한국산학기술학회논문지
Vol. 8, No. 2, pp. 257-266, 2007

Enhancement of Wavelet-coded Image by Directional Filtering

Byong-Seok Min[1*], Seung-Jong Kim[2] and Dong-Kyun Lim[3]

방향성 필터링에 의한 웨이블릿 부호화 영상의 화질 개선

민병석[1*], 김승종[2], 임동균[3]

Abstract In many multimedia applications, image compression is required to substantially reduce the amount of image data. This compression, however, sometimes brings artifacts. Typical artifacts are blocking artifacts and mosquito noise in DCT-coded images, and ringing artifacts around edges in wavelet-coded images. We propose a new directional postprocessing algorithm, which includes detection of the edge direction, interpolation scheme, and directional nonlinear filtering, to enhance the quality of decoded images. Simulation results show that the proposed algorithm is as effective as or more effective than other nonlinear filtering techniques.

Key Words : Directional filtering, Edge direction, Correlation, Postprocessing

요 약 멀티미디어 응용분야에서 영상 압축부호화는 영상의 데이터량을 줄이기 위해 필수적이다. 그러나 이러한 압축부호화는 화질의 열화를 발생시킨다. 블록킹 현상과 모기 잡음은 이산여현변환 기반의 압축 영상에서 전형적인 열화이고, 에지 주변의 링잉 현상은 웨이블릿 기반의 압축영상에서 발생한다. 본 논문에서는 방향성을 고려한 후처리 방법을 제안하고자 한다. 제안하는 방법은 복호된 화질의 개선을 위해 에지의 방향 검출과 보간법, 방향성 비선형 필터링을 포함한다. 다양한 영상에 대해 시험한 결과, 제안된 방법이 다른 비선형 필터 방법에 비해 우수한 성능을 나타내었다.

1. Introduction

A number of new applications, such as mobile video communication, mobile multicast/broadcast, and video over the Internet, require very efficient data compression methods, preserving acceptable quality of reconstructed image. This is, however, challenging to existing coding algorithms and standards, since highly visible artifacts appear when the coding bit rate is low. Primary visual artifacts in transform-based image compression are blocking effect, mosquito noise and ringing effect [1,7,8]. The wavelet-based coding scheme has advantages over the block DCT-based coding method in terms of the rate-distortion tradeoff performance and visibility of coding artifacts, and has been adopted as the baseline one to the JPEG-2000 standard [2]. However, the reconstructed images still suffer from various artifacts at high compression ratios. Ringing artifact is caused by coarse quantization on high frequency components. It is most noticeable artifact in wavelet-based coding at low bit rates.

A work in postprocessing for subband/wavelet coding was proposed by O'Rourke and Stevenson in [3]. They designed a postprocessor that could reduce blocking artifacts in block-encoded images and ringing in subband-encoded images by maximizing the posterior probability(MAP) of an unknown image. A Huber Markov random field(HMRF) model is applied to impose spatial smoothness. The quantization constraints in the transform domain are used to ensure the data fidelity. Their approach results in a constrained optimization problem that could be iteratively solved by gradient descent search.

[1]Chungcheong Univ., Dept. of Electronic Communications, Professor
[2]Hanyang Women's College, Dept. of Computer Information, Professor
[3]Hanyangcyber Univ., Dept. of Computer Eng., Professor
*Corresponding author : Byong-Seok Min(min@ok.ac.kr)

Li and Kuo[4] point out that the method proposed in [3] is less effective in removing the wide spread ringing artifacts in wavelet coding. They proposed a multiscale post-processing technique to solve this problem. And, Luo et al.[5] also proposed a strict MRF model with the new parameter set for Huber minimax function. All these methods apply computationally intensive iterative techniques to obtain the solution. Besides, the data fidelity is often defined in the transform domain so that additional forward/backward transforms are required. Shen and Kuo[6] formulated the artifact reduction problem as a robust estimation problem, and found a sub-optimal solution using a nonlinear filtering technique. Their approach achieved desired postprocessing properties, such as noise smoothing, edge recovery, etc. However, the nonlinear-filtered output and the shape of the filter are independent of edge direction. Therefore, their approach still leaves ringing artifacts around sharp edges.

In this paper, we propose a new directional postprocessing algorithm to reduce ringing artifacts. First, we detect the edge direction in each block by finding the maximum correlation coefficient of boundary pixels neighboring the current block, and apply the proposed directional postprocessing technique to the decoded images along the detected edge direction. This paper is organized as follows. The nonlinear filtering technique is briefly reviewed in Section 2. Section 3 explains proposed several topics such as basic criterion for detection of the edge direction, measurement of the maximum correlation coefficient, interpolation scheme to obtain the interpolated boundary pixels, and directional postprocessing technique to enhance the quality of reconstructed images. Experimental results are given in Section 4. Finally, we state conclusions in Section 5.

