

H.263 비디오 통신을 위한 동일 움직임영역 기반 움직임벡터 복원

김정현[†]·손남례^{**}·박성찬^{**}·황성운^{***}·윤기송^{****}·손덕주^{****}·이귀상^{****}

요 약

이동통신 채널과 같이 여러 발생율이 높은 환경에서 부호화된 비디오를 전송할 때 채널에러에 손상된 비트스트림은 복호되기 어려울 뿐만 아니라, 비트스트림의 다른 부분으로까지 에러를 전파시킨다. 한 프레임에 손실블록이 있을 때 기존방법에서는 주변블록들의 움직임벡터의 평균을 구하거나 비슷한 예측을 통해 손실블록의 움직임벡터를 복원한다. 그러나 손실블록이 움직이는 객체의 경계부근에서 발생할 때 기존방법은 효율적이지 못하다. 따라서 제안 알고리즘은 기존방법보다 정확한 움직임벡터를 예측하기 위해 손실블록의 주변블록들 중에서 동일한 움직임을 갖는 블록들로 구성된 영역을 찾은 후, 동일움직임영역에 포함된 블록들의 움직임벡터를 이용하여 손실된 블록을 복원한다. 실험결과 제안방법이 기존방법에 비해 PSNR과 시각적인 화질면에서 우수한 성능을 보임을 알 수 있다.

Motion Vector Recovery Based on Homogeneous Motion Area for H.263 Video Communications

Jung-Hyun Kim[†]·Nam-Rye Son^{**}·Sung-Chan Park^{**}·Seong-Oun Hwang^{***}·
Ki-Song Yoon^{****}·Duk-Joo Son^{****}·Guee-Sang Lee^{****}

ABSTRACT

Bit streams corrupted by channel errors are not only difficult to be decoded but also propagate error to other part of the bit streams when highly compressed video is transmitted with noise as in mobile communication channels. In this paper, an algorithm for motion vector recovery is proposed. Given a lost block of an image, motion vectors of neighboring blocks are used to recover the motion vector of the missing block by calculating the average or similar estimations. Although such approach works moderately well in many cases, sometimes it fails to extract the best possible MV (Motion Vector) when edges of a moving object fall on the boundary of the missing block. In this paper, neighboring blocks with homogeneous movements are identified first and the motion vectors of such blocks are employed to produce more exact recovery of a motion vector of the missing block. Experimental results show that the proposed algorithm exhibits better performance in PSNR than existing error concealment methods.

키워드: 화상 및 음성처리, Error Concealment, Motion Vector Recovery, H.263

1. 서 론

최근 저속 이동 통신 채널에서의 멀티미디어 데이터 전송에 관련된 화상 전화나 화상 회의에 대한 관심이 많이

늘고 있다[1]. 이러한 멀티미디어 데이터 응용분야에서는 많은 양의 데이터를 전송해야 하기 때문에 높은 압축율을 갖는 부호화 알고리즘이 사용되는데, 대표적으로 초저속 통신망을 이용한 오디오/비디오 서비스에서 동영상 정보에 대한 압축표준인 H.263[2]을 들 수 있다. 그러나 이동 통신 채널처럼 에러율이 높은 통신망을 통해 이러한 방법으로 부호화된 비디오를 전송할 때 비트열에 에러가 발생하거나 패킷이 손실되면 에러가 발생한 영역의 화질 열화는 물론이고 발생한 에러가 시·공간적으로 전파되는 특성을 가진다.

이러한 채널에러의 영향을 줄이기 위해 ARQ (Automatic Retransmission on Request)[3], FEC (Forward Error Correction)[4, 5], 인터리빙(interleaving) 기법 등이 제안되었으

* 본 연구는 한국과학재단 목적기초연구과제(2000-1-30300-005-3)에 의하여 수행되었음.

† 준 회 원 : 한국전자통신연구원 컴퓨터소프트웨어기술연구소 인터넷 서비스연구부 네트워크소프트웨어연구팀 연구원

** 준 회 원 : 전남대학교 대학원 전산학과

*** 정 회 원 : 한국전자통신연구원 컴퓨터소프트웨어기술연구소 인터넷 서비스연구부 네트워크소프트웨어연구팀 연구원

**** 정 회 원 : 한국전자통신연구원 컴퓨터소프트웨어기술연구소 인터넷 서비스연구부 네트워크소프트웨어연구팀 책임연구원

**** 정 회 원 : 전남대학교 정보통신연구소 전산학과 교수

논문접수: 2000년 12월 18일, 심사완료: 2001년 2월 28일

나 이 방법들은 전송 데이터 증가 및 전송 지연의 단점이 있기 때문에, 저속 통신망에서 독자적으로 사용될 경우 시스템에서 요구하는 데이터 전송율을 유지하면서 적절한 오류 방지 효과를 얻는데 제한이 있다. 이에 비해 인코더의 수정이나 추가정보 없이 비디오 시퀀스의 시간적·공간적 상관성을 이용하여 손실된 영상을 복구하는 에러 은닉 기법[6]은 전송 대역이나 전송 지연에 크게 제약을 받지 않으면서도 우수한 성능을 보인다.

