
움직임 벡터의 효율적 예측을 이용한 고속 움직임 추정 기법

김종호

순천대학교 멀티미디어공학과

Fast Motion Estimation Technique using Efficient Prediction of Motion Vectors

Jongho Kim

Dept. of Multimedia Engineering, Sunchon National University

E-mail : jhkim@sunchon.ac.kr

요 약

비디오 부호화 기법에서 압축률 및 화질을 결정하는 중요한 부분인 움직임 예측 및 보상(motion estimation and compensation)의 성능을 개선하기 위한 방법을 제안한다. 가장 기본적인 움직임 예측 기법인 전역 탐색 방법은 가장 좋은 화질을 보이지만, 현재 프레임의 각 블록과 가장 유사한 블록을 찾기 위하여 탐색영역(search area)내의 모든 점에 대해 탐색을 수행하므로 그 계산량이 매우 많게 된다. 따라서 좋은 화질을 유지하면서 계산량을 낮추기 위한 많은 고속 알고리즘이 제안되었는데, MPEG-4 표준에 채택된 PMVFAST는 움직임 벡터 간의 상관도를 이용하여 계산량을 낮추면서도 전역 탐색 기법에 근접한 화질을 보인다. 본 논문에서는 움직임 벡터의 예측을 위하여 중간값(median) 계산법에 의한 새로운 방법을 제안하고, 이를 이용하여 움직임 예측의 계산량을 획기적으로 줄일 수 있음을 보인다. 실험결과 제안한 알고리즘은 PMVFAST보다 빠르면서도 전역 탐색 기법보다 높은 평균 PSNR을 보인다.

ABSTRACT

This paper proposes an enhanced motion estimation that is one of core parts affecting the coding performance and visual quality in video coding. Although the full search technique, which is the most basic method of the motion estimation, presents the best visual quality, its computational complexity is great, since the search procedures to find the best matched block with each block in the current frame are carried out for all points inside the search area. Thus, various fast algorithms to reduce the computational complexity and maintain good visual quality have been proposed. The PMVFAST adopted the MPEG-4 visual standard produces the visual quality near that by the full search technique with the reduced computational complexity. In this paper, we propose a new motion vector prediction method using median processing. The proposed method reduces the computational complexity for the motion estimation significantly. Experimental results show that the proposed algorithm is faster than the PMVFAST and better than the full search in terms of search speed and average PSNR, respectively.

키워드

움직임 예측 및 보상, 고속 움직임 예측, 움직임 벡터 예측, 미디언 프로세싱

1. 서 론

비디오 부호화 과정에서 시간적 중복성을 제거하기 위하여 움직임 예측 기법을 사용하는데, 대표적으로 블록 정합 알고리즘(block matching algorithm)을 들 수 있다. 각 프레임은 겹치지 않

는 여러 개의 매크로블록으로 분할되고, 참조 프레임에서 현재 매크로블록과 가장 유사한 블록을 찾아 그 위치한 움직임 벡터(motion vector)를 부호화하게 된다. 블록 간 유사도를 측정하는 기준은 여러 가지가 있는데, 그 중 가장 많이 사용되는 것은 SAD (Sum of Absolute Difference)이다.

현재 프레임을 $M \times N$ 크기의 블록으로 분할하였을 때 (x, y) 에 위치한 현재 매크로블록과 $(x+v_x, y+v_y)$ 에 위치한 블록 간의 SAD는 식 (1)과 같이 표현할 수 있다.

$$SAD(v_x, v_y) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} |F_t(x+m, y+n) - F_{t-1}(x+v_x+m, y+v_y+n)| \quad (1)$$

여기에서 F_t, F_{t-1} 은 각각 현재 프레임과 참조 프레임을 나타낸다. 식 (1)에서 탐색 범위가 $S_x \times S_y$ 일 때 전역 탐색 기법을 이용한다면 하나의 매크로블록당 $S_x \times S_y \times M \times N$ 번의 덧셈 연산이 필요하게 되는데, 이는 엄청난 계산량이다.

