A Review of Region-of-Interest Coding Techniques of JPEG2000

Narvinder Kaur
University Institute of Engineering and Technology, Panjab University,
Chandigarh, India.

ABSTRACT
JPEG2000 is an international standard for still image coding and provides an extensive set of features vital to many high-end and emerging applications. The default performance metric for JPEG2000 is an optimisation of objective quality versus bit-rate. However, for high compression applications, it would be more appropriate that regions of interest are prioritised for interpretability. JPEG2000 allows for this in its region of interest (ROI) feature. JPEG2000 provides many different ROI coding mechanisms, namely the general scaling method, max-shift method, bitplane-by-bitplane shift method (BbBShift), partial significant bit-plane shift method (PSBShift) and ROITCOP (ROI coding through component priority) method. These method have advantages and disadvantages relative to one another and the choice of methods for ROI coding is very much dependant on the requirements of the application at hand.

Keywords: **JPEG2000, ROI, BbBShift, PSBShift, ROITCOP**

1. INTRODUCTION
As digital imagery becomes more commonplace and of higher quality, there is need to manipulate more and more data. Thus image compression must not only reduce the necessary storage and bandwidth requirements, but also allow extraction for editing, processing, and targeting particular devices and application. The JPEG2000 image compression system has a rate-distortion advantage. More importantly, it also allows extraction of different resolutions, pixel fidelities, regions of interest, components, and more, all from a single compressed bitstream. This allows an application to manipulate or transmit only the essential information for any target device from any JPEG2000 compressed source image [12]. The JPEG2000 standard exhibits a lot of nice features, most significant being the possibility to define Region-Of-Interest(ROI) in an image, the spatial and SNR(quality) scalability, the error resilience and possibility of intellectual property rights protection. From all the above features, the functionality of ROI is important in applications where certain parts of image are of higher importance than others. In such a case, these regions need to be encoded at higher quality than the background. During the transmission of image, these regions need to be transmitted first or at a higher priority (for example during progressive transmission) [2]. This paper will provide useful information of about various ROI coding methods of JPEG2000. A brief overview of JPEG2000 coding method is presented in section 2. Section 3 provides information about various ROI coding methods of JPEG2000. Each method has advantages and disadvantages relative to one another and choice of methods for ROI coding depends upon the requirements of the application. Finally, the paper will draw conclusions in section 4.

2. THE OVERVIEW OF JPEG2000 IMAGE CODING METHOD
With the increasing use of multimedia technologies, image compression requires higher performance as well as new features. To address this need in the specific area of still image encoding, a new standard is currently being developed, the JPEG2000 [2]. The JPEG2000 standard was intended to create a new image coding system for different types of still images (bi-level, gray-level, color, multi-component), with different characteristics (natural images, scientific, medical, remote sensing, text, rendered graphics, etc) allowing different imaging models (client server, real-time transmission, image library archival, limited buffer and bandwidth resources, etc) preferably within a unified system.

The JPEG2000 compression engine is illustrated in block diagram form in figure 1 [2]. At the encoder, the discrete transform is first applied on the source image data. The transform coefficients are then quantized and entropy coded before forming the output code stream. The decoder is reverse of encoder. The code stream is first entropy decoded, dequantized, and inverse discrete transformed, thus resulting in the reconstructed image data. Although this general block diagram looks like the one for the conventional JPEG, there are radical differences in all of the processes of each block of the diagram [2]. The whole image engine has been decomposed into three parts: the preprocessing, core processing, and bit-stream, formation parts, although there exist high inter-relation between them. In the preprocessing part the image tiling, the dc-level shifting and the component transformations are included. The core processing part consists of the discrete transform, quantization and entropy coding processes. Finally, the concepts of the precincts, code blocks, layers, and packets are included in the bit-stream formation part. It should be noted here that the basic encoding engine of JPEG2000 is based on EBCOT (Embedded Block Coding with Optimized Truncation of the embedded bitstream) algorithm which is described in detail in [11].

