

움직임 영역간 움직임 보상오차의 최소편차를 이용한 최적 블록정합 움직임 추정

조 영 창[†] · 이 태 홍^{††}

요 약

일반적으로 블록기반 움직임 추정에서 움직임 경계에 위치한 블록 내에는 서로 다른 움직임을 갖는 여러 움직임 영역이 공존한다. 이 때, 블록 내의 움직임 보상오차는 각 움직임 영역에 따라 다르게 나타난다. 이는 기존의 평균절대오차와 같이 블록 전체에 대해 누적된 정합오차들 사용할 경우, 그릇된 움직임을 추정하고, 블록 전체의 움직임 보상에서는 시간적으로 화질을 저하시키는 결과를 초래할 수 있다. 본 논문에서는 이러한 문제점을 해결하기 위해 시공간적으로 인접한 블록들의 움직임 정보를 이용하여 블록을 움직임에 따라 영역별로 나누고, 각 영역별 평균 움직임 보상오차를 구한 후, 영역간 보상오차의 최소편차를 고려하는 새로운 정합함수를 정의함으로써 최적의 움직임 벡터를 추정하기 위한 방법을 제안한다. 모의실험을 실시하여 기존의 평균절대오차를 사용한 전역 탐색법(full search : FS)과 윤곽기반 블록 정합법(edge oriented block matching algorithm)에 대한 추정결과를 서로 비교하여 제안한 방법의 우수성을 보였으며, 특히, 움직임 보상결과 움직임 경계에서의 화질향상을 얻을 수 있었다.

Optimal Block Matching Motion Estimation Using the Minimal Deviation of Motion Compensation Error Between Moving Regions

Young Chang Cho[†] · Tai Hong Yi^{††}

ABSTRACT

In general, several moving regions with different motions coexist in a block located on motion boundaries in the block-based motion estimation. In this case the motion compensation errors(MCEs) are different with the moving regions. This is inclined to deteriorate the quality of motion compensated images because of the inaccurate motions estimated from the conventional mean absolute error(MAE) based matching function in which the matching error per pixel is accumulated throughout the block. In this paper, we divided a block into the regions according to their motions using the motion information of the spatio-temporally neighboring blocks and calculated the average MCE for each moving region. And we proposed the new matching function considering the minimal deviation of MCEs between the moving regions to solve the problems mentioned. From the simulation results, we showed the improved performance of the proposed method by comparing the results from other methods such as the full search method and the edge oriented block matching algorithm. Especially, we improved the quality of the motion compensated images of blocks on motion boundaries.

1. 서 론

디지털 비디오 신호전송에서 프레임간의 중복성을 줄이기 위한 움직임 추정법으로 블록 정합 움직임 추정은 알고리즘의 단순성과 고속처리를 위한 하드웨어적인 집계의 용이성으로 인해 MPEG 계열[1, 2]이나 H.261[3], H.263[4] 등의 표준안에서 채택되어 많이 사용되고 있다. 블록 정합법은 이전 프레임의 탐색영역 내에서 현재 블록과의 정합오

차가 가장 작은 블록의 위치를 찾는다. 많이 사용되어 온 정합함수로는 평균절대오차(mean absolute error : MAE), 상호 상관함수(cross correlation function : CCF), 평균자승오차(MSE), 최소화된 최대 오차함수(minimized maximum error function : MME)[5] 등이 있다. 이들 중, 평균절대오차는 계산량이 적으며, 비교적 양호한 정합기준이 됨으로 인해 가장 많이 사용되고 있다. 다음 식은 움직임 벡터 MV에 대한 평균절대오차 MAE(·)를 나타낸다.

$$MAE(MV) = \frac{1}{|S_B|} \sum_{s \in S_B} |I_t(s) - I_{t-1}(s + MV)| \quad (1)$$

$$MV = (u, v), \quad s = (i, j)$$

[†] 정 회 원 : 경운대학교 컴퓨터전자정보공학부 교수
^{††} 정 회 원 : 영남대학교 전기전자공학부 교수
논문접수 : 2001년 4월 6일, 심사완료 : 2001년 9월 13일

위 식에서 (u, v) 는 움직임 벡터의 수평, 수직성분을 나타내며, (i, j) 는 화소의 수평, 수직 위치성분을 나타낸다. I_t 와 I_{t-1} 은 t 번째와 $t-1$ 번째 프레임을 나타내며, S_B 와 $|S_B|$ 는 현재 블록내의 전체 (i, j) 화소집합과 집합의 크기를 나타낸다. 식 (1)에 대해 블록의 움직임 벡터 MV_B 는 다음 식과 같이 결정된다.

