

論文97-34S-9-11

블록 움직임 벡터의 검출을 위한 화소 간축 방법에 대한 연구 (Pixel Decimation for Block Motion Vector Estimation)

李 榮 * , 朴 貴 泰 **

(Young Lee and Gwi-Tae Park)

요 약

움직임 벡터의 검출을 위한 새로운 화소간축 (pixel decimation) 방법에 대하여 제안 하였다. 블록내에서 부표본화된 화소만을 사용하여 움직임을 검출하는 화소간축 방법은 연산량은 줄어들지만, 움직임 검출에 반영되지 않은 화소들로 인해 움직임 벡터 검출의 정확도가 저하된다. 본 논문에서는 화소의 부표본화에 따른 움직임 벡터의 오검출 가능성을 줄여줄 수 있는 부표본 화소의 선택 기준을 확률적인 관점에서 설정 하였다. 이렇게 설정된 기준을 바탕으로 기존 방법과 대등한 연산량을 지니면서 예측오차를 감소시킬 수 있는 화소간축 방법을 제안하였다. MPEG 표준 영상을 사용하여 시뮬레이션한 결과, 제안한 방법은 기존의 화소간축 방법에 비해 평균 절대오차 면에서 우수한 성능을 보임을 알 수 있었다.

Abstract

In this paper, a new pixel decimation algorithm for the estimation of motion vector is proposed. In traditional methods, the computational cost can be reduced since only part of the pixels are used for motion vector calculation. But these methods limits the accuracy of motion vector because of the same reason. We derive a selection criteria of subsampled pixels that can reduce the probability of false motion vector detection based on stochastic point of view. By using this criteria, a new pixel decimation algorithm that can reduce the prediction error with similar computational cost is presented. The simulation results applied to standard images have shown that the proposed algorithm has less mean absolute prediction error than conventional pixel decimation algorithm.

I. 서 론

시간축 상에서 연속한 동영상의 압축 방법으로는 프레임간 움직임 보상 부호화 (Interframe Motion Compensated Coding: IMCC) 가 매우 효율적이다. 따라서 MPEG, H.261 등^[1,2] 규격화가 완료된 기존

의 동영상 압축 표준들에서는 모두 IMCC를 압축의 근간으로 사용하고 있다. IMCC 는 프레임간에 존재하는 움직임 벡터를 검출하고 부호화 하는 부분과, 이 움직임 벡터를 사용하여 보상된 예측 영상과 원 영상과의 차이, 즉 예측 오차를 부호화 하는 부분으로 구성된다. 여기서 움직임 벡터를 정확히 검출하면 예측 오차를 줄일 수 있기 때문에, 움직임 벡터 검출의 정확도는 압축 효율에 큰 영향을 미친다.

* 正會員, 高麗大學校 電氣工學科

(Dept. of Electrical Eng. Korea Univ.)

** 正會員, 高麗大學校 電氣工學科, 서울대 ERC-ACI 研究 委員

(Dept. of Electrical Eng. Korea Univ., ERC-ACI Researcher, Seoul Univ.)

接受日字: 1997年4月3日, 수정완료일: 1997年8月5日

화상압축을 위한 움직임 벡터의 검출 방법은 크게 블록 단위의 정합을 통한 방법과 (Block Matching Algorithm: BMA), 화소 단위의 움직임을 검출 하는 방법 (pel recursive algorithm)으로 나눌 수 있는데, BMA의 경우가 계산량이 비교적 적고 하드웨어의 구현이 용이하므로, 기존의 압축 표준에서 많이 사용 되

고 있다. BMA 는 화상 압축에서 가장 많은 연산량을 필요로 하는 부분이며, 특히 전역 탐색 BMA (Full search BMA :FBMA) 는 화상 압축기 전체 연산량의 50% 이상을 차지할 정도로 막대한 연산량을 필요로 한다^[3]. 이러한 연산량의 문제를 해결 하고 BMA 의 실시간 구현을 가능토록 하기위한 많은 연구가 행해져 왔다.

먼저 탐색점의 수를 줄이므로써, 전체적인 BMA 의 연산량을 감축 시키는 방법으로 2-D logarithm search, three step search, conjugate search 등이 있다^[4,5,6]. 이 방법들은 전역 탐색 BMA에 비해 연산량의 대폭 감축이 가능 하지만, local minimum 의 위치를 블록의 움직임 벡터 로 취할 수 있고, 따라서 압축 효율이 저하된다는 문제점을 안고 있다. 또한 하드웨어의 구현성 면에서도 FBMA 에 비해 어려움이 많다.

