반화소 움직임 벡터를 이용한 적응적 비디오 워터마킹

류 권 열*

Adaptive Video Watermarking Using Half-cell Motion Vector

Kwon-yeol Ryu*

요 약

본 논문에서는 적응적 임계값과 반화소 움직임 벡터를 이용한 비디오 워터마킹 방법을 제안한다. 기존의 방법은 모든 매크로블록에 대해 워터마크 삽입 조건을 검색하고, 임의의 움직임 벡터에 워터마크가 삽입되므로 움직임 벡터 변경이 많고, 비가시성이 떨어진다. 제안한 방법은 *PMV*를 이용하여 임계값을 생성하고, 임계값보다 큰 움직임 벡터에 워터마크를 삽입하므로 움직임 벡터 변경 횟수가 적고, 비가시성이 향상된다. 실험결과 제안한 방법은 비디오 비트스트림에 따라 평균 5.4 dB ~ 9.3 dB 의 비가시성 특성이 향상됨을 알 수 있었다.

ABSTRACT

In this paper, I propose a video watermarking method using adaptive threshold value and half-cell motion vector. Conventional method cause a lot of change of motion vectors and bring about a falling-off in invisibility, because it search embedding condition about whole macroblocks and watermarks are embedded in arbitrary motion vectors. Proposed method improve invisibility and decrease in number of change of motion vectors, because it create threshold value by using power of motion vector(*PMV*), and watermarks are embedded in motion vectors which greater than threshold value. Consequently, proposed method improve invisibility with on an average 5.4 dB ~ 9.3 dB according to video bit-stream

키워드

video watermarking, motion vector, blind watermarking, adaptive threshold

Ⅰ. 서 론

디지털 워터마킹은 소유권을 주장할 수 있는 정보를 미디어에 삽입하여, 소유권을 인정받을 수 있게 함으로써 미디어의 불법 복제를 방지하는 기술이며, 이 과정에서 삽입되는 정보나 데이터가 워터마크이다. 워터마크는 비가시성(invisibility)과 강인성(robustness)을 가져야한다. 비가시성은 삽입된 워터마크가 원본 미디어에서 시각적, 청각적으로 인지할 수 없게 하는 특성이며, 강인성은 워터마크를 제거하려는 다양한 공격에 대해서도 삽입된 워터마크가 검출되게 하는 특성을 나타낸다[1-3]. 따라서 워터마크의 비가시성과 강인성은 워터마크를 평가하는 중요한 척도이다. 비디오 워터마킹은 비가시성과 강인성뿐만 아니라 데이터 량이 방대하기 때문에 빠른 처리와 원본 없이 검출이 가능한 블라인드 워터마킹이 가능하여야 하며, 기존의 동영상 압축표준들과 호환성을 유지하여야 한다.

비디오 워터마킹에 관한 연구는 공간상의 원 비디오에

* 위덕대학교 게임소프트웨어학부

접수일자 : 2006. 2. 14

워터마크를 삽입하는 방법, MPEG 부호화 과정에서 워터마크를 삽입하는 방법, 비트스트림에 워터마크를 삽입하는 방법으로 분류할 수 있다. 원 비디오에 워터마크를 삽입하는 방법은 화질 열화가 발생하기 쉬우며[4,5], MPEG 부호화 과정에서 DCT 계수를 조작하여 워터마크를 삽입하는 방법은 부호기가 변형되어야 하기 때문에 기존의 표준들과 호환성이 떨어진다[6-8]. 비트스트림에 워터마크를 삽입하는 방법은 호환성을 유지하기 위해 삽입되는 워터마크 데이터 량의 제한을 받는다[9-11]. 움직임 벡터(motion vector)에 워터마크를 삽입하는 방법은 움직임 벡터에서 얻은 정보와 워터마크 정보의 상관관계에 따라 워터마크를 처리하는 방법으로 비트스트림을 변경하지 않기 때문에 기존의 표준들과 호환성이 좋으며, 비트스트림의 증가 없이 워터마크를 삽입할 수 있어 최근 많이 연구되고 있다[12,13].