에러 은닉은 전송된 비트열에 에러가 발생하여 정상적으로 복호화되지 않고 복원된 영상의 화질을 떨어뜨리는 블록을 수신부에서 은닉하는 방법으로 에러 검출 전처리 과정이 처리된 후에 수행되며, 주변 매크로블록 정보들을 이용하여 공간적인 영역에서 손실된 블록들을 보상하는 방법과 시간적 영역에서 보상하는 방법으로 나눌 수 있다. 시간적 영역에서 에러은닉 기법은 복원된 주변 블록들의 정보를 이용해서 손실블록의 움직임벡터를 추정한 후, 이전 프레임으로부터 손실된 블록을 보상하는 방법이다. 공간적 영역에서 에러은닉 기법은 움직임이 많은 영역, 즉 시간적 에러은닉 방법이 성공적이지 못할 경우 또는 인트라모드로 부호화된 블록에 효과적인 방법으로 주변 블록들을 보간함으로써 가능하다.

본 논문에서는 통신 채널을 통하여 전송되는 과정에서 발생한 에러에 의해 손상된 블록을 복원하기 위해 동일 움직임 영역 추출 기법을 이용해 움직임 벡터를 예측하는 방법을 제안하였다. 일반적인 비디오 시퀀스에서 하나의 큰 움직임 객체는 여러 개의 작은 블록들로 이루어져 있으며, 이렇게 공간적으로 인접한 블록들은 비슷한 속도로 거의 같은 방향으로 움직이게 된다[7]. 그래서 공간적으로 인접한 블록들의 움직임벡터 정보는 상관성이 높다고 할 수 있다. 이러한 특성을 이용하여 기존의 알고리즘에서는 손실블록의 움직임벡터를 복원하기 위해 주로 주변블록의 움직임 벡터 정보들의 평균값(mean)이나 중간값(median)을 사용한다. 이때 참조되는 블록들은 일반적으로 손실블록의 상하좌우 인접 블록이거나 모든 주변블록 들이 될 수 있다.

그러나 움직이는 객체의 경계 부근에서 손실이 발생할 때, 즉 움직임 객체의 경계가 손실블록과 주변블록에 걸쳐 있는 경우 기존방법을 이용하여 복원하게 되면 정확한 움직임벡터를 찾기 어렵다. 손실블록과 주변블록들 중 일부만이 움직임 객체에 포함되어 있지만 일반적인 경우에 참조하는 주변블록들을 똑같이 참조하기 때문이다. 따라서 주변블록들 중에서 어떤 블록들이 그 움직임 객체에 포함이 되는지를 판단하고 이 블록들과 움직임 객체에 포함되지 않는 블록들을 구별하여 참조할 필요가 있다. 제안 알고리즘에서는 손실블록의 주변블록들 중 일부블록만이 움직임 객체에 포함될 경우 주변블록들 중에서 동일한 움직임을 갖는 영역을 추출하고 이 영역에 포함된 블록들의 움직임벡터를 이용하여 손실블록의 움직임벡터를 복원하게 된다. 실

험은 H.263 부호화를 기반으로 하였으며, 제안 방법에서는 디코더가 손실된 매크로블록의 위치를 알고있음을 가정한다.

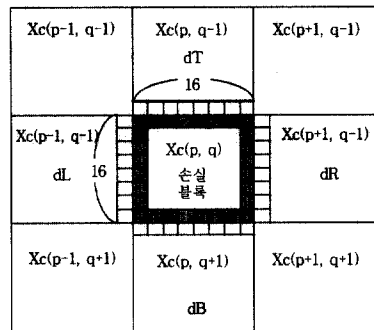
본 논문의 구성은 다음과 같다. 제2장에서는 기본 개념과 관련된 연구에 대해 언급하고, 제3장에서는 제안된 움직임 벡터 복원기법을 설명한다. 마지막으로 제4장과 제5장에서는 제안된 방법에 대한 실험결과를 제시하고 결론을 맺는다.

2. 관련 연구

시간적 에러은닉 방법 중에서 가장 간단한 방법으로는 손실된 블록과 동일한 위치의 블록을 이전 프레임으로부터 그대로 가져와 대체하는 방법이 있는데, 움직임이 적은 영상에 대해서는 효과적이지만 움직임이 큰 영상에 대해서는 영상 블록의 경계가 일치하지 못하는 현상이 발생한다.

Boundary Matching Algorithm (BMA)[8-10]은 손실블록의 에러은닉을 위해 가장 널리 사용되는 방법으로, 손실블록과 공간적으로 이웃한 블록들과의 경계에 있는 픽셀의 휘도값이 유사하다는 가정에서 이용한다. BMA는 후보 움직임 벡터 집합 중에서 손실블록과 인접블록과의 경계 픽셀 사이에 최소의 분산값을 갖는 후보 움직임벡터를 찾아 손실블록의 움직임벡터로 복원하게 된다.

