

# 웨이블릿 기반 압축영상의 화질 향상을 위한 방향성 후처리 기법

정회원 김 승 종\*, 정 제 창\*

## Directional Postprocessing Techniques to Improve Image Quality in Wavelet-based Image Compression

Seungjong Kim\*, Jechang Jeong\* *Regular Members*

### 요 약

영상 데이터는 방대하기 때문에 효율적인 전송 및 저장을 위해서는 압축이 필수적이거나, 고압축률에서는 여러 가지 화질 열화가 발생한다. 화질 열화의 종류에는 DCT 기반 블록변환 부호화에서 나타나는 블록킹 현상, 모기잡음이 있고 웨이블릿 변환 부호화에서는 에지 부근에 보이는 링잉 현상이 있다. 본 논문에서는 웨이블릿 변환에서 인간의 시각이 링잉 현상을 느끼는 부분이 에지 부근인 것을 감안하여, 블록단위로 에지의 방향을 검출하고 검출된 에지의 방향에 따라 방향성 후처리 기법을 제안한다.

제안하는 방법은 복원 영상에 대해 블록단위로 현재 블록에 이웃하는 상하좌우 4 블록의 경계화소를 이용하여 상관계수를 구하고 가장 큰 상관계수를 갖는 방향을 에지의 방향으로 검출한다. 다음에는 검출된 에지의 방향에 따라 방향성 후처리를 수행하며, 만약 현재 블록의 방향별 상관계수 값이 모두 같다면 평탄 영역으로 간주하고 후처리를 수행하지 않도록 하여 처리 시간의 단축을 가져오게 하였다.

### ABSTRACT

Since image data has large data amount, proper image compression is necessary to transmit and store the data efficiently. Image compression brings about bit rate reduction but results in some artifacts. This artifacts are blocking artifacts, mosquito noise, which are observed in DCT based compression image, and ringing artifacts, which is perceived around the edges in wavelet based compression image. In this paper, we propose directional postprocessing technique which improve the decoded image quality using the fact that human vision is sensible to ringing artifacts around the edges of image. First we detect the edge direction in each block. Next we perform directional postprocessing according to detected edge direction. Proposed method is that the edge direction is determined by finding the maximum correlation coefficients of boundary pixels of blocks neighboring the current block. Next performed directional postprocessing according to detected edge direction. If the correlation coefficients are equivalent to each directions, postprocessing is not performed. So, time of the postprocessing brings about shorten.

### I. 서론

사회가 고도로 정보화 됨에 따라 초고속 정보 통

신망, 인터넷을 비롯한 여러 종류의 채널이나 매체를 통해 다양한 멀티미디어 정보 서비스가 제공되고 있다. 멀티미디어 정보는 영상과 음향을 중심으로 문자, 도형, 음성 등이 결합된 다중 정보로서, 이

\* 한양대학교 전자통신공학과(ksj@icsp2.hanyang.ac.kr)  
 논문번호: 00068-0222, 접수일자: 2000년 2월 22일

정보의 단순 PCM(Pulse Code Modulation) 데이터의 양은 방대하여 고속 통신망으로도 감당하기 어렵게 된다. 특히, 멀티미디어 정보 중 영상 데이터는 대용량으로 한정된 대역폭을 통하여 전송하거나 저장하기 위해서는 압축을 필요로 한다. 대용량의 영상정보를 고 능력로 압축하면서 응용 분야에 따른 요구조건을 만족하기 위한 방법으로 JPEG<sup>[6]</sup>, MPEG-1<sup>[7]</sup>, MPEG-2<sup>[8]</sup>, H.261<sup>[9]</sup>과 H.263<sup>[10]</sup> 등 영상부호화에 대한 표준이 이미 제정되었으며, 차세대 영상부호화 방식으로 정지영상은 JPEG-2000<sup>[11]</sup>을 중심으로 동영상은 MPEG-4<sup>[12-3]</sup>를 중심으로 국제 표준의 제정이 완료되고 있으며 각국의 표준 및 기술 선점을 위한 연구가 활발히 진행되고 있다.

웨이블릿 변환은 신호를 다해상도(multiresolution)로 분해하며, 입력 신호는 서로 다른 해상도로 표현하기 위한 부 신호들과 나머지 신호들을 표현하기 위한 세부 신호들로 나누어진다. 웨이블릿 변환된 영상신호는 서로 다른 주파수 특징을 가지며, 이들 간에는 어느 정도의 상관성이 존재한다. 웨이블릿 변환을 이용하여 영상을 다해상도로 분해한 후 영상을 압축하는 것은 기존의 DCT를 이용한 방법보다 압축률을 높일 수 있고 블로킹 현상(blocking artifacts)이 일어나지 않아 인간의 시각에 보다 적절한 압축 방법이라 할 수 있다<sup>[4,5]</sup>.

그러나, 압축으로 인하여 전송 비트율은 낮아지지만 고압축률에서는 여러 가지 화질 열화가 발생한다. 화질 열화의 종류에는 JPEG, H.261, MPEG-1, MPEG-2, H.263의 부호화 영상에서 나타나는 블로킹 현상 또는 모기잡음(mosquito noise)이 있고 JPEG-2000과 같이 웨이블릿 변환을 이용하여 정지 영상을 압축하는 방법에서는 블로킹 현상은 없지만 에지 부근에 보이는 링잉 현상(ringing artifacts)이 있다.

블로킹 현상은 블록 단위의 양자화 과정에서 생기는 오차 때문에 나타나는 것으로 압축률이 높아지면 블록 경계에서의 불연속성이 마치 타일을 깔 것처럼 규칙적으로 나타나는 화질 열화이다. 한편, 모기잡음은 DCT 기반의 부호화에서 양자화로 인한 고주파수 데이터의 손실로 생기며 에지 부근에서 잡음이 보이는 현상이다. 블로킹 현상을 제거하기 위한 후처리 기법들은 다양하게 연구되어 왔다. Kim<sup>[11]</sup> 등은 블로킹 현상을 수평, 수직 방향 블록 경계에서의 고주파 성분으로 간주하여 1차원 또는 2차원 공간 저역 필터를 사용하여 블로킹 현상을 제거한다. 그러나, 이 방법은 구현이 간단하지만 급

격한 밝기 변화를 갖는 블로킹 현상은 잘 제거되지 않으며 블록간에 걸쳐 있는 에지들이 번질(blurring) 수 있는 단점이 있다.

Malvar<sup>[12]</sup> 등은 부호화 및 복호화를 수행할 때 블록간의 약간의 중복을 허용하는 방법으로써, LOT(Lapped Orthogonal Transform)와 같이 완전 복원 특성을 가지면서도 에너지 집중 특성이 좋은 변환 기법이 요구된다. Zakhor<sup>[13]</sup> 등은 양자화에 관한 사전 지식을 제약조건으로 하여 영상의 주파수 가중 에너지(frequency weighted energy)를 최소화하는 방법으로 블로킹 현상이 제거된 영상을 반복적으로 구하였다. Yang<sup>[14]</sup> 등은 양자화에 관한 사전 지식과 전체적인 블록 경계에서의 밝기 값의 차이로부터 두 종류의 블록(convex) 집합들을 정의한 다음 POCS(Projection On to Convex Sets)에 의한 반복적 투영으로 블로킹 현상이 제거된 영상을 얻고자 하였다. 또한 양자화에 관한 사전 지식을 제약조건으로 하여 원 영상의 평활(smoothness) 특성에 관한 함수를 최소화하는 방법으로 블로킹 현상이 제거된 영상을 얻고자 하였다.