Zhongjie 등은 반화소 단위 움직임 벡터의 수평, 수직 성분으로부터 특징벡터를 추출하고, 특징벡터와 워터마크 정보와의 4가지 상관관계에 따라 움직임 벡터 값을 변경함으로써 워터마크를 삽입하였다. 이 방법은 각 프레임의 모든 움직임 벡터에 대해 순차적으로 워터마크 처리과정이 적용되므로 움직임 벡터 변경 횟수가 많고, 비가시성이 떨어진다. 또한 워터마크가 삽입되지 않은 경우에도 블라인드 검출을 위해 움직임 벡터를 변경해야 하는 문제점이 있다.

따라서 본 논문에서는 각 프레임의 움직임 벡터에서 임계값을 생성하여, 임계값보다 큰 움직임 벡터에만 워터마크를 삽입함으로써 워터마크의 비가시성을 향상시키고, 움직임 벡터의 수평, 수직 성분 중에서 큰 성분에 워터마크를 삽입함으로써 블라인드 검출을 위한 불필요한 처리과정을 제거하는 비디오 워터마킹 방법을 제안한다. 본 논문의 구성은 2장에서 기존의 움직임 벡터 기반 비디오 워터마킹에 대해 기술하고, 3장에서 제안한 적응적 비디오 워터마킹에 대해 설명한다. 4장 및 5장에서는 제안한 방법에 대한 실험결과 및 결론을 나타낸다.

Ⅱ. 움직임 벡터를 이용한 기존의 워터마킹

동영상 압축 표준에서 다수의 프레임 내에 존재하는 움직임 벡터는 시간적 중복성을 제거하기 위한 효과적인 도구일 뿐만 아니라, 워터마크를 삽입하기 위한 효과적인 요소이다. 이 방법들은 기존의 동영상 압축 표준들과 호환성이 좋으며, 비트스트림의 증가 없이 워터마킹이 가능한 장점이 있기 때문에, 최근 움직임 벡터를 이용한 워터마킹 방법들이 많이 연구되고 있다. MPEG-2 부호화에서 B 프래임과 P 프래임은 16x16 매크로블록 단위로 하나 또는 두개의 움직임 벡터를 갖는다. 움직임 벡터를 이용한 기존의 워터마킹 방법은 그림 1과 같다.

step1 : Begin with the first GOP, $i = 0$
step2 : Compute feature vector h_i, v_i
$h_i = \mathrm{mod}(2H_i, 2)$, $v_i = \mathrm{mod}(2V_i, 2)$
step3 : Compute p_i of (h_i, v_i)
if $h_i = 0$ & $v_i = 0$
then insert watermark, $p_i = 0$
if $h_i = 1$ & $v_i = 0$
then $H_i = H_i + 0.5$, $p_i = 1$
if $h_i = 0$ & $v_i = 1$
then $V_i = V_i + 0.5$, $p_i = 2$
if $h_i = 1$ & $v_i = 1$
then not changed, $p_i = 3$
step4 : Insert watermark
if $w_k = 0$ then $H_w = H_i + 0.5$
if $w_k = 1$ then $V_w = V_i + 0.5$
step5 : $i = i+1$, repeat step2 to step4
until all bits of w_k are embedded.
step6 : Do next GOP

그림 1. 기존의 방법에 따른 워터마크 삽입 순서도
Fig. 1 Watermark embedding procedure according to conventional method.

그림 1에서 i는 매크로블록, H_i 및 V_i는 움직임 벡터의 수평 및 수직 방향 크기를 나타낸다. p_i는 특징벡터 클래스를 나타내며, w_k는 워터마킹 정보를 나타낸다. 그림 1의 단계 2에서 특징벡터 h_i 및 v_i는 움직임 벡터가 반화소인 경우는 1이 되고, 정화소인 경우는 0이 된다. 그림 1에 나타나는 바와 같이 기존의 워터마크 삽입 방법은 움직임 벡터의 수평, 수직 성분으로부터 특징벡터를 구하고, 4가지로 분류된 특징벡터 클래스 p_i에 따라, $p_i = 3$ 인 경우는 움직임 벡터를 변경하지 않으며, $p_i = 0$ 인 경우는 삽입되는 워터마크 정보에 따라 0이면 움직임 벡터의 수평 성분을 변경하여 $p_i = 1$ 이 되도록 설정하며, 워터마크 정보가 1이면 움직임 벡터의 수직 성

분을 변경하여 $p_i=2$ 가 되도록 설정한다. $p_i=1$ 또는 $p_i=2$ 인 경우는 워터마크는 삽입하지 않지만, 블라인드 워터마크가 되도록 하기위해 $p_i=0$ 이 되도록 움직임 벡터의 수평 또는 수직 성분을 변경한다.