2. Review on Nonlinear Filtering

In this section, we briefly explain the postprocessing algorithm given in [?]. To reduce the wavelet coding artifacts, Shen and Kuo proposed the nonlinear filter based on the robust M-estimator, which could achieve artifact reduction while preserving important image detail. The Fig. 1 shows their postprocessing algorithm.

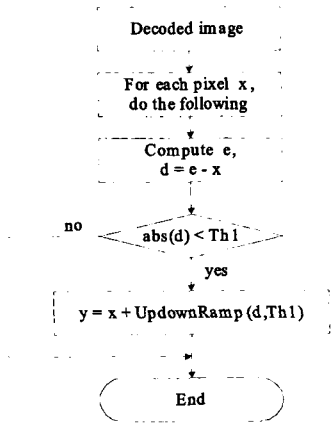


Fig 1. Block diagram of the nonlinear filtering technique.

Given a set of samples $x_1, x_2, ..., x_M$ and the potential function ρ_s, Shen and Kuo obtained the filtered output e by (1). The potential functions in (2), which smooth small artifacts while reducing the influence from large gross edges, could be used for artifact reduction.

$$e = arg\ min\ \{ \sum_{x_i \in M} \rho_s (x_i - x_j) \} \tag{1}$$

$$\begin{cases} Huber & \rho_s(\alpha) = \begin{cases} \alpha^2, & |\alpha| \le \gamma \\ \gamma^2 + 2\gamma(|\alpha| - \gamma), & |\alpha| > \gamma \end{cases} \\ L^{\gamma} & \rho_s(\alpha) = |\alpha|^{\gamma}, \quad 1 \le \gamma \le 2 \\ Truncated\ L^2 & \rho_s(\alpha) = \min\{\gamma\alpha^2, 1\} \\ Lorenz & \rho_s(\alpha) = \log(1 + 1/2(\alpha/\gamma)^2) \end{cases} \tag{2}$$

To reduce the computational complexity, they used the finite set of samples within a local window for finding the global minimum. The filtering window is shown in Fig. 2. For the experiments, they usually used a $'+'$ shaped window with size 9 and the potential function with $Truncated\ L^2$. To ensure the data fidelity, they applied the clipping process as in (3), where y is the final estimated value, and $UpdownRamp\,(d, Th1)$ is the clipping function as defined by (5).

$$y = x + UpdownRamp\,(d, Th1) \tag{3}$$

$$d = e - x \tag{4}$$

$$UpdownRamp(d, Th1) = sign(d) \cdot Max(0, abs(d) - Max(0, 2(abs(d) - Th1))) \quad (5)$$

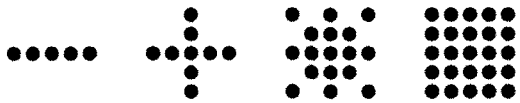


Fig 2. Window shapes for the nonlinear filtering.

3. Directional Postprocessing Method

In this section, we describe the proposed several schemes such as basic criterion for detection of the edge direction, measurement of the maximum correlation coefficient, interpolation algorithm to obtain the interpolated boundary pixels, and directional postprocessing technique to enhance the quality of reconstructed images.

3.1 Basic criterion for detection of the edge direction

The edge direction is determined by finding the maximum correlation coefficient of boundary pixels neighboring the current block in spatial domain. We define boundary pixels lp, rp, tp, and bp represent left-, right-, top-, and bottom-block boundary pixels respectively as shown in Fig. 3 and the correlation coefficients as a function of angle from 0° to 180°.

$$\theta = k\left(\frac{180^\circ}{16^\circ}\right), \quad k = 0, 1, \cdots, 15 \quad (6)$$

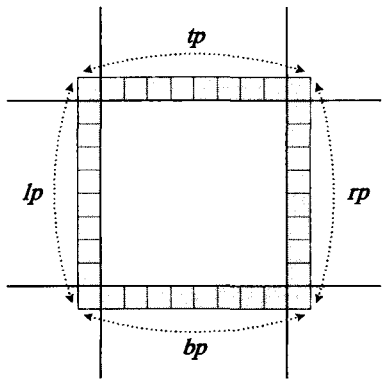


Fig 3. Definition of the boundary pixels

In (6), θ° represents the angle of edge direction and k is the direction index. Fig. 4 shows the edge directions as the direction index k.

