

H.264 비디오 표준에서의 칼만 필터 기반의 움직임벡터 복원

고 기 홍[†] · 김 성 환^{**}

요 약

MPEG-2, MPEG-4, H.263, H.264 와 같은 부호화 표준은 비디오 영상을 압축하여 대역폭이 제한된 유/무선 통신 시스템을 통하여 전송한다. 통신 시스템에서 고압축률의 비트스트림은 채널 잡음(channel noise)에 민감하여, 채널 잡음으로 인한 오류가 발생하기 쉽다. 이러한 오류는 수신부에서 디코딩할 때 비디오 영상을 심각하게 왜곡시키게 된다. 본 논문에서는 수신부 단에서 오류를 복원하는 기법(decoder error concealment) 중 손상된 움직임벡터를 복원하는 기법을 제안한다. 본 논문에서는 손실된 움직임벡터를 예측하기 위하여 인접 블록들의 움직임벡터를, 예측필터의 일종인 칼만 필터의 입력 치로 사용하여, 손실된 움직임벡터의 최적 예상치를 만들어 손상된 움직임벡터를 복구하게 된다. H.264 비디오 코딩을 적용한 표준 테스트 영상에 대하여, 손실된 MVD(motion vector difference) 값을 0 으로 대체한 뒤, H.264 비디오 코딩에서 사용하고 있는 기본 움직임벡터 예측만을 사용한 경우와 본 논문에서 제안한 칼만 필터를 사용한 복원기법을 비교하였으며, 복원된 움직임벡터와 원래 움직임벡터 값과의 차이를 나타내는 오차율을 비교한 결과 제안된 기법의 오차율이 평균 0.91 - 1.12 정도의 정확도가 향상된 것을 확인할 수 있다.

키워드: 송신부 오류 은닉, 수신부 오류 은닉, 칼만 필터, 움직임벡터 복구, 움직임벡터 예측

Kalman filter based Motion Vector Recovery for H.264

Ki-Hong Ko[†] · Seong-Wan Kim^{**}

ABSTRACT

Video coding standards such as MPEG-2, MPEG-4, H.263, and H.264 transmit a compressed video data using wired/wireless communication line with limited bandwidth. Because highly compressed bit-streams is likely to fragile to error from channel noise, video is damaged by error. There have been many research works on error concealment techniques, which recover transmission errors at decoder side [1, 2]. We designed an error concealment technique for lost motion vectors of H.264 video coding. In this paper, we propose a Kalman filter based motion vector recovery scheme, and experimented with standard video sequences. The experimental results show that our scheme restores original motion vector with more precision of 0.91 - 1.12 on average over conventional H.264 decoding with no error recovery.

Key Words : Forward Error Concealment, Decoder Error Concealment, Kalman Filter, Motion Vector Recovery, Motion Vector Prediction

1. 서 론

MPEG-2, MPEG-4, H.263, H.264와 같은 부호화 표준은 비디오 영상을 압축하여 대역폭이 제한된 유/무선 통신 시스템을 통하여 전송한다. 통신 시스템에서 고압축률의 비트스트림은 채널 잡음(channel noise)에 의하여 오류가 발생하기 쉽다. 전송 오류는 수신부에서 디코딩할 때 비디오 영상을 심각하게 왜곡시키며, 이러한 전송 오류를 처리하기 위한 다양한 오류 보정 기법들이 제안되었다[1, 2]. 이러한 기법들은 (1) 송신부에서 전송 오류를 예방하는 기법(forward error concealment), (2) 수신부에서 전송 오류를 처리하는 기법

(decoder error concealment)으로 구분할 수 있다.

송신부 단의 오류 복원(forward error concealment) 기법은 인코딩된 비트 스트림의 강인함을 향상시켜 전송 오류에 강인하도록 하는 기법으로, 채널 코딩(channel coding) 영역에서 수행되는 기법과 소스코딩(source coding) 영역에서 수행되는 기법으로 나눌 수 있다. 채널코딩 영역에서 수행되는 기법에는 대표적으로 FEC(forward error correction)[3], TLC(transport-level control)[4, 5] 등이 있으며, 소스코딩 영역에서 수행되는 기법에는 강인한 엔트로피코딩(robust entropy coding) [6, 7, 8, 9, 10, 11] 등이 있다. 소스코딩과 채널코딩 영역에서 동시에 이루어지는 방법으로는 전송 우선순위를 이용한 계층부호화기법(layered coding)[12], MDC(multiple-description coding) 기법이 있다 [13].

수신부 단의 오류 복원(decoder error concealment) 기법

[†] 정 회 원 : 서울시립대학교 컴퓨터통계학과(이학박사)

^{**} 정 회 원 : 서울시립대학교 컴퓨터과학부 교수

논문접수 : 2007년 7월 24일, 심사완료 : 2007년 9월 17일

은 송신부에서 보낸 데이터가 전송 오류에 의하여 손실되었을 때, 송신부의 부가적인 처리 없이 수신부에서 손실된 정보를 추정하거나, 복원하는 기법이다. 손실된 정보에 따라, 영상 정보의 화소 값이 손실되었을 경우와 움직임벡터 정보가 손실되었을 경우로 나눈다. 손실된 정보가 영상 정보의 화소 값인 경우, 움직임보간 예측기법 (motion-compensated temporal prediction)[14, 15], POCS (projection onto convex sets)들의 방법으로 복원한다[16]. 손실된 정보가 움직임벡터 일 경우, 움직임벡터 복원 방법을 사용한다[17].

