

H.264 의 고속 정수 단위 화소 움직임 예측을 위한 개선된 Uneven Multi-Hexagon-grid 검색 알고리즘

이인직*, 김정길*, 김신덕*
*연세대학교 컴퓨터과학과
e-mail : ijlee@parallel.yonsei.ac.kr

Fast Uneven Multi-Hexagon-Grid Search Algorithm for Integer Pel Motion Estimation of H.264

In-Jik Lee*, Cheong-Ghil Kim*, Shin-Dug Kim*
*Dept. of Computer Science, Yonsei University

요 약

본 논문에서는 H.264 표준화 기구인 Joint Video Team(JVT) 권고안의 정수 단위 화소 움직임 예측을 위한 Unsymmetrical-cross Multi-Hexagon-grid Search(UMHexagonS) 알고리즘에서 Uneven Multi-Hexagon-grid Search(UMHGS) 부분을 개선한 알고리즘을 제안한다. 제안하는 알고리즘은 이전 프레임의 동일위치 또는 상위 모드에서 이미 선택된 움직임 벡터(MV: Motion Vector)를 이용하여 신호 대 잡음 비(PSNR: Peak Signal to Noise Ratio) 및 평균 비트 율(Average Bitrates)을 유지하면서, 현재 매크로블록의 검색영역을 줄이는 것이 가능하다. 제안하는 알고리즘의 성능은 Full Search Block Matching Algorithm(FSBMA) 및 UMHexagonS 알고리즘의 integer pel 에 대한 SAD(Sum of Absolute Difference) 연산횟수로 비교평가 하였다. 그 결과, FSBMA 에 비하여 평균 97.64%, UMHexagonS 에 비하여는 평균 17.48%의 연산횟수를 감소시키는 우수함을 보였다.

1. 서론

아날로그 영상을 디지털 미디어에 저장하기 위한 필요성으로 인해 영상압축기술은 오래 전부터 연구되어 왔다. H.261, MPEG-1, MPEG-2, H.263, MPEG-4 그리고 H.264 는 익히 잘 알려진 영상압축 알고리즘에 대한 표준들로서, Hybrid block-based MC(Motion Compensation)/DCT(Discrete Cosine Transform) 알고리즘을 기반으로 하고 있다[1]. 이중 H.264 는 최신의 영상압축 표준으로서 국내에서는 디지털 멀티미디어 방송(DMB: Digital Multimedia Broadcasting)의 표준 영상압축 알고리즘으로 채택되기도 하였다.

이처럼 H.264 는 이전의 압축 알고리즘들과 동일하게 비디오 통신분야에 대한 적용을 목표로 하고 있으며, 높은 압축효율을 얻기 위해 기존의 압축표준들과 달리 움직임 예측(ME: Motion Estimation)을 위한 여러 가지 방법들이 추가 되었는데, 가변블록 움직임 보상(MC), 1/4 화소 정확도를 갖는 MV, 그리고 복수참조 영상과 같은 방법들이 이에 해당한다[2].

일반적으로 Hybrid block-based MC/DCT 부호화 방법에서는 움직임 보상을 위한 ME 가 압축수행시간의 상당 부분을 차지한다. 그리고 H.264 의 경우에는 기존의 압축표준들과 달리 위에서 설명한 추가적인 ME 방법들로 인해, ME 가 전체 압축수행시간의 60~80% 이상을 차지한다[3]. 그러므로 H.264 를 실시간 응용에 적용하기 위해서는 ME 의 수행속도 향상은 필수적이고, 이를 위해 다양한 고속 움직임 예측(FME: Fast Motion Estimation) 알고리즘이 제안되고 있다[4][5][6][7][8].

