H.264 비디오 코덱의 실시간 처리를 위한 고속 움직임 추정 알고리즘

유영일, 신기봉, 이승준*, 강동욱, 김기두
국민대학교 전자정보통신공학부
*엠큐브웍스㈜

A Fast Motion Estimation Algorithm for Real-Time Processing of H.264 Video Codec Standard

Young-Il Yoo, Ki-Bong Shin, Seung-Jun Lee*, Dong Wook Kang, Ki-Doo Kim
School of Electronics Engineering, Kookmin Univ.
*McubeWorks Corp.
E-mail : young0125@kookmin.ac.kr

Abstract

본 논문은 가장 최근의 동영상 국제표준인 H.264 비디오 코덱을 사용하여 QCIF 영상을 초당 10 프레임 정도의 속도로 실시간 부호화 하는 것을 목적으로, 부호화시 필요한 연산의 약 80%~90%를 차지하는 움직임 추정을 고속으로 처리할 수 있는 알고리즘을 개발하는 것을 내용으로 하고 있다.

제안하는 고속 움직임 추정 알고리즘은 MPEG-4 등의 고속 움직임 추정에 사용되었던 기존의 알고리즘을 다중 프레임 레퍼런스 등 새로운 특징을 갖는 H.264 코덱에 적합한 형태로 개선하고, 움직임 추정의 정밀도가 1/4 화소 단위로 향상됨으로써 늘어난 부화소단위 움직임 추정의 상대적 부담을 함께 고려하면서, 모드 선택 과정과 효과적으로 결합함으로써 보다 향상된 성능을 나타내고 있다.

모의실험 결과, 기존의 공식 JVT-AVC 레퍼런스 소프트웨어인 JM (Joint Model) 에 구현되어 있는 고속 움직임 추정 알고리즘에 비해서 최대 80%, 평균적으로 60%의 속도개선 효과가 있음이 입증되어, 최근 JM 의 새로운 고속 움직임 추정 알고리즘으로 채택된 JVT-F017 알고리즘에 본 논문에서 제안하는 레퍼런스 프레임 탐색 제한 알고리즘을 결합시킴으로써 추가적으로 약 45%의 속도 개선을 얻을 수 있음을 확인하였다.

I. 서론

H.261, MPEG-1, MPEG-2, H.263, MPEG-4, H.264 와 같은 비디오 압축 표준은 하이브리드 블록 코딩에 기반하고 있다. 이러한 코딩 방식에서 움직임 추정은 비디오 시퀀스의 시간적 중복성을 제거 하는데 매우 효과적인 방법이며, 압축효율에 가장 큰 영향을 주는 부분이다. 반면 가장 많은 시간을 소비하는 부분이기도 하다. 특히 H.264 비디오 코덱 표준에서는 향상된 예측 정확도와 높은 압축 효율을 위하여, 다중 예측모드, 다중 프레임 레퍼런스, 1/4 화소 움직임 추정 정밀도가 채택 되었는데, 그 결과 움직임 추정에 복잡도와 계산량은 매우 크게 증가 하게 되었다.[1] 비디오 코덱의 복잡도와 계산량의 증가는 코덱의 실시간 처리 능력을 제한 하게 되므로, 이동 전화 등에서 실시간 동영상 서비스를 제공하기 위해서는 비디오 코덱의 복잡도 개선이 매우 중요하다. 이와 같은 이유에서 움직임 추정 알고리즘은 산업계와 학술계에서 많은 연구가 이루어 지고 있다.

