

# POST-SEGMENTATION APPROACH FOR LOSSLESS REGION OF INTEREST CODING

Ji Hyun Lee<sup>1</sup> and Jung H. Kim<sup>2</sup>

<sup>1</sup>Department of Electrical Engineering, Tuskegee University, Tuskegee, Alabama, USA  
jhlee@mytu.tuskegee.edu

<sup>2</sup>Department of Electrical and Computer Engineering, NC A&T University, Greensboro,  
North Carolina, USA  
kim@ncat.edu

## **ABSTRACT**

*This paper presents a lossless region of interest coding technique that is suitable for interactive telemedicine over networks. The new encoding scheme allows a server to transmit only a part of a compressed image data progressively as a client requests it. This technique is different from region scalable coding in JPEG2000 since it does not define region of interest (ROI) when encoding occurs. In the proposed method, the image is fully encoded and stored in the server. It also allows a user to select a ROI after the compression is done. This feature is the main contribution of research. The proposed coding method achieves the region scalable coding by using the integer wavelet lifting, successive quantization, and partitioning that rearranges the wavelet coefficients into subsets. Each subset that represents a local area in an image is then separately coded using run-length and entropy coding. In this paper, we will show the benefits of using the proposed technique with examples and simulation results.*

## **KEYWORDS**

*Region Of Interest, Post-Segmentation, & Integer Wavelet Lifting*

## **1. INTRODUCTION**

Digitalization is widely used for storage, archiving, and transmission of images. Current compression schemes produce high compression rates if loss of quality is acceptable. However, images for medical applications cannot have any deficiency in diagnostically important regions. An approach that supports the transmission of the medical image with a high compression rate overall and truthful information in the ROI is thus necessary.

There are two types of region scalable coding. One is identifying and perfectly compressing diagnostically critical regions, while allowing high compression encoding to other regions. The main reason for preserving regions other than the ROI is to let the viewer more easily locate the position of the critical regions in the original image, and to evaluate possible interactions with surrounding organs. We call this scalable coding method a pre-segmentation approach since ROIs are identified and segmented before encoding actually happens. JPEG2000, set partitioning in hierarchical trees (SPIHT) [1], and EZW support the pre-segmentation approach [2, 3, 4]. The challenge of the pre-segmentation approach is to get an automatic segmentation. Segmentation is one of the most significant and difficult tasks in the area of image processing. If a segmentation

algorithm fails to detect correct ROIs, then the result would become a disaster since the important diagnostic data in the original image would possibly be lost.

Another method is that we call a post-segmentation approach. It can solve the problem of the pre-segmentation approach since a lossless compression is applied to the whole image and stored in the server. By applying special treatments such as set partitioning and rearrangement to the compressed data, small regions of an image are identified and compressed separately. A server that stores encoded medical images initially sends an image using a basic, low quality transmission to a user. If the user wants to see a specific area of interest with better quality, then the server only needs to send the part of the bit stream that represents the specific area. This will reduce data traffic across the network while satisfying the user's requirements.

Typically, medical images require huge amounts of data even though they may be compressed. Since lossless compression is applied to the image, a large amount of data is required to reduce the mean square error to an acceptable level for the high resolution. We illustrate the advantage of using region scalable coding using Figure 1. We applied EZW [4] to a magnetic resonance image (MRI) shown in Figure 1(a). In this example, we used an integer wavelet lifting scheme to obtain perfect reconstruction of the image. Figure 1(b) shows the relationship between the amount of the data and quality of the image. As shown in the figure, the quality of the picture began to saturate beyond 200 kbits. The picture quality, around 200 kbits, would be good since the mean square error (MSE) is small. The total number of bits required for perfect reconstruction was 1.8 Mbits, which are nine times larger than 200 kbits. As shown in the example, a server can send a doctor 200 kbits of data that is sufficient to locate potential ROIs. Then, if a doctor finds ROIs then he or she can request more data in only the ROI, which would dramatically reduce the amount of data that must be transmitted. This example explains the significance of the region scalable coding approach.

In the rest of this paper, details of the proposed approach will be discussed. In section 2, we will discuss our approach to achieve region scalable coding. In section 3, we will discuss the performance of the proposed technique using our simulation results. We conclude with research results in section 4.

