

Philips TM1300 미디어 프로세서에 적합한 효율적인 움직임 예측 방법

*서창호, *오승준, **양창모

*광운대학교, **전자부품연구소

coldfire@hanafos.com, sjoh@media.kw.ac.kr, and cmyang@nuri.keti.re.kr

An Efficient Motion Estimation Scheme for Philips TM1300 Media Processor

*Changho Seo, *Seoung-Jun Oh, and **Changmo Yang

*Kwangwoon University, and **KETI

요 약

본 논문에서는 영상의 특성을 이용하여 TM1300 미디어 프로세서에 적합한 효율적인 움직임 탐색 방법을 제안한다. 주위 블록 간의 공간 상관 관계를 이용해 예측 움직임 벡터를 구하고 이를 기준으로 탐색 영역을 축소하여 전체 연산량을 줄인다. TM1300의 특성을 활용하여 메모리 사용 효율 증대 및 비교, 분기의 사용을 줄여 최적화에 유리한 움직임 예측 방법을 제안한다. 제안된 방법을 Philips 140 MHz TM1300 상의 MPEG-4 SP@L3 (Simple Profile Level 3) 부호화기에 적용한 결과 화질은 전역 탐색 방법에 근접하며 기존의 고속 움직임 예측 방법에 비해 메모리 접근 및 처리 속도 면에서 향상된 성능을 보였다.

I. 서론

최근 디지털 멀티미디어 정보 서비스들이 급속도로 확산 되고 있다. 이러한 멀티미디어 정보의 효율적 전송과 저장을 위해 국제 표준화 기구에서 표준들을 정하고 있다. 그중에 하나인 MPEG-4 표준은 저가격이며 고성능인 멀티미디어 통신 서비스를 고려하여 유동적으로 기존의 방식 및 새로운 기능들을 모두 지원할 수 있는 부호화 도구와 알고리즘을 제공한다. 그러나 MPEG-4 비디오 부호화기는 연산량이 많아 실시간 부호화기 설계가 어렵다. 특히 부호화된 영상의 화질과 비트율을 좌우하는 비디오 부호화기의 핵심 요소인 움직임 예측 (Motion Estimation)은 대부분의 비디오 부호화기에서 가장 많은 연산을 필요로 하는 부분이다.

움직임 예측을 위해 일반적으로 사용되는 전역 탐색 방법 (Full Search Algorithm)은 블록 정합 방법 (Block Matching Algorithm)을 기반으로 탐색 영역 내의 모든 매크로블록 (Macroblock)을 검사하여 가장 작은 왜곡 (distortion)을 가진 매크로블록을 찾는 방식이다. 그러나 탐색 영역 내의 모든 탐색점 (search point)을 검사하는 것은 많은 연산량을 필요로 한다. 이러한 이유로 다양한 방식의 고속 움직임 예측 방법들이 제안되었다. 블록 정합 방법 기반의 고속 움직임 예측 방법은 탐색점 (Search Point)의 수를 줄임으로써 적은 SAD (Sum of

Absolute Difference) 연산을 수행하는 해결 방법을 채택하였다. 그러나 SAD 연산을 줄이기 위해 따르는 화질의 손상을 줄이기 위해 복잡한 과정이 추가된다. 이러한 복잡한 과정의 추가로 인해 기존의 고속 움직임 예측 방법은 TM1300 프로세서 구조에서 최적화된 성능을 내기에 적당하지 않다.

본 논문에서는 영상의 특성을 이용하여 TM1300 프로세서에 효율적인 움직임 탐색 방법을 제안한다. II장에서는 본 논문에서 제안하는 움직임 예측 방법의 배경을 설명하고 III장에서는 제안하는 움직임 예측 방법을 설명하며 IV장과 V장에서 실험결과와 결론을 보여준다.

II. 제안된 방법의 배경

TM1300의 VLIW (Very Long Instruction Word) CPU는 매 명령 사이클 (Cycle)당 5개의 명령어 수행할 수 있다. 각각의 명령어는 27개의 기능별 수행기중 하나에 입력되어 처리된다. 또한 이 기능별 수행기들은 완전히 파이프라인 구조로되어 각 명령어 수행 시에 소요되는 클럭 수에 관계없이 매 클럭사이클 마다 새로운 동작의 수행이 가능하다. 내장 DSP 유닛은 SIMD (Single Instruction Multiple Data)를 지원한다. SIMD는 32비트 데이터 처리를 4개의 8비트 혹은 2개의 16비트

로 병렬 처리하여 처리속도를 높일 수 있다.

