

## ORIGINAL ARTICLE

# Rate-dependent seam carving and its application to content-aware image coding

YUICHI TANAKA<sup>1,2</sup>, TAICHI YOSHIDA<sup>3</sup>, MADOKA HASEGAWA<sup>1</sup>, SHIGEO KATO<sup>1</sup> AND MASAOKI IKEHARA<sup>3</sup>

*Content-aware image resizing (CAIR) is desired because it preserves prominent regions in a resized image. However, CAIR requires high computational complexity to perform in mobile devices, though it is desired for these displays. Moreover, transmitting the side information for CAIR from the encoder is a problem since it usually requires high bitrates compared with those for image data. In this paper, we present a rate-dependent CAIR method that produce various retargeting results based on the bitrates for side information. Furthermore, we apply the proposed technique to wavelet-based image coding. Our proposed content-aware image coding method provides good performances for both CAIR and image coding.*

**Keywords:** Content-aware image coding, Content-aware image resizing, Image retargeting, Image coding, Wavelet, SPIHT

Received 13 March 2012; Revised 15 January 2013

## 1. INTRODUCTION

Recently, content-aware image resizing (CAIR) techniques, also known as image retargeting, have been attracting the attention of computer vision researchers [1–6]. These techniques are regarded as sophisticated image resizing methods. To resize an image into a different aspect ratio and/or one with complex structures, CAIR gives better results than traditional scaling and cropping. In image processing terms, CAIR is a non-uniform resampling of images for extracting region-of-interest (ROI) automatically. However, it generally requires high computational complexity to analyze local and/or global image structures such as objects, backgrounds, and saliency. Therefore, CAIR is a very challenging task for the built-in software in portable devices, such as cell phones and PDAs.

Seam carving (SC) [1, 2] is one of the pioneering techniques for CAIR. It determines a *seam*, defined as an eight-connected path of pixels from the top (left) to the bottom (right) of the original image shown in Fig. 1(b). The width (height) of the image is reduced by one pixel after pruning a seam. The final retargeted image is produced with iterative pruning of seams until the retargeted image reaches the desired width (height). An example of the retargeted image is shown in Fig. 1(d). Combinations of SC and other resizing

methods (scaling and/or cropping) have been reported in [7, 8].

There are a few approaches of content-aware image coding (CAIC) with SC-like methods [9–11]. Generally, they first perform CAIR for the original image and then the retargeted image<sup>1</sup> is compressed by some image coding standards. The side information, which would consist of seam path information (SPI) and pixel values of seams, will be stored and/or sent with the compressed image data. At the decoder side, the compressed core image is first decoded. If one needs the reconstructed image with arbitrary resolutions, the side information is used to reconstruct regions outside the core image (non-ROI).

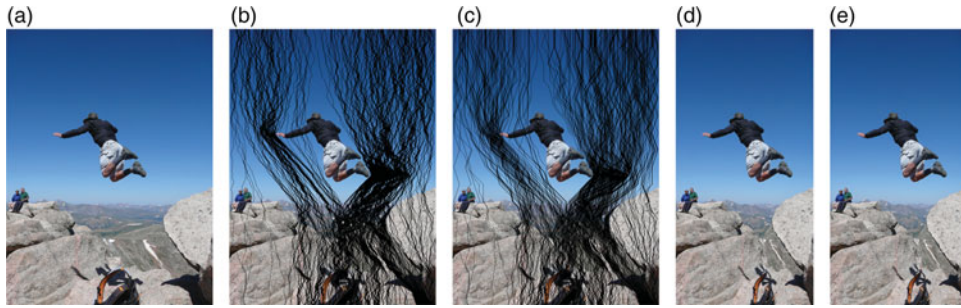
For these methods, the amount of SPI significantly affects the entire bitrates. Although the conventional method [9] uses arithmetic coding to encode SPI, it usually requires high bitrates. The methods in [10, 11] try to reduce the bitrates to employ “pillar”-style seams. Unfortunately, the structures of their seam paths are restricted.

In this paper, we present a novel rate-dependent SC (RD-SC). Intuitively, seam paths by RD-SC are the *lossy* version of the original seam paths and it is realized by a seam path approximation with piecewise predefined functions (Fig. 1(c)) while maintaining CAIR performance (Fig. 1(e)). Furthermore, we integrate RD-SC into image coding to realize an effective CAIC. A new parent–children relationship for a well-known wavelet-based image coding algorithm, set partitioning in hierarchical trees (SPIHT) [12], is proposed. In the experimental results, its image coding performances at full resolution are comparable to those of the regular

<sup>1</sup>Graduate School of Engineering, Utsunomiya University, Utsunomiya, Tochigi 321-8585, Japan

<sup>2</sup>Graduate School of BASE, Tokyo University of Agriculture and Technology, Koganei, Tokyo 184-8588, Japan

<sup>3</sup>Department of Electronics and Electrical Engineering, Keio University, Yokohama, Kanagawa 223-8522, Japan



**Fig. 1.** Visual comparisons of the CAIR methods (*Mt. Evans* image). Black colored regions represent pixels to be pruned. The original size is  $1024 \times 768$  pixels and it is resized to  $1024 \times 512$ . From left to right: (a) original image, (b) seam paths calculated by SC, (c) seam paths calculated by RD-SC, (d) CAIR result by SC and (e) CAIR result by RD-SC.

**Table 1.** Notation with Respect to an Image  $I$ .

Symbol	Description
$I(i, j)$	Pixel value at $i$ -th row and $j$ -th column
$I(i, :)$	$i$ -th row
$I(:, j)$	$j$ -th column
$I(L_0 : L_1, :)$	Sub-image where $L_0 \leq i \leq L_1$
$H_I$	Height
$W_I$	Width

SPIHT while preserving good CAIR qualities for retargeted images.

This paper is organized as follows: Section II gives a brief reviews of related methods of RD-SC. The following two sections present the main contributions of this paper. Section III discusses RD-SC using a piecewise approximation. A wavelet-based CAIC with the modified SPIHT algorithm is presented in Section IV. The experimental results are shown in Section V, and finally, Section VI summarizes the paper.

## A) Notation

Table 1 lists the symbols used in this paper for an image  $I$ . To simplify our description, we use the MATLAB-based notation for the sets of rows or columns in a matrix. Moreover, for simplicity, the  $i$ th element of a vector  $\mathbf{x}$  is represented as  $x(i)$  and a subvector of  $\mathbf{x}$  between  $L_0 \leq i \leq L_1$  is defined as  $\mathbf{x}(L_0 : L_1)$ .

In addition, this paper considers a vertical seam for the discussion about RD-SC unless specified. A vertical seam is defined as an eight-connected path of pixels of one-pixel width from the top to the bottom of an image. The process can be easily extended to a horizontal seam.

## II. REVIEW

### A) Seam carving

SC [1, 2] prunes seams classified as insignificant in an image. In the original SC technique, a dynamic programming approach is used to decide seams which can be pruned.

A cumulative cost  $M(i, j)$  for each seam is calculated by the following equation:

$$M(i, j) = \min \begin{pmatrix} M(i-1, j-1) + C_L(i, j) \\ M(i-1, j) + C_U(i, j) \\ M(i-1, j+1) + C_R(i, j) \end{pmatrix}, \quad (1)$$

where

$$\begin{aligned} C_L(i, j) &= |I(i, j+1) - I(i, j-1)| \\ &\quad + |I(i-1, j) - I(i, j-1)|, \\ C_U(i, j) &= |I(i, j+1) - I(i, j-1)|, \\ C_R(i, j) &= |I(i, j+1) - I(i, j-1)| \\ &\quad + |I(i-1, j) - I(i, j+1)|. \end{aligned}$$

The cost function is called *forward energy* [2]. Finally, the seam with the smallest  $M(H_I, j)$  is pruned. With this criterion, a retargeted image is forced to avoid the large transition of pixel values between the left and right pixels of the pruned seam. The process is iterated until the desired image width is reached. For other details, please refer to [1, 2].

### B) CAIC

So far, several methods were reported in CAIC [9–11, 13–15]. In this subsection, we briefly review the related works.

Image coding with selective data pruning (SDP) [16] is classified as one of the “decimation-then-compression” image coding approach. The SDP method has a very simple architecture for reducing an image size since it is based on line-based downsampling. That is, rows and/or columns in an image are simply pruned and then the pruned image is interpolated on the synthesis side. The pruned lines are located predominantly across low-frequency regions in the image and thus the interpolated image has few artifacts if a small number of lines are pruned. In the comparison of image coding without data pruning, SDP-based image compression improves the peak signal-to-noise ratio (PSNR) at low bitrates. However, when many lines are pruned, the pruned lines most likely cut across salient regions. As a result, the interpolated image could have artifacts. A similar method to SDP was also presented in [17].

Incorporating SC into image coding was investigated by Nguyen *et al.* [9]. In their method, SPI and the pixel values of seams as well as the core image are encoded.

The method can reconstruct the original image losslessly if the core image, the SPI and the pixel values of seams are losslessly encoded and transmitted. However, it demands a large amount of bitrates for SPI, and the overhead bitrate significantly affects the entire image coding performance.

The authors proposed an approach for CAIC called image concentration and dilution [10, 11], denoted as CD-based image coding hereafter. In the method, first, insignificant regions in the original image are discarded by the SC-like method and then the core image is compressed by some image coding standard. At the same time, the discarded pixel positions are stored. On the decoder side, the core image and the side information are synthesized to reconstruct the image with an arbitrary resolution. The discarded pixels, i.e., non-ROI pixels, are interpolated from the decoded core image. Unfortunately, reconstructed images after CD-based image coding sometimes have the interpolation artifacts at non-ROI.