본 논문에서는 수신부 단의 오류 복원(decoder error concealment) 기법에 사용하기 위해, 움직임벡터가 손상된 경우에 대한 움직임벡터 복원 기법을 제안한다. 제안한 기법은 인접한 블록들의 움직임벡터를 입력으로 하는 칼만 필터를 이용하여 움직임벡터를 예측하게 되며[18], 실제 H.264로 코딩된 바이트스트림에 대해서, 전송 상의 손실이 발생한 경우에 대해, 표준 비디오 테스트영상에 대해 실험한 결과를 보인다. 표준 테스트 영상에 대하여, H.264 비디오 표준에서 사용하고 있는 기본 움직임벡터 예측만을 사용한 경우와 제안된 칼만 필터를 사용하여 복원된 움직임벡터를 원래 움직임벡터 값과의 차이를 나타내는 오차율을 비교한 결과 제안된 기법의 오차율이 평균적으로 0.91 - 1.12 정도 향상된 결과를 나타냈다.

2. H.264 비디오 코딩표준의 움직임벡터 예측과 칼만 필터 기반의 예측기법

H.264 부호화 표준은 이전의 부호화 표준들인 MPEG-1, MPEG-2, MPEG-4, H.263과 유사하게 참조 프레임을 이용하여 움직임벡터 추정 (motion vector estimation)을 이용하며, 블록 간의 움직임벡터 상관관계를 이용하여 움직임벡터 예측 (motion vector prediction)을 사용한다. 이 절에서는 H.264 에서의 움직임벡터 예측에 대하여 기술한다. (그림 1)은 H.264에서 움직임벡터의 다양한 블록들을 보여준다. H.264는 압축 효율을 높이기 위하여 INTRA 모드에서는, 프레임 내 인접블록 사이의 예측모드를 수행하고, INTER 모드에서는 인접한 프

레이م 간의 움직임벡터 추정을 통하여 움직임벡터를 구한 후, 역시 압축의 효율을 높이기 위한 예측모드를 수행한다. (그림 1)에서처럼 H.264에서 움직임벡터는 16×16, 8×16, 16×8, 8×8, 4×8, 8×4, 4×4 등의 다양한 블록 크기를 갖는다[19].

다양한 블록 크기를 갖는 움직임벡터는 인접 블록의 다양한 크기로 인하여 다양한 움직임벡터 예측의 방법을 이용하게 된다 [19]. (그림 2)는 H.264의 다양한 움직임벡터 예측 기법을 보여준다. E가 현재 블록이며, A, B, C는 이웃 블록이다. (그림 2)와 같이, 현재 블록 E에서 움직임벡터 예측을 위한 MVP 계산은 이웃 블록들의 다양한 크기에 따라 다르게 구하게 된다.

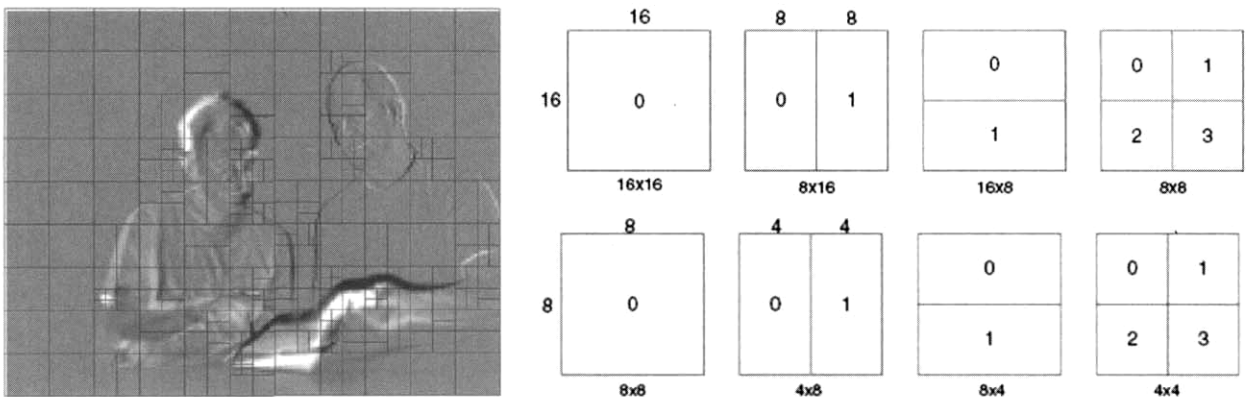
또한 H.264는 인코더에서 디코더로 MVD (motion vector difference)가 전송되며, 디코딩 시 움직임벡터 예측을 이용하여 MVP 값을 얻는다. 그리고 MVD와 MVP를 더하여 움직임벡터 값을 생성한다. 그러므로 손실된 움직임벡터 정보는 MVD이며, MVD 값의 손실 또는 변경은 (그림 3)에서 움직임벡터 값을 변경시키게 된다.

본 논문에서, 움직임벡터 복원에 이용하기 위한 칼만 필터는 이산시간에서, 관측치 (measurement) t 로부터 현재 상태 (state) s 를 추정하는 재귀적 처리과정이다. 칼만 방정식은 시간 단계 (time step) k 에서 관측치를 t_k , 상태를 s_k 라고 하면, 식 (1)로 표현된다. 식 (1)에서 s_k 는 $N \times 1$ 의 상태 (state) 벡터이며, A 는 $N \times N$ 행렬의 상태이동행렬 (state transition matrix)이고, w_k 는 분포 $N(0, \sigma_w^2)$ 를 갖는 $N \times 1$ 의 시스템 잡음 (process noise) 벡터이다. t_k 는 $M \times 1$ 의 관측치 벡터이며, H 는 $M \times M$ 의 관측치 행렬이고, r_k 는 분포 $N(0, \sigma_r^2)$ 을 갖는 $M \times 1$ 관측치 잡음 (measurement noise) 벡터이다. w_k 와 r_k 는 서로 독립이고, 정규확률분포 (normal probability distributions)를 따른다고 가정한다[18].

