

H.263 비디오 코덱의 채널 잡음에 대한 수신단 영상의 블록오류 검출 알고리즘

정회원 최 세 하*, 이 주 신**

Block Error Detection Algorithm of Received Image in the Channel Noise of H.263 Video Codec

Se Ha Choi*, Joo Shin Lee** *Regular Members*

요 약

본 논문에서는 H.263 동영상 압축 전송에서 수신단에서의 채널 잡음에 의한 블록 오류의 검출 알고리즘을 제안하였다. 제안한 오류 검출 알고리즘은 인접된 두 프레임에서 서로 대응되는 블록들간의 시간적 유사성을 이용하여 블록내의 화소값 변화가 큰 블록과 작은 블록으로 분류하였고, 블록내 화소값 변화가 큰 블록들에 대해서 공간적 유사성을 이용하여 주변 블록과 서로 인접한 화소들의 화소차 변화가 큰 것을 오류 블록으로 판정하였다. 제안한 알고리즘의 성능평가를 위해 H.263환경에서 동영상에 오류를 삽입한 후 기존의 VLC(Variable Length Code) 테이블에 의한 오류 검출 알고리즘과 제안한 알고리즘의 검출 결과를 비교 분석하였다. 그리고 오류로 검출된 은닉한 후 은닉된 영상에 대해 PSNR(Peak Signal to Noise Ratio)을 계산하여 정량적으로 화질을 평가하였다. 실험 결과는 본 논문에서 제안한 오류 검출 알고리즘이 채널 잡음에 의한 블록 오류를 효율적으로 검출할 수 있으며, VLC 테이블에 의한 오류 검출 방법보다 15dB 이상 화질이 개선됨을 보여주었다.

ABSTRACT

In this paper, we proposed an algorithm to detect block error caused by channel noise in H.263 video transmission. The proposed error detection algorithm classified macro blocks to large blocks and small blocks by the variation of pixel value between two corresponded blocks in the inter frames using temporal similarity. And it was decided that the error blocks which has large variance than adjacent blocks among the larger one at first procedure, using the spatial similarity. To evaluate the performance of the proposed algorithm, we compared the detection results with existence algorithm. and we also evaluated image quality using the PSNR of the image after concealed the error block by two methods. The simulation result showed that the proposed error detection algorithm can detect the error block more effectively than existing algorithm and the image quality improved more than 15dB compared with existence error detection algorithm.

I. 서 론

최근 디지털 영상부호화 기술의 발전으로 디지털 TV, HDTV, 주문형 비디오(VOD; Video on demand), 멀티미디어 데이터베이스 시스템 등과 같은 광범위한 응용분야를 목표로 다양한 연구성과를

이루어왔다^[1]. 또한 영상통신 분야의 이용 증가로 영상통신 시스템의 상호 호환성을 위해 영상 부호화와 관련된 국제표준의 필요성이 요구되었으며, 영상부호화 표준들은 영상통신 분야의 활발한 연구와 더불어 새롭게 제정되고 있는 실정이다^[2]. 일반적으로 문자나 음성 데이터는 전송시 오류가 발생되면

* 정보통신부 정책실 기술기준과(choiseha@mic.mic.go.kr)
논문번호: 98346-0810, 접수일자: 1998년 8월 10일

** 청주대학교 전자공학과

재전송을 요구할 수 있지만 영상의 경우에는 데이터량이 방대하기 때문에 재전송이 곤란하여 사람의 눈에 크게 거슬리지 않을 정도의 오류는 허용이 가능하다. 이러한 영상신호의 전송특성을 고려하여 채널오류에 대응하기 위한 화질 개선 방법으로 다양한 오류복원기법과 오류은닉 기법들이 연구되어지고 있다^{[3]-[7]}.

오류 검출 및 은닉기법에는 주파수 영역에서 이산 여현 변환(DCT: Discrete Cosine Transform) 계수를 이용한 방법과 공간영역이나 시간영역에서 화소 데이터를 이용하는 방법이 있다. 특히, 저전송율 동영상 전송에서는 압축율과 실 시간성, 하드웨어 구현의 용이성 등을 고려하여 오버헤드의 추가나 복잡한 계산이 요구되지 않는 간단한 방법에 의한 오류 검출 및 처리가 필요하다.

전송 채널의 잡음에 의한 화질의 저하를 막기 위한 방법으로 수신단에서의 오류 검출(패리티 체크) 및 오류 정정 기법, 채널 오류를 줄이기 위한 소스 부호화기를 수정하는 방법, 그리고 이미 수신된 훼손 데이터에 대해 오류를 검출하고 은닉하는 방법 등이 있다.

수신단의 오류 검출 및 은닉방법은 수신된 데이터와 이전에 수신된 데이터와의 비교를 통하여, 오류를 검출하고 은닉하는 방법으로서 비트스트림에 부가적인 정보의 추가가 없이 오류의 검출 및 은닉이 가능하다.