한 프레임에서 (p, q)에 위치한 크기가 n×n인 블록을 X(p, q), 이 블록에 해당하는 움직임벡터를 V(p, q)라 가정하고, 만일 V(p, q)가 손실되거나 손상되어 수신된 경우 복원된 움직임 벡터를 V'(p, q)로 나타낸다. [X(p, q)]^{row}과 [X(p, q)]^{col}은 각각 블록 X(p, q)의 1번째 행에 있는 픽셀들과 1번째 열에 있는 픽셀들을 의미한다. 즉, [X(p, q)]^{row} = (x₁₁, x₂₁, x₃₁, ..., x_{n1}) 이고 [X(p, q)]^{col} = (x₁₁, x₁₂, x₁₃, ..., x_{1n})과 같다.



(그림 1) BMA에서 참조하는 주변블록의 움직임 벡터[7]

각 후보 움직임벡터를 이용하여 손실블록을 복원한 후, (그림 1)에서 손실블록과 상하좌우에 인접한 주변블록과의 경계에 있는 픽셀들의 휘도값의 유사성을 계산한다. 이 값은 Side Match Distortion (SMD)으로 나타낼 수 있는데, SMD는 식 (1)과 같이 정의 할 수 있다.

$$dT = \sum_{j=1}^n | [X_{MC}(p, q)]_{1,j}^{row} - [X(p, q-1)]_{n,j}^{row} |$$

$$dL = \sum_{j=1}^n |[X_{MC}(p, q)]_{i,j}^{col} - [X(p-1, q)]_{i,j}^{col}| \quad (1)$$

$$dR = \sum_{j=1}^n |[X_{MC}(p, q)]_{i,j}^{col} - [X(p+1, q)]_{i,j}^{col}|$$

$$dB = \sum_{j=1}^n |[X_{MC}(p, q)]_{i,j}^{row} - [X(p, q+1)]_{i,j}^{row}|$$

$$d_{sum} = dT + dB + dL + dR$$

식 (1)에서 dT, dB, dL, dR 는 각각 손실블록과 상하좌우에 인접한 블록과의 경계에 있는 픽셀들의 휘도값을 비교한 것이며 d_{sum} 은 이 값들을 모두 더한 값으로 SMD를 나타낸다. 이때, $X_{MC}(p, q)$ 는 후보 움직임벡터를 이용하여 복원한 블록이다. 후보 움직임벡터 집합에서 최소의 SMD를 갖는 움직임벡터를 손실블록의 움직임벡터로 복원하게 된다.

본 논문에서 BMA을 얻기 위해 사용하는 후보 움직임벡터는 다음과 같다.

- 이전 프레임에서 손실블록과 같은 위치에 있는 블록의 움직임 벡터
- 주변블록의 움직임벡터의 중간 값
- 주변블록의 움직임벡터의 평균 값
- 0(zero)

일반적인 경우 BMA는 좋은 성능을 보이지만, 움직임 객체의 경계가 손실블록과 주변 블록에 걸쳐 있는 경우 최적의 움직임 벡터를 찾는 데 어려움이 있다. 즉 손실블록과 주변블록 중 일부만이 움직임 객체에 포함되어 있는 경우 손실블록의 움직임과 관련이 있는 주변블록만을 참조하는 것이 아니라 일반적인 경우와 같은 주변블록을 참조하게 되고 손실블록과 주변블록의 경계 픽셀간의 유사성만을 이용하기 때문에 최적의 움직임벡터를 못 찾고 결과적으로 손실블록에 대한 복원이 어렵게 된다.

3. 제안 알고리즘

위와 같은 문제점을 해결하기 위한 제안방법으로는 움직임 객체가 손실블록과 주변블록에 걸쳐 있는 경우 주변블록들 중에서 그 움직임객체에 포함되는 블록들을 구분한다. 이 블록들은 하나의 움직임 객체에 포함되기 때문에 모두 같은 움직임을 가지고 이동하게 되므로 동일움직임 영역으로 나타낸다. 손실블록의 주변블록들 중에서 동일움직임 영역을 찾은 후, 손실블록이 동일 움직임영역과 같은 움직임을 갖는지 조사한다. 즉, 동일 움직임영역이 포함된 움직임 객체에 손실블록이 포함되는지를 판단하는 것이다. 손실블록이 동일 움직임영역과 같은 움직임을 갖는 것은 동일 움직임영역과 손실블록이 같은 방향과 크기로 움직이는 것을 의미하므로 주변블록들 중에서 이 영역에 포함된 블록들의 움직임 벡터를 이용하여 손실블록의 움직임벡터를 복원하게 된다.

