

## 퍼지 추론을 이용한 영상은닉 알고리즘

# An Image Concealment Algorithm Using Fuzzy Inference

김하식\*, 김윤호\*\*

Ha-Sik Kim\*, Yoon-Ho Kim\*\*

### 요 약

본 논문에서는 비디오 코덱의 수신단 블록 오류를 퍼지추론을 이용하여 검출하고 영상을 은닉하는 방법을 제안하였다. 제안한 블록 오류 검출 알고리즘은 인접된 두 프레임에서 서로 대응되는 블록들 간의 시간적 유사성을 이용하여 SSD를 구하고, 1차 임계값보다 SSD가 큰 블록들을 1차적인 오류 블록으로 분류하였다. 그리고 각각의 파라미터를 가지고 퍼지데이터 구한 후에 비례상수  $\alpha$ 와 임계값 TH1과 TH2를 결정하였다. 제안된 알고리즘의 타당성을 검토하기 위하여 QCIF 동영상에 랜덤 오류를 삽입하여 오류 검출 및 은닉 실험을 하였으며, 알고리즘의 성능평가는 동영상에 오류를 삽입한 후 기존의 VLC 테이블에 의한 오류 검출 알고리즘과 검출 결과를 비교 분석하였다. 실험 결과, 제안한 오류 검출 알고리즘은 실험 영상의 오류 블록들을 모두 검출할 수 있었으며, 오류 은닉 후 영상의 화질이 기존의 오류 검출 알고리즘 보다 15dB 이상 개선됨을 알 수 있었다.

### Abstract

In this paper, we propose the receiver block error detection of the video codec and the image concealment algorithm using fuzzy inference. The proposed error detection and concealment algorithm gets SSD(Summation of Squared Difference) and BMC(Boundary Matching Coefficient) using the temporal and spatial similarity between corresponded blocks in the two successive frames. Proportional constant,  $\alpha$ , for threshold value, TH1 and TH2, is decided after fuzzy data is generated by each parameter. To examine the propriety of the proposed algorithm, random errors are inserted into the QCIF Susie standard image, then the error detection and concealment performance is simulated. To evaluate the efficiency of the algorithm, image quality is evaluated by PSNR for the error detection and concealed image by the existing VLC table and by the proposed method.

In the experiment<sup>1)</sup> results, the error detection algorithm could detect all of the inserted error, the image quality is improved over 15dB after the error concealment compare to existing error detection algorithm.

Key words : BMC, Fuzzy inference, SSD, QCIF

### I. 서 론

최근 영상통신 분야의 이용 증가로 인한 상호 호환성 유지를 위해 영상통신 부호화와 관련된 국제 표준안 제정이 활발히 진행되고 있으며, 영상부호화 알

고리즘들은 응용 분야에 따라 전송 비트율이 서로 다르게 국제 표준안이 제안되고 있다. 특히 영상통신 서비스는 협소한 채널 용량과 높은 전송 오류율로 인하여 고 압축율을 실현하기 위한 부호화 알고리즘이 요구되며 전송오류를 고려하여 영상품질 저하를 최

\* 강원관광대학 관광정보처리과(Department of Tourism Information Processing, Kangwon Tourism College)

\*\* 목원대학교 컴퓨터공학부(Division of Computer Engineering, Mokwon University)

· 제1저자 (First Author) : 김하식

· 접수일자 : 2007년 9월 29일

소화시키는 기술이 요구되고 있다[1]-[3]. 일반적으로 문자나 음성 데이터는 전송시 에러가 발생되면 재 전송을 요구할 수 있지만 영상의 경우에는 데이터양이 방대하기 때문에 재전송이 곤란하여 사람의 눈에 크게 거슬리지 않을 정도의 에러는 허용이 가능하다. 이러한 영상신호의 전송특성을 고려하여 채널에러에 대응하기 위한 화질 개선 방법으로 에러 복원(Error recovery)기법과 다양한 에러은닉(Error concealment) 기법들이 연구되어지고 있다[4]-[8]. 오류 검출 및 은닉기법에는 주파수 영역에서 DCT(Discrete Cosine Transform) 계수를 이용한 방법과 공간영역이나 시간영역에서 화소 데이터를 이용하는 방법이 있다. 특히, 저전송율 동영상 전송에서는 압축율과 실시간성, 하드웨어 구현의 용이성 등을 고려하여 오버헤드의 추가나 복잡한 계산이 요구되지 않는 간단한 방법의 의한 에러검출 및 처리가 필요하다[9]-[14].