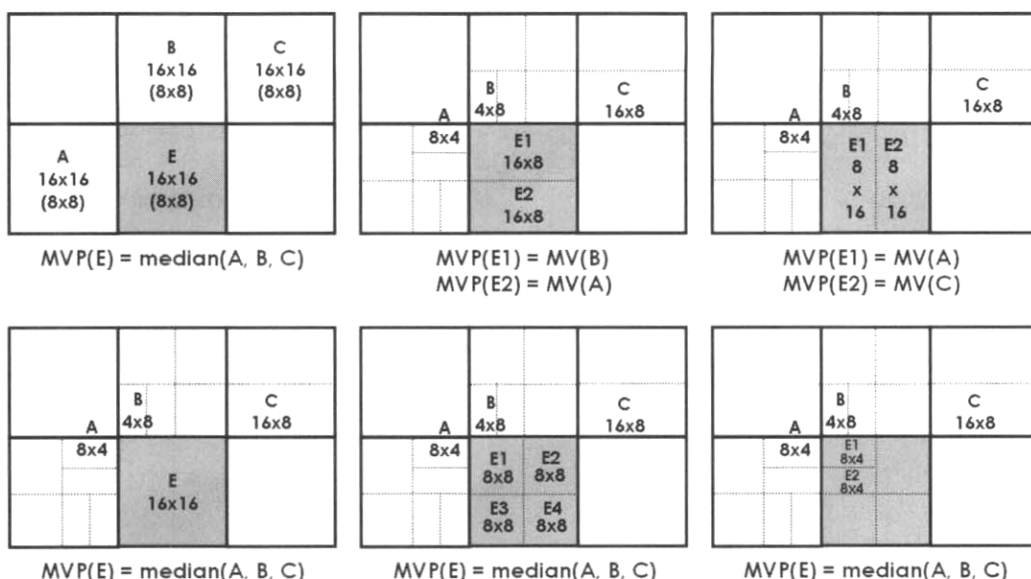
$$s_k = As_{k-1} + w_{k-1} \quad \text{식 (1)}$$

$$t_k = Hs_k + r_k$$

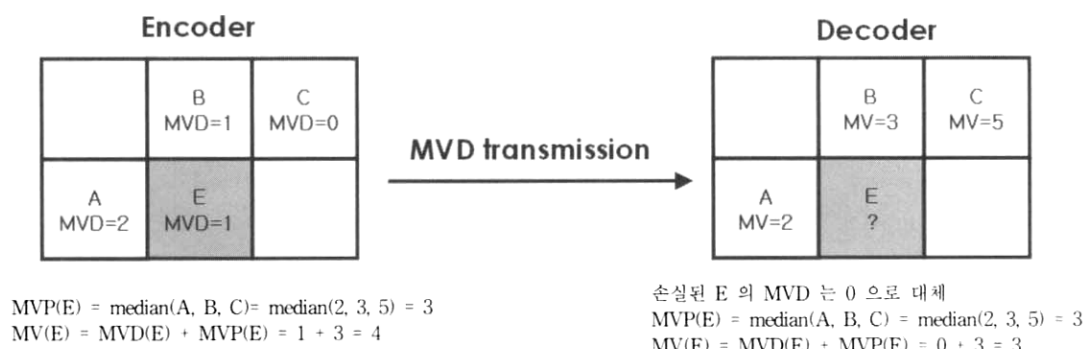
이제 시간 단계 (time step) k 에서 \hat{s}_k 를 사전 상태 추정치 (priors state estimate), \hat{s}_k 를 사후 상태 추정치 (posteriori



(그림 1) H.264 움직임예측단위: 다양한 크기의 서브 블록



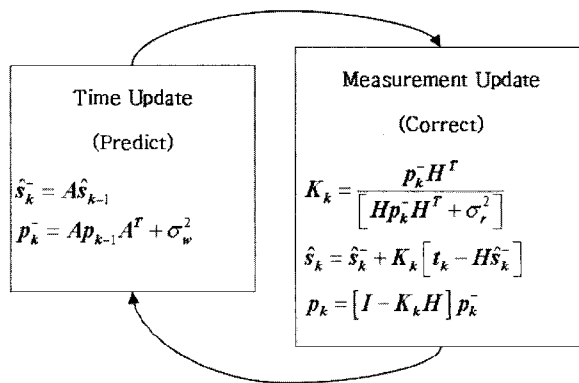
(그림 2) H.264에서의 움직임벡터 예측



(그림 3) 동영상디코더에서 MVD(E) 손실 시 얻게 되는 움직임벡터 MV(E)

