

# 전역 마스크를 이용한 비디오 워터마킹 방법

문지영, 호요성, 이승욱\*

광주과학기술원 정보통신공학과

\*한국전자통신연구원 S/W 컨텐츠기술연구부

## Video Watermarking Method Using Global Masking

Jiyoung Moon, Yo-Sung Ho and Seung-Wook Lee\*

Kwangju Institute of Science and Technology (K-JIST)

\*Electronics and Telecommunications Research Institute (ETRI)

E-mail: {jymoon, hoyo}@kjist.ac.kr, tajinet@etri.re.kr

### 요약

본 논문에서는 워터마크의 기본적인 요구사항인 비가시성(invisibility)을 유지하면서 강인함(robustness)을 최대한으로 하기 위해 워터마크를 삽입할 때 인간시각 특성(Human Visual System, HVS)을 이용한다. 특히, 주파수 마스크(frequency masking), 공간 마스크(spatial masking), 그리고 움직임 마스크(motion masking)를 결합한 전역 마스크(global masking) 기법을 제안한다. 주파수 마스크는 프레임의 DCT 변환 후 주파수 민감도 표(frequency sensitivity table)를 이용해 위치 정보를 구한다. 공간 마스크는 밝기(luminance) 및 윤곽선(edge) 정보를 이용해서 구하며, 움직임 마스크는 이웃하는 프레임 간의 움직임 변화를 이용하여 프레임 간의 상관성을 고려한다. 본 논문에서는 키(key) 값에 의해 발생된 랜덤 시퀀스와 로고의 배타적 논리합(exclusive OR)에 의해 결합된 워터마크를 전역 마스크 결과에 따라 압축되지 않은 비디오 프레임에 직접 적용했으며, 실험을 통해 MPEG 부호화 및 재부호화 공격에 대해서 제안한 방법이 강인함을 확인했다.

### 1. 서론

최근 디지털 네트워크 등의 발달로 정지영상, 비디오, 혹은 오디오 같은 디지털 데이터의 사용이 급증하고 있다. 디지털 데이터는 아날로그 데이터에 비해 많은 장점을 제공한다. 즉, 디지털 데이터는 아날로그에 비해 더 높은 품질을 가지며, 원본과 똑같은 사본을 대량으로 생산할 수 있다. 또한 인터넷과 같은 네트워크 시스템을 통해 쉽게 전송될 수도 있다 [1]. 하지만 이러한 특징은 저작권 보호와 관련된 문제를 야기한다. 따라서 디지털 데이터를 보호하기 위한 연구가 활발히 진행중이며, 그 중의 하나가 디지털 워터마킹 방법이다.

디지털 워터마킹은 워터마크(watermark)라고 하는 특정 정보를 데이터에 삽입하고, 필요할 때 그 정보를 검출함으로써 디지털 데이터의 불법 복사 및 저작권

보호 등에 이용될 수 있다.

디지털 워터마킹 방법은 일반적으로 비가시성(invisibility), 강인성(robustness), 그리고 보안성(security) 등의 요구조건을 만족해야 한다. 즉, 워터마크는 눈으로 감지할 수 없을 정도로 작아서 원래 데이터의 품질을 저하시켜서는 안된다. 또한, 디지털 신호 처리나 의도적인 공격 등에 대해 강인해야 하며, 승인되지 않은 워터마크를 검출하거나 제거하는 것이 불가능해야 한다. 특히 비디오 워터마킹 방식의 경우, 워터마크는 이러한 일반적 요구 조건 뿐만 아니라 여러 형태의 비디오 압축 및 프레임율의 변화에도 강인해야 한다. 또한 다양한 프레임 공격에서도 검출이 가능해야 하며, 응용에 따라서는 실시간 전송이 가능해야 한다. 이러한 요구 조건을 만족하기 위한 다양한 워터마킹 방법이 현재 활발히 연구되고 있다 [1].