본 논문에서는 비디오 코덱의 수신단 블록 오류를 퍼지추론을 이용하여 검출하고 영상을 은닉하는 방법을 제안하였다. 제안한 블록 오류 검출과 은닉 알고리즘은 시-공간적 유사성을 고려한 화소 변화량 분포특성을 이용하여 오류를 검출하였으며, 인접된 두 프레임에서 서로 대응되는 블록들간의 시간적 유사성을 이용하여 SSD(Summation of Squared Difference)를 구하고, 1차 임계값 TH1보다 SSD가 큰 블록들을 1차적인 오류 블록으로 분류하였다. 그리고 1차적인 오류로 분류된 매크로 블록들에 대해 공간적 유사성을 이용하여 BMC(Boundary Matching Coefficient)를 구한 후, 2차 임계값 TH2에 의해 주변 블록들보다 화소값 변화가 큰 블록을 오류 블록으로 판정하였다. 인터프레임간에 움직임이 큰 프레임 또는 에러 프레임을 검출하기 위하여 퍼지 클러스터링(Fuzzy clustering)을 이용한 퍼지추론 알고리즘을 적용하였다. 제안한 오류 검출 알고리즘의 타당성을 검토를 위해 QCIF Susie 표준 영상에 랜덤 오류를 삽입하여 오류 검출 및 은닉 실험을 하였으며, 알고리즘의 성능 평가를 위해 기존의 VLC 테이블에 의해 오류 검출 및 은닉한 영상과 제안한 방법으로 오류를 검출 및 은닉한 영상에 대해 PSNR을 계산하여 화질을 비교 평가하였다.

## II. 본 론

### 2-1. 코덱의 수신단 블록 오류 검출 알고리즘

본 연구에서는 비디오 코덱의 수신단을 사용하여 영상의 블록오류를 검출함으로써 비트 스트림에 임의의 부가적인 정보를 추가하지 않는다는 장점이 있다. 제안한 영상의 블록 오류 검출 알고리즘은 다음과 같이 4단계로 이루어졌다.

단계 1: 현재 프레임의 각 매크로 블록들에 대하여 SSD를 계산한다.  $f$ 와  $r, c$ 를 각각 프레임 번호와 매크로블록 좌표라 하고,  $B_{r,c}^f(i,j)$ 를  $f$ 번째 프레임의  $r$ 행  $c$ 열의 매크로 블록 내에서  $i, j$ 를 좌표로 하는 화소의 밝기값 이라면, 식 (1)에 의해 구할 수 있다.

$$\begin{aligned} SSD_{r,c}^f = & \sum_{j=0}^{N-1} |B_{r,c}^f(0,j) - B_{r,c}^{f-1}(0,j)|^2 \\ & + \sum_{i=0}^{N-1} |B_{r,c}^f(i,N) - B_{r,c}^{f-1}(i,N)|^2 \\ & + \sum_{j=1}^N |B_{r,c}^f(N,j) - B_{r,c}^{f-1}(N,j)|^2 \\ & + \sum_{i=1}^N |B_{r,c}^f(i,0) - B_{r,c}^{f-1}(i,0)|^2 \end{aligned} \quad (1)$$

단계 2:  $SSD_{r,c}^f > TH1$ 인 매크로블록을 찾아 1차적인 오류 블록으로 분류한다.

단계 3: 단계 2에서 찾아낸 매크로블록들에 대하여 BMC를 계산한다.