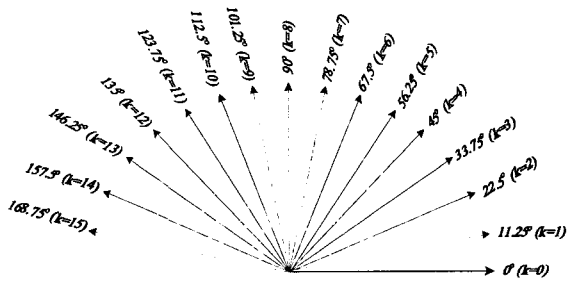


Fig 4. Edge directions as the direction index k varies.

3.2 Measurement of the maximum correlation coefficient

Let p_k^0 and p_k^1 be the two sets of boundary pixels at direction index k. Then the correlation coefficient CC_k is obtained by

$$CC_k = \frac{\langle p_k^0, p_k^0 \rangle}{\|p_k^0\| \|p_k^1\|}, \quad k = 0, 1, \cdots 15 \quad (7)$$

In (7), the numerator represents inner product of the two sets, and the denominator represents norm of each set. The number of elements in p_k^0 and p_k^1 depends on k. That is, in the case of $k = 0$ or $k = 8$, the edge direction correctly determines only eight boundary pixels. However, in other cases, the number of element of the sets p_k^0 and p_k^1 should be increased because only eight boundary pixels cannot determine the edge direction correctly. Table 1 shows the number of boundary pixels, according to k, to obtain the true edge direction. The true edge direction of the current block is the direction index k that gives the maximum correlation coefficient.

Table 1. Number of boundary pixels to obtain the true edge direction.

k	Number of pixels	k	Number of pixels
0, 8	8	4, 12	15
1, 9	10	5, 13	13
2, 10	11	6, 14	11
3, 11	13	7, 15	10

Now, we describe how the number of boundary pixels is determined according to k. For example, if the edge direction is $\theta = 45^\circ (k = 4)$, then we have the following two sets of boundary pixels (see Fig. 5).

$$p_4^0 = \{(2,0),(3,0),\cdots,(9,0),(9,1),\cdots,(9,7)\} \quad (8a)$$

$$p_4^1 = \{(0,2),(0,3),\cdots,(0,9),(1,9),\cdots,(7,9)\} \quad (8b)$$

The set $p_{k=4}^0$ is composed of the boundary pixels joined on the vertical line $x = 0$ and the horizontal line $y = 9$. And the set $p_{k=4}^1$ is composed of the boundary pixels across the set $p_{k=4}^0$. The lines have to pass through all the decoded pixels within the current block. Therefore, minimum number of boundary pixels for $k = 4$ is 15.

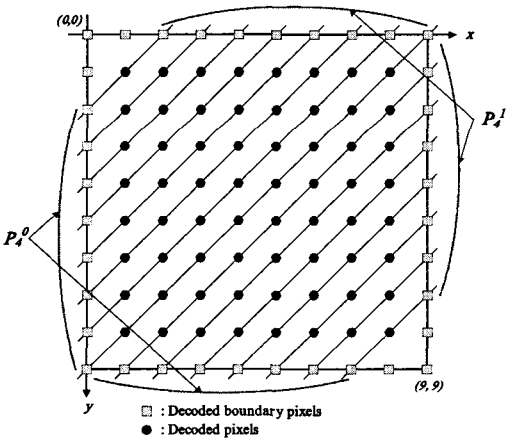


Fig 5. Illustration of the two sets to obtain the correlation coefficients at $k = 4$.

3.3 Interpolation scheme

In case of the following directions such as 0°, 45°, 90°, and 135°, the lines that pass through the pixel position in the current block are joined to the boundary pixels on the x axis or y axis. However, except for the above cases, some lines that pass through the pixel position in the current block are not joined to the boundary pixels. In such a case, to supplement appropriate number of pixels on each line, we adopt the linear interpolation scheme using the nearest neighboring pixels. To obtain the specific interpolated boundary pixels, we introduce the concept of changing the slope of line according to k.

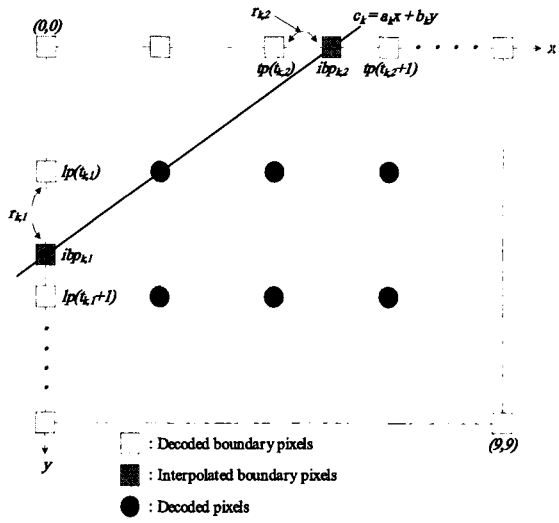


Fig 6. Illustration of the linear interpolation scheme

Fig. 7 shows the linear interpolation scheme to obtain the interpolated boundary pixels. If the line that passes through the decoded pixel is not joined to the decoded boundary pixels, the interpolated boundary pixels are obtained by using both the slope of the line, an intercept of the line, and the distance between the intercept position and the neighboring pixels. The proposed interpolation scheme consists of the following four steps:

