

# 정교한 움직임 유사도 기반 H.264/AVC에서의 시방향 에러 은닉 알고리즘

준회원 민 창 기\*, 정회원 진 순 중\*\*, 준회원 오 형 철\*, 정회원 정 제 창\*

## Fine Motion Similarity Based Temporal Error Concealment Algorithm for H.264/AVC

Changki Min\* *Associate Member*, Soonjong Jin\*\* *Regular Member*,  
Hyeongchul Oh\* *Associate Member*, Jechang Jeong\* *Regular Member*

### 요 약

통신 네트워크를 통하여 압축된 영상을 전송하는 동안에 에러가 발생한 경우, 에러 전파 방지와 화질 개선을 위해 에러 은닉 방법이 필요하다. 본 논문은 H.264/AVC을 위한 효율적인 시방향 에러 은닉 알고리즘을 제안한다. 제안하는 알고리즘은 화면간 부호화된 프레임에서 에러가 발생했을 경우, 손실된 매크로 블록의 움직임 벡터와 주변 움직임 벡터간의 움직임 유사도가 높은 성질을 이용하여 손실된 매크로블록의 움직임 벡터 후보군을 선택한다. 다음으로 손실 없이 복호된 화소가 가지고 있는 높은 신뢰성을 이용한 가중치 외부 경계 정합 알고리즘을 이용하여 선택된 움직임 벡터 후보군들 중에서 은닉에 적합한 움직임 벡터를 선택한다. 실험 결과 제안하는 알고리즘은 H.264/AVC에 사용되고 있는 이전의 방법과 비교하여 최대 3.02dB의 PSNR 성능 향상을 가져 온다.

**Key Words** : Error concealment, H.264, MV recovery, OBMA, video transmission

### ABSTRACT

If an error occurs during transmitting compressed video through the communication network, error concealment is needed to prevent error propagation and to improve quality. In this paper, we propose the high performance temporal error concealment algorithm for H.264/AVC. The proposed algorithm uses the property that the motion vector of the error macroblock between the motion vectors of the neighboring macroblock have the high similarity to select a group of candidate motion vectors, when an error occurs in the inter-coded frame. Next, weighting overlapped boundary matching algorithm using the credibility of decoded pixels without errors selects the best candidate motion vector among a group of candidate motion vectors. The experimental results show that the proposed algorithm improves PSNR up to 3.02 dB compared with the previous algorithms of H.264/AVC.

### 1. 서 론

최근 통신 시스템의 발달로 통신 네트워크를 통한

영상 데이터의 전송이 많이 이루어지고 있다. 영상 데이터의 전송 비용을 줄이기 위해서 영상 압축을 이용한다. H.264/AVC는 ITU-T와 ISO/IEC의 영상 압축

※ 본 연구는 서울시 산학연 협력사업(10570)으로 구축된 서울 미래형 콘텐츠 컨버전스 클러스터 지원으로 수행되었습니다.

※ 이 논문은 2010년도 두뇌한국21사업에 의하여 지원되었음.

\* 한양대학교 전자컴퓨터통신공학과 영상통신및신호처리 연구실(mck429@hotmail.com), \*\* LG전자 CTO DTV 연구소 (soonjong.jin@lge.com)  
논문번호 : KICS2010-04-175, 접수일자 : 2010년 04월 16일, 최종논문접수일자 : 2010년 9월 9일

표준으로 높은 압축 효율과 통신에 친화적인 영상 부호화를 목적으로 한다<sup>[1]</sup>. 압축된 영상 데이터는 통신 네트워크를 통한 전송 시에 채널상의 문제로 발생하는 에러에 민감하다. 이러한 에러는 복호된 영상의 화질을 저하시키는 원인이 된다. 또한 연속하는 블록이나 프레임으로 에러가 전파될 수 있다. 이러한 이유로 에러의 영향을 경감시키기 위해서 에러 내성 기법<sup>[2]</sup>, 에러 은닉 기법<sup>[3]</sup>과 같은 에러 제어 방법이 요구된다. 에러 내성 기법은 손실된 데이터를 다시 전송하는 ARQ (Automatic Repeat Request) 와 오류 정정 부호를 사용하여 에러에 강인한 스트림을 전송하는 FEC (Forward Error Correction) 가 있다. ARQ는 전송 채널에 심각한 과부하를 일으킬 수 있으며, FEC는 추가적인 전송 대역폭을 요구한다. 반면에 에러 은닉 방법은 복호화 단에서 독립적으로 구현될 수 있으며, 전송 대역폭의 증가나 부호화기의 수정을 필요로 하지 않기 때문에 채널 에러를 극복하는데 효과적이다.

