

고속 움직임 추정을 위한 움직임 추정 생략 기법

정회원 강 현 수*, 박 성 모**

Motion Estimation Skipping Technique for Fast Motion Estimation

Hyun-Soo Kang*, Seong-Mo Park** *Regular Members*

요 약

본 논문은 계산량을 줄이는 움직임 추정 (motion estimation: ME) 방법을 제안한다. 이 방법은 움직임 추정이 필요없다고 판단되는 매크로블록(macro-block: MB)들에 대한 움직임 추정 과정을 생략함으로써 고속의 움직임 추정을 가능하게 한다. 그래서, 제안된 방법은 움직임 생략 기법 (ME skipping technique: MEST)이라고 부른다. 일반적으로 움직임 추정은 정수 화소 단위 움직임 추정 (IME)과 반화소 단위 움직임 추정 (HME)로 구성된다. MEST는 IME 과정 바로 직전에 수행되고, 이미 부호화된 주위의 MB들의 움직임 추정 오차에 기초한 기준에 따라 IME 과정을 생략할지를 판단한다. 하나의 MB에 대한 IME 과정이 생략되는 것으로 판단되었을 경우 (이 경우를 ME 생략 모드 (ME skip mode)라 부른다), IME 과정은 생략되고 그 MB에 대한 정수 화소 단위 움직임 벡터를 예측벡터로 대체한다. 예측된 그 벡터는 HME의 입력으로 사용된다. 한편, ME 생략 모드가 아닌 경우(ME non-skip mode)의 MB들에 대한 IME 과정은 생략되지 않고 보통의 IME과정을 수행하게 된다. 따라서 MEST는 ME 생략 모드로 판단되는 MB들이 많을수록 계산량을 감소시키는데 효과적이다. 뿐만 아니라 MEST가 비디오 부호화에 적용되었을 경우, 좀 더 정확한율제어 (rate control)과 채널 오류에 좀 더 강한 비트열 (bitstream)을 제작하는데 도움을 준다. 실험을 통해, MEST가 같은 화질을 유지하면서도 이러한 장점을 가짐을 보인다.

Key Words : motion estimation; MPEG-4.

ABSTRACT

The paper proposes a motion estimation (ME) technique to reduce computational complexity. It is achieved by skipping ME process for macro-blocks decided to be in no need of the operation. Thus, it is called ME skipping technique(MEST). In general, the ME is composed of integer pixel precision ME (IME) followed by half pixel precision ME (HME). The MEST is performed just before an IME process and makes a decision on skipping the IME process according to a criterion based on ME errors of adjacent macro-blocks (MBs) already encoded. When the IME process for a MB is decided to be skipped, which is called ME skip mode, the IME process is skipped and the integer pixel precision motion vector of the MB is just replaced by a predicted vector and used as the input of HME. On the other hands, the IME processes for MBs in ME non-skip mode are not skipped but normally performed. Accordingly, the MEST is very effective to reduce computational complexity when MBs in ME skip mode is abundant. In addition, when the MEST is applied to video encoder, it contributes to more accurate rate control and more robustness for channel errors. It is experimentally shown that the MEST has the above advantages while maintaining good reconstructed image quality.

* 중앙대학교 첨단영상대학원 영상공학과 (hskang@cau.ac.kr), ** 한국전자통신연구원 반도체·원천기술 연구소(smpark@etri.re.kr)

논문번호 : 030196-0509, 접수일자 : 2003년 5월 23일

※이 논문은 2002년도 한국학술진흥재단의 지원에 의하여 연구되었음 (KRF-2002-041-D00495)