Hartung과 Girod는 MPEG-2 비트열에 워터마크를 삽입하고, 대역 확산(spread spectrum) 방식을 적용하였다 [2]. 워터마크는 원래 비디오 신호와 같은 차원의 의사잡음(pseudo noise, PN) 신호를 사용하여 발생된다. 각각의 압축된 비디오 프레임에 대해 해당되는 워터마크 신호 프레임이 8x8 블록 단위로 DCT 변환되고, 이것은 원래 비디오 프레임의 DCT 계수에 더해진다. 이 DCT 계수를 양자화한 후 허프만 부호화(Huffman coding)를 수행한다. 워터마크가 삽입된 비디오 시퀀스가 원래 비디오 시퀀스의 비트율을 넘지않게 하기 위해서 DCT 계수에서의 비트율이 증가하지 않은 경우에만 워터마크 계수가 전송된다. 그러나 이러한 방법은 비트율 제약 때문에 DCT된 워터마크 계수의 단지 몇 개만이 DCT된 비디오 프레임의 블록에 결합되며, 심지어 DC 계수만이 결합되는 경우도 있다. 그러므로 비트열 상의 워터마크 삽입 방법은 강인하지 않다.

Swanson은 압축되지 않은 비디오 데이터에 워터마크를 삽입하였다 [3]. 각각의 비디오 프레임에 시간축상의 웨이블릿 변환(temporal wavelet transform)을 수행하여 저주파 프레임과 고주파 프레임으로 나눈다. 이때 저주파 프레임은 비디오 프레임 중 움직임의 변화가 거의 없는 부분을 포함하며, 고주파 프레임은 움직임이

많은 부분을 포함한다. 각각의 부분에 워터마크를 삽입하는데, 이때 워터마크는 공간 및 시간 마스킹 효과를 고려해서 삽입된다. 그러나 이 방법은 비디오 프레임을 두 종류의 프레임으로 분리하기 위해 시간축을 따라 웨이블릿 변환을 실시하고, 또 웨이블릿 변환된 영상을 다시 DCT 변환을 시켜 워터마크를 삽입한 후 최종적으로 다시 역 웨이블릿 변환을 해야하므로 너무 복잡하다는 단점이 있다.

Zhu와 Zhang은 정지영상 워터마킹 방식을 그대로 비디오 워터마킹 방법에 적용하였다 [4]. 각각의 비디오 프레임에 웨이블릿 변환을 수행한 후 다중해상도(multi-resolution) 특성을 이용하여 고주파 웨이블릿 변수에 워터마크를 삽입하였다. 그러나 이러한 방법은 정지영상 워터마킹 방식을 그대로 비디오 워터마킹 방법에 적용함으로써, 비디오 프레임 간의 상관성을 고려하지 않았다.

그 외에 실시간 전송을 목적으로 하는 워터마킹 방법은 복잡성을 최소화해야 하기 때문에 상대적으로 강인성을 보장하지 않는다. 또한 다양한 비트율이나 부호화 방식 등에 적용될 수 없다.

본 논문에서는 다양한 부호화기나 비트율 등에 상관없는 일반적 워터마킹 기법을 위해 워터마크를 압축되지 않은 원래 비디오 데이터에 삽입한다. 또한 눈에 띄지 않으면서도 가능한 한 많은 워터마크를 비디오 프레임에 삽입하기 위해 인간시각 시스템(human visual system, HVS)을 이용한다.

그림 1은 제안한 방법의 전체적 구조를 나타낸다.



그림 1. 워터마크 방법의 전체적인 구조

우선 입력 비디오 프레임의 전역 마스킹(global masking)을 구하기 위해 HVS를 세가지 관점, 즉, 주파수 및 공간, 그리고 움직임 관점에서 고려한 후 그 결과를 결합한다. 이때 주파수 마스킹(frequency masking)은 프레임을 DCT 변환한 후 주파수 민감도(frequency sensitivity)를 이용해 얻은 위치 정보로 구한다. 공간 마스킹(spatial masking)은 프레임의 밝기(luminance) 및 윤곽선(edge) 정보를 이용해서 구하며, 움직임 마스킹은 이웃하는 프레임 간의 움직임 변화를 추출해서 구한다.

