Weighted DCT-IF for Image up Scaling

Jae-Yung Lee[1], Sung-Jun Yoon[2], Jae-Gon Kim[3] and Jong-Ki Han[1*]
[1] Department of Electrical Engineering, Sejong University
Seoul, South Korea
[e-mail: ljy321456@gmail.com, hjk@sejong.edu]
[2] SK Telecom,
Seoul, South Korea
[e-mail: sjyoon1671@gmail.com]
[3] School of Electronics and Information Engineering, Korea Aerospace University
Goyang, South Korea
[e-mail: jgkim@kau.ac.kr]
*Corresponding author: Jong-Ki Han

Abstract

The design of an efficient scaler to enhance the edge data is one of the most important issues in video signal applications, because the perceptual quality of the processed image is sensitively affected by the degradation of edge data. Various conventional scaling schemes have been proposed to enhance the edge data. In this paper, we propose an efficient scaling algorithm for this purpose. The proposed method is based on the discrete cosine transform-based interpolation filter (DCT-IF) because it outperforms other scaling algorithms in various configurations. The proposed DCT-IF incorporates weighting parameters that are optimized for training data. Simulation results show that the quality of the resized image produced by the proposed DCT-IF is much higher than that of those produced by the conventional schemes, although the proposed DCT-IF is more complex than other conventional scaling algorithms.

Keywords: Up/down scaler, DCT-IF, Edge enhancement

1. Introduction

The scalers are typically found inside consumer electronics devices such as HD/UHD televisions, mobile devices, DVD, Blu-ray disc players, and video game consoles. The scaler modules are often combined with other video processing devices or algorithms to create a video processing module that improves the apparent definition of video signals. In some systems, scaling is done by video processing chips that are installed in digital display devices, where the role of the video processing chip would be to upscale and process the video signal to fit the display resolution. As the resolutions supported by these systems become more diverse, the design of an efficient scaler module to increase the quality of the resized image becomes more important.

A scaling scheme based on interpolation is used to reconstruct the intermediate values between discrete pixel values. Over the last several decades, various interpolation methods, including the nearest neighbor [1], bilinear [1], cubic B-spline [2], cubic convolution [3], interpolation using wavelet transform [4], and discrete cosine transform-based interpolation filter (DCT-IF) methods [5], have been studied to increase the quality of the resized image. Because edge data are visually important information in the image, the enhancement of edge information is one of the most crucial issues in applications using images and video signals. While edge data contain detailed information in the spatial domain, such data also contain high-frequency components in the frequency domain. Because the kernels of various scalers cause blurring effects like a low pass filter (LPF), blurred edge data are more frequently found in the resized image as the scaling ratio increases. To overcome this problem, various interpolation methods to enhance the edge data have been researched [6]-[15]. The directional cubic convolution [6] proposed by Zhou *et al.* preserves sharp edges and details by measuring the direction of the edges and combining weighted data at weak edge locations. Whereas the directional cubic convolution [6] supports only an integer scaling ratio, the scheme proposed by Rui and Qiong [7] resizes an image with an arbitrary scaling ratio, where the authors analyzed the relationship between the polynomial coefficients of the bicubic interpolation method and proposed a new kernel to obtain a high-resolution image by enhancing the edge information. The curvature interpolation method [8] was designed by estimating the curvature of a low-resolution (LR) image, with Jacobi iteration used to solve some equations related to the problem. The discrete wavelet transform (DWT) and stationary wavelet transform (SWT) based interpolation scheme proposed by H. Demirel *et al.* [9] also resizes an image considering frequency subband images. LR images are firstly decomposed different subband images using the DWT. After calibrating the decomposed high frequency subband images with high frequency subband images obtained by the SWT, all these subbands are combined to generate a new high resolution image by using inverse DWT (IDWT). The scheme is easy to implement, but the quality of obtained HR images tends to be poor. Huang *et al.* [10] proposed a novel interpolation chip that used the interpolation error theorem to enhance the edge features of the resized image. On the other hand, unlike conventional interpolation schemes concentrating on local linear relationship among neighboring pixels to reconstruct a high-resolution (HR) image form a corresponding LR input image, F. Cao *et al.* proposed a novel interpolation scheme which is based on a low-rank matrix completion to determine the local order of the linear model implicitly. The scheme can efficiently obtain a HR image and handle noisy data such as image denoising because of the low-rank matrix completion [11]. J. –K. Chae *et al.* concentrate on solving a scaling problem to minimize the information loss

resulting from the resizing process. They proposed a optimized cubic convolution kernel considering the loss in both spatial and frequency domains. The performance of the scheme improved in terms of peak signal-to-noise ratio (PSNR) without a large increase in computational complexity [12]. S. Zhu *et al.* noticed that interpolation schemes which are based on solving minimum mean-square-error (MMSE) for local geometric similarity between a low and high resolution image cannot generate a reliable high resolution image. To solve the problem, they consider nonlocal geometric similarity which consists of MMSE and least squares error that is built upon a number of dual-reference patches drawn from the given LR image and regularized by the directional gradients of these patches [13].

In this paper, we propose a weighted DCT-IF (W-DCT-IF) method, where the weighting parameters are optimized to enhance the edge information. The weighting parameters are calculated using an optimization algorithm, which is similar to the scheme used to make a Wiener filter.

This paper is organized as follows. In Section 2, the DCT-IF is briefly explained. The problem to be solved is formulated in Section 3, where the weighted DCT-IF is modeled, and the adaptive edge enhancement scheme is explained. The optimization schemes for the parameters of the weighted DCT-IF are proposed in Section 4, where one-dimensional and two-dimensional models are optimized for training data. In Section 5, the performance of the proposed technique is evaluated and compared with those of several conventional schemes. Section 6 concludes this paper.

2. DCT-IF

DCT-IF is an interpolation scheme whose kernel is constructed by combining the coefficients of DCT and inverse DCT. Among the various types of DCT, the most common type is type-II because it is asymptotically equivalent to the optimal Karhunen–Loève transform (KLT) [16]. DCT-IF is one of the best interpolation methods and provides highly accurate interpolation [5]. The DCT-IF especially has remarkable performances in comparison with other conventional interpolation methods such as bilinear and cubic convolution at an arbitrary scaling ratio.

When the given pixels are denoted by $f(i)$, the value $\hat{f}(s)$ interpolated by DCT-IF is represented as follows [17], [18].

$$\hat{f}(s) = \sum_{i=0}^{M-1} f(i) \sum_{u=0}^{M-1} a^2(u) C_{2M}^{(2s+1)u} C_{2M}^{(2i+1)u} \tag{1}$$

where

$$a(u) = \begin{cases} \sqrt{\dfrac{1}{M}}, & if \ u = 0 \\[4mm] \sqrt{\dfrac{2}{M}}, & else \ if \ u \neq 0 \end{cases} \tag{2}$$

$$C_k^l = \cos(\frac{\pi l}{k})$$ (3)

Note that s is a position to be interpolated, and M is the number of taps.

When DCT-IF is applied to a 2D signal such as a picture, the DCT-IF can be extended to 2D DCT-IF as follows.

$$\hat{f}(s,z) = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} f(i,j) \left\{ \begin{array}{l} \sum_{u=0}^{M-1} a^2(u) C_{2M}^{(2s+1)u} C_{2M}^{(2i+1)u} \\ \\ \times \sum_{v=0}^{N-1} a^2(v) C_{2N}^{(2z+1)v} C_{2N}^{(2j+1)v} \end{array} \right\}$$ (4)

where

$$a(u) = \begin{cases} \sqrt{\dfrac{1}{M}}, & \textit{if } u = 0 \\ \\ \sqrt{\dfrac{2}{M}}, & \textit{else if } u \neq 0 \end{cases}$$ (5)

$$a(v) = \begin{cases} \sqrt{\dfrac{1}{N}}, & \textit{if } v = 0 \\ \\ \sqrt{\dfrac{2}{N}}, & \textit{else if } v \neq 0 \end{cases}$$ (6)

In (4), s and z are vertical and horizontal positions to be interpolated, and M and N are the numbers of taps in the vertical and horizontal directions, respectively. (s,z) indicates the arbitrary interpolation point between the locations of the given pixels. The interpolated pixel value $\hat{f}(s,z)$ is found using the weighted sum of the neighbor pixels around (s,z) in the given image. Note that the DCT-IF is a separable process [19]. Thus, 2D DCT-IF can be implemented by consecutively applying the 1D DCT-IF of (1) in the horizontal and vertical directions.