본 논문에서는 H.263 동영상 압축 전송에 적합한 수신단에서의 블록 오류의 검출 알고리즘을 제안한다. 오류 검출 알고리즘은 매크로 블록의 시·공간적 유사성을 이용하여, 이전 프레임의 대응하는 매크로 블록과 유사성, 주변 블록들과의 정합 특성을 가지고 오류 매크로 블록을 검출하였다. 본 논문의 II장에서는 채널오류에 의한 화질의 영향을 분석하였고, III장에서는 기존의 오류검출알고리즘에 대한 분석하고 제안한 오류검출알고리즘에 대해 설명하였으며, IV장에서는 실험 및 고찰을 통하여 본 논문에서 제안한 방법의 성능을 분석하였으며, 기존의 오류검출 알고리즘보다 좋은 성능을 가짐을 보여주고 V장에서 결론을 맺고자 한다.

II. 채널 오류에 의한 화질의 영향

H.263은 MPEG이나 H.261등 디지털 영상압축 표준과 같이 공간적 상관성을 이용한 압축 수법으로서 DCT를, 시간적 상관성을 이용한 수법으로서

움직임 예측 및 보상기법을 각각 채택하고 있다. 또한 압축 효율을 높이기 위하여 DCT연산 결과에 양자화를 실행하며, 부호 발생의 통계적 성질에 따른 가변장 부호화를 사용하여 더욱 높은 정보 압축을 얻고 있다^{[9]-[10]}.

이렇게 압축된 비트열은 Picture, GOB(Group of block), 매크로 블록(Macro Block), 블록 등의 계층(Layer)으로 나뉘어지며, 가장 기본적인 정보인 움직임 벡터와 DCT계수 외에도 각 계층의 시작과 끝을 나타내는 동기신호와 기타 부호화 계층의 특성을 표현하는 다양한 정보가 포함되어 있다. 만약 동기신호나 계층의 헤더정보에 오류가 발생하면, 작게는 한 GOB를, 크게는 한 영상이나 GOP 또는 전 시퀀스를 복원할 수 없게 된다. 만약 오류의 영향이 움직임 벡터나 DCT계수가 오류에 의해 손상되면, 오류는 1개 매크로 블록이나 블록에 국한되게 된다. 그러나 VLC에 의해 한 비트의 오류라 할지라도 디코딩에 동기를 잃어 전체 블록에 퍼져나가게 되어 화면이 블록이나 매크로 블록 단위로 깨지게 된다. 또한 움직임 예측과 보상기법의 사용으로 인하여, 한 프레임에서 나타난 오류가 그 다음의 연속하는 프레임들에 계속 전파(Propagation)되어 나타나며 주위로 확산되게 된다. 이러한 오류를 줄이기 위해서 오류 정정 코드를 사용하더라도 정정되지 않은 비트 오류들이 존재하게 됨으로 복호기에서의 동기상실이 유발된다^{[7],[8],[14]}.

III. 오류 검출 알고리즘

채널 잡음에 의한 오류를 최대한 줄이기 위하여, 기존의 오류 은닉 알고리즘들은 부호화기에서 비트스트림에 동기신호와 같은 부가적인 정보를 삽입하여 수신단에서 훼손된 영상의 블록의 위치를 판별하여 은닉할 수 있게 한다. 이러한 기법들은 비록 효율적으로 오류를 검출할 수 있지만 부가정보의 추가로 인해 데이터량이 증가되게 되어 H.263과 같은 저전송율 통신 환경에서는 적합하지 않다. 또한 BCH코드와 같은 오류 정정코드를 사용하면 비트 오류는 검출하고 복원할 수 있지만 데이터량이 증가되며, 통신채널에서 많이 발생하는 연접 오류(Burst Error)에 대해서는 복원이 불가능하기에 적용한계가 있다.

1. VLC 테이블에 의한 오류 검출방법

수신단에서 부호화시 오류의 검출이나 은닉을 위

한 부가 정보를 추가하지 않은 경우, 기존의 수신단 오류는넉기법들은 주로 다음과 같은 두 가지 방법을 들 수 있다. 한가지 방법은 수신된 한 프레임내의 매크로 블록의 개수를 세어 화면 규격의 수보다 적거나 많으면 그 프레임에 오류가 발생한 것으로 판정하고, 적절한 위치에서 분할과 병합을 하여 오류를 복구한다⁴⁾. 그러나 이 방법은 한 화면의 여러 부분에 오류가 발생하여 VLC코드의 병합과 분할 현상이 동시에 발생하면, 오류를 검출 할 수 없게 된다. 다른 한 가지 많이 사용되고 있는 방법은 VLC 테이블에 의한 오류검출 방법으로서 디코딩한 여러 가지 코드워드들이 참조 테이블에 존재여부에 따라 오류를 검출한다⁶⁾. 이 방법은 비교적 효율적으로 오류를 검출할 수 있다. 그러나 오류로 인하여 한 코드워드가 다른 유효한 코드워드로 바뀌면 즉, 오류가 발생한 코드가 여전히 참조 테이블에 존재 하면 오류를 발견할 수 없다. VLC테이블에 의한 오류 검출 알고리즘은 한 프레임에서 발생한 오류를 발견하지 못하면 그 다음 프레임에 전파된 오류에 대해서는 검출할 수 없다. 이렇게 전파된 오류는 인트라 프레임(Intra Frame)이 도착할 때까지는 시간적으로나 공간적으로 계속 전파되게 된다.

