

시간적 상관성을 이용한 적응적 블록 정합 알고리즘

윤 효 순[†] · 이 귀 상^{††}

요 약

움직임 추정과 움직임 보상기법은 연속한 비디오 프레임간의 시간적 중복성을 이용하여 동영상내에 존재하는 중복된 데이터를 제거하기 때문에 비디오 영상 압축에서 중요한 역할을 하지만 많은 계산량으로 인하여 실시간 응용 및 고해상도 응용에 많은 어려움을 가지고 있다. 만일 움직임 추정을 하기 전에 블록의 움직임을 예측할 수 있다면 이를 바탕으로 탐색영역에서 초기 탐색점 위치와 탐색 패턴을 결정할 수 있을 것이다. 본 논문에서는 움직임의 높은 시간적 상관성을 이용하여 초기 탐색점 위치와 탐색 패턴을 결정함으로써 적응적으로 움직임을 추정하는 새로운 기법을 제안한다. 실험을 통하여 제안된 알고리즘을 다이아몬드 탐색 기법과 비교하였을 경우, 제안된 알고리즘은 움직임 보상 예측된 화질에 있어서 약 0.1~0.5(dB)정도 성능을 향상시켰으며 움직임 벡터 추정의 속도에 있어서 약 50% 이상 높은 성능 향상을 보였다.

An Adaptive Block Matching Algorithm Based on Temporal Correlations

HyoSun Yoon[†] · GueeSang Lee^{††}

ABSTRACT

Since motion estimation and motion compensation methods remove the redundant data to employ the temporal redundancy in images, it plays an important role in digital video compression. Because of its high computational complexity, however, it is difficult to apply to high-resolution applications in real time environments. If we have information about the motion of an image block before the motion estimation, the location of a better starting point for the search of an exact motion vector can be determined to expedite the searching process. In this paper, we present an adaptive motion estimation approach based on temporal correlations of consecutive image frames that defines the search pattern and determines the location of the initial search point adaptively. Through experiments, compared with DS(Diamond Search) algorithm, the proposed algorithm is about 0.1~0.5(dB) better than DS in terms of PSNR(Peak Signal to Noise Ratio) and improves as high as 50% compared with DS in terms of average number of search point per motion vector estimation.

키워드 : 움직임 추정(Motion Estimation), 움직임 벡터(Motion Vector), 블록 정합 기법(Block Matching Algorithm)

1. 서 론

최근 정보 통신의 발달로 동영상 처리에 대한 요구가 증가 되므로 이에 따른 가장 큰 문제점은 데이터 증가이다. 동영상 내에 존재하는 중복된 데이터는 시간적, 공간적, 통계적 중복성을 이용하여 압축한다. 특히 동영상에서 가장 많은 데이터 중복성을 가지고 있는 시간적 중복성은 참조 프레임의 데이터를 이용하여 움직임 추정과 움직임 보상을 수행하고 이 때 추정된 움직임 벡터(Motion Vector : MV)에 의해서 보상된 영상과 원 영상과의 차 신호를 부호화 함으로써 높은 데이터 압축률을 가져온다.

일반적으로 알려진 움직임 추정(Motion Estimation : ME)

기법으로는 크게 블록 단위로 동일한 움직임을 갖는다는 가정 하의 움직임 정보를 찾는 블록 정합 알고리즘(Block Matching Algorithm : BMA)과 경사법을 이용하여 화소 단위로 움직임을 추정하는 화소 재귀적 알고리즘(Pel Recursive Algorithm : PRA)으로 나눌 수 있다. 현재 많은 비디오 코딩에서는 데이터 흐름의 규칙성, 계산의 복잡도, 하드웨어의 구현을 고려하여 블록 정합 알고리즘을 많이 사용하고 있다. 가장 간단한 블록 정합 알고리즘으로는 전역 탐색 기법이 있는데 이 기법은 움직임을 추정할 때 탐색범위내의 가능한 모든 블록을 조사하여 움직임 벡터를 찾는다. 전역 탐색 기법은 탐색범위 내에서 가장 적합한 움직임 벡터를 구할 수 있지만 계산량이 많으므로 실시간 비디오 코딩 응용 분야 및 소프트웨어 구현에 많은 어려움을 가지고 있다. 이러한 문제점을 해결하기 위하여 여러 가지 고속 블록 정합 기법(Fast Block Matching Algorithm : FBMA)들이 개발되었는데 대표적인 고속 블록 정합 기법에는 3단계 탐색(Three Step Search :

* 본 연구는 한국과학재단 목적기초 연구사업 지원(R02-2000-00280)에 의해 수행됨.