워터마크를 검출하는 방법은 수신된 각각의 움직임 벡터로부터 특징벡터를 계산하여, 특징벡터 클래스가 $p_i=0$ 또는 $p_i=3$ 이면 워터마크 정보가 없는 것으로 처리하고, $p_i=1$ 이면 워터마크 정보가 0인 것으로 처리하고, $p_i=2$ 이면 워터마크 정보가 1인 것으로 처리한다. 이러한 기존의 방법은 각 프래임의 모든 움직임 벡터에 대해 순차적으로 워터마크 처리과정이 적용되고, 워터마크가 삽입되지 않은 경우에도 블라인드 워터마크를 위해 움직임 벡터를 변경해야 하므로 비가시성이 떨어지고 처리시간이 많이 걸리는 단점이 있다.

Ⅲ. 제안한 비디오 워터마킹

3.1. *PMV*를 이용한 적응적 임계값 생성

움직임 벡터가 큰 매크로블록은 해당 객체의 빠른 움직임을 나타내며, 움직임 벡터의 변경에 따른 시각적인 특성이 움직임 벡터가 적은 매크로블록에 비해 덜 민감하다. 따라서 본 논문에서는 시각적 특성과 워터마크 정보의 삽입을 보장하기 위해 움직임 벡터의 크기가 큰 매크로블록부터 워터마크를 삽입함으로써 워터마크의 비가시성을 향상시키는 방법을 제안한다. 제안한 방법에서는 각 프래임을 16x16 크기의 매크로블록으로 분할하고, 분할된 각 매크로블록에 대한 움직임 벡터의 크기를 계산한 후, 그림 2와 같이 *PMV*에 대한 누적빈도를 이용하여 임계값을 생성한다. *PMV*는 식 (1)과 같다.

$$PMV_i=\sqrt{(2\times H_i)^2+(2\times V_i)^2},\quad 0<i<MB \qquad (1)$$

식 (1)에서 H_i 및 V_i는 i번째 매크로블록에 대한 움직임 벡터의 수평 및 수직 성분을 나타내고, 움직임 벡터의 각 성분들은 반화소 단위로 표현되기 때문에 정수로 표현하기 위해서 2를 곱하여 PMV_i를 계산한다.

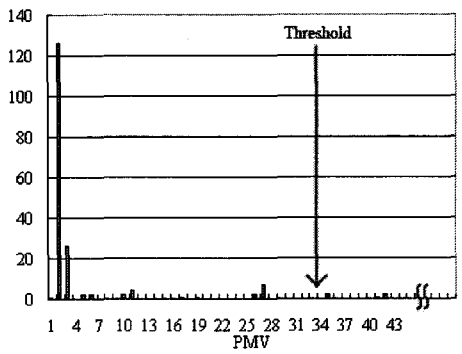


그림 2. PMV 히스토그램를 이용한 임계값
Fig. 2 Threshold value using PMV histogram.

그림 2에서 수평 및 수직 축은 PMV와 빈도수를 나타내며, 임계값은 PMV 누적 빈도수의 큰 값을 기준으로 하여 모든 워터마크 정보를 포함하는 값으로 설정한다.

임의의 k 번째 프래임에서 워터마크 삽입을 위해 선택된 매크로블록 E_k는 식 (2)와 같다.

$$E_k=M_k\cdot F_k,\quad M_k=\begin{cases}1, & PMV_i\geq T\\ 0, & PMV_i<T\end{cases} \qquad (2)$$

식 (2)에서 F_k는 프래임, E_k는 선택된 매크로블록, M_k는 임계값보다 큰 PMV를 갖는 매크로블록을 나타낸다.

3.2. 워터마크 삽입

제안한 방법에서 워터마크 삽입 방법은 PMV가 임계값보다 큰 움직임 벡터의 수평 및 수직 성분에 대한 위상각도 Θ를 식 (3)과 같이 계산하고, 특징벡터와 워터마크 정보로 구성되는 삽입 조건에 따라 움직임 벡터를 변경한다.

$$\Theta_i=\arctan\frac{V_i}{Hi}\quad (i\in E_k) \qquad (3)$$

식 (3)에서 V_i 및 H_i는 움직임 벡터의 수평 및 수직 성분을 나타낸다. 워터마크 삽입 순서는 그림 3과 같다.

