ISSN: 2582-8118 ***Volume 3, Issue 2; March 2023***

INTERNATIONAL JOURNAL OF RESEARCH AND ANALYSIS IN SCIENCE AND ENGINEERING

Web: https://www.iarj.in/index.php/ijrase/index

4. Set Partitioning Embedded Block Coder Algorithm

Avadhesh Kumar Maurya

Assitant Professor,
Department of Electronics and Communication Engineering,
Lucknow Institute of Technology,
Lucknow, U.P., India.

Vishwa Nath Maurya

Executive Vice-Chancellor and Distinguished Professor,
Chartered International Da Vinci University,
Delaware, USA.

Rakesh Kumar Singh

Professor and Ex Director,
Bipin Tripathi Kumaon Institute of Technology (Autonomous),
Affiliated to Uttarakhand Technical University,
Dehradun, India.

ABSTRACT

Present analysis focuses to design of efficient compression of multidimensional, multispectral and volumetric digital images. The digital representation of images and videos allows processing and archiving tasks to be integrated in multimedia platforms, computing and communications. The efficient representation of visual information is at the centre of image compression systems. The efficiency of a representation refers to the capture of significant information about an object of interest in a smaller description. Image compression techniques have numerous applications in the field of multimedia broadcasting channels, video conferencing, medical imaging and neural networks. An increasing demand for multimedia content such as digital images and video has led to great interest in research into compression techniques. The development of higher quality and less expensive image acquisition devices has produced steady increases in both image size and resolution, and a greater consequent for the design of efficient compression systems. Although storage capacity and transfer bandwidth has grown accordingly in recent years, many applications still require compression. In general, present paper investigates still image compression in the transform domain. The main objective is to design a compression system suitable for processing, storage and transmission, as well as providing acceptable computational complexity suitable for practical implementation. The basic rule of compression is to reduce *the numbers of bits needed to represent an image. SPECK use sets in the form of blocks of contiguous coefficients to exploit similarities within the sub bands, whereas the SPIHT algorithm uses spatial trees which span and exploit the similarity across different sub bands of wavelet decomposition. SPECK comprises of three stages initialization, sorting, and refinement, and uses two linked lists: list of insignificant sets (LIS) and list of significant pixels (LSP).*

KEYWORDS:

Image compression algorithms, SPECK Coding method, Discrete Fourier Transform (DFT), Hadamard-Haar Transform (HHT), Wavelet Transforms (WT)

1. Introduction:

The digital representation of images and videos allows processing and archiving tasks to be integrated in multimedia platforms, computing and communications. The increasing demand for multimedia content such as digital images and video has led to great interest in research into compression techniques. The development of higher quality and less expensive image acquisition devices has produced steady increases in both image size and resolution, and a greater consequent for the design of efficient compression systems. Although storage capacity and transfer bandwidth has grown accordingly in recent years, many applications still require compression. In general, present paper investigates still image compression in the transform domain. Present analysis focuses multidimensional, multispectral and volumetric digital images. The main objective is to design a compression system suitable for processing, storage and transmission, as well as providing acceptable computational complexity suitable for practical implementation. The basic rule of compression is to reduce the numbers of bits needed to represent an image. SPECK use sets in the form of blocks of contiguous coefficients to exploit similarities within the sub ands, whereas the SPIHT algorithm uses spatial trees which span and exploit the similarity across different sub bands of wavelet decomposition. SPECK comprises of three stages initialization, sorting, and refinement, and uses two linked lists: list of insignificant sets (LIS) and list of significant pixels (LSP). The coding algorithm proceeds as follows. In the initialization stage, initial threshold, which depends on the maximum value in the wavelet coefficients pyramid, is defined and the list LSP is initialized empty. Then, the transformed image is partitioned into two sets: a set of type S which is the root of the pyramid, and a set of type I which is the remaining part. LIS is initialized with type S blocks. A significant set S is partitioned by using quad partitioning scheme resulting in four equal sets being added to LIS. A set of size 1x1 is a coefficient, and when a coefficient is found significant for the first time, corresponding sign bit is also coded and coefficient is sent to LSP. After the processing of type S sets, type I set is processed. A significant I set is partitioned by octave band partitioning resulting in three type S sets of the size equal to chopped portion of I and remaining portion as new type I. Once the set I is partitioned, the three S sets are processed in the regular image-scanning order. Once all the sets in LIS have been processed for a particular threshold, the refinement pass is initiated which refines the quantization of the pixels in the LSP, i.e., those pixels that had tested significant during the previous sorting passes. The threshold is then reduced by a factor of two and the sequence of sorting and refinement passes is repeated for sets in the LIS against this lower threshold.