블록 매칭 움직임 추정 방식은 대부분의 경우 이전 프레임의 미리 정해진 탐색 범위 또는 적응적인 탐색 범위 안에서 현재 프레임의 16x16 매크로 블록(macro-block) 과 가장 정합이 잘되는 위치를 찾는 것을 기본으로 하고 있다. 가장 정합이 잘 되는 위치를 결정 하기 위해 사용되는 평가 기준은 대부분의 경우 SAD(Sum of Absolute Differences) 를 사용하고 있다.

$$SAD(dx,dy)=\sum_{m=x}^{x+N-1}\sum_{n=y}^{y+N-1}\left|I_k(m,n)-I_{k-1}(m+dx,n+dy)\right| \quad (1)$$

식 (1) 로부터 SAD 계산시 SUB, ADD, ABS 의 연산이 필요함을 알 수 있다. 그러므로 SAD 를 사용하는 FSBMA(full-search block-matching algorithm) 의 계산량은 다음과 같다.[2]

$$O_p = 3\cdot 2p\cdot 2p\cdot N_h\cdot N_v\cdot f\cdot nm\cdot nr \quad (2)$$

p :탐색범위
$N_h\cdot N_v$:이미지크기
f :프레임률(*frame*/sec)
nm :예측 모드 갯수
nr :레퍼런스 프레임 갯수

H.264 에서 모든 예측모드(nm = 7) 와 모든 레퍼런스 프레임(nr = 5)을 사용하고, 탐색 범위를 16(p = 16), 초당 프레임률을 30(f = 30)으로 하여, QCIF 영상(N_h = 176, N_v = 144) 을 코딩할 경우 초당 8.17 x 10^{10} 번의 정수 연산을 수행하여야 한다. 실시간 동영상 서비스를 제공하기 위해 초당 8.17x10^{10} 번의 정수 연산은 너무 큰 부담이며 이를 개선 하기 위한 많은 고속 알고리즘들이 개발 되어있다. 고속 움직임 추정 알고리즘은 표 1 과 같이 분류 할 수 있다.[2]

표 1. 고속 움직임 추정 알고리즘

Preprocessing ✓ Feature extraction (e.g. morphological filters) ✓ Frequency-domain transform ✓ Multi-resolution image generation (e.g. by filters, pel sub-sampling)	Flexibility ✓ Content adaptive parameter selection ✓ Algorithm selection ✓ Summation truncation ✓ Criteria to stop the search
Temporal/spatial prediction of the MV/center of search area	
Hierarchical parameter & algorithm modification	
Search area sub-sampling	
Pel decimation	
Matching criterion: block/feature-matching	✓ R/D-optimization
Save one or multiple candidate MV	✓ C/D-optimization

지금 까지 나와있는 H.264 용 고속 움직임 추정 알고리즘들은 위의 기본 방법들의 하나 또는 여러 개의 결합으로 구성 되어있다.

현재 JM 소프트웨어에는 움직임 추정의 속도 개선을 위해서 다중 레퍼런스 프레임과 다중 예측 모드 사용시 레퍼런스 프레임의 인덱스와 예측 모드에 따라 움직임 벡터의 탐색 범위를 제한할 수 있는 부호화 옵션을 제공하고 있다. 즉, 탐색 범위 제한 옵션을 사용하면, 레퍼런스 프레임 인덱스가 상위번호(0→4)로 증가하면 움직임 벡터의 탐색 범위는 이에 반비례하여 줄어들게 되며, 예측 모드에 대해서는 16x16 블록 움직임을 추정하는 모드 1 을 제외한 모든 하위모드(2~7)에 대해서는 움직임 벡터의 탐색 범위가 다시 절반으로 줄어들게 된다. 이 옵션을 사용하면 움직임 추정에 필요한 연산 시간은 상당히 줄어들지만, 부호기의 부호율-왜곡 특성은 나빠진다. 특히 몇몇 영상에 대해서는 부호율-왜곡 성능의 커다란 저하를 일으키기도 하기 때문에 이 옵션을 사용하는 것이 그다지 바람직스럽다고 할 수 없다.