DCT-IF has been employed for motion compensation in a high-efficiency video codec (HEVC) [5], [20], where the fixed coefficients of DCT-IF are prepared for 1/2 and 1/4 pixel positions because the motion vectors are searched at those resolutions in HEVC. However, in various applications using up/down scalers, the interpolation filter would resize the resolution of the original image at an arbitrary scaling ratio, where the coefficients of DCT-IF should be calculated at each interpolation position. In this paper, we modify the coefficients of DCT-IF to enhance the edge information of the resized image. Because the edge contains important information related to the perceptual quality, the optimization of the coefficients of DCT-IF to enhance the edge data is a crucial problem in a variety of image processing applications.

3. Problem Formulation
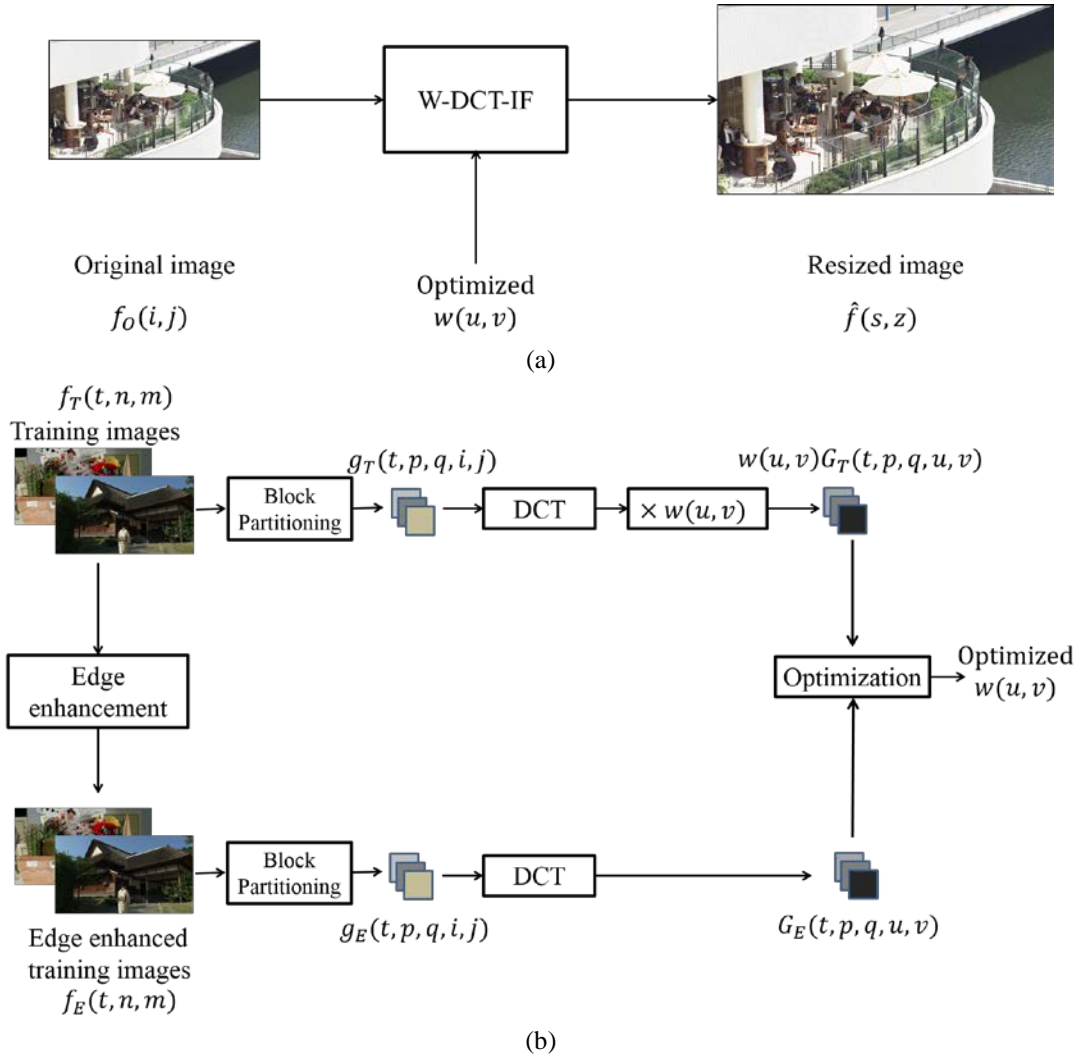
3.1 Interpolation system using weighted DCT-IF

(a)

(b)

Fig. 1. (a) Weighted DCT-IF (W-DCT-IF) and (b) optimization process of $w(u,v)$

Fig. 1 (a) shows the interpolation system using DCT-IF, where the coefficients of DCT-IF are weighted to enhance the edge information. The W-DCT-IF is derived as follows. The DCT data of the original image $f_O(i,j)$ are calculated by the following equation.

$$F_O(u,v) = a(u)a(v) \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f_O(i,j) C_{2M}^{(2i+1)u} C_{2N}^{(2j+1)v} \tag{7}$$

If the DCT data $F_O(u,v)$ are returned to the pixel domain after applying the weighted parameters $w(u,v)$ to the DCT data, the reconstructed pixel data are

$$\hat{f}(s,z) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} a(u)a(v)w(u,v)F_O(u,v)C_{2M}^{(2s+1)u}C_{2N}^{(2z+1)v} \tag{8}$$

where $w(u,v)$ is the weighting parameters for the corresponding frequencies (u,v). Combining (7) and (8) produces

$$\hat{f}(s,z) = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} f_O(i,j) \left[\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} \begin{array}{l} a^2(u)a^2(v)w(u,v) \\ \times C_{2M}^{(2s+1)u}C_{2N}^{(2z+1)v}C_{2M}^{(2i+1)u}C_{2N}^{(2j+1)v} \end{array} \right] \tag{9}$$

Equation (9) represents the 2D W-DCT-IF scheme proposed in this paper. We optimize the weighting parameter $w(u,v)$ to enhance the edge information of the resized image.

In the optimization process for the weighting parameter $w(u,v)$ shown in **Fig. 1 (b)**, the weighting parameters are calculated using various training images. Because the main goal in this paper is to modify the coefficients of DCT-IF to enhance the edge information of the resized image, the proposed algorithm is applied to the gray signal. When we resize the color image, the algorithm is used for Y signal only, where Y is a component of the YUV system. The training images are represented as $f_T(t,n,m)$, where t is an index for training images, and n and m are indexes for the row and column locations of each pixel in the ith training image. When the size of DCT is $M \times N$, each training image is divided into disjointed blocks whose size is $M \times N$. In **Fig. 1 (b)**, $g_T(t,p,q,i,j)$ denotes the (i^{th}, j^{th}) pixel value in the (p^{th}, q^{th}) block of the tth training image $f_T(t,n,m)$. p and q are the indexes of the blocks in the vertical and horizontal directions, respectively. The relation is represented as follows:

$$g_T(t,p,q,i,j) = f_T(t,Mp+i,Nq+j) \tag{10}$$

Here, $f_E(t,n,m)$ is the tth edge enhanced training image created by applying an edge enhancement scheme to the tth training image $f_T(t,n,m)$. In this figure, $g_T(t,p,q,i,j)$ denotes the (i^{th}, j^{th}) pixel value in the (p^{th}, q^{th}) block of the tth edge enhanced image $f_E(t,n,m)$. The relation is represented as follows:

$$g_E(t,p,q,i,j) = f_E(t,Mp+i,Nq+j) \tag{11}$$

To analyze the property of the training data, each block is transformed by DCT as follows.