I. 서론

움직임 추정 (motion estimation: ME)은 비디오 시퀀스의 시간 방향 잔여성분 (redundancy)를 줄이는 데 효과적이므로 H.261, H.263, MPEG-1, MPEG-2, MPEG-4 등의 비디오 압축 표준에서 뿐만 아니라 움직임 보상 부호화 기법이 적용된 비디오 부호화기에 널리 채용되고 있다. 그래서 좋은 복원 화질을 유지하면서, 좀 더 빠르고 정확한 움직임 벡터를 찾을 수 있는 방법이 요구된다. 전역탐색법 (full search algorithm)은 최적의 움직임 벡터를 찾는 반면 많은 계산량이 요구된다. 그래서 이러한 단점을 해결하기 위해, 많은 고속 알고리즘들이 제안되어 왔다. 예를 들어, 2-D logarithmic 탐색법, 3단계 탐색법, conjugate direct 탐색법, cross 탐색법, 4단계 탐색법, diamond 탐색법 등이 있다^{[1][2][3][4]}.

또한, 전역탐색법 자체에 대한 고속화 알고리즘 또한 활발히 연구되었는데, PDE(partial difference elimination algorithm), SEA (successive elimination algorithm), MSEA (multi-level SEA) 등이 대표적이다. PDE는 MPEG의 reference s/w에서 구현되어 있는 방식으로서, SAD (sum of absolute difference)의 계산 과정 중 이전의 최소 SAD를 초과하는 경우 SAD의 계산을 더 이상 수행하지 않는 방식이다^{[5][6][7]}. SEA는 블록의 평균값으로부터 최적 벡터가 될 수 있는지의 여부를 판단함으로써 전체 블록에 대한 SAD의 계산이 불필요한 블록에 대한 계산을 수행하지 않음으로써 계산량을 감축하는 방법이다^[8]. MSEA는 SEA의 다계층 접근방식으로 기본적인 아이디어는 동일하나 계산량을 크게 줄이는 방법이다^{[9][10]}. 여기서 언급된 PDE, SEA, MSEA는 모두 전역탐색법과 동일한 성능을 보이면서 계산량을 줄이는 고속 전역탐색법이라고 할 수 있다.

이러한 고속 알고리즘들과 같은 맥락으로, 본 논문에서 제안된 방법은 ME의 계산량을 감소시키는데 그 목적이 있다. 그러나, 기존의 고속 알고리즘은 탐색 방법에 대한 연구로서 ME 알고리즘 자체에 대한 계산량 감소에 그 목적이 있는 반면, 제안된 방법은 ME의 생략 여부를 판단하여 생략 가능할 경우 ME를 생략함으로써 ME에서 소요되는 계산량을 줄이는데 그 목적이 있다. 즉, MB들에 대한 ME 과정이 필요없다고 판단될 때, ME 과정을 생략함으로써 계산량을 감소시킬 수 있다. 그래서, 제안된 방법은 PDE, SEA, MSEA를 포함한 기존의

어떠한 고속 움직임 추정 알고리즘과도 같이 적용될 수 있다.

II. 기존의 움직임 추정 과정과 움직임 벡터 부호화

그림1은 일반적인 전체 ME 모듈 (module)을 보여주고 있다. 그림 1에서 보인 바와 같이, ME 모듈은 크게 IME 모듈과 HME 모듈로 구성되어 있다. IME 모듈은 정수 화소 단위 움직임 벡터를 찾는 역할을 담당하는 반면, HME 모듈은 반화소 단위 움직임 벡터를 찾는 역할을 담당한다. IME 과정은 ME 방식에 의존하지만 대체로 많은 탐색점 (search points)이 있음에 따라 많은 계산량을 요구하는 반면, HME 과정은 비록 반화소들에 대한 보간 (interpolation)과정을 필요로 하기는 하지만, IME 과정에서 얻어진 움직임 벡터 주위의 9개의 반화소들에 대한 탐색만을 수행한다. 따라서, IME 과정은 일반적으로 HME 과정에 비해 훨씬 많은 계산량을 요구한다. 그래서 본 논문에서는 HME 과정 보다는 IME 과정의 계산량을 줄이는데 초점을 맞춘다. 즉, IME 과정의 생략 가능 여부를 결정하고, 이로부터 계산량을 절감하는 방법을 제안한다.