O'Rourke<sup>[15]</sup> 및 Luo<sup>[16]</sup> 등은 양자화에 대한 사전 지식과 HMRF(Huber Markov Random Field) 모델을 이용한 MAP 추정을 반복적 접근으로 구함으로써 블로킹 현상이 제거된 영상을 얻고자 하였는데, 이들 두 가지 방법은 결국 제약조건의 최소화에 의한 반복적 처리 방법이다. 이러한 반복적 처리 방법은 공간 필터를 사용하는 방법들 보다 성능이 우수한 반면, 좋은 결과를 얻기 위해서는 영상 전체에 대하여 대체로 10회 이상의 반복이 필요하며 이로 인하여 수행 시간이 길어지는 단점이 있다.

한편, 웨이블릿 변환 영역에서 처리하는 방법은 신호의 국부적 주파수 성분들을 효율적으로 분석 처리할 수 있도록 변환 영역의 다층 스케일 공간상에서 블록 경계 근방의 국부적인 주파수 성분을 추출하고 블로킹 현상에 의한 고주파 성분만을 제거함으로써 블로킹 현상을 제거한다. 이 방법은 시스템의 구현이 비교적 용이하고, 블록 기반 부호화의 방법에 상관없이 처리할 수 있으며, PSNR 및 주관적 화질도 우수한 장점을 갖고 있다.

Kim<sup>[17]</sup> 등은 블로킹 현상이 수평, 수직 방향의 블록 경계를 따라 수평, 수직 방향으로만 나타나는 점에 착안하여, 영상 신호를 수평, 수직 방향의 1차원 신호의 집합으로 간주하고 Mallat<sup>[18]</sup> 등에 의해 제안된 가우시안 형태의 1차 도함수를 모 웨이블릿(mother wavelet)으로 하는 1차원 웨이블릿 변환을

수평, 수직 방향 신호에 대하여 2번째 스케일까지 수행하여 블록킹 현상을 제거하고자 하였다. 즉, 블록 경계의 첫 번째 스케일에서의 웨이블릿 변환 신호에서 임펄스 형태로 나타나는 블록킹 현상에 의한 신호 성분을 제거한 후, 역 웨이블릿 변환을 수행함으로써 블록킹 현상이 제거된 영상을 얻고자 하였다. 그러나, 웨이블릿 변환과 역 웨이블릿 변환을 2번째 스케일까지 수행해야 하므로 웨이블릿 변환에 의한 계산량이 많아져 수행 시간이 길어지는 단점이 있다. 또한 각 블록 경계 영역을 구분할 때 처리하고자 하는 영상의 전체적인 블록킹 현상의 심한 정도에 맞추어서 처리할 수 있도록 하는 역치(threshold)를 사용하였는데, 대상 영상과 부호화한 압축물에 따라 역치를 실험적으로 구해야 하는 단점이 있다.

링잉 현상은 고주파수 대역의 웨이블릿 변환 계수들이 양자화로 인해 손실됨으로써 에지 부근에 허상의 에지가 보이는 것으로 깁스 현상(Gibbs effect)이라고도 하며 강한 에지 부근에서 잘 나타난다. 텍스처(texture) 영역에서는 링잉 현상이 발생하더라도 텍스처 리플(ripple)에 의해 발생한 것인지 아니면 링잉 현상 때문인지 구분이 되지 않는 반면 에지가 있는 평탄(flat)한 영역에서는 링잉 현상이 인간의 시각에 확연히 거슬린다. 링잉 현상을 제거하기 위한 방법은 아직 많은 연구가 이루어져 있지 않으나, Shen<sup>[9]</sup>은 주변의 화소들에서 대표 값을 추출하고 대표 값과 현재 화소 값과의 차가 임계값 이내이면 링잉 현상으로 판단하여 후처리하였다. 그러나, 이 방법은 인간의 시각은 링잉 현상을 느끼는 부분이 에지 부근인 것을 감안하지 않았으며, 또한 에지의 방향성도 고려하지 않고 후처리 하였다. 따라서, 본 논문에서는 웨이블릿 변환에서 인간의 시각이 링잉 현상을 느끼는 부분은 에지 부근인 것을 감안하여, 먼저 블록단위로 에지의 방향을 검출하고 검출된 에지의 방향에 따라 방향성 후처리 방법을 제안하며, 만약 에지가 없는 평탄한 영역에 대해서는 후처리를 수행하지 않도록 하여 처리 시간의 단축을 가져오게 하였다.

제안하는 방법은 Shen이 제안한 방법을 기반으로 복원된 영상에 대해 블록단위로 에지의 방향을 현재 블록에 이웃하는 상하좌우 4 블록의 경계화소를 이용하여 상관계수를  $0^\circ \sim 180^\circ$  까지 구하고 가장 큰 상관계수를 갖는 방향을 에지의 방향으로 검출한다. 다음에는 검출된 에지의 방향에 따라 방향성 후처리를 수행하며, 만약 현재 블록의 상관계수

값이 모두 같다면 평탄 영역으로 간주하여 후처리를 수행하지 않도록 하여 처리 시간의 단축을 가져오게 하였다.

본 논문의 구성은 다음과 같다. II장에서는 본 논문에서 제안한 상관계수를 이용한 에지의 방향을 검출하는 방법을 소개하고 III장에서는 Shen이 제안한 후처리 방법을 소개한다. IV장에서는 본 논문에서 제안한 에지의 방향을 고려한 방향성 후처리 방법에 대해 설명한다. V장에서는 Shen이 제안한 방법과 본 논문에서 제안한 방법의 실험 결과를 비교 분석하고 VI장에서 결론을 맺는다.

## II. 상관계수를 이용한 에지의 방향 검출

블록 단위의 에지 방향 검출은 공간영역에서 현재 블록에 이웃하는 상하좌우 4 블록들의 경계화소를 이용하여 상관계수를  $0^\circ \sim 180^\circ$  까지 구하고 가장 큰 상관계수를 갖는 방향을 에지의 방향으로 검출한다. 제안하는 에지 방향 검출 과정은 다음과 같다. 먼저, 그림 1과 같이 현재 블록에 이웃하는 블록들의 경계화소들을 설정한다. 그림 1에서 *lbbp*, *rbbp*, *tbbp*, *bbbp*는 각각 왼쪽 블록, 오른쪽 블록, 위쪽 블록, 아래쪽 블록의 경계화소 값을 의미한다. 다음에는 식(1)과 같은 각도로  $0^\circ \sim 180^\circ$  까지 상관계수를 구하고 가장 큰 상관계수 값을 가지는 *k*를 결정한다. 한편, 식(1)에서  $180^\circ \sim 360^\circ$  까지의 상관계수는  $0^\circ \sim 180^\circ$ 의 경우와 같기 때문에 따로 구할 필요가 없으며, 에지의 방향이 *k*에 따라 근사화되는 이유는 현재 블록을 둘러싸고 있는 경

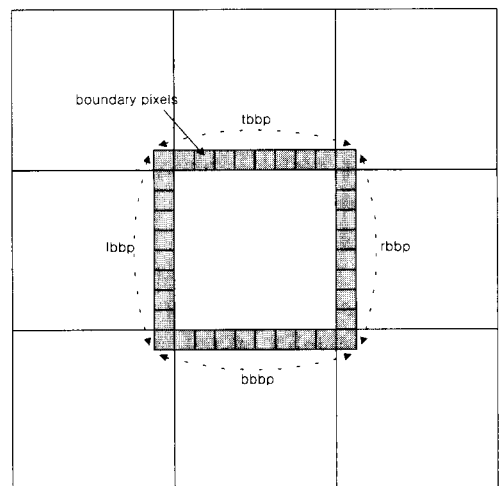


그림 1. 에지의 방향 검출을 위한 경계화소 설정.