For processing the sets in LIS, the sets are processed in increasing order of their size. The whole process is repeated until the desired rate is achieved.

2. Overview of Speck Algorithm

The main task of image compression algorithms is reduction of redundant and irrelevant information. In the present scenario, SPECK Coding method of compressing still images exist. In any image compression scheme, three basic steps are involved:

Transformation, Quantization and Encoding. Transformations reduce the numerical size of the data items that allow them to represent by fewer binary bits. The technical name given to these methods of transformation is mapping Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Walsh-Hadamard Transform (WHT), Hadamard-Haar Transform (HHT), Karhune-Loeve Transforms (KLT), Slant-Haar Transform (SHT), Short Fourier Transforms (SFT), and Wavelet Transforms (WT). The procedure of approximating the continuous set of values in the image data with a finite, preferably small set of values is called quantization. The original data is the input to a quantizer and the output is always one among a limited number of levels.

The quantization step may also be followed by the process of thresholding. Encoding process reduces the overall number of bits required to represent the image. SPECK use sets in the form of blocks of contiguous coefficients to exploit similarities within the sub ands, whereas the SPIHT algorithm uses spatial trees which span and exploit the similarity across different sub bands of wavelet decomposition. SPECK comprises of three stages initialization, sorting, and refinement, and uses two linked lists: list of insignificant sets (LIS) and list of significant pixels (LSP).

The coding algorithm proceeds as follows. In the initialization stage, initial threshold, which depends on the maximum value in the wavelet coefficients pyramid, is defined and the list LSP is initialized empty. Then, the transformed image is partitioned into two sets: a set of type S which is the root of the pyramid, and a set of type I which is the remaining part. LIS is initialized with type S blocks. A significant set S is partitioned by using quad partitioning scheme resulting in four equal sets being added to LIS. A set of size 1x1 is a coefficient, and when a coefficient is found significant for the first time, corresponding sign bit is also coded and coefficient is sent to LSP. After the processing of type S sets, type I set is processed. A significant I set is partitioned by octave band partitioning resulting in three type S sets of the size equal to chopped portion of I and remaining portion as new type I. Once the set I is partitioned, the three S sets are processed in the regular image-scanning order. Once all the sets in LIS have been processed for a particular threshold, the refinement pass is initiated which refines the quantization of the pixels in the LSP, i.e., those pixels that had tested significant during the previous sorting passes. The threshold is then reduced by a factor of two and the sequence of sorting and refinement passes is repeated for sets in the LIS against this lower threshold. For processing the sets in LIS, the sets are processed in increasing order of their size. The whole process is repeated until the desired rate is achieved. Numerous previous noteworthy researchers [1-10] have confined their studies in the relevant field. However, here we specifically mention research works of [1,2,3,8.9.10] wherever required.

3. Speck State Recording of Block Sets

In SPECK, the blocks are recursively and adaptively partitioned such that high energy areas are grouped together into small sets whereas low energy areas are grouped together in large sets. This algorithm makes use of the adaptive quad tree splitting to zoom into high energy areas within a region to code them with minimum significance maps [3, 8]. The algorithm includes encoder and decoder, which implements initialization, sorting pass, refinement pass & quantization steps [9].

Threshold selection and Pixel significance in an entire set (T) of pixels are carried out using equation the algorithm makes use of rectangular regions of image. These regions or sets are called a set of type S. The dimension of a set S depends on the dimension of the original image and the sub band level of the pyramidal structure at which the set lies.