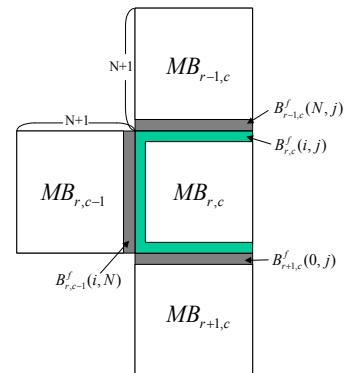


그림 1. BMC의 계산 순서  
Fig 1. BMC calculation step

BMC 는 그림 1에서와 같이 한 매크로 블록의 주변 화소들이 그 주위의 매크로 블록들의 인접된 화소와의 정합정도를 나타낸다. BMC는 식 (2)를 이용하여 구할 수 있다.

$$\begin{aligned}
 BMC_{r,c}^f = & \sum_{j=0}^N |B_{r,c}^f(0,j) - B_{r-1,c}^f(N,j)|^2 \quad (2) \\
 & + \sum_{j=0}^N |B_{r,c}^f(N,j) - B_{r+1,c}^f(0,j)|^2 \\
 & + \sum_{i=1}^N |B_{r,c}^f(i,0) - B_{r,c-1}^f(i,N)|^2
 \end{aligned}$$

단계 4: 단계 2에서 분류된 매크로 블록들에서  $BMC_{r,c}^f > TH2$  인 매크로 블록을 찾아 오류 블록으로 검출한다. 여기서  $TH2$  는 임계값이다.

그림 2는 블록 오류 검출 알고리즘의 흐름도를 나타낸 것이다.

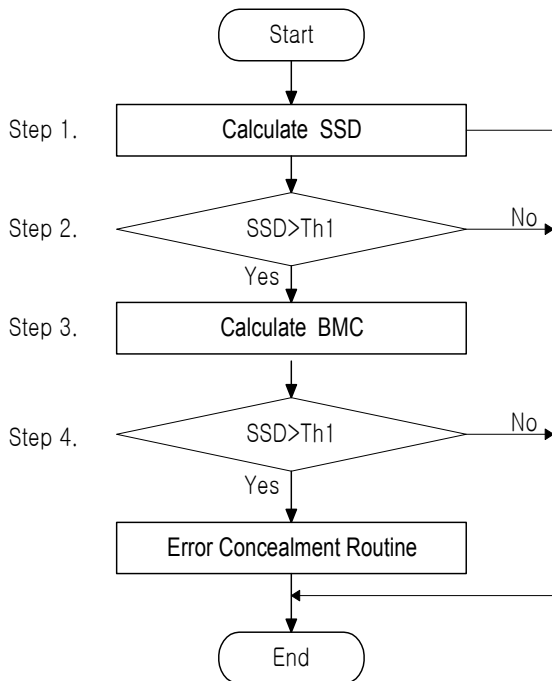


그림 2. 블록오류 알고리즘의 흐름도  
Fig 2. Block diagram of BE algorithm

2-2. 퍼지데이터를 이용한 임계값의 결정

단계 2와 단계 4에서의 사용한 임계값 TH1과 TH2는 식 (3)에 의해 구할 수 있다.

$$THk = \frac{1}{99} \sum_{i=1}^{99} [(Aver_i \times \alpha)^2 \times M] \quad k=1,2 \quad (3)$$

여기서  $TH1$  의 계산에 참가하는 화소수 M은 한 개 매크로블록의 주변의 픽셀개수이며,  $TH2$  에 대해서 각각의 화소 수는 주변의 매개 블록과의 인접한 픽셀의 개수이다. 그리고  $Aver_i$  는  $TH1$  에 대해서 한 매크로블록의 주변 화소들의 대응하는 프레임간의 평균 화소차이며,  $TH2$  에 대해서는 주변 매크로블록의 인접 화소들의 화소차의 평균값을 나타낸다. 또한 인터 프레임간에 움직임이 큰 프레임 또는 에러 프레임을 검출하기 위하여 퍼지 클러스터링(Fuzzy clustering)을 이용한 퍼지추론 알고리즘을 적용하였으며, 첫째 프레임에서 SSD를 구한 후, 퍼지화 데이터를 생성하여 멤버십 함수를 그림3과 같이 만들고, 두 번째 프레임에서 SSD를 추출 후, 퍼지화 데이터를 생성하여 멤버십 함수를 만들어 그림4와 같이 두 멤버십 함수를 추론하기 위한 생성 규칙을 만든 후, Max-min 연산에 의해 각각의  $\alpha$ 값을 결정하였다. 멤버십 함수를 결정하기 위한 알고리즘은 다음과 같다.

단계1: SSD를 구한 다음 각 프레임에 대한 SSD의 평균을 구한다.

단계2: 각 프레임에서 SSD의 최소값과 최대값을 구한다.