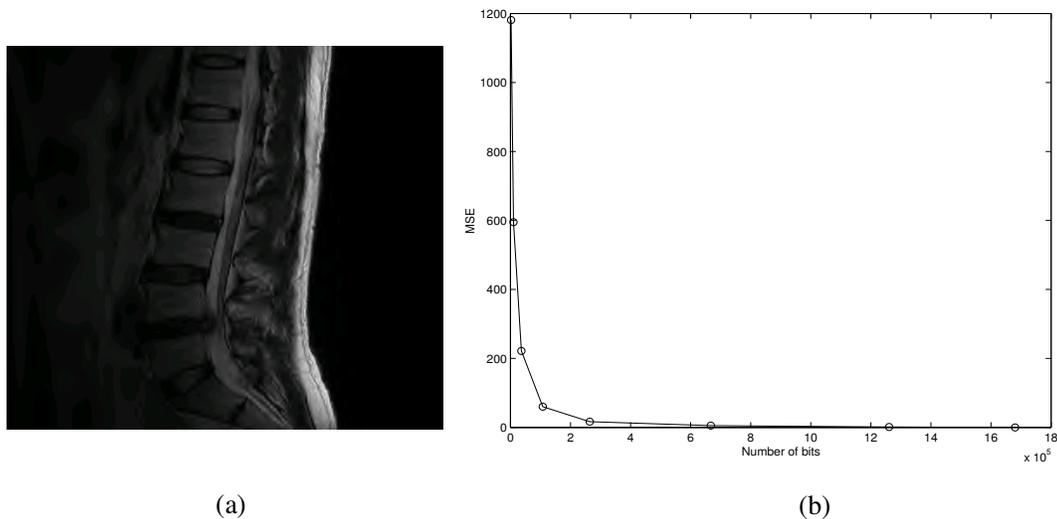


Figure 1. Number of bits versus MSE. (a) Spine image (b) Results of EZW compression

## 2. APPROACH

### 2.1. Scalable Coding

Scalable coding is capable of delivering the bit stream with a coarse reproduction that can be improved to a fine reproduction when it is needed. More information can be obtained with the transmission of additional bits. Figure 2 shows an example of scalable coding for the entire image using EZW algorithm with 9/7 filter bank. The initial threshold is given

$$T_0 = 2^{\lfloor \log_2 \max|y| \rfloor} \quad (1)$$

where  $y$  is wavelet coefficients and at each iteration the threshold is halved. This figure shows the reconstruction quality is getting better with each iteration. The peak to peak signal to noise ratio (PSNR) for the last iteration shows that this scheme can approach near lossless quality but perfect reconstruction is impossible. This is due to wavelet filter coefficients truncation to the finite precision.

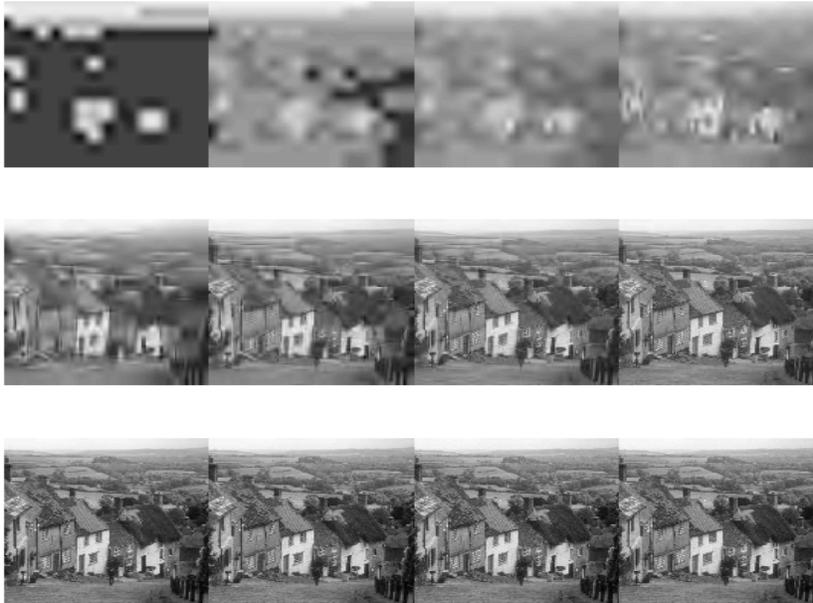


Figure 2. An Example of Scalable Coding for Gold Hill using EZW with 9/7. The corresponding PSNR are 9.23 dB, 18.53dB, 20.85dB, 21.96dB, 23.68dB, 25.85dB, 28.66dB, 32.55dB, 37.90dB, 44.53dB, 50.21dB, and 57.22dB from upper left to lower right.