2. 제안한 오류검출 알고리즘

본 연구에서 제안한 오류 검출 알고리즘은 기존의 부호화기를 그대로 사용하며, 비트 스트림에 임의의 부가적인 정보도 추가하지 않는다. 제안한 오류 검출 알고리즘은 동영상의 시·공간적인 유사성을 이용하여, 수신된 비트스트림에 대한 디코딩이 끝난 후, 화소영역에서 매크로 블록의 화소 변화가 시간적으로나 공간적으로 급변하는 블록을 오류로 판정하였다. 알고리즘은 미처 발견되지 않은 오류가 그 다음 프레임에 전파되는 경우에도 잘 적용할 수 있다.

오류 검출 알고리즘은 구체적으로 다음과 같은 4개 단계로 이루어진다.

단계 1: 현재 프레임의 한 매크로 블록의 SSD (Summation of Squared Difference)를 계산한다.

SSD는 한 매크로 블록이 그전 프레임의 대응하는 매크로 블록과의 유사정도를 나타내는 척도로서, 그림 1에서 볼 수 있는바와 같이 현재 프레임과 이전 프레임의 대응하는 매크로 블록의 주변 화소들의 절대 차의 합을 말한다. 그림 1에서 N은 휘도신호 Y에 대해서 N=15이며, 색차신호 Cb, Cr에 대해

서 N=7이다.

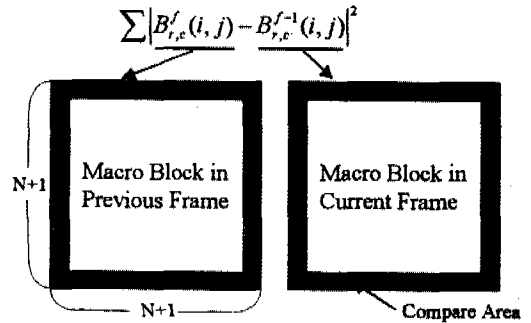


그림 1. SSD의 계산 방법
Fig. 1. The calculate method of SSD

f와 r,c를 각각 프레임 번호와 매크로 블록 좌표라 하고, $B'_{r,c}(i,j)$ 를 f번째 프레임의 r행 c열의 매크로 블록 내에서 i,j를 좌표로 하는 화소의 밝기 값이라면, 식 (1)에 의해 구할 수 있다.

$$SSD'_{r,c} = \sum_{j=0}^{N-1} |B'_{r,c}(0,j) - B'^{-1}_{r,c}(0,j)|^2 + \sum_{j=0}^{N-1} |B'_{r,c}(i,N) - B'^{-1}_{r,c}(i,N)|^2 + \sum_{j=0}^{N-1} |B'_{r,c}(N,j) - B'^{-1}_{r,c}(N,j)|^2 + \sum_{j=0}^{N-1} |B'_{r,c}(i,0) - B'^{-1}_{r,c}(i,0)|^2 \quad (1)$$

SSD는 1개 매크로 블록내의 주변에 있는 화소들과 그 전 프레임의 대응하는 화소의 절대 차의 합으로서 한 매크로 블록이 그 전 프레임의 대응하는 매크로 블록에 비해 변화가 크면 큰 값을 나타내게 되고, 변화가 적으면 상대적으로 작은 값을 나타내게 된다.

단계 2: $SSD'_{r,c} > TH1$ 인 매크로 블록을 찾는다.

여기서 TH1은 오류블록을 검출하기 위한 임계값이다. 임계값의 결정에 대한 문제는 다음절에서 상세히 설명하려한다. 어느 한 매크로 블록에 오류가 발생하면, 그전 프레임과 비교할 때 유사도가 떨어져 큰 SSD값을 나타내기에 오류를 검출해낼 수 있다. 그러나, 움직임이 큰 영역에서도 전후프레임의 상관도가 떨어져 큰 SSD값을 나타내게 되기 때문에 오류가 아닌 블록을 오류로 오 검출하는 현상이 나타날 수 있다. 따라서 큰 SSD값을 갖는 매크로 블록들에 대해 오류에 의한 것인지 아니면 움직임에 의한 것인지 아래의 3, 4단계를 거쳐 정확히 구분하여야 한다.

단계 3: 두 번째 단계에서 찾아낸 매크로 블록들에 대하여 경계 정합(Boundary Matching)을 실행한다.

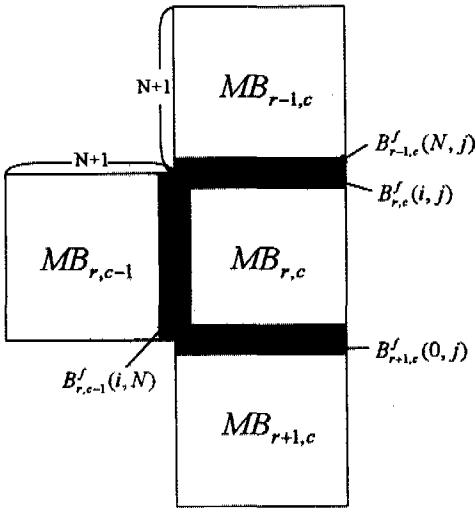


그림 2. BMC의 계산 방법
Fig. 2 The calculate method of BMC

경계정합은 그림 2에서 처럼 한 매크로 블록의 주변 화소들이 그 주위의 매크로 블록들의 인접된 화소와의 정합정도를 나타낸다. 한 매크로 블록에 오류가 발생하면 공간적인 유사성이 떨어지게 되며, 경계정합계수가 큰 값을 나타내게 된다. 블록정합계수는 식 (2)을 이용하여 구할 수 있다.