† 준 회원 : 전남대학교 전산학과

†† 종신회원 : 전남대학교 정보통신연구소, 전산학과 교수

논문접수 : 2001년 11월 10일, 심사완료 : 2002년 3월 5일

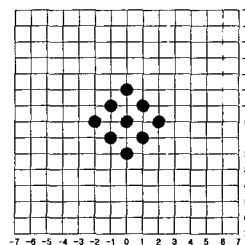
TSS)[1], 새로운 3단계 탐색(New Three Step Search : NTSS)[2], 2차원 로그형 탐색(2 Dimension LOGarithmic search : 2DLOG)[3], 4단계 탐색(Four Step Search : 4SS)[4], 다이아몬드 탐색(Diamond Search : DS)[5, 6], 그리고 2단계 탐색(2 Step Search : 2SS)[7, 8] 등이 있다. 그리고 동영상 이 내포하고 있는 움직임 특성, 즉 시간적 상관성을 바탕으로 참조 프레임에서 구해진 움직임 정보를 다음 프레임의 움직임 벡터를 구하기 위한 초기값으로 설정하는 기법들이 있다[9, 10]. 참조 프레임의 움직임 정보를 이용하면 적은 탐색 점 사용으로 양호한 움직임 보상된 화면과 현재 프레임의 움직임 벡터를 빠르게 추정할 수 있다. 위에서 나열한 기법들은 동영상의 구성요소에 상관없이, 즉 움직임이 없는 배경에서부터 움직이는 물체에 이르기까지 동일한 탐색 기법을 사용하여 움직임 추정을 하기 때문에 움직임 벡터 추정 시 많은 탐색점들을 사용하고, 해당 블록에 대한 움직임 정보를 가지고 있지 않기 때문에 항상 탐색 영역의 원점에서부터 움직임 추정을 해야하는 문제점을 가지고 있다. 이러한 문제점들은 움직임 벡터 추정 시간에 많은 영향을 미친다. 기존의 블록 정합 기법들의 문제점을 해결하기 위하여 본 논문에서는 연속된 프레임사이에 존재하는 시간적 상관성을 바탕으로, 참조 프레임의 움직임 벡터들을 이용하여 해당 프레임의 움직임 벡터를 추정하기 위한 초기 탐색 시작점을 결정하고 움직임 특성에 맞는 적응적인 탐색 패턴을 사용하여 움직임 추정을 하는 새로운 기법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 움직임 추정 기법에 대하여 설명하고, 3장에서는 제안된 기법을 기술하였다. 그리고 4장에서는 기존 기법들과 성능을 비교한 후, 마지막으로 5장에서는 결론을 맺는다.

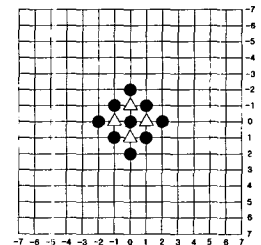
2. 기존의 움직임 추정(Motion Estimation)기법

동영상 안에 존재하는 시간적 중복성은 연속되는 화면사이의 유사성을 이용하는 방법으로 움직임 추정 및 보상 기법을 통하여 화면간 상관도를 최소화한다. 이를 위한 움직임 추정 기법은 영상의 각 단위로 블록 또는 화소 단위로 적용된다. 이 중 계산 복잡도 및 하드웨어 구현에 있어서 용이한 블록단위의 움직임 추정이 널리 사용된다. 블록단위의 움직임 추정은 두 가지 전제 조건을 가지고 있는데, 동일한 블록 내의 화소들은 동일한 움직임을 갖는다는 것과 블록들은 수평, 수직으로만 움직인다는 것이다. 즉 블록 정합 기법은 영상의 한 프레임을 동일한 크기의 블록들로 나누고 이들의 각 블록에 대하여 참조 프레임(Reference Frame)의 탐색 영역 내에서 정합 오차가 가장 작은 블록을 찾는다. 이때 현재 프레임(Current Frame)의 한 블록과 참조 프레임 내에서 가장 정합이 잘 되는 블록간의 위치차이를 움직임 벡터라 한다. 움직임 벡터를 추정하기 위하여 많은 고속 블록 정합 기법들