최근 표준으로 채택된 Chen 등의 고속 움직임 추정 알고리즘은 화소 단위 움직임 추정은 UMHexagonS (Hybrid Unsymmetrical-cross Multi-Hexagon-grid Search) 알고리즘을 사용하고, 부화소 단위의 움직임 추정은 CBFPS(Center Biased Fractional Pel Search) 알고리즘을 사용한다.[1] 또한 모든 변환 계수가 양자화 후 0 이 되는 블록을 조기에 발견하여 탐색 과정을 중단 시킬 수 있으며, 변환과 양자화 과정 또한 생략할 수 있도록 설계되었다. 이와 같은 방법을 사용함으로써 기존의 H.264 JM 에 이미 구현되어 있는 고속 움직임 추정 속도를 약 60% 정도 개선할 수 있음이 발표되었다. 이 방식은 근본적으로 부호율-왜곡 성능의 저하를 야기하지 않는다. 따라서 이 방식이 JM 에 구현된다면 기존의 고속 움직임 추정 기법은 이 방식으로 완전히 대체될 것으로 예상된다.

본 논문에서는, 최근 표준으로 채택된 Chen 의 고속 움직임 추정 알고리즘의 속도를 더욱 향상시키기 위해서, 다중 레퍼런스 프레임 움직임 추정 과정에서 나타나는 모드 사이에 존재하는 최적 레퍼런스 프레임의 높은 상관관계를 이용하기 위한 기법을 제안하고자 한다. 2 장에서는 모드 사이에 존재하는 최적 레퍼런스 프레임의 상관성을 분석하고, 이에 바탕하여 효율적인 레퍼런스 프레임 탐색 제한 알고리즘을 소개한다. 그리고 3 장에서는 JM6.1e 를 사용한 모의실험 결과를 소개하고 4 장에서 결론을 맺는다.

II. 레퍼런스 프레임 탐색 제한 알고리즘

2.1 모드간 최적 레퍼런스 프레임의 상관성 분석

모드간 최적 레퍼런스 프레임의 상관성을 몇 가지 모드 조합에 대해서 분석하였다. 모드란 H.264 에서 움직임 추정을 위한 blocktype 을 규정하는 것으로서 모드 번호가 증가할수록 움직임 예측을 위한 블록의 크기가 줄어들도록 정해져 있으며, 구체적인 구조는 그림 1 에 보인 바와 같다.

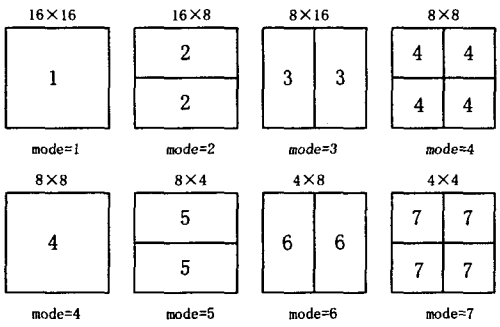

그림 1. H.264 예측 모드

분석을 위해서 5 개의 QCIF 영상 (akiyo, foreman, planet, mononoke, bugs life)과 1 개의 CIF 영상 (mobile)에 대해서 각 모드를 독립적으로 전체 탐색을 행하여 최적 레퍼런스 프레임을 찾았다.

표 2 는 몇 가지 모드 조합에 대해서 최적 레퍼런스 프레임의 일치 확률을 나타내고 있다. 선택된 모드 조합은 직관적으로 그 움직임 벡터들의 레퍼런스 프레임이 서로 일치할 가능성이 높은 조합들이다.

표 2. 모드 조합에 따른 최적 레퍼런스 프레임의 일치 확률

모드 조합	최적 레퍼런스 프레임이 일치하는 확률		
	최대	최소	평균
1,2 3 과 4	98.44%	84.81%	91.04%
4 와 5	95.04%	86.11%	90.19%
4 와 6	97.13%	81.84%	88.36%
4 와 7	96.77%	78.57%	86.41%
1,2,3,4 와 5	97.03%	93.47%	94.93%
1,2,3,4 와 6	98.24%	90.97%	94.19%
1,2,3,4 와 7	98.18%	89.65%	93.22%

표 2 를 통해서 알 수 있는 바와 같이, 상위 모드에서 선택된 최적 레퍼런스 프레임이 하위 모드의 최적 레퍼런스 프레임과 일치할 확률은 영상 종류와 비교 모드에 따라 최고 98%, 최저 82%로 매우 높게 나타난다. 특히 모드 1,2,3,4 와 모드 5, 6, 7 이 일치할 확률은 실험에 사용한 모든 영상 시퀀스에서 90%이상이다.