$$\begin{aligned}
 BMC'_{r,c} = & \sum_{j=0}^N |B'_{r,c}(0,j) - B'_{r-1,c}(N,j)|^2 \\
 & + \sum_{j=0}^N |B'_{r,c}(N,j) - B'_{r+1,c}(0,j)|^2 \\
 & + \sum_{i=1}^N |B'_{r,c}(i,0) - B'_{r,c-1}(i,N)|^2
 \end{aligned} \quad (2)$$

단계 4: 두 번째 단계에서 찾아낸 매크로 블록들에서 $BMC'_{r,c} > TH2$ 인 매크로 블록을 찾아 오류로 판정한다. 여기서 $TH2$ 는 임계값이다. 임계값의 결정은 다음절에서 논한다.

움직임이 큰 영역에서도 한 프레임내의 주변블록과의 유사도는 오류가 발생했을 때보다 크기에, $BMC'_{r,c} > TH2$ 인 매크로 블록을 오류로 판정할 수 있다.

3. 임계값의 결정

단계 2와 단계 4에서의 임계값은 다음과 같은 방

법으로 결정할 수 있다. $TH1$ 의 계산에 참가하는 화소수 n 은 1개 매크로 블록의 주변의 픽셀개수이며, 휘도 신호 Y 에 대해서는 60이고, 색차신호 Cb, Cr 에 대해서는 28이다. $TH2$ 에 대해서 각각의 화소수는 주변의 매개 블록과의 인접한 픽셀의 개수이며, 휘도신호 Y 와 색차신호 Cb, Cr 에 대해서 각각 48과 24이다. $TH2$ 에 대해서는 왼쪽 상하 두 모서리의 픽셀은 계산에 두 번 참가하게 된다. 전후 프레임간의 대응하는 화소차는 주로 화면의 움직임 정도와 프레임율에 의해 결정된다. 따라서 본 연구에서는 임계값을 식 (3)과 같은 방법으로 결정하였다.

$$THi = (Aver_i \times 1.5)^2 \times n_i \quad (i=1,2) \quad (3)$$

여기서 $Aver_i$ 는 화소차의 평균값이고, n 은 위에서 설명한 계산해야 할 화소의 수이다. $Aver_i$ 는 $TH1$ 에 대해서 한 매크로 블록의 주변 화소들의 대응하는 프레임간의 평균 화소차이며, $TH2$ 에 대해서는 주변 매크로 블록의 인접 화소들의 화소차의 평균값을 나타낸다.

그림 3은 위의 4개 단계 오류 검출 알고리즘의 흐름도를 보여 준다.

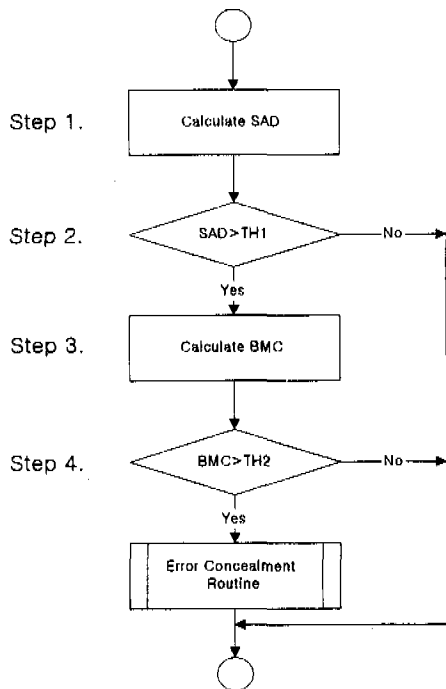


그림 3. 오류검출 알고리즘의 블록도
Fig. 3 The block diagram of error detection algorithm

IV. 실험 및 고찰

본 연구에서는 알고리즘의 효율성을 평가하기 위하여, ITU-T H.263 부호기를 사용하여 모의 실험을 하였다.^[15] 실험에서 사용된 영상은 QCIF포맷의 150프레임의 SUSIE 원 영상을 2프레임의 스킵으로 각각 15FPS와 10FPS로 압축하여, 50프레임의 압축 시퀀스를 얻었다. 오류 검출의 정확성을 판단하기 위해, 각각의 부호화된 스트림의 지정한 매크로 블록에 비트오류(Bit Error)와 연집오류(Burst Error)를 삽입하였다. 오류의 비트오류율(BER)은 48kbps에서 1E-4, 24kbps에서 1E-5로 하였다. 그림 4는 본 연구에서 사용한 테스트 모델의 블록도를 보여주고 있다.