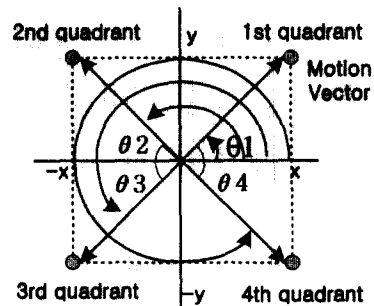
주변블록들 중 움직임이 같은 블록들을 동일 움직임영역으로 구분한다. 이때 블록의 움직임은 그 블록의 움직임벡터의 방향과 크기로 표현된다. 움직임벡터의 방향과 크기는 그 움직임벡터의 x, y 좌표에 대한 기울기와 유클리디언 거리를 계산함으로써 구할 수 있다. 식 (2)는 블록의 움직임벡터를 가지고 블록의 움직임 크기를 구하는 방법이다. 블록의 움직임 방향은 그 블록의 움직임벡터의 기울기를 이용하여 얻을 수 있기 때문에 기울기를 구하기 위해 우선 식 (3)을 이용하여 원점에서 움직임벡터가 위치한 좌표까지 이은 직선과 X좌표 축이 이루는 각을 계산한다. 즉 이 각은 (그림 2)에서 보이는 것처럼 움직임벡터의 x, y좌표가 위치한 사분면에 따라 각각 1사분면에서 4사분면까지 차례로 각각 $\theta 1, \theta 2, \theta 3, \theta 4$ 로 나타낼 수 있다.

$$d = \sqrt{x^2 + y^2} \quad (2)$$

$$rad = \arcsin\left(\frac{|y|}{\sqrt{x^2 + y^2}}\right)$$

$$\theta = rad \times \frac{\pi}{180} \quad (3)$$

식 (3)에 의해 계산된 θ 를 수정함으로써 움직임벡터의 기울기를 구할 수 있다. <표 1>은 움직임벡터의 x, y 좌표가 위치한 사분면에 따라 θ 를 수정하는 방법을 보인다.



(그림 2) 움직임벡터가 위치한 사분면[11]

<표 1> 움직임벡터가 위치한 사분면에 따른 기울기 계산[11]

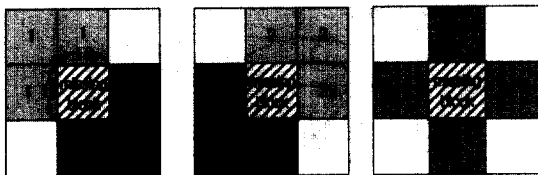
움직임벡터의 좌표 위치	움직임벡터의 기울기
1 사분면	$G = \theta$
2 사분면	$G = 180 - \theta$
3 사분면	$G = 180 + \theta$
4 사분면	$G = 360 - \theta$

위와 같은 방법으로 얻은 움직임벡터의 기울기를 이용하여 블록의 움직임 방향을 계산하게 되는데, 처리과정을 단순화하고 각 블록의 움직임벡터 사이에 있어서 어느 정도 오차를 허용하기 위해 <표 2>에서 보이는 바와 같이 기울기를 0° 에서 360° 까지 45° 간격으로 9가지 방향으로 구분한다. 즉, 모든 블록은 움직임벡터의 기울기에 따라 9가지 방향 중 하나를 갖게 되는 것이다.

〈표 2〉 움직임벡터의 기울기와 블록의 움직임 방향과의 관계

움직임벡터의 좌표 위치	움직임벡터의 기울기
$G = 0^\circ$	0 direction
$-22.5 < G < 22.5$	1 direction
$22.5 < G < 67.5$	2 direction
$67.5 < G < 112.5$	3 direction
$112.5 < G < 157.5$	4 direction
$157.5 < G < 202.5$	5 direction
$202.5 < G < 247.5$	6 direction
$247.5 < G < 292.5$	7 direction
$292.5 < G < 337.5$	8 direction

손실블록과 이웃한 8개의 주변블록들에 대한 움직임 방향과 크기를 위와 같은 방법으로 구한 후 이것을 이용하여 동일움직임영역을 찾는다. 동일 움직임영역의 빠른 탐색을 위해 주변블록을 (그림 3)의 (a)와 같이 4개의 영역으로 미리 구분하고 각 영역에 포함된 블록들이 모두 같은 움직임 방향을 갖는지 조사한다. 한 영역에 내에 있는 모든 블록의 움직임 방향이 같으면 그 블록들간의 움직임 크기를 비교하게 되는데, 블록들간의 움직임 크기의 차가 임계값 이하인 경우 이 영역을 동일 움직임영역으로 판단한다. 즉, 이 영역에 포함된 블록들의 움직임 방향은 모두 같고 움직임 크기 또한 비슷한 값을 갖게 된다.



(a) 동일움직임 영역구분 (b) 움직임이 독립적일 때 참조블록
(그림 3) 주변블록의 구분

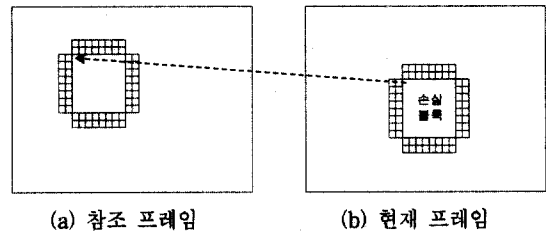
동일 움직임영역을 찾은 후, 손실블록이 이 동일 움직임 영역과 같은 움직임을 갖는지 아니면 다른 움직임을 갖는지를 판단하여 만일 손실블록의 움직임이 이 영역의 움직임과 같은 경우, 즉 손실블록 또한 이 동일 움직임영역이 구성하고 있는 움직임 객체에 포함될 가능성이 높을 때는 이 영역에 포함된 블록들만을 이용하여 손실블록을 복원한다. 손실블록의 움직임이 동일 움직임영역과 독립적일 때는 주변블록들 중 특정 영역에 손실블록이 포함된다고 할 수 없기 때문에 (그림 3)의 (b)와 같이 주변블록들 중 색깔이 어두운 블록들, 즉 상하좌우로 인접한 블록들을 참조하게 된다.