움직임 예측에 필요한 계산량을 줄이기 위하여 많은 고속 알고리즘들이 제안되었는데, 2D log 탐색, TSS (Three-Step Search), NTSS (New TSS), 다이아몬드 탐색 기법 등이 그것이다[1-2]. 이러한 알고리즘들은 탐색점의 개수를 줄임으로써 계산량은 줄였지만 화질이 열화되는 문제점이 있었다. MPEG-4 표준에 채택된 MVFAST (Motion Vector Field Adaptive Search Technique)와 PMVFAST (Predictive MVFAST)는 움직임 벡터 간의 상관관계를 이용하여 계산량 뿐만 아니라 화질의 열화를 감소시켰다[3-4]. 특히, 많은 알고리즘들이 $(0, 0)$ 벡터를 중심으로 탐색을 하는데 비해 PMVFAST는 주변 매크로블록의 움직임 벡터들의 중간값(median)을 중심으로 탐색을 실시하여 계산량을 줄이면서도 높은 화질을 보인다.

본 논문에서는 PMVFAST보다 더 정확하게 움직임 벡터를 예측하고 움직임 벡터 간의 상관성을 더욱 높이는 새로운 움직임 예측 기법을 제안한다. PMVFAST는 주변 블록의 움직임 벡터를 이용하여 탐색 시작점을 결정하는데 비해, 제안하는 알고리즘은 주변 블록뿐만 아니라 참조 프레임에서의 동일한 위치에 있는 매크로블록의 움직임 벡터도 이용하여 예측을 한다. 또한 시작점 예측이 정확하게 이루어지기 때문에 작은 탐색 패턴만을 이용하여 탐색점의 개수를 더욱 줄일 수 있고, 종료조건 등을 간소화할 수 있어 더 높은 PSNR을 보이게 된다.

II. 기존 알고리즘의 특징

고속 움직임 예측 알고리즘 중 성능이 비교적 좋은 다이아몬드 탐색기법은 통계적으로 많은 움직임 벡터가 $(0, 0)$ 주위에 분포한다는 점을 이용하여 초기 탐색점을 $(0, 0)$ 으로 결정하고, LDSP (Large Diamond Search Pattern)과 SDSP (Small Diamond Search Pattern)의 두 가지 탐색 패턴을 이용하여 반복적으로 최소 SAD를 찾는 방법을 사용한다. 다이아몬드 탐색 기법은 배경과 같이 움직임이 거의 없는 부분에서는 좋은 결과를 보

이는 반면, 움직임이 큰 부분에서는 초기 탐색점과 실제 움직임과의 오차로 인해 좋은 성능을 보이지 못한다. 이러한 단점을 보완한 알고리즘이 MVFAST와 PMVFAST인데, 이들은 현재 매크로블록의 움직임 벡터가 주변 매크로블록의 움직임 벡터와 상관성이 높다는 점을 이용하여 탐색 시작점을 적응적으로 결정하여 성능을 높인다.

특히 PMVFAST의 경우 주변 세 개의 예측 벡터($MV_{left}, MV_{top}, MV_{top-right}$)들의 중간값으로 탐색 시작점을 결정함으로써 정확하게 초기 탐색점을 예측하고, 이에 따라 움직임 벡터를 부호화하는데 필요한 비트수를 줄임으로써 화질을 높이고 있다.

III. 제안하는 고속 움직임 예측 알고리즘

제안하는 알고리즘은 PMVFAST에 새로운 초기 탐색점 결정 방법을 적용하여 속도와 화질 측면에서 뚜렷한 개선을 얻는다. 초기 탐색점을 정확하게 예측하면 되면 움직임 예측 속도를 향상시킬 수 있을 뿐만 아니라, 문턱값(threshold)과 초기 종료 조건을 줄일 수 있어 화질도 향상시킬 수 있기 때문이다.

1. 시간적 상관도를 포함한 중간값 예측

PMVFAST는 초기 탐색점을 결정하기 위해 세 개의 움직임 벡터를 사용한다. 초기 탐색점이 결정된 후에 영벡터와 MV_{t-1} 이 추가된다. 각 벡터의 위치는 그림 1에서 나타난 바와 같다.