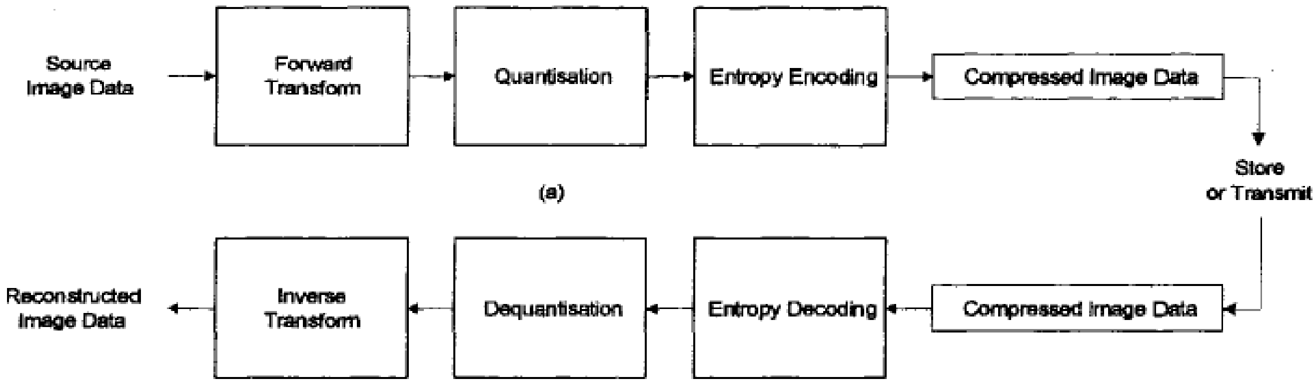

Figure 1. Block diagrams of the JPEG2000 (a) encoder and (b) decoder.

3. ROI CODING METHODS OF JPEG2000

Region Of Interest(ROI) coding is a prominent feature of some image coding systems aimed to prioritize specific areas of the image through the construction of a codestream that, decoded at increasing bit-rates, recovers the ROI first and with higher quality then rest of the image. JPEG2000 is a wavelet-based coding system that is supported in the Digital Imaging and Communications in Medicine (DICOM) standard [7]. Among other features, JPEG2000 provides lossy-to-lossless compression and ROI coding, which are especially relevant to medical community. In this section, different ROI coding methods are discussed.

3.1 General coding based method

The very first method of the ROI coding in JPEG2000 is the scaling method of ROI coding where the bits representing the wavelet coefficients contributing to the ROI region, are shifted upward by a user-defined value [1] as shown in Figure 2, [3]. The general scaling based method [3] places ROI associated bits in the higher bitplanes by scaling the bitplanes of ROI coefficients up, so that ROI coefficients can be coded first in the embedded bitplane coding. This method allows the use of arbitrary scaling value, so allows fine control on the relative importance between ROI and BG [10]. The scaling method has two major drawbacks. First, it needs to encode and transmit the shape information of the ROIs. This rapidly increases the algorithm complexity. Second, if arbitrary ROI shapes are desired, the shape coding will consume a large number of bits, which significantly decreases the overall coding efficiency [9]. Further, the general scaling based method requires the generation of an ROI mask and the distinction of ROI/BG coefficients at both encoder and decoder sides. This increases decoder complexity and processing overhead. Thus to simplify the decoding problem, only rectangular and elliptical ROI shapes are defined in the standard [3].

After ROI coding, the more important ROI coefficients are in the most significant bit-planes and are thus, fed into bit-plane arithmetic block coder prior to the less important background coefficients. The ROI coefficients are then placed in the very beginning of the code-stream and are transmitted and decoded first.

3.2 Max-shift based coding method

In the max-shift method, the wavelet transform is applied to the original image at the encoder and the resulting sub-band coefficient bits not associated with the ROI (i.e. background (BG)) are downshifted below those belonging to the ROI as shown in Figure 3, [4] This has the effect of scaling coefficients, which are involved in reconstructing the ROI, by 2^{Rshift} for some up-shift value, *Rshift*, which is signaled in the codestream to the decoder. The scale factor is a power of 2 so that the scaling process is equivalent to left shifting the magnitude bit-planes by *Rshift*. A valid up-shift value is a value that ensures no overlap between the ROI and BG bit-planes. To achieve this encoder must select