DSF 유닛은 움직임 예측을 위해 SAD 값을 구하는 명령어를 제공하며 SIMD를 통해 동시에 4개 화소의 SAD를 구할 수 있다. VLIW CPU에서 SAD는 2개의 명령어를 동시에 사용할 수 있으므로 매 클럭사이클마다 8개 화소의 SAD를 구할 수 있다.

TM1300 VLIW CPU의 메모리 접근은 32KB의 명령어캐시 (Instruction Cache)와 16KB의 데이터캐시 (Data Cache)를 통하여 외부 메모리 혹은 I/O와 연결된다. 이는 외부 SDRAM의 속도가 연산 속도를 따라오지 못하기 때문이다. 데이터캐시는 블록 단위로 관리된다. 하나의 블록은 64바이트이며 한번의 외부 SDRAM의 읽기/쓰기는 64바이트 단위로 전송된다. 비디오를 부호화하기 위해서는 적어도 한 개의 화면을 저장해야 하고 이는 데이터 캐시에 모두 적재될 수 없다. 따라서 메모리에 접근을 줄이는 방법은 속도 향상에 지대한 영향을 미친다.

앞에서 언급한 바와 같이 블록 정합 방법 기반의 고속 움직임 예측 방법은 SAD 연산량을 줄이기 위해 복잡한 과정이 불가피하다. VLIW CPU와 같은 명령어 기반 병렬 처리에 의해 성능을 향상시키는 프로세서에서는 이런 복잡한 과정에서 생기는 분기, 비교 등의 증가는 성능 저하를 유발한다. 이러한 구조를 채택한 다른 프로세서에서는 SAD 연산량이 압도적으로 많으므로 상대적으로 다소의 효율이 저하되더라도 탐색점을 줄이는 방법이 나은 선택이었다. 그러나 TM1300과 같은 미디어 프로세서의 특수한 구조에서는 표 1에서와 같이 SAD 연산량이 줄어들므로 메모리 접근, 분기 및 비교와 같은 다른 요소들의 영향력이 커지게 된다. 고속 움직임 예측 방법은 이전 과정의 결과를 기준으로 다음의 행동을 결정하는 패턴을 가지고 있다. 이는 불 착한 메모리의 접근을 유발하여 데이터캐시의 사용을 저하시킬 뿐 아니라 많은 분기를 포함하고 있어 코드의 최적화를 어렵게 한다.

표 1. SAD 명령어 사용 결과

		Total Cycle(*1000 cycle)
SAD 명령어 사용	SAD_Macroblock	69733
	FindHalfPel	17670
SAD 명령어 사용 안 함	SAD_Macroblock	634224
	FindHalfPel	280953

III. 제안된 방법

1. 움직임 벡터 예측

TM1300에서 임의의 메모리 접근은 데이터의 크기 관계없이 64바이트 단위로 전송한다. 그러므로 전역 탐색 방법의 순차적인 메모리 접근은 메모리 사용면에서 고속 움직임 탐색 방법보다 월등한 효율을 가지고 있으

나 전체적인 연산량이 많아서 고속 움직임 탐색 방법의 많은 비교/분기 부담 (Overhead)을 고려하더라도 상당히 느리다. 제안된 방법은 전역 탐색 방법의 단점인 많은 연산량을 보완하기 위해 MPEG-4에서 사용하고 있는 중간값 (Median) 움직임 벡터 예측을 사용한다. 움직임 벡터의 분포는 움직임이 많은 영상일수록 원점을 기준으로 했을 시 보다 움직임 벡터 예측을 통해 선택된 점의 주변에 존재할 확률이 증가한다.[7] 이러한 특성을 이용해 탐색 영역을 축소 할 수 있다.

2. 블록 탐색

앞 절의 실험 결과들을 기준으로 본 논문에서는 예측 움직임 벡터를 기준점으로 9x9의 크기를 가지는 고정 블록을 탐색한다. 메모리 사용 효율을 높이기 위해 탐색은 저장된 순서를 따라 좌에서 우로 순차적으로 탐색한다. TM1300에서 SAD 연산 기능은 데이터 정렬 (alignment)의 문제로 연속된 4개의 매크로블록을 한번에 계산하는 것이 유리하다. TM1300에서 외부 메모리로부터의 데이터 읽기/쓰기는 64바이트 단위로 데이터캐시에 전송되고 횡축으로 연속된 4개의 매크로블록을 한번에 계산하는 것이 구조적으로 효율적이기 때문에 횡축으로의 순차적인 탐색은 최적의 성능을 낼 수 있다.