에러 은닉 방법은 영상이 가진 성질 중에 인접한 화소 간의 상관도와 프레임간 상관도가 높다는 성질을 이용하여 손실된 정보를 복원한다. 에러 은닉 방법은 일반적으로 공간적 에러 은닉 방법과 시방향 에러 은닉 방법 두 가지로 분류된다. 공간적 에러 은닉 방법은 공간적 보간법을 사용하는데, 공간적 보간법은 인접한 화소 간의 높은 상관도를 이용하여 매크로 블록을 복원한다. 하지만 매크로 블록의 크기는 16×16으로 화소 간의 거리가 점점 멀어져 상관도가 낮아진다. 이런 경우 원치 않는 블록왜곡 (block artifact)이 나타난다. 반면에 시방향 에러 은닉 방법은 프레임간 상관도가 높다는 성질을 이용하여 에러를 은닉한다. 대체적으로 시방향 에러 은닉 방법이 공간적 에러 은닉 방법보다 더 만족할만한 화질을 보인다. 본 논문은 시방향 에러 은닉 방법에 중점을 두었다.

시방향 에러 은닉 방법 중 가장 기본적인 방법은 TR (Temporal Replacement) 기법으로 이전 프레임에서 손실된 매크로 블록의 동일한 위치의 매크로 블록으로 대체한다. TR 기법은 단순한 과정을 가지기 때문에 연산량이 매우 적고 움직임이 없는 영역에서만 좋은 화질을 보인다. 하지만 움직임이 큰 영역에서는 매우 큰 화질 열화를 보인다.

BMA (Boundary Matching Algorithm)<sup>[4]</sup>은 가장 잘 알려진 시방향 에러 은닉 기법으로 H.264의 참조 모델<sup>[5]</sup>에도 적용되어 있다. BMA는 인접한 화소 간에 높은 공간적 상관도가 존재한다는 사실에 기반한 방법이다. 손실된 매크로 블록은 매크로 블록의 좌측 상단의 위치에서부터 인접한 블록들의 움직임 벡터들이

가리키는 이전 프레임의 매크로 블록들 중 내부 경계와 인접 매크로 블록의 내부경계와의 차이가 가장 작은 매크로 블록으로 대체된다. BMA는 TR의 단점을 개선시켜 단조로운 움직임이 있는 영역에서는 좋은 화질을 보인다. 하지만 움직임이 복잡하거나 인접한 매크로 블록의 움직임 벡터가 상반된 움직임을 가지는 영역에서는 부적절한 움직임 벡터를 선택할 수 있다.

M.-J. Chen등은 BMA의 단점을 개선하기 위하여 OBMC (Overlapped Block Motion Compensation)<sup>[6]</sup>과 BMA를 결합한 알고리즘을 제안하였다<sup>[7]</sup>. T. Chen등은 손실된 매크로 블록을 분할하여 각 분할 영역에 대해 서로 다른 움직임 벡터를 사용하는 RBMA (Refined Boundary Matching Algorithm) 기법을 제안하였다<sup>[8]</sup>. BMA에 기반한 시방향 에러 은닉 방법들은 적절하지 못한 움직임 벡터를 선정하고 에러 은닉에 사용 경우에는 에러를 은닉한 매크로 블록과 주변에 인접한 매크로 블록들 사이의 블록화 현상이 발생하여 화질을 저하시킨다는 단점이 있다.

이러한 단점을 극복하기 위한 방법으로 공간적 보간법을 이용하여 손실된 매크로 블록의 움직임 벡터를 복원하여 은닉하는 기법들이 제안되었다. 먼저 Zheng등은 매크로 블록을 4×4 화소 단위로 고려하고 LI (Lagrange Interpolation) 을 움직임 벡터를 복원하는데 이용하였다<sup>[9]</sup>. Al-Mualla등은 시방향 에러 은닉에 양선형 보간법을 적용한 MFI (Motion Field Interpolation)을 제안하였으며<sup>[10]</sup>, B. N. Chen등은 MFI를 개선한 SMFI (Selective Motion Field Interpolation)을 제안하였다<sup>[11]</sup>. SMFI는 인접한 매크로 블록의 움직임 벡터 중에서 손실된 매크로 블록과 상관도가 높은 움직임 벡터 2개를 선정하여 움직임 벡터를 복원한다. 이러한 공간적 보간법을 사용한 기법들은 블록화 현상을 크게 감소 시킨다. 하지만 작은 영역에서 움직임이 발생한다면 좋은 성능을 나타내지 못한다는 단점이 있다.

본 논문에서는 H.264/AVC를 위한 보다 능률적인 시방향 에러 은닉 알고리즘을 제안한다. 제안하는 알고리즘은 보다 정확한 은닉을 위해서 손실된 매크로 블록을 분할한다. 분할된 영역은 움직임 유사도 방향에 따른 움직임 벡터 후보군을 형성한다. 다음으로 선택된 움직임 후보군 중에서 은닉에 적합한 움직임 벡터를 이용하여 에러를 은닉한다. 제안하는 알고리즘을 다양한 움직임을 가지는 다수의 테스트 영상에서 실험해본 결과 기존의 시방향 에러 은닉 기법들과는 다르게 테스트 영상의 특성, 매크로 블록 에러율 및 비트율의 변화에 영향을 받지 않고 좋은 성능을 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 알고리즘에 대해 설명한다. 3장에서는 제안하는 알고리즘을 다양한 테스트 영상에서 실험한 결과를 기존의 시방향 에러 은닉 기법과의 비교하여 나타내었다. 마지막으로 4장에서 결론을 맺는다.