### 2.2. Integer Wavelet Lifting

A still image compression standard (JPEG2000) is also based on the discrete wavelet transform (DWT). Most of the DWT-based codecs can produce an embedded bitstream. This means that the quality of the reconstructed image increases as more encoded bits become available to the decoder. The decoding can be stopped at any point of the bitstream. The main drawback of the DWT is that the wavelet coefficients are real numbers. In this case, efficient lossless coding is not possible using linear transforms. The lifting scheme (LS) presented by Sweldens [5], allows an efficient implementation of the DWT. Another of its properties is that perfect reconstruction is ensured by the structure of the LS itself. This allows new transformations to be used. One such

transformation is the integer wavelet transform (IWT) [6]. It is a basic modification of linear transforms, where each filter output is rounded to the nearest integer.

The LS can easily construct IWT from any general filter specification. Quantizers are inserted immediately after the calculation of the prediction and the updated terms but before modifying the odd or even sample value. The quantizers typically perform an operation such as truncation or rounding to the nearest integer, thus creating an integer value output. When it is extended to two dimensions, the resulting IWT is nonlinear. The order in which the transform is applied to the rows and the columns impact the final results. To recover the original sample values losslessly, the inverse transform must be applied in exactly the reverse row-column order of the forward transform. For example, converse the 5/3 filter bank into an IWT by adding two quantizers  $Q_{p1}(w) = -\lfloor -w \rfloor$  and  $Q_{u1}(w) = \lfloor w + 1/2 \rfloor$  to the prediction and update steps, respectively. The resulting forward transform is given by,

$$\begin{aligned} y(2n+1) &= x(2n+1) - \left\lfloor \frac{x(2n) + x(2n+2)}{2} \right\rfloor, \\ y(2n) &= x(2n) + \left\lfloor \frac{y(2n-1) + y(2n+1) + 2}{4} \right\rfloor \end{aligned} \quad (2)$$

The inverse transform, which losslessly recovers the original sample values, is given by,

$$\begin{aligned} x(2n) &= y(2n) - \left\lfloor \frac{y(2n-1) + y(2n+1) + 2}{4} \right\rfloor, \\ x(2n+1) &= y(2n+1) - \left\lfloor \frac{x(2n) + x(2n+2)}{2} \right\rfloor \end{aligned} \quad (3)$$

IWT can be used to produce a unified lossy and lossless codec. We use IWT to support lossless coding for medical images that strictly require the lossless property. For lossless coding it is necessary to make an invertible mapping from an integer image input to an integer wavelet representation.

The computational complexity of IWT is represented by the required number of addition and shift operations. Table 1 shows the required operation per two input samples for three representative integer wavelets. IWT has lower computational complexity than DWT. Its low computational complexity and efficient lossless compression performance is very useful for real time image transmission.

Table 1. Computational complexity

	Additions	Shifts	Total operations
S transform	2	1	3
5/3 integer lifting	5	2	7
9/7 integer lifting	9	3	12

### 2.3. Set Partitioning & Rearrangement

To achieve the proposed post-segmentation, the wavelet coefficients are partitioned into subsets. A subset consists of a pixel in the lowest sub-band and its children from a quad tree structure. Each set represents a small area corresponding to a block in the spatial domain. Figure 3 and Figure 4 explain how the wavelet coefficients are rearranged to make up an image that consists of sets. In Figure 3, The H means the horizontal head information and the number inside the circle indicates the location.