state estimate)라고 정의하면, 사전 추정 오류 (priori estimate error)는 $e_k^- \equiv s_k - \hat{s}_k^-$ 이고, 사후 추정 오류 (posteriori estimate error)는 $e_k \equiv s_k - \hat{s}_k$ 이다. 이것을 이용하면 사전 추정 오류 공분산(priori estimate error covariance)은 $p_k^- = E[e_k^- e_k^{-T}]$ 이고, 사후 추정 오류 공분산(posteriori estimate error covariance)은 $p_k = E[e_k e_k^T]$ 이다. 칼만 필터를 이용하여 구하고자 하는 것은 사후 상태 추정치이다. 그런데, 사후 상태 추정치 \hat{s}_k 는 사전 상태 추정치 \hat{s}_k^- 와 잔차의 선형 조합으로 표현할 수 있으며, 잔차는 관측치 t_k 와 관측 예측치(measurement prediction) $H\hat{s}_k^-$ 사이의 차이를 의미한다. 따라서 $\hat{s}_k = \hat{s}_k^- + K(t_k - H\hat{s}_k^-)$ 이다. 여기서 K 는 칼만 이득 (gain) 이다. K 는 사후 추정 오류 공분산을 최소화 시키는 값이어야 한다. 사후 상태 추정치 식 $\hat{s}_k = \hat{s}_k^- + K(t_k - H\hat{s}_k^-)$ 은 사전 상태를 알아야 풀 수 있는 재귀적인 식이다. 칼만 필터 계산은 (그림 4)에서 보는 바와 같이, 사전 상태와 사후 상태를 계산하는 2개 부분으로 구성하며, 시간 갱신 부분 (time update)은 사전 상태 값을 예측하는 과정이고, 관측치 갱신 부분 (measurement update)은 사후 상태 값을 수

정하는 과정이다. (그림 4)의 칼만 이득 K_k 에서, 만약 σ_r^2 이 0에 근접한 값이라면, $K_k \approx H^{-1}$ 이 된다. 이 값을 사후 상태 추정치 \hat{s}_k 에 대입하면, $\hat{s}_k = H^{-1}t_k = K_k t_k$ 가 된다. 이는 관측치 만으로도 사후 상태를 추정할 수 있기 때문에, 관측치의 신뢰도가 커짐을 의미한다. 즉 관측치의 값이 예측 값이 된다. 만약



(그림 4) 칼만 필터 사이클

p_k 가 0에 근접한 값이라면, 시스템 잡음이 없음을 의미하고 이는 예측을 위한 칼만 이득이 없게 된다. 그러므로 $K_k \approx 0$ 이 된다. 이 값을 사후 상태 추정치 \hat{s}_k 에 대입하면, $\hat{s}_k = \hat{s}_k$ 가 된다. 이는 사전 상태 추정치로 사후 상태를 추정할 수 있기 때문에 사전 상태 추정치의 신뢰도가 커짐을 의미한다. 이것은 관측치의 신뢰도가 상대적으로 떨어져, 관측치 값에 상관없이 시스템의 상태 이동 행렬과 사전 상태에 의하여 사후 상태 추정치를 얻을 수 있음을 의미 한다 [18].

3. 칼만 필터 기반의 손상된 움직임벡터 복원기법

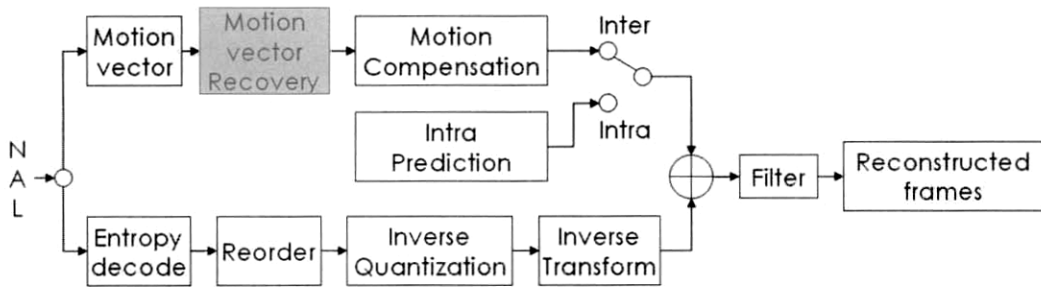
움직임벡터 정보가 손실이 발생한 경우, 칼만 필터를 사용하여 손실된 움직임벡터를 복원하는 시스템은, (그림 5)와 같이, H.264 비디오코딩 표준을 준수하는 비디오코더를 수정하였으며, 네트워크에서 전송된 NAL(network abstraction layer) 스트림이 디코드 되고, 수신된 움직임 블록 중 손실된 움직임벡터를 복원하게 된다.

H.264는 인코더에서 디코더로 MVD가 전송되며, 디코딩 시 움직임벡터 예측을 이용하여 MVP 값을 얻는 것을 2 절에서 살펴보았다. 또한 본 연구에서는 움직임벡터를 복원하기 위하여 2 절에서 살펴본 칼만 필터 모델을 이용한다. 전송 시 손실된 블록의 움직임벡터는 MVD가 손실된 상태이

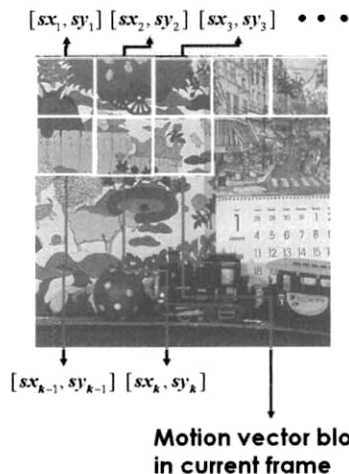
다. 그러므로 칼만 필터에서는 MVD 값을 추정하게 된다. 따라서 칼만 필터에서 현재 상태는 현재 블록의 MVD 값이다. (그림 6)에서처럼 프레임 내에서 블록 위치 (x_k, y_k) 인 현재 블록의 MVD는 $s_k = [sx_k, sy_k]$ 으로 표현한다. 현재 블록의 움직임벡터를 구하기 위해서는 이전 블록의 움직임벡터가 필요하다. 이전 블록의 움직임벡터는 $s_{k-1} = [sx_{k-1}, sy_{k-1}]$ 로 표현하며, (그림 6)과 같다.