## II. 제안하는 알고리즘

제안하는 알고리즘은 세 단계의 과정으로 이루어진다. 먼저 움직임 벡터간의 유사도 계산을 통해 손실된 매크로 블록의 움직임 벡터의 방향성을 추정한다. 둘째로, 전 단계에서 추정된 방향성을 고려하여 움직임 벡터의 후보군을 생성한다. 셋째로, 움직임 후보군중에서 은닉에 적합한 움직임 벡터를 선택하여 은닉에 이용한다.

그림 1에서 묘사되어 있는 것과 같이 H.264/AVC 표준은 움직임 예측과 보상의 효율을 높이기 위해서 다양한 블록 크기를 지원한다. 제안하는 알고리즘은 복원 효율을 높이기 위해서 H.264/AVC 표준이 지원하는 다양한 블록 크기 중에 가장 작은 4x4 블록 크기를 사용한다.

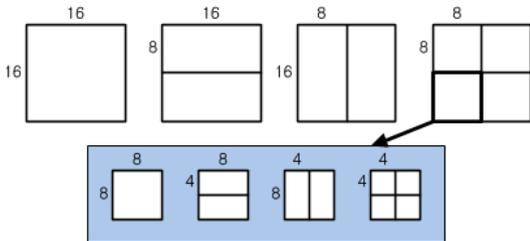


그림 1. H.264/AVC 표준에서 지원하는 블록 크기

### 2.1 움직임 벡터간의 유사도 계산

움직임 벡터간의 유사도를 계산함에 있어서 가장 정확한 방법은 손실된 매크로 블록의 전방향, 즉 360°에 대한 움직임 유사도를 계산하는 것이다. 하지만 이는 매우 큰 계산을 필요로 한다. 따라서 움직임 벡터간의 유사도 계산에 필요한 방향을 좀 더 정확하게 선정할 필요가 있다.

$$\Theta_{center} = \{0^\circ, \pm 11^\circ, \pm 22^\circ, \pm 31^\circ, \pm 39^\circ, \pm 45^\circ, \pm 51^\circ, \pm 59^\circ, \pm 68^\circ, \pm 79^\circ, 90^\circ\} \quad (1)$$

식 (1)에서  $\Theta_{center}$  는 손실된 매크로 블록의 중심을 가로지르는 움직임 벡터의 방향을 의미한다.  $\Theta_{center}$  는

손실된 매크로 블록의 전방향을 고려한 것이다.

제안하는 알고리즘은 손실된 매크로 블록의 움직임 유사도를  $\Theta_{center}$  의 방향을 이용하여 계산한다. 만약 손실된 매크로 블록을 가로지르는 움직임 벡터들이 서로 유사하다면, 손실된 매크로 블록이 해당 블록들과 동일한 객체일 확률이 매우 높다. 즉 손실된 매크로 블록과 해당 블록간의 상관성이 매우 높다는 것이다. 따라서 이 블록들의 움직임 벡터를 시방향 에러 은닉 과정에서 이용하는 것이 효과적이다.

움직임 유사도 계산을 하기 위해서 먼저  $\Theta_{center}$  를 다음과 같이 2개의 그룹으로 분할한다.

$$\theta_1 = \{\pm 11^\circ, \pm 31^\circ, \pm 45^\circ, \pm 59^\circ, \pm 79^\circ\} \quad (2)$$

$$\theta_2 = \{0^\circ, \pm 22^\circ, \pm 39^\circ, \pm 41^\circ, \pm 68^\circ, 90^\circ\} \quad (3)$$

$\theta_1$  는 손실된 매크로 블록의 중심을 가로지르면서 동시에 손실 없이 복원된 주변 블록의 중심을 가로지르는 벡터의 방향이고,  $\theta_2$  는 손실된 매크로 블록의 중심을 가로지르면서 동시에 인접한 2개의 손실 없이 복원된 주변 블록의 사이의 중심을 가로지르는 벡터의 방향이다. 그룹  $\theta_1$ , 그룹  $\theta_2$  에 속해있는 방향의 움직임 유사도 계산은 각각 식 (4)와 식 (5)와 같다.

$$S_{\theta_1} = |MV_i - MV_j| \times \alpha \quad (4)$$

$$S_{\theta_2} = |MV_i - MV_j| + |MV_k - MV_l| \quad (5)$$

여기서 식 (4)의  $\alpha$  는 그룹  $\theta_1$  의 유사도 계산에 사용하는 블록의 개수와 그룹  $\theta_2$  의 유사도 계산에 사용하는 블록의 개수의 비례를 고려한 변수로 2를 사용하고,  $|MV_i - MV_j|$  는 다음 식 (6)과 같다.

$$|MV_i - MV_j| = |MV_i(x) - MV_j(x)| + |MV_i(y) - MV_j(y)| \quad (6)$$

여기서  $MV(x)$  와  $MV(y)$  는 각각 움직임 벡터의 수평, 수직 성분을 의미한다. 그림 2는 그룹  $\theta_2$  에 해당하는  $-68^\circ$  방향에서 움직임 벡터 유사도 계산의 예

를 묘사한다.