주변블록들의 움직임벡터를 참조하는 방법은 식 (4)와 같다. 참조할 블록들의 움직임벡터의 평균을 구한 후, 각 블록의 움직임벡터와 이 평균과의 차가 최소인 것을 손실블록의 움직임벡터로 복원한다. 식 (4)에서 n은 참조할 블록의 개수를 나타낸다.

$$Ave = \left(\sum_{i=1}^n MV_i \right) / n \quad (4)$$

$$Arg \min_{i=1 \dots n} (MV_i - Ave)$$

제안방법에서는 손실블록의 움직임이 동일 움직임영역과 같은지 아니면 독립적인 움직임을 갖는지 판단하기 위해 MSME (Modified Side Motion Estimation) 값을 이용한다. MSME는 SME[12]를 수정한 방법으로 (그림 4)에서 보여지는 것처럼 현재 프레임에서 손실블록의 경계 바깥쪽에 있는 픽셀과 참조프레임에서 움직임 보상된 블록의 경계 바깥쪽에 있는 픽셀간의 휘도값의 유사성을 비교하는 것으로 식 (5)에 계산하는 방법을 보인다.



(그림 4) Modified Side Motion Estimation

$$dT = \sum_{i=1}^2 \sum_{j=1}^n | [X_{MC}(p, q-i)_{1,j}^{row} - X(p, q-i)_{n,j}^{row}] |$$

$$dL = \sum_{i=1}^2 \sum_{j=1}^n | [X_{MC}(p-i, q)_{1,j}^{col} - X(p-i, q)_{n,j}^{col}] | \quad (5)$$

$$dR = \sum_{i=1}^2 \sum_{j=1}^n | [X_{MC}(p+i, q)_{1,j}^{col} - X(p+i, q)_{n,j}^{col}] |$$

$$dB = \sum_{i=1}^2 \sum_{j=1}^n | [X_{MC}(p, q+i)_{1,j}^{row} - X(p, q+i)_{n,j}^{row}] |$$

$$d_{sum} = dT + dL + dR + dB$$

MSME는 SME에서 사용하는 식의 계산량을 줄이기 위해 픽셀간 차의 제곱 대신 절대값을 이용한다. 객체의 예지가 손실블록의 경계부근에 있을 때 BMA에서 사용하는 SMD를 이용할 경우 정확한 움직임벡터를 찾는데 실패할 확률이 높은 반면에, MSME는 손실블록 자체를 비교하는 대신 블록의 바깥쪽에 위치한 픽셀들을 비교하기 때문에 SMD에 비해 더 효율적이다. SME 알고리즘과 유사한 방법으로는 DMVE (Decoder Motion Vector Estimation)[13]가 있다.

동일 움직임영역에 포함된 블록들을 이용하여 손실블록을 복원 했을 때와 (그림 3)의 (b)와 같이 손실블록의 움직임이 독립적일 때 참조하는 블록을 사용하여 복원했을 때 각각의 MSME값을 계산하여 비교한다. 이때 더 작은 MSME값을 갖는 쪽의 움직임벡터를 손실블록의 움직임벡터로 복원하게 된다. 제안 알고리즘을 (그림 5)과 같이 손실블록의 주변블록 중에서 동일 움직임 영역을 찾은 후 추출한 동일 움직임영역의 개수에 따라 처리한다.

동일 움직임영역의 개수가 하나인 경우는 위에서 설명한

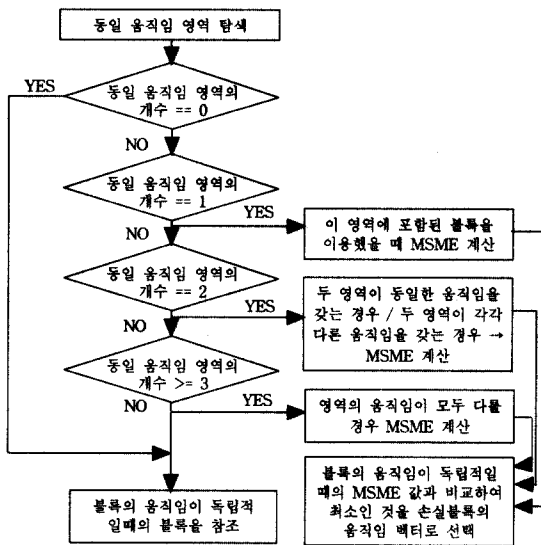
것과 같은 방법으로 처리하게 되며 2개일 때는 우선 두 영역이 같은 움직임을 갖는지 조사한다. 두 영역의 움직임이 같은 경우 두 영역 모두 움직임 재체에 포함되어 되어 같은 방향과 크기로 이동하는 것을 의미하므로 (그림 6)에 나타난 것처럼 두 영역의 위치에 따라 블록을 참조하게 된다. 이때 진한색 블록이 참조되는 블록들이며, (e)와 (f)는 주변 블록에 전체적으로 분포되어 있기 때문에 움직임이 독립적일 때와 같은 방법으로 블록을 참조한다. 두 영역의 움직임이 다른 경우에는 하나일 때와 마찬가지로 각 움직임 영역에 포함된 블록을 이용하여 손실블록을 복원했을 때의 MSME값과 손실블록의 움직임이 독립적일 때의 방법을 이용하여 복원한 경우의 MSME를 비교하여 최소의 MSME 값을 갖는 것을 손실블록의 움직임벡터로 복원한다. 동일 움직임영역의 개수가 3개 이상일 때는 각 영역이 모두 다른 움직임을 갖을 경우에만 위에서 설명한 것과 같은 방법으로 처리하고, 이외의 경우 즉 동일움직임이 없는 경우는 주변블록들 중 특정영역의 상관성을 찾기 어렵기 때문에