본 논문에서는 전역 마스킹 결과를 이용하여 가중치된 워터마크를 비디오 프레임에 삽입하며, 이때 원래 프레임과 워터마크가 삽입된 프레임의 Peak-Signal-to-Noise Ratio(PSNR)을 비교하여 워터마크 삽입량을 적절히 조절한다. 워터마크는 키(key) 값에 의해 발생된 랜덤 시퀀스와 로고 영상을 결합한 것으로, 여러 공격 등에 대해서 제안한 워터마크 추출 방법으로 로고 영상을 추출할 수 있음을 알 수 있다. 실험을 통해 제안한 방법이 MPEG 부호화 및 재부호화(re-encoding) 공격 등에 대해 강인함을 확인할 수 있다. 또한 전역 마스킹을 사용함으로써 HVS를 사용하는 다른 기존의 방법보다 훨씬 많은 양의 워터마크가 삽입됨을 알 수 있다.

## 2. 전역 마스킹

워터마킹 방법에서는 워터마크가 강인하면서도 비가시성을 유지하는게 중요하다. 즉, 눈에 덜 민감한 지역에 워터마크를 삽입하기 위해 HVS 특성을 이용한다. 인간의 눈은 영상의 주파수, 밝기 및 색상 정보에 따라 다르게 반응하므로, 이러한 HVS 특성을 이용하여 워터마크 삽입량을 조절할 수 있다.

인간의 눈은 밝기 차가 높은 영역에 삽입된 잡음에 덜 민감하다. 그리고 영상의 윤곽선에 가까운 부분에 삽입된 잡음은 일정한 밝기(intensity) 값을 가진 영역에 삽입된 잡음보다 눈에 덜 민감하다. 또한 주파수가 높은 영역에 삽입된 잡음에 덜 민감하며, 비디오 프레임의 경우 장면전환과 같은 부분에서는 잡음에 대한 민감도가 떨어지는 특성을 가지고 있다. 즉, 인간 눈은 움직임이 많은 부분에 삽입된 잡음에 덜 민감하다.

인간 눈의 지각 모델(perceptual model)은 일반적으로 마스킹 현상으로 설명된다. 마스킹이란 어떤 신호가 주위의 다른 신호에 대한 가시성 임계값(visibility threshold)을 넘는 경우 나타나는 특성을 말한다 [3]. 즉, 어떤 신호의 출현이 또 다른 신호의 존재를 눈에 띄지 않게 하는 현상을 설명한 것이다. 예를 들어, 주파수 마스킹은 한 주파수의 출현이 다른 주파수의 지각을 숨기는 현상이며, 밝기 마스킹이란 국소적 밝기가 명암(contrast) 변화를 숨기는 것이다. 그래서 이 마스킹 개념을 워터마킹 방식에 도입함으로써 효과적으로 워터마크를 눈에 띄지 않게 원래 데이터에 삽입할 수 있다.

지금까지 발표된 많은 논문에서 HVS를 이용한 워터마킹 방법이 제안되었지만, 대부분의 기존 방법들은 DCT 영역에 워터마크를 삽입하기 위해 주파수 마스킹 특성만을 이용하고 있다. 이러한 방법은 공간적 국지(localization) 특성을 고려하지 못한다. 즉, 주파수 마스킹만을 고려해 DCT 영역에 삽입된 워터마크는 전체 프레임 곳곳에 퍼지게 되고, 이는 곧 균일한 지역과 같이 마스킹 효과가 미비한 지역까지도 워터마크가 삽입됨을 뜻한다. 따라서 본 논문에서는 주파수 마스킹 뿐만 아니라 공간 마스킹 효과도 고려하였다. 또한 비디오 워터마킹 방식에서는 HVS가 인접 프레임 간 움직임 변화 부분에 삽입된 잡음에 덜 민감한 특성을 고려하였다.

### 2.1 주파수 마스킹

본 논문에서 사용한 주파수 마스킹은 Watson의 DCT 기반 시각 모델에 기반한다 [5]. 먼저 영상을 8x8 블록으로 나눈 후, 주파수 민감도 표(frequency sensitivity table)를 이용하여 마스킹을 계산한다. 주파수 민감도 표  $[i, j]$ 은 실험적으로 측정된 양으로서, 한 블록 내에서 마스킹 잡음이 작용하지 않을 때 인식되는 DCT 계수의 가장 작은 값을 나타낸다 [5].

그림 2는  $[i, j]$ 을 나타내는데, 이 값이 작을수록 인간시각 시스템이 그 주파수에 더 민감함을 의미하므로 저주파 부분보다 고주파 부분이 눈에 덜 민감함을 알 수 있다.