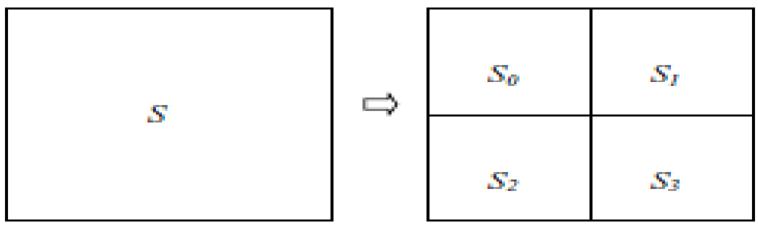


Figure 3.1: Partitioning of stet *S*. [9]

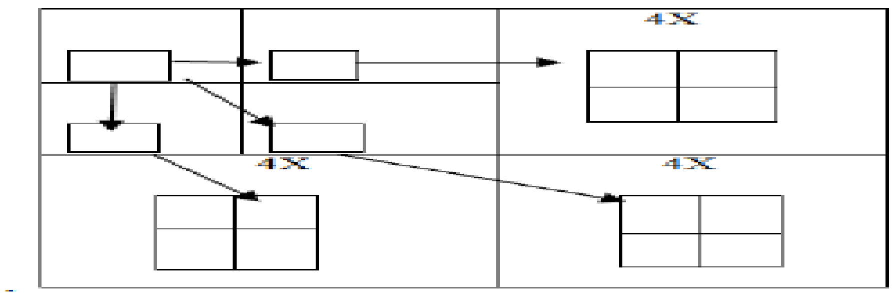


Figure 3.2 Parent Offspring Dependencies in Tree-Based Organization in Wavelet Transform [9]

During the course of the algorithm, sets of various sizes will be formed, depending on the characteristics of pixels in the original set. Note that a set of size 1consists of just one pixel. The other types of sets used in the SPECK algorithm are referred to as sets of type I. The algorithm maintains two linked lists: LIS - List of Insignificant Sets, and LSP - List of Significant Pixels.

The LIS contains sets of type S of varying sizes, which have not found significant against a threshold n while LSP obviously contains those pixels that have tested significant against n. Use of multiple lists will speed up the encoding/decoding process. Following flow chart describes the algorithm [1,2,10].