$$G_T(t,p,q,u,v) = a(u)a(v)\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} g_T(t,p,q,i,j)C_{2M}^{(2i+1)u}C_{2N}^{(2j+1)v} \tag{12}$$

$$G_E(t,p,q,u,v) = a(u)a(v)\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} g_E(t,p,q,i,j)C_{2M}^{(2i+1)u}C_{2N}^{(2j+1)v} \tag{13}$$

In **Fig. 1 (b)**, the weighting parameters $w(u,v)$ are optimized to minimize the total difference between the DCT domain data $G_E(t,p,q,u,v)$ of the edge-enhanced images and

the weighted DCT domain data $w(u,v)G_T(t,p,q,u,v)$ of the training images. The weighting parameters $w(u,v)$ are optimized to minimize the following cost function.

$$D = \sum_{t=0}^{L-1}\sum_{p=0}^{P-1}\sum_{q=0}^{Q-1}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}\left\{ \begin{array}{c} G_E(t,p,q,u,v) \\ -w(u,v)G_T(t,p,q,u,v) \end{array} \right\}^2 \qquad (14)$$

where L is the total number of training images. P and Q denote the numbers of the transform blocks in the vertical and horizontal directions, respectively. The optimization procedure is represented as follows.

$$w(u,v) = \underset{w(u,v)}{\arg\min} D \qquad (15)$$

The optimized $w(u,v)$ can be used in (9) to enhance the edge information of the resized image in the proposed system.

3.2 Edge Enhanced Scheme

In this subsection, we explain the edge enhancement technique used in **Fig. 1 (b)**, which makes the image subjectively clearer by improving the contrast effect in the edge region [21], [22]. The enhancement process can be represented as follows:

$$f_E(t,n,m) = f_T(t,n,m) + h(f_T(t,n,m)) \qquad (16)$$

where $h(f_T(t,n,m))$ denotes the enhancement component at the location (n,m). It depends on the property of pixel values at the local area that includes (n,m). To get the edge information from the image, the Sobel edge detector [21] is applied to each training image $f_T(t,n,m)$, which produces an edge-detected image as shown in **Fig. 2**.
In (16), we design the enhancement component $h(f_T(t,n,m))$ as follows.

$$h(f_T(t,n,m)) = P(e_{n,m}) \times \left(f_T(t,n,m) - f_{LPF}(t,n,m) \right) \qquad (17)$$

where $f_{LPF}(t,n,m)$ is the image produced by applying the LPF to $f_T(t,n,m)$. The LPF used in (17) is

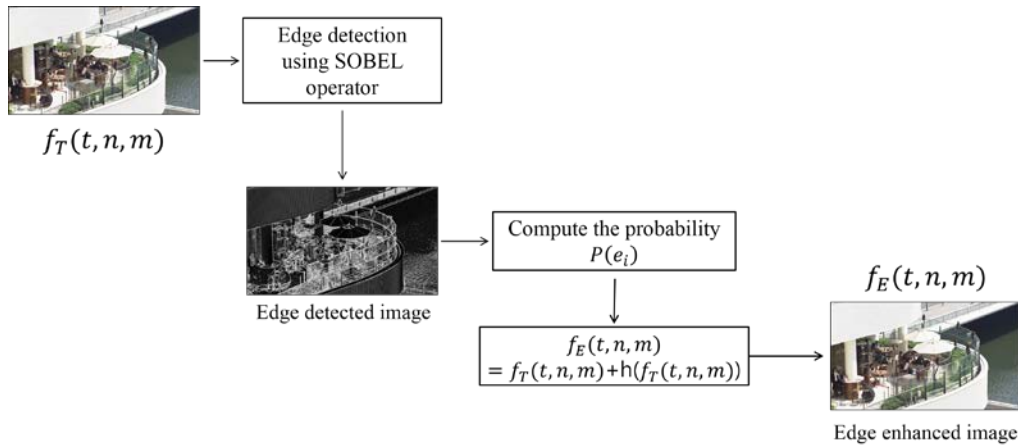$$LFP = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \qquad (18)$$

Fig. 2. Diagram of adaptive edge enhancement

In (17), $P(e_{n,m})$ denotes the probability that the edge pixel $e_{n,m}$ at (n,m) will be blurred. $P(e_{n,m})$ is determined by the psychometric function [23] as follows.

$$P(e_{n,m}) = 1 - \exp\left(-\left| \frac{w(e_{n,m})}{w_{JNB}(e_{n,m})} \right| \right) \tag{19}$$

where $w(e_{n,m})$ and $w_{JNB}(e_{n,m})$ are the widths of the edge $e_{n,m}$ and the edge with Just Noticeable Blur (JNB), respectively. Note that $w_{JNB}(e_{n,m})$ depends on the local contrast in the neighbor of edge $e_{n,m}$. In calculation of $w_{JNB}(e_{n,m})$, the width around the edge is measured by counting the number of pixels with increasing or decreasing gray scale values in one direction of the edge pixel. Whereas the authors of [23] used the equation for $P(e_{n,m})$ shown in (19) to propose an objective metric for evaluating the image sharpness, we use the concept to calculate the amount of alteration of the pixel values for enhancing the edge data. Based on the probability $P(e_{n,m})$, if the blurring will be unnoticeable, the pixel data are not altered. On the other hand, if the probability that the blurring will be noticeable is high, the pixel values are modified, with the amount of alteration proportional to $P(e_{n,m})$.

4. Optimization of Weighting Parameters

In this section, we explain the optimization process for parameters $w(u,v)$ using various training images. Because we use the training images, the parameters are denoted by the Parameters optimized for Training Images (PTI). The cost function of (14) is used in the optimization of PTI. This cost function can be minimized by taking the partial derivative with respect to $w(u,v)$ as follows.

$$\frac{\partial D}{\partial w(u,v)} = -2 \times \sum_{t=0}^{L-1} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \left[\begin{pmatrix} G_E(t,p,q,u,v) \\ -w(u,v)G_T(t,p,q,u,v) \\ \times G_T(t,p,q,u,v) \end{pmatrix} \right] = 0 \tag{20}$$

Equation (20) gives a solution for the weighting parameters as follows.

$$w(u,v) = \frac{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} G_E(t,p,q,u,v) G_T(t,p,q,u,v)}{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} G_T(t,p,q,u,v) G_T(t,p,q,u,v)} \tag{21}$$

The weighting parameter in (21) is used in (9) to implement for 2D W-DCT-IF.

   When PTI is derived using the 1D DCT-IF, the parameters are independently optimized for the horizontal and vertical directions, because the properties of the data in these two directions are different from each other. The 1D DCT-IF scheme and the parameters optimized for the horizontal and vertical directions are represented as follows.

$$\hat{f}(s,j) = \sum_{i=0}^{M-1} f_O(i,j) \left\{ \sum_{u=0}^{M-1} a^2(u) w_V(u) C_{2M}^{(2s+1)u} C_{2M}^{(2i+1)u} \right\} \tag{22}$$

$$\hat{f}(i,z) = \sum_{j=0}^{N-1} f_O(i,j) \left\{ \sum_{v=0}^{N-1} a^2(v) w_H(v) C_{2N}^{(2z+1)v} C_{2N}^{(2j+1)v} \right\} \tag{23}$$

$$w_H(v) = \frac{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} \sum\limits_{i=0}^{M-1} A_H(t,p,q,i,v) B_H(t,p,q,i,v)}{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} \sum\limits_{i=0}^{M-1} B_H(t,p,q,i,v) B_H(t,p,q,i,v)} \tag{24}$$

$$w_V(u) = \frac{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} \sum\limits_{j=0}^{N-1} A_V(t,p,q,u,j) B_V(t,p,q,u,j)}{\sum\limits_{t=0}^{L-1} \sum\limits_{p=0}^{P-1} \sum\limits_{q=0}^{Q-1} \sum\limits_{j=0}^{N-1} B_V(t,p,q,u,j) B_V(t,p,q,u,j)} \tag{25}$$

where

$$A_H(t,p,q,i,v) = a(v) \sum_{j=0}^{N-1} f_E(t,Mp+i,Nq+j) C_{2N}^{(2j+1)v} \tag{26}$$

$$B_H(t,p,q,i,v) = a(v) \sum_{j=0}^{N-1} f_T(t,Mp+i,Nq+j) C_{2N}^{(2j+1)v} \tag{27}$$

$$A_V(t,p,q,u,j) = a(u) \sum_{i=0}^{M-1} f_E(t,Mp+i,Nq+j) C_{2M}^{(2i+1)u} \tag{28}$$

$$B_V(t,p,q,u,j) = a(u) \sum_{i=0}^{M-1} f_T(t,Mp+i,Nq+j) C_{2M}^{(2i+1)u} \tag{29}$$