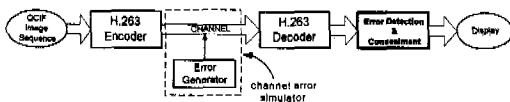


그림 4. 실험에서 사용한 테스트 모델의 블록도
Fig. 4. The block diagram of test model

그림 4에서 알 수 있는 바와 같이 인코딩 된 후의 비트열은 채널시뮬레이터를 통하여 오류가 삽입되며, 수신단에서 디코딩된 후 오류 검출 및 은닉처리를 하고 화면에 디스플레이 된다.

본 연구에서 제안한 오류검출 알고리즘의 성능을 정량적으로 분석하고자, 기존의 VLC 테이블에 의한 오류검출 방법으로 같은 입력시퀀스에 대해 오류검출을 하여 그 결과를 비교 분석하였다^[6].

또한, 오류 검출 성능을 영상에서 직관적으로 보여주기 위해 두가지방법으로 검출해낸 오류에 대해 블록경계정합 오류은닉 알고리즘을 사용하여 은닉하였으며^[11-13], 은닉처리를 한 후의 영상의 PSNR을 계산하여 화질 개선 효과를 비교 평가하였다.

표 1. 임계값과 임계값의 계산에 사용한 데이터
Table 1. Thresholds and data to calculate thresholds

		Aver	M	임계값
TH1	Y	15	60	30000
	Cb/Cr	15	28	15000
TH2	Y	20	48	45000
	Cb/Cr	20	24	20000

표 1에는 임계값 TH1과 TH2를 계산하기 위한 평균 화소차 등 데이터를 보여주고 있으며, 표 2에

는 표 1의 데이터를 이용하여 얻은 임계값과 각 단계에서 임계값에 의해 검출된 블록들을 보여주고 있다.

표 2. 기존 방법과 제안한 방법의 검출 성능 비교
Table 2. Comparison of performance for two algorithms

입력영상		VLC테이블에 의한 오류검출 방법으로 검출해낸 매크로 블록	제안한 오류검출 방법으로 검출해낸 매크로 블록	
오류 유형 및 BER	오류 삽입 위치(매크로 블록)		단계 2	단계 4
랜덤 오류	48kbps BER=1E-4	Frame 14 : 18, 36, 83, 85 Frame 19 : 52, 57, 71, 79	Frame 14 : 18, 36, 83, 85 Frame 19 : 52, 57, 69, 71, 79, 81, 90	Frame 14 : 18, 36, 83, 85 Frame 19 : 52, 57, 71, 79
	24kbps BER=1E-5	Frame 15 : 17, 77, 85, 90 Frame 34 : 50, 56, 71, 82	Frame 15 : 17, 77, 85, 90 Frame 34 : 50, 56, 71, 79, 82, 91	Frame 15 : 17, 77, 85, 90 Frame 34 : 50, 56, 71, 82
연집 오류	48kbps 15 FPS	Frame 14 : 18, 37, 83, 90 Frame 17 : 12, 39, 71, 85	Frame 14 : 18, 37, 79, 83, 90 Frame 40 : 12, 27, 39, 41, 69, 71, 85	Frame 14 : 18, 37, 83, 90 Frame 17 : 12, 39, 71, 85
	24kbps 10 FPS	Frame 13 : 37, 53, 81, 86 Frame 27 : 23, 37, 46, 81	Frame 13 : 37, 53, 81, 86 Frame 27 : 23, 37, 46, 57, 78, 81	Frame 13 : 37, 53, 81, 86 Frame 27 : 23, 37, 46, 81

표 2는 동일한 입력 시퀀스에 대한 기존 알고리즘과 제안한 알고리즘의 오류 검출 결과를 보여준다. 기존 방법과 제안한 방법을 비교하여 볼 때, VLC테이블에 의한 오류검출 방법은 4가지 입력 영상에 대해 모두 검출해내지 못한 매크로 블록이 존재하나 제안한 오류 검출 방법은 정확하게 오류가 발생한 매크로 블록을 찾아냄을 볼 수 있다.

또한, 표 2의 제안한 알고리즘의 단계 2와 단계 4를 보면, 48kbps, BER=1E-4일 때, Frame 14에는 움직임이 큰 영역이 없기에 TH1과 TH2에 의해 검출된 매크로 블록이 같으며, 48kbps, BER=1E-4일 때, Frame 17에서 매크로 블록 69, 81, 90과

24kbps, BER=1E-5일 때, Frame 19에서 매크로 블록 58, 78, 91, Frame 24에서 매크로 블록 57, 78은 화면에서 움직임이 가장 큰 부분에 위치한 블록들이다. 이런 움직임이 큰 영역내의 매크로 블록은 앞 절의 분석에서와 같이 전후프레임사이의 상관도가 낮기에 TH1에 의해 오류로 오검출 되었다. 그러나 TH2에 의한 주변블록과의 정합정도의 비교로 이러한 오검출된 매크로 블록들은 배제되고 오류블록들만 오류은닉을 하게된다. 그림 5는 랜덤오류, 48kbps, BER=1E-4인 영상의 19번째 프레임의 휘도신호 Y를 예로 오류검출과정의 단계 2에서의 SSD값의 분포와 그에 대응하는 임계값 TH1의 위치, 그리고 단계 4에서의 BMC값의 분포와 그에 대응하는 임계값 TH2의 위치를 각각 보여주고 있다.