The SC-based content-aware video coding was proposed in [13]. It reduces side information bitrate, i.e., for SPI, by employing key points of seam paths and a seam merging process. In low bitrate video coding, the method succeeded to reduce total bitrates for the equivalent video quality. However, it is also non-perfect-reconstruction video coding as well as SDP- and CD-based image coding. That is, seams should be interpolated from reconstructed core video frames.

Frequency-domain SC and its image coding application were reported in [14, 15]. It calculates optimal seam paths on the LL subband of DWT. The SPI is then used to determine the scanning order of SPIHT. The seams are scanned and transmitted in the energy descending order (high-energy seams are firstly encoded). In full resolution, seams in the LL subband result block-based seams and therefore the decomposition level should not be too large (practically, the decomposition level will be less than or equal to 3 [15]). Therefore, its image coding performance at full resolution is substantially inferior to the original SPIHT.<sup>2</sup>

Briefly speaking, the following two main problems should be resolved for CAIC:

- (1) Reducing side information for CAIC while preserving CAIC performance.
- (2) Maintaining image coding performance at full resolution.

For both encoding and decoding, the side information, especially SPI, is usually required to obtain a retargeted image with an arbitrary resolution. However, the original SC used in [9] is not designed to encode SPI effectively. Therefore, the amount of side information is quite large if we do not consider designing a SPI encoding method.

Another problem is the image coding performance. Although the core image can be compressed effectively by any existing standards, non-ROI is difficult to compress with a good quality since the positions of those pixels are not structured and highly depend on images.

<sup>2</sup>In [15], 1.8 dB average PSNR degradation was reported.

In the following sections, we introduce the RD-SC and the modified tree structure of SPIHT, which try to resolve these problems.

### III. RD-SC WITH PIECEWISE APPROXIMATION

In this section, we present details of RD-SC. The seam paths of RD-SC are constructed from those of the original SC based on a piecewise approximation. The RD-SC structure introduced in this section is illustrated in Fig. 2 and it is briefly summarized as follows:

- (1) Calculate a seam by the original SC. If there exists already pruned seams, a modified cost map calculated from previous pruned seam paths is used (Section III-F).
- (2) Approximate the seam with the consideration of a rate-distortion performance (Section III-A–III-E).
- (3) If possible, prune multiple seams (Section III-G).
- (4) Store SPI and return to (1) for the next seam approximation.

The rest of this section describes detailed algorithms.

#### A) Piecewise function

Let  $\mathbf{s}$  be a seam path. In general, it can be defined as follows:

$$\mathbf{s} = \{j \in \mathbb{N} : j = f(i), 0 \leq i \leq H_I - 1\}, \quad (2)$$

where  $f(i)$  is a function to determine a seam path. A specific coordinate of the seam path is represented as  $(i, \mathbf{s}(i))$ . According to the purpose of SC [1, 2], the following conditions are usually imposed on  $f(i)$ :

- *Connectivity*:  $|f(i) - f(i + 1)| \leq 1$ .
- *Rectangular restriction*:  $0 \leq f(i) \leq W_I - 1$ .

The first condition is the requirement that a seam path should be a set of eight-connected pixels. The second one is imposed to preserve a rectangular shape of the resized image.

Usually  $\mathbf{s}$  has a complex structure since SC calculates  $\mathbf{s}$  pixel-by-pixel and thus, it is difficult to compress efficiently. In this paper, we propose an approximation method of  $\mathbf{s}$  with piecewise predefined functions. We define an approximated seam path  $\mathbf{a}$  as follows:

$$\mathbf{a}_k = \{j \in \mathbb{N} : j = f_k(i_k), 0 \leq i_k \leq L(k) - 1\}, \quad (3)$$

where  $\mathbf{a}_k$  ( $k = 0, 1, \dots$ ) is the  $k$ th portion of  $\mathbf{a}$ ,  $L(k)$  defines the interval for  $\mathbf{a}_k$  and  $f_k(i_k)$  is an approximation function.  $\mathbf{a}_k$  is calculated within the interval  $\left[ \sum_{p=0}^{k-1} L(p), \sum_{p=0}^k L(p) - 1 \right]$ . The relationship between  $\mathbf{s}$  and  $\mathbf{a}$  is illustrated in Fig. 3.

#### B) Candidates of $f_k(i_k)$

Any functions can be candidates of  $f_k(i_k)$  as long as they follow the conditions mentioned in the previous subsection.

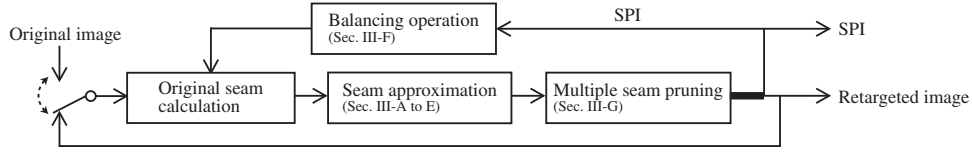
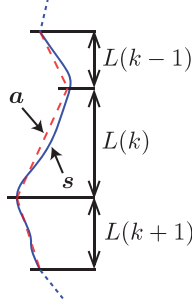


Fig. 2. Structure of RD-SC.

Fig. 3. Relationship between  $s$  and  $a$ . Blue solid and red dashed lines represent  $s$  and  $a$ , respectively.

We consider to reduce the bitrate for SPI by choosing one function from several templates. A template set of  $f_k(i_k)$ , shown as  $f_{k,g}(i_k)$ , is heuristically defined as follows:

$$f_{k,0}(i_k) = C_k, \quad (4)$$

$$f_{k,\pm 1}(i_k) = \pm i_k + C_k, \quad (5)$$

$$f_{k,\pm 2}(i_k) = \pm \left\lceil \frac{1}{3} i_k \right\rceil + C_k, \quad (6)$$

$$f_{k,\pm 3}(i_k) = \pm \left\lceil \frac{i_k^2}{2L(k)} \right\rceil + C_k, \quad (7)$$

$$f_{k,\pm 4}(i_k) = \pm \min \left( \left\lceil \frac{1}{2} \sqrt{L(k) \cdot i_k} \right\rceil, i_k \right) + C_k, \quad (8)$$

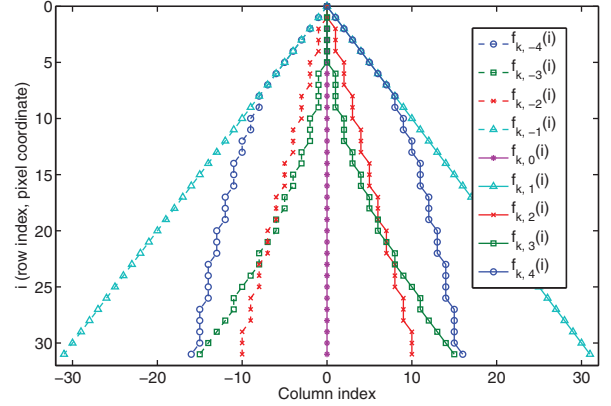
where  $C_k = a_k(0)$  and  $\lceil \cdot \rceil$  is the rounding operator.  $f_{k,g}(i_k)$  are shown in Fig. 4. Clearly, (4) is the constant function. The upper and lower bounds of seam paths from the connectivity condition are represented in (5). In (8), the minimum operator is used since the square-root functions violate the connectivity condition without the operator.

We append one extra condition as follows:

$$C_k = a_k(0) = \begin{cases} s_o(0) & \text{if } k = 0, \\ a_{k-1}(L(k-1) - 1) & \text{otherwise,} \end{cases} \quad (9)$$

where  $s_o$  is the original seam path to be approximated. This means the initial position of  $a_k$  is the same as the terminated position of  $a_{k-1}$ . With this restriction, we can reduce the connection information between  $a_{k-1}$  and  $a_k$ .

As a result, the SPI for  $a_k$  consists of a selected index among (5–8) and the interval  $L(k)$ . Furthermore, the overall initial position  $s_o(0) = a(0)$  is required for each  $a$ .

Fig. 4.  $f_{k,g}(i_k)$  where  $L(k) = 32$ . The number in legend is corresponding to those in (4–8).

### C) Recursive optimization of approximated seams

The optimal  $L(k)$  should not be determined as a constant. For some  $k$ ,  $L(k)$  can be large if  $s_o$  goes through very smooth regions or it is well approximated by one of the templates (4–8). Whereas  $L(k)$  should be small at the textured region where  $s_o$  has a complex structure. Hence, we present the algorithm for recursive optimization of  $L(k)$ .

We assume  $s_o$  is calculated before the approximation. To estimate  $L(k)$  effectively, we define the simplified forward energy criterion as follows:

$$C_{SFE}(s) = \frac{1}{s_L} \sum_i |I(i, s(i) - 1) - I(i, s(i) + 1)|, \quad (10)$$

where  $s_L$  is the length of  $s$ . This criterion is similar to the forward energy of the improved SC [2] but nevertheless is less complex. In the optimization process, we use (10) since many repetitive calculations are required.