계화소들의 배열이 원의 모양이 아니라 정사각형 모양이기 때문이다. 그림 2는  $k$ 에 따른 각도의 변화를 보이고 있으며, 에지의 방향을 결정하는 상관 벡터의 설정 및 후처리 방향은 그림 2와 같은 방향 대로 수행한다.

$$degree \approx k \left( \frac{180}{2N} \right), k=0,1,\dots,2N-1 \quad (1)$$

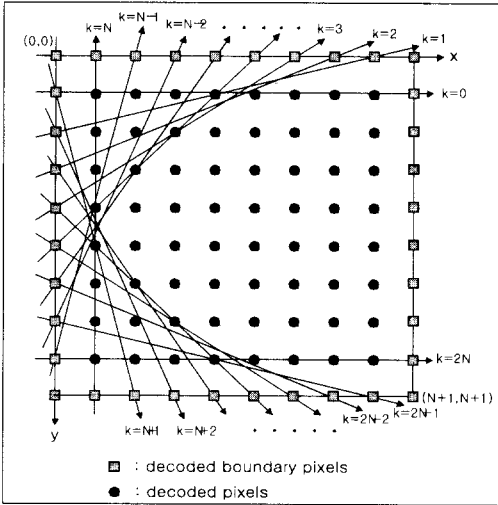


그림 2. 각도에 따른  $k$ 값의 변화.

$$correlation = \max \left\{ \frac{\langle \underline{p}_k^0, \underline{p}_k^1 \rangle}{\| \underline{p}_k^0 \| \| \underline{p}_k^1 \|} \right\} \quad (2)$$

식(2)는 방향  $k$ 에 따른 상관계수를 구하는 것으로,  $\langle \underline{p}_k^0, \underline{p}_k^1 \rangle$ 은 두 벡터의 내적(inner product)을 의미하고  $\| \underline{p}_k^0 \|, \| \underline{p}_k^1 \|$ 는 각 벡터의 norm을 의미한다. 상관계수를 구할 때 이용되는 두 벡터  $\underline{p}_k^0$ 와  $\underline{p}_k^1$ 은  $k$ 에 따라 취해지는 벡터의 크기가 다르다. 즉,  $k$ 가 0과  $N$ 일 경우에는  $N$ 개의 경계화소만 있으면 블록 전체의 영역을 포함하여 정확하게 수평 또는 수직의 에지 방향을 결정할 수 있다. 그러나,  $k$ 가 0과  $N$ 일 경우를 제외하고는  $N$ 개의 경계화소로는 방향 별로 에지의 방향을 정확하게 결정할 수 없기 때문에 상대적으로 화소의 수를 증가시켜야 한다. 그림 3은  $k$ 가 4일 경우에 취해지는 경계화소들을 보이고 있으며 각각의 직선이  $x=0, y=N+1$ 축과 만나는 지점의 경계화소 값들이  $\underline{p}_k^0$ 가 된다. 또한  $y=0, x=N+1$ 축과 만나는 지점의 경계화소 값들이  $\underline{p}_k^1$ 가 된다.

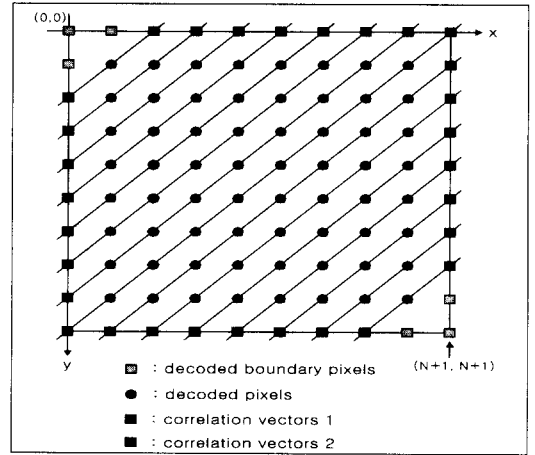


그림 3.  $N=8, k=4$ 일 경우 상관벡터의 설정 및 후처리 방향.

그런데, 그림 4와 같이  $k$ 에 따른 직선이  $x, y$ 축과 만나는 지점에 원래의 경계화소 값이 존재하지 않는 경우에는 이웃하는 경계화소를 이용하여 선형 보간으로 상관계수를 구할 때 이용되는 각각의 상관 벡터를 구한다.  $k$ 에 따른 상관벡터를 구하는 방법은 다음과 같이 5단계를 거쳐 얻어진다.

- 1 단계 :  $k$ 에 따른 직선의 방정식을 구한다.
- 2 단계 : 직선의 방정식에서  $x$ 의 계수  $a_k$ 와  $y$ 의 계수  $b_k$ 를 구한다.
- 3 단계 : 그림 4에서 직선의 절편  $c_k$ 를 구하는 단계로  $k$ 에 따라 결정되는 절편  $c_k$ 는 식(3)과 같이 구해지며, 표 1에  $k$ 에 따른  $a_k, b_k$ 의 값을 나타내었다.

$$c_k = a_k x + b_k y, x, y = 1, \dots, N \quad (3)$$

표 1.  $k$ 에 따른 상수  $a_k, b_k$ 의 값.

N=8일 경우					
$k$	$a_k$	$b_k$	$k$	$a_k$	$b_k$
0	0	1	5, 13	3	2
1, 9	1	4	6, 14	7	3
2, 10	3	7	7, 15	4	1
3, 11	2	3	8	1	0
4, 12	1	1			

표 1에 나타낸  $k$ 에 따른 상수  $a_k, b_k$ 의 값을 살펴보면 식(1)에 의해 구해진 각도가  $90^\circ$  이상에서

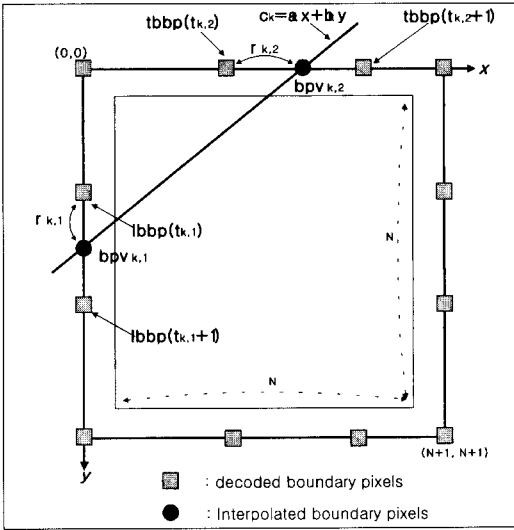


그림 4. 에지의 방향 검출을 위한 경계화소의 선형 보간 방법(단계 1~5).

부터 180° 이하까지의 값이 0° 부터 90° 까지의 값과 같은 것을 볼 수 있다. 왜냐하면, 90° 부터 180° 까지의 과정은 0° 부터 90° 까지의 과정과 동일하게 수행되되 각 블록의 경계화소 값을 그림 1에서 시계방향으로 90° 회전시키고 식(4)와 같이 경계화소들의 순서를 재배열하면 된다. 즉, 원래의 아래쪽 블록의 경계화소 값이 왼쪽 블록으로 재배열되었고 왼쪽 블록은 위쪽 블록으로 순서를 거꾸로 하여 재배열되었음을 알 수 있다. 나머지 블록의 경계화소 값도 마찬가지이다. 결국 90° 부터 180° 까지의 k에 따른 상수 a<sub>k</sub>, b<sub>k</sub>의 값은 따로 각각의 파라미터를 구할 필요 없이 수행할 수 있다. 식(4)에서 i=0,1,...,N+1 인 관계가 있다.

$$\begin{cases} r_{lbbp}(i) = b_{bbp}(i) \\ r_{tbbp}(i) = l_{bbp}(N+1-i) \\ r_{rrbbp}(i) = t_{bbp}(i) \\ r_{rbbbp}(i) = r_{bbp}(N+1-i) \end{cases} \quad (4)$$

• 4 단계: 보간에 필요한 경계화소들의 좌표 t<sub>k,1</sub>, t<sub>k,2</sub>를 구하는 단계로 k와 절편 c<sub>k</sub>의 범위에 따라 다른 값을 갖는다[부록 A.1 참조]. t<sub>k,1</sub>, t<sub>k,2</sub>는 그림 4에서와 같이 직선이 각각 x, y축과 만나는 지점 즉, 현재 블록을 사이에 두고 서로 맞은편에 위치한 경계화소의 좌표를 의미한다.

