

HEXBS를 적용한 고속 가변 블록 움직임 예측

*이규호, 이귀상

전남대학교 전산학과

e-mail : herolkh@cad.chonnam.ac.kr

A Fast Variable Block-size Motion Estimation by Using Hexagon-based Search

*Kyu-Ho Lee, Guee-Sang Lee

Dept. of Computer Science, Chonnam National University

Abstract

가장 최근의 동영상 압축 표준인 H.264는 매크로블록의 최적 모드를 결정하기 위하여 총 7가지 모드를 사용하여 움직임 예측을 수행하기 때문에 부호화 효율과 수행 성능 면에서 기존의 방식에 비해 좋은 결과를 보이지만, 움직임 예측 과정에서 많은 계산 시간을 필요로 한다는 것이 커다란 단점으로 지적되고 있다. 따라서, 본 논문에서는 H.264에서 가변 블록 단위의 움직임 예측 시 인접한 블록과의 상관성을 분석하여 다음 움직임 예측 모드의 사용 여부를 결정하는 블록 병합 알고리즘을 HEXBS(Hexagon-based Search) 고속 움직임 탐색 알고리즘에 적용하여 움직임 예측 시간을 효율적으로 절약하는 방법을 제안한다.

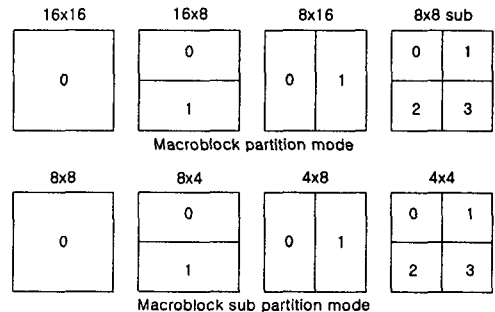
단위의 움직임 예측 방법을 채택하고 있다[1]. 이러한 가변 블록 단위의 움직임 예측을 사용하는 이유는 블록의 크기나 모양이 움직임 예측에 있어서 중요한 요소로 작용하기 때문이다.

H.264에서 사용되는 여러 가지 가변 블록 단위의 움직임 예측 방법이 고정 블록 단위의 움직임 예측보다 높은 수행 성능을 보이는 실험을 통하여 그 우수성이 입증되었다[2]. 하지만, H.264 부호화기가 높은 수행 성능을 가져왔을지라도 복잡도 측면에서는 기존의 동영상 압축 표준에 비하여 엄청난 계산량과 시간을 요구하고 있다. 그 중 부호화기에서 움직임 예측 부분이 약 70%~80%를 차지하고 있다. 그러므로 H.264 부호화기에서 7가지 모드의 움직임 예측 과정을 최소화 할 수 있는 효율적인 방안이 필요하다. 기존의 움직임 예측 알고리즘은 탐색점의 수를 감소시키는 방향으로 많이 제시되었다[3-5].

I. 서론

최근 들어 무선 통신 기술과 인터넷 보급률의 확산으로 인하여 동영상 서비스에 대한 요구가 높아져 가고 있는 가운데 가장 최근에 표준화가 완성된 H.264는 인터넷을 통해 DVD 수준의 동영상까지 전송할 수 있는 차세대 동영상 압축 기술로서 자리를 잡아가고 있다. H.264는 현재 동영상 압축 국제 표준으로 널리 사용되는 H.261/H.263 과 MPEG-1/MPEG-2/MPEG-4에 비하여 새로운 요소들과 방식들이 채택되면서 이 분야에서 중요한 기술로 부각되고 있다.

H.264 부호화기가 기존의 부호화기에 비해 가장 크게 달라진 부분의 하나는 움직임 예측 방법이다. 기존의 방식에서는 16x16 또는 8x8의 고정된 블록 단위로 처리했던 반면 H.264에서는 매크로블록을 총 7가지 블록 단위로 나누어 움직임 예측을 수행하는 가변 블록

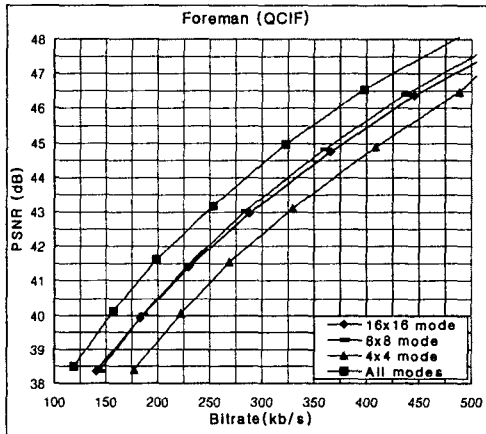


<그림 1> H.264 움직임 예측 모드

그림 1은 H.264에서 움직임 예측 과정에서 사용되는 7가지 모드를 보여주고 있다. 기존의 고정 블록 단위의 움직임 예측 방법과 달리 가변 블록 단위로 여러

가지 블록 형태의 움직임 예측이 이루어진다. 매크로 블록의 최적의 모드를 결정하기 위하여 매크로블록을 4개의 8x8 블록으로 나누고, 다시 4x4 블록으로 나누어서 4x4 블록을 시작으로 16x16 매크로블록 단위까지 움직임 예측이 수행된다.