손실된 매크로 블록의 유사도 방향은 계산된 모든 방향 중에서 움직임 벡터 차이가 가장 적은 방향을 유사도 방향으로 선택한다.

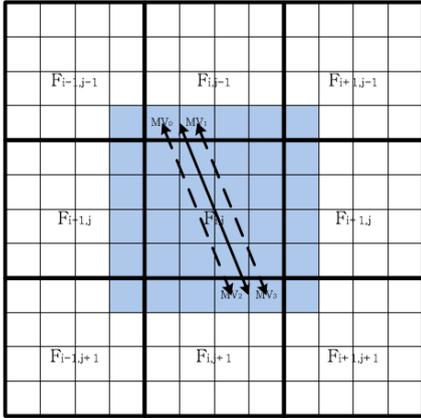


그림 2. 움직임 벡터간 유사도 계산

2.2 위치에 따른 에러 은닉 순서

제안하는 알고리즘은 손실된 매크로 블록을 4x4 블록 크기로 은닉을 시행하기 때문에 보다 정확한 은닉을 위해서 은닉 순서를 정의할 필요가 있다.

손실된 매크로 블록은 그림 3의 (a)와 같이 손실된 매크로 블록을 8x8 블록 크기의 영역으로 분할된다. 분할된 영역은 그림 3의 (b)와 같이 4x4 블록 크기로 다시 분할되어 서로 독립적인 은닉 순서로 은닉을 시행한다.

은닉 순서를 정하기 위해서 두 조건이 사용되었다. 첫 번째, 주변 블록과의 거리가 가깝다. 두 번째, 많은 손실 없이 복원된 매크로 블록과 인접해 있다. 두 조건이 사용된 이유는 다음과 같다. 은닉을 시행할 블록과 손실 없이 복원된 주변 블록과의 거리가 가까울수록 블록간의 상관도는 높아진다. 또한, 주변에 손실 없이 복원된 매크로 블록이 더 많은 위치의 블록은 이

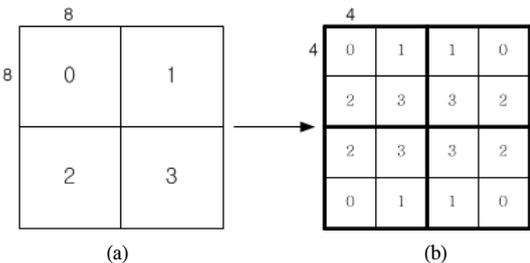


그림 3. 은닉 순서 (a)8x8 블록 크기 영역 분할 (b)8x8 블록 크기 영역내 블록 은닉 순서

용할 수 있는 정보의 신뢰성이 높아진다.

따라서, 이 두 가지의 조건을 만족하는 손실된 매크로 블록의 모서리에 위치한 블록들이 우선적으로 은닉이 되고, 그 다음으로 주변 블록과 거리는 같지만 이용할 정보가 적은 블록들의 순서로 은닉을 시행한다. 마지막으로 거리가 멀리 떨어진 블록의 은닉을 시행한다.

2.3 움직임 벡터 후보군 선택

각각의 4x4 블록은 이전 절에서 결정된 움직임 유사도의 방향에 따르는 서로 다른 유사도 벡터 후보군과 함께 손실된 매크로 블록 내의 4x4 블록의 위치가 8x8 블록 크기로 분할된 영역의 어느 위치에 속해있느냐에 따라서 서로 다른 위치 벡터 후보군을 가진다. 위치 벡터 후보군은 다음 식 (7)과 같다.

$$\begin{aligned}
 PCMG_0^S &= \{ZM, MV^U, MV^L\} \\
 PCMG_1^S &= \{ZM, MV^U, MV^R\} \\
 PCMG_2^S &= \{ZM, MV^D, MV^L\} \\
 PCMG_3^S &= \{ZM, MV^D, MV^R\}
 \end{aligned} \tag{7}$$

여기에서  $PCMG_i^S$ 는 8x8 블록 크기로 분할된  $i$ 번의 영역의 위치에 따른 위치 벡터의 후보군을 의미한다.  $ZM$ 은 제로 움직임 벡터를 의미하고,  $MV^U$ ,  $MV^D$ ,  $MV^L$  그리고  $MV^R$ 은 각각 해당 블록의 상, 하, 좌, 우에 인접한 블록의 움직임 벡터를 의미한다.