탐색점의 수를 줄이지 않고 각 탐색점에서의 연산량을 줄이는 방법으로, Koga et al은 [5] 블록내에서 일정한 간격으로 부표본화(subsampling)된 화소값만을 사용하여 움직임 벡터를 구하는 화소간축(pixel decimation)방법을 제안 하였다. 또한 블록내에서 작은 크기의 부분블록(subblock)만을 사용하여 움직임 검출의 연산량을 줄이는 방법도 제안 되었다. 이러한 방법들은 일정한 위치에서 부표본화된 화소값만을 움직임 검출을 위한 연산에 사용하기 때문에 모든 화소들을 사용하는 FBMA에 비해 움직임 벡터 검출의 정확도가 떨어지며 예측 오차의 증가가 크다. Liu와 Zaccarin은^[7] 4:1 교번 부표본화(4:1 alternate subsampling) 알고리즘을 제안 하였는데, 여기서는 기준 블록과 탐색 영역내의 모든 화소가 움직임 벡터의 연산에 균등하게 반영될 수 있도록 일정한 교번 규칙을 정하고 이 규칙에 따라 부표본 화소를 선택하므로써 부표본화에 따른 예측 오차를 감소 시켰다. Liu 의 방법은 기존의 화소 간축 방법에 비해 복잡도는 크게 증가 시키지 않으면서 예측 오차의 개선이 크므로 화소 간축의 대표적인 방법으로 알려져 있다. 그러나 이 방법 역시 역시 연산에 반영되지 않은 화소들의 영향으로 인해 영상의 종류에 따라 성능의 편차가 심하다. 최근에 영상의 특성에 따라 적응적으로 부표본 화소를 선택하는 방법이 제안되었는데^[8], 기존의 방법들 보다 예측 오차를 개선시켰으나 알고리즘의 복잡성으로 인해 하드웨어적인 구현 측면에서 어려움이 있다.

본 연구는 기준 블록과 탐색 영역내의 부표본화된 화소값만을 사용하여 블록 단위의 평균 절대 오차를 최소화 하도록 움직임 벡터를 구하는 일반적인 화소 간축의 경우를 대상으로 한다. 먼저 기존의 연구 결과를^[9,10] 활용하여 평균 절대 오차의 확률 분포를 구하고 부표본화된 화소값과 평균 절대 오차의 확률 분포간의 관계 고찰을 통해, 움직임 벡터의 오검출 가능성이 작아지도록 부표본 화소를 선택 하는 방법에 대하여 다루었다. 이를 바탕으로 구현상의 복잡도를 증가 시키지 않으면서 기존의 화소간축 방법보다 움직임 벡터의 검출 정확도가 향상된 새로운 화소간축 방법을 제안하고 표준 영상을 사용한 시뮬레이션을 통해 성능을 비교 하였다.

II. 부표본 화소의 선택 방법

$M \times M$ 크기의 기준 블록 내에서 임의로 부표본화된 n 개의 화소값들을 $\{r_1, \dots, r_n\}$ 이라 하고, 탐색영역내 임의의 탐색점에서의 $\{r_1, \dots, r_n\}$ 에 대응되는 위치의 화소값들을 $\{s_1, \dots, s_n\}$ 이라고 하자. 블록 정합의 판단 기준으로 평균 절대 오차 D 를 사용하면,

$$D = \frac{1}{n} \sum_{i=1}^n |s_i - r_i|, \quad n \leq M^2 \quad (1)$$

탐색거리를 $\pm P$ 라고 할 때, 이 블록의 움직임 벡터는 $(2P+1)^2$ 크기의 탐색 영역내의 모든 위치에 대해 D 를 구하였을 때 이를 최소로 하는 탐색점의 위치를 말한다.

탐색 대상 블록이 $N[\mu_s, \sigma_s]$ 의 분포, 즉 기대값 μ_s , 표준편차 σ_s 의 정규 분포를 갖는 화소값들로 이루어져 있다고 가정하자. 그러면, 탐색 대상 블록내 임의의 위치에서 부표본화된 s_i 도 $N[\mu_s, \sigma_s]$ 의 분포를 갖는 랜덤 변수로 생각할 수 있다. 참고 문헌 [9,10]에서는 이러한 분포 특성을 갖는 부표본화된 화소를 사용한 경우의 평균 절대 오차의 분포를 아래의 식(2)-식(5)를 통해 유도 하였다. 본 논문에서는 동일한 유도식을 채용하되 이의 해석 및 활용에 있어서는 다른 접근을 시도 하였다.