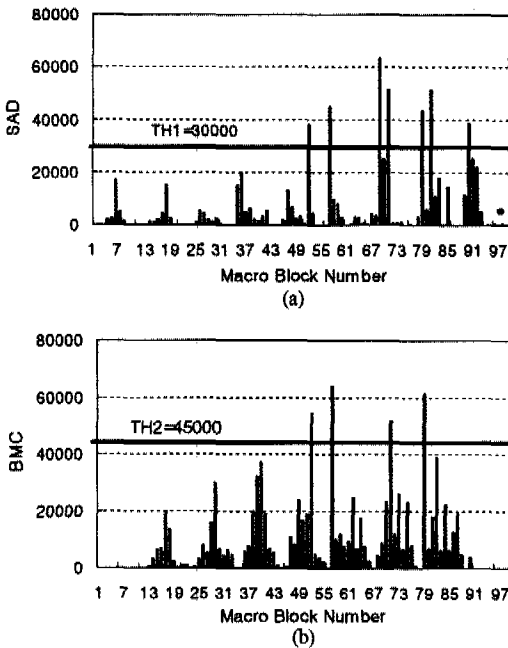


그림 5. (a) SSD의 분포도와 임계값의 위치
(b) BMC의 분포도와 임계값의 위치
Fig. 5. (a) The distribution of SSD and the position of thresholds
(b) The distribution of BMC and the position of thresholds

그림 6에는 랜덤오류, 48kbps, BER=1E-4인 영상에 대하여 제안한 알고리즘을 적용하였을 때, 검출된 오류블록을 은닉한 후의 영상을 보여주고 있다. SUSIE영상의 14번째 프레임과 19번째 프레임이 오류가 발생했을 때와 은닉 결과를 대조적으로 보여주고 있다. 그림 6(b)의 오른쪽 그림에서 볼 수 있

는 바와 같이 화면 중간 부분에서 GOB동기신호의 상실로 인해 화면에 분열(split)현상이 나타남을 볼 수 있다. 그리고 은닉후의 영상 그림 6(c)오른쪽 그림을 보면 비록 분열현상은 여전히 존재하지만 블록 오류들은 은닉이 잘 되었음을 볼 수 있다.

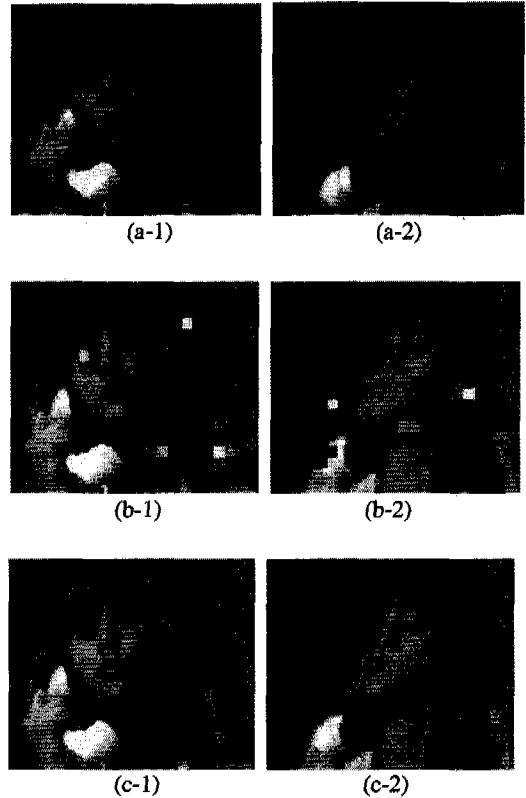


그림 6. (a-1, 2) SUSIE 원 영상의 14번째 프레임과 19번째 프레임
(b-1, 2) 오류가 발생한 영상의 14번째 프레임과 19번째 프레임
(c-1, 2) 은닉 후 영상의 14번째 프레임과 19번째 프레임
Fig. 6. (a-1, 2) 14th frame and 19th frame of susie original sequence
(b-1, 2) 14th frame and 19th frame of error occurred sequence
(c-1, 2) 14th frame and 19th frame of concealed sequence

그림 7에는 이전 프레임에서 검출되지 않은 오류들이 그 다음 프레임들에 확산되었을 경우 제안한 알고리즘의 검출 효과를 보여준 것이다. 그림에 보여준 영상은 오류 검출 성능을 보여주고자 연접오류, 24kbps, 10FPS의 영상에서 오류가 발생한 후 두 프레임 전파시킨 영상과 이 영상에 대하여 검출 및 은닉을 한 영상이다. 그림은 어떤 원인으로 인하여 오류가 검출되지 않고 그 다음 프레임에 전파되

었을 경우, 전파된 오류에 대해서도 효율적으로 검출됨을 보여준다.