$$Rshift \geq \max (M_b) \qquad (1)$$

Where *Mb* is the largest number of magnitude bit-planes for any background coefficient for sub-band *b*. This ensures that the decoder can decide whether or not a coefficient belongs to the ROI or BG by comparing the number of decoded bits of the current coefficient with *Mb*; non-ROI coefficients are scaled up to their original bit-planes before the inverse wavelet transform is applied. The downshifting of BG coefficient bits towards the least significant bit-planes isolates ROI bits in the most significant bit-planes. Therefore, during the embedded bit-plane coding process, which encodes bit-planes from the most significant bit-plane to the least significant bit-plane, all ROI coefficient bits will be encoded and placed in the code-stream before those belonging to the BG. The decoding of the code-stream will thus improve the ROI rapidly with little change in the background, until all ROI information has been received. No flexibility exists in controlling the degree of importance between the ROI and the BG.

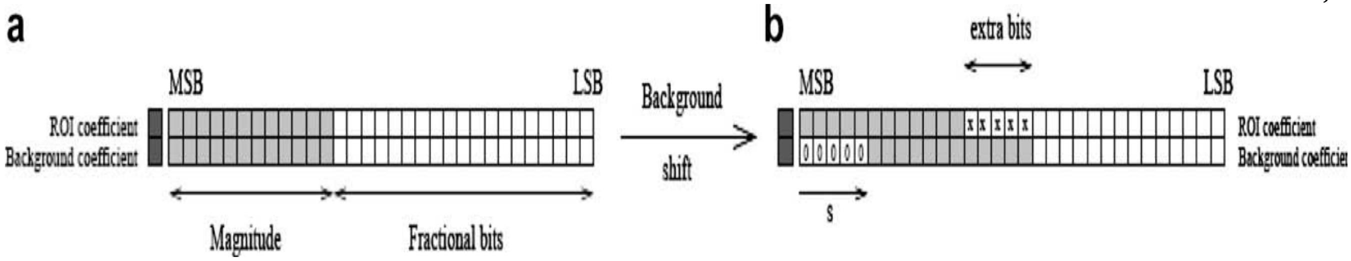

Figure 2. ROI scaling operation where's' is the scaling value: (a) no scaling and (b) with scaling's'.

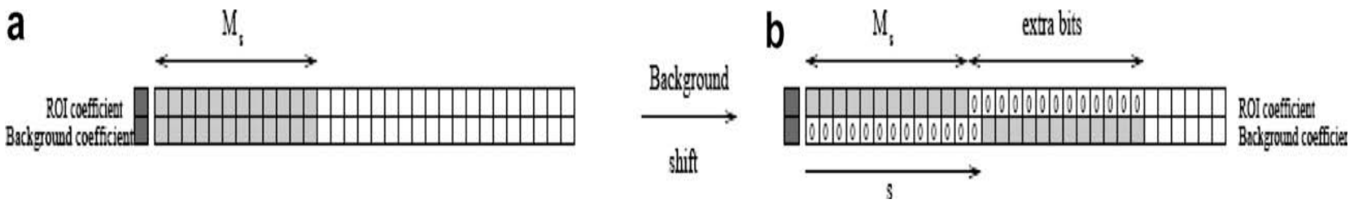

Figure 3. ROI Maxshift method where's' is scaling of the ROI coefficients (a) no scaling and (b) with scaling's'.

3.3 Bit-plane by bit-plane shift coding method

In bitplane-by-bitplane shift (BbBShift) method, instead of shifting the bitplanes all at once by the same scaling value as in maxshift, BbBShift shifts them on a bitplane-by-bitplane basis. An illustration of the BbBShift method is shown in Figure 5, [5]. Two parameters, s1 and s2, are used in BbBShift. The sum of s1 and s2 must be equal to the largest number of magnitude bitplanes for any ROI coefficient. In this paper, we index the top bitplane as bitplane 1, the next to top as bitplane 2, and so on.

At the encoder, the bitplane shifting scheme is as follows:

1) For any bitplane of an ROI coefficient:
   If $b \leq s1$, no shift;
   If $s1 < b \leq s1+s2$, shift it down to bitplane $s1+2(b-s1)$.
2) For any bitplane b of a BG coefficient:
   If $b \leq s2$, shift it down to bitplane $s1+2b-1$; If $b > s2$, shift it down to bitplane $s1+s2+b$.