• 5 단계: 그림 4에서 직선이 x축, y축과 만나는 지점은 경계화소가 없다. 따라서, 임의의 직선이 x축, y축과 만나는 지점의 경계화소 값이 필요하며, 이는 이웃하는 경계화소 값을 이용하여 선형 보간 하면 얻을 수 있다. 선형 보간에 의해 얻어진 값이 b<sub>pv,k,1</sub>, b<sub>pv,k,2</sub>이며, k와 절편 c<sub>k</sub>의 범위에 따라 구해진다[부록 A.2 참조]. 한편, 그림 4에서 r<sub>k,1</sub>, r<sub>k,2</sub>는 b<sub>pv,k,1</sub>, b<sub>pv,k,2</sub>를 구하기 위해 선형 보간에 필요한 거리를 제공한다[부록 A.1 참조].

위와 같이 5단계를 거쳐 k에 따른 상관벡터들을 취하면 현재 블록에 걸쳐 존재하는 에지의 방향을 정확하게 결정할 수 있다. 표 2는 N이 8일 경우 k에 따라 취해지는 경계화소의 수를 보이고 있으며, k에 따른 경계화소의 수는 각 방향별로 에지의 방향을 정확하게 결정하기 위해 필요한 최소한의 화소 수를 의미한다. k에 따른 상관벡터들이 결정되었으면 식(2)를 이용하여 상관계수를 구한다. 식(2)에 의해 얻어진 상관계수 값에 대하여 가장 큰 값을 갖는 상관계수를 취하면 그때의 k가 에지의 방향을 결정하게 되며, 결정된 에지의 방향대로 후처리를 수행하여 복원 영상의 화질을 향상시킨다.

표 2. N=8일 경우, k에 따른 경계화소의 수.

k	화소 수	k	화소 수	k	화소 수	k	화소 수
0,8	8	2,10	11	4,12	15	6,14	11
1,9	10	3,11	13	5,13	13	7,15	10

### III. 링잉 현상 제거를 위한 후처리 기법

Shen이 제안한 후처리 방법은 주변의 화소들에서 대표값을 추출하고 그 대표값과 현재의 화소값과의 차가 임계값 이내이면 링잉 현상으로 판단하여 후처리하는 방법이다. 후처리 과정은 화소 단위로 수행하며 블록도는 그림 5와 같다.

$$e = \arg \min \sum_{x_i \in N} \rho(x_i, \bar{x}_j, x_j) \quad (5)$$

먼저 화소 단위로 주변의 값들과 비교하여 식(5)와 같이 대표값 e를 선택한다. 식(5)에서 x는 화소 값이고, arg는 최소가 되는 화소의 값을 의미한다.

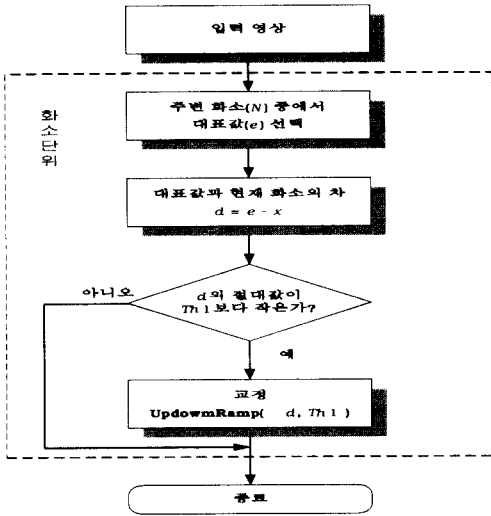


그림 5. Shen이 제안한 후처리 방법의 블록도

$N$ 은 화소들의 집합을 의미하며 그림 6과 같이 현재 화소와 주위의 화소 사이의 기하학적인 관계로 정의된다. 그림 6에서  $N$ 이 5 화소의 ‘+’ 모양을 갖는 제일 왼쪽의 경우가 대표값  $e$ 를 구하는 시간이 가장 짧고 오른쪽으로 갈수록 대표값  $e$ 를 구하는 시간이 길다. 또한,  $\rho(a)$ 는 전위 함수(potential function)로써 식(6)과 같이 네 가지 중에서 하나가 선택된다. 식(6)에서  $a$ 는 식(5)에서  $x_i - x_j$ 이고,  $\gamma$ 는 상수이다.

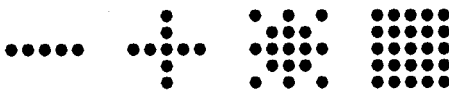


그림 6. 주변화소(N) 선택 예.

$$\left\{ \begin{array}{l} \text{Huber} : \rho(a) = \begin{cases} a^2, & |a| \leq \gamma \\ \gamma^2 + 2\gamma(|a| - \gamma), & |a| > \gamma \end{cases} \\ L^\gamma : \rho(a) = |a|^\gamma, \quad 1 \leq \gamma \leq 2 \\ \text{Truncated } L^2 : \rho(a) = \min\{\gamma a^2, 1\} \\ \text{Lorenzian} : \rho(a) = \log\left(1 + \frac{1}{2}\left(\frac{a}{\gamma}\right)^2\right) \end{array} \right. \quad (6)$$

Shen은 실험에서  $N$ 을 9화소의 ‘+’모양을 갖는 것으로 선택하고 전위 함수  $\rho(a)$ 는 Truncated  $L^2$ 을 주로 사용했다. 이것은 처리 시간의 효율성을 고려하여 계산 시간이 상대적으로 적은 것이 선택된 것으로 볼 수 있다. 이렇게 구한 대표값  $e$ 는  $N$ 의 구성 화소들 중 하나의 화소값이 선택된 것이며

0~255 범위 내의 값이다. 다음에는 선택된 대표값  $e$ 와 현재의 화소값  $x$ 와의 차  $d$ 를 식(7)과 같이 구한다. 현재의 화소값  $x$ 에 후처리 과정으로 교정된 값  $UpDownRamp(d, Th1)$ 를 합한 값  $y$ 는 식(8)에 의해 얻어진다.

$$d = e - x \quad (7)$$

$$y = x + UpDownRamp(d, Th1) \quad (8)$$

식(8)에서  $UpDownRamp(d, Th1)$ 의 의미는  $d$ 의 절대값이 임계값  $Th1$ 의 2배보다 작은 경우에는 후처리를 수행하고 임계값  $Th1$ 의 2배보다 크면 후처리를 수행하지 않는다는 것을 의미한다. 즉, 주어진 화소  $x$ 는 비선형 필터링인 후처리 과정을 거친 교정된 값  $y$ 를 얻는 것을 의미한다.

$UpDownRamp(d, Th1)$ 는 식(9)와 같다.

$$UpDownRamp(d, Th1) = sign(d, Th1) * Max(0, abs(d) - Max(0, 2 * (abs(d) - Th1))) \quad (9)$$

그림 7은 식(9)를 도식적으로 표현한 것으로,  $d$ 의 절대값이  $Th1$ 보다 작은 경우에는 교정치  $UpDownRamp(d, Th1)$ 와의 기울기가 “1”이고  $Th1$ 과  $2Th1$ 의 사이인 경우에는 기울기가 “-1”인 것을 볼 수 있다. 결국 고립점(isolate point), 단일 화소 선(single pixel line)과 같은 특징들을 비선형 필터링을 통해 전체적으로 평탄화 하거나 제거하도록 한다.