기준 블록에서 부표본화된 임의의 화소값 r_i 와 랜덤 변수 s_i 의 차이값인 $s_i - r_i$ 는 $N[\mu_s - r_i, \sigma_s]$ 의 분포를 갖는 랜덤 변수이다. $|s_i - r_i|$ 의 분포를 구하기

위해 $\mu = \mu_s - r_i$ 이고 $\sigma = \sigma_s$ 라고 하면, 참고문헌 [11]에 의해

$$\mu_{|s_i - r_i|} = \sigma \sqrt{\frac{2}{\pi}} e^{-\frac{\mu^2}{2\sigma^2}} + 2\mu G\left(\frac{\mu}{\sigma}\right) - \mu \quad (2)$$

$$\sigma_{|s_i - r_i|}^2 = \sigma^2 + \mu^2 - \mu_{|s_i - r_i|}^2 \quad (3)$$

여기서 G 는 Gaussian 연산자를 나타낸다. 식(1)로 주어지는 평균 절대 오차 D 의 확률 분포를 구하기 위해 $|s_i - r_i|$ 이 모든 $i (i=1, \dots, n)$ 에 대하여 상호 독립적이라고 가정한다. 그러면 central limit 정리에 의해, 부표본화 화소의 수 n 이 증가함에 따라 $|s_i - r_i|$ 의 합은 식(4)와(5)를 만족하는 정규 분포 특성을 갖게된다고 볼 수 있다.

$$\mu_D = \frac{1}{n} \sum_{i=1}^n \mu_{|s_i - r_i|} \quad (4)$$

$$\sigma_D = \frac{1}{n} \left(\sum_{i=1}^n \sigma_{|s_i - r_i|}^2 \right)^{\frac{1}{2}} \quad (5)$$

이상의 과정으로 부터 부표본화된 화소를 사용할 때의 평균 절대 오차는 $N[\mu_D, \sigma_D]$ 의 분포를 갖는다고 생각할 수 있다.

[9,10]에서는 식(4)와(5)의 결과를 coarse to fine의 2 단계 블록 정합시 연산 cost를 최소로 하는 부표본 화소의 개수 n 을 결정하는데 사용하였다. 즉, 부표본화된 화소만을 사용하여 식(1)의 coarse 정합을 수행해가면서, D 의 값이 임계치 t 보다 작아지는 경우에는 블록내의 모든 화소를 사용하여 Fine 정합을 행하는 2 단계 블록 정합의 경우에 있어서, 연산 cost는 $C + pF$ 로 표현할 수 있다. 여기에서 C 는 coarse 정합의 cost로서 n 에 반비례 하고, F 는 fine 정합의 cost, p 는 D 의 값이 임계치가 t 이하가 될 확률이다. t 가 주어지면 p 는 식(4)와 (5)에 의해 계산될 수 있다. 이로부터, 임계치 t 가 주어진 경우 연산 Cost $C + pF$ 를 최소로 하는 부표본 화소의 개수 n 을 구할 수 있다.

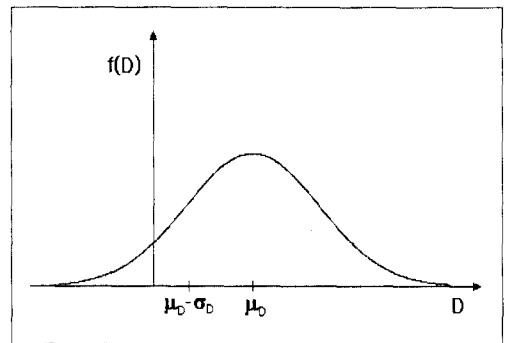
본 연구에서는 기존의 화소 간측 방법과 같이^[5,7] n 이 정해진 경우에 있어서, 움직임 벡터의 오검출 확률이 작아지도록 부표본 화소를 선택하는 방법에 대하여 다루었다. $N[\mu_D, \sigma_D]$ 의 분포특성을 갖는 D 의 확률 밀도 함수를 $f(D)$ 라고 하면,

$$f(D) = \frac{1}{\sigma_D \sqrt{2\pi}} e^{-\frac{(D - \mu_D)^2}{2\sigma_D^2}} \quad (6)$$

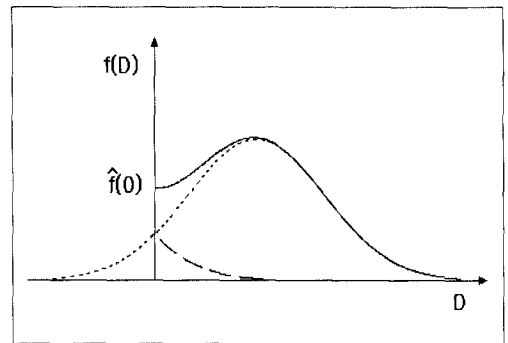
으로 나타낼 수 있다. 임의의 μ_D 와 σ_D 에 대하여 $f(D)$ 의 모양을 그림 1(a)에 보였다. 여기에서 D 는 평균 절대 오차이므로 음의 값을 가질 수 없다. 따라서 실질적인 D 의 모양은 그림 1(b)의 모양을 갖게된다. 여기에서 D 가 최소값을 갖게될 확률은 D 가 0이 될 확률로서 이값을 $\hat{f}(0)$ 라고하면,

$$\hat{f}(0) = \frac{2}{\sigma_D \sqrt{2\pi}} e^{-\frac{\mu_D^2}{2\sigma_D^2}} \quad (7)$$

의 값을 갖게된다.