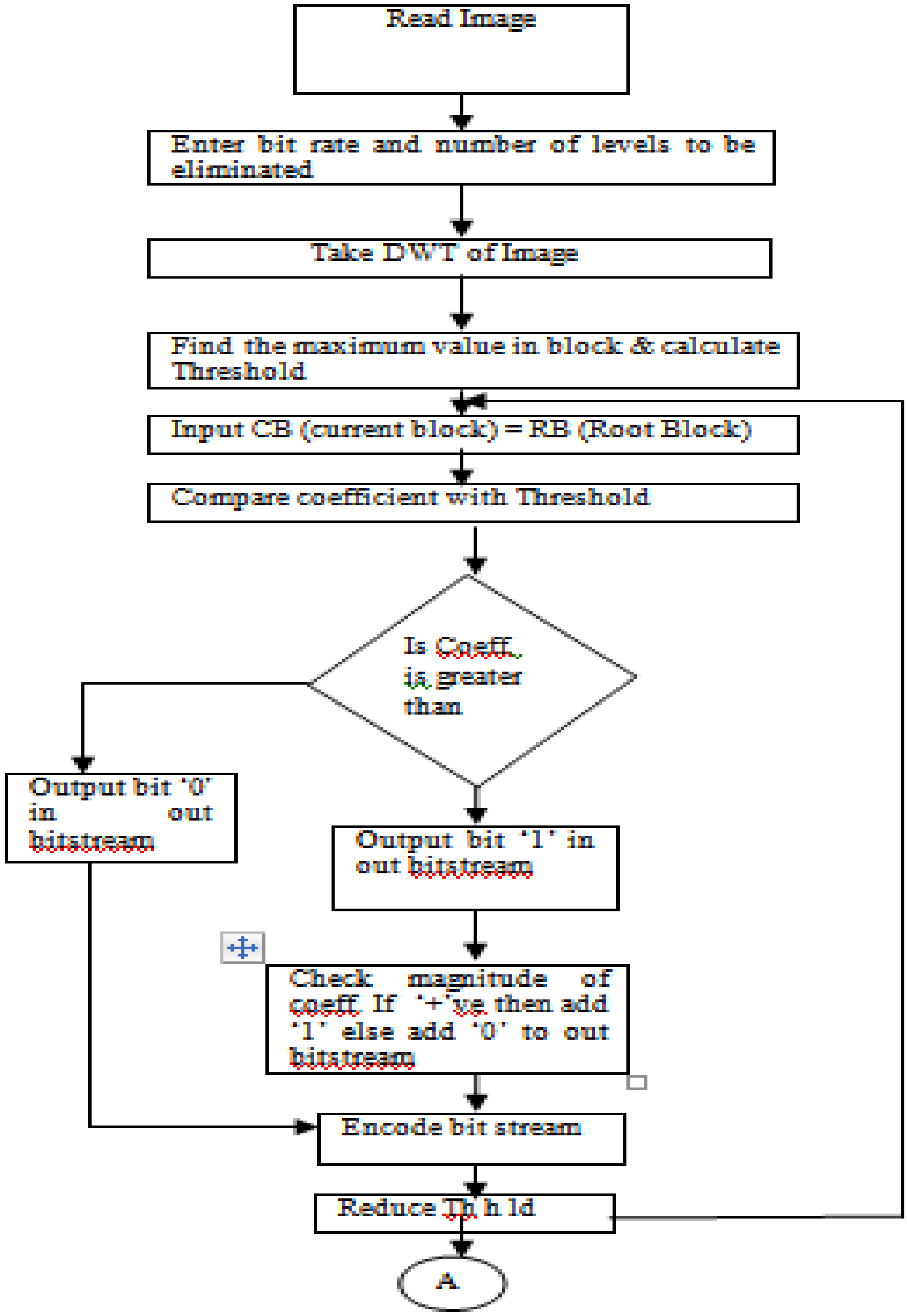
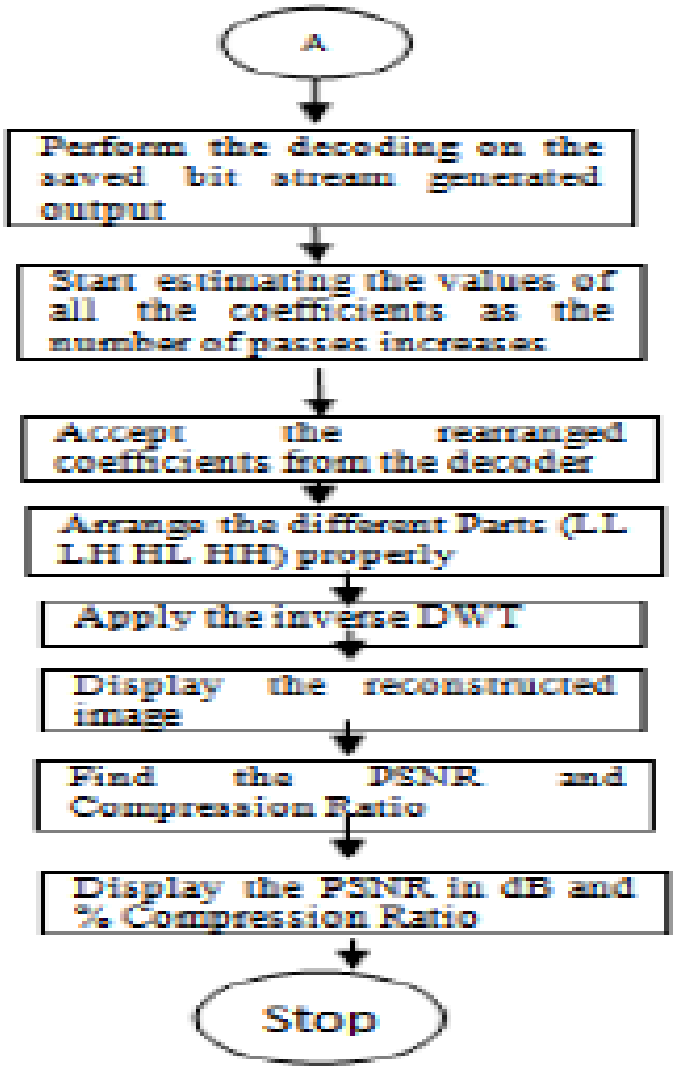


Figure 3.3. Flow Chart of Encoder [2]



4. The Coding Methodology:

In this section, we explain the idea of the SPECK coding scheme and its terminology. Consider an image X which has been adequately transformed using an appropriate sub band transformation (most commonly, the discrete wavelet transforms).