실제로 칼만 필터를 계산할 때, 계산을 간편하게 하기 위해서는 x 성분과 y 성분으로 분리하여 sx_k 는 x 성분에 대한 MVD 값이고, sy_k 는 y 성분에 대한 MVD 값이다. x, y 성분의 이전 블록의 MVD는 각각 sx_{k-1}, sy_{k-1} 이다. 이를 수식으로 정리하면, 식 (2)와 같다. 여기서 wx_k, wy_k 은 각각 x, y 성분에 대한 시스템 잡음 벡터이다. 관측치 t 는 식 (3)과 같이 나타내며, 각각은 x, y 성분을 나타낸다. 역시 rx_k, ry_k 는 각각 x 와 y 성분에 대한 관측치 잡음 벡터이다.

$$\begin{aligned}
 sx_k &= a_1 sx_{k-1} + wx_{k-1} & sy_k &= b_1 sy_{k-1} + wy_{k-1} & \text{식 (2)} \\
 wx_{k-1} &\sim N(0, \sigma_w^2) & wy_{k-1} &\sim N(0, \sigma_w^2) \\
 \sigma_u^2 &= E[(wx_k)^2] = E[(wy_k)^2] \\
 a_1, b_1 &: \text{an state transition constant, } a_1 = b_1 = 0.98 \\
 w &: \text{process noise with } N(0, \sigma_w^2), \sigma_w^2 = 0.75
 \end{aligned}$$

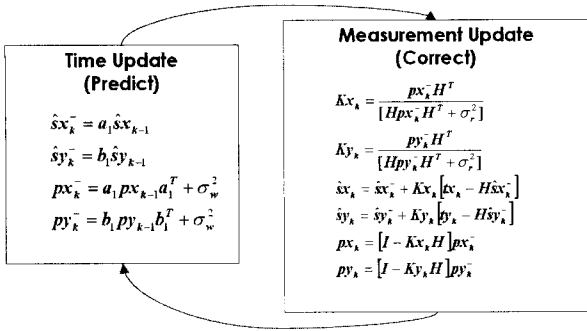


(그림 5) 칼만 필터 기반의 움직임벡터 복원 시스템



$$\begin{aligned}
 \text{A set of MV: } s &= \{[sx_k, sy_k] | k = 1, 2, \dots, n\} \\
 MV(sx_k) &= MVD(sx_k) + MVP(sx_k) \\
 MV(sy_k) &= MVD(sy_k) + MVP(sy_k)
 \end{aligned}$$

(그림 6) 칼만 필터 적용을 위한 MVD 블록의 구성



(그림 7) MVD 추정을 위한 칼만 필터 사이클

$$tx_k = sx_k + rx_k \quad ty_k = sy_k + ry_k \quad \text{식 (3)}$$

$$rx_k \sim N(0, \sigma_r^2) \quad ry_k \sim N(0, \sigma_r^2)$$

$$\sigma_r^2 = E[(rx_k)^2] = E[(ry_k)^2]$$

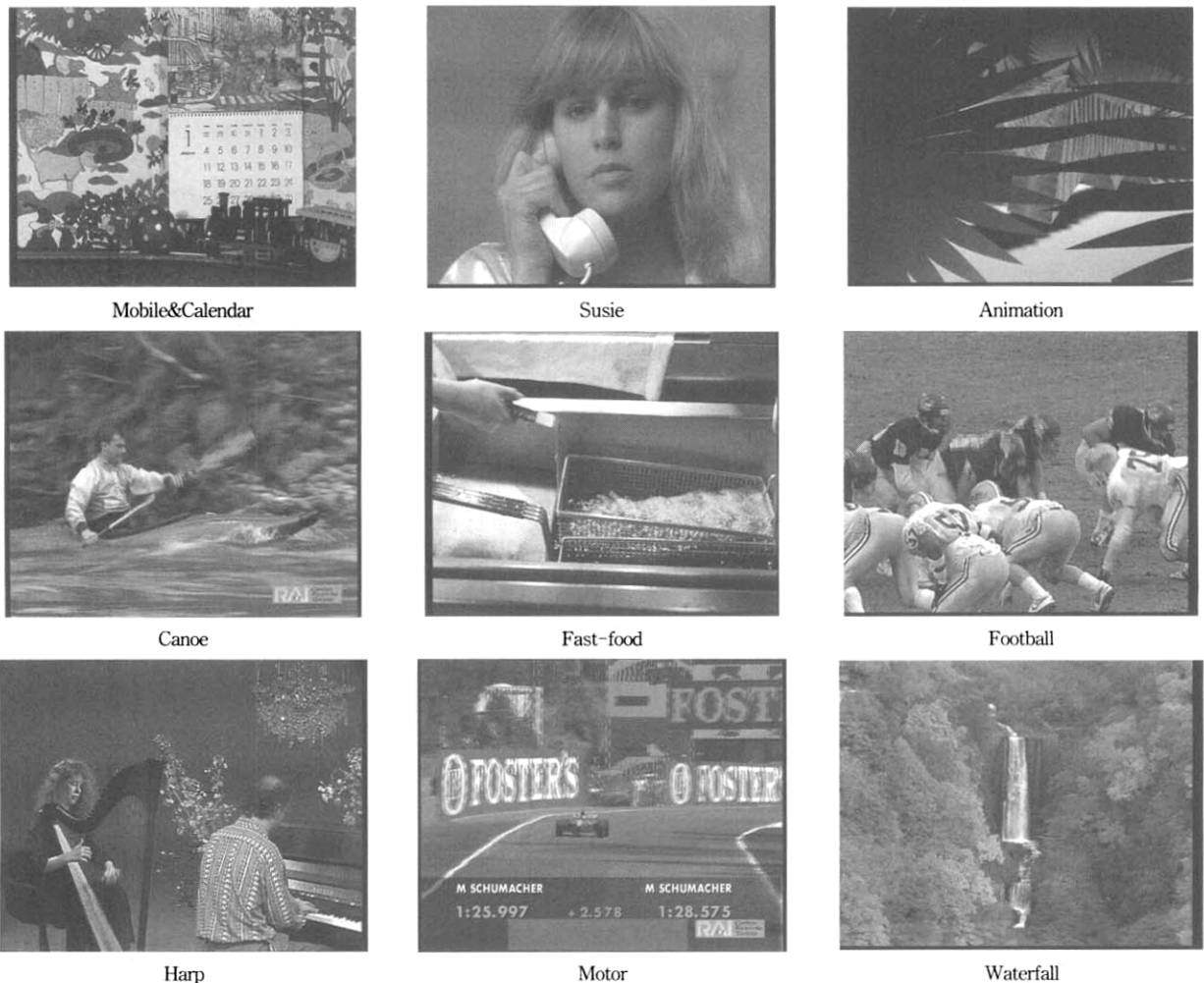
r : measurement noise with $N(0, \sigma_r^2)$, $\sigma_r^2 = 0.25$