1.40	1.01	1.16	1.66	2.40	3.43	4.79	6.56
1.01	1.45	1.32	1.52	2.00	2.71	3.67	4.93
1.16	1.32	1.52	2.00	2.71	3.64	4.60	5.88
1.66	1.52	2.59	3.77	4.55	5.30	6.28	7.60
2.40	2.00	2.98	4.55	6.15	7.46	8.71	10.17
3.43	2.71	3.64	5.30	7.46	9.62	11.58	13.51
4.79	3.67	4.60	6.28	8.71	11.58	14.50	17.29
6.56	4.93	5.88	7.60	10.17	13.51	17.29	21.15

그림 2. DCT 주파수 민감도 표

가시성 임계값  $t_{th}[i, j]$  은 그 레벨 아래로는 신호가 보이지 않는 최소의 레벨을 나타내는 것으로,  $t[i, j]$  을 이용하여 아래와 같이 구할 수 있다.

$$t_{th}[i, j]_k = t[i, j] \left[ \frac{\tilde{T}[0,0]_k}{\tilde{T}_{avg}[0,0]} \right]^\alpha \quad (1)$$

위 식에서  $\tilde{T}[0,0]_k$  는 원래 영상의  $k$  번째 블록의 DC 계수이고,  $\tilde{T}_{avg}[0,0]$  는 영상의 DC 계수들의 평균값이며,  $\alpha$  는 0.649로 정한다. 식 (1)에서 알 수 있듯이, 블록의 평균 밝기가 크면 더 많은 양에 의해 DCT 계수가 변할 수 있다. 즉, 영상의 밝은 부분이 인식되지 않고 더 많은 변화를 받아들일 수 있음을 나타낸다. 이때 다른 주파수 신호의 작용으로 발생하는 마스크는 아래와 같다.

$$\tilde{F}[i, j]_k = t_{th}[i, j] \cdot \max \left\{ 1, \left[ \frac{|\tilde{T}[i, j]_k|}{t_{th}[i, j]_k} \right]^w \right\} \quad (2)$$

위 식에서  $\tilde{T}[i, j]_k$  는  $k$  번째 블록의 DCT 계수이고,  $w$  는 0과 1 사이의 상수로 Watson은 0.7을 사용하였다. 그림 3은 주파수 마스크 함수를 나타낸다.

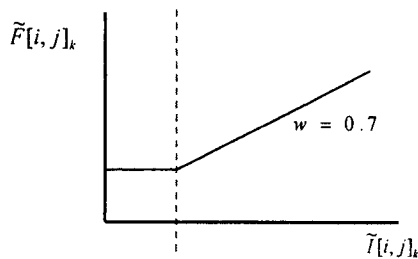


그림 3. 주파수 마스크

그림 3에서처럼  $\tilde{T}[i, j]_k$  가  $t_{th}[i, j]$  를 넘지 못하는 경우에는 임계값에 어떠한 영향을 미치지 않는다.  $\tilde{T}[i, j]_k$  가  $t_{th}[i, j]$  를 넘는 경우는 모든 마스크 주파수에서의 임계값이 거의 직선을 따라 증가함을 알 수 있다.

그림 4는 Forman 시퀀스의 한 프레임에 주파수 마스크를 수행했을 때 나온 결과를 보여준다. 그림 4에서 알 수 있듯이, 블록 내에서 높은 주파수 쪽으로 갈수록 더 높은 마스크 값을 가진다.



그림 4. (a) 원영상. (b) 주파수 마스크

그림 4. (a) 원영상. (b) 주파수 마스크

본 논문에서는 주파수 마스크 뿐만 아니라, 공간 마스크 및 움직임 마스크 효과를 고려한 전역 마스크 맵(map)을 구하기 위해, 식 (2)에서 얻은 값을 역DCT 변환한 후 공간 영역에서의 주파수 마스크 위치정보를 구한다.

$$F = IDCT(\tilde{F}) \quad (3)$$

## 2.2 공간 마스크

공간 영역에서 워터마크 삽입 방법을 고려할 때, 윤곽선은 중요한 역할을 한다. 만약 워터마크가 영상 전체에 같은 비중으로 적용되면, 텍스처(texture) 상의 평평한 부분 즉, 거의 일정한 강도를 가지는 영역에 삽입된 워터마크는 윤곽선 혹은 그것에 가까운 영역에 삽입된 워터마크보다 훨씬 더 눈에 잘 띄게 되는 경향이 있다. 즉, 영상에서 윤곽선은 다른 어떤 부분보다 더 높은 크기의 신호를 가릴 수 있어서 마스크 효과가 높다. 따라서 본 논문에서는 HVS가 윤곽선에 가까운 영역의 잡음에 덜 민감함을 이용하여 공간 마스크를 모델링한다.