유사도 벡터 후보군을 형성하기 위해서 이전에 결정된 유사도 방향의 벡터를 각 4x4 블록의 중심으로 이동시키고, 벡터가 지나가는 방향에 존재하는 손실 없이 복원된 블록의 움직임 벡터를 이용한다. 그림 4는 결정된 유사도 방향이 -68°일 때 유사도 벡터 후보군을 선택하는 예를 보여준다. 유사도 벡터 후보군은 결정된 유사도 방향에 따라 포함되는 움직임 벡터의 개수가 달라지는데 다음 식 (8)과 같다.

$$SCMG = \begin{cases} \{MV_0, MV_1, MV^W\}, & MSD = 0^\circ, \pm 45^\circ, 90^\circ \\ \{MV_0, MV_1, MV_2, MV_3, MV^W\}, & otherwise \end{cases} \tag{8}$$

여기에서  $SCMG$ 는 유사도 벡터 후보군을 의미한다.  $MV_j$ 는 4x4 블록의 중심을 가로지르는 유사도 벡터의 방향이 지나가는 손실 없이 복원된 주변 블록들의 움직임 벡터들이다.  $MV^W$ 는 움직임 벡터를 보다

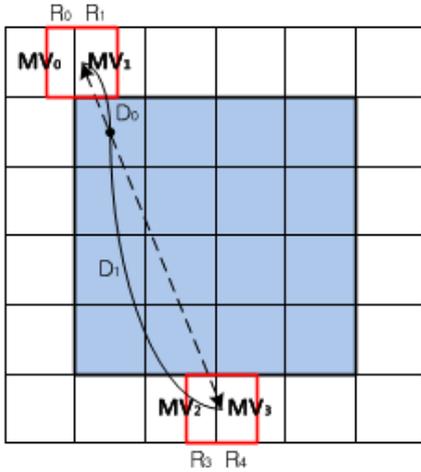


그림 4. 유사도 벡터 후보군 선택의 예

정확히 추정하기 위해, 식 (9)를 사용하여  $MV_j$ 의 가중치 평균을 계산한 것이다.

$$MV^w = \sum_{j=0}^l (W_j \cdot MV_j), l = \begin{cases} 1, & MSD = 0^\circ, \pm 45^\circ, \pm 90^\circ \\ 3, & otherwise \end{cases} \quad (9)$$

여기에서  $W_j$ 는 가중치를 의미하는데 만약  $l=1$ 일 때 식 (10)을 사용하여 가중치 계산을 하고  $l=3$ 일 때 식 (11)을 사용하여 가중치 계산을 한다.

$$W_0 = \frac{D_1}{D_0 + D_1}, W_1 = \frac{D_0}{D_0 + D_1} \quad (10)$$

$$W_0 = \frac{D_1}{D_0 + D_1} \times \frac{R_0}{R_0 + R_1}, W_2 = \frac{D_0}{D_0 + D_1} \times \frac{R_2}{R_2 + R_3} \quad (11)$$

$$W_1 = \frac{D_1}{D_0 + D_1} \times \frac{R_1}{R_0 + R_1}, W_3 = \frac{D_0}{D_0 + D_1} \times \frac{R_3}{R_2 + R_3}$$

여기서  $D_i$ 와  $R_j$ 은 블록이 유사도 벡터의 방향을 따라 주변 블록까지 이동했을 때 거리와 주변 블록과 겹쳐지는 면적을 각각 의미한다.

$$UCMG = PCMG_i^S \cup SCMG \quad i = 0, 1, 2, 3 \quad (12)$$

최종적으로 식 (12)와 같이 유사도 벡터 후보군과 위치 벡터 후보군을 모두 포함한 결합 움직임 벡터 후보군을 나타내는 UCMG를 형성한다.

### 2.4 가중치 외부 경계 정합 알고리즘

은닉을 시행하기 전에 추정된 움직임 벡터 후보군에서 가장 적합한 움직임 벡터를 선택해야 한다. 이를 위해서 제안하는 알고리즘은 외부 경계 정합 알고리즘 (OBMA)를 개선한 가중치 외부 경계 정합 알고리즘 (WOBMA)가 사용된다. OBMA의 외부 경계 화소값의 차이 계산은 다음 식 (13)과 같이 나타낸다.