(a)



(b)

그림 1. D 의 확률 밀도 함수

Fig. 1. Probability density function of D .

$\hat{f}(0)$ 가 클수록 최소값을 갖는 탐색점의 수가 많고 볼 수 있으며 따라서 잘못된 움직임 벡터를 검출할 확률이 높아진다고 생각할 수 있다. 이로부터, 기준 블록내의 부표본 화소 r_i 의 선정 기준으로서 움직임 벡터의 오검출 확률이 작아지도록 하기 위해서는, $\hat{f}(0)$ 의 크기가 작아지도록 r_i 가 선정되어야 한다는 것을 알 수 있다. 그림 2에 $\mu (= \mu_s - r_i)$ 와 $\hat{f}(0)$ 간의

관계를 보였다. μ 가 0 일 때 $\hat{f}(0)$ 는 최대값을 갖는다. 즉, 기준 블록내 임의의 위치에서 부표본화된 r_i 가 탐색 대상 블록내 같은 위치에 있는 화소의 기대값 μ_s 와 같은 값을 가질 때 잘못된 움직임 벡터를 검출하게될 확률이 가장 높다. $|\mu|$ 가 큰 값을 가질수록 $\hat{f}(0)$ 는 작은 값을 갖는다. 다시말해, 부표본 화소값 r_i 와 μ_s 의 차이가 클수록 움직임 벡터를 정확히 검출할 가능성이 높다고 생각할 수 있다. 본 논문의 경우에는 기준 블록과 탐색 대상 블록내의 모든 화소값을 정확히 알 수 있으므로 $\mu_s = s_i$ 라고 생각할 수 있다. 따라서 움직임 벡터의 검출 정확도를 높이기 위해서는 탐색 대상 블록내 화소값과 가장 큰 절대 차이를 보이는 기준 블록의 화소를 부표본 화소로 선택해야함을 알 수 있다. $|\mu|$ 가 큰 값을 가질수록 $\hat{f}(0)$ 는 작은 값을 갖는다. 다시말해, 부표본 화소값과 탐색 대상 블록내의 평균 화소값과의 차이가 클수록 움직임 벡터를 정확히 검출할 가능성이 높다고 생각할 수 있다. 이상으로 부터, 탐색 대상 블록의 평균 화소값과 가장 큰 절대 차이를 보이는 화소를 부표본 화소로 선택해야함을 알 수 있다. 3장에서는 이러한 해석의 결과를 이용한 새로운 화소 간축 방법에 대하여 설명한다.

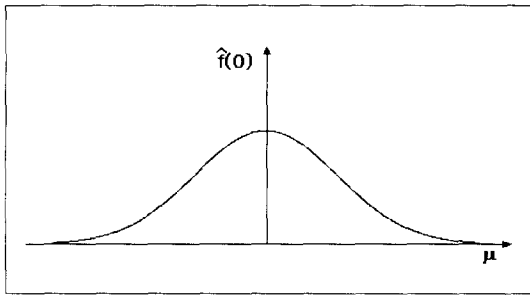


그림 2. μ 와 $\hat{f}(0)$ 간의 관계
Fig. 2. Relationship between μ and $\hat{f}(0)$.

III. 제안된 화소간축 알고리즘

본 연구에서는 $M \times M$ 크기의 기준 블록을 $b \times b$ 크기의 부블록으로 나누고 각 부블록마다 1개의 부표본 화소를 선택하는 일반적인 화소 간축 방법을 대상으로 한다. 이렇게 하므로써 기준 블록내의 전 영역에 걸쳐서 균등하게 부표본 화소를 선택하는 효과를 가질 수 있으며, 하드웨어의 구현성 면에서도 용이해진다. 이때

부표본 화소의 수 $n = \frac{M^2}{b^2}$ 이다.