PTI is made by considering all of the edge information in several training images. Thus, PTI can generally be used for various test images to be scaled, after these have been optimized for the training images. Note that the test image to be scaled is not included in the set of training images. Because the PTI optimization is an offline process, it is not included in the scaling process. Thus, the computational complexity of the scaler system using PTI is similar to those of the conventional DCT-IF technique.

5. Simulation Results

5.1 Implementation

A scaler based on the proposed algorithm can be implemented using the Verilog hardware description language and fitted on FPGA-based system. On the other hand, a software version of the scaler optimized for GPU computing can also be implemented in the GPU-based devices. The software scaler is capable of displaying Full HD (1,920x1,080 pixel) video output at roughly 60 fps on a device incorporating a 3.4GHz 6-Core Processor and main memory of 8 GB.

5.2 Configuration to Evaluate Schemes

We evaluate the performance of the proposed method in this section, where W-DCT-IF using the PTI is compared with various conventional interpolation methods [1]-[3], [7], [8], [10], [17] from the viewpoint of edge enhancement. When PTI is used, the number (L) of training images is set to 20. We employed 20 training images among the dataset composed of 100 standard 512×512 grayscale test images which are shown in **Fig. 4** of [24] to train the proposed weighted DCT-IF. Specifically, the training images are selected from the dataset randomly to avoid bias that might be in the set of training images.

When the weighting coefficients are trained, as shown in **Fig. 1 (b)**, each training image is divided into disjointed blocks whose size is M×N. In the case that the resolution of each training image is 512×512 and the size of block is 8×8, the number of training blocks made from 20 training images is 81,920. I think it is sufficient for our simulations.

We utilize various measurement tools such as PSNR, SSIM, the image sharpness metric, edge map, and subjective quality for objective and subjective evaluations. When PSNR and SSIM are used as measurement tools, we need a reference image, which is compared with the images resized from the various scaling methods. **Fig. 3** shows a block diagram of the PSNR and SSIM measurement. Various scaling algorithms are used to enlarge the low-resolution (LR) image to the resolution of the original image, where the LR image is made by applying LPF and bilinear interpolation to the original image.

Fig. 3. Block diagram of PSNR and SSIM measurement

To evaluate the performances of the techniques for various test images, we use 10 images which are 'FourPeople', 'Johnny', 'KristenAndSara', 'SlideEditing', 'SlideShow' (HD 720 YUV 4:2:0 images), and 'BasketballDrive', 'BQTerrace', 'Cactus', 'Kimono1', 'ParkScene' (HD 1080 YUV 4:2:0 images).

5.3 Evaluation based on PSNR

The PSNRs of the images resulting from the proposed and various conventional schemes are compared in **Tables 1** and **2**. The PSNR is defined as follows.

$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX^2}{MSE}\right) \text{ [dB]} \tag{30}$$

where *MAX* is usually 255 and represents the possible maximum value of pixels when a pixel is represented with 8 bits. The MSE is the mean squared error between the reference picture and a picture resized by one of the various schemes. As PSNR increases, the objective quality of the resized image increases. For YUV 4:2:0 images, PSNR is calculated with the Y component. The 2D W-DCT-IF is the proposed algorithm using the weighting parameters optimized for 2D data, such as those in (21). The 1D W-DCT-IF is the algorithm using (22) - (25).

As observed in **Tables 1** and **2**, the PSNRs of the proposed methods are always much higher than those of the conventional methods [1]-[3], [7], [8], [10], [17]. This is because PTI has been optimized for the training images. In the comparison between 1D W-DCT-IF and 2D W-DCT-IF implementations of the proposed algorithm, the performance of the 1D implementation is slightly lower than that of the 2D implementation, because the training procedure using $w_H(v)$ or $w_V(u)$ makes the filter non-separable. Thus, there are some cases where the optimized parameters $w_H(v)$ and $w_V(u)$ are mismatched for the data to be scaled.

To show the performance tendencies for various scaling ratios, **Fig. 4** shows the PSNRs of the schemes for a variety of ratios. Based on the results of **Tables 1** and **2**, and **Fig. 4**, the proposed algorithm outperforms other conventional schemes in relation to the objective

quality of the resized image.

Table 1. PSNR Comparison of Interpolation Algorithms for 1.79 Times Zoom

Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	33.71	36.62	33.69	19.27	22.78	34.15	27.50	32.66	38.46	34.61	31.35
Cubic B-Spline[2]	35.62	38.02	35.26	19.64	22.78	34.96	28.84	34.00	39.96	36.20	32.53
Cubic Convolution[3]	35.13	37.91	35.07	19.67	23.12	34.89	28.50	33.76	39.75	35.80	32.36
1-D DCT-IF[17]	36.29	38.84	36.00	19.92	23.25	35.41	29.25	34.55	40.54	36.60	33.07
CIM[8]	35.09	37.52	34.90	19.92	23.25	34.86	28.89	34.07	39.64	36.04	32.42
CIM$_5$ [8]	34.88	37.30	34.66	19.86	23.18	34.81	28.62	33.82	39.37	35.84	32.23
NIC [10]	36.26	38.86	36.13	19.98	23.39	35.44	29.36	34.69	40.75	36.79	33.17
ISAI [7]	35.55	38.24	35.46	19.78	23.19	35.09	28.81	34.10	40.15	36.17	32.65
2D W-DCT-IF (PTI)	**37.04**	**39.31**	**36.67**	**20.12**	**23.44**	**35.77**	**29.93**	**35.22**	**41.22**	**37.35**	**33.61**
1D W-DCT-IF (PTI)	36.83	39.19	36.56	20.07	23.36	35.42	29.74	35.04	40.92	37.05	33.42