한편, ME 모듈에서 얻어진 최종적인 움직임 벡터 (HME의 출력 벡터)는 VLC 모듈로 그대로 전달되는 것이 아니라 움직임 벡터 간의 상관성 제거를 위하여 움직임 벡터 예측기 (predictor)로 전달된다. 결국, 얻어진 움직임벡터에서 예측기 모듈의 결과인 예측벡터를 빼줌으로써 움직임 벡터 차 (motion vector difference: MVD) 성분을 얻게 되고 이것이 VLD 모듈에 의해 부호화된다. MPEG-4의 경우, 움직임 벡터의 예측 벡터는 현재 블록의 바로위쪽 블록의 벡터와 위쪽-우측 블록의 벡터와 바로좌측블록의 벡터의 중간값으로 정해지도록 되어 있다.

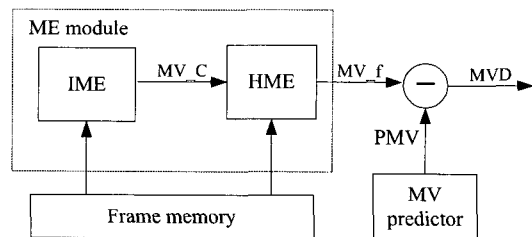


그림1 보통의 전체 ME 모듈

III. 제안된 방법

1 움직임 추정 방법

그림2는 다음 절에 자세히 기술될 MEST를 적용한 전체 ME 모듈을 설명하고 있다. 보통의 ME 모듈과는 달리 제안된 ME 모듈에는 ME 생략 판단 모듈이 새로이 첨가되어 있다. 그 ME 생략 판단 모듈은 IME 과정을 생략할지를 판단하는 역할을 수행할 뿐만 아니라 정수 화소 단위의 움직임 벡터를 예측하는 역할을 수행한다. 그림2에서 보는 바와 같이, 어떤 MB에 대한 IME 과정의 생략 여부에 대한 판단은 IME 과정 직전에 이루어진다. 그 판단 결과에 따라 현재 블록의 움직임 벡터 MV_c 는 MV_p 또는 MV_{act} 로 정해진다. 여기서 MV_p 는 예측된 움직임 벡터로서 IME 과정이 생략 가능하고 판단된 경우 (ME skip mode) MV_c 는 MV_p 가 되며, MV_{act} 는 IME 과정을 실제로 수행하여 얻은 결과 벡터로서 IME 과정을 생략하지 않아야 한다고 판단된 경우 (ME non-skip mode) MV_c 는 MV_{act} 가 된다. MV_c 가 결정된 후, MV_c 는 HME 모듈의 입력으로 사용된다. MV_p 와 ME 생략 방법의 자세한 내용은 다음 소절에서 설명된다. 앞으로 ME 생략 판단 모듈과 IME 모듈을 통합하여 제안된 IME 모듈 (PIME)이라고 부른다.

이제 ME skip mode로 판단된 MB들과 ME non-skip mode로 판단된 MB들에 대한 PIME 과정의 계산량을 조사해 보자. ME skip mode로 판단된 MB는 오직 ME 생략 판단 과정에 소요된 계산량만 필요하다. 반면, ME non-skip mode로 판단된 MB는 ME 생략 판단 과정 뿐만 아니라 IME 과정

이 필요하다. 여기서 주목할 점은 ME 생략 판단 과정은 IME 과정과 비교할 때 계산량에 있어서 극히 적다 (계산량의 적음은 다음 소절에서 생략 판단 과정을 자세히 기술할 때 설명될 것이다). 따라서 MV_c 를 얻는데 있어서, ME skip mode로 판단된 MB에 대해서는 거의 계산량을 필요로 하지 않는 반면, ME non-skip mode로 판단된 MB에 대해서는 IME 과정과 거의 같은 계산량을 유지한다. 즉, MV_c 를 얻는데 필요한 계산량의 감소량은 ME skip mode로 판단된 MB들의 개수에 정비례한다. 그래서, PIME 모듈의 계산량의 척도로서 총 MB의 개수와 ME skip mode로 판단된 MB의 개수의 비로서 생략율 (skip rate)를 다음과 같이 정의 한다.

$$r_{skip} = \frac{\#MBs \in ME \ skip \ mode}{\#total \ MBs} \quad (1)$$

예를 들어, $r_{skip} = 0.4$ 일 때, 총 MB의 40%의 MB들에 대한 IME 과정이 생략된다.