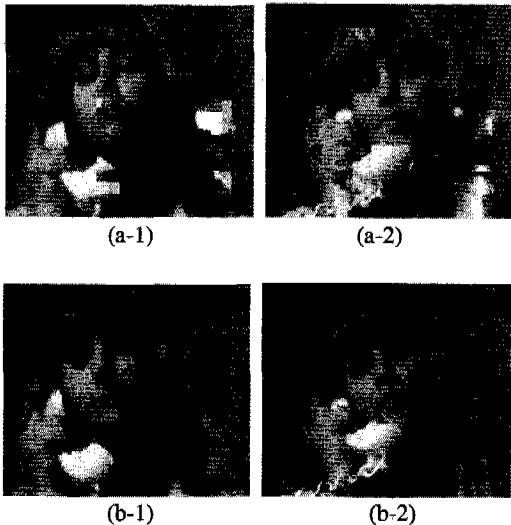


그림 7. (a-1, 2) 오류가 두 프레임 전파 된 영상
(b-1, 2) 전파된 오류에 대해 검출 및 은닉한 영상
Fig. 7. (a-1, 2) the erroneous images of two frame propagation
(b-1, 2) the images after conceal detected error

오류 검출 성능을 객관적으로 평가하고자, 영상의 PSNR을 구하여 비교하였다. 영상의 PSNR은 다음과 같은 방법으로 구하였다.

원 영상에서 한 화소를 $p(i, j)$, 오류 삽입후의 영상에서의 한 화소를 $p_e(i, j)$ 라면, 평균 제곱 오류는 식 (4)으로 나타낼 수 있으며,

$$rmse = \sqrt{\frac{1}{M \times N} \sum_{i=0}^M \sum_{j=0}^N [p(i, j) - p_e(i, j)]^2} \quad (4)$$

PSNR은 다음 식 (5)과 같이 구할 수 있다.

$$PSNR = 10 \log \left(\frac{255}{rmse} \right) \quad (5)$$

QCIF포맷에서 M,N은 각각 176, 144이다.

그림 8에는 각 입력 영상에 대하여 두가지 알고리즘으로 오류검출 및 은닉을 한 영상의 PSNR비교를 보여 준다. 매개 그림에서 그래프①은 오류가 삽입된 실험영상의 PSNR이고, 그래프②는 VLC테이블에 의한 오류검출 방법의 PSNR을 표시하고, 그래프③은 제안한 알고리즘에 의한 오류검출 방법의 PSNR을 보여준다. 그림에서 볼 수 있는바와 같이 제안한 알고리즘은 기존의 VLC테이블에 의한 오류검출 방법보다 우수한 성능을 보여, 은닉후의 화질

이 약 15dB정도로 더 향상됨을 알 수 있다.

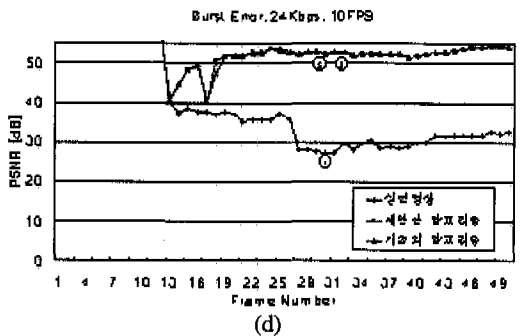
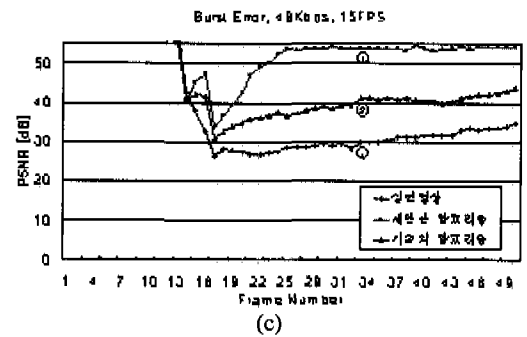
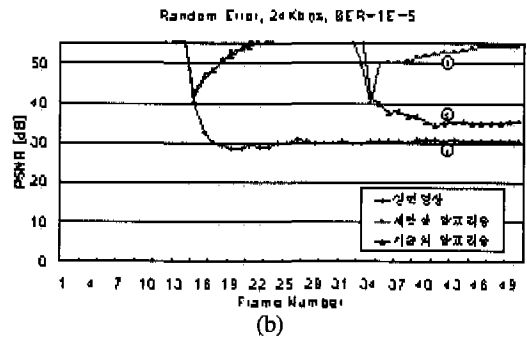
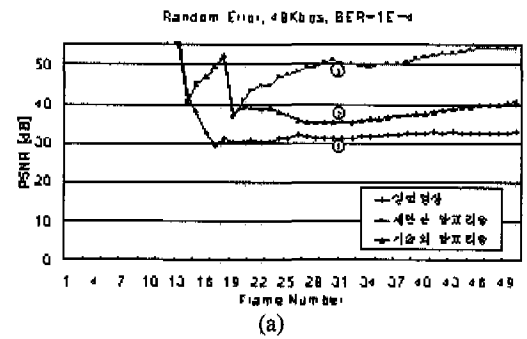


그림 8. (a, b, c, d). 각 입력영상과 두가지 방법으로 검출 및 은닉한 후의 영상의 PSNR 비교
Fig. 8 (a, b, c, d). PSNR of erroneous sequence and concealed sequence

V. 결론

본 논문에서는 H.263 동영상 압축 전송에서 수신 단에서의 채널 잡음에 의한 블록 오류의 검출 알고리즘을 제안하였다. 오류 검출 알고리즘은 매크로 블록의 시·공간적 유사성을 이용하여, 이전 프레임의 대응하는 매크로 블록의 유사성과 주변블록들과의 정합특성을 가지고 오류블록을 검출하였다. 제안된 알고리즘의 효율성은 검증하기 위해, H.263부호기를 사용하고, 입력 영상은 Susie 시퀀스를 사용하였으며, 부동한 오류 환경에서 검출 실험을 하였다. 실험에서 사용한 휘도신호와 색차신호에 대한 임계값은 오류 검출 2단계에서의 TH1은 60000, 30000으로 하였으며, 4단계에서의 TH2는 50000과 25000으로 하였다.