이렇게 구해진 블록의 상태 정보를 이용하여 (그림 7)과 같은 MVD 추정을 위한 칼만 필터 사이클을 구성한다.

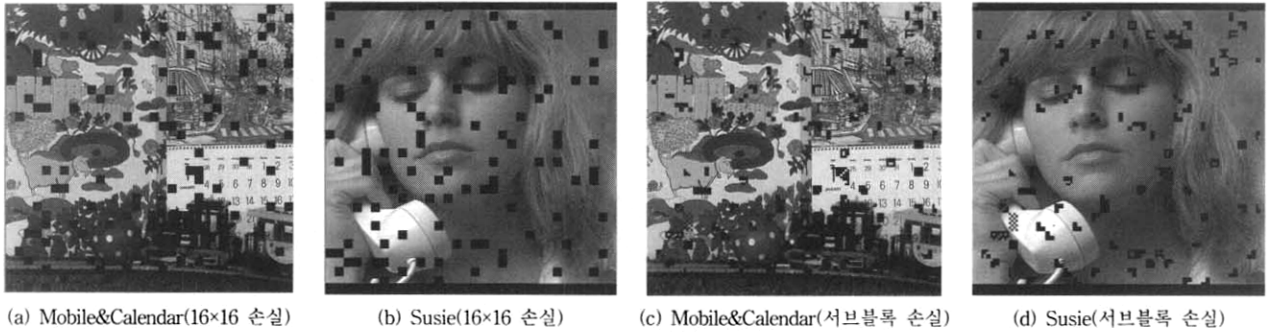
4. 실험 결과 및 분석

칼만 필터 기반의 움직임벡터 복원 성능을 비교하기 위하여, H.264 비디오코딩 표준에서 사용하고 있는 움직임벡터 복원기법과 비교하였으며, (그림 8)과 같은, H.264 비디오코딩 표준에서 사용하고 있는 표준영상자료를 사용하였다. H.264 비디오코딩은, I와 P 프레임만이 사용되었다. 실제 네트워크상에서의 다양한 에러형태를 시뮬레이션하기 위해, 다양한 에러 프로파일 (error profile)에 대해 실험하였으며, 대개의 비디오 전송이 인터리빙 기법을 이용하기 때문에, 비디오 전송에서 흔히 나타나는 폭주에러(burst error)의 경우에도 무작위 에러(random error)의 형태로 나타난다고 가정하였다. (그림 9)는 움직임벡터가 손실된 블록을 나타낸다. (그림 9(a)(b))는 손실 블록이 16x16인 경우에 한하여 발생한 경우이고, (그림 9(c)(d))는 다양한 서브블록 별로 손실이 발생한 경우를 나타낸다. H.264는 움직임벡터가 서브블록별로 생성되므로, 각각의 서브블록의 움직임벡터에 대한 손실이 발생한 경우에 대하여도 비교할 필요가 있다.