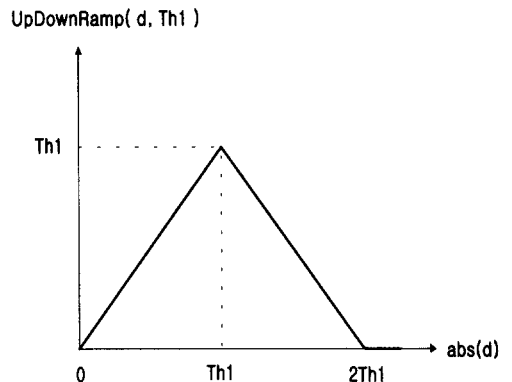


그림 7.  $UpDownRamp(d, Th1)$ 와  $abs(d)$ 의 관계.

#### IV. 에지의 방향을 고려한 방향성 후처리 기법

Shen이 제안한 후처리 방법은 영상 내 존재하는

에지의 방향성을 고려하지 않고 후처리를 수행하기 때문에, 후처리 수행에 필요한 대표 값  $\epsilon$ 의 선정에 문제가 있으며, 또한 텍스처 리플(ripple)에 의해 발생한 것이 링잉 현상으로 잘못 인식될 수 있는 단점을 가지고 있다. 따라서, 본 논문에서는 에지의 방향에 따라 후처리를 수행하여 Shen 방법의 문제점을 해결하는 후처리 방법을 제안한다. 또한 Shen이 제안한 방법은 화소 단위로 영상 전체를 후처리한다. 따라서 JPEG-2000의 실험 영상 중 2048 × 2048, 2048 × 2560 크기를 갖는 Aerial2, Bike 등과 같은 경우에는 후처리에 많은 시간이 소요된다. 실제로 Shen의 방법으로 후처리한 후, 화소값이 바뀐 화소들을 조사해 보면 주로 고주파 성분이 많은 에지 부분과 텍스처 부분이라는 것이 확인된다. 따라서, 본 논문에서는 후처리 전에 후처리 과정을 거처도 변화되지 않는 평탄(flat)한 영역을 미리 제거함으로써 처리 시간의 단축을 가져오는 방법을 제안한다. 본 논문에서 제안하는 후처리 방법에 대한 전체적인 알고리즘은 그림 8과 같다.

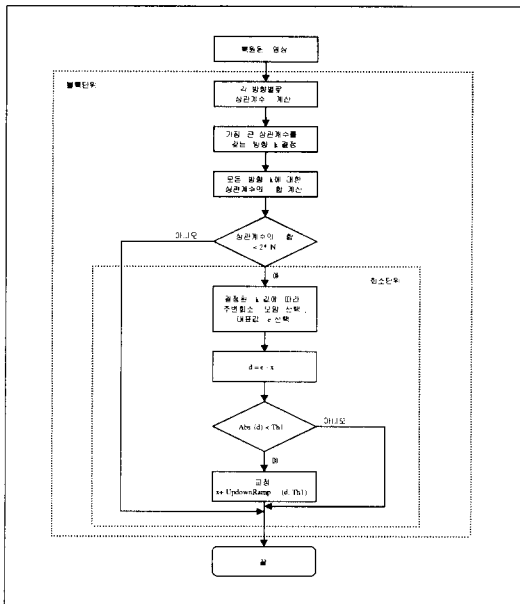


그림 8. 제안하는 후처리 알고리즘의 전체적인 블록도

먼저, II장에서 제안한 방법에 의해 후처리 하고자 하는 현재 블록에서 가장 큰 상관계수 값을 갖는 에지의 방향  $k$ 가 결정되었으면, 결정된 방향  $k$ 에 따라 후처리를 수행한다. 그러나, 실질적으로 링잉 현상이 나타나는 곳 중에서 인간 시각이 화질

열화로 느끼는 부분은 에지 부근인 것을 고려해 볼 때, 영상 중 에지 주위만 후처리하는 것은 타당하다. 한편, 텍스처 영역에서는 링잉 현상이 발생하더라도 텍스처 리플에 의해 발생한 것인지 아니면 링잉 현상 때문인지 구분이 되지 않는다. 따라서, 본 논문에서는 후처리를 수행하기 전에 평탄 영역을 미리 제거하여 후처리 시간의 단축을 가져오게 한다. 즉, 현재 후처리 하고자 하는 블록이 에지가 존재하지 않는 평탄 영역이라면 거의 모든  $k$ 의 방향에 대한 상관계수 값이 “1”로 계산된다. 그러나, 에지가 존재하게 되면 에지가 존재하는 방향의 상관계수 값만 제외하고 나머지 방향에 대해서는 상관계수 값이 “1” 보다 작은 값으로 계산된다. 따라서, 각 방향에 대한 상관계수의 합을 구하여 상관계수의 합이  $2N$ (  $N$  : 블록 사이즈) 보다 작으면 에지가 존재하는 것으로 가정하고 후처리를 수행하고 그렇지 않으면 평탄 영역으로 간주하여 후처리를 수행하지 않는다.

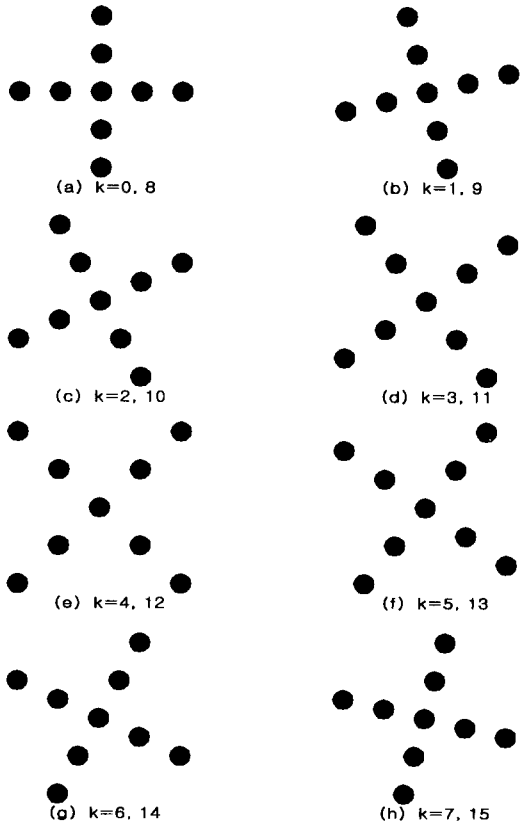


그림 9. 에지 방향  $k$ 에 따른 주변화소의 기하학적인 표현

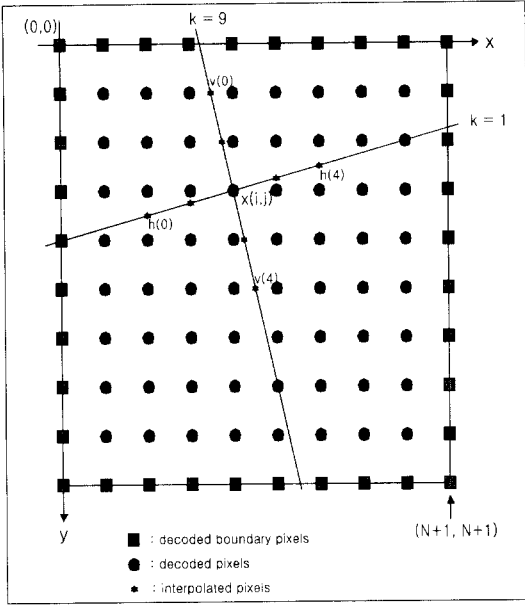


그림 10.  $k=1,9$ 일 경우, 제안한 방향성 후처리의 예.