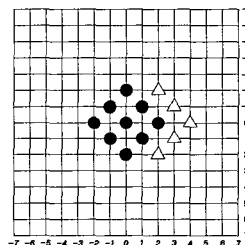
이 제안되었는데, 3단계 탐색 기법은 넓은 영역에 걸쳐 몇 개의 탐색점을 조사한 후 점차 범위를 좁혀 나가는 방식이고, 새로운 삼단계 탐색 기법은 움직임 벡터가 탐색 영역의 중심에 분포한다는 사실을 이용하여 3단계 탐색 기법을 보완한 기법이며, 4단계 탐색 기법은 새로운 3단계 탐색 기법의 계산량을 보완한 기법이다. 그리고 2차원 로그형 탐색 기법은 대부분의 움직임이 상하좌우 방향으로 일어난다는 사실을 이용한 기법이고, 다이아몬드 탐색 기법은 움직임 벡터의 분포의 형태가 다이아몬드라는 사실을 이용한 기법이다. 움직임 벡터가 일반적으로 탐색 영역의 중심에 치우쳐 존재한다는 가정하의 다이아몬드 탐색 기법은 영상의 움직임이 크고 작음에 무관하게 화질면에서나 속도면에서 가장 좋은 성능을 보이는데, 다이아몬드 탐색 기법은 다음과 같은 알고리즘을 수행하여 움직임 벡터를 추정한다[5, 6].



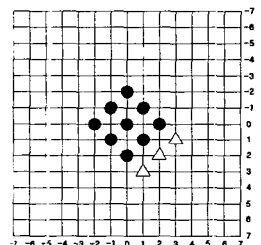
(A) 초기 탐색점 배치



(B) 최소 블록 오차를 가진 점의 위치가 중앙인 경우



(C)



(D)

- (C) 최소 블록 오차를 가진 점의 위치가 상, 하, 좌, 우 방향에 있는 경우 추가 탐색점의 위치
- (D) 최소 블록 오차를 가진 점의 위치가 대각선 방향에 있는 경우 추가 탐색점의 위치

(그림 1) 다이아몬드 탐색 기법

먼저 1단계에서는 (그림 1)의 (A)와 같이, 탐색 영역의 원점을 중심으로 9개의 탐색 점들을 배치한 후 각각의 탐색 점들에 대하여 블록 정합을 수행하여 최소 블록 정합 오차(Minimum Block Distortion : MBD)를 가지고 있는 점을 결정한다. 만약 최소 블록 정합 오차를 가진 점의 위치가 다이아몬드 패턴 중심인 경우 2단계를 수행하고, 그렇지 않은 경우에는 3단계를 수행한다. 그리고 2단계에서는 (그림 1)의 (B)와 같이 4개의 추가 탐색 점들을 배치한 후 이 점들에 대해 블록 정합을 수행하여 최소 블록 정합 오차를 가진 점을

결정한다. 이 때 최소 블록 정합 오차를 가진 점의 변위를 움직임 벡터로 추정하고 탐색을 마친다. 3단계에서는 최소 블록 정합 오차를 가진 점이 다이아몬드 패턴의 중심점을 중심으로 상, 하, 좌, 우에 있는 경우, (그림 1)의 (C)와 같이 5개의 추가 탐색점을 배치한다. 그리고 최소 블록 정합 오차를 가진 점이 대각선 방향에 있는 경우, (그림 1)의 (D)와 같이 3개의 추가 탐색점을 배치하여 새로운 다이아몬드 패턴을 만든다. 마지막으로, 4단계에서는 최소 블록 정합 오차를 가진 점의 위치가 다이아몬드 패턴의 중심이 될 때까지 위의 단계를 반복 수행하여 움직임 벡터를 추정한다.