The transformed image is said to exhibit a hierarchical pyramidal structure defined by the levels of decomposition, with the topmost level being the root. The finest pixels lie at the bottom level of the pyramid while the coarsest pixels lie at the top (root) level.

The image *X is* represented by an indexed set of transformed coefficient $\{C_{i,j}\}$ located at pixel position $\{i,j\}$ in the transformed image.

Pixels are grouped together in sets which comprise regions in the transformed image. Following the ideas of SPIHT, we say that a set of *T* pixels is *significant* with respect to n if

$$\max_{(i,j)\epsilon T}\{|C_{i,j}|\} \gg 2^n \quad (4.1)$$

Otherwise, it is *insignificant*. We can write the significance of a set *T* as a function of *n* and the set *T*, i.e.

$$\mathrm{S_n}(T) = \begin{cases} 1 & if\ 2^n \ll \max_{(i,j)\epsilon T}\{|C_{i,j}|\} \gg 2^{n+1} \\ 0 & \end{cases} \quad (4.2)$$

The SPECK algorithm makes use of rectangular regions of image. These regions or sets, henceforth referred to as sets of type *S*, can be of varying dimensions. The dimension of a set S depends on the dimension of the original image and the sub band level of the pyramidal structure at which the set lies. We define the size of a set to be the cardinality *C* of the set, i.e., the number of elements (pixels) in the set.

$$\text{Size (S)} = \text{C(s)} = |S| \quad (4.3)$$

During the course of the algorithm, sets of various sizes will be formed, depending on the characteristics of pixels in the original set. Note that a set of size 1 consists of just one pixel. The other type of sets used in the SPECK algorithm is referred to as sets of type I.

These sets are obtained by chopping off a small square region from the top left portion of a larger square region. A typical set I is illustrated in Figure. 4.1. A set I is always decomposed into S sets in a prescribed way, so as to progress through the transformed image from coarser to finer resolution sub bands.

The coding part of SPECK always takes place on the S sets. The general idea of this scheme to encode an S set follows closely that of the SPIHT algorithm [9]. The difference lies in the sorting pass where instead of using spatial orientation trees for significance testing, we use sets of type S as defined above. The idea behind this is to exploit the clustering of energy found in transformed images and concentrate on those areas of the set which have high energy. This ensures that pixels with high information content are coded first. We maintain two linked lists: LIS - List of Insignificant Sets, and LSP - List of Significant Pixels.

The former contains sets of type S of varying sizes which have not yet been found significant against a threshold n while the latter obviously contains those pixels which have tested significant against n.

Alternatively, as will become obvious later on, we can use an array of smaller lists of type LIS, each containing sets of type S of a fixed size, instead of using a single large list having sets S of varying sizes. These smaller lists are ordered by size from smallest single pixel sets first (top) to largest sets last (bottom).

This ordering is the functional equivalent of separating the LIS into two lists, an LIP (list of insignificant points) and an LIS (list of insignificant multi-point sets), as done in SPIHT. Use of multiple lists will speed up the encoding/decoding process.

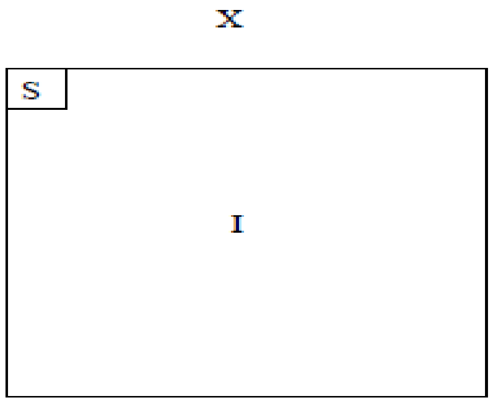


Figure 4.1: Partitioning of Image *X* into *S* and *I* [9]

5. The Speck Coding Algorithm:

Having set-up and defined the terminology used in the SPECK coding method, we are now in a position to understand the actual algorithm. The SPECK coding algorithm is presented in pseudo-code in Figure.5.1.