Table 2. PSNR COMPARISON OF INTERPOLATION ALGORITHMS FOR 1.58 TIMES ZOOM

| Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	34.41	37.29	34.38	19.67	23.11	35.00	28.46	33.62	39.50	35.64	32.11
Cubic B-Spline[2]	36.45	38.75	36.18	20.26	23.33	35.86	30.03	35.12	40.97	37.38	33.43
Cubic Convolution[3]	35.94	38.65	35.82	20.16	23.52	35.85	29.63	34.86	40.86	36.96	33.23
1-D DCT-IF[17]	37.26	39.74	36.95	20.54	23.72	36.45	30.53	35.79	41.66	37.87	34.05
CIM[8]	35.90	38.23	35.63	20.57	23.77	35.72	29.93	35.02	40.33	36.81	33.19
CIM$_5$ [8]	35.70	38.03	35.41	20.47	23.68	35.67	29.64	34.79	40.21	36.77	33.04
NIC [10]	37.32	39.80	37.08	20.58	23.87	36.52	30.67	35.95	41.91	38.09	34.18
ISAI [7]	36.45	39.08	36.28	20.29	23.62	36.07	29.99	35.25	41.26	37.37	33.57
2D W-DCT-IF (PTI)	**38.38**	**40.47**	**37.98**	**20.85**	**23.99**	**36.84**	**31.36**	**36.56**	**42.31**	**38.72**	**34.75**
1D W-DCT-IF (PTI)	38.08	40.32	37.83	20.75	23.88	36.43	31.12	36.34	42.01	38.36	34.51
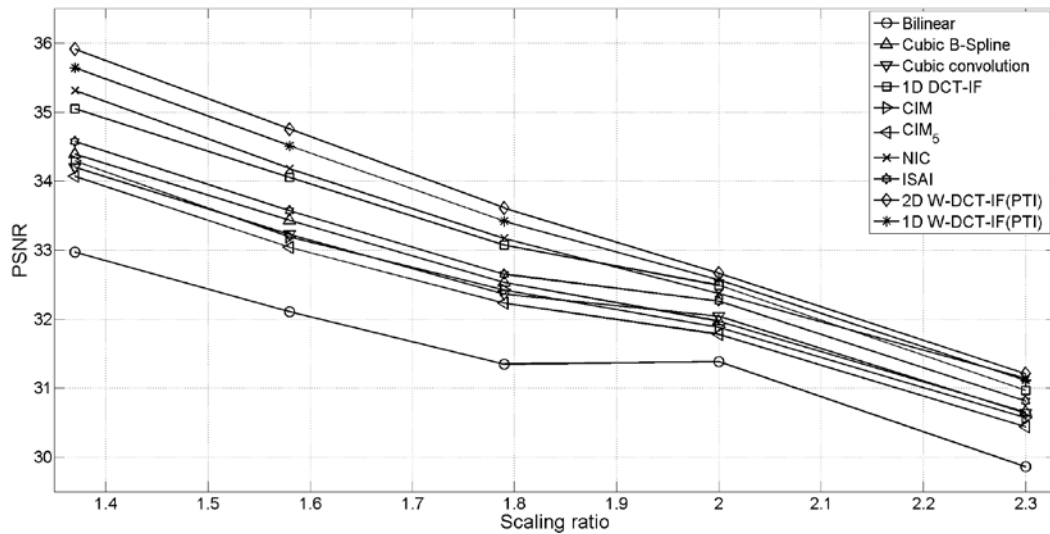
Fig. 4. PSNR comparison at different scaling ratios

5.4 Evaluation based on SSIM

SSIM is one of the most important tools for evaluating the perceptual quality of test images [25]. SSIM is used to evaluate the structural similarity between a reference image and a resized image. SSIM is defined as follows.

$$SSIM = \frac{(2\mu_O\mu_R + c_1)(2\sigma_{OR} + c_2)}{(\mu_O^2 + \mu_R^2 + c_1)(\sigma_O^2 + \sigma_R^2 + c_2)} \tag{31}$$

where μ_O and μ_R are the averages of the pixel values in the reference and resized images, respectively. σ_O and σ_R are the variances of the pixel values in these images, respectively. σ_{OR} implies the covariance of pixel values in the reference and resized images. As SSIM increases, the structural similarity between the resized image and the reference picture increases.

Table 3. SSIM COMPARISON OF INTERPOLATION ALGORITHMS FOR 1.79 TIMES ZOOM

Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	0.9561	0.9513	0.9638	0.7535	0.8543	0.9178	0.8521	0.9049	0.9499	0.9212	0.9025
Cubic B-Spline[2]	0.9678	0.9604	0.9708	0.7783	0.8599	0.9254	0.8885	0.9231	0.9594	0.9405	0.9174
Cubic Convolution[3]	0.9659	0.9588	0.9704	0.7778	0.8675	0.9250	0.8788	0.9193	0.9580	0.9362	0.9158
1-D DCT-IF[17]	0.9709	0.9643	0.9732	0.7842	0.8715	0.9286	0.8956	0.9274	0.9622	0.9442	0.9222
CIM[8]	0.9651	0.9603	0.9695	**0.7963**	0.8703	0.9239	0.8936	0.9240	0.9574	0.9411	0.9202
CIM₃ [8]	0.9641	0.9597	0.9687	0.7941	0.8693	0.9235	0.8898	0.9216	0.9562	0.9386	0.9186
NIC [10]	0.9711	0.9634	0.9736	0.7937	0.8766	0.9293	0.8974	0.9290	0.9630	0.9462	0.9243
ISAI [7]	0.9679	0.9604	0.9716	0.7814	0.8692	0.9264	0.8854	0.9228	0.9601	0.9401	0.9185
2D W-DCT-IF (PTI)	**0.9735**	**0.9668**	**0.9745**	0.7939	**0.8781**	**0.9317**	**0.9093**	**0.9342**	**0.9655**	**0.9511**	**0.9279**
1D W-DCT-IF (PTI)	0.9729	0.9667	0.9742	0.7914	0.8751	0.9295	0.9066	0.9322	0.9643	0.9487	0.9262

The SSIMs produced by various scaling schemes are compared in **Tables 3** and **4**, and **Fig. 5**. These results show that the proposed algorithms outperform the various conventional methods in relation to SSIM. The performance tendencies of the algorithms are similar to that of the simulation based on PSNR.

Table 4. SSIM COMPARISON OF INTERPOLATION ALGORITHMS FOR 1.58 TIMES ZOOM

| Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	0.9626	0.9572	0.9683	0.7737	0.8682	0.9276	0.8801	0.9198	0.9580	0.9358	0.9151
Cubic B-Spline[2]	0.9730	0.9666	0.9747	0.8034	0.8789	0.9351	0.9136	0.9365	0.9657	0.9525	0.9300
Cubic Convolution[3]	0.9714	0.9650	0.9744	0.7975	0.8812	0.9347	0.9046	0.9331	0.9650	0.9490	0.9276
1-D DCT-IF[17]	0.9764	0.9713	0.9777	0.8098	0.8875	0.9387	0.9206	0.9407	0.9685	0.9560	0.9347
CIM[8]	0.9701	0.9664	0.9727	**0.8219**	0.8896	0.9333	0.9176	0.9366	0.9626	0.9511	0.9322
CIM₃ [8]	0.9693	0.9659	0.9720	0.8196	0.8887	0.9330	0.9144	0.9348	0.9620	0.9497	0.9309
NIC [10]	0.9766	0.9703	0.9778	0.8145	0.8905	0.9393	0.9216	0.9420	0.9693	0.9576	0.9360
ISAI [7]	0.9735	0.9669	0.9758	0.8022	0.8833	0.9361	0.9109	0.9363	0.9667	0.9523	0.9304
2D W-DCT-IF (PTI)	**0.9796**	**0.9745**	**0.9795**	0.8209	**0.8943**	**0.9419**	**0.9327**	**0.9468**	**0.9712**	**0.9619**	**0.9403**
1D W-DCT-IF (PTI)	0.9788	0.9742	0.9792	0.8178	0.8913	0.9397	0.9302	0.9451	0.9702	0.9597	0.9386
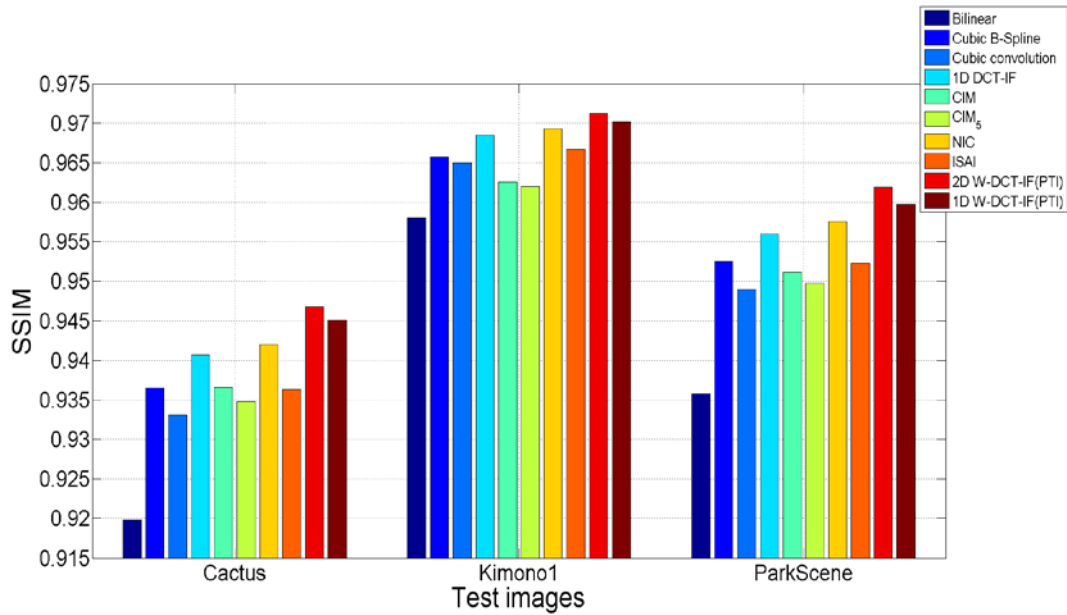
Fig. 5. SSIM comparison for 1.58 times zoom