움직임 벡터 탐색의 배경이 탐색 영역 내의 최소 SAD 값을 가지는 점이 하나며 최소점에서 멀어 질수록 SAD 값이 커진다는 것을 이용하면 간단한 비교에 의해 탐색 블록 내의 탐색 영역을 반으로 줄일 수 있다. 탐색 블록 안에서 예측된 움직임 벡터가 속한 라인의 최소 SAD 값을 가지는 점과 그 바로 위의 라인 내의 최소 SAD 값을 가지는 점을 비교하여 SAD 값이 작은 쪽이 속한 영역만을 검색할 수 있다. 이 방법은 첫 번째 과정의 결과가 다음의 행동에 영향을 미치나 단순화가 가능하여 성능 저하가 거의 없으며 연산량을 반으로 줄일 수 있는 장점이 있다.

3. 제안된 방법

본 논문에서 제안하는 방법을 그림 1을 참조하여 보면 다음과 같다.

단계 1) 움직임 벡터 예측을 통해 예측 움직임 벡터 ①을 찾는다.

단계 2) 예측 움직임 벡터 ①이 속한 라인 L1을 모두 탐색하여 최소 SAD 값을 가지는 ②를 찾는다.

단계 3) L1의 한줄 위 라인 L2를 모두 탐색하여 최소 SAD 값을 가지는 ③를 찾는다.

단계 4) ②와 ③을 비교하여 ②의 값이 ③보다 작다면 Rd 영역을 모두 탐색한다. 반대의 경우는 Ru 영역을 모두 탐색하여 최소 SAD 값을 가지는 ④를 찾는다.

제안된 방법은 4 단계로 되어 있으며 하나의 조건 만을 가진다. TM1300은 간단한 조건을 1-2 명령사이클에 처리할 수 있는 특수 명령어를 지원하므로 이를 사용하여 분기를 없앨 수 있다.

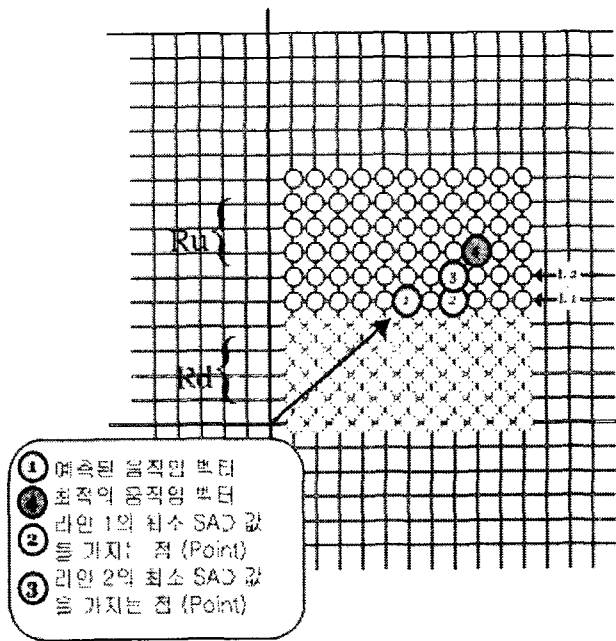


그림 1. 제안된 방법의 탐색 과정

IV. 실험결과

본 논문에서 제안한 움직임 탐색 방법의 성능을 측정하기 잘 알려진 여러 개의 실험 영상에 대해 고속 움직임 예측 방법인 TSS (Three Step Search)와 전역 탐색 방법을 적용하여 부호화 속도, 명암 성분에 대한 PSNR을 비교 하였다. 실험에 사용된 MPEG-4 부호화기는 TM1300에 최적화 되어있지 않으며 모션 움직임을 위한 매크로 블록의 SAD 값을 구하는 부분만이 최적화 되었다.