윤곽선을 추출하는 방법으로는 여러 가지가 있다 [6]. 본 논문에서는 그 중 가장 일반적인 소벨 윤곽선 검출(Sobel Edge Detection) 방식을 사용하였다. 그림 5(a)는 소벨 검출 방식을 사용한 마스크 맵을 보여준다. 그림 5에서 보듯이, 단순히 윤곽선 마스크만 사용해서 마스크 맵을 만들 경우에 윤곽선이 아닌 부분까지 추출됐음을 알 수 있다. 즉, 텍스처 상에 있는 불규칙한 윤곽선 부분까지도 고려했음을 알 수 있다. 또한 주위 배경의 밝기를 고려하지 않았다. 따라서 본 논문에서는 이러한 문제를 해결하기 위해 윤곽선을 추출하기 전처리 단계로 영상의 명암(contrast)를 조절한다. 즉, 윤곽선 추출시 더 좋은 성능을 가진 마스크 맵을 얻을 수 있도록 영상의 명암을 측정된 후 그 결과로 나온 영상에 윤곽선 추출 방법을 사용한다.

영상의 명암을 고려한 웨버의 법칙(Weber's Law)은 밝기에 따른 임계값을 정의한 것으로, 어떤 한 신호의 인식은 그 주위 배경의 밝기 차와 관련됨을 말한다 [7]. 그래서 잡음에 대한 가시성 임계값은 어두운 영역에서 보다 밝은 영역에서 더 높음을 알 수 있다. 이때 영상의 명암은 상대적인 밝기 차에 의해 측정된다. 그러나 이 법칙은 실제 영상에 적용하기에는 너무 단순하며, 또한 높은 밝기 값에서의 민감도를 제대로 평가하지 못해서 영상의 밝은 영역에 삽입된 워터마크를 눈에 띄게 하는 문제를 야기한다 [8]. 이러한 단점을 보완하기 위해 Kutter가 제안한 변형된 밝기 함수를 사용한다.

식 (4)에서  $I[x, y]$  는 원래 영상의 밝기 값이며,  $a$  는 0.05로 정한다.

$$S[x, y] = 1 + 99 \frac{\log(1 + I[x, y] \cdot a) - \log(1 + a)}{\log(1 + 100a) - \log(1 + a)} \quad (4)$$

식 (4)를 적용한 후 윤곽선 검출을 수행함으로써 공간 마스크를 얻을 수 있다. 그림 5(b)는 제안한 공간 마스크 결과를 나타낸다. 밝기 함수를 적용하지 않고 윤곽선 검출을 수행한 그림 5(a)와 비교해서 그림 5(b)의 결과가 훨씬 더 개선됨을 알 수 있다.

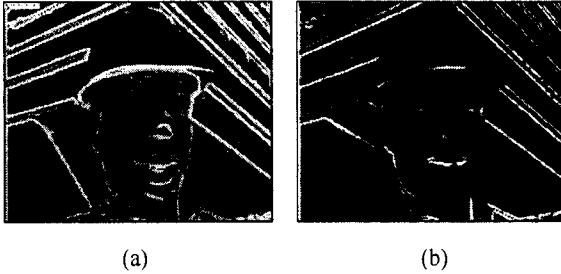


그림 5. 공간 마스크 (a) 윤곽선 검출만 사용, (b) 밝기 함수 사용 후 윤곽선 검출 사용

### 2.3 움직임 마스크(Motion Masking)

정지 영상 워터마킹과는 달리 비디오 워터마킹은 비디오 프레임의 구조적 특성을 이용할 수 있다. 즉, 정지 영상들로 이루어진 비디오에서, 이웃하는 프레임 간의 움직임 변화 부분은 워터마크를 삽입하기에 좋은 특성을 제공한다. 먼저 이웃하는 프레임 사이에서 움직임 변화를 구한 후, 적당한 검출 방법을 써서 윤곽선을 추출한다.