$$OBM_U = \sum_{x=x_0}^{x_0+N-1} \left| P_{x,y_0-1}^{cur} - P_{x+M_x,y_0-1+M_y}^{ref} \right|$$

$$OBM_D = \sum_{x=x_0}^{x_0+N-1} \left| P_{x,y_0+N}^{cur} - P_{x+M_x,y_0+N+M_y}^{ref} \right|$$

$$OBM_L = \sum_{y=y_0}^{y_0+N-1} \left| P_{x_0-1,y}^{cur} - P_{x_0-1+M_x,y+M_y}^{ref} \right|$$

$$OBM_R = \sum_{y=y_0}^{y_0+N-1} \left| P_{x_0+N,y}^{cur} - P_{x_0+N+M_x,y+M_y}^{ref} \right| \quad (13)$$

여기서  $OBM_U$ ,  $OBM_D$ ,  $OBM_L$  그리고  $OBM_R$ 은 각각 블록의 상, 하, 좌, 우의 경계 화소값의 차이를 나타낸다.  $P_{x,y}^{cur}$ 와  $P_{x,y}^{ref}$ 는 각각 현재 프레임과 참조프레임의  $(x,y)$  위치에서의 화소값을 의미하며,  $x_0$ 와  $y_0$ 는 블록의 가장 좌상단에 위치한 화소의 수평, 수직 좌표를 나타낸다.  $M_x$ 와  $M_y$ 는 움직임 벡터 후보의 수평, 수직 성분을 의미하고,  $N$ 은 블록의 크기를 나타낸다. 움직임 벡터 후보를 선택하는 기준이 되는 비용 함수는 식 (13)에서 얻어진 각 외부 경계 화소값 차이를 이용하여 식 (14)와 같이 정의 된다.

$$Cost = OBM_U + OBM_D + OBM_L + OBM_R \quad (14)$$

제안하는 알고리즘은  $8 \times 8$  블록 크기로 분할된 영역에 따라 서로 독립적인 순서로 은닉되기 때문에 블록의 위치가  $8 \times 8$ 로 분할된 영역의 어느 위치에 속해 있느냐에 따라 서로 다른 비용함수가 요구된다. 그렇기 때문에 OBMA를 제안하는 알고리즘에 사용하기 위해서 식 (14)를 다음 식 (15)와 같이 수정하였다.

$$Cost_{WOBM,0} = W(a) \cdot OBM_U + W(b) \cdot OBM_L$$

$$Cost_{WOBM,1} = W(a) \cdot OBM_U + W(b) \cdot OBM_R$$

$$Cost_{WOBM,2} = W(a) \cdot OBM_D + W(b) \cdot OBM_L$$

$$Cost_{WOBM,3} = W(a) \cdot OBM_D + W(b) \cdot OBM_R \quad (15)$$

여기서  $a$ 와  $b$ 는 각 블록의 위치에 따른 가중치 변수로 그림 5에 묘사되어 있으며,  $W$ 는 외부 경계 화소

1,1	1,0	1,0	1,1
0,1	0,0	0,0	0,1
0,1	0,0	0,0	0,1
1,1	1,0	1,0	1,1

그림 5. 각 블록의 가중치 변수 (e.g. 1,1 : a=1,b=1)

값 차이에 부여하는 가중치를 의미하며 식 (16)과 같이 나타낸다.

$$W(\mu) = \begin{cases} 1 & \mu=0 \\ 2 & \mu=1 \end{cases} \quad (16)$$

은닉에 이용할 수 있는 정보 중에서 가장 신뢰성이 높은 정보는 손실된 매크로 블록 주변의 손실 없이 복원된 외부 경계 화소값이다. 하지만, 제안하는 알고리즘은 4×4 블록 크기로 은닉을 시행하기 때문에 블록의 위치에 따라서 먼저 은닉된 블록의 내부 경계 화소값을 이용하여 경계 화소값 차이를 계산하는 경우도 있다. 비용값 계산시 신뢰성이 낮은 은닉된 블록의 내부 경계 화소값을 사용하는 경계값 차이 보다 신뢰성이 높은 손실 없이 복원된 외부 경계 위치의 화소값을 사용하는 경계값 차이가 작을수록 정확한 은닉이 가능해진다. 따라서, 손실 없이 복원된 외부 경계 화소값 차이에 가중치를 부여함으로써 보다 정확한 움직임 벡터를 선택할 수 있는 확률이 높아진다.

WOBMA는 최종적으로 손실된 블록의 움직임 벡터는 움직임 벡터 후보군 중에 경계 화소값의 차이를 최소화하는 것으로 선택한다.

### III. 실험 결과

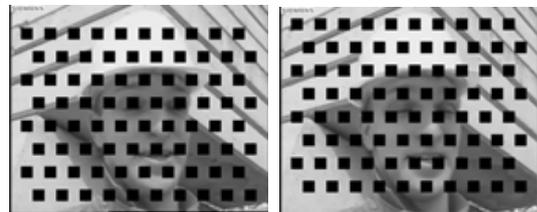
제안하는 알고리즘의 성능을 평가하기 위해서 H.264/AVC 참조 모델 (JM 15.1)<sup>[4]</sup>에 알고리즘을 적용하였다. 테스트 영상은 CIF (352 × 288) 크기의 4개의 영상 (Container, Foreman, Football, Mobile)을 사용하였고, 첫 프레임부터 100프레임까지 부호화했다. GOP (Group of Picture)의 구조는 첫 프레임만 화면내 부호화를 수행하고 나머지 프레임은 화면간 부호화를 사용한 IPPP... 구조를 사용하였다. 객관적

성능 평가 기준으로 휘도 성분의 PSNR (Peak Signal to Noise Ratio)을 이용하였다. 제안하는 알고리즘의 성능을 보다 명시적으로 평가하기 위해 TR, BMA, LI, 세 가지 방법과 비교하였다.