At the decoder, for any given nonzero wavelet coefficient, the first step is to identify whether it is an ROI coefficient or a BG coefficient. This can be done by examining the bitplane level of its most significant bit (MSB). The set of ROI associated bitplanes is given by:

$$B_{roi} = \{b \backslash b \leq s1 \text{ or } b = s1+2k,\ k=1, 2...s2\} \qquad (2)$$

If the wavelet coefficient's MSB is at bitplane b € B_{roi}, then it must be an ROI coefficient. Otherwise, it is a BG coefficient. The bitplanes are then shifted back to their original levels by reversing the bitplane shifting scheme in the encoder.

In comparison with the general scaling based methods, where only rectangle and ellipse ROI shapes are allowed, the BbBShift method supports arbitrary shaped ROI coding. BbBShift also bears the same advantage of maxshift that different wavelet subbands can have different ROI definitions. In BbBShift, if we set s2=0 and choose s2 according to (1), then BbBShift is equivalent to maxshift. In other words, maxshift is a special case of BbBShift. Compared with maxshift, BbBShift has the flexibility to have an arbitrary scaling value to adjust the relative importance between ROI and BG coefficients. This flexibility may lead to improved quality of ROI coding, depending on the application.

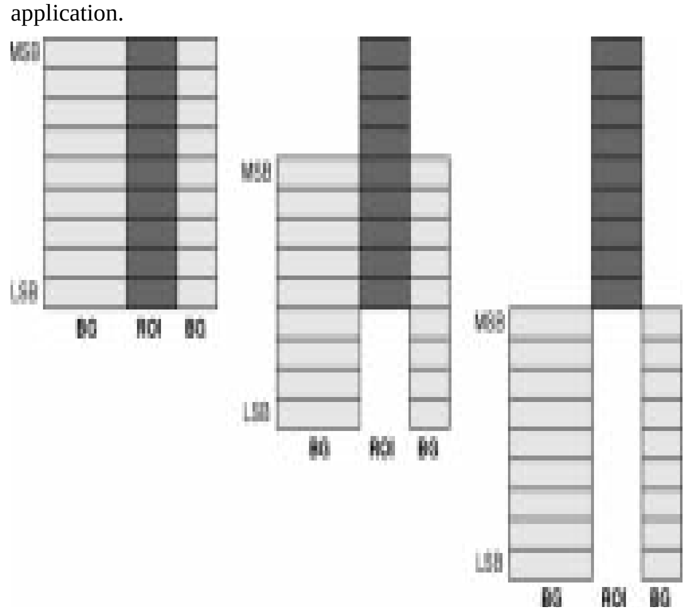

Figure 4. ROI coding methods in JPEG2000 (bitplanes are represented by the gray bars). (a) No ROI coding, no scaling; (b) scaling based method, s = 4; and (c) maxshift method, s = 9.

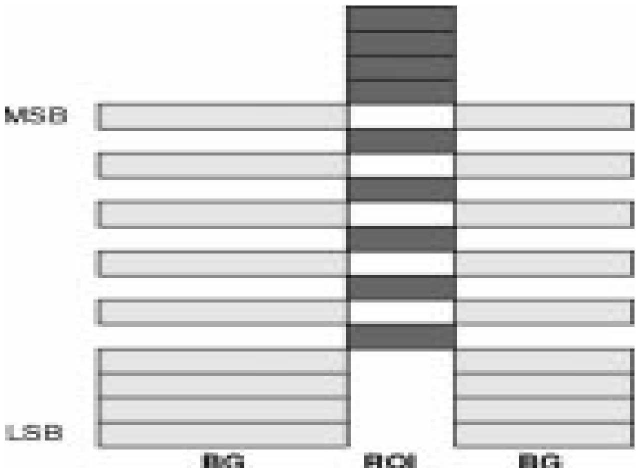

Figure 5. BbBShift methods with s = 4and s = 5(Bitplanes are represented by the gray bars).