다음에는 현재 블록이 후처리를 수행하는 블록으로 결정되었으면, 이미 결정된 에지의 방향  $k$ 에 따라 주변화소 모양을 결정하고 결정된 주변화소 모양에 따라 방향성 후처리를 수행한다. 그림 10은 현재 블록의 에지 방향이  $k=1$ 일 경우, 본 논문에서 제안한 방향성 후처리 방법의 예를 보이고 있다. 후처리 하고자 하는 화소가  $x(i, j)$ 라고 가정하면,  $x(i, j)$  화소를 지나는 직선들에서 주변화소를 선택해야 한다.  $x(i, j)$  화소를 지나는 직선은  $k=1$ 일 경우와 그것의 수직인 직선  $k=9$ 가 존재한다. 따라서, 주변화소를 선택하는 기하학적인 모양은 그림 9(b)가 선택된다. 그러나, 그림 10에서 주변화소의 선택 시, 직선이 지나가는 위치에 복원된 영상 자체의 화소는 존재하지 않으므로 이웃하는 두 화소를 이용하여 선형 보간으로 보간된 화소를 얻어야 한다. 선형 보간은 두 화소 사이를 지나가는 직선의 위치와 비교하여 보다 가까운 위치에 있는 화소에 많은 영향을 받게끔 보간한다. 보간하는 방법은 다음과 같다.

먼저,  $k$ 에 따른 직선의 방정식을 식(3) 및 표 1을 이용하여 구하면  $x$ 의 계수  $a_k$ 와  $y$ 의 계수  $b_k$ 를 알 수 있으며, 주변화소의 수평방향 5개 화소들을  $\{h_k\}$ 라하고 수직방향 5개 화소들을  $\{v_k\}$ 라 하자. 그러면,  $k$ 에 따른 주변화소들은 식(10) 및 식(11)과 같이 얻을 수 있다.

$$\begin{cases} h_k(0) = \frac{(b_k - 2a_k)x(i, j - 2) + 2a_kx(i + 1, j - 2)}{b_k} \\ h_k(1) = \frac{(b_k - a_k)x(i, j - 1) + a_kx(i + 1, j - 1)}{b_k} \\ h_k(2) = x(i, j) \\ h_k(3) = \frac{(b_k - a_k)x(i, j + 1) + a_kx(i - 1, j + 1)}{b_k} \\ h_k(4) = \frac{(b_k - 2a_k)x(i, j + 2) + 2a_kx(i - 1, j + 2)}{b_k} \end{cases} \quad (10)$$

$$\begin{cases} v_k(0) = \frac{(b_k - 2a_k)x(i - 2, j) + 2a_kx(i - 2, j - 1)}{b_k} \\ v_k(1) = \frac{(b_k - a_k)x(i - 1, j) + a_kx(i - 1, j - 1)}{b_k} \\ v_k(2) = x(i, j) \\ v_k(3) = \frac{(b_k - a_k)x(i + 1, j) + a_kx(i + 1, j + 1)}{b_k} \\ v_k(4) = \frac{(b_k - 2a_k)x(i + 2, j) + 2a_kx(i + 2, j + 1)}{b_k} \end{cases} \quad (11)$$

Shen이 제안한 방법과 본 논문에서 제안한 방법의 가장 큰 차이점은 바로 주변화소의 선택 방법이 다르다는 것이다. 즉, Shen은 그림 6과 같이 에지의 방향과는 무관하게 같은 모양의 주변화소를 선택하여 후처리를 수행하지만 본 논문에서는 에지의 방향에 따라 주변화소의 모양이 다르며, 후처리 방향도 에지의 방향에 따라 다르게 수행됨을 알 수 있다. 식(10) 및 식(11)을 이용하여 에지의 방향  $k$ 에 따른 주변화소를 선택하였으면, 식(5)와 같은 방법에 의해 선택된 주변화소 중에서 대표값  $e$ 를 추출한다. 대표값  $e$ 가 선택되었으면, 현재의 화소 값  $x$ 와의 차  $d$ 를 식(7)과 같이 구한다. 다음에는  $d$ 의 절대값이 이미 결정된 임계값  $Th1$ 보다 작은 경우,  $UpDownRamp(d, Th1)$ 를 식(9)를 이용하여 구하고 현재의 화소값  $x$ 에 교정된 값을 합한  $y$ 를 식(8)에 의해 구하면 후처리 과정이 완료된다.

## V. 실험 결과 및 고찰

실험에 사용된 영상은  $512 \times 512$  크기의 Lena, Target 영상,  $2048 \times 2048$  크기의 Aerial2 영상, 그리고  $2048 \times 2560$  크기의 Bike, Cafe, Woman 영상을 원 영상으로 사용하였다. 실험에 이용된 복호화된 영상은 JPEG-2000 Verification Model(VM) 5.02의 디폴트(default) 옵션을 이용하여 각각 비트율 0.25bpp, 0.125bpp, 0.0625bpp로 부호화 한 뒤 다시 복호화 한 것을 사용했다. 웨이블릿 분해는 5



레벨까지 분해하였고 변환 시 이용한 변환 필터는 Daubechies 9/7 탭 필터를 이용하였다. 실험에 사용된 각 파라미터 값을 표 3에 나타내었다. 즉,  $Th1$  값은 0.25bpp, 0.125bpp, 0.0625bpp일 때 각각 8, 10, 12로 하였으며, 전위 함수는 Lorenzian을 이용하였고  $\gamma$  값은 3으로 설정했다. 또한 예지의 방향 결정은 블록 단위로 하였으며, 블록의 크기는  $8 \times 8$ 로 하였다.

표 3. 실험에 사용한 파라미터 값.

파라미터	Target, Lena, Aerial2, Bike, Cafe, Woman		
$\gamma$	3		
$\rho(a)$	$\log\left(1 + \frac{1}{2}\left(\frac{a}{\gamma}\right)^2\right)$		
bit rate[bpp]	0.0625	0.125	0.25
$Th1$	12	10	8

표 4는 본 논문에서 제안한 방법을 이용하여 후처리했을 경우, 각 영상 및 비트율에 따라 후처리하는 블록의 개수를 보이고 있다. Shen이 제안한 후처리 방법은 모든 블록에 대해 후처리를 수행하기 때문에 후처리 시간이 많이 소요되지만, 본 논문에서는 현재 블록에 대해 각 방향에 대한 상관계수의 합을 구하여 상관계수의 합이 임계치 보다 작으면 예지가 존재하는 것으로 가정하고 후처리를 수행하고 그렇지 않으면 평탄 영역으로 간주하여 후처리를 수행하지 않기 때문에 Shen의 방법보다는 후처리 시간이 훨씬 적게 소요됨을 알 수 있다.

표 4. 제안한 방법의 각 영상 및 비트율에 따른 후처리 블록( $8 \times 8$ ) 개수.

	총 블록 수	0.25 bpp	0.125 bpp	0.0625 bpp
Aerial2	65536	57700	53337	49196
Bike	81920	41356	41246	40395
Cafe	81920	65877	64954	63446
Target	4096	2678	2587	2497
Lena	4096	2489	2418	2358
Woman	81920	40100	34547	28227

표 5~10은 각각 0.25bpp, 0.125bpp, 0.0625bpp에서 Shen의 방법과 제안한 방향성 후처리 방법의 PSNR 결과를 보이고 있다. 표에서 1, 2, 3의 의미

는 3번 반복하여 후처리한 결과를 의미한다. 즉, 복호된 영상을 한번 후처리하고 후처리된 영상을 또 다시 후처리한 것을 의미한다. 표에서도 알 수 있듯이 Shen의 방법보다는 본 논문에서 제안한 방향성 후처리 방법이 PSNR 측면에서 우수함을 알 수 있다. 이는 인간의 시각이 링잉 현상을 느끼는 부분이 예지 부근인 것을 감안하여, 예지의 방향성을 검출하고 검출된 예지의 방향에 따라 후처리를 수행하는 방법이 보다 효율적임을 입증하는 것이라고 할 수 있겠다.