기존의 화소 간축 방법에서는 일정한 위치에서 부표본화된 n 개의 화소만을 사용하고 있고, Liu의 방법에서도 정해진 교번 규칙에 따라 부표본화된 n 개의 화소를 사용하여 움직임 벡터 연산을 수행하고 있다. 부블록내 화소들이 유사한 값을 가질 경우 이러한 부표본 화소만을 사용하여도 큰 오차없이 움직임 벡터를 검출할 수 있게 된다. 그러나 부블록내 화소값들의 변화가 클 경우 어떤 위치의 화소가 부표본 화소로 선택 되었느냐에 따라 평균 절대 오차값이 크게 달라질 수 있으며, 움직임 연산의 결과도 달라지게 된다. 움직임 벡터 연산의 결과를 좌우하는 것은 배경 부분 같이 비교적 균일한 분포를 하고 있는 화소들이 아니라, 물체의 경계 부분 같이 화소값의 변화가 큰 부분이라는 것을 감안할 때, 화상의 특성에 관계 없이 부표본 화소를 선택하는 기존의 화소간축 방법들로는 움직임 벡터 검출의 정확도에 한계가 있다. 2장의 해석을 근거로 각 부블록을 대표 하는 부표본 화소를 선택하므로써 움직임 벡터 검출의 정확도를 높일 수 있는 화소 간축 방법에 대하여 설명하기로 한다.

화소 간축 방법 A.

연산량의 문제를 고려하지 않고 2장의 해석 결과를 그대로 채용하여 타당성을 살펴 보기 위한 방법으로서는, 각 부블록내에서 탐색 대상 화소값과 가장 큰 절대 차이를 보이는 화소를 해당 부블록의 부표본 화소로 선택한다. 그림 3에 탐색 위치가 (dx, dy) 인 경우의 부표본 화소의 선택 예를 보였다.

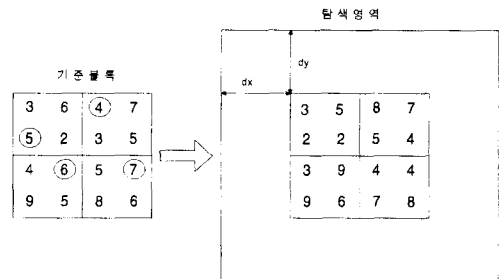


그림 3. 방법A의 부표본 화소의 선택예
Fig. 3. Subsampled pixels of method A.

선택된 부표본 화소만을 사용하여, 탐색 영역내의 모든 탐색위치에 대하여 식(1)의 D 를 구하였을 때

D를 최소로 하는 탐색 위치가 이 블록의 움직임 벡터를 나타낸다. 4장의 시뮬레이션 결과를 보면(표1) 제안한 방법(화소간축A)에 의해 Liu의 화소 간축 방법^[7] 보다 0.25-1.2 dB 정도 예측 성능이 개선되었다. 따라서 2장의 해석을 근거로한 제안된 화소 간축 방법은 기존의 방법보다 움직임 벡터의 검출 정확도를 향상 시킨다는 것을 알 수 있다. 해석 결과에 의존하지 않더라도, 제안한 방법은 부블록내의 모든 화소들 중에서 절대 오차 연산의 결과에 가장 큰 영향을 미치는 화소를 부표본 화소로 선택 하게 되므로 기존의 화소 간축 방법 보다 연산의 정확도가 향상 된다는 것을 알 수 있다. 그러나, 이 방법은 모든 화소에 대하여 가장 큰 절대 차이를 갖는 화소를 구하기 위한 연산을 수행해야하고, 각각의 탐색점 마다 이러한 연산이 반복적으로 수행되어야 하므로, FBMA와 거의 대등한 연산량을 필요로 한다. 연산량의 증가에 반해 성능은 FBMA대비 0.01-0.1 dB 정도 저하 되므로 효율성면에서 문제가 있다.

제안된 화소 간축 방법 B.

방법A가 갖는 연산량의 문제를 해결 하면서 2장의 해석 결과를 최대한 활용하기 위한 화소 간축 방법으로, 그림 4에 부표본 화소의 선택 방법을 도시 하였다.

기 준 블 록

max	min	max	min
min	max	min	max
max	min	max	min
min	max	min	max

그림 4. 부표본 화소의 선택 방법
Fig. 4. Rule for subsampled pixel selection.

탐색을 시작하기 전에 기준 블록의 각 부블록에 대하여 최대값 또는 최소값을 갖는 화소 위치를 찾는다. 그림 4에 나타난 것 처럼 한 부블록에 대해 최대값을 갖는 화소 위치를 구했다면, 이 부블록에 수평 수직 방향으로 이웃한 부블록에서는 최소값을 갖는 화소 위치를 구한다. 이렇게 최대 최소값을 갖는 화소 위치를

교번 시켜가며 구한후 이 화소들을 기준 블록의 부표본 화소로 선택하고, 각 탐색점에서 부표본 화소의 위치에 대응되는 탐색 대상 블록의 화소들을 탐색 대상 블록의 부표본 화소로 선택한다. 그림 5에 부표본 화소의 선택 예를 보였다.