This is followed by a description of the functions called by the algorithm. These four functions are Process S (), Code S (), Process I() and Code I().

5.1 Initialization:

- Partition image transform X into two sets: S = root and I = X -S (see Figure. 2.5)
- output $n = log_2 (\max_{(i,j)\epsilon X}\{|C_{i,j}|\})$
- add S to LIS and set LSP = ∅

5.2 Sorting Pass

- in increasing order of size C of sets
- for each set S ϵ LIS,
- ProcessS(S)
- ProcessI()

5.3 Refinement Pass

- for each (i, j) ϵ LSP, except those included in the last sorting pass, output the nth MSB of $C_{i,j}$

5.4 Quantization Step:

- decrement n by 1, and go to step 5.2

ProcessS($\mathcal{S}$)
{
- output $S_n(\mathcal{S})$
- if $S_n(\mathcal{S}) = 1$
  - if $\mathcal{S}$ is a pixel, output sign of $\mathcal{S}$ and add $\mathcal{S}$ to LSP
  - else CodeS($\mathcal{S}$)
  - if $\mathcal{S} \in$ LIS, remove $\mathcal{S}$ from LIS
- else
  - if $\mathcal{S} \notin$ LIS, add $\mathcal{S}$ to LIS

}
CodeS($\mathcal{S}$)
{
- Partition $\mathcal{S}$ into four equal subsets $\mathcal{O}(\mathcal{S})$
- for each $\mathcal{O}(\mathcal{S})$
  - output $S_n(\mathcal{O}(\mathcal{S}))$
  - if $S_n(\mathcal{O}(\mathcal{S})) = 1$
    * if $\mathcal{O}(\mathcal{S})$ is a pixel, output sign of $\mathcal{O}(\mathcal{S})$ and add $\mathcal{O}(\mathcal{S})$ to LSP
    * else CodeS($\mathcal{O}(\mathcal{S})$)
  - else
    * add $\mathcal{O}(\mathcal{S})$ to LIS

}
ProcessI()
{
- output $S_n(\mathcal{I})$
- if $S_n(\mathcal{I}) = 1$
  - CodeI()

}
CodeI()
{
- Partition $\mathcal{I}$ into four sets — three $\mathcal{S}$ and one $\mathcal{I}$
- for each of the three sets $\mathcal{S}$
  - ProcessS($\mathcal{S}$)
- ProcessI ()

}

Figure 5.1: The SPACT Algorithm [9]

6. Applications:

Image compression techniques have numerous applications in the field of multimedia broadcasting channels, video conferencing, medical imaging and neural networks. If we talk about TV programs broadcasting, Image compression technique using SPECK Coding has become essential tool. Usually, a full colour image is made up of three primary colours: red, green, and blue. In most cases, there is a strong correlation between the red, green, and blue components of the colour image. However, this colour space is not optimal for lossless compression since the energy is distributed over these planes. Reversible RGB to LC transformations reduce the correlation between the R, G and B components and pack the energy of the colour image into one component and by using SPECK Coding we can save the cost of heavy storage device. SPECK Coding also reduces the cost of extra bandwidth used when uncompressed image is transmitted to satellite for broadcasting. SPECK Coding can also be used in compression of live video streaming in video conferencing.

7. Future Scope of Research Work:

The thesis has contributed in development of low memory wavelet based embedded coders. The significant contribution is SPECK algorithm. Following the results and analysis described in this thesis, a number of future works could be taken up as follows.