2.2 모드간 최적 레퍼런스 프레임의 상관성을 이용한 레퍼런스 프레임 탐색 제한 알고리즘

모드간 최적 레퍼런스 프레임이 일치하는 확률이 표 2 에서 보인 바와 같이 매우 높으므로, 하위 모드에서 레퍼런스 프레임을 탐색할 때 상위 모드의 최적 레퍼런스 프레임 중에서만 탐색하도록 함으로써 탐색 시간을 줄일 수 있다. 표 2 에서는 몇 가지 상위-하위 모드 조합에 대하여 최적 레퍼런스 프레임의 일치 확률을 보여주고 있는데, 그 중에 어떤 조합을 이용하는가에 따라 부호화 성능과 속도 사이의 trade-off 가 발생한다. 본 논문에서는 모드 1, 2, 3, 4 에서 선택 되지 않은 레퍼런스 프레임은 모드 5, 6, 7 의 레퍼런스 프레임 후보로 사용 하지 않는 것을 구현해 보았다. 이 조합만을 구현한 이유는 이 조합의 경우에는 레퍼런스 프레임 일치 확률이 영상에 상관없이 90%이상이기 때문에 부호화 성능의 저하가 예상되는 매크로블록의 수가 전체의 10% 미만일 것을 예상할 수 있고, 따라서 전체적인 부호화 성능의 저하가 크지 않을 것이 예상되었기 때문이다. 다른 조합들은 영상에 따라 어느 정도의 부호화 성능 저하를 감수해야만 하기 때문에 매우 높은 속도의 움직임 추정이 필요한 경우에만 이용하는 것이 바람직하다고 판단하였다. 그러나 응용 분야에 따라서는 다른 모드의 조합을 사용 하여 보다 높은 속도 개선 효과를 얻을 수 있을 것이다. 표 2 에서 보인 각각의 모드의 조합에 기반한 레퍼런스 프레임 탐색 제한 알고리즘은 기존의 레퍼런스 프레임 인덱스와 예측 모드에 따른 움직임 벡터 탐색 범위 제한 알고리즘에 비해서 모두 더 우수한 부호율-왜곡-속도 성능을 보인다. 각 경우에 대한 구체적인 성능 분석은 차기 JVT 표준화 회의에서 발표할 예정이다.

IV. 모의 실험 결과

제안된 레퍼런스 프레임 탐색 제한 알고리즘을 Chen 의 고속 움직임 추정 알고리즘과 함께 JM 6.1e 소프트웨어에 추가하여 부호율-왜곡-속도 특성에 대하여 실험하였다. 실험 조건은 예측 모드는 7 가지를 모두 사용하였으며, 움직임 벡터 탐색 범위는 가로, 세로 16 화소

로 하였다. 엔트로피 코딩 알고리즘은 CAVLC 를 사용하였으며, R-D 최적화, Hadamard 변환, 그리고 B 프레임은 사용 하지 않았다. 모든 영상은 IPPP 구조로 정화질(constant-quality) 부호화하였다(QP 값은 31). 움직임 추정에 사용하는 레퍼런스 프레임의 수는 H.264 가 지원하는 최대의 수인 5 로 하였다. 실험에 사용된 영상 시퀀스는 foreman, akiyo, 원령 공주, bugs life, 뮤직 비디오('허락', 베이비 복스), 축구경기(한국 vs.미국, 불가리아 vs. 덴마크) 등 총 7 가지 영상 시퀀스를 사용 하였으며, 각각의 시퀀스 당 100 프레임을 코딩 하였다. 성능 비교의 대상은 레퍼런스 프레임 탐색 제한을 추가하지 않은 Chen 의 알고리즘, Chen 의 알고리즘에 움직임 벡터 탐색범위제한 옵션을 선택한 경우, 그리고 Chen 의 알고리즘에 단일 레퍼런스 프레임을 선택한 경우 등 세가지로 하였다. 표 3 과 4,5 가 각각 위 세 가지 경우의 결과를 요약하고 있다.