주변블록들 중 특정영역의 상관성을 찾기 어렵기 때문에 손실블록의 움직임이 독립적일 때의 방법을 이용하여 복원한다.

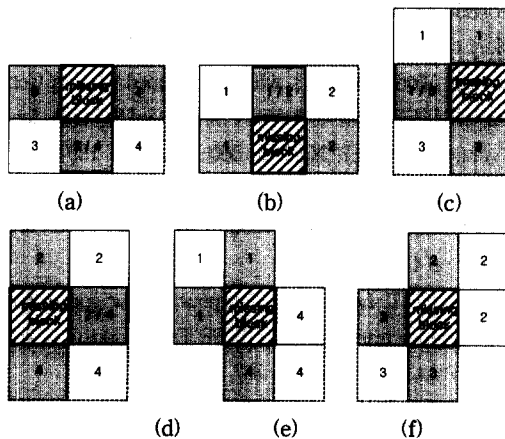
4. 실험 결과

<표 3> 실험영상에 대한 블록 손실율에 따른 PSNR 비교

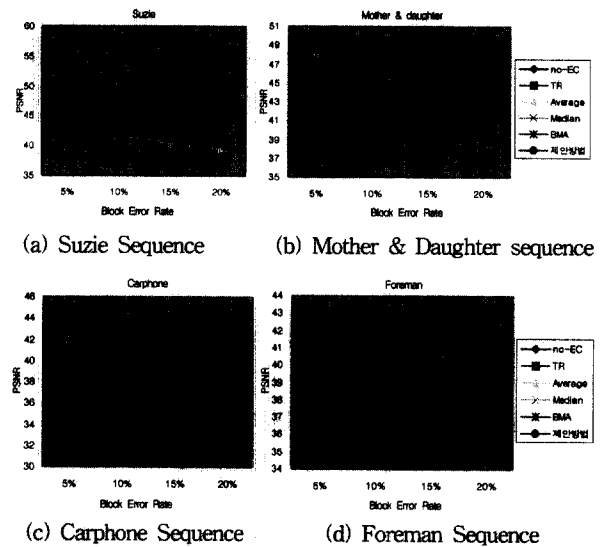
Image	Block Error Rate	No-EC	TR	Average	Median	BMA	Proposed Method
Suzie	5%	45.7	52.29	54.25	53.32	53.32	54.75
	10%	37.85	38.89	41.07	42.09	42.03	43.83
	15%	37.42	37.87	40.42	41.24	41.24	42.92
	20%	36.92	37.01	39.42	40.68	40.82	42.04
Mother & Daughter	5%	41.98	43.72	48.05	48.67	48.67	49.42
	10%	40.08	42.69	47.01	47.01	47.01	48.31
	15%	38.68	40.65	45.52	45.59	45.59	46.42
	20%	37.12	39.44	43.79	43.79	43.79	45.72
Carphone	5%	34.25	38.91	42.08	43.08	43.08	43.87
	10%	33.97	35.68	39.47	39.47	39.47	40.32
	15%	33.41	35.18	39.16	39.02	39.16	39.89
	20%	32.84	34.88	38.44	39.09	39.09	39.42
Foreman	5%	37.36	41.57	41.73	41.39	41.73	42.58
	10%	37.15	41.05	40.07	40.88	41.05	41.74
	15%	36.72	40.84	40.58	40.52	40.84	41.25
	20%	36.39	40.41	40.05	40.05	40.41	40.92