일반적으로 영상의 움직임에 무관하게 안정된 결과를 보이는 다이아몬드 패턴은 탐색 영역의 중심에서 탐색 영역의 가장자리로 점진적인 탐색을 해나갈 때 탐색 영역의 중심에 있는 일부 탐색점들을 블록 정합 대상에서 제외시킴으로 정확한 움직임 벡터를 찾지 못하고, 정확한 움직임 벡터를 찾는 데 많은 탐색점을 사용하는 문제점들을 가지고 있다.

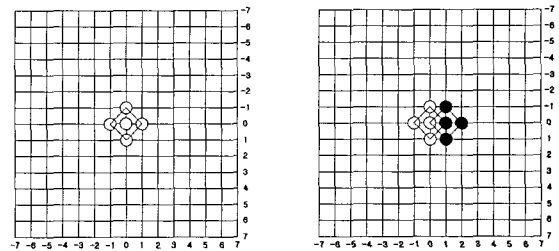
3. 시간적 상관성을 이용한 적응적 블록 정합 기법

동영상에서 인접 프레임간의 시간 간격은 매우 짧기 때문에 단위 프레임의 시간당 움직임 크기 변화량은 적은 범위로 제한된다고 볼 수 있다. 즉 연속하는 두 프레임간의 움직임에 많은 시간적 중복성을 가지고 있으므로 참조 프레임의 움직임 정보를 현재 프레임의 동일한 위치 매크로 블록의 탐색 시작점으로 사용함으로써 적은 탐색점들을 사용하여 움직임 벡터를 구할 수 있고 양호한 보상된 결과를 얻을 수 있다. 본 논문에서는 참조 프레임의 움직임 정보를 이용하여 해당 프레임에서 해당 블록의 움직임 추정을 위한 초기 탐색 점을 설정하고, 움직임의 특성에 맞게 적응적으로 움직임 패턴을 사용한 새로운 움직임 탐색 기법을 제안한다.

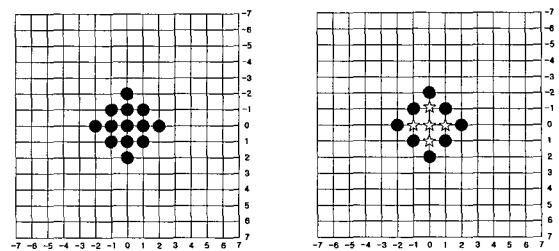
제안된 알고리즘에서는 참조 프레임의 움직임 벡터가 (0,0) 일 경우, 즉 배경과 같이 움직임이 없는 경우 현재 프레임에서 해당 블록의 움직임 벡터를 추정 할 때 탐색영역의 원점을 중심으로 (그림 2)와 같이 작은 다이아몬드 패턴을 사용하여 움직임 추정을 하는데, 이는 움직임이 없는 영역은 적은 탐색점만 사용하여 움직임 벡터를 추정한다. 그리고 참조 프레임의 움직임 벡터가 (0,0)이 아닌 경우, 탐색 영역에서 탐색 원점의 위치를 참조 프레임의 움직임 벡터만큼 이동시켜 새로운 탐색 원점을 설정한 후, (그림 3)과 같이 수정된 다이아몬드 기법을 이용하여 움직임을 추정한다.

본 논문에서는 기존의 다이아몬드 기법을 개선한 수정된 다이아몬드 기법을 사용하여 움직임 추정을 하는데, 기존의 다이아몬드 기법은 탐색점 추가 시 탐색영역의 중심에 있는 일부의 탐색점들을 정합 에러 계산에서 제외시킴으로 적합한 움직임 벡터를 찾지 못하고 많은 탐색점을 사용하는 문제점을 가지고 있다. 수정된 다이아몬드 기법은 일반적으로

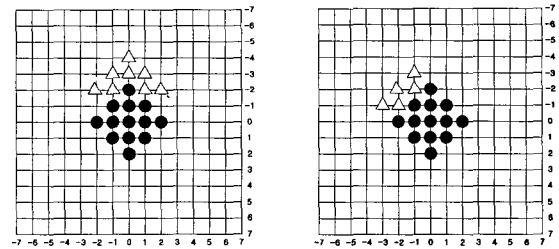
영상의 움직임 벡터들이 (그림 4)와 같이 탐색 영역 원점을 중심으로 반경 2pixels 이내에 분포할 확률이 약 56.72%~98.70[5, 6]라는 사실을 이용하여 초기 탐색점들을 배치하였고, 탐색 영역의 중심에서 가장 자리로 점진적으로 움직임 추정을 해 나갈 때, 탐색 영역의 중심에 있는 모든 탐색점들이 블록 정합의 대상이 되도록 추가 탐색점들을 적절히 배치하였다.