$$MV_B = \min_{MV} [MAE(MV); \quad (2)$$

$$MV = (u, v), -L \leq u \leq L, -L \leq v \leq L]$$

식 (2)에서 L 은 각 움직임 벡터성분의 최대값을 나타낸다. 이와 같이 기존의 정합함수들은 블록 내의 영역별 보상오차 분포보다 블록 전체에 대한 평균오차의 크기만을 이용하기 때문에 여러 움직임 영역을 갖는 블록에 있어서는 잘못된 움직임을 추정하게 되고, 이로 인해 블록화 현상과 같은 결함들이 움직임 보상에 나타날 수 있어 움직임 보상영상의 화질이 크게 저하될 수 있다. 블록 정합법이 갖는 이러한 문제점을 해결하기 위해 다양한 방법들이 제안되었다. 그 중, Anandan은 움직임 벡터의 정확성을 높이기 위해 다중 해상도의 계층적 블록 정합법을 제안하여 정합오차에 대한 잡음의 영향을 줄이고자 했으며[6], 특히, Auyeung는 블록화 현상을 줄이기 위해 중첩된 블록의 사용을 제안하였다[7]. 또한, M.H. Chan은 블록화 현상의 감소 및 움직임 경계에서의 보다 정확한 움직임 벡터를 추정하기 위해 블록을 여러 개의 소블록으로 나눈 후, 소블록별로 움직임을 추정하는 가변 블록 정합법을 제안하였다[8]. 그러나, 가변 블록 정합법은 현재의 표준안에서 사용되고 있는 블록 정합법과는 구조적으로 상충되기 때문에 적용하는데 어려움이 따른다. 이러한 방법들은 움직임 보상에서 다소 블록화 현상을 줄이는데 기여는 하지만, 여러 움직임을 갖는 블록에 대해 기존의 정합함수가 갖는 문제점은 크게 개선시키지 못한다. 최근, Y. L. Chan은 평균절대오차 대신에 영상의 공간적인 윤곽정보를 이용함으로써 움직임 경계에서 시각적으로 개선된 움직임 보상영상을 얻는 방법을 제안하였다[9]. 그러나, 이 방법은 물체의 윤곽이 흐려지거나 움직임이 다른 윤곽이 블록 내에 공존할 경우, 잘못된 움직임을 추정할 수 있다. 본 논문에서는 움직임 경계에서 움직임 벡터의 정확성을 개선하고 움직임 보상시의 블록화 현상을 줄이기 위해 시공간적으로 인접한 이웃블록의 움직임 정보를 이용하여 블록 내 화소들을 움직임에 따라 영역별로 분할하고 [10], 각 영역에 대한 평균 움직임 보상오차를 구한 후, 영역간 보상오차의 최소편차를 고려하는 새로운 정합함수를 정의함으로써 최적의 움직임 벡터를 추정하기 위한 방법을 제안한다. 본 논문은 II 장에서 블록 내의 서로 다른 움직임 영역에 대한 정합오차의 영향을 설명하고, III 장에서는 윤곽기반 정합에 대해 설명하며, IV 장에서는 제안한 방법을, V 장에서는 제안한 방법의 성능분석을 위한 모의실험결과를, 마지막으로 VI 장의 결론으로 구성된다.

2. 움직임 영역별 정합오차

블록 내에 여러 움직임 영역이 존재하는 경우, 식 (1)의 평균절대오차는 각 움직임 영역별로 표현될 수 있다. 블록 내에 서로 다른 움직임을 갖는 K 개의 영역을 각각 R_1, R_2, \dots, R_K 라 정의하고, 각 영역의 실제 움직임 벡터를 각각 MV_1, MV_2, \dots, MV_K 라 두면, 각 영역이 프레임 주기동안 형태적으로 변하지 않는다고 가정할 때, 이 블록에 대한 움직임 보상오차는 다음과 같다.

$$MAE(MV) = \frac{1}{|R_t|} \sum_{k=1}^K \sum_{s \in R_k} \quad (3)$$

$$| I_{t-1}(s + MV_k) - I_{t-1}(s + MV) |$$

$$R_t = R_1 \cup R_2 \cup \dots \cup R_K \quad (4)$$

식 (4)에서 R_t 는 전체 블록영역을 나타내며, $|R_t|$ 는 R_t 의 크기를 나타낸다. 블록의 움직임 벡터를 MV_B 라 하고, 블록 내의 같은 움직임을 갖는 영역을 R_B 라 할 때, 블록 내의 움직임 영역들은 식 (5)와 같이 MV_B 에 대해 정합영역 R_{match} 과 비정합영역 $R_{nomatch}$ 으로 나눌 수 있다.

$$R_{match} = R_B$$

$$R_{nomatch} = R_t - R_B = R_1 \cup R_2 \cup \dots \cup R_{B-1} \cup R_{B+1} \cup \dots \cup R_K \quad (5)$$

따라서, 블록의 움직임 벡터는 블록 내의 정합영역과 비정합영역에 대해 보상오차를 최소로 하는 벡터로 결정된다. 식 (6)은 블록의 움직임 벡터를 MV_B 라 하고, 블록 내의 같은 움직임을 갖는 영역을 $R_B (= R_{match})$ 라 할때, MV_B 가 블록의 움직임 벡터가 되기 위한 조건을 나타낸다.

$$\sum_{k=1}^K \sum_{\substack{s \in R_k \\ R_k \subset R_{nomatch}}} | I_{t-1}(s + MV_k) - I_{t-1}(s + MV_B) |$$

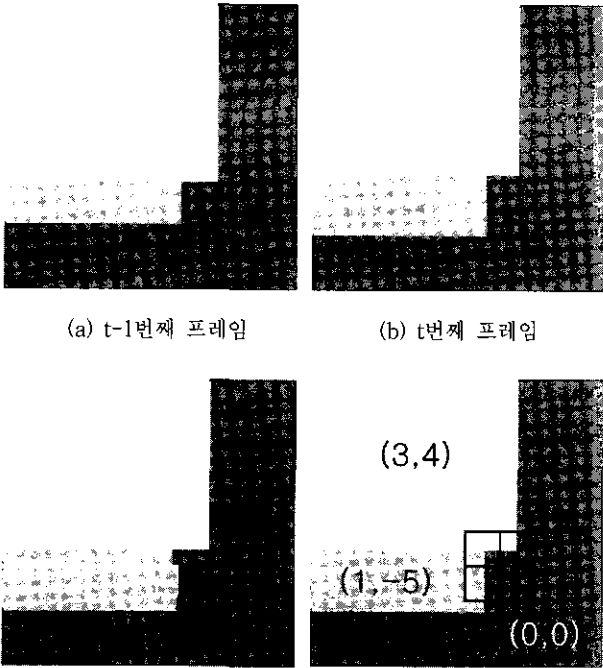
$$+ \sum_{k=1}^K \sum_{\substack{s \in R_k \\ R_k \subset R_{match}}} | I_{t-1}(s + MV_k) - I_{t-1}(s + MV_B) |$$

$$< \sum_{k=1}^K \sum_{s \in R_k} | I_{t-1}(s + MV_k) - I_{t-1}(s + MV) |, \quad (6)$$

$$\text{where } MV_B = (u_B, v_B), \quad MV = (u, v) \neq (u_B, v_B),$$

$$-L \leq u \leq L, -L \leq v \leq L.$$

여기서, 식 (6)을 만족시키는 MV_B 가 비정합영역 $R_{nomatch}$ 에서 큰 보상오차를 가질 경우, 비정합영역에서는 움직임 보상에 왜곡이 발생된다. (그림 1)은 움직임이 다른 세 개의 물체영역에 대한 움직임 추정 예를 보여주기 위해 본 논문에서 사용한 256×256 '모의영상'(그림 3)의 일부를 나타낸 것으로써 (a)는 $t-1$ 번째 프레임, (b)는 t 번째 프레임, (c)는 전역 탐색법에 의해 추정된 움직임 벡터로부터 구한 움직임 보상영상, (d)는 각 물체영역에 대한 움직임 벡터와 네 개의 16×16 블록을 보여준다.