Algorithm 1 presents the detailed optimization process of  $L(k)$  for the  $k$ th interval of  $I$ :

$$I_{sub} = I \left( \sum_{p=0}^{k-1} L(p) : \sum_{p=0}^k L(p) - 1, : \right).$$

In Algorithm 1,  $s_{o,sub}$  is a original sub-seam path for  $I_{sub}$ ,  $T > 0$  is a threshold and  $a_{sub}$  is the approximated seam path for  $I_{sub}$ . Moreover,

$$a_{k,g} = \{j : j = f_{k,g}(i_k), 0 \leq i_k \leq L(k) - 1\}.$$

It is clear that the approximated seam path  $a_{k,g}$  can be completely recovered from the returned parameters. The



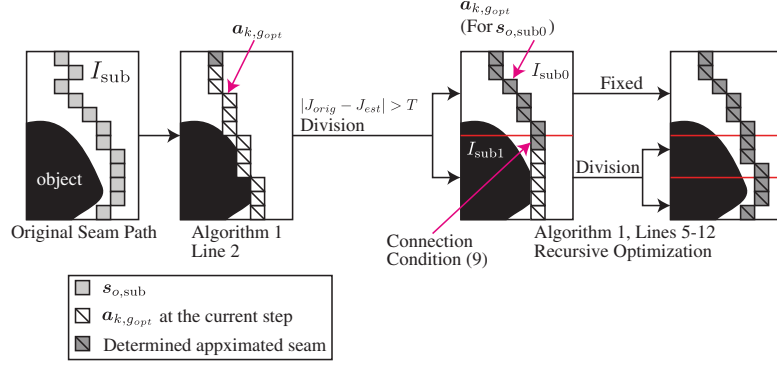


Fig. 5. Recursive seam approximation process. The red lines represent the division points of  $s_{o,sub}$ .

---

### Algorithm 1 Optimization of $L(k)$ with piecewise functions

---

Require:  $C_k$

```

1: function OPTHEIGHT( $I_{sub}, s_{o,sub}, T$ )
2:    $g_{opt} \leftarrow \arg \min_g (\|s_{o,sub} - a_{k,g}\|_2)$  % Selecting one of candidates from (5-8).
3:    $J_{orig} \leftarrow C_{SFE}(s_{o,sub})$ 
4:    $J_{est} \leftarrow C_{SFE}(a_{k,g_{opt}})$ 
5:   if  $|J_{orig} - J_{est}| > T$  and  $L(k) > 4$  then % Recursive optimization.
6:      $I_{sub0} \leftarrow I_{sub}(0 : [L(k)/2] - 1, :)$  % Dividing  $I_{sub}$  into two halves.
7:      $I_{sub1} \leftarrow I_{sub}([L(k)/2] : L(k) - 1, :)$ 
8:      $s_{o,sub0} \leftarrow s_{o,sub}(0 : [L(k)/2] - 1, T)$  % Dividing  $s_{o,sub}$  into two halves.
9:      $s_{o,sub1} \leftarrow s_{o,sub}([L(k)/2] : L(k) - 1, T)$ 
10:    OPTHEIGHT( $I_{sub0}, s_{o,sub0}, T$ )
11:    OPTHEIGHT( $I_{sub1}, s_{o,sub1}, T$ )
12:  else
13:    return  $L(k), g_{opt}$ 
14:  end if
15: end function

```

---

simplified structure of this algorithm is shown in Fig. 5. In applications shown in this paper, the maximum row height of  $I_{sub}$  (denoted as  $L_{max}$ ) is limited to 32. As a result, each  $L(k)$  is selected from  $\{32, 16, 8, 4\}$ .

### D) Iterative optimization

In Algorithm 1, the optimal  $a_k$  associated with  $L(k)$  is obtained for a given threshold  $T$ . As a further extension, the optimal threshold  $T$  is sought to control the bitrate for the SPI and the CAIR quality. The optimization is realized by an iterative optimization of Algorithm 1. The object of RD-SC is to find the nearest seam  $a_{opt}$  to  $s_o$  under a bitrate constraint for the SPI, which is represented as follows:

$$a_{opt} = \arg \min_a (D_{PA}(a) + \lambda R_{PA}(a)). \quad (11)$$

where  $\lambda \geq 0$  is the Lagrange multiplier. In this approach, we define  $D_{PA}(a)$ , shown below, based on the reconstruction error by the simple interpolation.

$$D_{PA}(a) = \text{MSE}(I(:, s_o), \bar{I}(:, s_o)) - \text{MSE}(I(:, a), \bar{I}(:, a)), \quad (12)$$

where MSE is a mean squared error between the original and interpolated seams and  $\bar{I}$  is defined by a linear

interpolation from six available neighbors as

$$\bar{I}(i, v(i)) = \frac{1}{6} \sum_{u_0=-1}^1 \sum_{u_1 \in \{-1,1\}} I(i + u_0, v(i) + u_1), \quad (13)$$

which is depicted in Fig. 6. The calculation method of  $R_{PA}(a)$  is shown in the following subsection (Section III-E).

Finally,  $a_{opt}$  in (11) is sought by gradually changing  $T$  shown in Algorithm 2, where  $a_{N_{cur}}$  is the approximated seam path at the current iteration and  $T_{PA}$  contains  $N_T$  threshold values and they are monotonically increasing. In this paper, we use  $T_{PA} = [1, 5, 10, \dots, 50]$ .

It is a similar strategy to the rate-distortion optimization for video coding [18]. Instead of the brute-force optimization, our approach is *climbing up* the R-D slope as shown in Fig. 7 to reduce computational complexity.

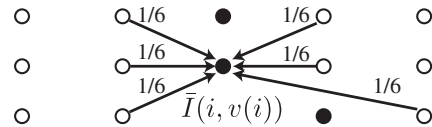


Fig. 6. Illustrative example of (13). White and black dots represent core (remaining) and seam pixels, respectively.

**Algorithm 2** Iterative optimization of  $\mathbf{a}$ 


---

**Require:**  $I_{sub}, \mathbf{s}_{o,sub}, C_k$

- 1: **function** ITEROPTSEAM( $\lambda, T_{PA}$ )
- 2:  $J_{prev} \leftarrow \infty$
- 3:  $J_{cur} \leftarrow D_{PA}(\mathbf{s}_{o,sub}) + \lambda R_{PA}(\mathbf{s}_{o,sub})$
- 4:  $N_{cur} \leftarrow 0$
- 5: % Climbing up the slope at the following loop.
- 6: **while**  $J_{cur} < J_{prev} \wedge N_{cur} < N_T$  **do**
- 7:  $T \leftarrow T_{PA}(N_{cur})$
- 8: OPTHEIGHT( $I_{sub}, \mathbf{s}_{o,sub}, T$ )
- 9:  $J_{prev} \leftarrow J_{cur}$
- 10:  $J_{cur} \leftarrow D_{PA}(\mathbf{a}_{N_{cur}}) + \lambda R_{PA}(\mathbf{a}_{N_{cur}})$
- 11:  $N_{cur} \leftarrow N_{cur} + 1$
- 12: **end while**
- 13: **end function**

---

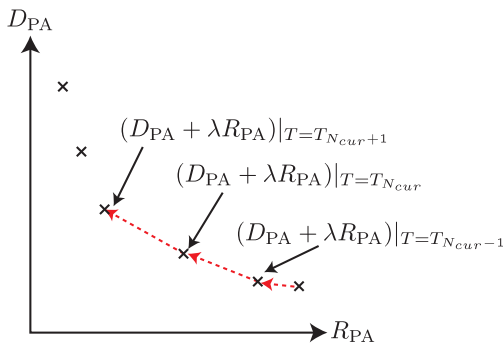


Fig. 7. Illustrative example of Algorithm 2.

**E) Bitrate calculation**

In Algorithm 2,  $R_{PA}(\mathbf{a})$  should be properly calculated from the approximated seam  $\mathbf{a}$ . As previously mentioned, the function index  $g_{opt}$  in Algorithm 1 and the set of  $L(k)$  are the side information for SPI. Since Algorithm 1 is based on the division of  $I_{sub}$  into two halves, we can represent a set of  $L(k)$  with a simple binary representation. That is, 0 is stored if  $I_{sub}$  is not divided, whereas 1 is stored if  $I_{sub}$  is divided. As a result, a binary vector  $\mathbf{b}_{div}$  can be obtained for the division information, which corresponds to  $L(k)$ .

By our preliminary experiment, codewords for function indices  $g_{opt}$  are determined as shown in Table 2. The required bitrate for the approximation functions is calculated based on the assigned codewords.

Finally,  $R_{PA}(\mathbf{a})$  can be represented as follows:

$$R_{PA}(\mathbf{a}) = \#(\mathbf{b}_{div}) + \sum b_{g_{opt}}, \quad (14)$$

where  $\#(\mathbf{b}_{div})$  is the number of elements in  $\mathbf{b}_{div}$  and  $b_{g_{opt}}$  is the number of bits to encode  $g_{opt}$ .

**F) Weighted cost matrix**

Each rate-dependent seam path can be calculated by RD-SC mentioned previously. However, a reconstructed image

**Table 2.** Codeword Assignment for  $f_{k,g}(i)$  of RD-SC

$g_{opt}$	Codeword
0	00
2	11
-2	101
4	010
-4	011
-1	10000
1	10001
-3	10010
3	10011

sometimes presents visible artifacts due to seam pruning from almost the same spatial positions. To alleviate this problem, we consider *balancing* the positions of the seams. For this purpose, we modify the cost of the original SC  $M(i, j)$  in (1). When an already pruned seam is defined as  $\hat{\mathbf{a}}_m$  ( $m = 0, 1, \dots$ ), a new cost map is represented as follows:

$$M_b(i, a(i)) = w \sum_m \delta_{\hat{\mathbf{a}}_m(i), a(i)}, \quad (15)$$

$$\hat{M}(i, j) = M(i, j) + M_b(i, j), \quad (16)$$

where  $w$  and  $\delta_{i,j}$  are the weight and the Kronecker delta, respectively. Briefly speaking, the extra cost map  $M_b(i, j)$  appends  $w$  for the conventional cumulative cost  $M(i, j)$  when a seam candidate crosses (or touches) a pruned seam path. We use  $\hat{M}(i, j)$  instead of  $M(i, j)$  in (1) for calculating a seam path. As a result, the updated cost  $\hat{M}(i, j)$  avoids pruning a seam adjacent to already pruned seams.