(A) 초기 탐색점 배치 (B) 추가 탐색점
(그림 2) 작은 다이아몬드 탐색 기법(참조 프레임의 움직임 벡터가 (0,0)일 때 사용)



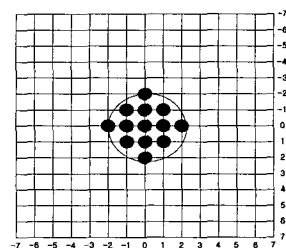
(A) 초기 탐색점 (B) 마지막 단계



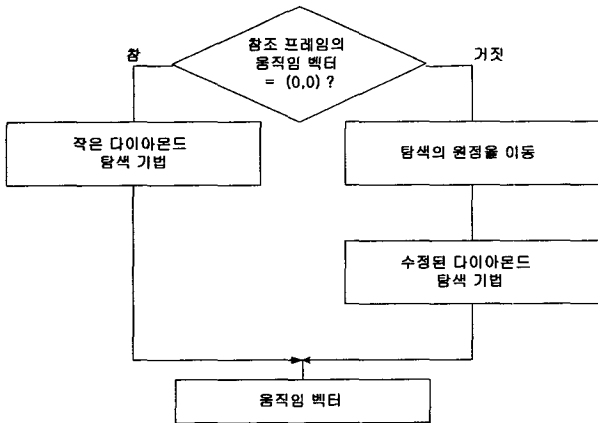
(C) (D)

(C) 최소 블록 오차를 가진 점의 위치가 상, 하, 좌, 우 방향에 있는 경우 추가 탐색점의 위치
(D) 최소 블록 오차를 가진 점의 위치가 대각선 방향에 있는 경우 추가 탐색점의 위치

(그림 3) 수정된 다이아몬드 탐색 기법(참조 프레임의 움직임 벡터가 (0,0)이 아닐 때 사용)



(그림 4) 움직임 벡터 분포도(움직임 벡터의 분포 확률을 이용한 초기 탐색점 배치)



(그림 5) 제안한 알고리즘의 순서도

(그림 5)는 제안한 알고리즘의 순서를 간단히 나타낸 것으로 제안된 알고리즘에서는 참조 프레임의 움직임 벡터값에 따라 현재 프레임에서 해당 매크로 블록의 탐색 기법을 적극적으로 선택하는데 만약 참조 프레임의 움직임 벡터가 (0,0)인 경우에는 작은 다이아몬드 탐색 기법을 사용하고 그렇지 않는 경우에는 수정된 다이아몬드 탐색 기법을 사용하여 움직임 추정을 하는데, 다음과 같은 알고리즘을 수행하여 움직임 벡터를 추정한다.

제안 알고리즘 : 시간적 상관성을 이용한 적응적 블록 정합 탐색 기법

1단계 : 참조 프레임의 움직임 벡터가 (0,0)인 경우에는 2단계를 수행하여 움직임 벡터를 추정하고, 그렇지 않는 경우 3단계를 수행하여 움직임 벡터를 추정한다.

2단계 : 작은 다이아몬드 탐색 - 참조 프레임의 움직임 벡터가 (0,0)인 경우에는

- ① (그림 2)의 (A)와 같이, 탐색 영역의 원점을 중심으로 5개의 탐색점들을 배치한 후 각각의 탐색점들에 대하여 블록 정합을 수행하여 최소 블록 정합 오차를 가진 점을 결정한다. 만약 최소 블록 정합 오차를 가진 점의 위치가 작은 다이아몬드 패턴의 중심에 있는 점이라면 이 점을 움직임 벡터로 추정하고 탐색을 마친다. 그렇지 않는 경우 ②를 수행한다.
- ② 최소 블록 정합 오차를 가진 점의 위치가 작은 다이아몬드 패턴 중심에 있는 점이 아닌 경우, (그림 2)의 (B)와 같이 3개의 추가 탐색점들을 배치하여 새로운 작은 다이아몬드 패턴을 만든다.
- ③ 최소 블록 정합 오차를 가진 점의 위치가 작은 다이아몬드 패턴의 중심에 있는 점이 될 때까지 위의 단계를 반복 수행하여 움직임 벡터를 추정한다.