실험결과는 제안된 알고리즘이 채널에서 발생한 블록오류를 효율적으로 검출해낼 수 있으며, 기존의 VLC테이블에 의한 오류 검출 방법보다 우수한 성능을 보여, 은닉후의 화질이 약 15dB정도로 더 향상됨을 보여 주었다.

향후 연구 과제는 비트스트림의 계층 헤더의 오류 즉, 오류의 크기가 블록이나 GOB보다 더 큰 오류에 대한 검출방법과 그에 따르는 오류 은닉 기법이라고 보아진다.

참고 문헌

[1] Ralf Schafer and Thomas Sikora, "Digital Video Coding Standards and Their Role in Video Communications", Proc. IEEE, Vol. 83, No. 6, pp.907-924, Jun. 1995.

[2] Haibo Li, Astrid Lundmark and Robert Forchheimer, "Image sequence coding at very low bitrates : A Review", IEEE Trans. on Image Processing, Vol. 3, No. 5, pp.589-609, Sep. 1994.

[3] O.Robert Mitchell and Ali J. Tabatabai, "Channel error recovery for transform Image Coding", IEEE Trans. on Commun. Vol. Com-29, No. 12, pp.1754-1762, Dec. 1981.

[4] W. M. Lam and A. R. Reibman, "An error concealment algorithm for image subject to channel errors", IEEE Trans. on Image Processing, Vol. 4, No. 5, pp.

533-542, May. 1995.

[5] Jian Feng, Kwok-Tung Lo and Hassan Mehrpour, "Error concealment for MPEG video transmissions", Contributed Paper, IEEE, pp. 183-187, Mar. 1997.

[6] S.Aign and K.Fazel, "Temporal & Spatial error concealment techniques for hierarchical MPEG2 video codec", IEEE 0-7803-2486-2/95, pp.1778-1783, 1995.

[7] J. W. Park, D. S. Kim and S. U. Lee, "The error concealment technique for DCT based image coding", IEEE 0-7803-1775-0/94, III.293-III.296, 1994.

[8] J. S. Hwang et. al "Block Error Concealment Algorithm for Compressed Video Bitstream", Telecommunication Review, pp.263-275, Vol. 8, No. 2, 1998.

[9] ITU-T Rec. H.263, "Video coding for low bitrate communication", Mar. 1996.

[10] ISO/IEC JTC1/SC29/WG11/N802, ITU-T H.262, ISO/IEC 13818-2 Draft International Standard, "Generic coding of moving pictures and associated audio", Video, Nov. 1994.

[11] M. J. Chen, L. G. Chen and R. M. Weng, "Error concealment of lost motion vectors with overlapped motion compensation", IEEE Trans. on Circuit and Systems for Video Technology, Vol. 7, No. 3, pp. 560-563. June. 1997.

[12] W. M. Lam, A. R. Reibman and B. Liu, "Recovery of lost or erroneously received motion vectors", ICASSP, Vol. 5. pp.417-420, 1993.

[13] W. Keck, "A Method for Robust Decoding of Erroneous MPEG-2 Video Bitstreams", IEEE Trans on Consumer Electronics, Vol. 42, No. 3, Aug. 1996.

[14] J. W. Kim, J. W. Park and S. U. Lee, "Channel error detection and concealment techniques for the MPEG2 video standard", 韓國通信學會論文誌 Vol. 21, No. 10, pp.2563-2578, 1996.

[15] Telenor Research, H.263 Simulation Software, ftp://bonde.nta.no/pub/tmn/software.

최 세 하(Se-Ha Choi)

정회원



1975년 2월 : 명지대학교 전자공학과 졸업
1988년 8월 : 연세대학교 산업대학원 전자공학과 석사학위 취득
1991년 8월~1995년 8월 : 청주대학교 일반대학원 전자공학과 박사과정 수료

현재 : 정보통신부 정책실 기술기준과 통신사무관
<주관심분야> 정보통신 표준화, 영상처리 등임

이 주 신(Joo-Shin Lee)

정회원

1948년 1월 : 17일생 (사진:1997년 제22권 3호 참조)
1975년 2월 : 명지대학교 전자공학과 졸업
1977년 2월 : 홍익대학교 대학원 전자공학과(공학석사)
1986년 2월 : 명지대학교 대학원 전자공학과(공학박사)
1979년 9월~현재 : 청주대학교 전자공학과 교수
<주관심분야> 영상처리, 멀티미디어 등임