(그림 5) 제안 알고리즘



(그림 6) 동일 움직임 영역이 두개 일 때 참조되는 블록

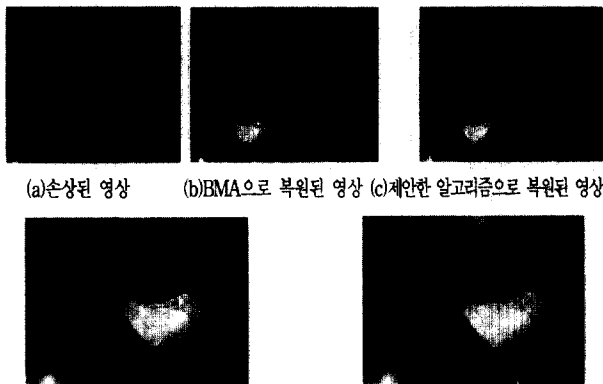


(그림 7) 각 알고리즘의 PSNR 성능 비교

실험은 H.263비디오를 기반으로 4개의 QCIF 영상을 블록 손실율에 따라 실험하였다. 각 영상은 Suzie, Carphone, Mother&Daughter 그리고 Foreman 영상이며 블록 손실율은 5%~20%까지 변화시키면서 실험하였다. 실제 통신 채널상에서의 전송효과를 재현하기 위해 에러는 랜덤하게 주었으며 디코더에서 손실블록의 위치는 알고있다는 것으로 가정

한다. <표 3>은 블록 손실율에 따른 각 결과 영상에 대한 PSNR값을 나타내며, 이때 'No-EC'는 어떤 에러언코드 적용하지 않고 복원한 것을 나타내며, 'TR(Temporal Replacement)'은 모든 손실블록의 움직임벡터에 0을 대입한 것이고, 마지막으로 'Average'와 'Median'은 주변블록의 움직임벡터의 평균과 중간값을 이용하여 복원한 경우이다. (그림 7)는 <표 3>에 나타난 각 알고리즘에 따른 PSNR (Peak-to-Signal Noise Ratio) 성능비교를 그래프로 표현한 것이다.

(그림 8)은 Suzie 영상의 실험 결과 영상이다. (그림 8)의 (a)는 Suzie 영상 중 20%의 블록 손실율이 있는 29번째 프레임이며 (b)와 (c)는 각각 BMA와 제안 방법을 이용하여 복원한 결과 영상이다. 그리고 (그림 8)의 (c)와 (d)는 두 알고리즘을 이용하여 복원했을 때 차이가 나는 부분을 확대한 영상이다. 실험 결과로 보면 제안 알고리즘은 기존 BMA에 비해 PSNR이나 시각적인 화질면에서 우수한 성능을 보임을 알 수 있다.



(a)손상된 영상 (b)BMA로 복원된 영상 (c)제안한 알고리즘으로 복원된 영상

(d) 영상 (b)를 확대한 부분 (e) 영상 (c)를 확대한 부분

(그림 8) 블록 손실율이 20%인 Suzie 영상의 29번째 프레임

5. 결 론

압축된 비디오 스트림을 통신 채널을 통해 전송할 때, 전송 중 발생하는 채널에러에 의해 비디오 스트림은 에러가 발생한 영역의 화질열화뿐만 아니라 발생한 에러가 시·공간적으로 전파되어 그 스트림의 다른 부분으로까지 영향을 미치게 된다. 이러한 채널 에러의 영향을 최소화하는 방법인 에러언코딩법은 인코더의 수정이나 원래의 비디오 스트림 외에 추가 정보가 필요하지 않고 전송대역이나 전송지연에 크게 제약받지 않기 때문에 많이 이용되고 있다. 본 논문에서는 통신 채널을 통해 전송되는 과정중 발생한 에러를 복원하기 위해 손실블록의 주변에 있는 블록들의 움직임벡터 정보를 이용하여 손실블록의 움직임벡터를 복원하는 방법을 제안하였다.

움직이는 객체의 경계 부근에서 손실이 발생할 때 즉 움직임 객체의 경계가 손실블록과 주변 블록에 걸쳐 있는 경

우 기존 방법을 이용할 경우 최적의 움직임 벡터를 찾는 데 어려움이 있다. 이러한 문제점을 해결하기 위해 제안 방법에서는 주변블록들 중에서 어떤 블록들이 동일한 움직임 객체에 포함 되는지를 판단하고 포함되는 블록들과 움직임 객체에 포함되지 않는 블록들을 구별하여 참조하는 동일 움직임영역 추출을 이용한다. 손실블록의 주변블록들 중 일부블록만이 움직임 객체에 포함될 경우 주변블록기법 중에서 동일한 움직임을 갖는 영역을 추출하고 이 영역에 포함된 블록들의 움직임벡터를 이용하여 손실블록의 움직임벡터를 복원하게 되므로 기존의 BMA에 비해 효과적이다.

실험을 통해 제안 알고리즘이 기존의 방법에 비해 PSNR과 시각적인 화질면에서 우수한 성능을 보임을 알 수 있다. 따라서 채널 에러에 의해 손상된 비디오를 효과적으로 복원할 수 있는 방법으로 사용될 수 있을 것으로 보인다.