UMHexagonS 알고리즘은 위에서 언급했던 FME 알고리즘들 중의 하나로 H.264 표준화 기구인 JVT 의 권고안으로 채택되었다. UMHexagonS 알고리즘에서는 인코딩시간을 줄이기 위해 MV 예측, 예측된 SAD 값을 이용한 early termination, 그리고 3 단계로 이루어진 검색패턴을 이용한 검색과 같은 방법들을 사용하였다. 그러나 여전히 UMHGS 부분에서는 성능개선의 여지가 있다. 따라서, 본 논문에서는 UMHexagonS 알고리즘에서 사용되는 이전 프레임의

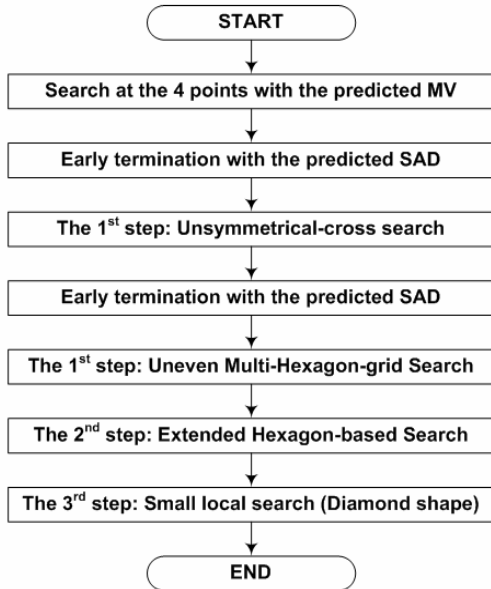
동일한 위치 또는 상위 모드에서 이미 선택된 MV 를 이용하여 현재 매크로블록의 검색영역을 줄이는 FME 알고리즘을 제안한다. 그 결과, 제안하는 알고리즘은 기존의 FSBMA 및 UMHexagonS 알고리즘과 비교하여 PSNR 과 평균 비트 율을 유지하면서, FSBMA 에 비하여 평균 97.64%, UMHexagonS 에 비하여는 평균 17.48%의 SAD 연산 횟수를 감소시킬 수 있었다.

본 논문의 2 장에서는 UMHexagonS 알고리즘의 주요부분에 대해 간략히 살펴보고, 3 장에서 UMHGS 알고리즘과 이를 개선한 제안하는 알고리즘의 수행절차에 대해 설명하도록 한다. 그리고 4 장에서는 FSBMA 및 UMHexagonS 알고리즘과 제안하는 알고리즘의 성능을 PSNR 및 평균 비트 율과 정수 단위 화소 검색에 대한 SAD 연산회수를 이용하여 비교하였고, 5 장에서 결론에 대하여 논하도록 하겠다.

2. Unsymmetrical-cross Multi-Hexagon-grid Search

2.1 UMHexagonS 알고리즘

UMHexagonS 알고리즘은 크게 두 부분으로 나눌 수 있다. 첫 번째는 일반적인 움직임의 처리를 위해 예측된 MV 에 의해 결정된 검색 시작점들에서 Diamond Shape Search(DSS) 형태의 4 점의 위치에서 검색을 수행하고, 두 번째는 비정형적인 움직임의 처리를 위해 3 단계로 나누어 UMHGS, Extended Hexagon-base search(EHS) 그리고 DSS 를 수행하며, 각 단계에서 선택된 MV 는 다음 단계의 검색 시작점으로 이용된다. (그림 1)은 UMHexagonS 알고리즘의 순서도이다.



(그림 1) UMHexagonS

전체 알고리즘을 수행하는 동안 이전 프레임과 상위 모드의 SAD 값 그리고 median 연산된 SAD 값을 이용하여 두 번의 early termination 을 수행한다[7][9].