Furthermore,  $\hat{M}(i, j)$  is utilized to determine the maximum number of seams. If  $\min(\hat{M}(H_I, j)) > \tau$ , where  $\tau$  is an arbitrary threshold, we stop pruning seams since the remaining retargeted image is sufficiently *condensed* and no more seams should be pruned from the image.

**G) Multiple seam pruning**

In the original SC, just one seam is pruned at each seam calculation. Hence, pruning many seams increases the computational complexity. Therefore, we reuse a primal seam path for pruning the seams around it. In this paper, we use the seam-by-seam calculation for  $\mathbf{s}_o$  to clarify our contribution. To further reduce the computational complexity, we could use an improved SC method, such as [19], which uses one saliency map for pruning several seams. It is able to speed up the calculation of  $\mathbf{s}_o$ .

We present an algorithm to prune multiple seams with a primal seam path  $\mathbf{a}_p$  in Algorithm 3. In the algorithm,  $\mathbf{B} = \{\mathbf{b}_{-(N_m-1)/2}, \dots, \mathbf{b}_0, \dots, \mathbf{b}_{(N_m-1)/2}\}$  is a set of seams to be pruned at the same time,  $N_m$  is the (odd) number of pruned seams and  $T_m$  is the threshold to determine whether multiple seams can be pruned.

**Algorithm 3** Multiple seam pruning**Require:**  $I$  and  $\mathbf{a}_p$ 

```

1: function MULTISEAMPRUNE( $N_m, T_m$ )
2:    $E_p \leftarrow \text{MSE}(I(:, \mathbf{a}_p), \bar{I}(:, \mathbf{a}_p))$  % Set MSE of the primal seam.
3:    $\mathbf{b}_0 \leftarrow \mathbf{a}_p$ 
4:   Set  $N_m - 1$  seams of  $\mathbf{B}$  so that  $b_{q+1}(i) - b_q(i) = 2$  for  $\forall q, i$ 
5:    $E_m \leftarrow \frac{1}{N_m} \sum_{q=1}^{N_m} \text{MSE}(I(:, \mathbf{b}_q), \bar{I}(:, \mathbf{b}_q))$  % Calculate MSE of the multiple seams.
6:   if  $E_p - E_m < T_m$  then
7:     Set  $\mathbf{B}$  to  $N_m$  seams to be pruned
8:   else
9:     Set  $\mathbf{a}_p$  to one seam to be pruned
10:  end if
11: end function

```

## IV. WAVELET-BASED CAIC

In this section, the wavelet-based multiresolution representation of images for RD-SC is presented. Furthermore, the modified SPIHT algorithm is shown.

**A) Lifting-based seamlet**

For preserving the image coding performance at full resolution, the pixel values of seams should be encoded as well as the core image. Before constructing a full multiresolution image, seams are transformed by the lifting-based discrete wavelet transform (DWT).

The wavelet transform along seam paths, named seamlet, was proposed by Conger *et al.* [20]. It generates coarse and detail coefficients similar to the conventional DWT, but its downsampling positions are determined by seam paths. Though it is able to use any wavelet filters, the original seamlet recommended Haar wavelet. However, we focus on the image coding application using CAIR. The detail coefficients should be as small as possible for image coding. Thus, lifting-based 5/3 DWT is employed for seamlet in this paper. The detail coefficient is calculated as follows:

$$d(i, a(i)) = I(i, a(i)) - [(I(i, a(i) - 1) + I(i, a(i) + 1)) / 2], \quad (17)$$

where  $d(i, a(i))$  is the detail coefficient at  $(i, a(i))$ . This lifting-based transformation is shown in Fig. 8. In our approach, the coarse coefficients remain unchanged different from the original seamlet. That is, the five-tap lowpass DWT filtering is *not* performed, since these unfiltered original pixels in the core image are finally transformed by the conventional DWT. Finally,  $d(i, a(i))$ 's are collected and one column of detail coefficients is yielded. The process is recursively performed for the number of seams.

In this paper, first vertical seams are calculated and transformed by the lifting-based seamlet and then the horizontal seams are calculated and transformed, shown in Fig. 9. Finally, the core image is transformed by the multi-level DWT to construct a fully decomposed multiresolution image. Note that this can be easily extended to the optimal order seam pruning [8] for 2-D CAIR; however, we choose

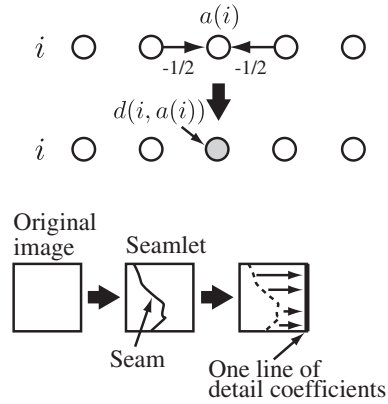


Fig. 8. Lifting-based seamlet. Top: transformation of  $I(i, a(i))$ . Bottom: illustrative example of seamlet for one seam.

this simple 2-D CAIR because of the computational complexity of the optimization of the seam pruning sequence. A retargeted image with an arbitrary resolution is easily obtained by interpolating seam paths.

**B) Pre-quantization of detail seam coefficients**

It is clear that pixel values of the seams are not as important as the core image since prominent regions are preserved in the core image through RD-SC. Therefore, we consider to quantize detail seam coefficients before SPIHT encoding. It is represented as follows:

$$d_q(i, a(i)) = \text{sgn}(d(i, a(i))) \left[ \frac{|d(i, a(i))| + f_q}{\Delta} \right], \quad (18)$$

where  $d_q(i, a(i))$  is the pre-quantized detail seam coefficient,  $f_q$  is the rounding control parameter, and  $\Delta$  is the quantization step size. Clearly, no quantization is performed when  $\Delta = 1$ . The pre-quantized signal is reconstructed by the following equation:

$$d'(i, a(i)) = \Delta d_q(i, a(i)). \quad (19)$$

This pre-quantization is based on the scalar quantization with offset used in H.264/AVC [21] and we use  $f_q = \Delta/3$

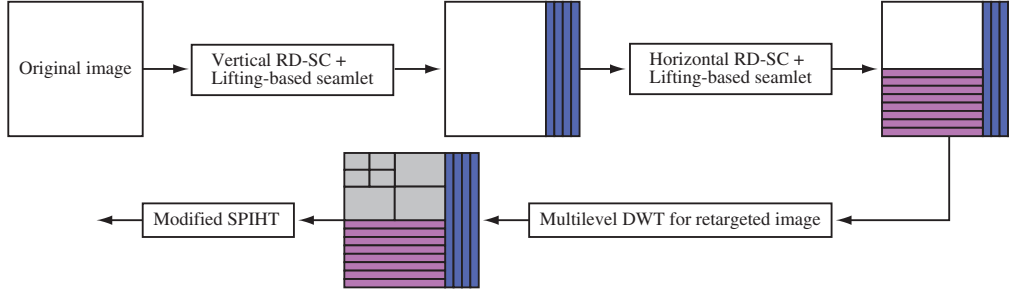


Fig. 9. Encoder structure of the proposed CAIC.

similar to the intra-coding of H.264/AVC. By using this pre-quantization, we can control the qualities between the core image and seams.

### C) Modified tree structure of SPIHT for CAIC

For the proposed multiresolution image representation, we introduce a new parent–children relationship between the transformed coefficients in the core image and detail coefficients of seams for SPIHT encoding [12].

The ordinary parent–children relationship in [12] is defined to connect a pixel to four pixels in the same subband for the next lower decomposition (higher frequency) level. In this paper, the relationship is used for transformed coefficients of the core image. Let  $O(q, r)$  be a set of children indices of the parent index  $(q, r)$ . The relationship is defined as

$$O(q, r) = \{(2q, 2r), (2q, 2r + 1), (2q + 1, 2r), (2q + 1, 2r + 1)\}. \quad (20)$$

Besides the ordinary relationship, we introduce the new relationship between the core image and seams. We define  $(q_c, r_c)$  as the index of a transformed coefficient of the core image after DWT in the HL subband for the lowest decomposition (the highest frequency) level, corresponding to  $I(i, a(i) - 1)$ . Furthermore,  $(q_s, r_s)$  is defined as the index of a detail coefficient of a seam corresponding to  $d_q(i, a(i))$ .

The new parent–children relationship is represented as follows:

$$O(q_c, r_c) \ni \begin{cases} (q_s, r_s) & \text{if } (q_s, r_s) \text{ exists,} \\ \phi & \text{otherwise.} \end{cases} \quad (21)$$

The new parent–children relationship is illustrated in Fig. 10. This relationship means that detail coefficients of seams are related to the nearest core pixels. The case of horizontal seams is similarly defined as well.

## V. EXPERIMENTAL RESULTS

In this section, we validate image coding and CAIR performances. Four eight-bit grayscale images *Park Joy*, *Pisa Tower*, *Beach* and *Crew* are used as test images. Their sizes are summarized in Table 3 and the original images are shown in Fig. 11. All core image sizes are determined

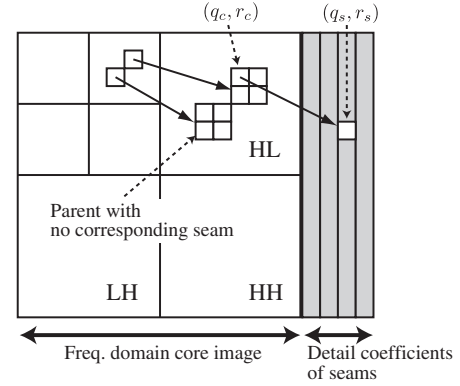


Fig. 10. New parent–children structure of the proposed CAIC. An arrow with a solid line represents the relationship.