단계3: 각 프레임에 대한 SSD 최소값과 최대값 사이를 10단계로 나눈다.

단계4: 등급 5의 값을 평균값으로 정한다.

단계5: 등급 0은 0으로 결정한다.

단계6: 최대값의 경우 이전 프레임과의 매크로블록 화소차 평균값이 30 이하일 경우에는 오류가 발생하지 않은 경우로 가정하여 현재 프레임의 SSD 최대값이 50,000 미만일 경우에는 등급 10을 현재 프레임의 최대값으로 정하고, 50,000 이상일 경우에는 등급 10을 최대값/2로 결정한다.

단계7: 이전 프레임을 기준 함수로 하여 현재 프레임의 퍼지함수 중앙값이 이전 프레임의 각 등급의 값들과 비교하여 최소값을 갖는 등급을 현재 프레임의 퍼지값을 1로 하여 정규화 한다.

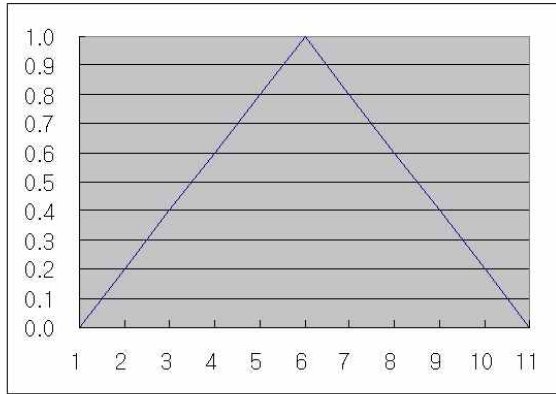


그림 3. 퍼지 멤버십 함수  
Fig 3. Fuzzy membership function

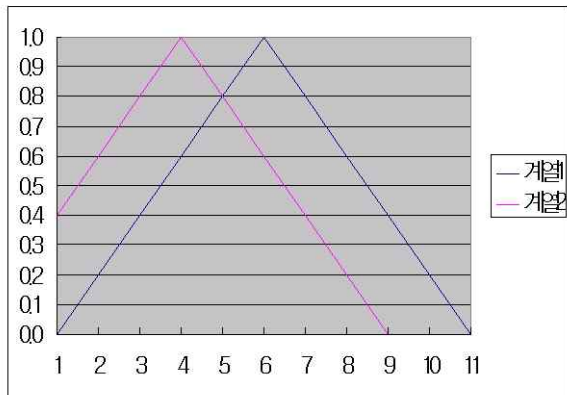


그림 4. 연속된 프레임의 퍼지 멤버십 함수  
Fig 4. FMF of successive frame

### III. 실험 및 고찰

본 연구에서는 제안한 기법의 타당성을 검증하기 위하여 QCIF포맷의 150프레임의 SUSIE 원 영상을 2 프레임의 스킵으로 각각 15FPS와 10FPS로 압축하여, 50프레임의 압축시퀀스를 얻었다. 식(3)에서  $TH1$  과  $TH2$  를 구하기 위해서 표1과 같이 데이터를 사용하였다.  $M$ 은 휘도 신호  $Y$ 에 대해서는 60이고, 색 차신호  $Cb, Cr$ 에 대해서는 28이다.  $TH2$ 에 대해

서 매개 블록과의 인접한 픽셀의 개수는 휘도신호  $Y$ 에 대해서는 48이고  $Cb, Cr$ 에 대해서는 24이다.

표 1. 임계값과 임계값 파라미터

Table 1. Threshold value and it's parameter

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

표 2. 단계 2와 단계 4에서 검출된 오류 블록

Table 2. Error blocks detected in step 2-4

오류삽입위치	단계 2에서의 오류블록	단계 4에서의 오류 블록
Frame 14 : 18, 36, 83, 85	Frame 14 : 18, 36, 83, 85	18, 36, 83, 85
	Frame 15 : 47, 70, 71, 80, 81, 82, 92, 93	
	Frame 16 : 81, 82, 92, 93	
	Frame 17 : 68, 81	
	Frame 19 : 52, 57, 71, 79	52, 57, 71, 79
Frame 19 : 52, 57, 71, 79	Frame 21 : 82, 94, 95	
	Frame 22 : 81, 82, 83, 95	83
	Frame 23 : 71, 72, 73, 81, 82, 83	