5.5 Evaluation based on Image Sharpness Metric

To evaluate the sharpness of the resized images objectively, we consider the sharpness metric [23]. The sharpness metric based on the JNB is used to evaluate the amount of perceived sharpness without any reference image. Note that a higher sharpness metric value results in more edges in the image being preserved after up or down scaling. The sharpness values of the test images are listed in **Table 5**, where the scaling ratio is set to 2.3. As you see from **Table 5**, the sharpness values of the proposed methods are much higher than those of the various conventional schemes.

Table 5. SHARPNESS METRIC COMPARISON OF INTERPOLATION ALGORITHMS FOR 2.3 TIMES ZOOM

Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	4.547	3.237	3.388	13.199	11.014	4.927	11.533	9.412	4.300	11.521	7.708
Cubic B-Spline[2]	5.086	3.744	3.950	15.898	12.039	6.872	13.585	11.757	5.831	14.062	9.282
Cubic Convolution[3]	4.860	3.596	3.829	14.658	12.093	6.131	12.907	11.188	5.387	13.292	8.794
1-D DCT-IF[17]	4.809	3.674	3.922	16.061	13.687	6.853	13.819	12.135	5.888	14.061	9.491
CIM[8]	4.836	3.589	3.711	14.998	11.026	7.078	13.513	12.101	5.572	13.985	9.041
CIM$_5$ [8]	4.718	3.467	3.535	14.030	10.634	6.708	12.298	11.395	4.991	13.346	8.512
NIC [10]	**5.449**	4.126	4.202	15.778	13.016	7.758	14.321	13.382	6.925	15.087	10.004
ISAI [7]	5.270	3.764	4.003	15.246	12.674	6.904	13.721	11.974	5.993	14.272	9.3820
2D W-DCT-IF (PTI)	5.208	**4.211**	**4.246**	16.595	**14.581**	**8.242**	14.532	**14.010**	**7.137**	**15.353**	**10.412**
1D W-DCT-IF (PTI)	5.317	4.180	4.216	**16.798**	14.284	7.843	**14.584**	13.474	6.842	15.143	10.268

5.6 Evaluation based on Edge Map

In this subsection, we consider the edge maps of specific enlarged regions of a Cactus image that have been magnified with a scaling ratio of 4.0. The edge information of the enlarged images is found using the Canny edge detector [26], which is well known as a powerful tool for detecting edges. In **Fig. 6**, the edge map of image up-scaled by the W-DCT-IF method has more edge data than the others. This implies that the edge of an image scaled by the proposed algorithm is more significantly enhanced than those interpolated by the conventional methods.
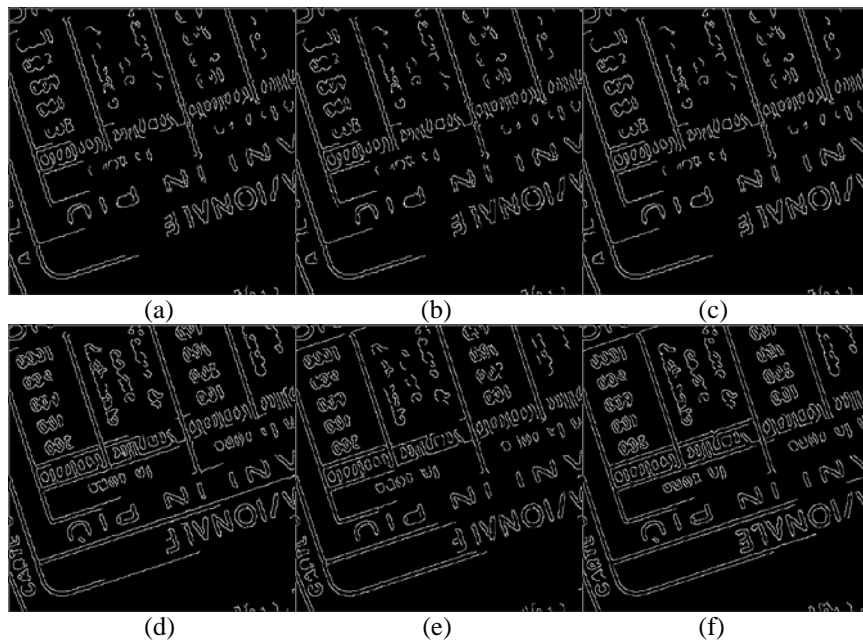

(a)   (b)   (c)

(d)   (e)   (f)

Fig. 6. Edge maps of the enlarged Cactus images when scaling ratio is 4.0 (a) Bilinear, (b) CIM, (c) CIM_5 , (d) NIC, (e) ISAI, (f) 2D W-DCT-IF (PTI)

5.7 Evaluation based on subjective quality

The subjective quality of the scaled image is evaluated in **Fig. 7**, where the Cactus image is magnified with a scaling ratio of 4.0. As can be observed from this figure, the images enlarged by the proposed algorithms provide a higher subjective quality than others. Because the edge provides important data in a subjective quality evaluation, the quality of the edge-enhanced image is perceptually higher.
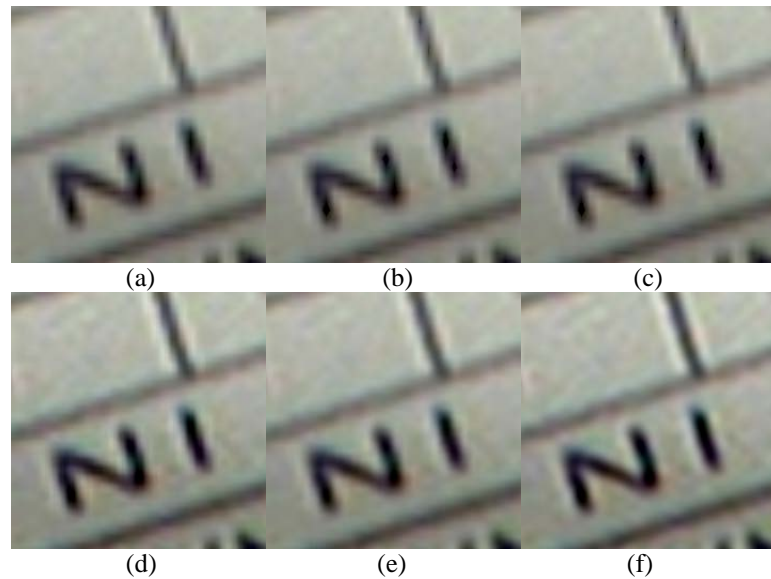
(a)   (b)   (c)

(d)   (e)   (f)

Fig. 7. Particular regions of the enlarged Cactus images when scaling ratio is 4.0 (a) Bilinear, (b) CIM, (c) CIM5 , (d) NIC, (e) ISAI, (f) 2D W-DCT-IF (PTI).