It is not necessary for the BbBShift method to have a shape coding component, which is essential in the general scaling based methods. Similar to the general scaling based method and the maxshift method, the coding efficiency of BbBShift decreases in comparison with JPEG2000 without any ROI coding. It needs to be mentioned that the proposed BbBShift method is not compatible with the current JPEG2000 ROI coding definitions, in which only maxshift and rectangle and ellipse shape scaling based ROI coding are defined. In order to use BbBShift, a new ROI coding mode must be added to the standard [5].

3.4 Partial significant bit-planes shift

A new and flexible scaling-based method called *partial significant bitplanes shift* (PSBShift) [6], which can combine the advantages of the two standard methods and efficiently compress multiple ROIs with different degrees of interest. This method is mainly based on the facts that at low bit rates, ROI in an image is desired to sustain higher quality than BG, while at the high bit rates, both ROI and BG can be coded with high quality and the difference between them is not very noticeable. So we can just isolate a certain number of bitplanes of ROI in the MSBs to adjust the importance between ROI and BG. That means we only need to shift part of the MSBs of ROI coefficients instead of shifting the whole bitplane as the standard methods do. Thereby, we name this method as PSBShift. An illustration of the PSBShift method is shown in Figure 6(d) [6] (where s = 6). The whole bitplanes of ROI coefficients are divided into two parts: *the MSBs* and *the residual significant bitplanes*. The number of the MSBs is the same as the scaling value, and the number of the residual significant bitplanes N_{lsbs} is represented as

$$N_{\text{lsbs}} = \begin{cases} M_b - s, & \text{if } s \le M_b \\ 0, & \text{if } s > M_b. \end{cases}$$

At the encoder, the MSBs of ROI coefficients are not shifted, while the residual significant bitplanes of ROI coefficients are down-shifted toward LSB with BG coefficients. At the decoder, ROI coefficients can be identified in the same way as maxshift. All bits lower than the sth bitplane are scaled up s bitplanes, and combined with the bits higher than the sth bitplane. It is a little different from the maxshift decoding method, in which no combination operation is included. Since the bits above the sth bitplane of BG coefficients are all zeros, this modified decoder can also handle the standard code stream generated by the maxshift method. The PSBShift method can code ROI in an image with higher or the same quality as BG.

PSBShift JPEG2000 ROI coding method that has four primary advantages: 1) it supports arbitrarily shaped ROI coding without coding the shape; 2) it allows different wavelet subbands to have different ROI definitions; 3) it can control the relative importance between ROIs and BG by using appropriate scaling values; and 4) the new method can efficiently code multiple ROIs with different priorities in an image at the low bit rates [6].

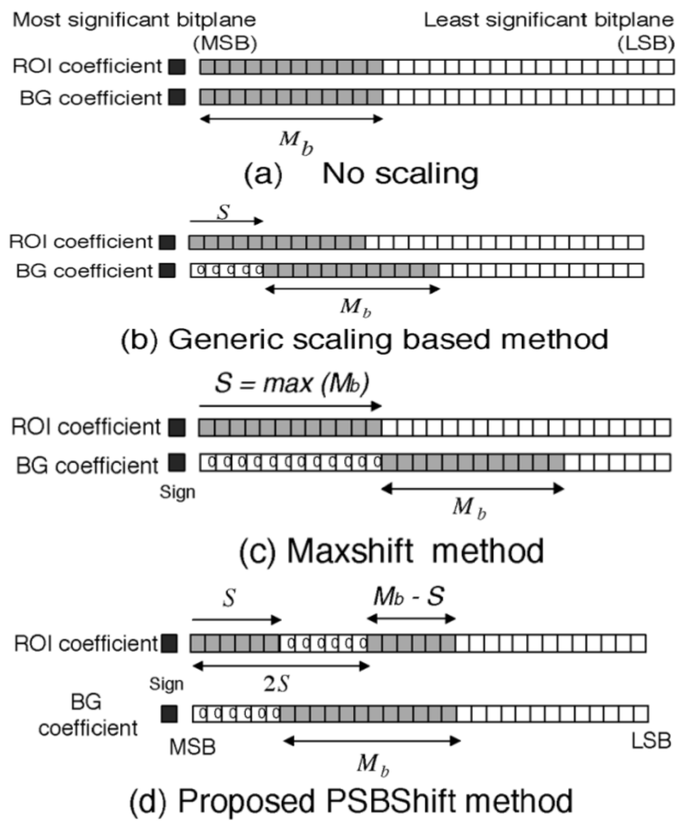

Figure 6. Standard ROI coding methods and the PSBShift method, where bitplanes are represented by the gray blocks, and black blocks represent the sign of coefficients.