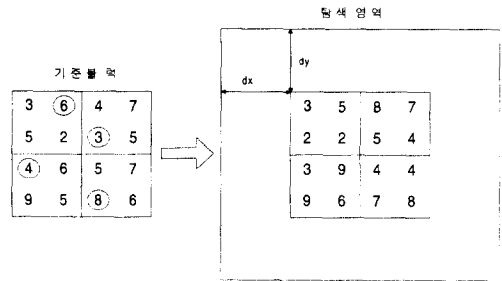


그림 5. 방법B의 부표본 화소의 선택 예
Fig. 5. Subsampled pixels of method B.

방법 A에서 부표본 화소를 선택하기 위해, 기준 부블록과 탐색 부블록간에 가장 큰 절대 차이를 보이는 화소 위치를 직접 연산에 의해 구했던 것에 반해, 방법B에서는 기준 부블록내의 최대 혹은 최소값을 갖는 화소를 부표본 화소로 선택한다. 탐색 부블록의 크기는 2x2로 비교적 작으므로 부블록내의 화소들이 유사한 값을 갖는다고 가정할 수 있다. 그러면 방법 A의 연산은 기준 부블록내의 화소들에서 최대값이나 최소값중의 하나를 선택 하는 것으로 대체될 수 있다. 탐색 부블록내 화소들이 유사한 값을 갖지 않을 경우에도, 기준 부블록내에서 최대 또는 최소값을 갖는 화소가 탐색 부블록내의 화소와 가장 큰 절대 차이를 보이게 될 가능성이 높다. 여기서 최대값을 갖는 화소를 택할 것인지 최소값을 갖는 화소를 택할 것인지를 정확히 결정하기 위해서는 기준 부블록내 화소들과 탐색 부블록내 화소들간의 크기 비교를 위한 연산이 필요하게 되는데 방법 B에서는 연산량을 최대한 줄이기 위해서 이러한 비교 연산 과정을 생략하기로 한다. 영상의 특성에 따라 부표본 화소로 선택되는 최대 화소 및 최소 화소의 개수와 선택 위치가 달라져야 하지만, 모든 종류의 영상에 대하여 동일한 선택 규칙을 적용시키기 위해서 그림 4에서와 같이 최대 최소 화소의 선택 개수를 같게 하였으며, 각각의 선택 위치가 블록 전체에 균일하게 분포하도록 하였다.

이상의 부표본 화소 선택을 위한 연산은 탐색을 시

작하기 전에 기준 블록만을 사용하여, 기준 블록내의 각 부분 블록마다 최대값 또는 최소값을 갖는 화소를 찾아내는 비교 연산을 통해 수행되며, 방법A와 달리 각 탐색점마다 이러한 비교 연산을 다시 수행할 필요는 없다. 기존의 화소 간축 방법과 비교할 때 한 블록당 $\frac{(b^2-1) \times M^2}{b^2}$ 회의 비교 연산만이 추가되는 정도이며, $M=16$, $b=2$ 이고 탐색 거리가 ± 15 인 경우에 대하여 비교하면, 기존의 화소 간축 방법이 122047회의 덧셈셈과 61504회의 절대값 연산을 필요로 하는데 반해, 방법B에서는 192회의 비교 연산이 추가된다. 덧셈셈 연산과 절대값 연산 및 비교 연산이 모두 대등한 연산 부하를 갖는다고 생각하면 방법B의 사용에 따른 연산량의 증가는 0.01% 정도이다.

IV. 시뮬레이션 결과

제안한 화소 간축 방법의 성능을 평가하기 위해 4종의 대표적인 시험 영상을 사용하여 시뮬레이션을 수행하였다. 시험영상으로는 짝수 필드만을 취한 704×240 크기의 Football과 Flower garden 영상 및, 352×288 크기의 Salesman과 Miss america 영상을 사용한다. 각각 30 프레임의 영상에 대해 성능의 차이를 PSNR 측면에서 비교 하였다. Football 및 Flower garden은 16×16 블록과 최대 15 화소의 탐색 영역을 사용 하였고, Salesman 및 Miss america는 16×16 블록과 최대 7 화소의 탐색 영역을 사용 하였다. MV의 검출 방법으로는 본 연구에서 제안한 방법A와 방법B의 화소간축 방법을 사용 하였고, 비교대상으로는 기존의 전역 탐색과 Liu의 화소간축 방법을 사용 하였다.

부분블록의 크기가 2×2 인 경우에 대하여 각각의 영상을 사용하여 실험한 결과를 표1에 보였다.

표 1. 움직임 검출 알고리즘별 평균 PSNR
Table 1. Average PSNR of motion estimation algorithms.