H.264의 7가지 움직임 예측 모드 중에서 대표적인 단일 모드인 16x16, 8x8, 4x4 세 가지 모드의 수행 성능을 실제 실험을 통하여 R-D(Rate-Distortion)를 서로 비교해 본 결과 위의 세 가지 단일 모드 중에서는 8x8 모드가 가장 좋은 결과를 보였다[그림2].



<그림 2> Foreman 영상의 비트율-왜곡 곡선

본 논문에서는 실험 결과 가장 좋은 성능을 보인 8x8 움직임 예측 모드를 기본 모드로 사용하여 8x8 모드의 움직임 예측이 끝난 후 인접한 블록과의 거리와 임계값을 이용하여 다음 움직임 예측 모드의 사용 여부를 결정하고, 다음 움직임 예측 모드를 사용할 경우 상향식 방법으로 블록들간의 병합이 전개되는 알고리즘을 소개한다. 병합 절차는 8x8 블록을 시작점으로 블록 사이의 움직임이 거의 없다고 판단되는 16x16 블록 방향으로 병합된다. 그리고 H.264 부호화기의 정수 단위 화소에서의 움직임 탐색 기법인 전역 탐색법(Full-Search) 대신 HEXBS(Hexagon-based Search)를 병합 알고리즘에 적용하였다[6].

II. 제안한 알고리즘

2.1 병합을 위한 임계값 결정 방법

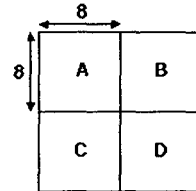
매크로블록의 최적의 모드를 결정하기 위하여 H.264 부호화기에서는 그림 1과 같이 7가지 예측 모드를 사

용하여 움직임 예측이 이루어진다. 대표적인 단일 모드로는 가장 작은 블록 크기의 모드인 4x4 모드를 포함하여 8x8, 16x16 모드가 있다. 실험 결과 대부분의 영상에서 8x8 모드가 다른 단일 모드보다 수행 성능이 좋았음을 확인할 수 있었다. 따라서, 기본 예측 모드로 사용하는 8x8 모드를 시작으로 병합이 이루어진다. 8x8 블록 단위의 움직임 예측이 종료되면 각각의 블록마다 움직임 벡터가 구해진다. 여기에서 다음 모드를 수행하기 전에 블록간의 거리를 구하여 각각의 블록들이 이웃한 블록들과 얼마나 상관성이 있는지 알아본다. 그리고 블록의 거리를 임계값과 비교하여 병합의 여부를 판단한다. 블록간의 거리는 두 블록의 움직임 벡터의 좌표값을 식(1)에 적용하여 얻을 수 있다. 여기에서 $(x_1, y_1), (x_2, y_2)$ 는 두 블록의 움직임 벡터를 나타낸다.

$$Distance = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \text{ -----(식1)}$$

블록간의 병합을 판단하는 임계값을 결정하는 기준으로는 인접한 블록에 대하여 움직임 벡터의 거리가 $\sqrt{2}$ 안에 존재하면 동일한 움직임으로 보는 연구 결과 토대로 $\sqrt{2}$ 를 임계값으로 결정하였다[7].

2.2 제안 병합 알고리즘

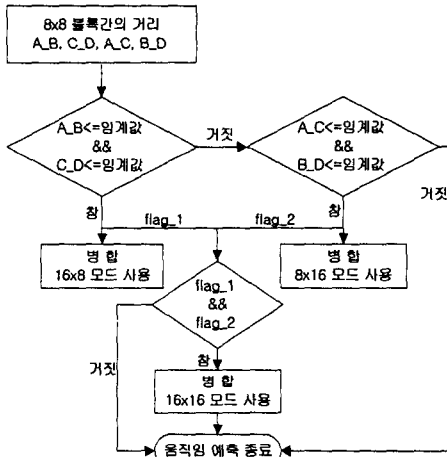


<그림 3> 매크로블록

본 논문에서 사용한 기본 예측 모드인 8x8 모드의 움직임 예측이 실행되면 A, B, C, D 4개의 8x8 블록에 대한 움직임 벡터가 구해진다[그림 3]. (식1)을 이용하여 블록간의 거리를 각각 구한다. 블록간의 거리는 인접한 블록과 병합의 여부를 결정하는 정보로 사용한다. A와 B의 거리가 임계값 안에 존재하고, C와 D의 거리 또한 임계값 안에 존재할 경우 A와 B, C와 D 블록은 서로 병합되고, 16x8 움직임 예측 모드를 실행한다. 같은 방법으로 A와 C의 거리, B와 D의 거리가 임계값 안에 존재할 경우 각각의 블록은 서로 병합되고, 8x16 모드를 적용하여 움직임 예측을 하게 된다. 만약에 전자(16x8 모드)와 후자(8x16 모드)가 모두 참일 경우 다시 두 개의 가변 블록이 병합되어 매크로블

특인 16x16 모드 움직임 예측을 수행한다.