Table 3. Sizes of Test Images ( $H_I \times W_I$ )

Image	Original size	Core size
<i>Park joy</i>	704 × 1024	672 × 736
<i>Pisa tower</i>	768 × 1024	608 × 960
<i>Beach</i>	512 × 640	384 × 608
<i>Crew</i>	576 × 704	544 × 640

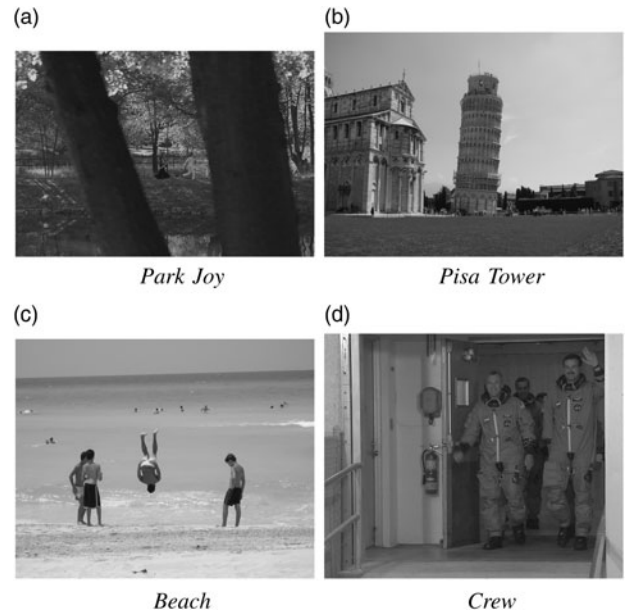


Fig. 11. Test images.



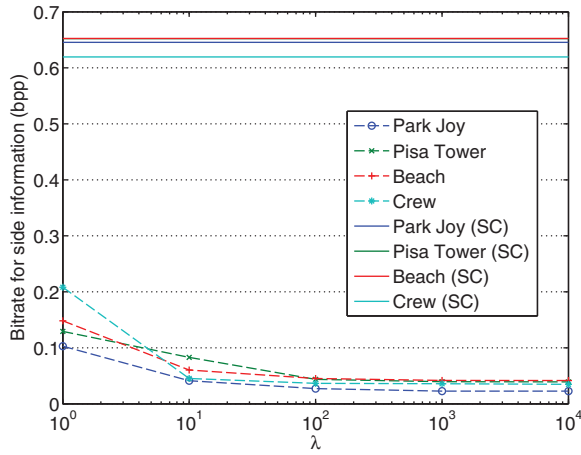
**Table 4.** Parameter Settings for RD-SC.

Parameter	Setup	Purpose	Section described
$L_{max}$	32	Upper limit of $L(k)$	Section III-C
$\lambda$	10	Trade-off factor between seam accuracy and bitrate	Section III-D
$\omega$	0.5	Balancing seam positions	Section III-F
$\tau$	$1.5H_l$ (vertical seams) $1.5W_l$ (horizontal seams)	Limiting core image size	
$T_m$	4	Energy for multiple seam pruning	Section III-G
$N_m$	3	Limiting # of multiple seams pruned at the same time	

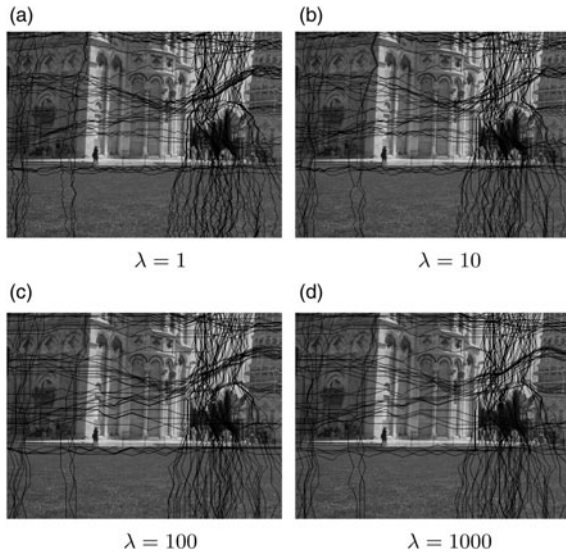
automatically based on the parameter  $\tau$  introduced in Section III-F. Experimentally set parameters are summarized in Table 4.

### A) Effects of parameter setup

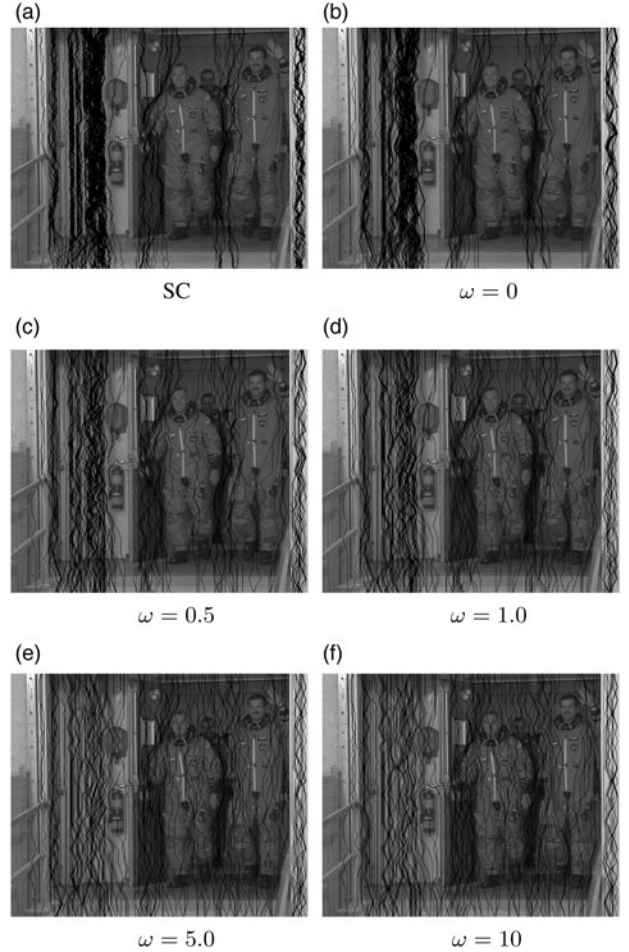
The theoretical optimization of parameters shown in Table 4 is quite difficult since the optimization process could



**Fig. 12.** Bitrates for side information. The solid lines represent required bitrates of SPI by SC.



**Fig. 13.** Approximated seams by RD-SC. The enlarged portions of *Pisa Tower* are shown. Black lines represent approximated seam paths.

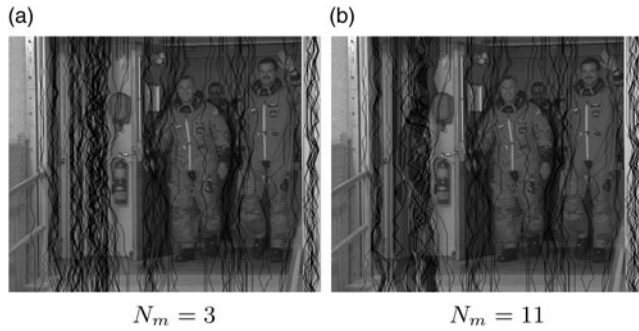
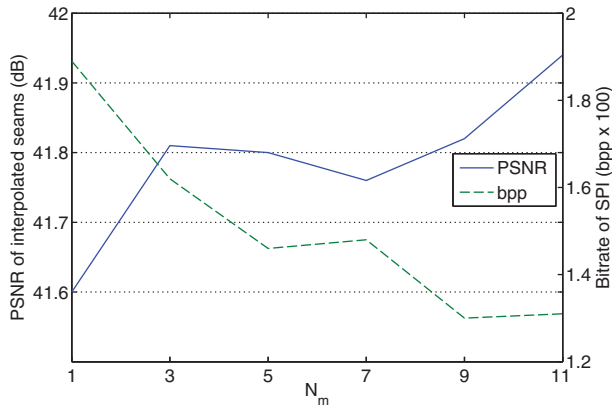


**Fig. 14.** Approximated seams by RD-SC for various  $\omega$ . A total of 128 seams are removed regardless of  $\tau$  for comparison purpose.

be highly non-linear and the best parameter set is certainly depending on the one who views the retargeted and full resolution images. Therefore, all of them are set heuristically as previously mentioned. Here, we describe the effect of parameter settings. In the following experiments in this subsection, one parameter is changed and the remaining ones are the same as Table 4 unless specified. Since effects of a few parameters are relatively clear, i.e., those of  $L_{max}$  and  $\tau$  are analogous, we describe about  $\lambda$ ,  $\omega$ ,  $T_m$ , and  $N_m$ . It is an interesting future work to determine optimal parameters automatically. As a simple approach, we could append a parameter tuning step by a user with some specific test images.