Algorithm	FBMA	Liu's	방법A	방법B
Image				
Football	23.77	23.43	23.68	23.61
Flower garden	25.92	25.60	25.86	25.80
Salesman	35.06	33.86	35.05	34.86
Miss america	37.84	37.25	37.74	37.51

Liu 의 화소간축 방법에 비해 방법A는 영상에 따라 0.25-1.2 dB의 PSNR의 개선을 보이고 있다. 그러나 방법A는 Liu의 방법 보다 약 4배의 연산량을 필요로 하며 전역 탐색과 거의 대등한 연산량이 소요 되는데 반해 성능은 0.01-0.1 dB 정도 저하되므로 사용 효과가 없다. 방법B는 Liu의 방법에 비해 대등한 연산량으로 0.18-1.0 dB의 성능 개선을 보인다. 전역 탐색과 비교하면 약 1/4의 연산량으로 0.12-0.33 dB의 성능 저하를 보이고 있다. 한가지 특이한 사실은 Liu의 화소 간축 방법이 영상의 종류에 따라 매우 큰 성능의 차이를 보인다는 점이다. Salesman 영상의 경우 전역 탐색에 비해 1.2 dB의 큰 성능 저하가 있다. 이 경우에도 제안한 방법B는 0.2 dB 정도의 성능 저하만을 보인다. 이로 부터 영상의 특성을 고려치 않고 일정한 위치에서 부표본 화소를 선택하는 기존의 화소 간축 방법으로는 움직임 벡터 검출의 정확성에 한계가 있음을 알 수 있다.

본 논문의 방법은 영상의 종류에 관계없이 기존의 전역 탐색에 비해 PSNR의 저하가 크지않다. 이는 영상의 특성에 따라 부분블록을 가장 잘 대표하는 화소를 부표본 화소로 선택하기 때문이며, 연산량을 줄이기 위해 이러한 선택 과정을 간략화 시킨 방법 B의 경우에도 큰 성능 저하를 보이지 않음을 알 수 있었다. 그림 6에서 그림 9까지 각 영상들의 프레임에 따른 PSNR의 변화를 보였다. 프레임에 관계없이 방법A와 방법B의 성능이 비교적 잘 유지되고 있음을 알 수 있다.

표 2. 8 이웃 전역 탐색시의 평균 PSNR
Table 2. Average PSNR of FBMA for 8 connected neighborhood.

Algorithm	FBMA	방법A	방법B
Image			
Football	23.77	23.70	23.69
Flower garden	25.92	25.92	25.88
Salesman	35.06	35.06	35.04
Miss america	37.84	37.75	37.63

화소 간축으로 인한 성능의 저하를 개선 시키기 위해, 화소 간축을 통해 구해진 움직임 벡터와 이를 중심으로 8개의 이웃한 탐색점에 대해 전역 탐색을 수행 하여 보았다. 표2에 그 결과를 보였다. 방법B에서 전역 탐색에 비해 0.02-0.21 dB 정도로 약간 뒤지는 성

능을 보이고 있다. 이에 따른 연산의 추가는 탐색 거리가 ± 15 인 경우 0.9% 정도이고 ± 7 인 경우 4%정도로 미미하므로 실시간 적용에 문제가 없다.

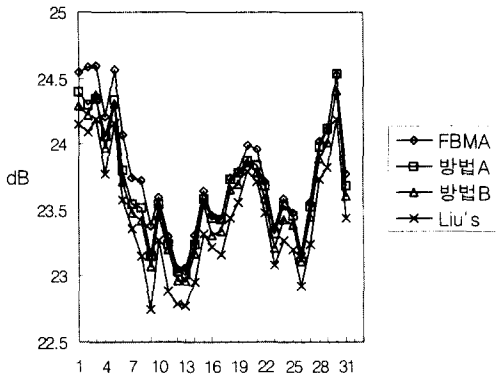


그림 6. Football 영상의 PSNR
Fig. 6. PSNR of Football.

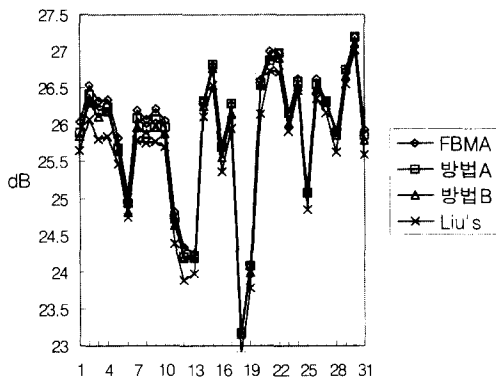


그림 7. Flower garden 영상의 PSNR
Fig. 7. PSNR of Flower garden.