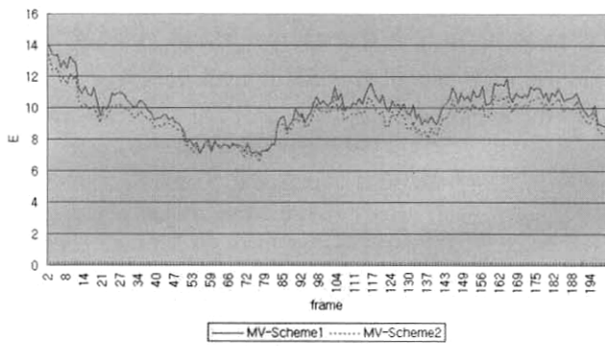
본 연구에서는, H.264 비디오코딩표준에서 사용하고 있는



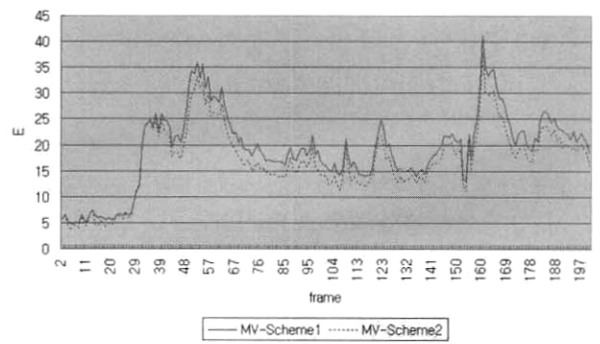
(그림 8) 실험에 이용된 표준 영상 자료



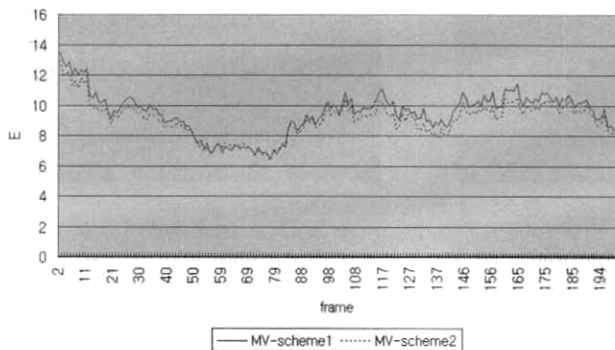
(그림 9) MVD 손실 프로파일 시뮬레이션



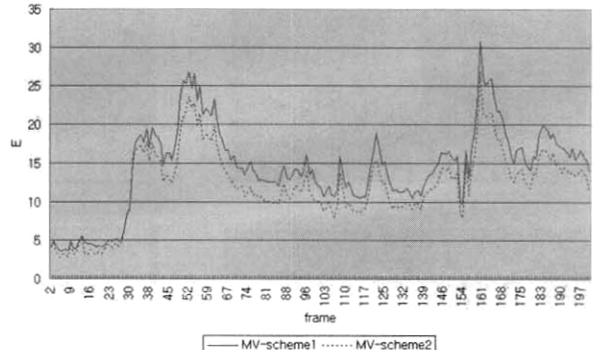
(a) Mobile&Calendar: 16x16 손실 블록



(b) Susie: 16x16 손실 블록



(c) Mobile&Calendar: 서브블록 손실 블록



(d) Susie: 서브블록 손실 블록

(그림 10) 방법 1과 방법 2의 움직임벡터 차이 비교

움직임벡터 복원기법과 비교하고 있으며, H.264 비디오코딩 표준에서는, 손실된 움직임벡터 MVD 값을 0으로 대체하고 H.264의 움직임벡터 예측 기법을 이용한 방법 (방법 1)을 사용하고 있다. 본 연구에서 비교하고 있는 움직임벡터 복원기법은 칼만 필터를 사용하여, 이전 움직임벡터로부터, 현재 움직임벡터 복원을 위한 최적치를 찾는 과정이며, 방법 2로 비교한다.

실험의 비교를 위해 사용된 방법은 각 블록에 대한 실제 움직임벡터 값과 방법 1의 차이 값과 방법 2의 차이 값을 구한 뒤, 한 프레임에 대한 평균값을 구하여 움직임벡터의 차이 값을 비교한다. 비교를 위하여 식 (4)를 사용하였다. 식 (4)에서 밑 첨자 $real-x$, $real-y$ 는 한 블록에 대한 x , y 방향에 대한 실제 움직임벡터 값을 나타내고, $scheme-x$,

$scheme-y$ 는 방법 1과 방법 2의 방법으로 복원된 x , y 방향에 대한 움직임벡터 값이다. 분모 N 은 움직임벡터를 갖는 블록의 개수이다.

$$E = \sqrt{\frac{(MV_{real-x} - MV_{scheme-x})^2 + (MV_{real-x} - MV_{scheme-y})^2}{N}}$$

(그림 10)은 Mobile&Calendar 영상과 Susie 영상에 대하여 각 프레임 당 움직임벡터의 차이 값을 그래프로 표현한 것이다. 오차율 E 는 0 값에 가까울수록 원 MV 값에 근접한 값을 갖게 되며, 방법 2 기법이 방법 1 기법보다 오차율이 작음을 알 수 있다. 그러므로 방법 1의 기법보다 제안한 기법인 방법 2가 좀 더 정확한 예측을 하는 것을 볼 수 있다.

<표 1>은 표준 영상에 대하여 E 값을 계산한 결과이다.

〈표 1〉 표준 영상에 대한 오차율 비교

영상	평균 오차율	16×16 손실 블록		다양한 서브 블록 손실	
		방법 1	방법 2	방법 1	방법 2
Mobile&Calendar	$E_{Mean\ of\ Total}$	10.0	9.4	9.5	9.1
Susie	$E_{Mean\ of\ Total}$	19.2	17.1	14.4	12.1
Animation	$E_{Mean\ of\ Total}$	11.4	9.5	8.1	6.9
Canoe	$E_{Mean\ of\ Total}$	7.1	6.6	6.8	6.4
Fast-food	$E_{Mean\ of\ Total}$	18.9	17.1	14.7	13.0
Football	$E_{Mean\ of\ Total}$	8.1	7.4	9.3	8.7
Harp	$E_{Mean\ of\ Total}$	12.4	11.4	11.6	10.7
Motor	$E_{Mean\ of\ Total}$	12.5	11.3	12.4	12.1
Waterfall	$E_{Mean\ of\ Total}$	7.4	7.1	8.2	7.8

$E_{Mean\ of\ Total}$ 값은 각각의 프레임에 대한 오차율을 전체 200 프레임에 대하여 평균을 한 값이다. 역시 마찬가지로 제안된 기법인 방법 2가 좀 더 좋은 결과를 얻을 수 있다. 16×16 블록의 손실이나, 서브블록들에 대한 손실 마찬가지로 제안된 기법이 우수함을 또한 알 수 있다.