지금까지, 우리는 제안된 MEST에 대한 계산량에 대해 관심을 가진 반면, MEST에 의한 움직임 보상 오차를 고려하지 않았다. 즉, ME skip mode로 판단된 MB들에 대해 MV_p 를 사용함으로써 움직임 추정 오차가 증가할 것이라는 것을 쉽게 알 수 있다. 이 오차는 MEST의 성능과 밀접한 관계를 가지기 때문에 이 오차에 대한 논의는 MEST의 자세한 설명과 함께 이루어질 것이다.

2 ME 생략 기법 (MEST)

그림2에서 보여진 것처럼, MEST는 정수 화소 단위 움직임 벡터를 위한 IME 과정 직전에 수행되고 그 과정의 생략여부를 결정한다. H.263^[11]와 MPEG-4^[12]을 기준으로 MEST는 다음과 같이 수행

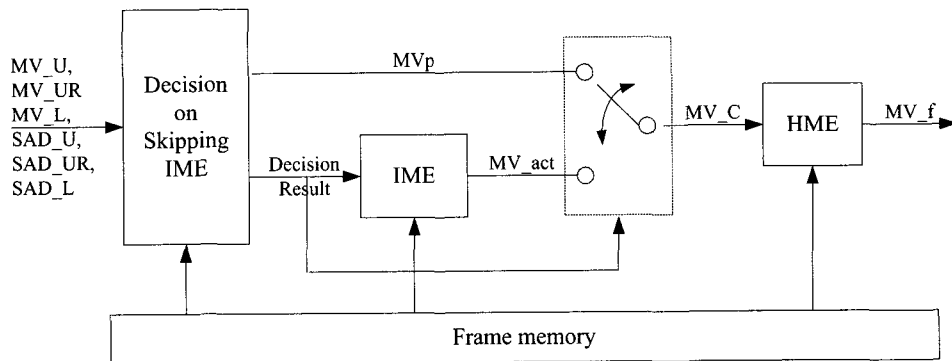


그림2 제안된 전체 ME 모듈

된다.

첫번째 단계: 부호화할 현재 MB, MV_c 의 예측 움직임 벡터 MV_p 의 각 방향 성분에 대해 다음과 같이 계산한다.

$$MV_p = \text{Median}\{[MV_L], [MV_U], [MV_{UR}]\} \quad (2)$$

여기서 MV_L , MV_U , MV_{UR} 은 각각 MB_c 의 왼쪽에 위치한 MB (MB_L)의 움직임 벡터, MB_c 의 위쪽에 위치한 MB (MB_U)의 움직임 벡터, MB_U 의 오른쪽에 위치한 MB (MB_{UR})의 움직임 벡터이다. 그리고, $\text{Median}\{a, b, c\}$ 은 a, b, c 의 중간값을 반환하는 함수이고 $[\cdot]$ 는 영에 가깝도록 정수형으로 절삭(truncation)하는 연산자이다. 예를 들어, $[3.5] = 3$, $[-3.5] = -3$. $[\cdot]$ 를 사용하는 이유는 두번째 단계에서 수행될 움직임 보상 시 연산을 간단히 하기 위함이다. 이 연산의 결과로서 MV_p 는 정수 화소 단위의 움직임 벡터이므로 MV_p 를 이용한 움직임 보상 과정에서 보간을 수행할 필요가 없게 된다.