3단계 : 수정된 다이아몬드 탐색 - 참조 프레임의 움직임 벡터가 (0,0)이 아닌 경우에는

- ① 탐색 영역의 탐색 원점을 참조 프레임의 동일한 위치 매크로 블록의 움직임 벡터만큼 이동시키고, 옮겨진 점의 변위를 새로운 탐색 영역의 탐색 원점으로 한다.
- ② (그림 3)의 (A)와 같이, 탐색 영역의 원점을 중심으로 13개의 탐색 점들을 배치한 후 각각의 탐색점들에 대하여 블록 정합을 수행하여 최소 블록 정합 오차를 가진 점을 결정한다. 만약 최소 블록 정합 오차를 가진 점의 위치가 (그림 3)의 (B)와 같이 다이아몬드 패턴 중심에 있는 5개의 점 중 하나 라면 이 점을 움직임 벡터로 추정하고 탐색을 마치고, 그렇지 않는 경우 ③을 수행한다.
- ③ 최소 블록 정합 오차를 가진점이 다이아몬드 패턴을 중심으로 상, 하, 좌, 우에 있는 경우, (그림 3)의 (C)와 같이 추가 탐색점들을 배치한다. 그리고 최소 블록 오차를 가진 점이 다이아몬드 패턴을 중심으로 대각선 방향에 있는 경우, (그림 3)의 (D)와 같이 추가 탐색점들을 배치하여 새로운 다이아몬드 패턴을 만든다.
- ④ 최소 블록 정합 오차를 가진 점의 위치가 (그림 3)의 (B)와 같이 다이아몬드 패턴 중심에 있는 5개의 점 중 하나가 될 때까지 위의 단계를 반복 수행하여 움직임 벡터를 추정한다.

4. 실험 결과

제안된 기법의 성능을 평가하기 위하여 실험 영상으로는 QCIF인 Suzie, Foreman, Mother and Daughter, Carphone, Salesman, Stefan, Table 그리고 Claire 영상의 각각 150 프레임을 사용하였고, 비교 탐색 기법으로는 전역 탐색 기법(FS), 삼단계 탐색 기법(TSS), 새로운 삼단계 탐색 기법(NTSS), 2단계 탐색 기법(2SS), 4단계 탐색 기법(FSS) 그리고 다이아몬드 탐색 기법(DS)들을 사용하였다.

성능 비교 함수로는 영상 화질의 품질을 평가하기 위하여 PSNR(Peak Signal-to-noise Ratio)을 사용하였는데, PSNR은 다음과 같이 정의되었다.

$$MSE = \left(\frac{1}{MN} \right) \sum_{m=1}^M \sum_{n=1}^N [x(m, n) - \hat{x}(m, n)]^2 \quad (1)$$

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (2)$$

위의 식 (1)에서 M, N은 영상의 가로와 세로의 크기이고, $x(m, n)$ 는 원 영상화면을 나타내고, $\hat{x}(m, n)$ 는 움직임 보상 예측된 화면을 나타낸다. 각 영상에 대한 실험 결과를 <표 1>, <표 2>에 나타냈는데 <표 1>에는 각 실험 영상에 대한 PSNR(dB)의 평균을 나타내었고, <표 2>에는 각 실험 영상에서 각 블록의 움직임 벡터 추정 시 사용되는 평균 탐색점 수 나타내었다. 제안된 탐색 기법은 움직임 추정 속도면이나