2.2 검색 시작점 예측

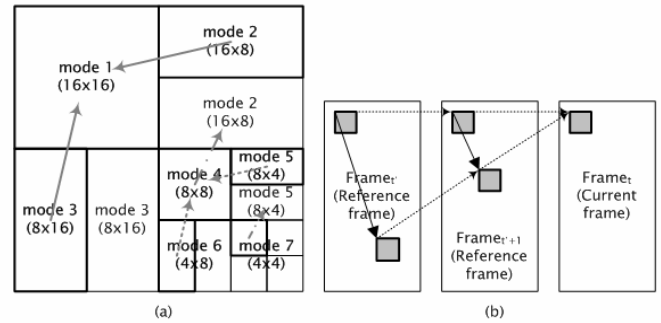
UMHexagonS 알고리즘은 최소의 검색시간으로 최적의 MV 를 구하기 위해 2.1 절에서 언급한 첫 번째 부분에서 사용되는 검색 시작점 예측을 다음과 같은 4 가지 방법으로 수행한다.

Median prediction 은 현재 블록과 주변 블록 사이의 공간적 연관성을 이용하여, 이미 인코딩된 현재블록 주변에 위치한 블록들의 중간 값을 갖는 MV 를 선택하게 된다.

$$MV_{pred} = median(MV_a, MV_b, MV_c)$$

Up-Layer prediction 은 매크로블록의 크기로 구분되는 7 가지 모드 사이의 연관성을 이용하는 것으로, 크기가 작은 하위모드의 매크로블록에 대한 예측을 수행하기 위해 상대적으로 크기가 큰 상위 모드의 MV 를 사용하여 예측을 수행한다.

- Mode2,3 - Mode1 사용
- Mode4 - Mode2 사용
- Mode5,6 - Mode4 사용
- Mode7 - Mode5 사용



(그림 2) (a) Up-Layer (b) Neighboring Ref-frame

Corresponding-block prediction 은 참조프레임의 동일한 위치에 존재하는 매크로블록의 MV 를 현재 프레임의 매크로블록 인코딩의 검색 시작점으로 선택한다.

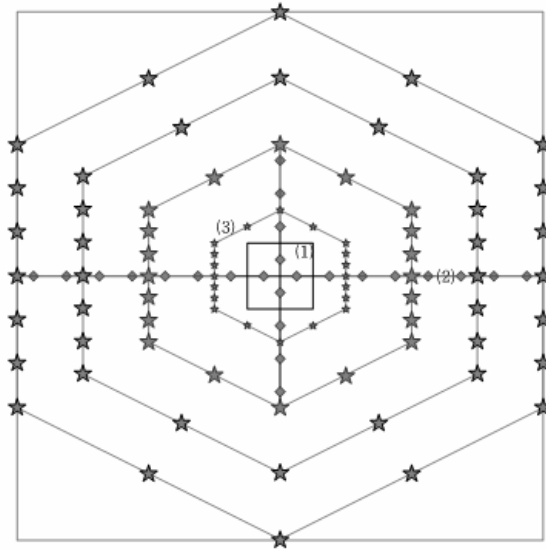
$$MV_{pred(t)} = MV_{t-1}$$

Neighboring Ref-frame prediction 은 연속적인 프레임들 사이의 시간적 연관성을 이용하여, 여러 개의 참조프레임들 사이의 동일한 위치에 존재하는 매크로블록의 MV 에 대해 상대적 비율을 적용하여 예측한다.

$$MV_{pred(t)} = MV_{t'+1} \times \frac{t - t'}{t - t' - 1}$$

2.3 Uneven Multi-Hexagon-Grid Search (UMHGS)

2.1 절에서 언급한 두 번째 부분인, UMHGS 의 기본적인 구조에 대해 살펴보면 (그림 3)과 같다.