(a) t-1번째 프레임 (b) t번째 프레임

(c) (b)에 대한 움직임보상영상 (d) 물체영역의 움직임 벡터

(그림 1) 움직임 경계에 위치한 블록의 움직임 보상액극

(그림 1)에서 (d)에 표시된 좌, 상단 블록 내에는 세 개의 움직임 영역이 존재한다. 이 블록에 대해 평균절대오차를 사용해서 추정된 블록의 움직임 벡터 (3, 4)는 식 (6)을 만족하지만, (c)에서 처럼 다른 움직임 영역인 물체②나 배경에 대해서 왜곡을 일으킨다. 또한, 좌, 하단 블록에 대해서도 추정된 움직임 벡터 (0, 0)에 대해 물체② 영역의 움직임 보상에 왜곡이 나타난다. 이와 같이 평균절대오차와 같은 기존의 정합함수들은 식 (6)의 조건만을 평가하기 때문에 왜곡을 줄이기 위해서는 추정된 움직임 벡터에 대한 영역별 움직임 보상오차에 대한 고려가 필요하다.

3. 윤곽기반 블록 정합법

윤곽기반 블록 정합법은 평균절대오차 대신, 윤곽간의 정합오차를 이용하는 방법으로써 움직임 경계에서 시각적으로 양호한 움직임 보상영상을 얻기 위해 제안되었다. 식 (7)은 Sobel 연산자를 나타내며, 프레임 내의 윤곽 추출을 위해 사용된다.

$$g_x = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, g_y = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (7)$$

그리고, 식 (8)은 윤곽끼리의 정합을 위해 t번째와 t-1번째 프레임내의 윤곽만 추출한 윤곽영상을 나타낸다.

$$I_t^{edge}(s) = |\Gamma_t(s) * g_x| + |\Gamma_t(s) * g_y| \quad (8)$$

$$I_{t-1}^{edge}(s) = |\Gamma_{t-1}(s) * g_x| + |\Gamma_{t-1}(s) * g_y|$$

식 (8)에서 Γ_t 와 Γ_{t-1} 은 평균치 필터를 사용하여 필터링된

I_t 와 I_{t-1} 를 각각 나타내며, 이는 윤곽추출에 대한 잡음의 영향을 줄이기 위한 것이다. 여기서, *는 컨벌루션 연산을 나타낸다. 그러므로, 식 (8)에 대한 움직임 벡터 MV의 윤곽기반 정합오차 $EA(\cdot)$ 는 식 (9)와 같이 정의되며, 이에 대한 블록의 움직임 벡터는 식 (10)을 만족하는 벡터로 결정된다.

$$EM(MV) = \frac{1}{|S_B|} \sum_{s \in S_B} |I_t^{edge}(s) - I_{t-1}^{edge}(s + MV)| \quad (9)$$

$$MV_B = \min_{MV} [EM(MV); MV = (u, v), -L \leq u \leq L, -L \leq v \leq L] \quad (10)$$

이 정합법은 간단하면서 비교적 움직임 경계에서의 보상 영상에 대한 왜곡을 감소시키는데 있어 효과적이다. 그러나, 식 (8)에 의해 추출된 윤곽이 흐려진 경우, 블록화 현상이 심화되는 문제점이 있다.

4. 제안한 알고리즘

본 논문에서는 블록 내 화소들을 시공간적으로 인접한 이웃 블록의 움직임 벡터에 대한 움직임 보상오차를 이용하여 움직임 영역별로 분할하고, 분할된 영역간 평균 보상오차의 편차를 이용함으로써 움직임 경계에 대해 최적의 움직임 벡터를 추정하는 방법을 제안한다.

4.1 움직임 영역의 결정

블록 내의 움직임 영역 중에서 이웃블록의 움직임과 같은 영역은 이웃블록의 움직임 벡터에 대해 작은 움직임 보상오차를 갖는다. 본 논문에서는 이웃블록의 움직임 벡터에 대한 보상오차를 이용하여 블록 내의 화소들 움직임 영역별로 분할한다.

시공간적으로 인접한 N개의 이웃블록에 대한 움직임 벡터를 각각 $MV_B^{(1)}, MV_B^{(2)}, \dots, MV_B^{(N)}$ 라 할 때, n번째 이웃블록의 움직임 벡터에 대해 16×16 블록 내의 각 화소 $s (s \in S_B)$ 에 대한 움직임 보상오차는 다음과 같다.

$$MCE^{(n)}(s) = |I_t(s) - I_{t-1}(s + MV_B^{(n)})| \quad (11)$$

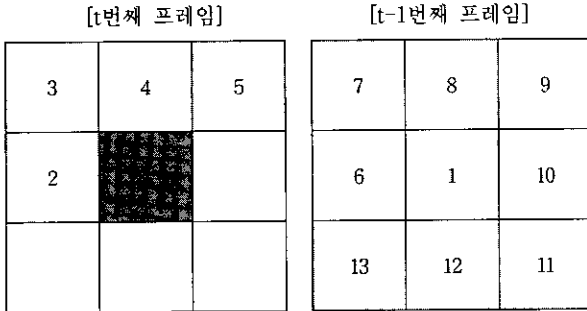
만약, 식 (11)에서 s가 n번째 이웃블록과 같은 움직임을 갖는 블록 내의 영역에 속하면, 식 (11)의 움직임 보상오차는 작은 값을 갖는다. 식 (12)는 n번째 이웃블록과 같은 움직임을 갖는 블록 내의 영역 $R_{match}^{(n)}$ 을 정의한 식이다.