**Table 5.** Linear interpolation results for *Crew* image.

$\omega$	0	0.5	1.0	5.0	10
PSNR of interpolated seams (dB)	40.98	41.81	42.17	42.35	41.71

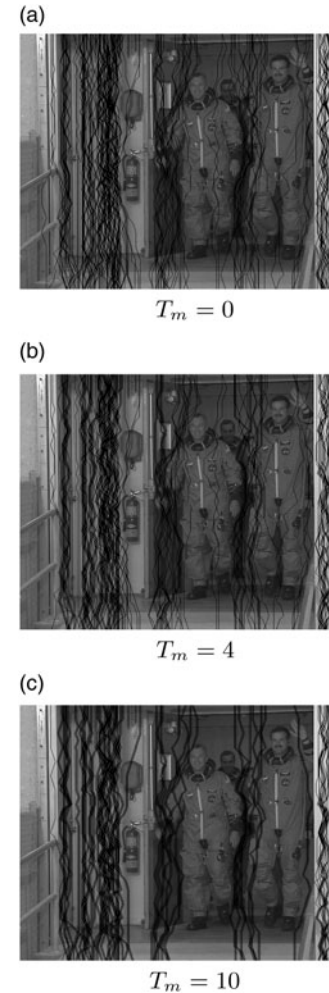
**Fig. 15.** Approximated seams by RD-SC for different  $N_m$  ( $T_m = 4$ ). A total of 128 seams are removed regardless of  $\tau$  for comparison purpose.**Fig. 16.** PSNRs of interpolated seams and bitrates for SPI (*Crew* image, 128 seams).1)  $\lambda$ 

It controls bitrates for SPI and the accuracy of seam paths. In the experiment,  $W_I/4$  vertical seams and  $H_I/4$  horizontal seams are calculated regardless of  $\tau$ . The SPI of SC is encoded by arithmetic coding similar to [9]. Figure 12 is a plot of the bitrate of SPI as a function of  $\lambda$ . As expected, the bitrate decreases as  $\lambda$  increases and the bitrates are much lower than those of SC.

Figure 13 shows the calculated seams of *Pisa Tower* for various  $\lambda$ . The seams represent different paths depending on  $\lambda$ . The seams move finely like SC for a small  $\lambda$ , whereas for a large  $\lambda$ , the seam paths are a coarse approximation of the original ones.

2)  $\omega$ 

It is used for balancing seam positions.  $\omega = 0$  means there are no restrictions for the positions. Figure 14 shows approximated seam paths of *Crew* for various  $\omega$ . The seam paths are significantly changed within the range  $[0, 5]$ . Without the restriction, i.e.,  $\omega = 0$ , seams are mainly located on the smooth areas and the positions are concentrated similar

**Fig. 17.** Enlarged portions of retargeted *Crew* image.**Fig. 18.** Approximated seams by RD-SC for different  $T_m$  ( $N_m = 1$ ). A total of 128 seams are removed regardless of  $\tau$  for comparison purpose.

to the original SC. Whereas for the large  $\omega$ , such as  $\omega = 5.0$  or  $10$ , the seams are distributed uniformly. Seams for the large  $\omega$  are sometimes passing through the prominent regions of the image, e.g., faces of astronauts.



**Fig. 19.** Resizing results. The left, center, and right columns correspond to the resized images by scaling, SC, and RD-SC, respectively. From top to bottom: *Park Joy, Pisa Tower, Beach, and Crew.*

For the retargeting purpose,  $\omega$  should be small; which yields similar seam positions to those of the original SC (Fig. 14(a)). Whereas for the image coding purpose, seam paths should be distributed uniformly. As a simple comparison, Table 5 presents PSNRs of the interpolated seams by linear interpolation. It is clear that  $\omega = 0$ , i.e., the most concentrated case, shows the worst interpolation result since the small  $\omega$  requires interpolating wide areas from few remaining pixels. It is also observed that  $\omega$  should not be too large. Therefore, as a trade off issue, we picked up  $\omega = 0.5$ .

### 3) $N_m$ AND $T_m$

$N_m$  determines the maximum number of seams removed at the same time. The large  $N_m$  permits removing many seams by one seam path calculation step. The seam paths for  $N_m = 3$  and 11 are shown in Fig. 15. As expected, many seams share

one seam paths in Fig. 15(b). PSNRs of interpolated seams by linear interpolation and SPI bitrates are compared in Fig. 16. It seems that the large  $N_m$  is recommended anytime. However, as shown in Fig. 17, the large  $N_m$  significantly affects the retargeted image quality. In the figure, the straight line in the original *Crew* image was broken in the retargeted image with  $N_m = 11$ . Hence, we picked up  $N_m = 3$ , which seems a good trade-off as shown in Fig. 16.

**Table 6.** Execution time (s) of *Beach* image.

# Seams	Seam approx.	Seam dec.	SC
16	2.92	0.031	1.18
32	7.22	0.038	2.32
64	18.37	0.047	4.54
128	31.43	0.089	8.70



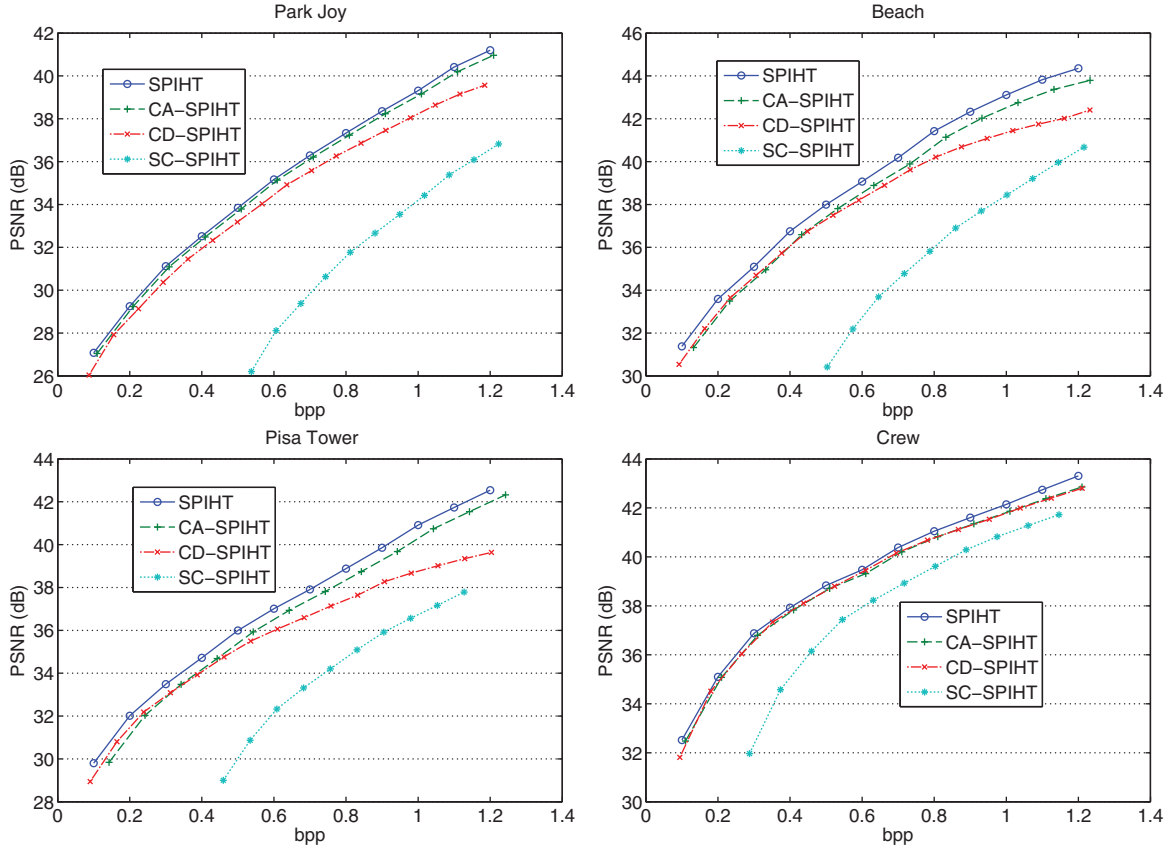


Fig. 20. R-D curves of reconstructed image at full resolution. From left to right, top to bottom: *Park Joy*, *Beach*, *Pisa Tower*, and *Crew*.

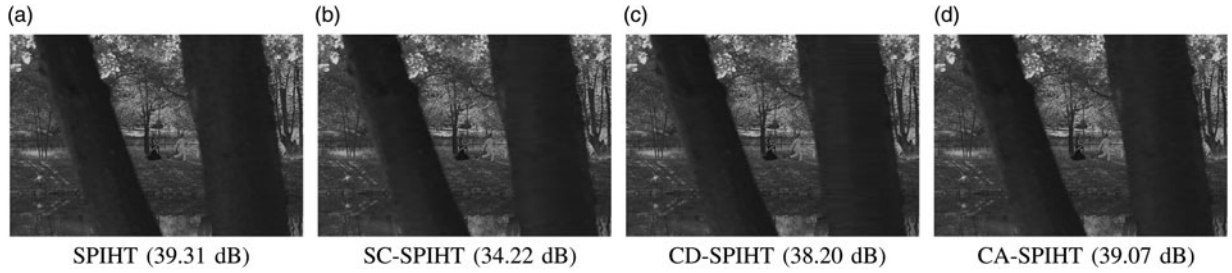


Fig. 21. Reconstructed *Park Joy* images. Each image is encoded at 1.0 bpp.

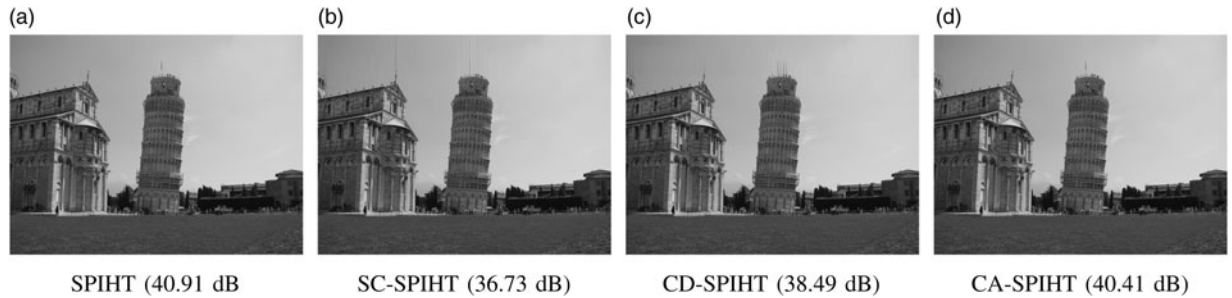


Fig. 22. Reconstructed *Pisa Tower* images. Each image is encoded at 1.0 bpp.