순서도를 이용하여 제안한 병합 알고리즘을 표현하면 그림 4와 같다. A_B, C_D, A_C, B_D는 이웃하는 8x8 블록들간의 거리를 나타낸다. 두 블록 사이의 거리가 임계값을 벗어난 경우에는 인접한 두 블록의 움직임이 서로 다르다고 판단한다. 이런 경우 다음 모드를 사용하지 않기 때문에 불필요한 모드의 사용을 줄임으로써 움직임 예측에 소요되는 시간을 단축시킬 수 있는 이점이 있다.



<그림 4> 블록의 병합 순서도

2.3 HEXBS 움직임 탐색 기법

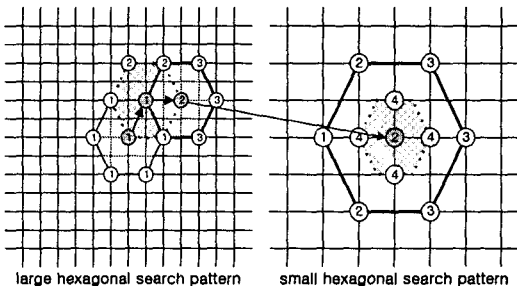
본 논문에서는 제안한 병합 알고리즘에 움직임 예측 시 움직임 벡터의 빠른 검색을 위하여 HEXBS 고속 움직임 탐색 기법을 적용하였다. 현재 JM 부호화기에 구현되어 있는 전역 탐색법은 탐색 원점을 중심으로 나선형의 방향으로 모든 탐색점을 검사한다. 따라서, 많은 계산량을 필요로 한다. 반면, HEXBS는 탐색 원점을 중심으로 먼저 육각형 모양의 6개의 점을 탐색한

다. 만약, 움직임 추정치의 최소값이 원점이 아닌 6개의 탐색점 중 하나이면 그 점을 중심으로 육각형 모양을 형성한 다음 추가로 발생하는 탐색점을 조사해 나간다(LHSP). 그리고 움직임 추정치의 최소값이 육각형 모양의 가운데에 존재할 경우 마지막으로 그 점을 중심으로 동서남북 방향의 4개의 점을 탐색한 후 움직임 탐색을 종료하게 된다(SHSP). 그림 5는 HEXBS의 움직임 탐색 방법을 LHSP와 SHSP로 나누어서 설명하고 있다. 그림에서 숫자는 탐색점이 추가되는 순서를 표현한 것으로 전역 탐색법에 비하여 움직임 탐색점의 계산량을 현저하게 줄일 수 있다.

III. 실험 결과

제안한 알고리즘을 검증하기 위하여 본 논문에서는 JM 6.1a 버전의 부호화기를 사용하여 실험하였다. 기본 움직임 예측 모드로는 일반적으로 수행 성능이 가장 좋은 8x8 모드를 사용하였으며, 상향식 방법의 병합이 이루어진다. 실험 영상은 총 3개의 QCIF 영상으로 Foreman, Stefan, Table-Tennis를 사용하여 실험하였다. 각 영상은 150장이고, 프레임율은 15Hz로 동일한 프레임율을 적용하였다. 양자화 값은 28, 32 두 개의 값을 사용하였고, 각각의 영상에 동일한 양자화 값을 적용하여 실험하였다. 각 영상에 대하여 IPPP의 순서로 부호화하였고, B-프레임은 실험에서 사용하지 않았으며, 엔트로피 부호화 방법으로는 CAVLC를 사용하였다.

계산량과 부호화 시간의 비교는 4가지 움직임 예측 모드(16x8, 16x8, 8x16, 8x8)에 전역 탐색법을 사용한 경우를 본 논문에서 제안한 방법과 비교하였다. 표 1은 각각의 실험 영상의 움직임 예측 시 매크로블록 당 탐색점의 개수를 수치화 한 것으로 전역 탐색법 비해서 제안한 알고리즘이 많은 수의 탐색점을 줄였음을 확인할 수 있다. 부호화 시간은 영상의 종류와 양자화 값에 따라 차이가 있지만, 약 30%~35% 정도의 부호화 시간이 단축되었음을 실험을 통하여 확인할 수 있었다 [표 2]. PSNR의 경우 Stefan 영상은 0.2dB~0.3dB 증가했고, 나머지 영상은 0.15dB~0.2dB 정도의 미세한 감소를 보였다[표 3].