다양한 테스트 조건을 위하여 QP(Quantization Parameter)와 MBER(MacroBlock Error Ratio)의 변화를 주었다. 비트율의 변화에 대한 알고리즘의 성능 평가를 위해서 QP는 24와 28 두 가지의 파라미터를 사용하였고, 에러율의 변화에 대한 성능 평가를 위해서 프레임당 에러 매크로 블록의 비율인 MBER을 5%, 10%, 20%로 변경하여 각 알고리즘에 적용하여 실험 비교하였다. 다음 표 1은 실험 결과로 얻어진 복원된 영상의 평균 PSNR을 나타낸다.

표 1에서 Error-free은 에러 없이 복원된 영상의 PSNR을 뜻한다. 실험 결과에서 관찰되었듯이 기존의 알고리즘은 영상의 특성에 따라서 성능이 달라진다. 작은 영역에서 움직임이 있는 Container 영상에서 BMA가 LI보다 좋은 성능을 보이고, 이외는 반대로 Container 영상 보다 움직임이 조금 더 큰 Foreman 영상에서는 반대로 LI가 BMA보다 좋은 성능을 보인다. 하지만 제안하는 알고리즘은 영상의 특성에 상관없이 좋은 성능을 보인다. 또한 비트율과 에러율의 변화에 상관없이 좋은 성능을 보인다. 특히 에러율이 10%일 때 가장 좋은 성능을 보인다. 제안하는 알고리즘은 BMA보다 최소 0.01 dB에서 최대 3.02 dB 높은 성능을 보이고, LI보다 최소 0.15 dB에서 최대 1.48 dB 높은 성능을 보인다.

그림 6은 MBER 20%일 때의 에러 패턴을 나타낸다. 다음으로 그림 7은 복원된 영상의 주관적 화질을 나타낸다. 그림 7은 MBER이 10%이고 QP가 28일 때의 Foreman 영상의 30번째 프레임으로 주관적 화질도 마찬가지로 제안하는 알고리즘의 성능이 기존의 알고리즘보다 프레임 전체에서 우수함을 보인다. 특히 배경부분이 BMA와 LI 보다 주관적 화질이 뛰어난을 보인다.



(a) Odd frame

(b) Even frame

그림 6. 에러 패턴



그림 7. 복원된 영상의 주관적 화질 비교

표 1. MBER에 따른 PSNR 성능 비교

Sequence	QP	Error-free	PSNR(dB)	MBER		
				5%	10%	20%
Container	24	38.74	TR	34.33	32.48	28.73
			BMA	34.33	32.48	28.73
			LI	33.44	31.75	28.57
			Propose	34.35	32.49	28.78
	28	36.17	TR	32.89	31.29	28.09
			BMA	32.89	31.29	28.09
			LI	32.62	31.16	28.04
			Propose	33.27	31.58	28.30
Foreman	24	39.28	TR	24.93	21.72	19.94
			BMA	29.86	25.82	24.28
			LI	30.82	28.24	26.24
			Propose	31.83	28.43	26.69
	28	36.74	TR	25.2	21.88	19.91
			BMA	29.83	25.83	23.72
			LI	31.34	28.33	25.87
			Propose	32.44	28.85	26.28
Football	24	38.09	TR	17.91	15.77	14.26
			BMA	21.93	19.44	16.97
			LI	21.10	19.08	16.78
			Propose	22.44	20.17	17.76
	28	34.66	TR	17.84	15.47	13.95
			BMA	21.54	19.25	16.87
			LI	20.71	18.96	17.12
			Propose	22.19	20.08	17.27
Mobile	24	39.01	TR	19.88	17.4	14.66
			BMA	22.30	19.81	16.90
			LI	23.72	21.48	18.59
			Propose	24.74	21.95	19.01
	28	35.32	TR	19.64	17.25	14.53
			BMA	22.21	19.58	16.92
			LI	23.33	20.90	18.44
			Propose	24.04	21.56	18.82

#### IV. 결 론

본 논문에서는 H.264/AVC를 위한 효율적인 시방향 에러 은닉 알고리즘을 제안하였다. 제안하는 알고리즘은 보다 효과적인 화질 개선을 위하여 손실된 매크로블록을 4×4 크기의 블록으로 분할한 뒤에 인접한