$$R_{match}^{(n)} = \{s : MCE^{(n)}(s) < T, s \in S_B\} \quad (12)$$

$$R_{nomatch}^{(n)} = R_t - R_{match}^{(n)}$$

식 (12)에서 T는 블록 내의 밝기분포로부터 미리 계산된 움직임 보상오차의 허용치를 나타낸다. 또한, 식 (11)과 식 (12)에 대해 모든 이웃블록들을 적용함으로써 블록 내의 다

른 움직임 영역들에 대해서도 정합영역과 비정합영역을 결정할 수 있다. 그러므로, 식 (12)로부터 결정된 N개의 정합영역 $R_{match}^{(n)}$ 과 비정합영역 $R_{nomatch}^{(n)}$ 은 추정된 움직임 벡터에 대해 식 (6)에서 비정합영역의 보상오차가 왜곡에 미치는 영향을 감소시키기 위해 사용된다. 본 논문에서 사용된 이웃블록(N=13)은 (그림 2)와 같으며, 이는 움직임 벡터의 시간공간적인 상관성과 인과적인 순서를 고려하여 정하였다.



(그림 2) 이웃블록의 움직임 벡터 참조순서(■: 현재 블록)

4.2 제안한 정합함수

식 (6)에서와 같이 움직임 보상에 블록화 현상은 잘못 추정된 블록의 움직임 벡터 MV_B 에 대해 식 (12)의 $R_{nomatch}^{(n)}$ 영역에 대한 오차가 클 경우에 현저히 나타난다. 그러므로, 본 논문에서는 $R_{match}^{(n)}$ 와 $R_{nomatch}^{(n)}$ 영역에서의 평균 보상오차를 구하고, 영역간의 편차를 고려하는 새로운 정합함수를 제안함으로써 $R_{nomatch}^{(n)}$ 영역의 보상오차에 대한 영향을 줄이고자 한다.

본 논문에서 제안한 움직임 벡터 MV 에 대한 새로운 정합함수(new motion compensation error: NMCE) $NMCE(\cdot)$ 는 다음과 같이 평균 절대오차량과 움직임 영역내의 평균 움직임 보상오차에 대한 영역간 편차를 고려하는 항으로 정의된다.

$$NMCE(MV) = MAE(MV) + \lambda D_{max}(MV) \quad (13)$$

식 (13)에서 우변의 $D_{max}(MV)$ 는 영역간 움직임 보상오차의 최대 편차로써 식 (14)와 같이 정의한다.

$$D_{max}(MV) = \max_D [D^{(n)}(MV), 1 \leq n \leq N] \quad (14)$$

$$D^{(n)}(MV) = | \mu_{match}^{(n)}(MV) - \mu_{nomatch}^{(n)}(MV) | \quad (15)$$

식 (15)에서 $\mu_{match}^{(n)}(MV)$ 와 $\mu_{nomatch}^{(n)}(MV)$ 는 각각 n번째 이웃블록의 움직임 벡터로부터 얻어진 움직임 영역에서 정합영역과 비정합영역에 대한 평균 움직임보상오차를 나타내며, 다음 식 (16), 식 (17)과 같이 주어진다.

$$\mu_{match}^{(n)}(MV) = \frac{1}{|R_{match}^{(n)}|} \sum_{s \in R_{match}^{(n)}} |I_t(s) - I_{t-1}(s + MV)| \quad (16)$$

$$\mu_{nomatch}^{(n)}(MV) = \frac{1}{|R_{nomatch}^{(n)}|} \sum_{s \in R_{nomatch}^{(n)}} |I_t(s) - I_{t-1}(s + MV)| \quad (17)$$

식 (13)에서 λ 는 밝기에 대한 움직임 보상오차량과 평균 움직임보상오차의 영역간 편차량에 대한 비중을 조정하기 위한 파라미터이다. 그러므로, 제안한 정합함수에 대한 블록의 움직임 벡터 MV_B 는 다음 식과 같이 결정된다.

$$MV_B = \min_{MV} [NMCE(MV); MV = (u, v); -L \leq u \leq L, -L \leq v \leq L] \quad (18)$$

본 논문에서 제안한 정합함수는 블록 내의 각 움직임 영역들에 대한 평균 보상오차를 고려하여 평균절대오차를 최소화하는 동시에 움직임 영역간의 편차를 최소화하는 움직임 벡터를 블록의 움직임 벡터로 결정함으로써 움직임 경계에서 추정된 움직임 벡터의 정확성을 높이고, 나아가 움직임 보상영상의 화질을 개선시키는데 기여한다.

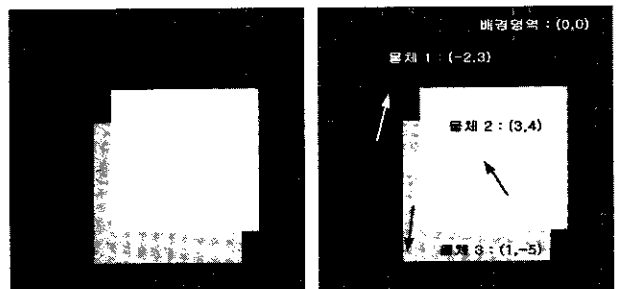
5. 실험 결과

본 논문에서 제안한 알고리즘을 검증하기 위해 '모의영상', 'Table Tennis', 'Football', 그리고, 'Mobile and Calendar' 시퀀스에 대한 모의실험을 실시하였다. 실험결과, 제안한 알고리즘에 대한 성능비교를 위해 전역 탐색법과 윌슨기반 정합법에 의해 각각 얻어진 프레임별 움직임 보상영상에 대한 평균절대오차와 움직임 보상영상을 비교하였다. 움직임 보상영상에 대한 평균절대오차 계산식은 식 (19)와 같다.

$$MAE \text{ per frame} = \frac{1}{|R_t|} \sum_{s \in R_t} |I_t(s) - \hat{I}_t(s)| \quad (19)$$

<표 1> 파라미터에 대한 사용값

파라미터	식	사용값
L	식 (2)	7
평균치 필터	식 (8)	5×5
T	식 (12)	1.0
λ	식 (13)	0.5
N	식 (14)	13



(그림 3) '모의영상'의 t번째 프레임과 물체영역별 움직임 벡터

식 (19)에서 I_t 는 t번째 프레임을 나타내며, \hat{I}_t 는 움직임 보상된 t번째 프레임을 나타낸다. 본 논문에서 사용된 파라미터는 <표 1>과 같으며, 영상의 밝기 특성과 계산량을 고려해서 실험적으로 선택하였다. (그림 3)은 '모의영상'의 t번째 프레임과