Step 1: We first obtain the equation of the line according to k, and then obtain both the coefficient of x, a_k and the coefficient of y, b_k. The table 2 shows the constant values of a_k and b_k according to k, where we can see that all the values of coefficients a_k and b_k for $k \in [1,7]$ are the same values for $k \in [9,15]$, with the block size $N = 8$. These results are explained as follows: Let the boundary pixels rotate 90° clockwise, as shown in Fig. 8. Then lp is changed to tp, and tp is to rp, and so on. In Fig. 8, we can see that the

direction for $k = 12$ is the same as the direction for $k = 4$ in the rotated figure. If the boundary pixels in each block are rearranged by (9), the coefficients a_k and b_k for $k \in [9,15]$ don't need to be calculated.

$$\begin{cases} new\_lp(i) = bp(i) \\ new\_tp(i) = lp(9-i) \\ new\_rp(i) = tp(i) \\ new\_bp(i) = rp(9-i) \end{cases}, \quad i = 0,1,....,9 \tag{9}$$

Step 2: Assume that c_k is an intercept of the line as the direction index k. Then c_k is obtained by (10). The intercept c_k plays an important role in the linear interpolation. Let a line that passes through the decoded pixel exist. The line intersects two points on both the x axis and y axis, as shown in Fig. 6. The position of these two points is obtained by the intercept c_k. In (10), the coefficients a_k and b_k are gained in Table 2. Therefore, the intercept c_k represents the m-th line passing through the point (x,y) in the current decoded block.

$$c_k = a_k x + b_k y, \; x,y = 1,2,\cdots,8 \tag{10}$$

Table 2. Coefficient values of a_k and b_k according to k.

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			

Step 3: In this step, we obtain the boundary pixel positions $t_{k,1}$ and $t_{k,2}$, which are needed to perform the linear interpolation. The position $t_{k,1}$ represents a point at which the m-th line c_k intersects the line $y = 0$ or $x = 9$. Also $t_{k,2}$ represents a point where the m-th line c_k intersects with the line $x = 0$ or $y = 9$. Therefore, the positions $t_{k,1}$ and $t_{k,2}$ have different values according to k and c_k, in Table 3, where $\lceil x \rceil$ represents the nearest integer which does not exceed to x.

Table 3. Postition values $t_{k,1}$ and $t_{k,2}$.

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

Step 4: The m-th line c_k intersects two points on the x axis and y axis. However, the original boundary pixel does not exist. Let $ibp_{k,1}$ and $ibp_{k,2}$ be the interpolated boundary pixels. Then $ibp_{k,1}$ and $ibp_{k,2}$ are obtained by interpolating from two nearest neighboring pixels in Table 4, where the boundary pixels in each block are rearranged by (9) when $9 \le k \le 15$. Table 5 shows the distances, $r_{k,1}$ and $r_{k,2}$, from the intersect point to the nearest neighboring pixel. The symbol $\mathbf{0}$, represents the remainder operations.

Table 4. Interpolated boundary pixels $ibp_{k,1}$ and $ibp_{k,2}$.