<표 1> 각 실험 영상의 PSNR

	FS	TSS	NTSS	FSS	2SS	DS	제안
stefan	23.887	20.117	22.249	22.620	23.858	22.775	23.160
foreman	29.547	26.735	28.195	28.220	29.243	28.663	29.064
suzie	32.196	30.395	31.646	31.549	32.162	31.890	32.085
table	26.509	23.682	25.607	24.811	26.274	25.671	25.702
carphone	30.889	29.418	30.742	30.150	30.777	30.485	30.709
salesman	32.703	32.365	32.691	32.538	32.709	32.621	32.695
claire	35.059	34.618	34.911	34.744	35.014	34.857	34.933
M & D	31.527	31.192	31.478	31.345	31.515	31.429	31.491

<표 2> 각 실험 영상의 평균 탐색점 수

	FS	TSS	NTSS	FSS	2SS	DS	제안
stefan	961	25	20.06	18.94	255	16.24	8.95
foreman	961	25	19.39	18.66	255	15.41	7.224
suzie	961	25	18.65	17.84	255	14.41	7.00
table	961	25	19.78	8.701	255	15.50	8.22
carphone	961	25	18.62	17.83	255	14.44	6.89
salesman	961	25	17.15	17.05	255	13.09	5.17
claire	961	25	17.24	17.08	255	13.15	5.20
M & D	961	25	17.37	17.15	255	13.27	5.44

움직임 추정 정확도면에서 다이아몬드 탐색 기법보다 더 나은 성능을 보였다. <표 1>에 나타나 있는 것과 같이 움직임 보상 예측된 화면의 화질면에 있어서 제안된 탐색기법을 다이아몬드 탐색 기법과 비교해 보았을 때, 제안된 기법은 Suzie, Salesman, Claire, Mother and Daughter 처럼 움직임이 작은 영상인 경우에는 0.1(dB) 정도 그리고 Stefan, Foreman, Carphone, Table 같이 많은 움직임이 있는 영상인 경우에는 현재 프레임에서 해당 매크로 블록의 탐색 원점을 참조 프레임의 동일한 위치 매크로 블록의 움직임 벡터만큼 이동시키고, 옮겨진 점의 변위를 새로운 탐색 영역의 탐색 원점으로 하고 이점을 중심으로 수정된 다이아몬드 탐색 기법을 이용하여 움직임 추정을 하므로 지역적 최소값에 수렴하는 문제를 다소 해결하여 0.5(dB) 정도 성능 향상을 보였다. <표 2>에 나타나 있는 것과 같이 제안된 탐색 기법은 Suzie, Salesman, Claire, Mother and Daughter 처럼 움직임이 작은 영상인 경우에는 영상의 프레임간의 시간적 상관성이 많이 존재하므로 다이아몬드 탐색 기법에 비해 움직임 추정에 필요한 탐색점 수를 65% 이상 줄였고 Stefan, Foreman, Carphone, Table 영상들 같이 많은 움직임을 가지고 있는 영상인 경우에는 프레임간의 시간적 상관성이 조금 존재하므로 상관성이 많이 존재하는 영상보다는 더 많은 탐색점 수를 필요로 하지만, 다이아몬드 탐색 기법에 비해 움직임 추정에 필요한 탐색점 수를 50% 이상 줄임으로 속도면에서 높은 성능 향상을 보였다. <표 1>에 나타나 있는 것과 같이 2단계 탐색 기법은 제안된 기법보다 좋은 PSNR 값을 가지고 있는데, 이는 <표 2>에

나타나 있는 것과 같이 제안된 탐색 기법보다 95배 이상 많은 탐색점 수를 사용하여 움직임 벡터를 추정하기 때문이다. 실험을 통하여, 제안된 탐색 기법은 전역 탐색 기법에 매우 근사한 움직임 예측 성능을 가지면서, 탐색 속도 면에서는 높은 성능 향상을 보였다.