그림 6은 움직임 마스크 결과를 보여준다. 그림에서도 알 수 있듯이, 움직임이 많은 부분인 얼굴 부분에 더 높은 마스크 값이 할당됨을 알 수 있다.



그림 6. 움직임 마스크

### 2.4 전역 마스크 모델링

최종적으로 위에서 구한 주파수 마스크, 공간 마스크, 그리고 움직임 마스크를 조합함으로써 전역 마스크를 구할 수 있다. 먼저 각각의 마스크 결과를 일정한 범위로 정규화(normalize)한 다음 조합한다. 기존 논문에서는 마스크 결과를 조합시 단순히 그것들의 곱을 계산해서 사용하였으나 [9], 본 논문에서는 주파수, 공간 및 움직임 마스크 효과를 모두 고려한 전역 마스크를 위

해서 아래의 식과 같은 조합 방식을 택한다.

$$G = (F + S + M) \quad (5)$$

식 (5)에서  $G$  는 전역 마스크를 나타낸다.

그림 7은 전역 마스크 결과를 보여준다. 그림 7(a)는 기존 논문에서 사용한 방식이며, 그림 7(b)는 제안한 방식의 전역 마스크 결과를 보여준다. 기존의 방식보다 제안한 방식의 전역 마스크가 더 효과적으로 워터마크를 삽입할 수 있음을 확인할 수 있다.



(a) 기존 방법 (b) 제안한 방법

그림 7. 전역 마스크

## 3. 제안한 워터마킹 알고리즘

### 3.1 워터마크 삽입

본 논문에서 워터마크는 이전 레벨의 로고 영상을 사용한다. 즉 소유자가 임의로 정한 키 값에 의해 발생된 랜덤 시퀀스와 로고를 배타적 논리합을 수행하여 나온 결과를 워터마크  $W$  로 한다. 즉,

$$W = X_1 \oplus X_2 \quad (6)$$

이때,  $X_1$  은 랜덤 시퀀스이며,  $X_2$  는 로고 영상이다.

본 논문의 워터마크 삽입 과정은 다음과 같다.

- 1) 프레임을 블록  $I_k$  으로 나눈다. 이때,  $I_k$  는  $k$  번 째 블록을 말한다.
- 2) 각각의 블록에 대해 주파수, 공간 및 움직임 마스크를 구한다. 이때 움직임 마스크는 두 번째 프레임부터 이전 프레임의 같은 위치에 있는 블록과 비교해서 구한다.
- 3) 위에서 구한 공간, 주파수 및 움직임 마스크를 각각 정규화 시킨 후 전역 마스크를 구한다. 즉,

$$G[x, y] = (F[x, y] + S[x, y] + M[x, y])$$

이때,  $x, y = 0 \dots 7$  이다.

- 4) 전역 마스크 맵  $G[x, y]$  에 의해 가중된 워터마크를 원래 블록에 더한다. 즉,

$$I_k[x, y] = I_k[x, y] + \alpha \cdot G[x, y] \cdot W[x, y]$$

이때, 제어 변수  $\alpha$  는 1로 한다.

- 5) 워터마크가 삽입된 블록과 원래 블록의 PSNR을 구해  $\alpha$  를 적절히 조절한다. 즉, 구한 PSNR이 목표값보다 더 작으면  $\alpha$  를 감소시키고, 더 크면  $\alpha$  를 증가시킨다.

위 모든 과정은 블록 단위로 수행한다.

그림 8은 본 논문에서 제안한 워터마킹 방법의 블록도를 나타낸 것이다.

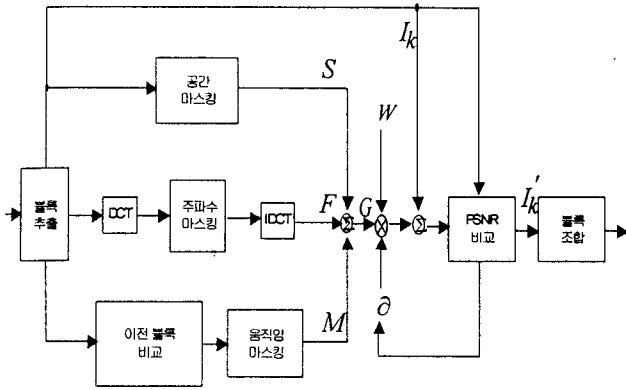


그림 8. 제안한 워터마킹 방법의 블록도

### 3.2 워터마크 추출

본 논문에서 워터마크 추출 과정은 다음과 같다.