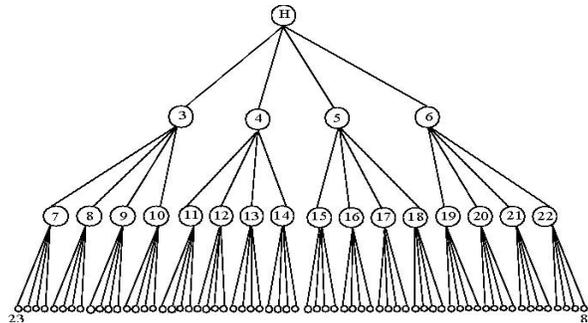


Figure 3. The quad tree structure of wavelet coefficients for horizontal direction

A set represents a local area corresponding to an  $m \times m$  block in the spatial domain. In Figure 4, a set is extracted from a wavelet image. In the figure,  $LHn$  represents the  $n$ th sub-band in horizontal direction. A set consists of one wavelet DC value and its children in horizontal, vertical, and diagonal directions. The elements in each sub-band are arranged by the raster scanning order. Assuming the four steps of the wavelet decomposition of the image of size  $256 \times 256$ , the wavelet coefficients would have 256 sets as shown in Figure 4. Each set will consist of a header and  $32 \times 32$  arrays of coefficients.

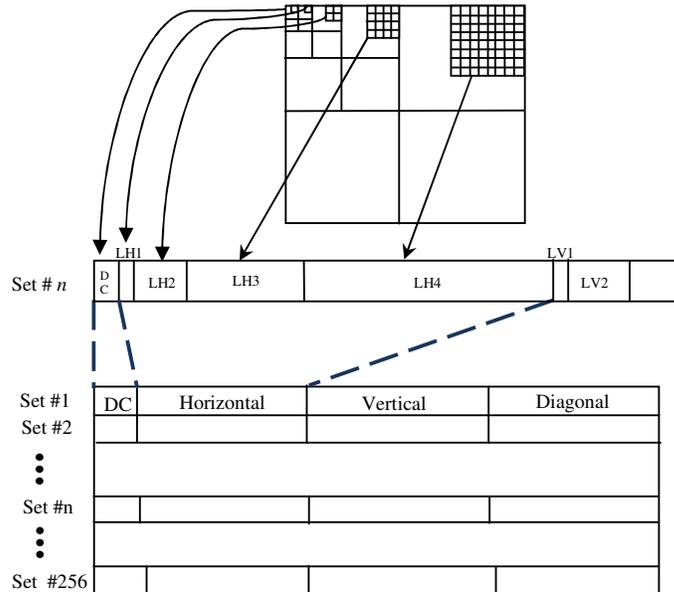


Figure 4. Set partitioning of the wavelet coefficients

## 2.4. Successive Quantizing and Layering

Scalable coding is mainly used in progressive transmitting scheme. It uses multiple quantizers for layering and creates a bit plane for each layer. The compressed image is expressed as an embedded bit stream comprising of a basic layer and many enhancement layers. The enhancement layers are refined to the quality of the image that has been reconstructed using the basic layer. EZW [4] and SPIHT [1] are two kinds of typical scalable coding algorithms based on DWT. The EZW algorithm uses zerotree to filter insignificant wavelet coefficients and only codes the significant wavelet coefficients. SPIHT is another scheme introduced by Said and Perlman that codes the wavelet coefficients by successive approximation. Similar with the EZW, there are two kinds of passes in each iteration time: sorting pass and refinement pass. In the refinement pass, you get a gradual approximation of all significant pixels.

Details of successive quantization and layering are given as follows. In each set, a unique quantizing threshold (QT) is used for quantization and layering. The quantizer Q1 is the quantizer with a center dead zone for the basic layer. In the basic layer, the wavelet coefficients are thresholded and quantized by Q1. In the enhancement layer, the coefficients are thresholded and quantized by Q2 with thresholds (T1, T2). The quantization occurs between T1 and T2 ( $T1 > T2$ ). Nonzero values outside [T1, T2] are set to zero in the enhancement layer. For further layering, the system requires a set of quantizers (QL, QL-1, ..., Q1) to produce one basic layer and (L - 1) enhancement layers. If we choose the step size for the threshold to be a power of two, then the quantization scheme would be similar to the successive approximation scheme used in the EZW algorithm.

