Region-of-Interest Coding Based on Set Partitioning in Hierarchical Trees

Keun-hyeong Park and HyunWook Park, Senior Member, IEEE

Abstract—In many image-coding applications such as web browsing, image databases, and telemedicine, it is useful to reconstruct only a region of interest (ROI) before the rest of the image is reconstructed. In this paper, an ROI coding functionality is incorporated with the set partitioning in hierarchical trees (SPIHT) algorithm for wavelet-based image coding. By placing a higher emphasis on the transform coefficients pertaining to the ROI, the ROI is coded with higher fidelity than the rest of the image in earlier stages of progressive coding. The general thrust of this research is to identify necessary coefficients in wavelet-transform domain for the decoder to reconstruct the desired region. This new method provides better performance than the previously presented methods.

Index Terms—Parent of ROI (PROI), progressive transmission, region of interest (ROI) coding, set partitioning in hierarchical trees (SPIHT), wavelet transform.

I. INTRODUCTION

The set partitioning in hierarchical trees (SPIHT) algorithm [1] achieves an excellent rate-distortion performance and retains an attractive embedded code property useful for progressive transmission. It is essential for the coder to provide a good rate-distortion performance. In addition, a lot of other requirements become important in still image compression. Examples of such requirements are ability to provide lossy and lossless compression within a single encoding system, ability to provide a scalability of fidelity and resolution, and ability to give higher priority to a region of interest (ROI) [2]. It would be desirable to incorporate the above features into an image coding system without incurring heavy costs such as increased computational complexity or reduced rate-distortion performance.

Like the embedded zero-tree wavelet (EZW) [3], SPIHT generally operates on an entire image at once. The SPIHT captures large sets of insignificant coefficients within data structures called spatial orientation trees [1] to make the rate-distortion performance efficient. The coding of these sets achieves greater rate-distortion efficiency than the coding of individual coefficients. However, the spatial orientation trees make it difficult to incorporate a functionality of ROI coding into the SPIHT algorithm. Reference [4] modified the SPIHT to quantize and code the wavelet coefficients in the arbitrary ROI only. This method is optimized for wavelet coding of an arbitrary-shaped region, not for entire image coding with higher fidelity for the ROI. Reference [5] proposed a method considering the entire image coding with an emphasis on the ROI, while its rate-distortion performance is not competitive with the original SPIHT algorithm.

In this work, we incorporate an ROI coding functionality into the SPIHT algorithm without compromising other desirable features such as rate-distortion performance and computation time. Necessary data for the decoder to reconstruct the desired region can be identified by the proposed ROI coding method without any overhead of bit stream.

This paper is organized as follows. Section II presents a new ROI coding method. Section III shows experimental results, comparing the proposed method with the previous work and the original SPIHT. Finally, conclusions are given in Section IV.

II. ROI CODING

A. Generation of ROI Mask

When an image is coded with an emphasis of ROI, it is necessary to identify the wavelet coefficients needed for the reconstruction of the ROI. Thus, the ROI mask is introduced to indicate which wavelet coefficients have to be transmitted exactly in order for the receiver to reconstruct the ROI [6].

Once an arbitrarily shaped ROI is defined by user, generation of the ROI mask is performed for rows and columns at each decomposition level. The process is then repeated for the remaining levels until the entire wavelet tree is processed, as shown in Fig. 1. The wavelet coefficients that are required to reconstruct a pixel are selected with dependency on the wavelet length. For example, let the original samples be denoted $x(n)$ and the samples belonging to the low- and high-frequency subbands be denoted $L(n)$ and $H(n)$, respectively. Then, for the
Fig. 2. The 9/7-tap inverse wavelet transform: The coefficients necessary to reconstruct \( X(2n) \) and \( X(2n + 1) \) are \( L(n - 1) \) to \( L(n + 2) \) and \( H(n - 2) \) to \( H(n + 2) \), respectively.

Fig. 3. Example of ROI mask generation for 9/7-tap wavelet: The region of interest is a rectangle of \( 4 \times 4 \).