참 고 문 헌

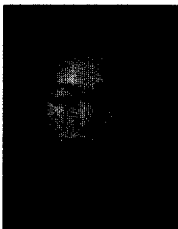
- [1] ITU-T Recommendation H.324/Annex C, "Multimedia telephone terminals over error prone channels," Jan. 1998.
- [2] ITU-T Recommendation H.263, "Video coding for low bit rate communication," March 1996.
- [3] M. Wada, "Selective recovery of video packet loss using error concealment," IEEE J. Select. Areas Commun., Vol.7, 807-814, June 1989.
- [4] E. Ayanoglu, P. Pancha, and A. R. Reibman, "Video transport in wireless ATM," in Proc. ICIP'95, Washington, DC, pp. III 400-403, Oct. 1995.
- [5] E. Ayanoglu, P. Pancha, A. R. Reibman, and S. Talwar, "Forward error control for MPEG-2 video transport in a wireless ATMLAN," in Proc. ICIP'96, Lausanne, Switzerland, pp. II 833-836, Sept. 1996.
- [6] Y. Wang, Q-F. ZHU, "Error Control and Concealment for Video Communication : A Review," proceedings of the IEEE Vol.865, pp.974-997, May 1998.
- [7] M.-J. Chen, L.-G. Chen, and R.-M. Weng, "Error Concealment of Lost Motion Vectors with Overlapped Motion Compensation," IEEE Transaction on Circuits and Systems for Video Technology, Vol.7, No.3, pp.560-563, June 1997.
- [8] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," Proc. ICASSP, Vol.5, pp.417-420, 1993.
- [9] J. Feng, K. T. Lo, and H. Mehrpour, "Error concealment for MPEG video transmissions," IEEE Trans. Consumer Electronics, Vol.43, No.2, pp.183-187, May 1997.
- [10] K. W. Kang, S. H. Lee, and T. Kim, "Recovery of coded video sequences from channel errors," Proc. SPIE Vol.2501, pp.19-27, 1995.

- [11] J. Y. Nam, J. S. Seo, J. S. Kwak, and M. H. Lee, "New block-matching algorithm for motion estimation based predicted direction information," Proc. SPIE Vol.4067, pp.1212-1220, 2000.
- [12] Z. Yu, H. R. Wu, and S. Yu, "Lost motion vector recovery for digital video communication," Proc. SPIE Vol.4067, pp.1250-1260, 2000.
- [13] J. Zhang, J. F. Arnold, M. R. Frater, and M. R. Pickering, "Video Error Concealment Using Decoder Motion Vector Estimation," TENCON '97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE Vol.2, pp.777-780, 1997.



김정현

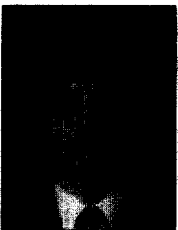
e-mail : jhkim@cs.chonnam.ac.kr
 1999년 전남대학교 전산학과 졸업(학사)
 2001년 전남대학교 대학원 전산학과 졸업(석사)
 2001년~현재 한국전자통신연구원 연구원
 관심분야 : 영상 압축 및 복원, 에러은닉



손남례

e-mail : nrson@cs.chonnam.ac.kr
 1996년 호남대학교 컴퓨터공학과 졸업(학사)
 1999년 전남대학교 대학원 전산학과 졸업(석사)
 2000년~현재 전남대학교 대학원 전산학과 박사과정

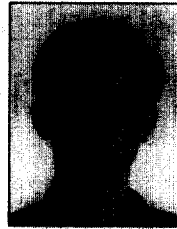
관심분야 : 영상 압축 및 복원, 화상통신, 네트워크



박성찬

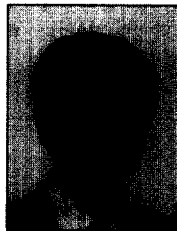
e-mail : schparkl@cs.chonnam.ac.kr
 1995년 동신대학교 전자계산학과 졸업(학사)
 1999년 전남대학교 대학원 전산학과 졸업(석사)
 2000년~현재 전남대학교 대학원 전산학과 박사과정

관심분야 : 비디오코딩, 에러은닉



황성운

e-mail : sohwan@etri.re.kr
 1993년 서울대학교 수학과 졸업(학사)
 1998년 포항공과대학교 정보통신학과(공학석사)
 1994년~1996년 LG-EDS Systems. Inc. 근무
 1998년~현재 한국전자통신연구원 연구원
 관심분야 : 암호학 이론 및 응용, 네트워크 보안, 전자현금 프로토콜



윤기승

e-mail : ksyoon@etri.re.kr
 1984년 부산대학교 조선공학과(학사)
 1988년 NewYork City Univ. 전산학(석사)
 1993년 NewYork City Univ. 전산학(박사)
 1993년~현재 한국전자통신연구원 연구원



손덕주

e-mail : djson@etri.re.kr
 1976년 서울대학교 수학교육과(학사)
 1978년 한국과학기술원 전산학과(석사)
 1978년~현재 한국전자통신연구원(책임연구원, 인터넷서비스연구부장)
 관심분야 : 이동컴퓨팅, 분산시스템의 신뢰성, 데이터베이스 시스템, 네트워크 자료저장 시스템



이귀상

e-mail : gslee@chonnam.ac.kr
 1980년 서울대 공대 전기공학과 졸업(학사)
 1982년 서울대 대학원 전자계산기공학과(석사)
 1982년 금성통신 연구소 근무

1991년 Pennsylvania 주립대학 전산학과 박사
 1984년~현재 전남대 전산학과 교수
 관심분야 : 멀티미디어통신, 영상처리 및 복원, 테스트, 논리합성, VLSI/CAD