그림 3에 나타나는 바와 같이 첫째, 움직임 벡터 수평,

수직 성분의 위상각도 Θ가 45° 보다 적거나 같고, 워터마크 정보와 특징벡터 값이 다를 때 움직임 벡터의 수평 성분을 변경한다. 둘째, 움직임 벡터 수평, 수직 성분의 위상각도 Θ가 45° 보다 크고, 워터마크 정보와 특징벡터 값이 다를 때 움직임 벡터의 수직 성분을 변경한다. 셋째, 움직임 벡터 변경 확률이 1/2보다 클 때 워터마크 정보를 반전하여 움직임 벡터 변경 확률을 1/2 미만으로 한다.

step1 : Begin with the first GOP, $i = 0$
step2 : Compute PMV_i
step3 : Compute threshold value and select motion vector to watermark
step4 : Compute phase angle Θ_i of MV
step5 : Insert watermark
in case of $\Theta_i \leq 45°$
if $h_i \neq w_k$ then $H_w = H_i + 0.5$
if $h_i = w_k$ then not changed
in case of $\Theta_i > 45°$
if $v_i \neq w_k$ then $V_w = V_i + 0.5$
if $v_i = w_k$ then not changed
if *changable probability* $\geq \frac{1}{2}$
then inverse transform of watermark
step6 : $i = i + 1$, repeat step4 to step5 until all bits of w_k are embedded.

그림 3. 제안한 방법에 따른 워터마크 삽입 순서도
Fig. 3 Watermark embedding procedure according to proposed method.

제안한 방법에서는 움직임 벡터의 변경을 최소화하기 위해 식 (4)와 같이 움직임 벡터의 수평, 수직 성분에 대한 특징벡터 ρ를 계산하고, 식 (5)와 같이 특징벡터와 워터마크 정보가 일치하지 않을 확률(α)을 조사한 후, α 값이 1/2 이상일 때는 워터마크 정보를 반전시킴으로써 α 값을 1/2 미만이 되도록 한다. 따라서 움직임 벡터가 변경될 확률을 1/2 미만으로 유지하게 되므로 비가시성이 향상된다.

$$\rho[i] = \mathrm{mod}(2h_i, 2) \oplus \mathrm{mod}(2v_i, 2), \quad i \in E_k \tag{4}$$

$$\alpha = \sum_{i=1}^{N} \rho[i] \oplus w[i] / N \tag{5}$$

식 (5)에서 w_k는 워터마크 정보를 나타내며, 직교성을 갖는 하다마드(hadamad) 행열을 사용한다. 제안한 방법에 따른 비디오 워터마크 삽입 구성도는 그림 4와 같다.

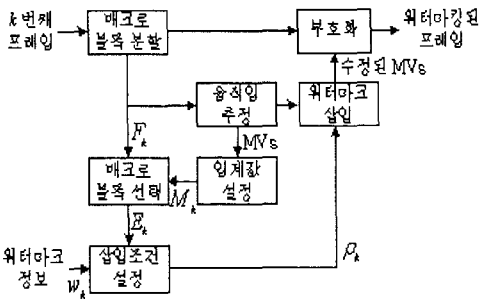


그림 4. 제안한 방법에 따른 워터마크 삽입 구성도
Fig. 4 Watermark embedding diagram according to proposed method.

3.3. 워터마크 검출

워터마크 정보의 검출 과정은 삽입 과정과 같이 프래임을 일정한 크기의 매크로블록으로 분할하고, 분할된 매크로블록에 대해 움직임 벡터와 PMV를 이용한 임계값을 생성한다. 임계값을 이용하여 워터마크가 삽입된 움직임 벡터를 선택하고, 그림 5와 같이 워터마크를 검출한다.