5. 결 론

본 논문에서는 동영상 프레임간에 존재하는 시간적 상관성을 이용하여 현재 프레임에서 해당 매크로 블록의 움직임 벡터 추정 시, 참조 프레임에서 동일한 위치에 있는 매크로 블록의 움직임 정보를 이용하여 해당 매크로 블록의 탐색 원점과 탐색 패턴을 적응적으로 바꾸는 새로운 탐색 기법을 제안하였다. 실험을 통하여 알 수 있듯이, 제안된 기법은 움직임 추정 속도면이나 움직임 추정의 정확도에 있어서 다이아몬드 탐색 기법보다 나은 결과를 보였다. 다이아몬드 탐색 기법과 비교해서 볼 때, 제안된 기법은 움직임 추정에 필요한 탐색점수를 50% 이상 줄임으로서 속도 면에서 높은 성능 향상을 보였고, 움직임 예측면에서도 약 0.1~0.5(dB) 정도의 성능 향상을 보였다. 움직임 추정 시 움직임 벡터의 시간적 상관성 뿐 만 아니라 공간적 상관성을 이용하여, 현재 프레임의 매크로 블록 탐색 원점을 예측하면 보다 빠르게 움직임 추정을 할 수 있을 것이고 또한 움직임 예측면에서도 더 나은 결과를 얻을 수 있을 것이라 기대된다.

참 고 문 헌

- [1] T. Koga, K. Iinuma, A. Hirano, Y. Ishiguro, "Motion compensated interframe coding for video conference," Proc. NTC81, pp.G5.3.1-5.3.5, Nov. 1981.
- [2] R. Li, B. Zeng and M. L. Liou, "A New Three Step Search Algorithm for Block Motion Estimation," IEEE Trans. on Circuits and System for Video Technology, Vol.4, No.4, pp. 438-441, Aug. 1994.
- [3] J. R. Jain and A. K. Jain, "Displacement measurement and its application in Interframe image Coding," IEEE Trans. on Communications, Vol.29, No.12, pp.1779-1808, Dec. 1981.
- [4] L. M. Po and W. C. Ma, "A Novel Four Search Algorithm for Block Motion Estimation," IEEE Trans. on Circuit and Systems for Video Technology, Vol.6, pp.313-317, June, 1996.
- [5] J. Y. Tham, S. Ranganath and A. A. Kassim, "A Novel Unrestricted Center-Biased Diamond Search Algorithm for Block Motion Estimation," IEEE Transactions on Circuits and Systems for Video Technology, Vol.8, pp.369-377, Aug. 1998.
- [6] S. Zhu and K. K. Ma, "A New Diamond Search Algorithm for Fast Block Matching Motion," IEEE Transaction on Image Processing, Vol.9, No.2, pp.287-290, Feb. 2000.

- [7] Yuk Ying Chung, Neil W. Bergmann, "Fast Search Block Matching Motion Estimation Algorithm using FPGA," Visual Communications and Image Processing, Proc. SPIE, Vol.4067, pp.913-921, 2000.
- [8] Danian Gong, Yun He, "Fast Motion Estimation Algorithm using Horizontal and Multi-grid Search Strategy," Picture Coding Symposium, pp.362-365, 2001.
- [9] Jae Yeal Nam, Jae Soo Seo and Jin Suk Kwak, "New block-matching algorithm for motion estimation based on predictive direction information," Visual Communications and Image Processing, Proc. SPIE, Vol.4067, pp.1212-1220, 2000.
- [10] Alexis M, Tourapis, Goubin Shen, Ming L. Liou, Oscar C. Au, Ishfaq Ahmad, "A New Predictive Diamond Search Algorithm for Block Based Motion Estimation," Visual Communications and Image Processing, Proc. SPIE, Vol.4067, pp.1365-1373, 2000.
- [11] M. J. Chen, L. G. Chen, T. D. Chiueh and Y. P. Lee, "A new block matching criterion for motion estimation and its implementation," IEEE Transactions on Circuits and Systems for Video Technology, Vol.5, pp.231-236, June, 1995.



윤 효 순

e-mail : estheryoon@hotmail.com

1993년 호남대학교 전산통계학과 졸업(학사)

2002년 전남대학교 대학원 전산학과(석사)

현재 전남대학교 대학원 전산학과 박사과정

관심분야 : 영상 압축 및 복원, 에러 은닉



이 귀 상

e-mail : gslee@chonnam.ac.kr

1980년 서울대 공대 전기공학과 졸업(학사)

1982년 서울대 대학원 전자계산기공학과 석사

1982년 금성통신 연구소

1991년 Pennsylvania 주립대학 전산학과

박사

1984년~현재 전남대 전산학과 교수

관심분야 : 멀티미디어통신, 영상처리 및 복원, 논리합성, VLSI/

CAD