9/7-tap wavelet filter, the coefficients necessary to reconstruct \( X(2n) \) are \( L(n - 1) \), \( L(n + 1) \), \( H(n - 2) \), \( H(n - 1) \), \( H(n) \), and \( H(n + 1) \). And to reconstruct \( X(2n + 1) \), we need \( L(n - 1) \), \( L(n) \), \( L(n + 1) \), \( L(n + 2) \), \( H(n - 2) \), \( H(n - 1) \), \( H(n) \), \( H(n + 1) \), and \( H(n + 2) \) [2], [7], as shown in Fig. 2. A simple example of ROI mask creation using 9/7-tap wavelet filter is shown in Fig. 3. The image size is \( 32 \times 32 \) and the decomposition level is 2.

B. Generation of Parent of ROI (PROI) Mask

The previous ROI coding algorithm [5] examines whether each node is necessary for the decoder to reconstruct the ROI, i.e., whether it is an ROI coefficient, before testing whether it is significant (node test). Then, if the node is not an ROI coefficient, its node test is skipped and performed later. After the node tests for all ROI coefficients, the encoder examines whether descendants of each node are significant (descendant test) without testing whether descendants of the node are the ROI coefficients, i.e., whether the node is a PROI coefficient.

However, the information about whether its descendants are significant or not is not necessary for the decoder to reconstruct the ROI if a node is not a PROI coefficient. If we know that any descendant of a node within the spatial orientation trees is not an ROI coefficient, then its descendant test does not need to be taken. Therefore, a new mask of PROI is necessary, which indicates the nodes whose descendants are ROI coefficients. The PROI mask is a bit plane indicating which nodes have at least one descendant necessary for the reconstruction of the ROI. The generation of the PROI mask is depicted in Fig. 4. Except for the highest frequency subband that has no descendant, the relation between the ROI mask and the PROI mask is defined as follows:

\[
\text{PROI}(i, j) = \text{ROI}((\text{Offs}_1(i, j)) \lor \text{ROI}(\text{Offs}_2(i, j))) \\
\lor \text{ROI}(\text{Offs}_3(i, j)) \lor \text{ROI}(\text{Offs}_4(i, j))
\]

where \( \text{PROI}(i, j) \) and \( \text{ROI}(i, j) \) represent the binary values of the PROI mask and the ROI mask at node \((i, j)\), respectively. \( \text{Offs}_k(i, j) \) represents the \( k \)th offspring of the node \((i, j)\), and \( \lor \) denotes the binary OR operation.

C. Divisions of the List of Insignificant Pixels and the List of Insignificant Sets

We use the following function to indicate the significance of a set of coordinates \( T \) and use the following sets of coordinates as defined in the original SPIHT:

\[
S_n(T) = \begin{cases} 
1, & \max\{\|\mathbf{c}_{\mathbf{i}, \mathbf{j}}\| \geq 2^n \\
0, & \text{otherwise} 
\end{cases} 
\]

where

\( \mathbf{O}(i, j) \) set of all offspring of node \((i, j)\); \n\( \mathbf{D}(i, j) \) set of all descendants of node \((i, j)\); \n\( \mathbf{L}(i, j) = \mathbf{D}(i, j) - \mathbf{O}(i, j) \)

In the SPIHT, node tests and descendant tests are performed with the maximum threshold, \( 2^N \), first. Then, the tests are repeated with smaller threshold, \( 2^n, n = N - 1, N - 2, \ldots \), iteratively until the compressed bit amount reaches a predefined value. The node test of a node \((i, j)\) is performed for a threshold value of \( 2^n \) as follows:

\[ |\mathbf{c}_{i, j}| \geq 2^n \]

The descendant test of a node \((i, j)\) is also described for a threshold value of \( 2^n \) as follows:

\[
\max\{\|\mathbf{c}_{k, l}\| \geq 2^n \}
\]

where \( \mathbf{c}_{k, l} \) is a wavelet coefficients at \((k, l)\). If the descendant test of a node \((i, j)\) in (4) is satisfied with a threshold value of
The set $D(i, j)$ is partitioned into four subsets whose roots are four offspring of the node $(i, j)$. The node and the descendant tests are repeated to the four subsets.