표 5. Shen의 후처리 방법(0.25 bpp).

	복호 영상의 PSNR[dB]	후처리 후의 PSNR[dB]		
		1	2	3
Aerial2	28.590	28.285	28.142	28.015
Bike	29.663	29.629	29.596	29.555
Cafe	23.169	23.157	23.139	23.120
Target	27.989	28.017	28.024	28.023
Lena	34.133	33.654	33.390	33.151
Woman	30.032	29.871	29.761	29.666

표 6. 제안한 방향성 후처리 방법(0.25 bpp).

	복호 영상의 PSNR [dB]	후처리 후의 PSNR [dB]		
		1	2	3
Aerial2	28.590	28.410	28.196	28.000
Bike	29.663	29.800	29.768	29.694
Cafe	23.169	23.243	23.254	23.245
Target	27.989	28.137	28.198	28.222
Lena	34.133	33.975	33.645	33.324
Woman	30.032	29.924	29.742	29.575

표 7. Shen의 후처리 방법(0.125 bpp).

	복호 영상의 PSNR [dB]	후처리 후의 PSNR [dB]		
		1	2	3
Aerial2	26.517	26.361	26.300	26.238
Bike	26.391	26.387	26.378	26.362
Cafe	20.794	20.773	20.756	20.738
Target	23.740	23.747	23.744	23.739
Lena	30.970	30.695	30.536	30.393
Woman	27.385	27.269	27.196	27.131

표 6, 8, 10에서 Aerial2 및 Woman 영상은 후처리 한 영상의 PSNR이 복호된 영상의 PSNR 보다 낮은 것을 볼 수 있다. 왜냐하면 Aerial2는 자연 영

상을 촬영한 항공사진으로써 강한 에지보다는 텍스처 부분이 많기 때문에 후처리를 수행하면 PSNR 측면에서 복호된 영상과 비교하여 다소 낮아진다. 또한 Woman 영상도 여자가 입고 있는 옷이 주로 텍스처 영역이기 때문에 후처리하면 PSNR은 떨어지나, 얼굴 부분 및 손, 팔 부분들은 강한 에지가 존재하기 때문에 후처리를 수행하면 주관적인 화질은 훨씬 향상됨을 알 수 있었다.

표 8. 제안한 방향성 후처리 방법(0.125 bpp).

	복호 영상의 PSNR [dB]	후처리 후의 PSNR [dB]		
		1	2	3
Aerial2	26.517	26.456	26.371	26.282
Bike	26.391	26.536	26.576	26.576
Cafe	20.794	20.845	20.857	20.854
Target	23.740	23.801	23.822	23.829
Lena	30.970	30.968	30.856	30.714
Woman	27.385	27.317	27.209	27.107

표 9. Shen의 후처리 방법(0.0625bpp).

	복호 영상의 PSNR [dB]	후처리 후의 PSNR [dB]		
		1	2	3
Aerial2	24.611	24.503	24.464	24.422
Bike	23.816	23.816	23.812	23.804
Cafe	19.074	19.048	19.030	19.014
Target	20.194	20.204	20.203	20.208
Lena	27.921	27.810	27.744	27.678
Woman	25.613	25.503	25.443	25.388

표 10. 제안한 방향성 후처리 방법(0.0625 bpp).

	복호 영상의 PSNR [dB]	후처리 후의 PSNR [dB]		
		1	2	3
Aerial2	24.611	24.604	24.581	24.552
Bike	23.816	23.928	23.975	23.993
Cafe	19.074	19.098	19.102	19.099
Target	20.194	20.237	20.259	20.277
Lena	27.921	27.976	27.973	27.950
Woman	25.613	25.535	25.444	25.360

그림 11은 Bike 영상(0.125bpp)에 대한 Shen의 방법과 본 논문에서 제안한 방법의 실험 결과 영상을 보이고 있다. Shen의 후처리 결과 영상 (c)는 표 7에 보인 바와 같이 PSNR은 복호된 영상 (b)와 비교하여 다소 낮지만, 에지 주위에 존재하는 링잉 현

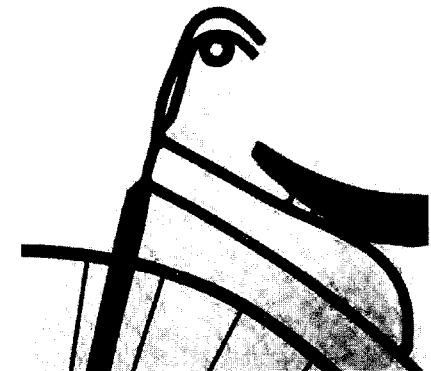
상은 현저히 감소된 것을 볼 수 있다. 한편, 본 논문에서 제안한 방향성 후처리 결과 영상 (d)는 PSNR 측면에서 Shen의 결과 영상 및 복호된 영상보다도 우수함을 알 수 있고 링잉 현상도 현저히 감소되어 주관적인 화질 또한 향상되었음을 알 수 있다.



(a) 원 영상.



(b) 후처리 전의 영상(0.125bpp).



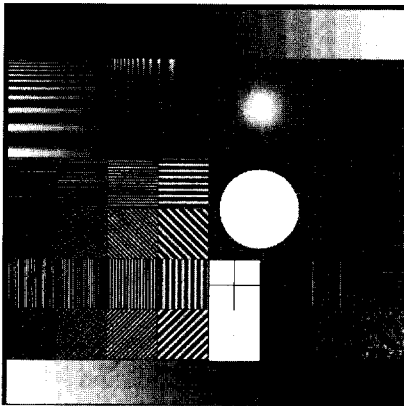
(c) Shen의 후처리 결과 영상.



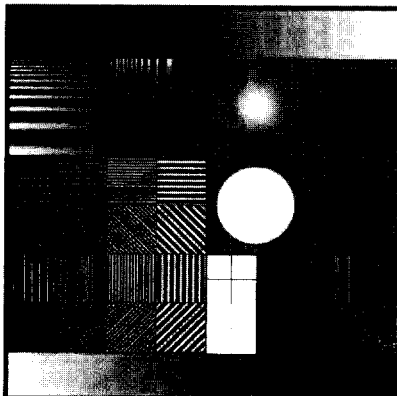
(d) 제안하는 방향성 후처리 결과 영상.

그림 11. Bike의 부분 영상.

그림 12는 Target 영상(0.25bpp)에 대한 Shen의 방법과 본 논문에서 제안한 방법의 실험 결과 영상을 보이고 있다. 그림 11과 마찬가지로 제안한 방법이 Shen의 방법과 비교하여 PSNR 측면에서 약 0.2dB 이상 우수한 화질을 보였으며, 에지의 방향

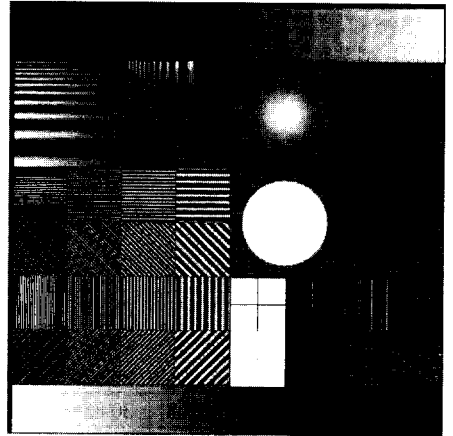


(a) 원 영상.