The quantization step using QT leads to generating a bit plane for each layer in each set. The bit plane is a binary image, which contains positional information for the significant coefficients to be coded. In a bit plane, a significant bit is denoted '1' and an insignificant bit is denoted '0'. Run length entropy coding is applied to each set in the bit plane. We use three mark-up code symbols for efficient encoding. Those symbols are the head coefficient of the zero-tree (HOT), the end of block code (EOB) and the end of set code (EOS). HOT has the same effect as the zero-tree symbol in the EZW algorithm. EOB will be used to indicate a block boundary. A block represents a branch in the tree including an  $m$  by  $m$  block in the last sub-band ( $m > 2$ ). EOS indicates the boundary of a set. Each set represents a spatial location in the image and is coded independently. Header information includes markers, iteration number, and number of bits in each set. The overhead for the header information is negligible compared to the image data. When each set is transmitted, the header does not need to be sent since reconstruction can be done without the header information. Therefore the header information does not affect the network traffic.

The server transmits the image data progressively in layers. It divides the whole compressed image into several independent units and transmits one or more units during each time based on the requirement of the client. Figure 5(a) shows the structure of the transmitting data unit. The first two characters in the data unit are the starting identifier (ID) of a set. This is then followed by the serial number of the set and the maximal iteration time (L) of the set. After these numbers follows the length of the bytes of compressed data in the set. The real image binary data is at the end of the set. In order to access the data unit conveniently, the system adds a header in front of the real image data shown in Figure 5(b). Figure 5(c) shows how the real binary data organized after the head. From the above system structure, the server can handle any required data units for a post segmentation approach.

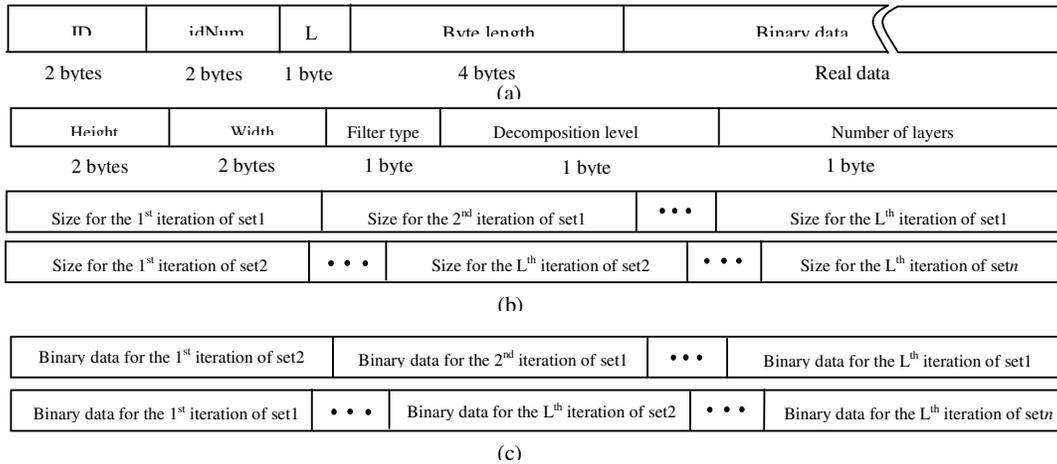


Figure 5. System Structure (a) structure of the data unit (b) header of the compressed data (c) real data arrangement

## 2.5. Post-Segmentation

In the pre-segmentation approach, ROI coding is accomplished by encoding the quantized wavelet coefficients corresponding to the ROI with a higher precision relative to the background, e.g., by scaling up the ROI coefficients or scaling down the background coefficients. In the post-segmentation approach, the ROI is selected in the spatial domain by specifying a convex polygonal area. The ROI in the spatial domain has to be converted to the corresponding areas in the transform domain since the scalable coding is performed in the transform domain. Figure 6 shows the relationship of ROI between the spatial domain and the wavelet domain.

Once a ROI in wavelet domain is identified, we then search indices of sets in the set partitioned image corresponding to the ROI. Any sets corresponding to the ROI that a client requests can easily be extracted and transmitted due to the efficient structure of the encoded data. In the client site, a decoder keeps log information that records transactions between the server and client.