k	c_k	$ibp_{k,1}$	$ibp_{k,2}$
0	$8a_k \le c_k \le 8b_k$	$lp(t_{k,1})$	$rp(t_{k,2})$
1, 2, 3, 9, 10, 11	$c_k \le 9a_k$	$\frac{(b_k - r_{k,1})lp(t_{k,1}) + r_{k,1}lp(t_{k,1}+1)}{b_k}$	$\frac{(a_k - r_{k,2})tp(t_{k,1}) + r_{k,2}tp(t_{k,2}+1)}{a_k}$
	$9a_k < c_k \le 9b_k$	$\frac{(b_k - r_{k,1})lp(t_{k,1}) + r_{k,1}lp(t_{k,1}+1)}{b_k}$	$\frac{(b_k - r_{k,2})rp(t_{k,2}) + r_{k,2}rp(t_{k,2}+1)}{b_k}$
	$c_k > 9b_k$	$\frac{(a_k - r_{k,1})bp(t_{k,1}) + r_{k,1}bp(t_{k,1}+1)}{a_k}$	$\frac{(b_k - r_{k,2})rp(t_{k,2}) + r_{k,2}rp(t_{k,2}+1)}{b_k}$
4, 12	$c_k \le 9a_k$	$\frac{(b_k - r_{k,1})lp(t_{k,1}) + r_{k,1}lp(t_{k,1}+1)}{b_k}$	$\frac{(a_k - r_{k,2})tp(t_{k,1}) + r_{k,2}tp(t_{k,2}+1)}{a_k}$
	$c_k > 9b_k$	$\frac{(a_k - r_{k,1})bp(t_{k,1}) + r_{k,1}bp(t_{k,1}+1)}{a_k}$	$\frac{(b_k - r_{k,2})rp(t_{k,2}) + r_{k,2}rp(t_{k,2}+1)}{b_k}$
5, 6, 7, 13, 14, 15	$c_k \le 9b_k$	$\frac{(b_k - r_{k,1})lp(t_{k,1}) + r_{k,1}lp(t_{k,1}+1)}{b_k}$	$\frac{(a_k - r_{k,2})tp(t_{k,1}) + r_{k,2}tp(t_{k,2}+1)}{a_k}$
	$9b_k < c_k \le 9a_k$	$\frac{(a_k - r_{k,1})bp(t_{k,1}) + r_{k,1}bp(t_{k,1}+1)}{a_k}$	$\frac{(a_k - r_{k,2})tp(t_{k,1}) + r_{k,2}tp(t_{k,2}+1)}{a_k}$
	$c_k > 9a_k$	$\frac{(a_k - r_{k,1})bp(t_{k,1}) + r_{k,1}bp(t_{k,1}+1)}{a_k}$	$\frac{(b_k - r_{k,2})rp(t_{k,2}) + r_{k,2}rp(t_{k,2}+1)}{b_k}$
8	$8b_k \le c_k \le 8a_k$	$tp(t_{k,1})$	$bp(t_{k,2})$

Table 5. Postition values $t_{k,1}$ and $t_{k,2}$.

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

3.4 Directional postprocessing scheme

The ringing effect around sharp edge is typical artifact in wavelet-based coding, caused by the coarse quantization of high frequency components. However, Shen and Cuo[6] do not consider the directionality of edges. That is, the filtered output e in (1) has no influence on the edge direction. Since human vision is sensitive to ringing artifacts around the edges, the window shape varying to the edge direction for the nonlinear filtering is desirable.

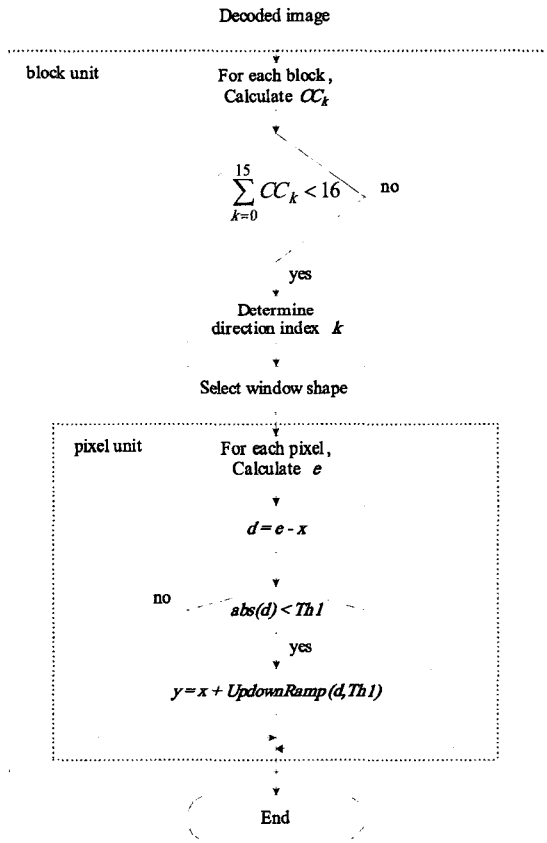


Fig 7. Flowchart of the proposed method

Fig. 7 shows a block diagram of the proposed post-processing technique. We first obtain the correlation coefficient CC_k and calculate its sum. If the sum of correlation coefficients is larger than 16, then the postprocessing is not performed for reducing the computational complexity. If the edge does not exist in the current block, the correlation coefficient of all directions is 1. However, if an edge exists, the single direction values about 1 with the maximum correlation coefficient. The correlation coefficient of other directions is smaller than 1. If an edge exists in the current block, *one direction with the maximum correlation coefficient* is determined. Then the window shape for the nonlinear filtering should be determined along that direction. Fig. 8 shows proposed window shapes as the direction index . The nonlinear filtering technique used the only plus-shaped window, but we alter the window shape as the edge direction. Now the nonlinear filtering along that the direction is performed to remove the ringing artifacts.