- The concept behind the low memory coders can be extended for color images, and volumetric images and videos.
- The coding efficiency of low memory coder suffers slight degradation than SPECK at some points. The improvement of coding efficiency can be attempted possibly by context formations.
- All of the analysis presented in this thesis work involved exhaustive simulations. The algorithm can be realized in hardware implementation as a future work.
- The proposed algorithm can also be a good option for the image processor of the wireless capsule endoscopic system.

8. Conclusion:

In the wireless networks the use of handheld multimedia devices like mobile is increasing day-by-day. Also emerging wireless multimedia sensor network needs real time image transmission among its nodes. Due to limited memory, battery life and processing power of these portable devices and sensor nodes, the real time transmission of images through these devices require an image coding algorithm that can compress images efficiently, requiring minimum possible dynamic memory and with reduced computational complexity We introduced a new set partitioning, block-based embedded image coding scheme, the SPECK image coder. The key to the algorithm is too properly and efficiently sort sets of wavelet coefficients according to their maximum magnitude with respect to a sequence of declining thresholds. The sorting mechanism is partitioning by quadric section guided by threshold significance tests. This allows processing of different regions of the transformed image based on their energy content. We explained the operation of SPECK in various modes of computational and memory complexity and presented extensive coding simulation results for monochrome. A new set partitioning, block-based embedded image coding scheme, the SPECK image coder. The key to the algorithm is too properly and efficiently sort sets of wavelet coefficients according to their maximum magnitude with respect to a sequence of declining thresholds. This chapter concludes a whole complex algorithm of SPACT and overview of Set partitioning block coder.

Acknowledgements:

The first author Asstt. Professor Avadhesh Kumar Maurya is thankful to his Institute management for all kind support in successful conduction of this research work.

References:

1. Grgic S., Grgic M. and Zovko-Cihlar B, “Performance analysis of image compression using wavelets”, IEEE Transaction on Industrial Electronics 2002; 48:682-95

2. Islam A. and Pearlman W.A, “An embedded and efficient low- complexity hierarchical image coder”, Visual Communications and Image Processing, Proceedings of SPIE, vol.36 No.53, pp. 294–305, Jan.1999.
3. Islam A. and William A., “Efficient, Low- Complexity Image Coding with a Set-Partitioning Embedded Block Coder,” IEEE Transactions Circuits and Systems for Video Technology, vol. 14, pp. 1219-1235, November 2004.
4. Maurya A.K., Maurya V.N. and Singh R.K., Computational approach for performance analysis of photonic band gap structure on defected ground surface with microwave and band stop filter, American Journal of Engineering Technology, Academic & Scientific Publishing, New York, USA, Vol.1(7),10-18 (2013)
5. Maurya V.N. and Maurya A.K, Application of Haar wavelets and method of moments for computing performance characteristics in electromagnetic materials, American Journal of Applied Mathematics and Statistics, Science & Education Publishing, USA, Vol. 2(3), pp. 96-105, 2014, ISSN (Print) 2328-7306, ISSN (Online) 2328-7292
6. Maurya V.N. and Bhattacharjee S., An empirical perspective on the design and use of embedded systems for stone fruits of Meghalaya, International Journal of Science Technology and Management, Vol.4, Special Issue No. 1, April 2015, pp. 369-383, ISSN (Online) 2394-1537, India
7. Maurya V.N. and Bhattacharjee S., Designing an embedded system for Orchard Management, American Journal of Computer Research Repository, Vol. 6, Issue No.1, pp. 1-7, 2021, Science and Education Publishing, USA, ISSN (Online) 2377-4266
8. Pearlman A., “Efficient, Low-Complexity Image Coding with A Set-Partitioning Embedded Block Coder,” http://www.cipr.~$ltiwavelet_paper. docrpi.edu/~pearlman/papers/speck_example.pdf, IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 11, pp. 1219 – 1235, 2004.
9. Said A. and Pearlman W.A., “A new, fast and efficient image codec based on set partitioning in hierarchical trees”, IEEE Transaction on circuits and systems for Video Technology, vol. 6, pp. 243–250, June 1996.
13. Sprljan N., Grgic Sonia and Grgic Mislav, “Modified SPIHT algorithm for Wavelet packet image coding”, Elsevier Real Time Imaging pp 378- 388, 11, 2005