In the ROI coding, the node and the descendant tests are performed on the ROI first. After completing the tests of the ROI, the list of insignificant pixels (LIP) and list of insignificant sets (LIS) have roots of all subsets that are not required to reconstruct the ROI. Some of the subsets on the ROI may be included in the LIP and the LIS after the descendant test with a threshold value of $2^n$ for $n < N$. If we know a threshold value of $2^n$ at which a subset is included into the LIP and the LIS during the test of ROI, the tests of the subset can start at the threshold value of $2^n$, not $2^N$ for the tests of non-ROI. Thus, we can save any redundant bit streams. However, we cannot know the threshold value of $2^n$ for each subset in the LIP and the LIS after the tests of ROI. The tests of non-ROI are performed from the maximum threshold value of $2^N$ for all subsets in the LIP and the LIS that are obtained from the tests of ROI.

---

Fig. 4. Example of PROI mask generation from ROI mask. (a) Relation of a PROI mask to an ROI mask. (b) Example of ROI and PROI masks.

Fig. 5. Divisions of the LIP and the LIS: Each offspring node is added to an LIP and an LIS according to a bit-plane of $n$ (or threshold $2^n$) at which it becomes root of the subtree.

Fig. 6. Block diagram of the proposed ROI coding.
This problem can be settled by dividing the LIP and the LIS into multiple LIPs and LISs, respectively. We divide the LIP and the LIS as follows. At the beginning of tests, if the $N$th bit-plane is the most significant bit-plane (MSB), the nodes in the highest tree level are added to LIP$\text{LIP}_N$, and those with descendants are added to LIS$\text{LIS}_N$ in Fig. 5 [1]. Then, these nodes are tested one by one. If a node becomes significant during the descendant test with the threshold value of $2^n$ (or bit-plane number $n$), its four offspring are added to an LIP$\text{LIP}_n$ and an LIS$\text{LIS}_n$.

As an example of Fig. 5, node $P_3$ is in the LIS$\text{LIS}_N$ at the beginning of the descendant test. If $P_3$ becomes significant at $n = N$ (i.e., $S_N(D(P_3)) = 1$), its four offspring ($P_1$, $P_2$, $P_3$, and $P_4$) are added to LIP$\text{LIP}_N$. Because $P_1$, $P_2$, $P_3$, and $P_4$ become roots of the subtrees at $n = N$, these nodes are added to LIS$\text{LIS}_N$. Next, if $P_3$ in LIS$\text{LIS}_N$ becomes significant at $n = N - 1$, its four offspring ($P_5$, $P_6$, $P_7$, and $P_8$) are added to LIP$\text{LIP}_{N-1}$ and LIS$\text{LIS}_{N-1}$. In this way, the LIP and the LIS are divided into LIP$\text{LIP}_n$ and LIS$\text{LIS}_n$ for $n = N, N - 1, N - 2, \ldots$ during the encoding and decoding processes.

D. Proposed ROI Coding Algorithm

The proposed ROI coding is based on the SPIHT algorithm. In the proposed ROI coding, once an ROI or multiple ROIs are identified, the transmission order of the encoder outputs is modified to place more emphasis on the ROI. That is, an embedded bit stream that is just reordered and has no overhead is generated in such an order that the ROI is refined earlier than the rest of the image. The proposed ROI encoding algorithm whose block diagram is shown in Fig. 6 is described as follows.

1) Start encoding the wavelet transform (WT) coefficients as the original SPIHT [1] (start at $n = N$, when maximum
magnitude of the WT coefficients is greater than or equal to $2^N$ and less than $2^{N+1}$). In the proposed method, however, several LIPs and LISs are used instead of an LIP and a LIS, as described in Section II-C.