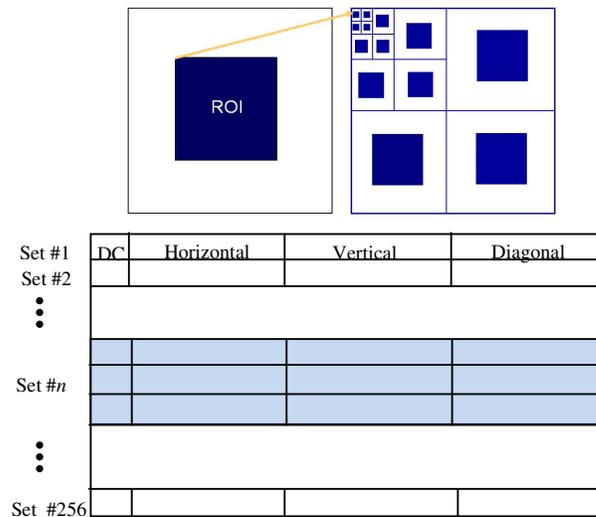


Figure 6. Relationship of ROI between the spatial domain and the wavelet domain (a) ROI in spatial domain (b) ROI in wavelet domain

### 3. RESULTS

We used lossless coding with integer wavelet transform for our method. We use lossless coding because of the strict requirements for telemedicine applications. However, when the image is decoded, only the ROIs are fully decoded. Therefore, we need to consider the properties of wavelets rather than the properties normally considered for image compression such as energy compaction. We evaluate three representative wavelets; Haar, 5/3, and 9/7 wavelets for integer wavelet lifting. An integer version of the orthogonal Haar transform is also known as the S transform. We apply the filters to an ultrasound image and MRI image. The length of the wavelet determines the region of support for the wavelet used. The effects of the transition of the data at the boundary can last for the full region of support. Therefore, we need to extend the boundary by an amount equal to the region of support for the wavelet filter used. Thus, the wavelet with a shorter region of support has an advantage. As shown in Table 2, the quality of the reconstructed images is perfect but the S transform shows the best compression performance.

Table 2. Comparison of coding performance

	Ultrasound (512x448, 8bpp)			MRI (512x512, 8bpp)		
	PSNR (dB)	Total bits (bits)	Bit rate (bpp)	PSNR (dB)	Total bits (bits)	Bit rate (bpp)
S transform	$\infty$	1097367	4.78	$\infty$	1569233	5.98
5/3 integer lifting	$\infty$	1186911	5.17	$\infty$	1673967	6.04
9/7 integer lifting	$\infty$	1220915	5.32	$\infty$	1680240	6.39

We now discuss the performance of our region scalable image compression (RSIC) method compared to other existing algorithms. We apply RSIC to the same images, and the results are summarized in Table 3.

Table 3. Comparison of PSNR after updating ROI

Area	Ultrasound		MRI	
	PSNR <sub>ROI</sub>	Total bits	PSNR <sub>ROI</sub>	Total bits
Background	28.12	212065	32.48	193843
ROI	$\infty$	532836	$\infty$	340129
Whole Image	$\infty$	1097367	$\infty$	1569233

RSIC saved three quarters of bits to encode a whole image losslessly. In our approach, the quality of the image within ROI is perfectly recovered while the background is reconstructed at a satisfying quality. Figure 7(a) shows the original ultrasound image and Figure 7(b) shows the reconstructed image with the basic quality. In Figure 7(c), the user specified ROI covering 15% of the image is shown. Figure 7(d) shows the reconstructed image after receiving maximum iterations for the ROI from a server that keeps the compressed image data. As can be seen in the figure, only the ROI has been enhanced. Figure 8 illustrates another example of using RSIC. Figure 8(a) shows the abnormal spine image, the area in the box contains abnormality of the spine. A doctor can identify the clinical problem from the image with basic quality that was reconstructed with a small number of bits. If a doctor needs to see a ROI with finer quality, then

the server will update only ROI in addition to the previously decoded image. Since this technique is post-segmentation based, the client can update multiple ROIs interactively. Figure 8(d) shows the updated image. From the Figure 8(d), we can clearly see that the enhanced ROI provides satisfactory results with a small increase in the number of bits.