3.5 ROI coding through component priority

Two relevant features of JPEG2000 to this method are: support for multi-component images, and component scalability. Component scalability stands for the ability of the coding system to allow the access and manipulation of components in the compressed domain without needing to decompress the image.

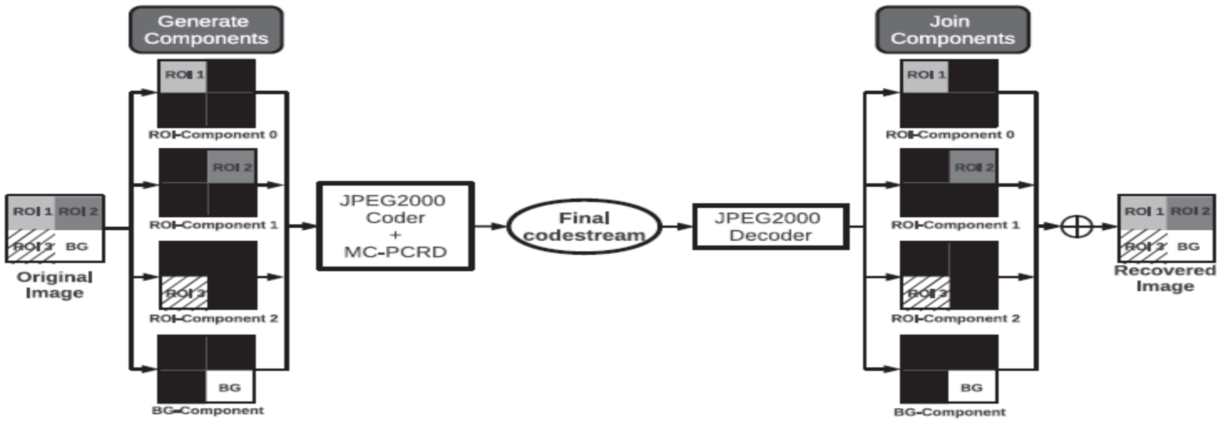

Figure 7. Show Operations for the ROI coding method. Two operations are added in coder/decoder pipeline: generate components and join components.

These features are employed by ROI coding Through Component Priority (ROITCOP) [7] to allocate each ROI in a component where the non-ROI area is set to zero. Then, through the use of rate-distortion optimization techniques these components are prioritized at desired priorities, generating a multi-component image with each ROI prioritized at will. To apply ROITCOP, the JPEG2000 core coding system requires two additional operations in the coding pipeline, and a slight modification of the PCRD method. We note that the ROITCOP requires a JPEG2000 encoder that implements some rate-distortion optimization method. Figure 7 depicts these two operations, called generate components, and join components. Generate components is an operation carried out in the encoder that defines as many components as ROIs have the image (referred to as ROI-components), plus one component for the background (referred to as BG-component). The operation Join Components sets the magnitude of each ROI coefficient to that recovered at the ROI-component with highest priority containing that ROI coefficient. The magnitude of BG coefficients is set to that recovered at the BG-component. For these Multi-Component (MC) images, a MC-PCRD is applied to combine the bitstreams from all components, minimizing the overall distortion. To correctly prioritize the desired ROI-components, a modification to MC-PCRD is introduced, Updating the distortion estimates for specific codeblocks and components according to

$$D'^{n_j}_{c,i} = \begin{cases} \mathcal{U}_c * D^{n_j}_{c,i} & \text{if } c \in ROI \\ D^{n_j}_{c,i} & \text{otherwise} \end{cases} \quad (3)$$

Where D_{cj}^{mj} denotes the distortion of component c for codeblock i at truncation point n_j, and U_c denote the priority for ROI-component c. This operation modifies the distortion estimation for all ROI-components. Note that through Eq. (3), rate-distortion estimates for coding passes are modified at codeblock level. The main difference with previous methods is that in ROITCOP codeblocks contain only ROI or BG coefficients, whereas in previous approaches codeblocks may contain both types of coefficients. With the use of ROITCOP method [7], perfect fine-grain accuracy is achieved.