표 3. Chen 의 움직임 추정 알고리즘에 레퍼런스 프레임 탐색 제한 기법을 결합함에 따른 왜곡-비트율-속도 성능 개선

	PSNR Increase	BitRate Saving	Time Saving
Maximum	0.08%	0.89%	51.92%
Minimum	-0.02%	-0.48%	43.35%
Average	0.01%	0.27%	46.97%

표 4. Chen 의 움직임 추정 알고리즘에 움직임 벡터 탐색범위제한 옵션을 사용하는 경우에 대한 레퍼런스 프레임 탐색 제한 기법을 결합하는 경우의 왜곡-비트율-속도 성능 개선

	PSNR Increase	BitRate Saving	Time Saving
Maximum	0.21%	4.28%	32.61%
Minimum	-0.01%	-0.62%	13.82%
Average	0.07%	1.08%	19.41%

표 5. Chen 의 움직임 추정 알고리즘에서 단일 레퍼런스 프레임 움직임 추정을 사용하는 경우에 대한 레퍼런스 프레임 탐색 제한 기법을 결합하는 경우의 왜곡-비트율-속도 성능 개선

	PSNR Increase	BitRate Saving	Time saving
Maximum	0.48%	13.08%	-76.61%
Minimum	0.03%	0.07%	-143.19%
Average	0.18%	3.72%	-119.53%

표 3 에서 보인 바와 같이 본 논문에서 제안하는 레퍼런스 프레임 탐색 제한 알고리즘을 결합함으로써 Chen 의 알고리즘은, 왜곡-비트율 성능에서 각각 0.1%와 1%미만의 변동 속에서, 최소 43.35%에서 최대 51.92%, 평균적으로 46.97%의 속도가 개선된다.

표 4 는 기존의 표준 방식인 레퍼런스 프레임 인덱스와 예측 모드에 따른 움직임 벡터 탐색범위의 제한보다는, 그러한 제한은 두지 않되 탐색할 레퍼런스 프레임을 제한하는 방식이 왜곡-부호율-속도의 모든 면에서 보다 효과적인 방식임을 말하고 있다. 특히 최대의 성능 차이를 살펴보면 부호율 절감이 4.28%에 이르는데, 이는 레퍼런스 프레임 인덱스 등에 따른 움직임 벡터 탐색범위의 일률적인 축소가 매우 비효율적으로 동작하는 영상이 존재함을 의미한다.

표 5 에서 주목할 점은 부호율 이득이 최대 13.08%라는 점이다. 이것이 의미하는 것은 영상에 따라서는 단일 레퍼런스 프레임으로는 적절한 움직임 추정이 불가능할 수도 있으며, 이 경우 소요 시간의 증가에도 불구하고, 다중 레퍼런스 프레임이, 당연하게 사용 되어야 한다는 것을 의미한다. 이때 지불해야 하는 것은 제안하는 알고리즘을 따를 때 약 2 배 정보의 소요 시간의 증가라는 것을 표 5 가 보여주고 있다.

V. 결론

본 논문에서는, 모드간 최적 레퍼런스 프레임의 높은 일치 확률을 응용하여, 기존의 고속 움직임 추정 알고리즘에 레퍼런스 프레임 탐색 제한 알고리즘을 결합함으로써, 부호율-왜곡 성능의 저하 없이 H.264 부호기의 속도를 평균적으로 약 45% 정도 향상시킬 수 있음을 보였다. 이는 모바일 환경에서 H.264 를 실시간 동영상 서비스에 응용할 수 있는 가능성을 제시한다는 점에서 중요한 의의를 갖는다.

참고문헌

[1] Zhibo Chen, Peng Zhou, Yun He, "Fast Integer Pel and Fractional Pel Motion Estimation for JVT", JVT-F017.doc, 6th Meeting: Awaji, Japan, Dec, 2002.

[2] Peter Kuhn, Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation, Kluwer Academic Publishers, 1999.