각 물체영역의 움직임 벡터 (u, v) 를 표시한 그림이며, 화소별로 0~25사이에 일양분포(uniform distribution)를 갖는 가산잡음을 추가하였다. (그림 3)에서 화살표는 각 물체영역의 실제 움직임을 나타낸다. (그림 3)에 대한 실험결과, 각 추정방법에 대한 움직임 보상영상과 니들그램(needlegram)은 (그림 4)와 같으며, 식 (1)에 의해 전역 탐색법으로 추정된 보상영상인 (a)의 경우, 움직임 경계에 속한 블록의 왜곡으로 화질이 손상된 것을 관찰할 수 있다. 그러나, 윤곽기반 정합법과 제안한 방법의 경우, 움직임 경계에서 우수한 추정결과를 얻을 수 있었다. 또한, 제안한 방법에 대한 움직임 벡터의 니들그램에서도 왜곡을 줄이기 위해 움직임 경계를 따라 움직임 벡터가 추정된 것을 볼 수 있었다. 제안한 방법에 대한 이러한 추정결과는 각 움직임 추정벡터에 대한 영역간 보상오차의 편차를 고려함으로써 비정합영역의 오차를 크게 하는 움직임 벡터에 대해 식 (13)에서 높은 정합오차를 부여하기 때문이다. 또한, <표 2>는 각 방법에 의해 얻어진 '모의영상'에 대한 움직임 보상영상의 평균절대오차를 나타내는 것으로 (그림 4)의 결과와는 달리 제안한 방법이 윤곽기반 정합법보다 평균절대오차에서 0.04정도 낮은 것을 보여준다. 이러한 평균절대오차에 대한 제안한 방법의 우수성은 다른 영상들에 대한 실험결과에서도 동일하게 관

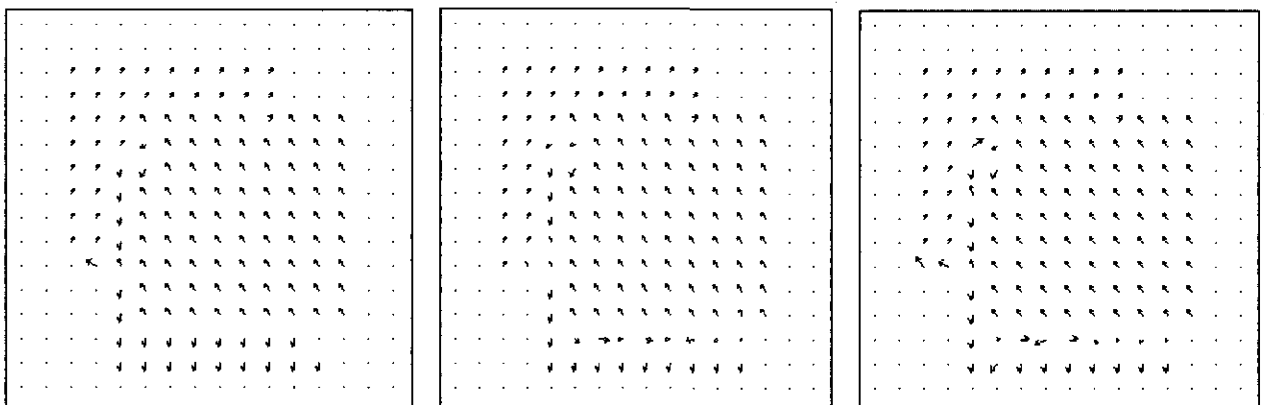
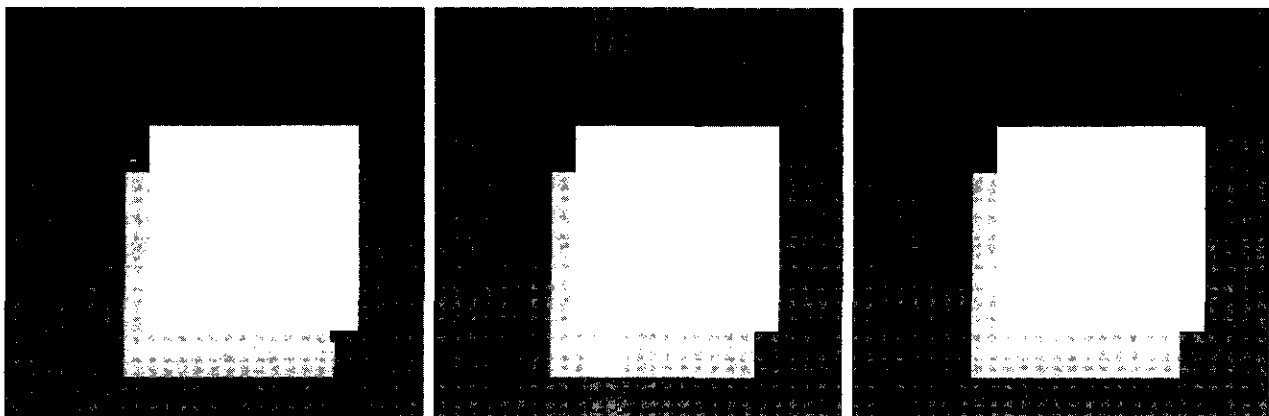
찰되며, 시각적인 측면 뿐만 아니라 객관적인 측면에서도 제안한 방법이 우수함을 검증한다.