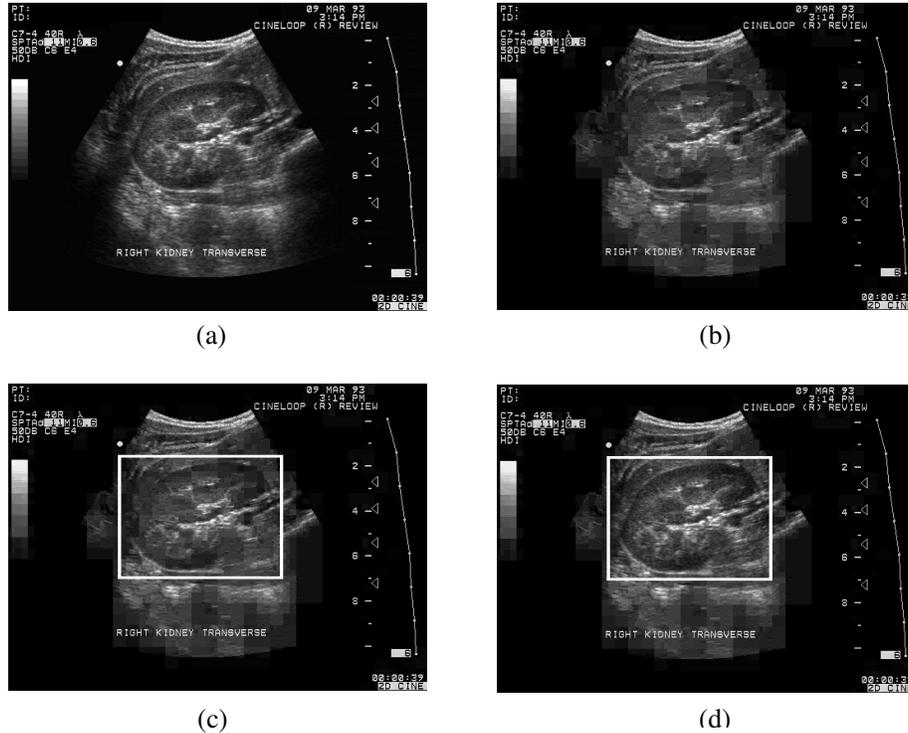


Figure 7. RSIC for Ultrasound (ROI: 15%) (a) original image (b) reconstructed image with basic quality (c) ROI selection from (b) (d) reconstructed image with updating ROI

Finally, Figure 9 compares the performance of the RSIC with the EZW and SPIHT methods. The solid line with squares and the dash-dotted line with circles show the results using the conventional method and the dotted line with circles shows the results using RSIC. The region of interest covers about 5 % of the whole image. As shown in Figure 9, RSIC saves many bits compared to EZW and SPIHT while keeping the image quality lossless in the ROI. RSIC saved a vast amount of bits compared to SPIHT at full resolution.

#### 4. CONCLUSION

The fully embedded scalability of the proposed coder allows us to transmit the basic layer with additional refinement data related only to a set or sets corresponding to the ROI in the image based upon a client's request. The proposed technique supports scalability for pre-defined ROI. It also allows a user to select ROI after the compression is done. The main contribution of this research is that it allows a client to select ROI at any time and update only the ROIs requested. This is a very important feature for medical image communications that satisfies both the requirement of quality for the ROI and compression ratio to make transmission over networks.