표 2는 제안한 알고리즘의 단계2와 단계4에서 검출된 오류 블록들을 나타내고 있다. 그림 5는 susie 표준영상의 14번째 프레임과 19번째 프레임의 원영상과 랜덤 오류 삽입 후의 영상과 제안한 알고리즘에 의해 오류 검출 후 은닉한 영상을 보여주고 있다. 그림 5(b)의 오른쪽 그림에서 볼 수 있는 바와 같이 화면 중간 부분에서 GOB동기신호의 상실로 인해 화면

에 분열(split)현상이 나타남을 볼 수 있다. 그리고 은닉후의 영상 그림 5(c)오른쪽 그림을 보면 비록 분열 현상은 여전히 존재하지만 블록 오류들은 은닉이 잘 되었음을 볼 수 있다.



(a-1)



(a-2)



(b-1)



(b-2)



(c-1)

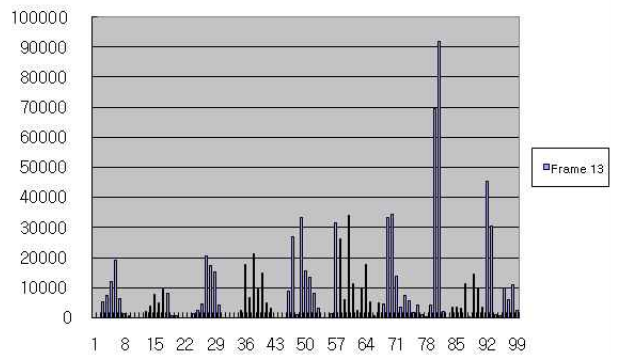


(c-2)

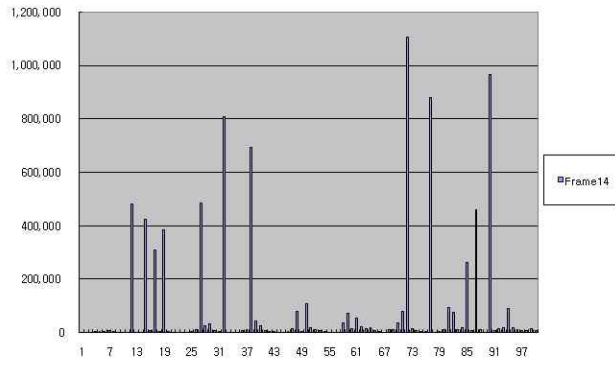
그림 5. (a-1,2) SUSIE 원 영상의 14번째 프레임과 19번째 프레임, (b-1,2) 오류가 발생한 영상의 14번째 프레임과 19번째 프레임, (c-1,2) 은닉 후 영상의 14번째 프레임과 19번째 프레임

Fig. 5. SUSIE image; (a-1,2) 14's frame and 19's frame

그림 6(a)와 그림6(b)는 Frame 13과 Frame 14에서 오류 검출의 정확성을 판단하기 위해, 각각의 부호화된 스트림의 지정한 매크로 블록에 랜덤 오류를 삽입한 후 SSD값을 히스토그램으로 표현 한 것이다. 그림7은 13번째 프레임과 매크로 블록에 랜덤 오류를 삽입한 14번째 프레임간의 맵버셉 함수에 의해 오류 프레임을 검출한 것이다.



(a)



(b)

그림 6. (a) Frame13의 히스토그램 (b) Frame14의 히스토그램

Fig 6. Histogram; (a) frame 13, (b)frame 14

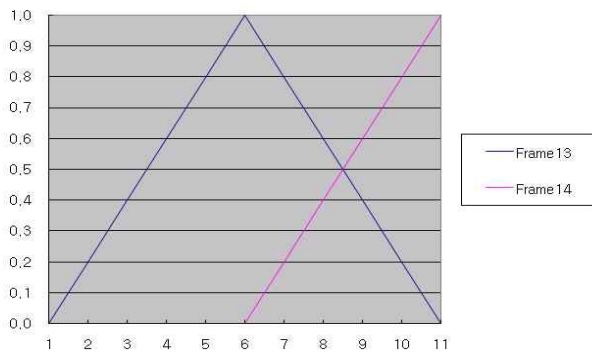


그림 7. Frame13과 Frame14의 멤버쉽 함수

Fig 7. Membership function in frame 13 and 14

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)로 나타낼 수 있으며, QCIF포맷에서 M,N은 각각 176, 144이다. PSNR은 다음 식(5)와 같이 구할 수 있다.