그림 5에 나타나는 바와 같이 워터마크 검출은 움직임 벡터의 수직, 수평 성분 중에서 큰 값의 특징벡터 값을 계산하여 결과 값을 워터마크 정보로 검출한다. 검출 조건에 의해 구해진 워터마크 정보는 식 (6)을 이용하여 유사도을 측정한다.

step1 : Begin with the first GOP, $i = 0$
step2 : Compute PMV_i
step3 : Compute threshold value and select motion vector to watermark
step4 : Compute phase angle of MV
step5 : Retrieving watermark
in case of $\Theta_i \leq 45°$
$w_i = \mathrm{mod}(2H_i, 2)$
in case of $\Theta_i > 45°$
$w_i = \mathrm{mod}(2V_i, 2)$
step6 : $i = i + 1$, repeat step2 to step5

그림 5. 워터마크 검출 순서도
Fig. 5 Watermark retrieving procedure.

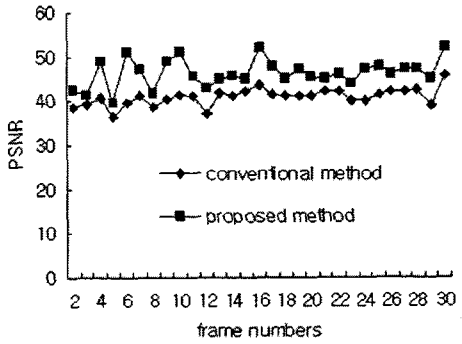
그림 5에 나타나는 바와 같이 워터마크 검출은 움직임 벡터의 수직, 수평 성분 중에서 큰 값의 특징벡터 값을 계산하여 결과 값을 워터마크 정보로 검출한다. 검출 조건에 의해 구해진 워터마크 정보는 식 (6)을 이용하여 유사도을 측정한다.

$$sim(W, \hat{W}) = ABS\left(\sum_{i=0}^{n-1} w[i] \cdot \hat{w}[i] \right) / \sum_{i=0}^{n-1} \hat{w}^2[i] \quad (6)$$

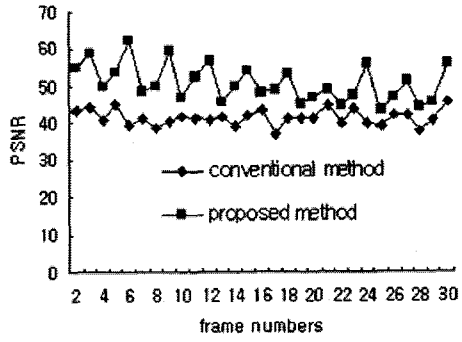
식 (6)에서 $w[i]$ 및 $\hat{w}[i]$ 는 원래 삽입된 워터마크 정보와 검출된 워터마크 정보를 나타낸다. 워터마크 삽입 과정에서 워터마크 정보가 반전되었을 경우 워터마크로 삽입된 하다마드 행렬의 특성에 의해 유사도 값이 음의 최대값을 갖기 때문에 $ABS()$ 함수를 이용함으로써 삽입 조건의 반전 유무에 관계없이 유사도를 측정한다.

Ⅳ. 실험 및 결과

제안한 방법에서 실험을 위한 비디오는 352×240 크기를 갖는 apollo와 sliding 비디오를 사용하였으며, apollo 비디오는 객체와 배경의 움직임이 빠른 것이며, sliding 비디오는 객체와 배경의 움직임이 느린 것이다. 움직임 벡터를 위한 매크로블록의 크기는 16×16으로 설정하였으며, 탐색영역은 -15~+15로 설정하였다. 비가시성의 척도는 PSNR을 사용하였으며, 강인성에 대한 척도는 식 (6)의 유사도를 사용하였다. 워터마크 정보는 하다마드 행렬을 이용하여 64개의 비트열를 생성하였고, 생성된 하다마드 열은 -1 과 1로 된 랜덤 비트 순열로 구성되어 있으나, 워터마크 정보를 0과 1의 비트열로 나타내기 위해 -1은 0으로 대치하였다. 반화소 단위 움직임 벡터를 이용한 비디오 워터마킹에서 제안한 방법에 대한 비가시성 실험결과는 그림 6과 같다.