$T_m$  has a similar effect to  $N_m$ , since it decides whether  $N_m$  seams can be pruned or not (see Algorithm 3). If  $N_m > 255^2$  (possible upper bound of MSE), the algorithm always

choose to prune  $N_m$  seams regardless of MSE. Based on the similar experiment about  $T_m$ , we experimentally set  $T_m = 4$ . The approximated seams for a few  $T_m$  are shown in Fig. 18.

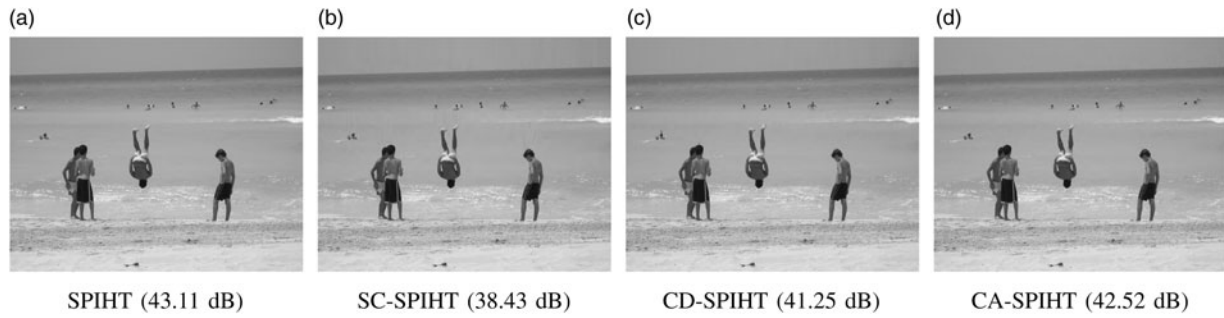


Fig. 23. Reconstructed *Beach* images. Each image is encoded at 1.0 bpp.

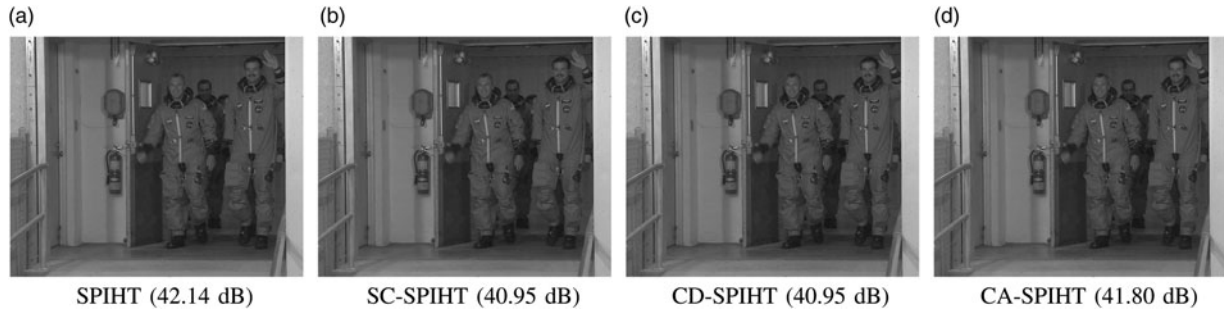


Fig. 24. Reconstructed *Crew* images. Each image is encoded at 1.0 bpp.

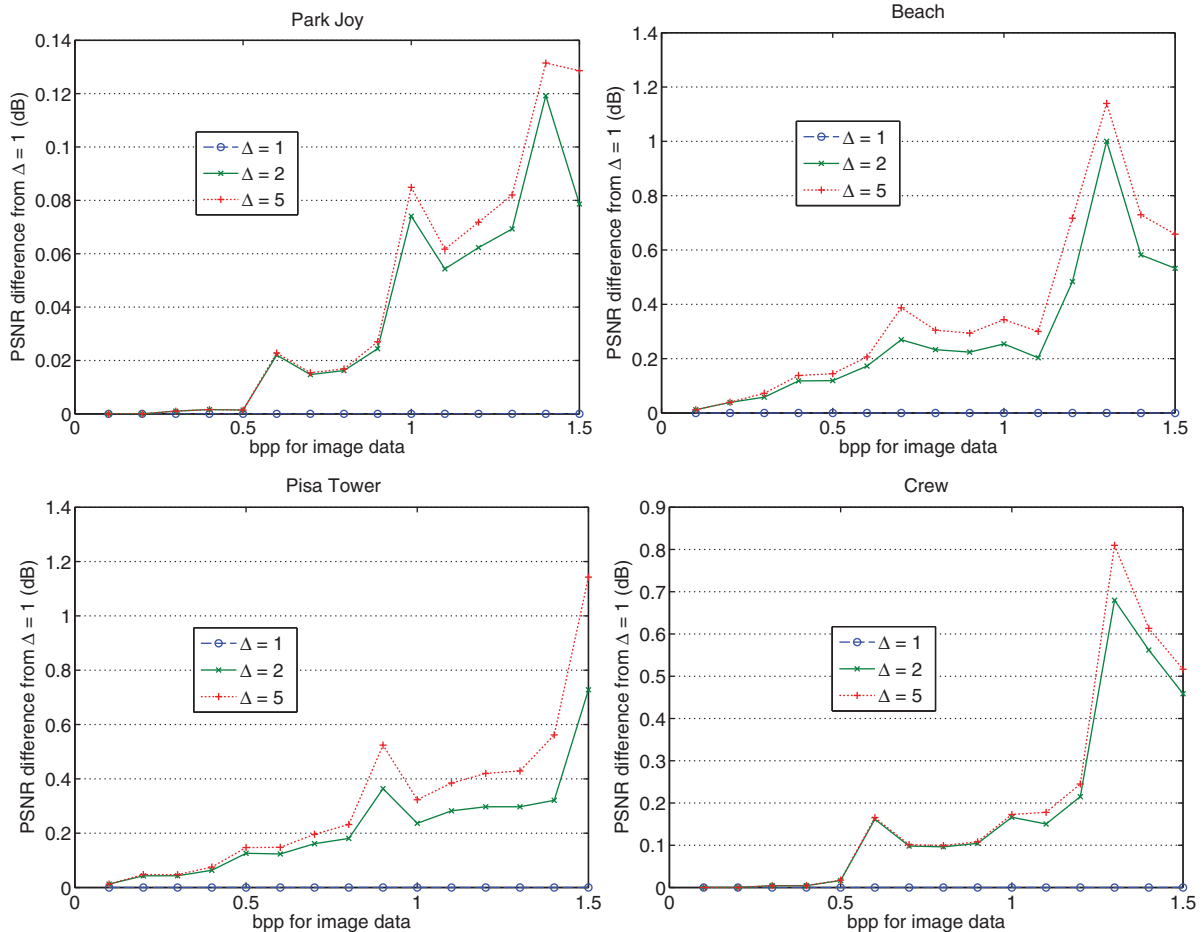


Fig. 25. R-D curves of reconstructed core images for various  $\Delta$ .



## B) CAIR performance

The second comparison is for CAIR performance. As far as we know, few numerical measures exist to compare CAIR performances. Hence, we present all of the retargeted images for visual quality comparison. Table 3 summarizes the sizes of the core images. Figure 19 shows CAIR results by using SC and RD-SC. For comparison purpose, scaled images by `imresize` function of MATLAB are also shown. All images are resized to have the same size of the retargeted images by RD-SC. From the figures, RD-SC presents very similar retargeted images to those obtained by SC.

## C) Computational complexity

It is worth noting that the computational complexity in the seam approximation process of RD-SC is higher than those of SC due to the recursive estimation. It is difficult to present the complexity theoretically due to the highly image-adaptive computations of RD-SC. Thus, the execution time is compared. All algorithms are implemented on a MATLAB (R2011a) platform. The simulation is run on the desktop computer with Intel 3.4-GHz CPU (Corei7-2600), 16-GB RAM. Table 6 shows the execution time of the average of five executions for *Beach* image. In the table, the seam calculation time by SC (the column of “SC”) is also shown for comparison. The computation time is proportional to the number of seams to be removed. Seam path approximation time is longer than that of the original SC. However, Table 6 also presents the seam path decoding time of RD-SC. Obviously, it is significantly faster than the seam calculation of SC. It is beneficial for the traditional client–server model, where the decoder will be a mobile device which often has poor computation power.

## D) Application to image coding

Here, we compare image coding performance. As the conventional encoders, SPIHT, the method in [9] (denoted as SC-SPIHT) and CD-based SPIHT [11] (denoted as CD-SPIHT) are used. Our CAIC method is referred to as CA-SPIHT in this section. Note that SC-SPIHT does not include pixel values of seams; that is, the compressed pruned image is interpolated by linear interpolation since the bitrate for SPI already requires a high percentage of the entire bit budget. All methods apply five-level 9/7 DWT for core images<sup>3</sup> before performing SPIHT.