Table 6. COMPLEXITY COMPARISON OF INTERPOLATION ALGORITHMS FOR 2.3 TIMES ZOOM

Scheme	FOUR PEOPLE	Johnny	Kristen AndSara	Slide Editing	Slide Show	Basket ballDrive	BQ Terrace	Cactus	Kimon1	Park Scene	Avg.
Bilinear[1]	0.83	0.84	0.83	0.83	0.83	1.73	1.72	1.72	1.73	1.72	1.28
Cubic B-Spline[2]	7.91	7.92	7.91	7.92	7.92	13.78	13.80	13.78	13.78	13.80	10.85
Cubic Convolution[3]	3.86	3.85	3.86	3.85	3.85	7.77	7.99	7.99	7.77	7.99	5.88
1-D DCT-IF[17]	37.86	37.84	37.86	37.65	37.64	76.67	76.24	76.55	76.67	76.55	57.15
CIM[8]	7.54	7.55	7.33	7.45	7.65	20.30	20.33	20.33	20.30	20.33	13.91
CIM5 [8]	7.35	7.35	7.35	7.55	7.35	19.78	19.72	19.72	19.78	19.72	13.57
NIC [10]	3.02	3.01	3.02	3.01	3.03	6.07	6.07	6.08	6.07	6.08	4.55
ISAI [7]	3.07	3.05	3.07	3.05	3.05	6.07	6.07	6.08	6.07	6.08	4.57
2D W-DCT-IF (PTI)	1824.91	1822.90	1824.51	1822.33	1821.70	3741.44	3752.07	3754.07	3740.33	3754.07	2785.83
1D W-DCT-IF (PTI)	37.96	37.94	37.96	37.75	37.75	76.76	76.66	76.66	76.76	76.67	57.29

To provide simulation results for the perceptual evaluation, the Single Stimulus Continuous Quality Evaluation (SSCQE) method [27] is chosen because the resolutions of scaled images are different from those of the original images. The Mean Opinion Score (MOS) is a simple measurement to obtain the viewers' opinions. The MOS provides a numerical indication of the quality perceived by viewers. The MOS is expressed as a single number in a range of 1–5, where 1 means bad and 5 means excellent. In this test, the enlarged Cactus, Kimono1, and ParkScene images with a scaling ratio of 2.3 were evaluated and scored by 15 viewers who were college students. **Fig. 8** shows the MOS values. The MOS values of the proposed methods are always much higher than those of the conventional methods [1]-[3], [7], [8], [10] ,[17].
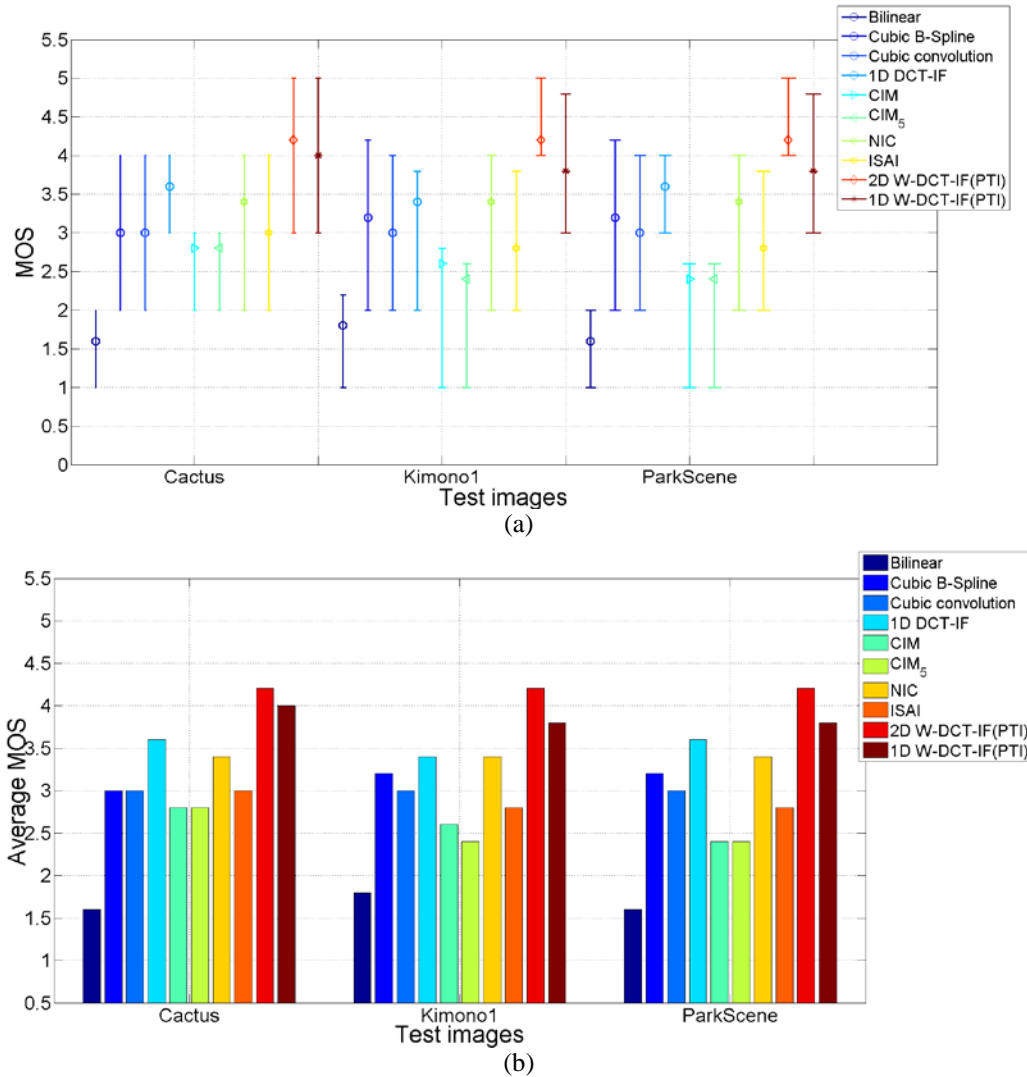
Fig. 8. Mean opinion score (MOS) values of three enlarged images with scaling ratio of 2.3. (a) Ranges of MOS values for the three enlarged images and (b) average MOS values for the three enlarged images.

5.8 Evaluation based on Complexity

Complexity is a crucial measurement because a scaler module is used in real-time applications. In this simulation, the complexity is measured by the consumed CPU time, where the algorithms are tested on a personal computer with a 3.4GHz 6-Core Processor and main memory of 8 GB.

The consumed CPU times are represented in **Table 6,** where the numbers for PTI don't include the training complexity. In this table, the complexity of the 2D W-DCT-IF (PTI) is the highest among those of all methods, because 2D W-DCT-IF (PTI) implemented by (9) needs the complex iterative process. Although 2D W-DCT-IF (PTI) is much more complex than other conventional schemes [1]-[3], [7], [8], [10] ,[17], the scheme produces a gain from the viewpoint of enhancing the subjective and objective quality. In contrast to 2D W-DCT-IF (PTI), the computational complexity of 1D W-DCT-IF (PTI) of (22) and (23) is almost equal

to that of the conventional 1D DCT-IF [17]. Note that the weighting parameters are determined before the scaling process is applied. Thus, the complexity of the optimization process for PTI is not included in these results. In order to reduce the complexity of the scaling system, 1D W-DCT-IF (PTI) can be employed instead of 2D W-DCT-IF (PTI).

5. Conclusions

In this paper, we designed the W-DCT-IF to enhance the edge information of a resized image. Because DCT-IF is known to be an outstanding method for resizing images, the optimization of DCT-IF is an important issue in image and video processing systems. Although the proposed 2D W-DCT-IF is more complex than other conventional scaling algorithms, the quality of the resized image produced by the proposed 2D W-DCT-IF is significantly improved. To reduce the complexity of the proposed method, the 2D W-DCT-IF can be implemented using 1D W-DCT-IF. While the performance of 1D W-DCT-IF is similar to that of 2D W-DCT-IF, the complexity of 1D W-DCT-IF is approximately equal to that of the conventional 1D DCT-IF. By employing the proposed algorithm, we can construct efficient scaling system without increasing the complexity.