표 2. CIF 고정비트율 384K에서 Foreman 영상 부호화 속도 (fps)

	초당 부호화 프레임 수
전역탐색방법	2.5
TSS	7.6
제안된 방법	8.3

표 2는 CIF 크기의 Foreman 영상을 384K의 고정비트율로 부호화할 때의 부호화 속도이다. 제안된 방법은 TSS에 비해 9%의 속도 우위를 보여준다.

표 3. CIF 384K에서 영상별 명암 성분의 300 프레임 평균 PSNR (dB)

	전역탐색방법	TSS	제안된 방법
Foreman	32.91	32.24	32.54
Coastguard	29.60	29.35	29.60
Table	34.04	33.50	33.58

표 3에서 나타난 바와 같이 제안된 방법은 전역 탐색 방법에 근접한 화질을 보이고 있다.

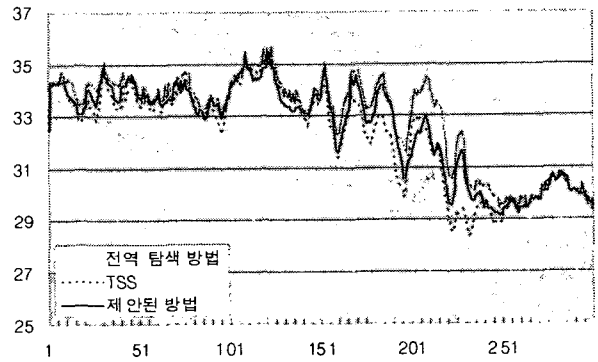


그림 2. Foreman CIF 영상을 384K로 부호화시 탐색 방법에 따른 프레임당 명암 성분의 PSNR

그림 2는 CIF 크기의 Foreman 영상을 384K로 부호화시 각 프레임 당 PSNR을 그래프로 나타낸 것이다. 제안된 방법은 전역 탐색 방법에 근접한 결과를 보이다 170 프레임 근처에서 220 프레임 사이의 화면 전체가 급격히 변하는 장면에서 성능의 저하를 볼 수 있었다.

V. 결론

본 논문에서는 예측 움직임 벡터를 이용해서 검색할 영역을 설정하고 그 영역 내에서 움직임 벡터를 구하는 방법을 제안하였다. 실험에 사용한 고속 움직임 예측 방법들에서는 적은 수의 탐색점을 가짐에도 불구하고 많은 비교 및 분기의 사용과 비효율적인 메모리 사용으로 효율이 떨어졌으나 제안한 방법에서는 다소 많은 탐색점을 가졌으나 효율적인 메모리 사용과 제한된 비교 및 분기 사용으로 움직임 탐색 모듈의 속도를 향상시킬 수 있었다. 제안된 방법은 움직임이 작고 부드러운 영상에서 전역 탐색 방법과 거의 비슷한 화질을 유지하면서 기존 고속 움직임 예측 방법인 TSS 보다 9%정도 빠른 속도를 보였다.

이러한 결과에 따라 본 논문에서 제안된 방법은 실시간 부호화를 필요로 하는 양방향 화상 통신 및 영상 편집 등에 적용이 가능하다.

참고문헌

- [1]. Philips "TM1300 Preliminary Data Book", Aug., 1999.
- [2]. Philips Semiconductors "Book2 - Cookbook Part D:Optimizing TriMedia Application", 08 1999.
- [3]. Peter Kuhn, "Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation", 1999 Kluwer Academic Publishers
- [4]. ISO/IEC, 14496-IS, "Generic Coding of Audio-Visual Objects - Part 2 : Visual", Dec. 1998.
- [5]. ISO/IEC, JTC1/SC29/WG11, N3515, "MPEG-4 Video Verification Model Version 17.0", Jul. 2000.
- [6]. ISO/IEC 14496-5/IS, "Generic coding of Audio-Visual Objects - Part : 5 Simulation Software, Dec., 1998.
- [7]. A. M. Tourapis, O. C. Au, and M. L. Liou, "Predictive Motion Vector Field Adaptive Search Technique (PMVFAST) - Enhancing Block Based Motion Estimation", in *proceedings of Visual Communications and Image Processing 2001 (VCIP-2001)*
- [8] Yu-Wen Huang, Shyh-Yih Ma, Chun-Fu Shen, and Liang-Gee Chen, "Predictive Line Search: An Efficient Motion Estimation Algorithm for MPEG-4 Encoding Systems on Multimedia Processors", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 111-117, Jan. 2003.