두번째 단계: MV_p 를 이용하여 움직임보상(motion compensation: MC)을 수행함으로써 오차의 절대값의 합(sum of absolute error: SAD)의 향으로 MB_c 에 대한 보상 오차 SAD_c 를 얻는다.

세번째 단계: 만약 $SAD_c < SAD_{TH}$ 이면, IME 과정을 생략하고 MV_p 를 정수 화소 단위 움직임 벡터 MV_c 로 결정한다. 그렇지 않으면 IME 과정을 수행하여 얻은 움직임 벡터 MV_{act} 를 MV_c 로 결정한다. 여기서 SAD_{TH} 는 문턱값으로서 이 값에 대한 설명은 아래에서 자세히 기술된다.

우선 MV_p 에 대해 살펴보자. 표준 부호화기에서 MVD를 얻을 때 움직임 벡터 예측기(PMV)를 사용하는데, MV_p 는 식(2)에서 보는 바와 같이 PMV를 구할 때와 동일한 과정으로 구하도록 하였다. 이것은 ME skip mode로 결정되는 MB의 MVD를 부호화하는 데 적은 비트가 소요되도록 하기 위함이다. 여기서 주목해야 할 점은 PMV는 반화소 단위의 움직임 벡터인 반면, MV_p 는 정수 화소 단위 움직임 벡터인 점이다. 한편, ME를 수행했을 때의 ME 오차보다 MV_p 를 사용한 ME 오차가 크지는 것은 당연하다. 그러나 전체 부호화기의 관점에서 보면, ME를 생략함으로써 발생하는 ME 오차의 증가는 DCT 계수에 더 많은 비트를 할당함으로써 보상될 수 있다. 즉, MB skip mode의 경우 MV_p 를 움직임 벡터로 사용함으로써, PMV와의 차이가 +/-1 이내로 한정함에 따라, MVD의 부호화로 발생하는

비트량이 적음에 따라, DCT 계수에 MVD의 부호화에서 절감된 비트를 추가적으로 할당할 수 있으므로 움직임 추정 오차의 증가를 보상하게 된다(즉, 화질은 기존의 부호화기와 동일하게 유지된다). 다시 말해서, 부호화기에서는 블록 또는 프레임 단위로 목표비트량이 우선 결정되고, 이 목표비트량은 크게 MVD와 DCT 계수에 할당된다. 이러한 이유로, MVD에 비트할당량이 적어지게 되면(ME skip mode의 블록의 경우 MVD 비트가 적게 발생), DCT 계수에 비트할당량이 증가하여 부정확한 움직임 추정 오차를 보상해주게 된다. 이러한 형태의 비트 할당은 비트율을 일정하게 유지하려는 속성을 가진 율제어 알고리즘에 의해 자동적으로 이루어지게 된다. 이러한 ME 오차 증가에 대한 자동적인 보상으로 인해 ME 생략이 화질을 저하시키지 않게 된다. 이는 실험 결과를 통해 검증할 것이다.