(그림 3) UMHGS (검색영역 $w = 16$)

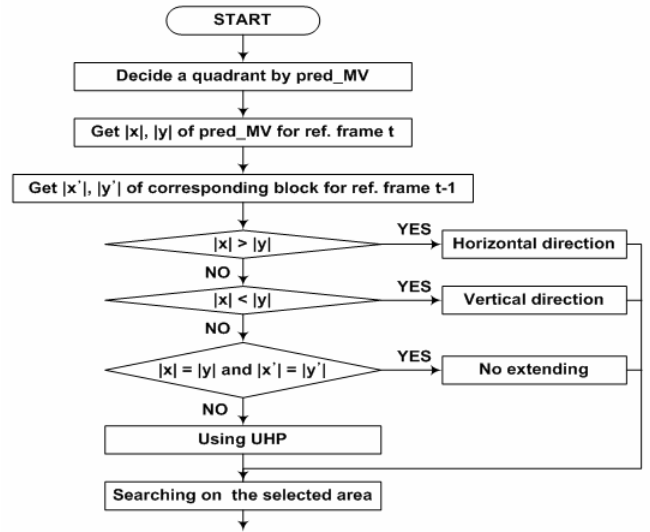
UMHGS는 이전 단계에서 결정된 검색 시작점으로부터 검색을 시작하며, 다음의 세 단계로 나누어 수행된다. 첫 번째 단계에서는 가로방향의 길이가 세로에 비해 상대적으로 긴, 비대칭적 십(+)자 모양의 검색을 수행하고(1), 두 번째 단계에서는 검색영역이 2인 전역 검색(Full Search)를 수행한다(2). 세 번째 단계에서는 (그림 3)의 (3)과 같은 Uneven Hexagon Pattern(UHP)의 16개 점에 대한 검색을 수행한다. 이때 hexagon의 크기가 1에서부터 $w/4$ 까지의 비율로 확장되면서 반복적인 검색을 수행하게 되고, 검색한 점들 중 minimum SAD 값을 갖는 위치를 UMHGS의 다음 단계인 EHS의 검색 시작점으로 사용하게 된다. 그러므로 UMHGS에서는 $(16+25+16 \times w/4)$ 의 정수 단위 화소에 대한 검색을 수행하게 된다.

3. 제안하는 알고리즘

본 논문에서는 UMHHexagonS 알고리즘 가운데 UMHGS 알고리즘을 개선하여 PSNR과 평균 비트율은 유지하면서 integer pel에 대한 연산량을 줄이는 알고리즘을 제안하고자 한다.

3.1 제안하는 알고리즘

본 논문에서 제안하고자 하는 알고리즘은 이전 프레임의 이미 선택된 동일 위치의 매크로블록 또는 상위 모드의 MV를 이용하여 UMHGS의 UHP의 검색영역을 줄이는 알고리즘이다. 일반적으로 현재의 매크로블록은 이전 프레임의 동일한 위치 또는 상위 모드에서 선택된 MV와 밀접한 연관성을 가지고 있기 때문에, 이러한 연관성을 이용하여 검색영역을 줄이는 것이 가능하다. (그림 4)는 제안하는 알고리즘의 흐름을 보여준다.