(a) apollo video bit-streams

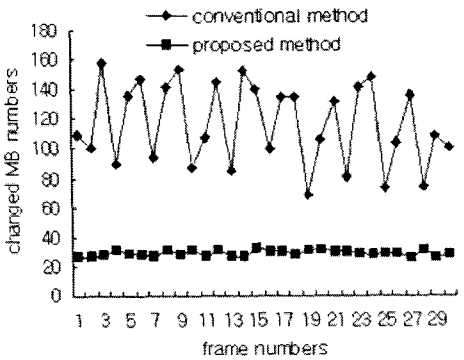


(b) sliding video bit-streams

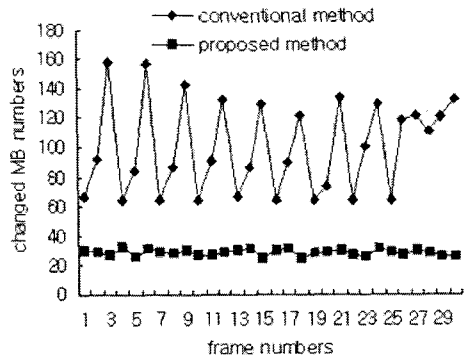
그림 6. 워터마크에 의한 비가시성 비교
Fig. 6 Comparison of invisibility by using watermark.

그림 6에 나타나는 바와 같이 apollo 비디오의 경우 프레임 평균 PSNR이 기존의 방법은 40.8 dB이고, 제안한 방법은 46.2 dB 이므로 워터마크 비가시성 특성이 기존의 방법에 비해 5.4 dB 향상됨을 알 수 있다. 또한 sliding 비디오의 경우 프레임 평균 PSNR이 기존의 방법은 41.3 dB이고, 제안한 방법은 50.6 dB 이므로 워터마크 비가시성 특성이 9.3 dB 향상됨을 알 수 있다.

본 논문에서는 워터마크 정보의 비가시성을 향상시키기 위해 하다마드 행열의 역변환을 이용함으로서 움직임 벡터의 변경을 최소화 하였다. 실험결과 각 프레임별 변경된 움직임 벡터 수는 그림 7과 같다.



(a) apollo video bit-streams

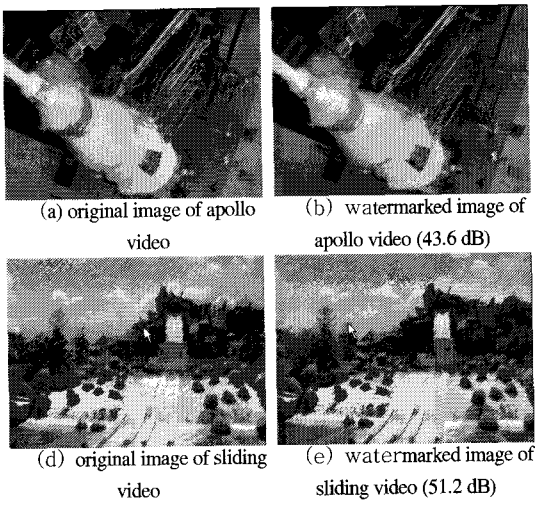


(b) sliding video bit streams

그림 7. 변경된 움직임 벡터 수 비교
Fig. 7 Comparison of changed motion vector numbers

그림 7에 나타나는 바와 같이 프래임별 움직임 벡터 평균 변경 횟수가 apollo 비디오 경우, 기존의 방법이 116회, 제안한 방법이 29회 이므로 움직임 벡터 변경이 87회 감소하였고, sliding 비디오 경우, 기존의 방법이 99회, 제안한 방법이 28회이므로 움직임 벡터 변경이 71회 감소하였다. 제안한 방법은 영상에 따라 71회에서 87회의 움직임 벡터 변경 특성이 개선됨을 알 수 있다.

제안한 방법에서 워터마크 삽입에 따른 영상의 시각적인 실험결과는 그림 8과 같다. 그림 8에서 (a) 및 (b)는 apollo 비디오의 원 영상 및 워터마크가 삽입된 영상을 나타나고, (c) 및 (d)는 sliding 비디오의 원 영상 및 워터마크가 삽입된 영상을 나타낸다.



(a) original image of apollo video
(b) watermarked image of apollo video (43.6 dB)
(d) original image of sliding video
(e) watermarked image of sliding video (51.2 dB)

그림 8. 제안한 방법에 따라 워터마크가 삽입된 영상
Fig. 8 Watermark embedded image according to proposed method.

그림 8에 나타나는 바와 같이 원 영상에 대한 워터마크가 삽입된 영상의 화질은 apollo 영상이 43.6 dB, sliding 영상이 51.2 dB를 나타내며, 시각적으로 워터마크의 비가시성 특성이 우수함을 알 수 있다.