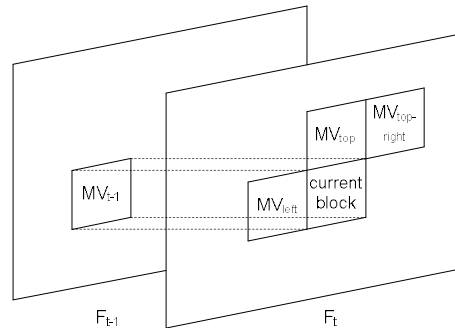


그림 1. 초기 탐색점을 결정하기 위한 예측벡터

표 1에 PMVFAST를 이용하여 부호화하였을 때 최종 움직임 벡터와 각 예측 벡터 간의 거리를 나타냈는데, Foreman을 제외한 모든 시퀀스에서 MV_{t-1} 이 최종 움직임 벡터와 가장 유사하다. 이는 MV_{t-1} 을 초기 탐색점 예측에 사용하면 정확한 예측을 할 수 있음을 뜻한다. 따라서 제안하는 알고리즘은 초기 탐색점 (d_x, d_y)를 다음과 같이 구한다.

$$\begin{cases} d_x = \frac{1}{2} \left[\sum_i MV_x^i - \max(MV_x^i) - \min(MV_x^i) \right] \\ d_y = \frac{1}{2} \left[\sum_i MV_y^i - \max(MV_y^i) - \min(MV_y^i) \right] \end{cases} \quad (2)$$

여기에서 $i \in \{\text{left}, \text{top}, \text{top-right}, \text{t-1}\}$ 를 의미한다. 즉, 네 개의 예측 벡터의 x, y 좌표 중 가장 큰 값과 가장 작은 값을 제외한 나머지 두 개의 평균값으로 초기 탐색점의 좌표 d_x, d_y 를 결정한다. 이는 평균 연산과 중간값 연산의 장점을 취하기 고안되었다. 프레임의 첫 번째 열의 매크로블록의 경우, MV_{left} 가 존재하지 않으므로 다음과 같이 기존의 중간값 예측을 그대로 적용한다.

$$\begin{cases} d_x = \text{median}(MV_x^{\text{top}}, MV_x^{\text{top-right}}, MV_x^{\text{t-1}}) \\ d_y = \text{median}(MV_y^{\text{top}}, MV_y^{\text{top-right}}, MV_y^{\text{t-1}}) \end{cases} \quad (3)$$

또한, 프레임의 첫 번째 행의 매크로블록의 경우 MV_{top} 과 $MV_{\text{top-right}}$ 가 존재하지 않으므로 영벡터를 추가하여 기존의 중간값 예측을 적용한다.

$$\begin{cases} d_x = \text{median}(MV_x^{\text{left}}, MV_x^{\text{t-1}}, 0) \\ d_y = \text{median}(MV_y^{\text{left}}, MV_y^{\text{t-1}}, 0) \end{cases} \quad (4)$$

프레임의 첫 번째 매크로블록, 즉 $(0, 0)$ 에 위치한 매크로블록의 경우에는 존재하는 예측 벡터가 $MV_{\text{t-1}}$ 뿐이므로 이것을 그대로 사용한다.

표 1. PMVFAST에서의 최종 움직임 벡터와 각 예측 벡터들간의 평균 거리

sequence	MV_{left}	MV_{top}	$MV_{\text{top-right}}$	$MV_{\text{t-1}}$
Bus	1.4481	1.6413	1.8814	0.8164
Coastguard	0.7329	0.6344	0.7378	0.5792
Foreman	1.3963	1.2969	1.6348	1.5396
Stefan	2.2522	2.5026	2.7951	2.0852
Table Tennis	0.7276	0.7172	0.8814	0.5720

초기 탐색점에서의 SAD가 충분히 작으면, 즉 문턱값 T_1 보다 작으면 더 이상 탐색을 하지 않고 초기 탐색점을 최종 움직임 벡터로 결정한다. 본 논문에서는 실험적으로 T_1 을 256으로 결정하였다. 표 1의 결과에서 볼 수 있는 바와 같이 $MV_{\text{t-1}}$ 과 최종 움직임 벡터는 상당히 유사하므로 탐색 시작점이 $MV_{\text{t-1}}$ 과 같다면 더 큰 문턱값을 사용할 수도 있다. 따라서 이 경우에는 다음과 같이 T_1 을 결정한다.

$$T_1 = \max(SAD_{t-1}, 256) \quad (5)$$

표 2는 PMVFAST와 제안한 알고리즘을 사용하였을 때 초기 탐색점이 최종 움직임 벡터로 선택되는 회수를 비교한 것이다. 제안한 알고리즘이 PMVFAST에 비해 훨씬 높은 수치를 보이는데, 두 알고리즘의 종료조건이 같으므로 제안하는 초기 탐색점 예측 방법이 더 정확함을 알 수 있다.