<표 2> '모의영상'에 대한 방법별 평균절대오차

추정방법	평균절대오차
전역 탐색법	1.38
윤곽기반 정합법	1.52
제안한 방법	1.48

<표 3> 시퀀스별 각 방법에 대한 MAE 비교

MAE for Methods			
Sequences	Full Search Method	Edge Oriented BMA	Proposed Method
Tennis	4.92	5.29	4.97
Football	6.55	6.77	6.57
Mobile and Calendar	8.37	8.84	8.42
Foreman	2.93	3.02	2.98
Carphone	2.46	2.56	2.52
Garden	7.93	8.14	7.98
Salesman	4.20	4.42	4.26
Claire	3.43	3.43	3.43

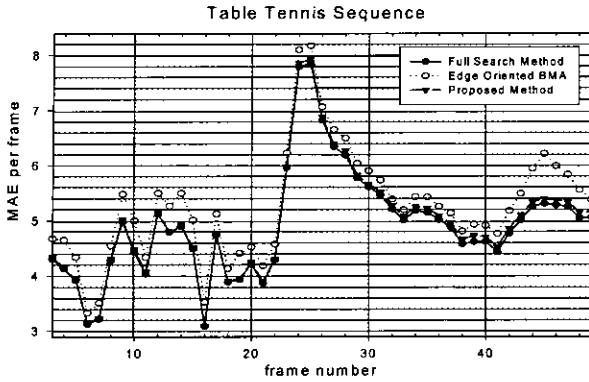


(a) 전역탐색법

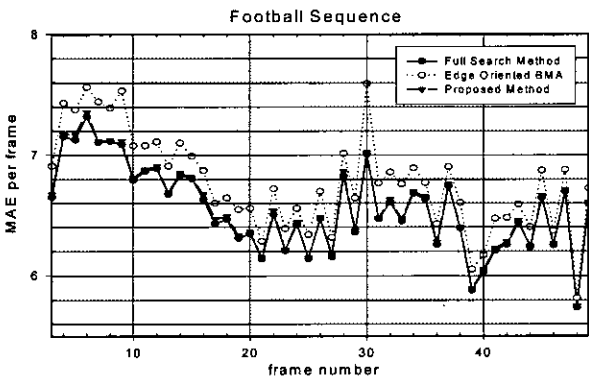
(b) 윤곽기반 정합법

(c) 제안한 방법

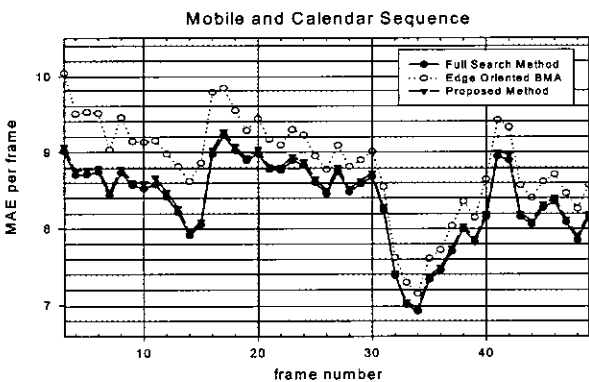
(그림 4) 방법별 추정된 움직임 보상영상 및 니들그램



(그림 5) 각 방법의 'Table Tennis'시퀀스에 대한 프레임별 평균절대오차



(그림 6) 각 방법의 'Football' 시퀀스에 대한 프레임별 평균절대오차



(그림 7) 각 방법의 'Mobile and Calendar' 시퀀스에 대한 프레임별 평균절대오차

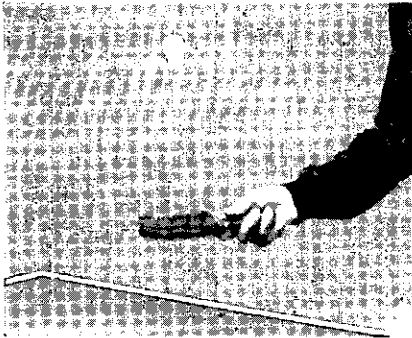
(그림 5), (그림 6), (그림 7)은 객관적인 추정결과를 비교하기 위하여 'Table Tennis', 'Football', 그리고 'Mobile and Calendar' 시퀀스에 대해 각각 전역 탐색법, 윤곽기반 정합법, 그리고 제안한 방법에 의해 얻어진 움직임 보상영상의 프레임별 평균절대오차를 나타낸 그래프이다. 제안한 방법에 의한 추정결과, 평균절대오차는 세 개의 시퀀스 모두에서 전역 탐색법에 의한 결과와 비슷하게 나타났으며, 기존의 방법인 윤곽기반 정합법에 의한 결과보다는 전체 프레

임에 대해 평균적으로 각각 'Table Tennis'시퀀스의 경우 0.32, 'Football' 시퀀스의 경우, 0.2, 그리고, 'Mobile and Calendar' 시퀀스의 경우는 0.42정도의 낮은 오차를 보이고 있어 평균절대오차에서 우수함을 확인할 수 있었다. <표 3>은 이 밖에 여러 다른 시퀀스들에 대해 각 방법별 평균절대오차를 정리한 것이다. 전체적으로 제안한 방법에 대한 오차가 윤곽기반 정합법에 의한 오차보다 낮음을 알 수 있다. 평균절대오차에 대한 실험결과, 본 논문에서 제안한 정합함수는 전체 블록에 대한 평균절대오차를 최소화하는 동시에 각 영역간의 오차편차를 최소화하는 움직임 벡터를 추정하기 때문에 움직임 보상시의 블록화 현상을 줄이고 화질을 개선시키는데 효과적임을 알 수 있다. (그림 8)은 화질비교를 위해 'Table Tennis' 시퀀스에서 3번, 22번, 그리고, 45번 프레임에 대해 각 방법별로 추정한 움직임 보상 영상을 보여준다.