V. 결 론

본 논문에서는 적응적 임계값과 반화소 움직임 벡터를 이용한 비디오 워터마킹 방법을 제안한다. 제안한 방법은 각 프래임의 움직임 벡터에서 PMV를 이용하여 임계값을 생성하고, 임계값보다 큰 움직임 벡터에 워터마크를 삽입함으로써 워터마크의 비가시성을 향상시키고, 움직임 벡터의 수평, 수직 성분 중에서 큰 성분에 워터마크를 삽입함으로써 블라인드 검출을 위한 불필요한 처리과정을 제거한다. 움직임 벡터의 변경을 최소화하기 위해 움직임 벡터의 수평, 수직 성분에 대한 특징벡터와 워터마크 정보가 일치하지 않을 확률(α)을 조사한 후, α 값이 1/2 이상일 때는 워터마크 정보를 반전시킴으로써 α 값을 1/2 미만이 되도록 한다. 실험결과 워터마크 삽입에 따른 프래임별 움직임 벡터 변경 특성이 기존의 방법에 비해 71회에서 87회 개선되고, 비디오 비트스트림에 따라 평균적으로 5.4 dB ~ 9.3 dB 의 비가시성 특성이 향상됨을 알 수 있었다.

참고문헌

[1] G. C. Langelaar, I. Setyanwan, and R. L. Lagendijk, "Watermarking digital image and video data," signal Processing Magazine, IEEE, Vol. 17, No. 5, pp. 20-46, 2000.

[2] M. D. Swanson, M. Kobayash, and A. H. Tewfik, "Multimedia data-embedding and watermarking technologies," Proceedings of IEEE, Vol. 86, No. 6, pp. 1064-1087, 1998.

[3] G. Voyatzis, N. Nikolaides, and I. Pitas, "Digital watermarking: an overview," EUSIPCO, Vol. I, pp.9-12, 1998.

[4] M. Barni, F. Bartolini, R. Caldelli, A. D. Rosa and A. Piva, "A robust watermarking approach for raw video," PV2000 10th International Packet Video Workshop in CD-ROM , 2000.

[5] J. Chae, and B. Manjunath, "Data hiding in video," ICIP, Vol. 1, pp. 311-315, 1999.

[6] A. Kusanagi, and H. Imai, "A method of watermarking for compressed video," SSCIS, 1999.

[7] D. Ghosh, K. and Ramakrishna, "Watermarking compressed video stream over Internet," APCC, Vol 2. pp. 711-715, 2003.

[8] M. Kutter, F. Jordan, and T. Ebrahimi, "Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video," Technical report M2281, ISO/IEC document, JTC1/SC29/ WG11, 1997.

[9] F. Hartung, and B. Girod, "Digital watermarking of MPEG2 coded video in the bit-stream domain," ICASSP, Vol. 4, pp. 2621-2624, 1997.

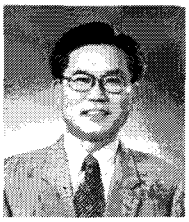
[10] Y. Dai, L. Zhang, and Y. Yang, "A new method of MPEG video watermarking technology," ICCT, Vol. 2. pp. 1845-1847, 2003.

[11] J. Zhang, H. Maitre, J. Li, and L. Zhang, Embedding watermark in MPEG video sequence," Multimedia Signal Processing, 2001 IEEE Fourth Workshop, pp. 535-540, 2001.

[12] Z. Zhao, N. Yu, and X. Li, " A novel video watermarking scheme in compressed domain based on fast motion estimation," ICCT, Vol. 2, pp. 1878-1882, 2003.

[13] Z. Zhongjie, J. Gangyi, Y. Mei, and W. Xunwei, "New algorithm for video watermark," ICSP, Vol. I, pp. 760-763, 2002.

저자소개



류 권 열(Kwon-yeol Ryu)

1982년 경북대학교 전자공학과 (공학사)
1990년 경북대학교 산업공학과 (공학석사)
1998년 부경대학교 전자공학과(공학박사)
1986년 ~ 1995년 포항공과대학교 전자계산소
1998년 ~ 현재 위덕대학교 게임소프트웨어학부 부교수
※관심분야 : 디지털영상처리, 멀티미디어정보보호, 3D 게임프로그래밍