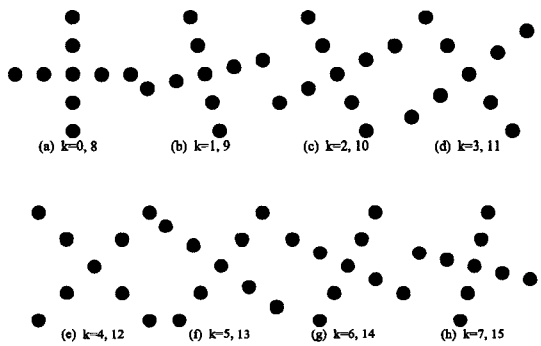


Fig 8. Proposed window shapes as the direction index k.

4. Experimental Results

We apply the proposed postprocessing technique to six test images, such as Target(512×512), Bike (2048×2560), Woman(2048×2560), Cafe (2048×2560), Aerial2(2048×2048) and Lena (512×512), at several bit rates with wavelet-based coding algorithms, JPEG-2000 VM 5.02 with the basic options, (7,9) biorthogonal filters and five-level decomposition. Threshold value $Th1$ is set to 8, 10, and 12 at bit rates equal to 0.25, 0.125, and 0.6125bpp, respectively. The potential function Lorenz is used and γ is set to 3. To determine the edge direction, the block size is set to 8×8. Table 7 shows *PSNR* of the nonlinear filtering technique and the

proposed algorithm at several bit rates. The proposed technique shows better performance than the nonlinear filtering one in all test images and all bit rates. These results are reasonable, since the proposed technique considers the edge direction and performs the nonlinear filtering along the detected edge direction.

Fig. 9 shows the subjective comparison between the proposed technique and the nonlinear filtering technique for the image at 0.125 bpp. The results show that while ringing artifacts are smoothed, the sharp edges have been well preserved by the proposed technique. Note that the nonlinear filtering technique still retains the ringing artifacts around the sharp edges, but in the proposed technique, the ringing artifacts are less visible.

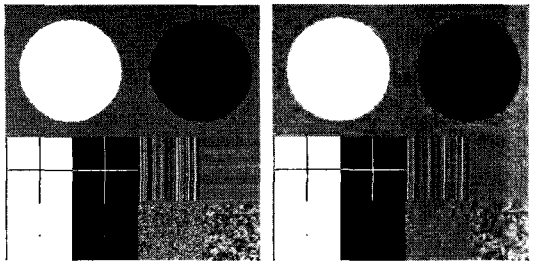
We use the same threshold for the nonlinear filtering. However, the postprocessing efficiency would be subject to threshold value. In case of Lena image, the best PSNR performance can be obtained at lower threshold values than that of the Target image. The postprocessing performance could be optimized in a rate-distortion sense by how to select the threshold. This needs further study.

Table 6. PSNR performances(dB)