6. 결 론

본 논문은 통신 시스템에서 영상 데이터 전송 시 손실된 움직임벡터를 복원하는 기법을 제안한다. 통신 시스템에서 고압축률의 비트스트림은 채널 잡음에 민감하여, 오류가 발생하기 쉽다. 이러한 오류는 수신부에서 디코딩할 때 비디오 영상을 심각하게 왜곡시키게 된다. 본 논문에서 제안된 기법은 손실된 움직임벡터를 예측하기 위하여, 예측 필터인 칼만 필터에 인접 블록들의 움직임벡터를 관측치로 사용하여, 손실된 움직임벡터의 최적 예상치를 추정하여 손상된 움직임벡터를 복구하게 된다. 본 논문에서 제안한 기법은, H.264 비디오코딩 표준에서 사용하고 있는 움직임벡터 복원 기법과 비교하였으며, H.264 비디오 코딩을 적용한 표준 테스트 영상에 에러프로파일을 적용하여, 손실된 움직임벡터를 복원하는 실험을 수행하였으며, 실제 움직임벡터 값과의 차이를 나타내는 오차율을 비교하였다. 비교 결과, 제안된 기법이 H.264 움직임벡터 복원기법에 비해, 0.91 - 1.12 정도 향상된 정확도를 나타내었다. 또한 움직임 블록이 16×16의 고정된 크기인 경우와 다양한 서브 블록으로 나누어지는 경우 각각에 대하여 실험한 결과 제안된 기법인 방법 2의 예측 정확도가 높아지므로, 서브블록으로 나누지 않고 비디오 코딩을 적용하는 다른 영상 압축 기법인 MPEG-1, MPEG-2 등의 비디오코딩표준에도 본 기법을 적용할 수 있다.

참 고 문 헌

- [1] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," IEEE Signal Proc. Magazine, Vol.17, pp.61-82, July 2000.
- [2] Y. Wang and Q. F. Zhu, "Error control and concealment for video communication: a review," Proc. IEEE., Vol.86, pp.974-997, May 1998.
- [3] K. N. Ngan, C. W. Yap, K. T. Tan, "Video coding for wireless communication systems," Marcel Dekker, 2001, pp. 158-161.
- [4] T. Turetti and C. Huitema, "RTP payload format for H.261 video streams," IETF RFC 2032, Oct. 1996.
- [5] C. Zhu, "RTP payload format for H.263 video streams," IETF draft, Mar. 1997.
- [6] J. C. Maxted and J. P. Robinson, "Error recovery for variable length codes," IEEE Trans. Info. Theory, vol.IT-31, pp.794-801, 1985.
- [7] T. J. Ferguson and J. H. Ranowitz, "Self-synchronizing Huffman codes," IEEE Trans. Info. Theory, vol.IT-30, pp.687-693, July 1984.
- [8] P. G. Neumann, "Self-synchronizing sequential coding with low redundancy," Bell Syst. Tech. J., Vol.50, pp.951-981, Mar. 1971.
- [9] S.-M. Lei and M.-T. Sun, "An entropy coding system for digital HDTV applications," IEEE Trans. Circuits Syst. Video Tech., Vol.1, pp.147-154, Mar. 1991.
- [10] R. Koenen, "Overview of the MPEG-4 standard," ISO/IEC JTC1/SC29/WG11 N1730, July 1997.
- [11] T. Ebrahimi, "MPEG-4 video verification model version 8.0," ISO/IEC JTC1/SC29/WG11 N1796, July 1997.
- [12] M. Ghanbari, "Two-layer coding of video signals for VBR networks," IEEE J. Select. Areas Comm., Vol.7, pp.801-806, June 1989.
- [13] V. A. Vaishampayan, "Design of multiple description scalar quantizers," IEEE Trans. Info. Theory, Vol.39, pp.821-834, May 1993.
- [14] M.-J. Chen, L.-G. Chen, and R.-M. Weng, "Error concealment of lost motion vector with overlapped motion compensation," IEEE Trans. Cir. Sys. Video Tech., Vol.7, No.3, pp.560-563, Jun. 1997.

- [15] M. Ghanbari, "Cell-loss concealment in ATM video codes," IEEE Trans. Cir. Sys. Video Tech., Vol.3, pp.238-247.
- [16] H. Sun and W. Kwok, "Concealment of damaged block transform coded images using projections onto convex sets," IEEE Trans. Image Proc., Vol.4, pp.470-477, Apr. 1995.
- [17] D-W. Kim, E-K. Kang, J-T. Kim, Y-I. Yoon, and J-S. Choi, "Motion vector recovery by surrounding region matching based on gradient difference," Lect. Notes in Computer Science, Springer-Verlag, Berlin Heidelberg, Vol.3320, pp. 106-110, 2004.
- [18] G. Welch and G. Bishop, "An introduction to the Kalman filters," available in <http://www.cs.unc.edu/~welch/kalman/index.html>.
- [19] I. E. G. Richardson, H.264 and MPEG-4 video compression, John Wiley & Sons, 2003, pp.159-176.

고 기 흥



e-mail : jedigo@venus.uos.ac.kr

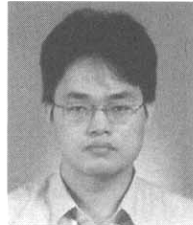
2001년 서울시립대학교 전산통계학과학과
(학사)

2003년 서울시립대학교 대학원 전산통계학과
(이학석사)

2007년 서울시립대학교 대학원 컴퓨터통계학과
(이학박사)

관심분야: 동영상 코덱, 영상처리, 오류은닉, 영상복원

김 성 환



e-mail : swkim7@uos.ac.kr

1999년 한국과학기술원 전자전산학과(박사)

1996년~2000년 LG전자 이동통신연구소
IMT-2000 SW1실 팀장

2000년~2001년 Cisco Systems MWBU
책임연구원

2002년~현재 서울시립대학교 컴퓨터과학부 교수

관심분야: 오류은닉, 영상복원, 게임공학