이제 생략율에 지대한 영향을 미치는 SAD_{TH} 에 대해 살펴보자. SAD_{TH} 의 증가는 ME 생략으로 결정되는 MB의 수를 증가시킴을 주목해야 한다. 그래서, 큰 값의 SAD_{TH} 는 ME에서 소모되는 계산량을 감소시키는 반면, ME 오차를 증가시킴으로써 복원 화질의 저하를 초래할 수 있다. 그래서 우리는 SAD_{TH} 의 결정에 있어서 그 값을 가능한 크게 유지하면서 MC 오차가 크지 않도록 하여야 한다. 본문에서는, SAD_L , SAD_U , SAD_{UR} 은 각각 MV_L , MV_U , MV_{UR} 의 ME 오차들이라고 할 때, $SAD_{TH} = \text{Median}\{SAD_L, SAD_U, SAD_{UR}\}$ 로 결정하였다. 그 이유는, 급격한 움직임이 있는 영역을 제외했을 때, MB_c 에 대한 ME를 수행하여 얻은 ME 오차는 주변 MB들의 ME 오차보다 현저히 감소하지 않을 것이라는 가정에 기초하였다. 그래서 MV_p 를 이용하여 얻은 ME 오차가 주위 MB들의 ME 오차와 비교하여 수용할 수 있는 정도이면 ME를 수행하지 않음으로써 계산량을 절감하는 한편, MVD의 부호화에서 절감된 비트를 DCT 계수에 할당함으로써(앞에서 언급한 바와 같이 DCT 계수에 대한 비트의 추가 할당은 율제어 알고리즘의 속성에 의해 자동적으로 이루어진다), ME 오차를 보상하는 것이 전체 부호화기의 관점에서 유리하다. 즉, $SAD_c < SAD_{TH}$ 인 경우, MV_p 는 ME 오차의 측면에서 MV_{act} 와 비교하여 크게 나쁘지 않으며 MV_p 는 MV_{act} 에 비해 적은 부호화 비트를 필요로 하므로 DCT 계수에 더 많은 비트를 할당할 수 있으므로 ME 오차의 증가를 보상해주게 된다.

3 제안된 방법의 장점

MEST는 IME 과정에 있어서 다음과 같은 세가지 면에서 장점이 있다.

(1) 계산량 측면: 탐색영역이 $[-2n, +2n-1]$ 인 전역 탐색법은 $2n+12n+1$ 번의 SAD를 얻는 연산을 필요로 하는 반면, ME skip mode의 MB에 소요되는 계산은 오직 보간 과정이 없는 한번의 MC와 세번의 Median 연산 뿐이므로, 전역탐색법의 계산량과 비교했을 때 무시할 수 있을 정도이다. 그래서 MEST에 의한 계산량 이득은 생략율과 동일하다고 말할 수 있다. 예를 들어, $r_{skip} = 0.4$ 이면, 총 MB의 40%에 대한 IME 과정이 단지 Median 연산과 MC 연산으로 대체된다. 따라서, 하나의 탐색점에 대한 SAD 연산 정도만을 추가적으로 필요로한다. 또한, 전역탐색법이 아닌 다른 고속 움직임 탐색법과 비교하더라도 ME skip mode의 MB들에 대한 ME 연산의 복잡도는 큰 비중을 차지하지 못한다. 이것은 MEST가 전역탐색법 뿐만 아니라 다른 움직임 탐색법에도 유용함을 의미한다.

(2) 채널 오류에 대한 강건성의 측면: MEST를 적용한 부호화기에 의해 제작된 비트열은 채널 오류에 강점을 가질 수 있다. 앞에서 언급한 바와 같이 제안된 방법으로 MV를 찾을 경우, MVD부호화하는데 소요되는 비트가 적음에 따라, MPEG-4의 선택사항(option) 중의 하나인 data partitioning mode로 부호화하는 경우, motion marker 앞쪽에 위치한 MVD 데이터의 길이가 짧아지므로, 채널 오류로 인해 움직임 정보가 손상될 가능성이 작아지고, 오류 발생시 오류 은닉에 유리하다.

(3) 율제어 측면: 제안된 방법을 사용함으로써 율제어를 좀 더 정확히 수행할 수 있다. 율제어는 DCT 계수의 양자화 계수(Qp)를 조절함으로써 이루어진다. 그래서 제안된 방법이 적용되었을 때, 적용되지 않은 경우에 비해 전체 비트열에서 차지하는 DCT 계수의 데이터량이 상대적으로 많으므로 율제어가 더 용이하다. 이러한 정확한 율제어 동작은 부호화 생략 프레임(skipped frame)의 수를 줄이는데 도움을 준다. 이에 대한 검증은 다음 장에서 실험을 통해 이루어진다.