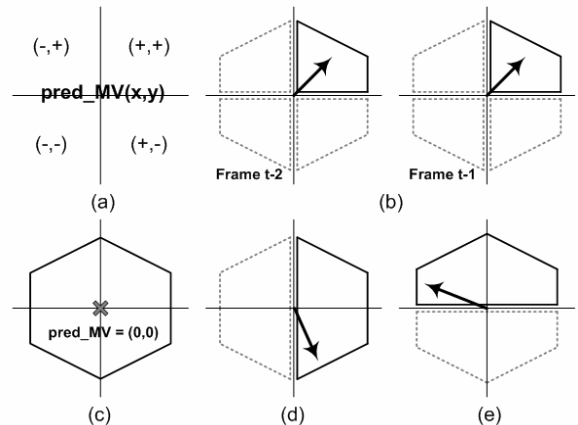


(그림 4) Fast UMHGS

최초 검색영역의 결정에 사용될 predicted MV(pMV_x, pMV_y)를 선택한다. 현재의 매크로블록이 16×16 인 경우, 상위 모드가 존재하지 않으므로, 이전 프레임에서 현재 매크로블록과 동일한 위치의 매크로블록에 대한 MV를 선택하고, 그 외의 모드인 경우 이전 프레임에서 상위모드의 매크로블록에 대한 MV를 선택한다. 선택된 MV를 이용하여 검색할 영역에 대한 사분면을 결정하고, 검색영역의 확장을 검토한다. 최종적으로 검색영역이 결정되면 minimum SAD 값을 찾아내기 위해 최소 $(5 \times w/4)$ 에서 최대 $(9 \times w/4)$ 크기의 정수 단위 화소에 대한 검색을 수행하게 된다.

3.2 검색영역의 결정 및 확장

제안하는 알고리즘은 ME의 수행 성능을 향상시키기 위해 검색영역을 확장하는 방법을 사용한다. (그림 5)는 검색영역의 형태를 결정하는 방법을 보여준다. 특별히 (그림 5)(b)의 경우, 연속적인 두 개의 참조 프레임에서의 MV의 크기를 사용하여 최종적으로 검색영역을 결정한다.



(그림 5) Extend searching area

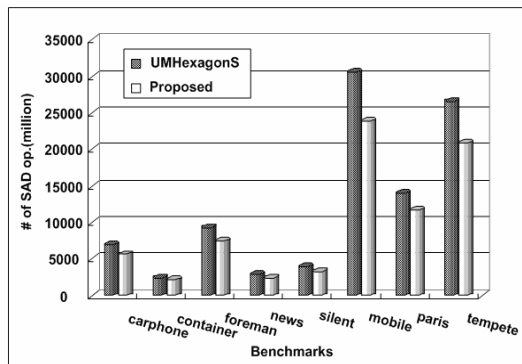
| Seq. | Size | Tot. frm. | PSNR | Bitrate | PSNR | Bitrate | PSNR | Bitrate | | | |
|----------|------|-----------|-------------|---------|---------|------------|-------|---------|----------|-------|---------|
| carphone | QCIF | 381 | Full Search | 39.06 | 150.83 | UMHexagons | 39.04 | 151.18 | Proposed | 39.02 | 150.95 |
| | | 299 | | 38.96 | 37.41 | | 38.96 | 37.42 | | 38.98 | 37.30 |
| | | 399 | | 38.20 | 133.97 | | 38.19 | 133.85 | | 38.19 | 134.33 |
| news | | 299 | | 38.96 | 73.39 | | 38.92 | 73.45 | | 38.94 | 73.55 |
| silent | | 299 | | 38.03 | 82.86 | | 37.99 | 83.43 | | 38.01 | 83.33 |
| mobile | CIF | 299 | Full Search | 35.07 | 1487.42 | UMHexagons | 35.06 | 1484.30 | Proposed | 35.06 | 1484.65 |
| paris | | 299 | | 37.61 | 472.58 | | 37.60 | 473.75 | | 37.59 | 473.89 |
| tempete | | 259 | | 36.98 | 1126.17 | | 36.96 | 1127.51 | | 36.98 | 1127.71 |

(그림 7) 실험 결과(QP = 28, Freq. = 30Hz)

검색할 사분면의 결정을 위해 선택된 pMV(pMV_x, pMV_y)의 x/y 성분의 부호에 따라 (그림 5)(a)와 같은 방법을 사용하여 검색을 수행할 사분면을 결정한다. 사분면이 결정되면, 검색영역의 확장여부를 결정하기 위해, pMV의 x/y 성분에 대한 절대값 |pMV_x|와 |pMV_y|를 구하여 이들의 차이 값을 이용해 (그림 5) (b)(c)(d)(e)와 같이 종/횡으로의 확장을 검토한 후, 결정된 검색영역에 대해 minimum SAD 값을 구하기 위한 검색을 수행한다.