2) As soon as the shape of the ROI is available to the encoder (the shape of the ROI can be defined in the encoder before the encoding process, or a decoder can define it and send it to the encoder after the decoder has reconstructed a poor image with the progressively transmitted bitstream in the receiver end), stop Step 1) and perform the following steps.

3) Create the ROI mask and the PROI mask from the ROI.

4) Start the encoding in the ROI coding mode as shown in Fig. 7 from $n = Q$ to $n = R$ (suppose that after completing the encoding of bit-plane $n = Q$, the encoder identifies the ROI). $R$ is a user-defined parameter that controls the relative importance of the ROI compared to the rest of the image.

5) Repeat Step 4) for the coefficients that are not yet tested from $n = Q$ to $n = R$, i.e., that are not included in the ROI.

6) Resume encoding the overall WT coefficients from $n = R - 1$ as in Step 1) until the compressed bit amount reaches a predefined value.

In Fig. 7, steps starting with "#" are the modification of the previous encoding algorithm [5] from the SPIHT algorithm [1] in order to incorporate the ROI coding functionality. By this modification, the node tests for the non-ROI coefficients can be skipped during the ROI coding mode. However, this modification is insufficient, as was mentioned in the previous subsections. Thus, additional modification is needed. Steps starting with "$" are the additional modification of the proposed encoding algorithm, which remove the coding redundancies by skipping the descendant tests for the non-PROI coefficients and by dividing an LIP and an LIS into multiple LIPs and LISs, respectively. The rest of the steps and the notations are the same as the original SPIHT algorithm [1].

III. EXPERIMENTAL RESULTS

In the experiments, the proposed ROI coding method was compared with the SPIHT coder [1] and the previous ROI coder.
Fig. 10. Lena images obtained from the original SPIHT. (a) 0.1 bpp. (b) 0.2 bpp. (c) 0.5 bpp. (d) 1.0 bpp.

Fig. 11. Lena images obtained from the proposed ROI algorithm ($R = 4$). (a) 0.1 bpp. (b) 0.2 bpp. (c) 0.5 bpp. (d) 1.0 bpp.

[5] The following results were obtained from monochrome Lena (256 × 256), Girl (512 × 512), and Baboon (512 × 512) images with 8 bits/pixel (bpp). The compressed bitstreams are obtained without arithmetic coding. Five levels of discrete wavelet transform with the 9/7-tap biorthogonal wavelet filters [8] are employed for Lena, and six levels for the Girl and the Baboon. The ROIs are specified as the 51 × 51 square region containing an eye for Lena, 101 × 101 square region for the Girl, and 101 × 151 rectangular region for the Baboon. The switching bit rate—the bit rate when the ROI coding mode starts—is 0.058 bpp. Since the SPIHT has better visual quality than JPEG at very high compression, the bit rate of 0.058 bpp usually provides a perceptible image for a user to define the desired ROI.

Figs. 8 and 9 show the PSNR values of the entire Lena image and the ROI. The values of relative significance $R$ are 0 and 4 in Figs. 8 and 9, respectively. One can see that the smaller the value of $R$ is, i.e., the more significant the ROI is, the higher the PSNR of the ROI is, while the lower the PSNR of the non-ROI is. These results show that the proposed method yields much better quality than the previous ROI coding methods.

Figs. 10–12 show the reconstructed Lena images from the SPIHT and the proposed method with $R = 4$ and 0, respectively, at bit rates of 0.1, 0.2, 0.5, and 1.0 bpp. The square regions (ROI) containing eyes in Figs. 11 and 12 are reconstructed with higher visual quality than those in Fig. 10. Thus, our ROI coding technique can allow the user to quickly view the ROI with higher quality without receiving the entire image. Figs. 13 and 14 show the PSNR values of the entire image and the ROI for the Girl and the Baboon, respectively ($R = 0$). These show similar results to the Lena image.

Tables I and II show execution times of the encoder and decoder, respectively. The results were generated on a PC with a 550-MHz Pentium III processor and 256-MB memory. The increase of execution time for the proposed ROI coding is not much.

