M값의 차이에 상관없이 움직임벡터가 0임을 의미하므로, 계산량의 감소를 위해서 똑같이 mc_skipped으로 처리한다. 그리고나서, 움직임벡터 재추정 후 매크로블록의 activity를 이용하여 모드를 재검정한다.

down sampling을 이용한 공간 해상도 변환 트랜스코딩을 수행하는 경우, MPEG-2의 매크로블록 4개는 MPEG-4의 매크로블록 1개에 해당하므로, MPEG-2 각각의 매크로블록 모드에 따라서 MPEG-4의 매크로블록 모드를 선택할 필요가 있다. MPEG-2의 매크로블록 모드가 intra인 경우가 3 개 이상인 경우, intra로 선택을 하고, intra가 2개인 경우에는 움직임 재추정 블록에서 매크로블록의 모드를 선택하도록 하고, 나머지는 inter로 설정한다.

선택된 매크로블록 모드 중에서 inter인 경우에는 움직임벡터 재추정 블록에서 inter와 inter4v로 재검정된다.

4. 적응적 움직임벡터 재추정 기법

움직임벡터 scaling을 이용하여 획득된 움직임벡터 후보는 선형 예측된 것이므로, 참조 영상의 시간적 불일치를 완전히 보상할 수 없다. 이를 보상하기 위해서 움직임벡터 재추정을 수행한다. 움직임벡터 재추정은 일반적인 움직임 예측과 달리 적은 탐색 영역을 갖고 최적의 움직임벡터를 찾는 것이다. 하지만, 제안된 트랜스코더는 참조영상의 시간적 불일치를 보상하기 위해서 움직임벡터 재추정 기법을 움직임벡터 후보의 크기에 따라서 탐색 방법을 달리 수행하는 것이다. 즉, 움직임벡터 후보의 크기가 작으면 full search의 방법을 사용하고, 움직임벡터 후보의 크기가 임계치보다 크면 diamond search 방법을 사용한다. 이 기법은 최적의 움직임벡터를 찾기 위해서 탐색 영역을 늘릴 수 있을 뿐만 아니라, 계산량도 일정하게 유지할 수 있는 장점을 갖기 때문에 고화질을 유지하면서 실시간 트랜스코딩을 가능하게 한다.

다. 그림 3에 움직임벡터 재추정 블록의 흐름도를 보이고 있다. 먼저 16x16 매크로블록에 대한 움직임벡터를 재추정 기법에 의해 찾은 후 네 개의 8x8 블록 각각에 대한 움직임벡터 후보를 중심으로 움직임벡터 재추정한다. 두 개의 결과를 비교하여 더 좋은 결과를 움직임벡터로 선정하고, 이 결과를 매크로블록의 activity와 비교하여 임계치보다 작으면 매크로블록 모드를 intra로 결정한다. 해상도 변환을 하는 경우에는 움직임벡터 scaling 블록에서 획득된 움직임벡터 후보군은 8x8 블록에 해당하는 것들이므로, 16x16 매크로블록의 움직임벡터 후보를 네 개의 움직임벡터 후보군에서 median filter를 사용하여 획득한다.

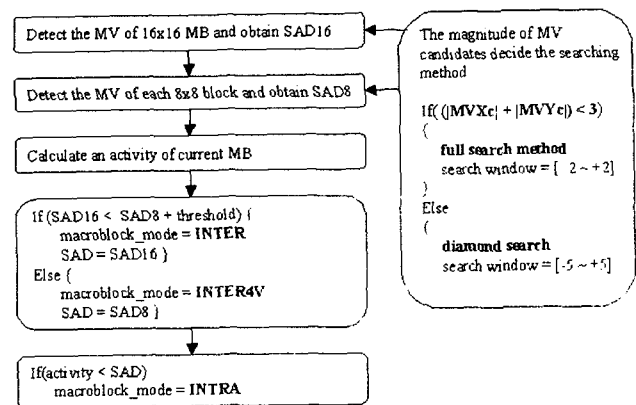


그림 2. 적응적 움직임벡터 재추정 기법의 블록도

5. 제안된 트랜스코더의 최적 구현

일반적으로 레퍼런스 소프트웨어를 사용하여 직렬 공간 영역 트랜스코더(CPDT)를 구현하는 경우, 최적화를 수행하지 않은 경우 DCT, IDCT, MC, ME 등의 계산량은 전체 계산량의 약 80% 이상 차지한다. 이러한 블록들은 일정한 양의 루프를 갖고 일정한 패턴의 산술 계산을 수행한다. 이를 최적화하기 위해서 사용하는 것이 SIMD 명령어들이다. 이 명령어들은 한 명령에 동시에 여러 데이터를 처리하는 것으로, MMX, SSE, SSE2 등이 있다. DCT와 IDCT 블록은 [6]을 기반으로 psllw, pmulhw, pmaddwd 등을 사용하여 최적화를 수행하였다. MC 블록의 경우 16x16 또는 8x8 블록에 대해서 half-pixel interpolation을 해야 하므로, psubusb, pavg 명령어를 사용하였다. MVRE 블록은 SAD를 이용하므로 psadbw, paddusw 명령어를 사용한다. 이들 블록들은 모두 loop unrolling 기법을 이용하여 루프로 인한 지연을 최소화하였다. 또한 최대 16바이트를 동시에 수행함으로써 계산속도를 최적화하였다.