References

[1] E. Meijering, "A chronology of interpolation: from ancient astronomy to modern signal and image processing," *Proceedings of IEEE*, vol. 90, no. 3, pp. 319–342, Mar. 2002. Article (CrossRef Link)

[2] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. I. Theory," *IEEE Trans. on Signal Processing*, vol. 41, no. 2, pp. 821–833, Feb. 1993. Article (CrossRef Link)

[3] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981. Article (CrossRef Link)

[4] H. Demirel and G. Anbarjafari, "Discrete Wavelet transform-based satellite image resolution enhancement," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 1997–2004, Jan. 2011. Article (CrossRef Link)

[5] K. McCann, W.-J. Han, and I.-K. Kim, "Samsung's response to the call for proposals on video compression technology," *JCTVC-A124*, Dresden, Germany, Apr. 2010.

[6] D. Zhou, X. Shen, and W. Dong "Image zooming using directional cubic convolution interpolation," *IET Image Processing*, vol. 6, no. 6, pp. 627–634, Aug. 2012. Article (CrossRef Link)

[7] L. Rui and L. Qiong, "Image sharpening algorithm based on a variety of interpolation methods," *in Proc. of IEEE Int. Conf. Image Analysis and Signal Processing (IASP)*, pp. 1–4, Nov. 2012. Article (CrossRef Link)

[8] H. Kim, Y.-J. Cha, and S.-J Kim, "Curvature interpolation method for image zooming," *IEEE Trans. on Image Processing*, vol. 20, no. 7, pp. 1895–1903, Jan. 2011. Article (CrossRef Link)

[9] H. Demirel, and G. Anbarjafari, "IMAGE Resolution Enhancement by Using Discrete and Stationary Wavelet Decomposition," *IEEE Trans. on Image Processing*, vol. 20, no. 5, pp. 1458-1460, May 2011. Article (CrossRef Link)

[10] C.-C Huang, P.-Y. Chen, and C.-H Ma, "A novel interpolation chip for real-time multimedia applications," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 22, no. 10, pp. 1512–1525, Oct. 2012. Article (CrossRef Link)

[11] F. Cao, M. Cai, and Y. Tan, "Image Interpolation via Low-Rank Matrix Completion and Recovery," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 25, no. 8, pp. 1261-1270, Aug. 2015. Article (CrossRef Link)

[12] J. –K. Chae, J. –Y. Lee, M. –H. Lee, J. –K. Han, T. Q. Nguyen, and W. –Y. Yeo, "Cubic Convolution Scaler Optimized for Local Property of Image Data," *IEEE Trans. on Image Processing*, vol. 24, no. 12, pp. 4796-4809, Dec. 2015. Article (CrossRef Link)

[13] S. Zhu, B. Zeng, L. Zeng, and M. Gabbouj, "Image Interpolation Based on Non-local Geometric Similarities and Directional Gradients," *IEEE Trans. on Multimedia*, vol. 18, no. 9, pp. 1707-1719, Sept. 2016. Article (CrossRef Link)

[14] S.-Y Yang, Y.-H Kim, and J.-C Jeong, "Fine edge-preserving technique for display devices," *IEEE Trans. Consumer Electronics*, vol. 54, no. 4, pp.1761–1769, Nov. 2008. Article (CrossRef Link)

[15] J.-W. Han, J.-H. Kim, S.-H. Cheon, and J.-O. Kim, "A Novel Image Interpolation Method Using the Bilateral Filter," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 1, pp. 175–181, Feb. 2010. Article (CrossRef Link)

[16] I.-H. Shin and H.-W. Park, "Adaptive up-sampling method using DCT for spatial scalability of scalable video coding," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 19, no. 2, pp. 206–214, Feb. 2009. Article (CrossRef Link)

[17] E. Alshina, J. Chen, A. Alshin, N. Shlyakhov, and W. J. Han, "CE3: experimental results of DCTIF by Samsung," *JCTVC-D344*, Daegu, Korea, Jan. 2011.

[18] S. Matsuo, S. Takamura, and H. Jozawa, "Improved intra angular prediction by DCT-based interpolation filter," *in Signal Processing Conference (EUSIPCO)*, pp. 1568–1572, Aug. 2012.

[19] H. Lv, R. Wang, X. Xie, H. Jia, and W. Gao, "A comparison of fractional-pel interpolation filters in HEVC and H.264/AVC," *in IEEE Int. Conf. Visual Communications and Image Processing (VCIP)*, pp. 1–6, Nov. 2012. Article (CrossRef Link)

[20] E. Alshina, J.-H Park, and W.-J. Han, "DCT based interpolation filter for motion compensation in HEVC," *in Proceedings of Applications of Digital Image Processing XXXV*, vol. 8499, pp. 13, Oct. 2012. Article (CrossRef Link)

[21] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd edition, Prentice-Hall, 2008, pp. 165–168.

[22] R. Crane, *A Simplified Approach to Image Processing in C*, Prentice-Hall, 1997, pp. 79–93.

[23] R. Ferzli and L. J. Karam, "A no-reference objective image sharpness metric based on the notion of just noticeable blur (JNB)," *IEEE Trans. on Image Processing*, vol. 18, no. 4, pp. 717–728, Apr. 2009. Article (CrossRef Link)

[24] J. A. Garcia, Rosa Rodriguez-Sanchez, and J. Fdez-Valdivia, "The emergence of a region-based approach to image transmission," *Optical Engineering*, vol. 44, no. 6, June 2005. Article (CrossRef Link)

[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Trans. on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004. Article (CrossRef Link)

[26] J. Canny, "A computational approach to edge detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 670–698, Nov. 1986. Article (CrossRef Link)

[27] *Rec. ITU-R BT.500-13*, "Methodology for the Subjective Assessment of the Quality of Television Pictures," Jan. 2012.

Jae-Yung Lee was born in Ulsan, Korea, on February 20, 1986. He received the B.S, M.S and Ph.D. degrees in the Department of Information and Communication engineering, Sejong University, Seoul, Korea in 2011, 2013 and 2018, respectively. He was involved in the development of High Efficiency Video Coding (HEVC) standard and the MV-HEVC. In 2015, he was a visiting scholar with the Department of Electronics and Computer Science, University of California, San Diego, USA. His research interests include video coding and high efficiency video coding.

Sung-Jun Yoon was born in Seoul, Korea, on January 3, 1990. He received the B.S and M.S degrees from the Department of Information and Communication engineering, Sejong University, Seoul, Korea in 2016 and from the Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Taejon, Korea in 2018, respectively. Currently, he is working at SK Telecom. His research interests include image processing and artificial intelligence.

Jae-Gon Kim received his BS in electronics engineering from Kyungpook National University, Daegu, Rep. of Korea, in 1990, and his MS and PhD in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Rep. of Korea, in 1992 and 2005, respectively. From 1992 to 2007, he was with ETRI, Daejeon, Rep. of Korea, where he was involved in the development of digital broadcasting media services, MPEG-4/7/21 standards, and related media technologies. He is currently an associated professor in the School of Electronics and Information Engineering at Korea Aerospace University, Goyang, Gyeonggi-do, Rep. of Korea. His research interests include video coding, video signal processing, and multimedia applications. He has participating in the standardization of video coding (HEVC, IVC).

Jong-Ki Han was born in Seoul, Korea, in September 1968. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejon, Korea, in 1992, 1994, and 1999, respectively. From 1999 to 2001, he was a Member of Technical Staff with the Corporate R & D Center, Samsung Electronics Company, Suwon, South Korea. He is currently a Professor with the Department of Electrical Engineering, Sejong University, Seoul, Korea. His research interests include image and video coding, transcoding, and signal processing for video sequence. He has participated in standardization of high efficiency video coding (HEVC) since 2010.