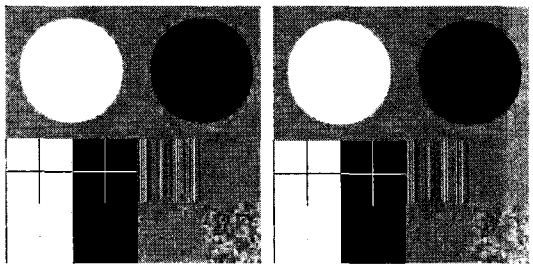
k	$\{h_k\}$ and $\{v_k\}$
0,1,2, 8,9,10	$h_k(0)=[(b_k-2a_k)x(i,j-2)+2a_kx(i+1,j-2)]/b_k$ $h_k(1)=[(b_k-a_k)x(i,j-1)+a_kx(i+1,j-1)]/b_k$ $h_k(3)=[(b_k-a_k)x(i,j+1)+a_kx(i-1,j+1)]/b_k$ $h_k(4)=[(b_k-2a_k)x(i,j+2)+2a_kx(i-1,j+2)]/b_k$ $v_k(0)=[(b_k-2a_k)x(i-2,j)+2a_kx(i-2,j-1)]/b_k$ $v_k(1)=[(b_k-a_k)x(i-1,j)+a_kx(i-1,j-1)]/b_k$ $v_k(3)=[(b_k-a_k)x(i+1,j)+a_kx(i+1,j+1)]/b_k$ $v_k(4)=[(b_k-2a_k)x(i+2,j)+2a_kx(i+2,j+1)]/b_k$
3,11	$h_k(0)=[a_kx(i+1,j-2)+(b_k-a_k)x(i+2,j-2)]/b_k$ $h_k(1)=[(b_k-a_k)x(i,j-1)+a_kx(i+1,j-1)]/b_k$ $h_k(3)=[(b_k-a_k)x(i,j+1)+a_kx(i-1,j+1)]/b_k$ $h_k(4)=[a_kx(i-1,j+2)+(b_k-a_k)x(i-2,j+2)]/b_k$ $v_k(0)=[a_kx(i-2,j-1)+(b_k-a_k)x(i-2,j-2)]/b_k$ $v_k(1)=[(b_k-a_k)x(i-1,j)+a_kx(i-1,j-1)]/b_k$ $v_k(3)=[(b_k-a_k)x(i+1,j)+a_kx(i+1,j+1)]/b_k$ $v_k(4)=[a_kx(i+2,j+1)+(b_k-a_k)x(i+2,j+2)]/b_k$
4,12	$h_k(0)=[a_kx(i+2,j-2)]/b_k$ $h_k(1)=[a_kx(i+1,j-1)]/b_k$ $h_k(3)=[a_kx(i-1,j+1)]/b_k$ $h_k(4)=[a_kx(i-2,j+2)]/b_k$ $v_k(0)=[a_kx(i-2,j-2)]/b_k$ $v_k(1)=[a_kx(i-1,j-1)]/b_k$ $v_k(3)=[a_kx(i+1,j+1)]/b_k$ $v_k(4)=[a_kx(i+2,j+2)]/b_k$
5,13	$h_k(0)=[b_kx(i+2,j-1)+(a_k-b_k)x(i+2,j-2)]/a_k$ $h_k(1)=[(a_k-b_k)x(i+1,j)+b_kx(i+1,j-1)]/a_k$ $h_k(3)=[(a_k-b_k)x(i-1,j)+b_kx(i-1,j+1)]/a_k$ $h_k(4)=[b_kx(i+1,j+2)+(a_k-b_k)x(i-2,j+2)]/a_k$ $v_k(0)=[b_kx(i-1,j-2)+(a_k-b_k)x(i-2,j-2)]/a_k$ $v_k(1)=[(a_k-b_k)x(i,j-1)+b_kx(i-1,j-1)]/a_k$ $v_k(3)=[(a_k-b_k)x(i,j+1)+b_kx(i+1,j+1)]/a_k$ $v_k(4)=[b_kx(i+1,j+2)+(a_k-b_k)x(i+2,j+2)]/a_k$
6,7, 14,15	$h_k(0)=[(a_k-2b_k)x(i+2,j)+2b_kx(i+2,j-1)]/a_k$ $h_k(1)=[(a_k-b_k)x(i+1,j)+b_kx(i+1,j-1)]/a_k$ $h_k(3)=[(a_k-b_k)x(i-1,j)+b_kx(i-1,j+1)]/a_k$ $h_k(4)=[(a_k-2b_k)x(i-2,j)+2b_kx(i-2,j+1)]/a_k$ $v_k(0)=[(a_k-2b_k)x(i,j-2)+2b_kx(i-1,j-2)]/a_k$ $v_k(1)=[(a_k-b_k)x(i,j-1)+b_kx(i-1,j-1)]/a_k$ $v_k(3)=[(a_k-b_k)x(i,j+1)+b_kx(i+1,j+1)]/a_k$ $v_k(4)=[(a_k-2b_k)x(i,j+2)+2b_kx(i-1,j+2)]/a_k$

Table 7. PSNR performances(dB)

Image	0.25bpp		0.125bpp		0.065bpp	
	Shen	Proposed	Shen	Proposed	Shen	Proposed
Aerial2	28.01	28.41	26.23	26.45	24.42	24.60
Bike	29.55	29.80	26.36	26.57	29.80	23.99
Cafe	23.12	23.24	20.73	20.85	19.01	19.09
Target	28.02	28.22	23.73	23.82	20.20	20.27
Lena	35.15	33.97	30.39	30.96	27.67	27.97
Woman	29.66	29.92	27.13	27.31	25.38	25.53



(a) Original image (b) Decoded image(0.25bpp)



(c) Result from [6] (d) The proposed method

Fig 9. Result images.

5. Conclusion

In this paper, we proposed a new directional postprocessing technique to reduce artifacts in wavelet-based coded images. We detect edge direction in each block by finding the maximum correlation coefficients of boundary pixels of blocks neighboring the current block. And we apply the proposed technique to the decoded images according to edge direction. To reduce the computational complexity, we determine whether the current block includes edges or not before postprocessing. That is, if the correlation coefficients of the current block are equivalent to each direction, the current block is regarded as the flat area and then postprocessing is not performed.