- 1) 원래 프레임  $I$ 에서 워터마크가 삽입된 프레임  $I'$ 를 뺀다. 이때 나온 결과는 전역 마스크 값에 의해 가중된 워터마크이다. 즉,

$$I - I' = G \cdot W^*$$

- 2) 위에서 나온 값과 키 값에 의한 랜덤 변수  $X_1$ 의 배타적 논리합을 수행한다. 즉,

$$G \cdot W^* \oplus X_1 = X_2^*$$

이때  $X_2^*$ 는 추출된 로고 영상이다.

## 4. 실험 결과

제안한 방법에 대한 실험을 위해 비디오 데이터로 30 프레임의 Forman(176×144, QCIF)과 Akiyo(176×144, QCIF)를 사용하였다. 워터마크를 삽입하기 위해 RGB(8bits/pixel)를 YUV로 변환시켰다. 그리고 보통의 압축 방식은 색차 성분보다 휘도 성분의 질을 떨어뜨리기 때문에 휘도 성분인 Y성분에 워터마크를 삽입하였다.

### 4.1 가시성 테스트

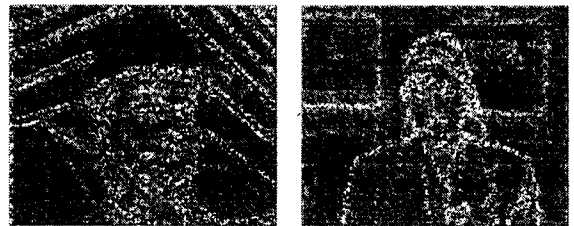
그림 9는 테스트 프레임에 대해 제안한 워터마크 방법의 실험 결과를 보인다. 그림 9(a)는 원래 프레임, 그림 9(b)는 워터마크가 삽입된 프레임을 나타낸다. 워터마크가 삽입된 프레임은 원래 프레임과 비교해 시각적으로 차이가 없음을 알 수 있다. 그림 9(c)는 삽입된 워터마크로 화면 출력을 위해 스케일(scale)시켰다. 전역 마스크 맵에 의해 워터마크가 높은 주파수를 가진 지역, 즉 윤곽선 부분과 움직임이 빠른 지역 부분에 집중되어 있음을 알 수 있다. 그림 9(d)는 3.2절에서 기술된 워터마크 추출 방법에 의해 추출된 로고 영상이다.



(a) 원래 프레임



(b) 워터마크가 삽입된 프레임



(c) 워터마크



(d) 추출된 로고

그림 9. 제안된 워터마크 방법의 실험 결과

### 4.2 MPEG 부호화에 대한 강인성 테스트

부호화에 대한 제안 워터마크 방법의 강인성을 테스트하기 위해 워터마크가 삽입된 프레임에 비트율(Bit Rate)을 달리 하면서 MPEG-2 부호화를 실시하고, 각 경우에 대해 로고를 추출할 수 있음을 확인할 수 있다.

표 1과 표 2는 테스트 프레임 Foreman과 Akiyo를 MPEG-2 부호화 시킨 후 원래 프레임과 워터마크가 삽입된 프레임의 PSNR을 비교한 결과를 나타낸다. 이때 MPEG-2 부호화는 비트율을 5 Mbps, 3 Mbps, 2.5 Mbps 등으로 달리 하면서 수행하였다. 원래 비디오 프레임과 비교해서 화질의 열화가 거의 없음을 알 수 있다.

그림 10과 그림 11은 MPEG-2로 부호화 된 테스트 프레임으로부터 추출된 로고 영상을 나타낸다. 압축 후에도 각각의 비트율에 대해 로고 영상을 추출할 수 있음을 알 수 있다.