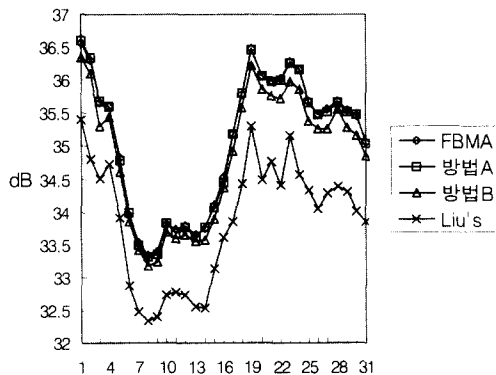


그림 8. Salesman 영상의 PSNR
Fig. 8. PSNR of Salesman.

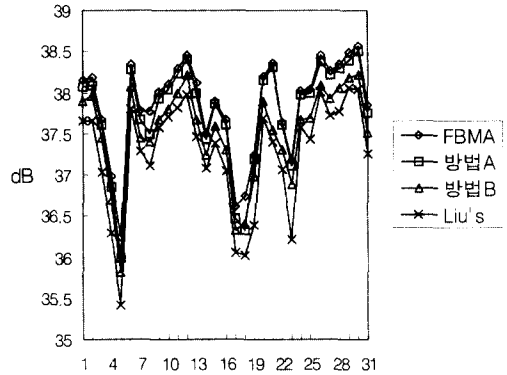


그림 9. Miss america 영상의 PSNR
Fig. 9. PSNR of Miss america.

V. 결 론

본 논문에서는 움직임 벡터의 연산 시간을 줄이기 위한 화소 간축 방법에 대하여 제안 하였다. 화소 간축 방법으로는 각 블록을 작은 크기의 부분블록으로 나누고, 각 부분블록 마다 1개의 부표본 화소를 선택하는 가장 일반적인 경우를 대상으로 하였다.

먼저, 화소의 부표본화에 따르는 움직임 벡터의 오검출 가능성을 줄이기 위해, 부표본 화소값이 평균 절대 오차의 확률 분포에 미치는 영향을 분석하고 이를 토대로 영상의 특성에 따라 적응적으로 부표본 화소를 선택하는 기준을 제시하였다. 이렇게 설정된 선택 기준을 최대한 만족 하면서 기존의 화소 간축 방법과 대등한 연산량을 사용하는 새로운 화소 간축 방법을 제안 하였다. 제안한 방법에 대한 시뮬레이션 결과 기존의 화소 간축 방법에 비해 움직임 벡터 검출의 정확도가 향상되고 영상의 종류에 관계없이 비교적 균일한 성능을 유지한다는 것을 알 수 있었다.

앞으로의 연구 과제는 제안된 알고리즘을 구현하기 위한 단순한 형태의 하드웨어를 설계하는 것이다.

참 고 문 헌

- [1] "Generic Coding of moving pictures and associated audio," Committee Draft of Recommendation H.262, ISO/IEC 13818-2, Nov. 1993.
- [2] "Draft revision of recommendation H.261:

- Video codec for audio visual services at p*64 kbits/sec," *Image Commun.*, vol.2, no.2, pp.221-239, 1990.
- [3] H.Fujiwara, M. Liou, M. Sun, K. Yang, "An all-ASIC implementation of a low bitrate video codec," *IEEE Trans. Circuits Syst. Video Technol.*, vol.2, no.2, pp.123-133, Jun 1992.
- [4] J.R.Jain and A.K.Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. COM-29, pp.1799-1808, Dec. 1981.
- [5] T.Koga and K. Iinuma, "Motion compensated interframe coding for video conferencing," in *Proc. NTC81*, pp.C9.6.1-9.6.5, New Orleans,LA, Nov./Dec. 1981.
- [6] M. Ghanbari, "The Cross-Search Algorithm for Motion Estimation," *IEEE Trans. Commun.*, vol.38, no.7, pp.950-953, 1990.
- [7] B.Liu and A.Zaccarin, "New fast algorithms for the estimation of block motion vectors," *IEEE Trans. Circuits Syst. Video Technol.*, vol.3, no.2, Apr.1993.
- [8] Y.L.Chan and W.C.Siu, "New Adaptive Pixel Decimation for Block Motion Vector Estimation", *IEEE Trans. Circuits Syst. Video Technol.*, vol.6, no.1, Feb.1996.
- [9] G.J.Vanderbrug and A.Rosenfeld, "Two Stage Template Matching", *IEEE Trans. Computers*, vol. C-26, no.4, Apr.1977.
- [10] A.Rosenfeld and G.J.Vanderbrug, "Coarse Fine Template Matching", *IEEE Trans. Syst. Man and Cyber.*, Feb.1977.
- [11] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. New York McGraw-Hill.

 저 자 소 개

李 榮(正會員) 第 34卷 S編 第 2號 參照

朴 貴 泰(正會員) 第 34卷 S編 第 2號 參照