The R-D curves, encoded with  $\Delta = 1$  in (18), are shown in Fig. 20. SC-SPIHT is inferior to the other methods due to its large amount of bitrates for SPI. Furthermore, CD-SPIHT is worse than SPIHT and CA-SPIHT especially for high bitrates since its non-ROI is just interpolated from the core image at the decoder. SPIHT and CA-SPIHT are comparable, but SPIHT slightly outperforms the CA-SPIHT. More precisely, in comparison to SPIHT, CA-SPIHT increases bitrates less than or equal to 10%. However, the

<sup>3</sup>For SPIHT, the core image is the original image itself.

bitrate increase would be a permissible penalty for users demanding both of a good CAIR quality and image coding performance at full resolution. In addition, reconstructed images at full resolution are shown in Figs. 21–24. Similar to the R-D curves, the reconstructed images by SC-SPIHT and CD-SPIHT have interpolation artifacts, whereas our CA-SPIHT significantly reduces those artifacts and the reconstructed images are very similar to those encoded by SPIHT.

Moreover, Fig. 25 represents PSNRs of core images for a few step sizes of the pre-quantization in Section IV-B. As expected, if  $\Delta$  in (18) is large, the quality of the core image is improved especially for high bitrate coding. In addition, PSNR gains from  $\Delta = 1$  depend on the test images. For example, seam paths for *Park Joy* pass through the very smooth regions, the PSNR gains due to the pre-quantization is marginal, whereas for *Beach*, the PSNR gain is significant since vertical seams pass through the highly textured sand area.

## VI. CONCLUSIONS

In this paper, we proposed rate-dependent structures of SC by piecewise approximation of the original seam paths. Furthermore, a new parent–children relationship of SPIHT is proposed for CAIC. In the experimental results, our approach shows comparable CAIR performances to those of SC with significant reductions of required bitrates for side information. Its image coding performances are comparable to those of the original SPIHT and also it consistently outperforms the conventional CAICs. Our RD-SC could be extended to video retargeting, which is one of our ongoing studies.

## ACKNOWLEDGEMENTS

The authors thank krossbow (*Mt. Evans*), wsupho01 (*Pisa Tower*), and wenko (*Beach*) of flickr ([www.flickr.com](http://www.flickr.com)) for making available images in creative commons license. This work was supported in part by JSPS KAKENHI Grant Number 24760288, SCAT Research Grant, The Nakajima Foundation, and Kayamori Foundation of Informational Science Advancement.

## REFERENCES

- [1] Avidan, S.; Shamir, A.: Seam carving for content-aware image resizing. *ACM Trans. Graph.*, **26**(3) (2007), 9 pp., Article 10. DOI = 10.1145/1239451.1239461. <http://dx.doi.org/10.1145/1239451.1239461>.
- [2] Rubinstein, M.; Shamir, A.; Avidan, S.: Improved seam carving for video retargeting. *ACM Trans. Graph.*, **27**(3) (2008), 9 pp., Article 16. DOI = 10.1145/1360612.1360615. <http://dx.doi.org/10.1145/1360612.1360615>.
- [3] Wolf, L.; Guttmann, M.; Cohen-Or, D.: Non-homogeneous content-driven video-retargeting, in *Proc. ICCV'07*, 2007.
- [4] Wang, Y.; Tai, C.-L.; Sorkine, O.; Lee, T.-Y.: Optimized scale-and-stretch for image resizing. *ACM Trans. Graph.* **27**(5) (2008), 8 pp.,

- Article 118. DOI = 10.1145/1409060.1409071. <http://doi.acm.org/10.1145/1409060.1409071>.
- [5] Guo, Y.; Liu, F.; Shi, J.; Zhou, Z. H.; Gleicher, M.: Image retargeting using mesh parametrization. *IEEE Trans. Multimedia*, **11**(5) (2009), 856–867.
  - [6] Domingues, D.; Alahi, A.; Vanderghyest, P.: Stream carving: an adaptive seam carving algorithm, in *Proc. ICIP'10*, 2010.
  - [7] Dong, W.; Zhou, N.; Paul, J.; Zhang, X.: Optimized image resizing using seam carving and scaling. *ACM Trans. Graph.*, **28**(5) (2009), 10 pp. Article 125. DOI = 10.1145/1618452.1618471. <http://doi.acm.org/10.1145/1618452.1618471>.
  - [8] Rubinstein, M.; Shamir, A.; Avidan, S.: Multi-operator media retargeting. *ACM Trans. Graph.*, **28**(3) (2009), 11 pp. DOI = 10.1145/1531326.1531329. <http://doi.acm.org/10.1145/1531326.1531329>.
  - [9] Nguyen, A.; Yang, W.; Cai, J.: Seam carving extension: a compression perspective, in *Proc. 17th ACM Int. Conf. on Multimedia*, 2009, 825–828.
  - [10] Tanaka, Y.; Hasegawa, M.; Kato, S.: Image coding using concentration and dilution based on seam carving with hierarchical search, in *Proc. ICASSP'10*, 2010, 1322–1325.
  - [11] ———, “Improved image concentration for artifact-free image dilution and its application to image coding”, in *Proc. ICIP'10*, 2010, 1225–1228.
  - [12] Said, A.; Pearlman, W. A.: A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Trans. Circuits Syst. Video Technol.*, **6**(3) (1996), 243–250.
  - [13] Décombas, M.; Capman, F.; Renan, E.; Dufaux, F.; Pesquet-Popescu, B.: Seam carving for semantic video coding, in *Proc. SPIE, Applications of Digital Image Processing XXXIV*, 2011.
  - [14] Deng, C.; Lin, W.; Cai, J.: Content-based image compression for arbitrary-resolution display devices, in *Proc. ICC 2012*, 2011, 1–5.
  - [15] ———, Content-based image compression for arbitrary-resolution display devices. *IEEE Trans. Multimedia*, **14**(4) (2012), 1127–1139.
  - [16] V̄o, D. T.; Solé, J.; Yin, P.; Gomila, C.; Nguyen, T. Q.: Selective data pruning-based compression using high-order edge-directed interpolation. *IEEE Trans. Image Process.*, **19**(2) (2010), 399–409.
  - [17] Wang, T.; Urahama, K.: Cartesian resizing of image and video for data compression, in *Proc. TENCON 2010*, 2010, 1651–1656.
  - [18] Ortega, A.; Ramchandran, K.: Rate-distortion methods for image and video compression. *IEEE Signal Process. Mag.*, **15**(6) (1998), 23–50.
  - [19] Chiang, C. K.; Wang, S. F.; Chen, Y. L.; Lai, S. H.: Fast JND-based video carving with GPU acceleration for real-time video retargeting. *IEEE Trans. Circuits Syst. Video Technol.* **19**(11) (2009), 1588–1597.
  - [20] Conger, D. D.; Radha, H.; Kumar, M.: Seamlets: Content-aware non-linear wavelet transform, in *Proc. ICASSP'10*, 2010, 1450–1453.
  - [21] Wedi, T.; Wittman, S.: Rate-distortion constrained estimation of quantization offsets, in *Joint Video Team (JVT), Doc. JVT-O066*, Busan, Korea, April 2005.

**Yuichi Tanaka** received B.E., M.E. and Ph.D. degrees in Electrical Engineering from Keio University, Yokohama, Japan,

in 2003, 2005, and 2007, respectively. He was a Postdoctoral Scholar at Keio University, Yokohama, Japan, from 2007 to 2008 supported by the Japan Society for the Promotion of Science (JSPS). From 2006 to 2008, he was also a visiting scholar at the University of California, San Diego (Video Processing Group supervised by Prof. T. Q. Nguyen). From 2008 to 2012, he was an Assistant Professor in the Department of Information Science, Utsunomiya University, Tochigi, Japan. Since 2012, he has been an Associate Professor in Graduate School of BASE, Tokyo University of Agriculture and Technology, Koganei, Japan. He is a recipient of Yasujiro Niwa Outstanding Paper Award in 2010, and also TELECOM System Technology Award in 2011. His current research interests are in the field of multidimensional signal processing which includes: image and video coding with computer vision techniques, distributed video coding, objective quality metric, and effective time–frequency transform design.

**Taichi Yoshida** received the B.E. and M.E. degrees in Electrical Engineering from Keio University, Yokohama, Japan, in 2006 and 2008, respectively. He is currently a Ph.D. candidate at Keio University, Yokohama, Japan, under the supervision of Prof. Masaaki Ikehara. His research interests are in the field of filter bank design and its image coding application.

**Madoka Hasegawa** received the Ph.D. in Engineering from Utsunomiya University, Japan in 1999. She is an Associate Professor at the Graduate School of Engineering, Utsunomiya University. Her current research interests include image coding, image processing, digital watermark, and graphical passwords. She is a member of IEICE (the Institute of Electronics Information and Communication Engineers) and IEEE.

**Shigeo Kato** received the B.S. in Engineering from Hokkaido University and Ph.D. in Engineering from Tokyo University, Japan, in 1972 and 1989, respectively. He is a Professor at the Graduate School of Engineering, Utsunomiya University. His current research interests include image coding, image processing, information theory, and digital watermark. He is a member of IEICE (the Institute of Electronics Information and Communication Engineers), IIEEJ (the Institute of Image Electronics Engineers of Japan), and IEEE. He is a fellow of IIEEJ.

**Masaaki Ikehara** received the B.E., M.E. and Dr. Eng. degrees in Electrical Engineering from Keio University, Yokohama, Japan, in 1984, 1986, and 1989, respectively. He was Appointed Lecturer at Nagasaki University, Nagasaki, Japan, from 1989 to 1992. In 1992, he joined the Faculty of Engineering, Keio University. From 1996 to 1998, he was a visiting researcher at the University of Wisconsin, Madison, and Boston University, Boston, MA. He is currently a Full Professor with the Department of Electronics and Electrical Engineering, Keio University. His research interests are in the areas of multi-rate signal processing, wavelet image coding, and filter design problems.