2. 예측 벡터로서 영벡터의 이용 방법

PMVFAST는 초기 탐색점에서 종료되지 않을 경우 5개의 예측 벡터를 더 사용하는데, $MV_{\text{left}}, MV_{\text{top}}, MV_{\text{top-right}}, MV_{\text{t-1}}$, 영벡터가 그것이다. 표 3은 영벡터를 제외한 나머지 네 개의 예측 벡터가 모두 영벡터가 아닐 때 최종적으로 영벡터가 선택되는 회수와 확률을 나타낸 것이다.

표 2. 초기 탐색점이 최종 움직임 벡터로 결정되는 회수 비교

Sequence	PMVFAST	Proposed
Bus	13542	15391
Coastguard	27941	32403
Foreman	24660	27205
Stefan	23572	24513
Table Tennis	46101	47066

표 3. 영벡터 이외의 벡터가 영벡터가 아닐 때 최종 움직임 벡터가 영벡터로 결정되는 확률

Sequence	회수	확률
Bus	1282	0.023
Coastguard	3472	0.031
Foreman	2828	0.026
Stefan	4540	0.041
Table Tennis	1178	0.011

표 3에서 예측 벡터들이 영벡터가 아닐 경우 영벡터가 선택될 확률은 극히 낮음을 알 수 있다. 이는 현재 매크로블록의 움직임 벡터가 영벡터라면, 대부분의 경우 예측 벡터 중 적어도 하나가 영벡터임을 의미한다. 즉 대부분의 경우 영벡터는 검사할 필요가 없음에도 검사하고 있다. 영벡터를 포함할 경우에는 더 낮은 SAD를 갖는 블록을 찾을 수 있지만, 영벡터를 제외할 경우에는 움직임 벡터 간의 상관성이 더 높아진다.

따라서 제안하는 알고리즘은 첫 번째 종료조건을 만족하지 않았을 경우 영벡터를 제외한 네 개의 예측벡터, 즉 $MV_{\text{left}}, MV_{\text{top}}, MV_{\text{top-right}}, MV_{\text{t-1}}$ 을 더 검사한다. 네 개의 예측 벡터를 검사한 후 최소의 SAD를 갖는 예측 벡터가 다이아몬드 탐색의 초기 탐색점이 된다. 다이아몬드 탐색의 초기 탐색점이 결정되면 문턱값 T_2 가 적용된다. 만약 다이아몬드 탐색의 초기 탐색점에서의 SAD가 T_2 보다 작으면 더 이상 검사를 수행하지 않고 종료되는데, T_2 는 다음과 같이 계산된다.

$$T_2 = \min(SAD_{\text{left}}, SAD_{\text{top}}, SAD_{\text{top-right}}) \quad (6)$$

T_2 가 너무 작은 값이거나 너무 큰 값으로 선택되지 않도록 하기 위하여 본 논문에서는 T_2 를 512와 1024 사이의 값으로 제한하였다. 초기 탐색점을 결정할 때와 마찬가지로 최종 움직임 벡터와 $MV_{\text{t-1}}$ 과의 상관성이 매우 크기 때문에 다이아몬드 탐색의 초기 탐색점과 $MV_{\text{t-1}}$ 이 같다면 T_2 를 다음과 같이 조정한다.

$$T'_2 = \max(SAD_{t-1}, T_2) \quad (7)$$

SAD가 이 문턱값보다 크면 SDSP를 사용하여 탐색을 계속 진행한다. 탐색점의 수를 줄이기 위하여 LDSP는 사용하지 않는다.

IV. 실험 결과

실험은 MPEG-4 VM 부호화기를 이용하여 수행하였다. 모든 영상은 IPPP 형태로 부호화되었고, 비트율 제어(rate control)은 TM5를 적용하였다. 실험에 사용된 영상은 QCIF (176×144), CIF (352×288), SIF (320×240)의 세 종류이며, 각 영상에서의 비트율과 프레임율, 탐색범위를 달리 적용하면서 실험을 하였다.