움직임 벡터간의 높은 유사도를 고려하여 움직임 벡터 후보군을 추정한다. 또한, 추정된 움직임 벡터 후보군에서 가장 낮은 비용값을 가지는 움직임 벡터를 은닉에 이용할 정보의 신뢰도에 따른 가중치 외부 경계 정합을 통해 선택하여 은닉에 이용한다. 실험 결과에서 나타나듯이 제안하는 알고리즘은 영상의 특성,

매크로 블록 에러율, 비트율 에 상관없이 안정적인 성능을 보인다. 또한 이전의 알고리즘에 비해서 객관적, 주관적 화질면에서 월등히 높은 성능을 보였다. 주관적인 화질면에서 블록화 현상이 상당히 완화되었다. 특히 객관적인 화질면에 있어서 비슷한 은닉 과정을 가지는 H.264의 BMA에 비교하여 PSNR이 최대 3.02 dB의 성능 향상을 보였다.

### 참 고 문 헌

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, Vol.13, No.7, pp.560-576, Jul. 2003.

[2] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error Resilient Video Coding Techniques - Real Time Video Communications over Unreliable Network," *IEEE Signal Processing*, Vol.17, No.4, pp.61-82, Jul. 2000.

[3] Y. Wang and Q. F. Zhu, "Error Control and Concealment for Video Communication: A Review," in *Proc. IEEE*, Vol.86, No.5, pp. 974-997, May. 1998.

[4] V. Varsa and M. M. Hannuksela, "Non-normative error concealment algorithms," ITU-T SG16 Doc. VCEG-N62, 2001.

[5] Joint Video Team (JVT) reference Software 15.1.

[6] M. Orchard and G. J. Sullivan, "Overlapped block motion compensation: An estimation theoretic approach," *IEEE Trans. Image Processing*, Vol.3, No.5, pp.693-699, Sep. 1994.

[7] M.-J. Chen, L.-G. Chen and R.-M. Chen, "Error concealment of lost motion vectors with overlapped motion compensation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol.7, No.3. pp. 560-563, Jun. 1997.

[8] T. Chen, X. Zhang, and Y. Q. Shi, "Error Concealment Using Refined Boundary Matching Algorithm," *IEEE Trans. Circuits and Syst. Video Technol.*, Vol.13, No.4, pp.560-576, Jul. 2003.

[9] J. H. Zheng and L. P. Chau, "A temporal error

concealment algorithm for H.264 using Lagrange interpolation," in *Proc. IEEE Int. Symp. Circuits Syst.*, Vol.2, pp.133 - 136, May 2004.

[10] M. E. Mualla, C. N. Canagarajah, and D. R. Bull, "Motion field interpolation for temporal error concealment," in *Proc. IEE Vis. Image Signal Process.*, Vol.147, No.5, pp.445-453, Oct. 2000.

[11] B. N. Chen and Y. Lin, "Temporal Error Concealment Using Selective Motion Field Interpolation," *IET Electronics Letters*, Vol.42, No.24, pp.1390-1391, Nov. 2006.

민 창 기 (Changki Min)

준회원



2009년 2월 세종대학교 전자공학과  
2009년 2월~현재 한양대학교 전자컴퓨터통신공학과 석사과정  
<관심분야> 영상처리, 영상압축, 에러 은닉

진 순 종 (Soonjong Jin)

정회원



2004년 2월 한양대학교 전자컴퓨터공학부  
2006년 3월 한양대학교 전자통신컴퓨터공학과 석사  
2010년 3월 한양대학교 전자통신컴퓨터공학과 박사  
2010년 3월~현재 LG전자 CTO DTV 연구소

<관심분야> 영상압축, 영상처리, H.264/AVC, Transcoding, Multi-view Video Coding, Scalable Video Coding

오 형 철 (Hyeongchul Oh)

준회원



2009년 2월 강남대학교 전자  
공학과

2009년 3월~현재 한양대학교  
전자컴퓨터통신공학과 석사  
과정

<관심분야> 영상처리, 영상압  
축, Frame Interpolation

정 제 창 (Jechang Jeong)

정회원



1980년 2월 서울대학교 전자  
공학과

1982년 2월 KAIST 전기전자  
공학과 석사

1990년 미국 미시간대학 전기  
공학과 공학박사

1980~1986 KBS 기술연구소  
연구원(디지털 TV 및 뉴미디어 연구)

1990~1991 미국 미시간대학 전기공학과 연구교수  
(영상 및 신호처리 연구)

1991~1995 삼성전자 멀티미디어 연구소 (MPEG,  
HDTV, 멀티미디어 연구)

1995~현재 한양대학교 전자통신컴퓨터공학과 교수  
(영상통신 및 신호처리 연구실)

1998년 11월 27일 과학기술자상 수상

1998년 12월 21일 정보통신부장관상 표창

<관심분야> 영상처리 및 영상압축