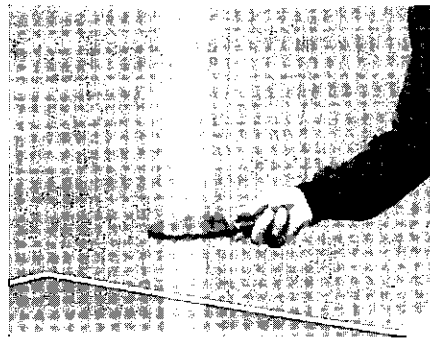
(그림 8) (a)의 전역 탐색법에 대한 경우, 배트(bat) 상단 부분에 왜곡이 관찰된다. 배트 상단에는 두 개의 움직임 영역, 즉 배경영역과 배트영역이 있다. 여기서, 식 (20)과 같이 배경영역의 움직임 벡터에 대한 비정합영역(배트영역)의 누적오차가 배트영역의 움직임 벡터에 대해 비정합영역(배경영역)의 누적오차보다 작기 때문에 배경영역의 움직임 벡터가 블록의 움직임 벡터로 추정된 것이며, 배트영역의 정합오차로 인해 경계에서 왜곡이 나타난 것이다.

$$\sum_{s \in R_{\text{background}}} |I_{t-1}(s + MV_{\text{bat}}) - I_{t-1}(s + MV_{\text{background}})| < \sum_{s \in R_{\text{nomatch}}} |I_{t-1}(s + MV_{\text{background}}) - I_{t-1}(s + MV_{\text{bat}})| \quad (20)$$

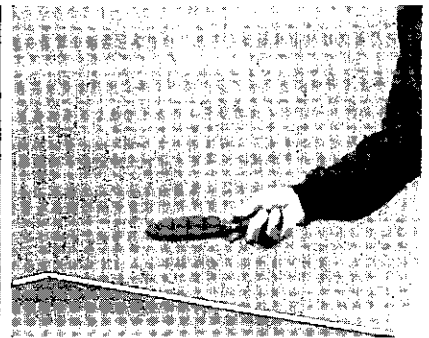
식 (20)에서 $R_{\text{background}}$ 는 배경영역에 대한 비정합영역으로 배트영역을 가리키며, $R_{\text{nomatch}}^{\text{bat}}$ 은 배트영역에 대한 비정합영역으로 배경영역을 가리킨다. 그리고, $MV_{\text{background}}$ 는 배경영역의 움직임 벡터를 나타내며, MV_{bat} 는 배트영역의 움직임 벡터를 나타낸다. (a)와 달리, (c)에 있어서는 제안한 방법에서 이러한 두 영역간 평균오차의 편차를 최소로 하는 움직임 벡터(배트영역의 움직임 벡터)를 추정하기 때문에 경계에서의 왜곡이 발견되지 않는다. 또한, 윤곽기반 정합법에 의한 (e)에 있어서도 공(ball)영역에 대한 윤곽의 흐려짐으로 인해 움직임이 잘못 추정된 것을 볼 수 있다. 그리고, (h)에서 흐려짐이 심한 왼쪽 손목영역에서도 추정된 움직임 벡터의 정확성이 크게 떨어져 블록화 현상이 심화된 것을 관찰할 수 있었다. 그러나, 제안한 정합함수가 사용된 움직임 보상영상에 있어서는 움직임 경계에서 보다 정확한 움직임을 추정할 수 있음을 확인할 수 있었다. (그림 9)는 'Table tennis'의 세 번째 프레임에 대해 추정된 움직임 벡터의 니들그램을 표시한 것이다. 배트상단 블록에 대한 니들그램의 경우, 제안한 방법에서는 배트영역과 같은 움직임 벡터가 추정되어 움직임 보상시의 왜곡이 감소될 수 있음을 알 수 있었다.



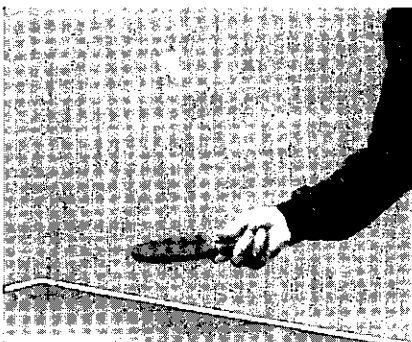
(a) 전역탐색법(3th frame)



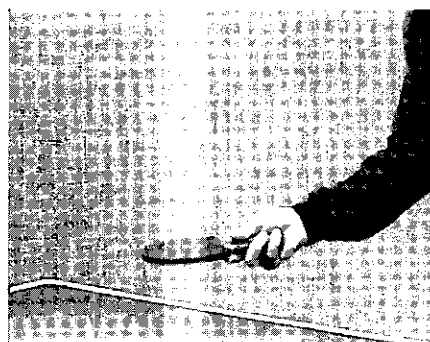
(b) 윤곽기반 정합법(3th frame)



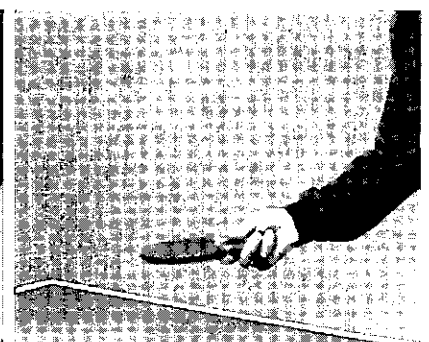
(c) 제안한 방법(3th frame)



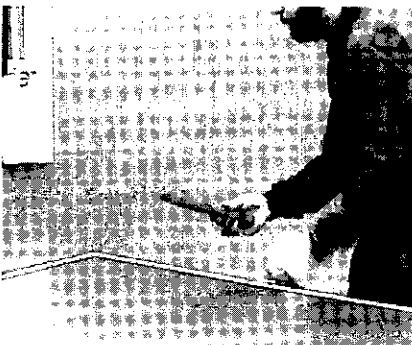
(d) 전역탐색법(22th frame)



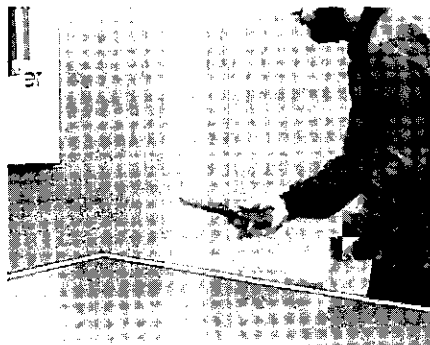
(e) 윤곽기반 정합법(22th frame)



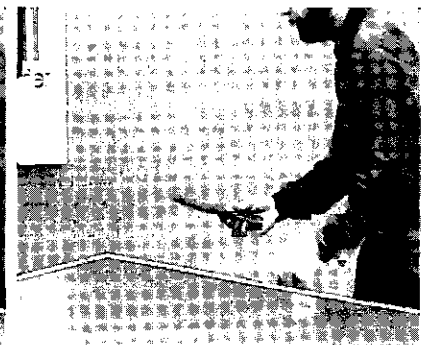
(f) 제안한 방법(22th frame)