Experimental result shows that the proposed method gives both subjectively and objectively excellent results compared to the nonlinear filtering technique. And the proposed technique could greatly enhance the visual quality of decoded images with a low computational cost. Although this paper mainly focuses on JPEG-2000 encoded images, it is possible to extend the proposed technique to video codecs, such as MPEG-1, MPEG-2 and MPEG-4, because the proposed technique is based on the block and macroblock.

References

[1] M. Shen and C. Kuo, "Review of postprocessing techniques for compression artifact removal," Journal of Visual Communication and Image Representation, vol. 9, no. 1, pp. 2-14, Mar. 1998.

[2] ISO/IEC JTC1/SC29/WG11 N1422, JPEG-2000 Verification Model 5.2, Aug. 1999.

[3] T. O'Rourke and R. Stevenson, "Improved image decompression for reduced transform coding artifacts," IEEE Trans. on Circuits and Systems for Video Technology, vol. 5, no. 6, pp. 490-499, Dec. 1995.

[4] J. Li and C. Kuo, "Coding artifact removal with multiscale postprocessing," in Proc. IEEE Int. Conf. Image Processing (ICIP' 97), Santa Barbara, CA, Oct. 1997.

[5] J. Luo, C. Chen, K. Parker, and T. Huang, Artifact reduction in low bit rate DCT-based image compression," IEEE Trans. Image Processing, vol. 5, no. 9, pp. 1363-1368, Sept. 1996.

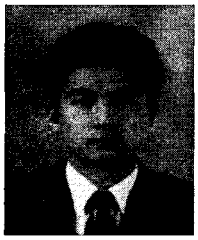
[6] M. Shen and C. Kuo, "Low-complexity postprocessing of wavelet-coded images via robust estimation and nonlinear filtering," in Proc. SPIE Visual Communications and Image Processing (VCIP' 99), San Jose, California, vol. 3653, pp.117-128, Jan. 1999.

[7] T. Chen, H. Wu, B. Qiu, "Adaptive postfiltering of transform coefficients for the reduction of blocking artifacts," Trans. on Circuits and Systems for Video Technology, Vol. 11, Issue 5, pp.594-602, May 2001

[8] F. Coudoux, M. Gazalet, P. Corlay, "An adaptive postprocessing technique for the reduction of color bleeding in DCT-coded images," Trans. on Circuits and Systems for Video Technology, Vol. 14, Issue 1, pp. 114 - 121, Jan. 2004

[9] Y, Zhao, G. Cheng, S. Yu, "Postprocessing technique for blocking artifacts reduction in DCT domain," Electronics Letters, Vol. 40, Issue 19, pp.1175-1176, Sept. 2004

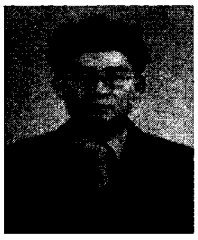
Byong-Seok Min [Regular Member]



- 1990. 2. : Hanyang Univ., Dept. of Electronic Communications Eng.(Bachelor)
- 1992. 8. : Hanyang Univ., Dept. of Electronic Communications Eng.(Master)
- 2002. 8. : Hanyang Univ., Dept. of Electronic Communications Eng.(Ph.D)
- 2003. 7. ~ 2004. 7.: The Univ. of Alberta, Dept. of Electric and Computer Eng. Visiting Professor
- 1995. 3. ~ present : Chungcheong Univ., Dept. of Electronic Communications, Professor

<Research Area>
Image Processing & Communication, Multimedia

Seung-Jong Kim [Regular Member]



- 1992. 2. : Hanyang Univ., Dept. of Mathematics(Bachelor)
- 1994. 2. : Hanyang Univ., Dept. of Electronic Communications Eng.(Master)
- 2000. 8. : Hanyang Univ., Dept. of Electronic Communications Eng.(Ph.D)
- 2000. 9. ~ present : Hanyang Women's College, Dept. of Computer Information, Professor

<Research Area>
Image Processing & Communication, Multimedia,

Dong-Kyun Lim [Regular Member]

- 1985. 2. : Hanyang Univ., Dept. of Electronic Communications Eng.(Bachelor)
- 1993. 2. : Hanyang Univ., Dept. of Electronic Communications Eng.(Master)
- 2001. 2. : Hanyang Univ., Dept. of Electronic Communications Eng.(Ph.D)
- 2003. 2. ~ present : Hanyangcyber Univ., Dept. of Computer Eng., Professor

<Research Area>
Control, Signal Processing, Microprocessor