<그림 5> LHSP와 SHSP 움직임 추정 기법

Sequence		Foreman	Stefan	Table-tennis
알고리즘	QP=28	9735.66	9735.66	9735.66
	QP=32	9735.66	9735.66	9735.66
제안 알고리즘	QP=28	75.41	84.72	85.67
	QP=32	78.15	84.65	88.66

<표 1> 매크로블록 당 탐색점의 개수

Sequence		Foreman	Stefan	Table-tennis
알고리즘	QP=28	205.779	255.439	223.689
	QP=32	193.405	233.642	196.859
제안 알고리즘	QP=28	144.108	189.215	145.097
	QP=32	128.733	165.188	132.152

<표 2> 영상의 부호화 시간

Sequence		Foreman	Stefan	Table-tennis
알고리즘	QP=28	35.59	34.08	35.74
	QP=32	32.95	30.53	33.20
제안 알고리즘	QP=28	35.45	34.37	35.67
	QP=32	32.73	30.73	33.06

<표 3> 영상의 PSNR

IV. 결론 및 추후 연구

H.264는 기존의 동영상 압축 표준에서 사용한 움직임 예측 방법 대신 매크로블록의 최적 모드를 찾아내기 위하여 총 7가지 움직임 예측 모드를 사용하여 가변 블록 단위의 움직임 예측을 수행한다. 따라서, 부호화 시간의 대부분이 움직임 예측 과정에서 소요됨으로써 이러한 예측 모드의 사용이 부호화기 전체의 복잡도를 높이는 주요 요인으로 작용하고 있다. 본 논문에서는 이러한 움직임 예측 과정을 줄이기 위해 이웃하는 블록들의 움직임 벡터를 이용하여 인접 블록과의 상관성을 조사하고, 서로 연관이 있는 블록에 대해서만 병합을 실시하며, 기존의 HEXBS 고속 움직임 탐색 기법을 병합 알고리즘에 적용함으로써 HEXBS만 적용할 경우보다 약 30% 정도의 계산량을 추가로 줄일 수 방법을 제시하였다. 모의실험 결과 제안한 알고리즘이 움직임 예측 모드의 사용 여부를 미리 판단하여 불필요한 모드의 사용을 줄였으며, 계산량의 감소와 함께 H.264 부호화 시간을 약 30%~35% 정도 단

축시킬 수 있었다. 추후 본 논문의 병합 알고리즘에 HEXBS를 포함한 다양한 움직임 탐색 기법들을 블록의 형태와 크기에 따라 가변적으로 적용한다면 부호화 효율면에서 더 나은 결과를 가져올 것이다.

참고문헌

- [1] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "Study of Final Committee Draft of Joint Video Specification (ITU-T REC. H.264 | ISO/IEC 14496-10 AVC)". JVT-100.doc, 6th Meeting: Awaji, Island, JP, 5-13 December, 2002.
- [2] Thomas Wiegand, Heiko Schwarz, Anthony Joch, Faouzi Kossentini, and Gary J. Sullivan "Rate-Constrained Coder Control and Comparison of Video Coding Standards", IEEE Trnas. Circuits Syst. Video Technol., vol. 13, NO. 7, July 2003.
- [3] Z.B. Chen, Y. He, "Prediction based Directional Refinement (PDR) algorithm for Fractional Pixel Motion Search Strategy", JVT-D069, 4th Meeting: Klagenfurt, Austria, 22-26 July, 2002.
- [4] J.N.Zhang, Y.W.He, S.Q.Yang, Y.Z. Zhong, "Fast Motion Estimation Method for MPEG-4 AVC," in ISO/IEC JTC1/SC29/WG11 MPEG2002/M8985, Shanghai, China, Oct. 2002.
- [5] Z.B. Chen, Y. He, "Fast Integer and Fractional Pel Motion estimation[3]", JVT-E045, 5th Meeting: Geneva, Switzerland, 9-17 October, 2002.
- [6] C. Zhu, X. Lin, L.P. Chau, K.P. Lim, H.A. Ang, C.Y. Ong, "A Novel Hexagon-based Search Algorithm For Fast Block Motion Estimation", IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings V.3, May 7-11, 2001, pp.1593-1596.
- [7] Ming-gang Liu and Chao-huan Hou, "A Fast Block-Matching Motion Estimation Algorithm Based on Spatial-Temporal Motion Vector Correlation." Proc. 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing, Hong Kong, May 2001.