표 1. Foreman 프레임의 PSNR 비교

		Bit Rate [Mbit/s]		
		5	3	2.5
PSNR [dB]	Original Frame	25.6	24.9	24.7
	Watermarked Frame	23.5	22.5	22.1



(a) 5 Mbps (b) 3 Mbps (c) 2.5 Mbps

그림 10. MPEG-2 부호화 된 Foreman 프레임으로부터 추출된 로고 영상

표 2. Akiyo 프레임의 PSNR 비교

		Bit Rate [Mbit/s]		
		5	3	2
PSNR [dB]	Original Frame	32.9	31.7	30.7
	Watermarked Frame	31.1	29.9	29



(a) 5 Mbps (b) 3 Mbps (c) 2 Mbps

그림 11. MPEG-2 부호화 된 Akiyo 프레임으로부터 추출된 로고 영상

#### 4.3 재부호화에 대한 강인성 테스트

MPEG-2 재부호화에 대해 제안한 워터마크 방법의 강인성을 테스트하였다. 4.2절에서 사용된 MPEG-2 부호화된 워터마크 삽입 프레임을 다시 MPEG-2 부호화 및 복호화 과정을 거친 후 3.2절에서 기술된 워터마크 추출 방법으로 로고 영상을 추출하였다.



(a) 5 Mbps (b) 3 Mbps (c) 2 Mbps

그림 10. 재부호화 후 Akiyo 프레임으로부터 추출된 로고 영상

## 5. 결론

본 논문에서는 전역 마스크를 이용한 비디오 워터마킹 방법을 제안하였다. 제안한 방법은 눈에 띄지 않으면서도 워터마크를 최대한 효과적으로 비디오 프레임에 삽입하기 위해 인간시각 특성을 이용한 전역 마스크 맵을 이용하였다. 이때 전역 마스크는 주파수, 공간, 그리고 움직임 관점에서 마스크 효과를 고려한 후, 위 세가지 마스크를 결합하여 구하였다. 본 논문에서 제안한 방법으로 비디오 프레임에 삽입된 워터마크 신호는 MPEG-2 부호화 및 재부호화 공격에 대해서도 강인함을 확인했으며, 비가시성 측면에서도 우수한 특성을 지님을 알 수 있었다.

### 감사의 글

본 연구는 광주과학기술원 초고속광네트워크 연구 센터(UFON)를 통한 한국과학재단우수연구센터(ERC)와 교육부 두뇌 한국21(BK21) 정보기술사업단의 지원에 의한 것입니다.

### 참고 문헌

- [1] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proc. IEEE*, vol. 87, no. 7, pp. 1079-1107, July 1999.
- [2] F. Hartung and B. Girod, "Digital watermarking of uncompressed and compressed video," *Signal Processing (Special Issue on Copyright Protection and Access Control for Multimedia Services)*, vol. 66, no. 3, pp. 283-301, 1998.
- [3] M. Swanson, B. Zhu and A. Tewfik, "Multiresolution scene-based video watermarking using perceptual models," *IEEE Select. Areas Commun. (Special Issue on Copyright and Privacy Protection)*, vol. 16, pp. 540-550, May 1998.
- [4] W. Zhu, Z. Xiong and Y.-Q. Zhang, "Multiresolution watermarking for images and video," *IEEE Trans. Circuits Sys. Video Tech.*, vol. 9, no. 4, pp. 545-550, June 1999.
- [5] A. B. Watson, "DCT quantization matrices optimized for individual images," *Human Vision, Visual Processing, and Digital Display IV*, SPIE-1913, pp. 202-216, 1993.
- [6] W. K. Pratt, *Digital Image Processing*, John Wiley & Sons, New York, 1978.
- [7] C. De. Vleeschouwer, J. F. Delaigle and B. Macq, "Invisibility and application functionalities in perceptual watermarking-an overview," *Proc. IEEE*, vol. 90, pp. 64-77, Jan. 2002.
- [8] M. Kutter and S. Winkler, "A vision-based masking model for spread-spectrum image watermarking," *IEEE Trans. Image Processing*, vol. 11, pp. 16-25, Jan. 2002.
- [9] H. H. Y. Tong and A. N. Venetsanopoulos, "A perceptual model for JPEG applications based on block classification, texture masking, and luminance masking," *Int'l Conference on Image Processing (ICIP)*, vol. 3, pp. 428-432, Oct. 1998.