양자화 및 역양자화는 시퀀셜하게 동작하기 때문에 SIMD 명령어의 사용은 적합하지 않다. 그래서 사용가능

한 모든 양자화 계수에 의한 값을 loop-up table로 메모리에 저장한 후 벡터 연산을 수행함으로써 최적화하였다.

6. 구현된 트랜스코더의 성능

제안된 트랜스코더는 인텔 제온 2GHz CPU를 갖는 PC에서 Windows2000 환경하에 MS VC++6.0 컴파일러를 이용하여 구현되었다. 구현된 트랜스코더의 실험 영상으로 축구, 농구, CF 등과 같이 움직임이 많은 영상과 드라마, 뉴스 등과 같이 움직임이 적은 영상을 사용하였다. 제안된 트랜스코더는 CPDT, 기존 MVR을 포함하는 트랜스코더 등과 화질 비교 또는 실시간 처리 성능 비교를 하였다.

표 2와 표 3은 실시간 성능에 대한 제안된 트랜스코더의 성능을 보인 것이다. 표 2는 352x240 영상에 대한 트랜스코딩 결과이고, 표 3은 720x480 영상을 352x240 영상으로 트랜스코딩한 결과이다. 최적화에 관계없이 CPDT는 실시간 수행이 어려운 반면에 제안된 트랜스코더는 최적화후 실시간 트랜스코딩이 가능함을 알 수 있다.

표 4와 표 5는 제안된 트랜스코더의 성능을 PSNR로 비교한 것들이다. 움직임이 많은 축구와 같은 영상에서 제안된 기법들은 보다 좋은 성능을 보이고 움직임 예측 범위에 따라서 CPDT보다 나은 결과를 얻기도 한다. 그리고 움직임이 적은 드라마, 뉴스와 같은 영상에서도 제안된 기법들이 좋은 결과를 얻음을 알 수 있다.

표 2. 352x240 영상의 트랜스코딩 동작시간 결과

video type	CPDT (최적화전)	CPDT (최적화후)	제안된 트랜스코더 (최적화 전)	제안된 트랜스코더 (최적화 후)
News	1.55fps	6.55fps	26.08fps	70.85fps
Soccer	1.57fps	6.61fps	25.39fps	66.91fps

표 3. 해상도 변환을 포함한 트랜스코딩 동작시간 결과

video type	CPDT (최적화전)	CPDT (최적화후)	제안된 트랜스코더 (최적화 전)	제안된 트랜스코더 (최적화 후)
Drama	0.23fps	1.12fps	8.10fps	34.67fps
Basketball	0.24fps	1.15fps	9.01fps	35.59fps

III. 결 론

본 논문은 실시간으로 MPEG-2 콘텐츠를 MPEG-4 SP 콘텐츠로 트랜스코딩하는 트랜스코더를 제안, 구현하였다. 트랜스코더는 계산량 감소 및 화질 유지를 위하

여 적응적 움직임벡터 재추정 기법, 매크로블록 모드 선택 기법, 움직임벡터 scaling 기법 등을 포함하고, SIMD 명령어 및 loop unrolling 등의 방법으로 최적화되었다. 트랜스코더는 30fps, 8Mbps, 720x480 해상도의 MPEG-2 콘텐츠를 다양한 비트율의 30fps, 352x240 해상도의 MPEG-4 콘텐츠로 실시간 변환할 수 있다.

표 4. 트랜스코더의 PSNR 성능 결과(움직임 많은 영상)

transcoded bitrate	video type	CPDT	conventional MVR	proposed transcoder
4M → 384K	soccer	23.01	20.29	22.81
	basket	18.88	18.47	18.94
	CF	34.06	32.31	33.60
4M → 512K	soccer	26.53	23.32	26.61
	basket	20.91	20.34	20.89
	CF	37.36	35.19	36.58

표 5. 트랜스코더의 PSNR 성능 결과(움직임 적은 영상)

transcoded bitrate	video type	CPDT	conventional MVR	proposed transcoder
4M → 384K	news	34.43	33.33	34.03
	drama	41.04	40.09	40.60
4M → 512K	news	37.28	36.33	37.09
	drama	44.68	43.78	44.41

참고문헌

- [1] W. X. Guo, Z. W. Guo, and Ahmad I, "MPEG-2 To MPEG-4 Transcoding," Proceedings of Workshop and Exhibition on MPEG-4/2001, pp.83-86, 2002.
- [2] Information technology - Generic Coding of Moving Pictures and Associated Audio : Video," ISO/IEC 13818-2, 1996.
- [3] Information technology - Coding of audio-visual object : Visual," ISO/IEC 14496-2, 1999.
- [4] J. Youn, M. Sun, and C. Lin, "Motion Vector Refinement for High-Performance Transcoding," *IEEE Trans. Multimedia*, vol.1, no.1, pp.30-40, March 1999.
- [5] J. Xin, M. T. Sun, and B. S. Choi, "Motion re-estimation for HDTV to SDTV transcoding," in Proc. IEEE Symp. Circuits and Systems, Scottsdale, AZ, vol. 4, May 2002, pp. 715 - 718
- [6] A Fast Precise Implementation of 8x8 Discrete Cosine Transform Using the Streaming SIMD Extensions and MMIX™ Instructions, Intel Application Note, AP-922, Copyright 1999.