**IV. CONCLUSION**

We incorporated an ROI coding functionality, which modifies the information ordering of the SPIHT to place a higher emphasis on the ROI. This was achieved without compromising the rate-distortion performance or computation time. The proposed algorithm allows the user to request an ROI or several ROIs at any moment and calculates a PROI mask that specifies which coefficients have at least one ROI descendant within the spatial orientation trees. Using the PROI and ROI masks and
Table I: Encoding time for the 256 x 256 Lena image with the SPIHT, the previous ROI coding algorithm, and the proposed ROI coding algorithm.

<table>
<thead>
<tr>
<th>Bit Rate (bpp)</th>
<th>SPIHT [1]</th>
<th>Previous ROI Coding [5]</th>
<th>Proposed ROI Coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 bpp</td>
<td>0.19 sec</td>
<td>0.20 sec</td>
<td>0.22 sec</td>
</tr>
<tr>
<td>0.9 bpp</td>
<td>0.18 sec</td>
<td>0.18 sec</td>
<td>0.20 sec</td>
</tr>
<tr>
<td>0.7 bpp</td>
<td>0.16 sec</td>
<td>0.16 sec</td>
<td>0.18 sec</td>
</tr>
<tr>
<td>0.5 bpp</td>
<td>0.14 sec</td>
<td>0.16 sec</td>
<td>0.16 sec</td>
</tr>
<tr>
<td>0.3 bpp</td>
<td>0.12 sec</td>
<td>0.14 sec</td>
<td>0.14 sec</td>
</tr>
<tr>
<td>0.1 bpp</td>
<td>0.10 sec</td>
<td>0.12 sec</td>
<td>0.12 sec</td>
</tr>
</tbody>
</table>

Table II: Decoding time for the 256 x 256 Lena image with the SPIHT, the previous ROI coding algorithm, and the proposed ROI coding algorithm.

<table>
<thead>
<tr>
<th>Bit Rate (bpp)</th>
<th>SPIHT [1]</th>
<th>Previous ROI Coding [5]</th>
<th>Proposed ROI Coding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 bpp</td>
<td>0.13 sec</td>
<td>0.13 sec</td>
<td>0.15 sec</td>
</tr>
<tr>
<td>0.9 bpp</td>
<td>0.12 sec</td>
<td>0.12 sec</td>
<td>0.14 sec</td>
</tr>
<tr>
<td>0.7 bpp</td>
<td>0.10 sec</td>
<td>0.10 sec</td>
<td>0.12 sec</td>
</tr>
<tr>
<td>0.5 bpp</td>
<td>0.09 sec</td>
<td>0.10 sec</td>
<td>0.10 sec</td>
</tr>
<tr>
<td>0.3 bpp</td>
<td>0.07 sec</td>
<td>0.08 sec</td>
<td>0.08 sec</td>
</tr>
<tr>
<td>0.1 bpp</td>
<td>0.05 sec</td>
<td>0.07 sec</td>
<td>0.07 sec</td>
</tr>
</tbody>
</table>

References

dividing the LIP and the LIS into several LIPs and LISs, respectively, necessary data for the decoder to reconstruct the ROI and the background can be identified without any overhead of bitstream. The ROI coding is especially valuable in interactive client/server applications linked through narrowband networks. The greatest advantage of the proposed ROI coding is to incorporate a new functionality while its performance is competitive with the original SPIHT algorithm.


**Keun-hyeong Park** received the B.S. degree in electronic engineering from Yonsei University, Seoul, Korea, in 1999, and the M.S. degree in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2001.

He has been with Hanaro Telecom, Inc., Seoul, Korea, since February 2001. His current research interests are image processing and image/video compression.

**HyunWook Park** (SM’99) received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Seoul, Korea, in 1983 and 1988, respectively.

He has been a Professor in the Electrical Engineering Department, KAIST, Seoul, Korea, since 1993. He was a Research Associate at the University of Washington at Seattle from 1989 to 1992 and a Senior Executive Researcher at Samsung Electronics Company Ltd. from 1992 to 1993. His current research interests include image computing system, image compression, medical imaging, and multimedia system.