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

$$PSNR = 20 \log_{10} \left( \frac{255}{rmse} \right) \quad (5)$$

그림 8은 각 입력 영상에 대하여 두 가지 알고리즘으로 오류검출 및 은닉을 한 영상의 PSNR을 나타낸 것이다. 매개 그림에서 그래프①은 오류가 삽입된 실험영상의 PSNR이고, 그래프②는 VLC테이블에 의한 오류검출 방법의 PSNR을 표시하고, 그래프③은 제안한 알고리즘에 의한 오류검출 방법의 PSNR을 나타낸 것이다. 그림에서 볼 수 있는바와 같이 제안한 알고리즘은 기존의 VLC테이블에 의한 오류 검출 방법보다 우수한 성능을 보여, 은닉후의 화질이 약 15dB정도로 더 향상됨을 알 수 있다.

Random Error, 48Kbps, BER=1E-4

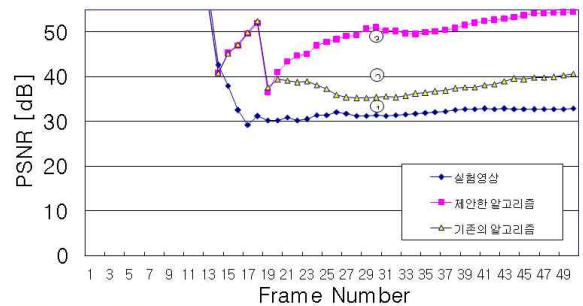


그림 8. 각 입력영상의 PSNR 비교

Fig 8. Comparison of PSNR in each images

#### IV. 결 론

본 논문에서는 동영상 압축 전송에서 수신단에서의 채널 잡음에 의한 블록 오류의 검출 알고리즘과 퍼지추론을 이용한 영상은닉 알고리즘을 제안하였다. 오류 검출 알고리즘은 매크로블록의 시공간적 유사성을 이용하여, 이전 프레임과 현재프레임간의 움직임 또는 에러프레임을 퍼지 추론에의 해 검출하였다. 오류블록 검출은 이전 프레임과 현재 프레임간의 매크로블록에 대한 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 오류블록 검출 알고리즘 단계 2에서 TH1은 60000과 30000으로 하였으며, 단계4에서 TH2는 50000과

25000으로 하였다.

제안된 알고리즘은 채널에서 발생한 블록오류를 효율적으로 검출해낼 수 있으며, 기존의 VLC 테이블에 의한 오류 검출 및 은닉 방법보다 우수한 성능을 보여, 은닉후의 화질이 약 15dB 정도로 더 향상됨을 보여 주었다. 향후 연구 과제는 비트 스트림의 계층 헤더의 오류 즉, 오류의 크기가 블록이나 GOB보다 더 큰 오류에 대한 검출방법과 그에 따르는 오류 은닉 기법이라고 보아진다.

### 참 고 문 헌

- [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", *ICA SSP*, 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.

## 김 윤 호 (金允鎬)



1986년 : 경희대학교 대학원 전자  
공학과 (공학석사)

1992년 : 청주대학교 대학원 전자  
공학과 (공학박사)

2005년 ~ 2006년 : Univ. of  
Auckland, NZ. Dept. of Computer  
Science, CITR Lab. Research

Fellow.

1992년 ~ 현재 : 목원대학교 컴퓨터공학부 교수, 멀티디  
어기술사.

관심분야 : 영상처리, 컴퓨터비전, 뉴로퍼지응용, 정보통  
신정책 등

## 김 하 식 (金河植)



1992년 : 청주대학교 대학원 전자  
공학과 (공학석사)

1999년 ~ 현재 : 청주대학교 대학  
원 전자공학과 (박사수료)

2000년 ~ 현재 : 강원관광대학 관  
광정보처리과 교수

관심분야 : 영상처리, 이동물체추

적, 컴퓨터비전 등