(g) 전역탐색법(45th frame)

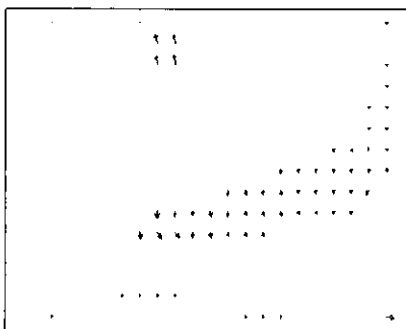


(h) 윤곽기반 정합법(45th frame)

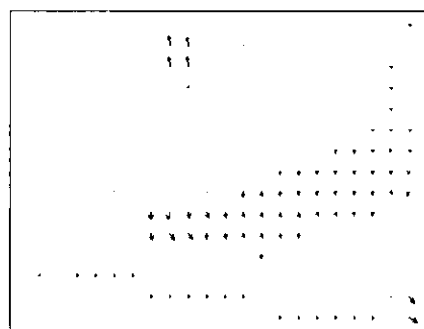


(i) 제안한 방법(45th frame)

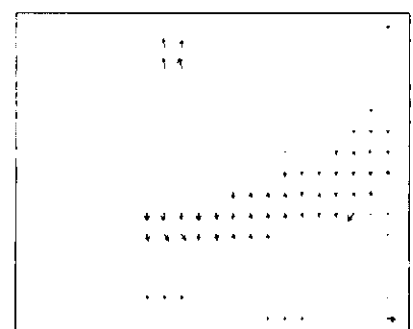
(그림 8) 'Table Tennis' 시퀀스에 대한 각 방법별 움직임 보상 결과영상



(a) 전역탐색법(3th frame)



(b) 윤곽기반 정합법(3th frame)



(c) 제안한 방법(3th frame)

(그림 9) 'Table Tennis' 시퀀스의 세 번째 프레임에 대한 각 방법별 니들그램

6. 결 론

본 논문에서는 움직임 경계에서 움직임 벡터의 정확성과 블록화 현상을 줄이기 위해 블록 내 화소들을 움직임 벡터의 시공간적 상관성을 이용하여 움직임 영역별로 분할한 후, 영역간 평균 움직임을 보상오차의 최소편차를 고려하는 새로운 정합함수를 정의함으로써 최적의 움직임 벡터를 추정하기 위한 방법을 제안하였다. 제안한 정합함수에 대한 검증결과 특히, 움직임 경계영역에서 정확한 움직임을 추정할 수 있었으며, 이를 통해 움직임 보상영상에 대한 시각적인 화질을 개선시킬 수 있었다. 또한, 움직임 보상영상에서 시각적으로는 윤곽기반 정합법과 유사한 결과를 보였지만, 프레임 전체에 대한 평균절대오차에서는 윤곽기반 정합법보다 나은 결과를 얻을 수 있음을 알 수 있었다.

참 고 문 헌

[1] ISO/IEC 11172-2(MPEG-1 Video), "Information Technology-Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbits/s : Video," 1993.

[2] ISO/IEC 13818-2 | ITU-T H.262((MPEG-2 Video), "Information Technology- Generic Coding of Moving Pictures and Associated Audio Information : Video," 1995.

[3] ITU-T Recommendation H.261, "Video codec for audio-visual services at P×64 k bits," ITU-T, March 1993.

[4] ITU-T DRAFT H.263, "Video coding for Low bitrate communication ITU-T," December 1995.

[5] Mei-Juan, Liang-Gee Chen, Tzi-Dar Chiueh, and Yung-Pin Lee, "A New Block-Matching Criterion for Motion Estimation and its Implementation," IEEE Trans. Circuits Syst. Video Technol., Vol.5, No.3, June 1995.

[6] P. Anandan, "A Unified Perspective on Computational Techniques for the Measurement of Visual Motion," International Conference on Computer Vision, London, UK, pp. 219-230, 1987.

[7] C. Auyeung, J. KosMach, M. Orchard, and T. Kalafatis, "Overlapped Block Motion Compensation," Proc. SPIE,

Visual Commun. Image Process., Boston, MA, Vol.1818, pp.561-571, 1992.

[8] M. H. Chan, Y. B. Yu, and A. G. Constantinides, "Variable Size Block Matching Motion Compensation with Applications to Video Coding," IEE Proc. I, Vol.137, No.4, pp. 205-212, 1990.

[9] Y. L. Chan and W. C. Siu, "Edge Oriented Block Motion Estimation for Video Coding," IEE Proc. Vis. Image Process., Vol.144, No.3, pp.136-144, June, 1990.

[10] Jyi-Chang Tsai, Chaur-Heh Hsieh, Shiuh-Ku Weng, and Mao-Fu Lai, "Block-matching motion estimation using correlation search algorithm," Signal Processing : Image Communication, 1998.



조 영 창

e-mail : yccho@kyungwoon.ac.kr
 1994년 영남대학교 전기공학과 졸업 (학사)
 1996년 영남대학교 대학원 전자공학과 (공학석사)
 2001년 영남대학교 대학원 전자공학과 (공학박사)

1999년~현재 경운대학교 정보통신공학과 전임강사
 관심분야 : 비디오 코딩, 움직임 추정, 동영상 복원 등



이 태 홍

e-mail : taehong@yeungnam.ac.kr
 1975년 서울대학교 공업교육학과 전자공학전공 졸업(학사)
 1980년 서울대학교 대학원 공업교육학과 전자공학전공(공학석사)
 1997년 서울대학교 대학원 전자공학과 (공학박사)

1976년~1979년 동양정밀(주) 중앙연구소 책임연구원
 1980년~1981년 금성전기(주) 기술연구소 기좌
 1981년~현재 영남대학교 전기전자공학부 교수
 1996년~1997년 한국통신학회 대구·경북 지부장
 관심분야 : 신호처리, 영상처리, 영상복원 등