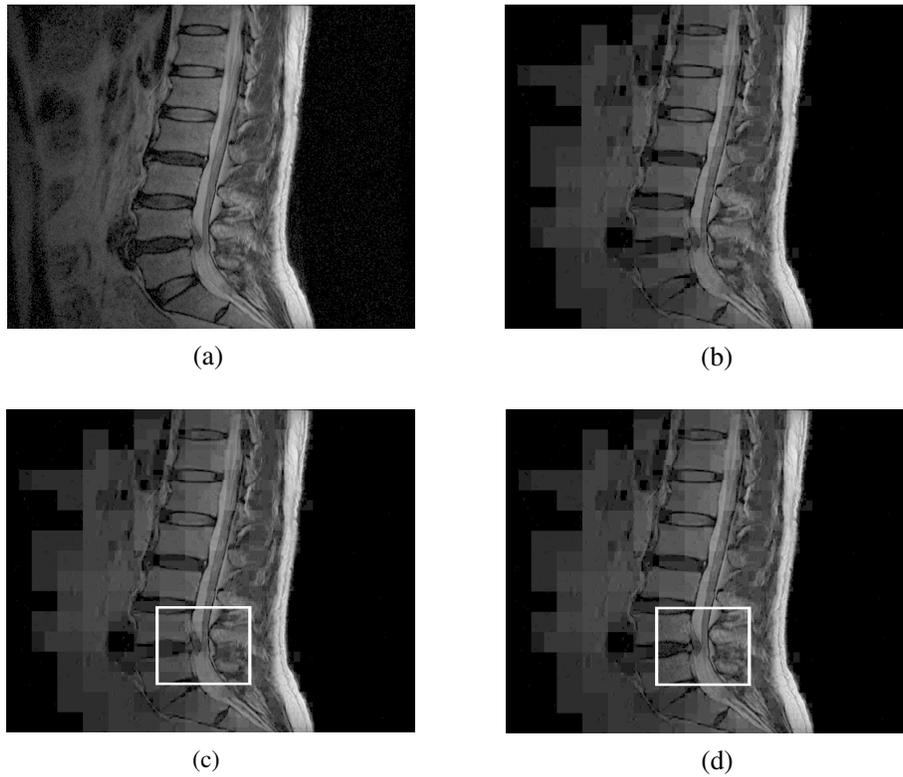


Figure 8. RSIC for MRI (ROI: 5%) (a) original image for a normal spine (b) original image for an abnormal spine (c) reconstructed image with basic quality (d) reconstructed image with ROI updated

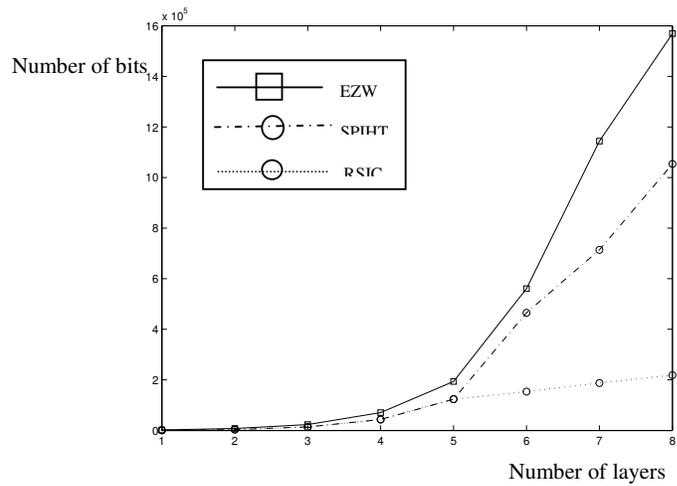


Figure 9. Comparison of EZW, SPIHT, and RSIC applied to a spine image

## REFERENCES

- [1] A. Said, W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, no. 3, pp. 243-250, 1996.
- [2] R. Sumalatha and M. V. Subramanyam, "Region based Coding of 3D magnetic Resonance Imaged for Telemedicine Applications," International Journal of Computer Applications, vol.5, no.12, August 2010.
- [3] Z. Wang and A. C. Bovik, "Bitplane-by-bitplane shift (BbBShift) – a suggestion for JPEG region of interest coding", IEEE Signal Processing Letters, Vol. 9, No. 5, May 2002.
- [4] J. Shapiro, "Embedded image coding using zero trees of wavelet coefficients", IEEE Transactions on Signal Processing, vol. 41, no. 12, pp. 3445-3462, 1993.
- [5] R. C. Calderbank, Ingrid Daubechies, Wim Sweldens, and Boon-Lock Yeo, "Lossless Image Compression using Integer to Integer Wavelet Transforms" International Conference on Image Processing, Vol. I, pp. 596-599, 1997.
- [6] Ali Bilgin, et al, "Scalable Image Coding Using Reversible Integer Wavelet Transforms," IEEE Transactions on Image Processing, vol. 9, no. 11, November 2000.