The key feature of this method is to allocate each ROI in a component and set coefficients of the non-ROI area of that component to zero. Then a modification of the distortion estimation is performed to prioritize the specific ROI within each component. Main advantages of this method are: (1) it avoids the dynamic range problem of the decoder; (2) it achieves very high fine-grain accuracy, comparable to that achieved by ROI coding methods based on modifying wavelet coefficients; (3) it is able to decode the ROI and the background in a lossy-to-lossless mode; (4) it enables the definition of multiple ROIs with different degrees of priority; (5) it is able to exclusively recover the desired ROI- simulating the MaxShift method – through the component scalability, and; (6) it is compliant with JPEG2000 standard [7].

4. CONCLUSION

We have provided an overview of state-of-the-art ROI coding techniques applied to medical images. ROI coding preserves image quality in diagnostically critical regions by performing advanced image compression, enabling better image examination and addressing issues regarding image handling and transmission in telemedicine systems. Therefore, ROI coding is considered quite important in distributed and networked electronic healthcare. Each of these schemes finds use in different application owing to their unique characteristics. Though there a number of coding schemes available, the need for improved performance and wide commercial usage, demand newer and better techniques to be developed.

5. ACKNOWLEDGMENTS

I would like to express my gratitude to Dr. Savita Gupta (prof. U.I.E.T, panjab university chd). Her guidance and supervision has been very important to me. Her guidance and kindness are unquestioning.

Finally, I would like to thank all my family and friends who have been supporting me all the time.

6. REFERENCES

[1] C.A. Christopoulos, A.N. skodras, and T.Ebrahimi, "JPEG2000 still image coding system: An overview, "IEEE trans. Consumer Electron, vol. 46, pp.1103-1127, Nov.2000

[2] M.A. Ansari, R.S. Anand,"Context based medical image compression for ultrasound images with contextual set partitioning in hierarchical trees algorithm", Advances in Engineering Software vol-40, pp-487–496, 2009

[3] A. Nguyen, V. chandran, S. Sridharan, R. Prandolini," Interpretability Performance Assessment of JPEG2000 and Part 1 Compliant Region of Interest Coding vol. 49, pp-808-817, 2003

[4] Z. Wang and A.C. Bovik, "bitplane-by-bitplane shift (BbBShift) - a suggestion for JPEG2000 region of interest coding," *IEEE Signal Processing Letters*, vol. 9, no.5, pp-160-162, May, 2002

[5] L. Liu and G. Fan, "A new JPEG2000 region-of-interest image coding method: partial significant bitplane shift," IEEE Signal Processing Letters, vol.10, no.2, pp-35-38, februrary, 2003

[6] Joan Bartrina-Rapesta , Joan Serra-Sagristà, Francesc Aulí-Llinàs, "JPEG2000 ROI coding through component priority for digital mammography", Computer Vision and Image Understanding 115 2011, pp-59–68

[7] Yang H, Long M, Tai HM.," Region of interest image coding based on EBCOT", IEEE proc Visual image signal process 2005;152(5)

[8] Christopoulos C, Askelof j, Larsson M.,"Efficient methods for encoding of regions of interest in upcoming jpeg 2000 still image coding standard", IEEE Signal Process Lett 2000; 7(9):pp-247-249

[9] Zhang Li-bao, Yu Xian-chuan, Multiple regions of interest image coding using compensation scheme and alternating shift, in: the 18th IEEE computer society international conference on pattern recognition(ICPR'06), vol-3,2006.p. 758-61

[10] D. Taubman, "High Performance Scalable Image Compression With EBCOT", *Proc. IEEE Int.* Conference Image Processing, Vo1.111, pp. 344-348, Kobe. Japan, October 1999.

[11] Michael W. Marcellin, Michael J. Gormish, Ali Bilgin, Martin P.Boliek, "An overview of JPEG2000", *Proc .of IEEE Data Compression Conference*, pp-523-541

[12] C. Doukas, I. Maglogiannis, "Region of Interest coding techniques for medical image compression", *IEEE Engineering in Medicine and Biology Magazine* 26(5) (227) 29-35.