IV. 실험 결과

제안된 방식의 성능을 테스트하기 위해 QCIF 포맷의 Foreman, Coast Guard, Stefan 시퀀스에 대해

MPEG-4 부호화 환경 아래 실험하였다. 표1은 각 시퀀스에 대해 MPEG-4로 부호화 결과와 MPEG-4 부호화에 제안된 방식을 적용한 부호화 결과를 보여준다. 적용하지 않은 부호화 방법과 적용한 방법을 각각 OEM (ordinary encoding method), PEM (proposed encoding method)라고 부른다. 사용된 MPEG-4 부호화기로는 MoMuSys에서 배포한 부호화기가 사용되었고, 총 200 프레임, 프레임율 15Hz, 비트율 64kbps, 4MV off, 재동기 신호(resynchronization marker) off, $[-16,+15]$ 의 탐색범위를 가진 전역탐색법이 사용되었다. 표1에 나타난 실험 결과는 다음과 같이 요약된다.

(1) 복원화질 측면에서, PEM이 ME skip mode의 MB들이 많은 부분을 차지함에도 불구하고, OEM에 비해 뒤떨어지지 않는다. 오히려 Coast Guard의 경우 적은 비트로 비슷한 화질을 유지하고 있다. Foreman와 Stefan 시퀀스의 경우 PEM의 평균 PSNR이 조금 떨어지지만 이는 총 발생 비트량이 적기 때문이라고 생각된다. 즉, Foreman 시퀀스와 Stefan 시퀀스에 대해 각각 120비트와 324비트 적게 발생하였다.

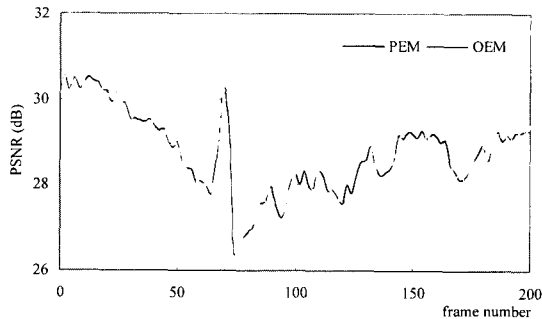
(2) 움직임 정보를 부호화하는데 소요된 비트량을 살펴 보면, 기대했던 바와 같이 PEM이 OEM에 비해 적은 비트를 소요한다. 이는 같은 목표비트량에 대해 DCT 계수에 더 상대적으로 많은 비트가 할당됨으로써 ME 오차의 증가분을 보상해준다. 또한, 이는 앞에서 설명한 바와 같이 채널 오류에 강한 장점이 있다.

(3) 계산량 측면에서 살펴보면, PEM은 중대한 이득을 가진다. 특히, Stefan 시퀀스의 경우 전체 MB의 40%가 ME Skip mode로 판단됨에 따라 계산량 또한 약 40%가 절감된다. 또한, PEM의 경우 부호화 생략 프레임의 수가 2장으로서 OEM의 경우 3장에 비해 적다. 이는 율제어 측면에서도 제안된 방법이 유리함을 말해 준다.

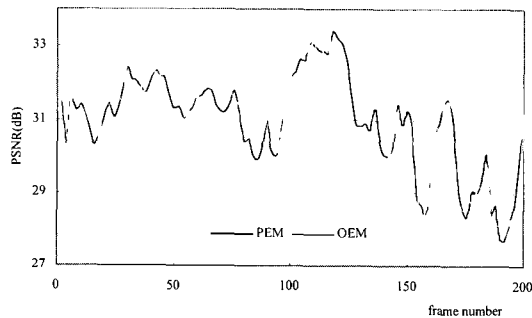
그림2와 그림3은 위의 세가지 시퀀스에 대한 자세한 결과를 보여준다. Stefan 시퀀스에 대한 그래프에서 시퀀스의 끝 부분에 불연속점이 있는 것은 부호화 생략 프레임이 존재하여 그 프레임에 대한 데이터가 유효하지 않기 때문이다. 이들 그림은 OEM과 PEM에 대한 PSNR은 프레임 단위로도 거의 같고, ME skip mode로 판단된 MB의 개수가 전체 시퀀스에 걸쳐 대체로 일정함을 보여준다.