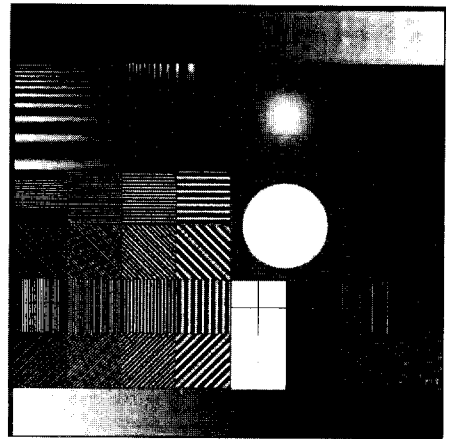


(b) 후처리 전의 영상(0.25bpp).

성을 고려하여 후처리를 수행했기 때문에 흰색과 검은색 원 주위에 존재하는 artifacts 들이 Shen의 방법보다도 현저히 감소되어 주관적인 화질측면에서도 우수함을 보였다.



(c) Shen의 후처리 결과 영상.



(d) 제안하는 방향성 후처리 결과 영상.

그림 12. Target 영상.

## VI. 결론

본 논문에서는 웨이블릿 기반 압축 영상에서 발생하는 링잉 현상을 제거하기 위해 에지의 방향성을 고려한 후처리 방법을 제안하였다. 제안하는 방법은 블록 단위로 현재 블록의 상하좌우 4 블록의 경계회소를 이용하여 상관계수를 구하고 가장 큰 상관계수 값을 갖는 방향을 에지의 방향으로 결정하고 결정된 에지의 방향에 따라 후처리를 수행하였다. 또한 링잉 현상이 발생하는 위치가 에지 부근인 것을 감안하여, 현재 블록의 상관계수 값이 모두



**부 록**

A.1 N=8일 경우, 보간에 필요한 경계화소들의 좌표( $t_{k,1}$ ,  $t_{k,2}$ ) 및 거리( $r_{k,1}$ ,  $r_{k,2}$ )

$k$	range	$t_{k,1}$	$t_{k,2}$	$r_{k,1}$	$r_{k,2}$
0	$8a_k \leq c_k \leq 8b_k$	$\lceil c_k \rceil$	$\lceil c_k \rceil$	0	0
1, 2, 3, 9, 10, 11	$c_k \leq 9a_k$ $9a_k < c_k \leq 9b_k$ $c_k > 9b_k$	$\lceil c_k/b_k \rceil$ $\lceil c_k/b_k \rceil$ $\lceil (c_k - 9b_k)/a_k \rceil$	$\lceil c_k/a_k \rceil$ $\lceil (c_k - 9a_k)/b_k \rceil$ $\lceil (c_k - 9a_k)/b_k \rceil$	$c_k \% b_k$ $c_k \% a_k$ $(c_k - 9b_k) \% a_k$	$c_k \% a_k$ $(c_k - 9a_k) \% b_k$ $(c_k - 9a_k) \% b_k$
4, 12	$c_k \leq 9a_k$ $c_k > 9b_k$	$\lceil c_k/b_k \rceil$ $\lceil (c_k - 9b_k)/a_k \rceil$	$\lceil c_k/a_k \rceil$ $\lceil (c_k - 9a_k)/b_k \rceil$	$c_k \% a_k$ $(c_k - 9b_k) \% a_k$	$c_k \% a_k$ $(c_k - 9a_k) \% b_k$
5, 6, 7, 13, 14, 15	$c_k \leq 9b_k$ $9b_k < c_k \leq 9a_k$ $c_k > 9a_k$	$\lceil c_k/b_k \rceil$ $\lceil (c_k - 9b_k)/a_k \rceil$ $\lceil (c_k - 9b_k)/a_k \rceil$	$\lceil c_k/a_k \rceil$ $\lceil c_k/a_k \rceil$ $\lceil (c_k - 9a_k)/b_k \rceil$	$c_k \% a_k$ $(c_k - 9b_k) \% a_k$ $(c_k - 9b_k) \% a_k$	$c_k \% a_k$ $c_k \% a_k$ $(c_k - 9a_k) \% b_k$
8	$8b_k \leq c_k \leq 8a_k$	$\lceil c_k \rceil$	$\lceil c_k \rceil$	0	0

※  $\lceil x \rceil$  :  $x$ 를 넘지 않는 최대의 정수, % : 모드(mod) 연산

A.2 N=8일 경우, 보간에 필요한 경계화소 값( $bpv_{k,1}$ ,  $bpv_{k,2}$ )

$k$	range	$bpv_{k,1}$	$bpv_{k,2}$
0	$8a_k \leq c_k \leq 8b_k$	$lbbp(t_{k,1})$	$rbbp(t_{k,2})$
1, 2, 3, 9, 10, 11	$c_k \leq 9a_k$ $9a_k < c_k \leq 9b_k$ $c_k > 9b_k$	$\frac{(b_k - r_{k,1})lbbp(t_{k,1}) + r_{k,1}lbbp(t_{k,1} + 1)}{b_k}$ $\frac{(b_k - r_{k,1})lbbp(t_{k,1}) + r_{k,1}lbbp(t_{k,1} + 1)}{b_k}$ $\frac{(a_k - r_{k,1})bbb p(t_{k,1}) + r_{k,1}bbb p(t_{k,1} + 1)}{a_k}$	$\frac{(a_k - r_{k,2})tbbp(t_{k,2}) + r_{k,2}tbbp(t_{k,2} + 1)}{a_k}$ $\frac{(b_k - r_{k,2})rbbp(t_{k,2}) + r_{k,2}rbbp(t_{k,2} + 1)}{b_k}$ $\frac{(b_k - r_{k,2})rbbp(t_{k,2}) + r_{k,2}rbbp(t_{k,2} + 1)}{b_k}$
4, 12	$c_k \leq 9a_k$ $c_k > 9b_k$	$\frac{(b_k - r_{k,1})lbbp(t_{k,1}) + r_{k,1}lbbp(t_{k,1} + 1)}{b_k}$ $\frac{(a_k - r_{k,1})bbb p(t_{k,1}) + r_{k,1}bbb p(t_{k,1} + 1)}{a_k}$	$\frac{(a_k - r_{k,2})tbbp(t_{k,2}) + r_{k,2}tbbp(t_{k,2} + 1)}{a_k}$ $\frac{(b_k - r_{k,2})rbbp(t_{k,2}) + r_{k,2}rbbp(t_{k,2} + 1)}{b_k}$
5, 6, 7, 13, 14, 15	$c_k \leq 9b_k$ $9b_k < c_k \leq 9a_k$ $c_k > 9a_k$	$\frac{(b_k - r_{k,1})lbbp(t_{k,1}) + r_{k,1}lbbp(t_{k,1} + 1)}{b_k}$ $\frac{(a_k - r_{k,1})bbb p(t_{k,1}) + r_{k,1}bbb p(t_{k,1} + 1)}{a_k}$ $\frac{(a_k - r_{k,1})bbb p(t_{k,1}) + r_{k,1}bbb p(t_{k,1} + 1)}{a_k}$	$\frac{(a_k - r_{k,2})tbbp(t_{k,2}) + r_{k,2}tbbp(t_{k,2} + 1)}{a_k}$ $\frac{(a_k - r_{k,2})tbbp(t_{k,2}) + r_{k,2}tbbp(t_{k,2} + 1)}{a_k}$ $\frac{(b_k - r_{k,2})rbbp(t_{k,2}) + r_{k,2}rbbp(t_{k,2} + 1)}{b_k}$
8	$8b_k \leq c_k \leq 8a_k$	$lbbp(t_{k,1})$	$bbb p(t_{k,2})$

※  $N + 1 \leq k \leq 2N - 1$ 의 범위에서는  $lbbp$ ,  $rbbp$ ,  $tbbp$ ,  $bbb p$ 들이 본문의 식(4)와 같이 변경됨.