4. 실험 및 결과 분석

본 논문에서는 JVT에서 참고 소프트웨어로 공개한 JM 10.1을 이용하여 실험하였고, 벤치마크로는 일반적으로 많이 사용되는 5개의 QCIF 및 3개의 CIF test sequence 들을 사용하였다.



(그림 6) 실험 결과(SAD 연산 횟수)

(그림 6)은 각 벤치마크에 대해 UMHexagonS와 제안하는 알고리즘을 적용했을 경우의 SAD 연산 횟수를 보여준다. 제안하는 알고리즘의 경우 FSBMA에 비해 평균 97.60%, UMHexagonS 알고리즘에 비해 16.14%만큼의 SAD 연산이 감소됨을 알 수 있다.

(그림 7)은 각 벤치마크에 대해 FSBMA, UMHexagonS, 그리고 제안하는 알고리즘을 적용했을 경우의 PSNR과 평균 비트율을 나타내는데, 평균적으로 FSBMA에 비하여 -0.2dB/+0.13kbit/s, UMHexagonS에 비하여는 0.0dB/+0.10kbit/s의 차이를 보임을 알 수 있다.

5. 결론

본 논문에서는 이전 프레임의 이미 선택된 동일 위치

의 매크로블록 또는 상위모드의 MV를 이용하여, UMHexagonS에서의 검색영역에 대한 형태를 결정하고, 현재 매크로블록의 검색영역을 제한함으로써 정수 단위 화소에 대한 SAD 연산횟수를 줄이는 알고리즘을 제안하였다. 그 결과 PSNR 및 평균 비트율은 유지하면서, FSBMA에 비해 97.64% UMHexagonS 알고리즘에 비해 17.48%의 연산횟수를 감소시켰다.

참고문헌

- [1] Thomas Sikora. " Trends and Perspectives in Image and Video Coding" Proceedings Of The IEEE, Vol. 93, No. 1, JAN. 2005
- [2] ThomasWiegand, Gary J. Sullivan, Gisle Bjøntegaard, and Ajay Luthra. " Overview of the H.264 Video Coding Standard" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 13, No. 7, JULY 2003
- [3] Zhibo Chen, Peng Zhou, and Yun He. " Fast Integer Pel and Fractional Pel Motion Estimation for JVT" JVT-F017.doc, 6th Meeting: Awaji, Island, JP, 5-13 December, 2002
- [4] Woong Il Choi, Byeungwoo Jeon, and Jechang Jeong. " FAST MOTION ESTIMATION WITH MODIFIED DIAMOND SEARCH FOR VARIABLE MOTION BLOCK SIZES" Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on Volume 2, pp. II-371-4 vol.3, 14-17 Sept. 2003
- [5] Libo Yang, Keman Yu, Jiang Li, and Shipeng Li. " An Effective Variable Block-Size Early Termination Algorithm for H.264 Video Coding" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 15, No. 6, JUNE 2005
- [6] M. Yang, H. Cui, and K. Tang. " Efficient tree structured motion estimation using successive elimination" IEE Proceedings. Visual Image Signal Process, Vol. 151, No. 5, October 2004
- [7] Jianfeng Xu, Zhibo Chen, and Yun He. " Efficient Fast ME Predictions and Early-termination Strategy Based on H.264 Statistical Characters" Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on Volume 1, Page(s):218 - 222 Vol.1, 15-18 Dec. 2003
- [8] Zhi Zhou, Ming-Ting Sun, and Yuh-Feng Hsu. " FAST VARIABLE BLOCK-SIZE MOTION ESTIMATION ALGORITHMS BASED ON MERGE AND SPLIT PROCEDURES FOR H.264/MPEG-4 AVC" Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on Volume 3, Page(s):III - 725-8 Vol.3, 23-26 May 2004
- [9] Zhibo Chen, Peng Zhou, and Yun He. "Fast Motion Estimation for JVT" JVT-G016.doc, 7th Meeting: Pattaya II, Thailand, 7-14 March, 2003