제안된 기법은 어떤 MB의 움직임 벡터가 충분히 예측 가능하면, 그 MB에 대한 ME를 수행하는 것

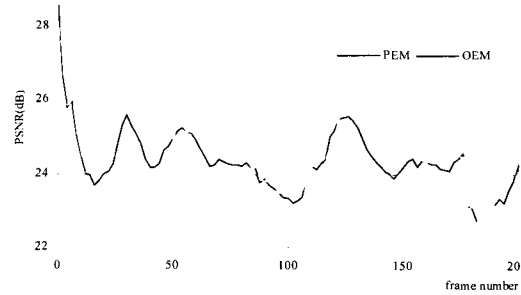
이 계산량 측면에서 효율적이지 못하다는 점에 착안하였다. 따라서 일관된 움직임을 가진 영상에 대해 움직임 벡터 예측 부호화가 효과적인 것과 마찬가지로 제안된 기법이 그러한 영상에 더욱 효과적일 수 있다. 실험 결과에서 보는 바와 같이 Foreman 시퀀스보다는 Coast Guard 시퀀스가 좀 더 일관된 움직임을 가지고 있으므로 더 효과적이다. 더욱이, 잡음과 같은 작은 물체들 (noise-like small objects) 이 빠르게 움직이는 Stefan 시퀀스와 같이, 프레임 간의 상관성이 적은 영상 시퀀스를 생각해 보자. 이 경우, 움직임 추정 방법이 프레임간 잔여성분을 감소시키는데 효과적이지 못하다. 그래서 이 경우에도 마찬가지로 ME 과정을 실제로 수행하는 대신 MEST를 적용함으로써 계산량 및 움직임 정보에 사용된 비트량을 감소시킬 수 있다. 이것이 빠른 움직임을 가진 시퀀스에 대해 생략율이 높은 이유이다. 다시 말해서, 어차피 움직임 추정을 하여도 제대로 된 움직임 벡터를 얻을 수 없을 경우에는 움직임 추정을 수행할 필요없이 제안된 방법을 적용함으로써 더 좋은 효과를 얻을 수 있음을 의미한다.



(a)



(b)



(c)

그림3 부호화 결과 (a) Coast Guard sequence (b) Foreman sequence (c) Stefan sequence

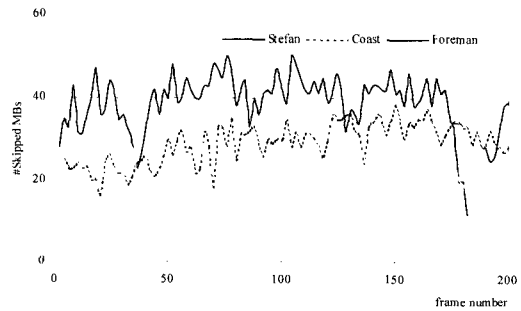


그림4 ME skip mode로 판정된 MB의 개수

표 1 PEM과 OEM의 부호화 결과

Image	Coast Guard		Foreman		Stefan	
	OEM	PEM	OEM	PEM	OEM	PEM
PSNR	28.74	28.74	31.02	30.94	24.48	24.38
#total bits	427896	427816	428016	427896	428376	428040
#MV bits	48794	44233	79273	73741	52436	48243
r_{skip}	-	0.293	-	0.211	-	0.400
#skipped frames	0	0	0	0	3	2

V. 결론

본 논문에서 움직임 추정이 필요없는 블록을 판단하는 알고리즘을 제안하였고, 이로 인해 얻어질 수 있는 장점들에 대해 기술하였다. 제안된 기법이 부호화에 적용될 때의 장점을 정리하면, 화질 저하를 일으키지 않으면서 일반적인 영상의 경우 적게는 20%에서 많게는 40%이상의 생략이 발생함에 따른 계산량의 절감, 일정 비트량의 제한 조건 아래