움직임 예측은 화질과 부호화 속도에 영향을 미치기 때문에 알고리즘의 성능은 PSNR과 속도의 측면에서 측정하였다. PSNR은 전체 프레임에 대한 평균 PSNR이고, 속도 향상비(Speed Up)는 전역 탐색 기법을 사용했을 때와 각 알고리즘을 사용했을 때의 탐색점 수의 비율을 의미한다. 표 4의 실험결과는 MVFAST와 PMVFAST의 성능이 다른 알고리즘에 비해 월등히 높으므로 다른 알고리즘과의 비교는 생략하고 이 두 가지 알고리즘과의 비교만을 제시한다.

표 4의 결과를 보면, 제안한 알고리즘은 모든 경우에서 MVFAST와 PMVFAST보다 빠른 것을 알 수 있다. 제안한 알고리즘은 MVFAST보다 평균 약 60% 빠르고, PMVFAST보다는 약 10%가 빠르다. 더욱이 평균 PSNR도 전역 탐색 기법보다 높은 것을 알 수 있다.

V. 결 론

본 논문에서는 기존의 알고리즘보다 정확한 초

기 탐색점을 예측함으로써 효율적인 부호화를 가능하게 하는 고속 움직임 예측 기법을 제안하였다. 제안된 알고리즘은 예측 움직임 벡터의 시공간적 상관성을 이용하여 탐색 시작점을 예측하기 때문에 예측이 매우 정확하고 속도와 화질 모두를 개선할 수 있다. 또한 제안하는 알고리즘은 PMVFAST와 같은 양의 메모리를 사용하면서도 더 빠르고 전역 탐색 기법보다도 높은 평균 PSNR을 보인다. 따라서 제안한 알고리즘은 PMVFAST와 같은 복잡도를 가지지만 화질과 속도 측면에서 더 우수한 알고리즘이라 할 수 있다.

<감사의 글>

본 논문은 지식경제부 및 정보통신연구진흥원의 지원을 받아 수행된 연구결과임 (09-기반. 산업원천기술개발사업)

참고문헌

- [1] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block matching motion estimation," IEEE Trans. Image Processing, vol. 9, no. 2, pp. 287-290, Feb. 2000.
- [2] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "A novel unrestricted center-biased diamond search algorithm for block motion estimation, IEEE Trans. Circuit and Syst. for Video Technol., vol. 8, no. 4, pp. 369-377, Aug. 1998.
- [3] P. I. Hosur and K. K. Ma, "Motion vector field adaptive fast motion estimation," Proc. Int. Conf. Information Comm. and Sig. Proc. (ICICS), Singapore, Dec. 1999.
- [4] A. M. Tourapis, O. C. Au, and M. L. Liou, "Predictive motion vector field adaptive

표 4. 제안한 알고리즘에 대한 실험 결과

Sequence	Format	Bit-rate (kbps)	Frame Rate	Search Area	Measure	Motion Estimation Algorithm			
						FS	MVFAST	PMVFAST	Proposed
Container	QCIF	10	7.5	16	PSNR(dB)	27.67	27.98	27.96	27.86
					Speed Up	1	218.96	285.33	305.59
				32	PSNR(dB)	27.89	27.90	27.85	27.90
					Speed Up	1	787.58	1053.63	1087.18
Foreman	CIF	112	10	16	PSNR(dB)	29.32	29.27	29.26	29.26
					Speed Up	1	84.52	103.33	105.71
				32	PSNR(dB)	29.53	29.43	29.45	29.46
					Speed Up	1	306.24	375.59	387.78
Table Tennis	SIF	1024	30	16	PSNR(dB)	31.77	31.73	31.75	31.75
					Speed Up	1	137.36	226.61	251.86
				32	PSNR(dB)	31.76	31.71	31.74	31.75
					Speed Up	1	516.55	848.29	934.35
				48	PSNR(dB)	31.77	31.69	31.73	31.74
					Speed Up	1	1093.70	1792.42	1977.80

search technique (PMVFAST) - Enhancing block based motion estimation," Proc. Visual Comm. and Image Proc. (VCIP), San Jose, CA., Jan. 2001.