

# A NOVEL HYBRID APPROACH BASED ON GEOMETRIC WAVELETS FOR EFFICIENT IMAGE COMPRESSION

<sup>1</sup>REHNA. V. J., <sup>2</sup>DR. JEYA KUMAR. M. K

<sup>1</sup>Research Scholar, Department of Electronics & Communication Engineering, Noorul Islam University

<sup>2</sup>Professor, Department of Computer Applications, Noorul Islam University, Tamil Nadu, India

E-mail: <sup>1</sup>[rehna\\_vj@yahoo.co.in](mailto:rehna_vj@yahoo.co.in), <sup>2</sup>[jeyakumarmk@yahoo.com](mailto:jeyakumarmk@yahoo.com)

## ABSTRACT

A novel hybrid algorithm based on geometric wavelets for efficient compression of digital images is proposed. The presented work combines the recent segmentation based binary space partition scheme with the popular geometric wavelet coding method to capture the edge singularities in a more effective way and to provide the sparse representation of the image. The BSP scheme uses polar co-ordinate representation of straight line for partitioning the image domain. This improved the choice of bisecting lines available for partitioning thereby enhancing the probability of reducing the cost functional. A new pruning algorithm is tried to optimize the rate distortion curve and achieve the desired bit rate. A new “geometric” context modeling scheme combined with arithmetic coding is designed to boost the performance of the algorithm. The signal-to-noise ratios are compared with state-of-the-art wavelet coders; recent segmentation based algorithms as well as the original geometric wavelet coding algorithm and found that the results outperform the existing methods. The results report a gain of 2.19 dB over the EZW algorithm and 1.35 dB over the SPIHT algorithm at the bit-rate 0.0625 bpp. The presented algorithm shows a gain of 1.01 dB over the original GW algorithm at the compression ratio of 128 for the Lena test image. But for the high computational complexity and hence increased time complexity, the algorithm gives remarkable results in terms of rate-distortion compression.

**Keywords:** *Binary Space Partition, Geometric Wavelets, Hybrid Coding, Rate-Distortion Compression, Signal-to-Noise Ratio*

## 1. INTRODUCTION

Storage and transmission of digital images has become an integral part of our daily life. As our dependence on the digital media continues to grow, finding competent ways of storing and conveying these large amounts of data has become a major concern. Because the amount of space required to hold unadulterated images can be extremely large in terms of cost, as well as of the huge bandwidth required to transmit them, researchers are seeking methods for efficient representations of these digital pictures to simplify their transmission and save disk space. At this point in time, the technique of image compression has become very essential and highly applicable. To date, substantial advancements in the field of image compression have been made, ranging from the traditional predictive coding approaches, classical and popular transform coding techniques and vector quantization to the more latest second generation coding schemes. Starting at 1 with the first digital picture in the early 1960s, the compression ratio has reached a saturation level of

around 300:1 recently. Even then, the reconstructed image quality still remains as an important issue to be investigated.

The Discrete Cosine Transform (DCT) [1] has been, until recently, the most popular technique for image compression because of its optimal performance and ability to be implemented at a reasonable cost. The popular JPEG standard [2] for still images and the MPEG standard for moving images are based on DCT. Wavelet-based image coding techniques [3] are the latest development in the field of image compression offering multiresolution capability resulting in superior energy compaction and high quality reconstructed images at low bit rates. The discrete wavelet transform forms the basis of the popular JPEG 2000. The wavelet transforms based coding approaches have taken over other classical methods particularly the cosine transform, due to its capability to solve the problem of blocking artefacts which is a common phenomenon in DCT based compression. However, the EZW [4], the SPIHT

[5], the SPECK [6], the EBCOT [7] algorithms and the current JPEG 2000 [8] standard are based on the discrete wavelet transform (DWT) [9]–[10].

Despite providing outstanding results in terms of rate-distortion compression, the transform-based coding methods do not take an advantage of the geometry of the edge singularities in an image. This led to the design of ‘Second Generation’ or the segmentation based image coding techniques [11] that make use of the underlying geometry of edge singularities of an image. To this day, almost all of the proposed ‘Second Generation’ algorithms are not competitive with state of the art (dyadic) wavelet coding. In this regard, inspired by a recent progress in multivariate piecewise polynomial approximation [12], we put together the advantages of the classical method of coding using wavelets and the segmentation based coding schemes to what can be described as a geometric wavelet approach.

This study focuses on a recent development in the field of piecewise polynomial approximation for image coding using Geometric wavelets [13]. This scheme efficiently captures curve singularities and provides a sparse representation of the image and thereby achieves better quality reconstructed images with higher compression ratios. Stress is given on the shared approach of image compression using geometric wavelets and the binary space partition scheme. The current study is envisaged to enhance the GW image coding [13] method and its improved version [14]. We use the polar co-ordinate form of the straight line [15] in the binary space partition scheme (BSP). Here the number of quantized bisecting lines is increased and hence probability of minimizing the cost functional and finding the optimal cut of the domain is improved. A rate-distortion optimization process is performed prior to encoding where a new pruning algorithm is tried to prune the BSP tree and achieve the desired bit rate. A new “geometric” context modeling scheme combined with arithmetic coding is designed to boost the performance of the algorithm.

The paper is organized as follows: Section 2 gives a brief summary of up to date research carried out in relation to this work, based on the literature survey. Section 3 deals with the basic concepts of binary space partition scheme and geometric wavelets. Sections 4 give the details of the geometric wavelet image coding algorithm. Section 5 provides experimental results which are compared with those of recent state-of-the-art wavelet and “sparse geometric representation” methods and also with GW and improved GW approaches. Summary & conclusion is presented in section 6.

## 2. LITERATURE SURVEY

A number of segmentation algorithms have been proposed for image coding till date, each claiming to be different or superior in some way. The first segmentation-based coding methods were suggested in the early 1980s [11]. These algorithms partition the image into complex geometric regions using a contour-texture coding method (1982) [15] over which it is approximated using low-order polynomials. One of the most popular segmentation based coding schemes investigated by researchers in the early days were the Quadtree-based image compression (1991) [16], which recursively divides the image signal into simpler geometric regions. Many variations of the ‘Second Generation’ coding schemes have since been announced that exploit the geometry of curve singularities of an image [17], [18], [19]. In one of the outstanding ‘Second Generation’ methods, Froment and Mallat (1992) constructed multi-scale wavelet-like edge detectors and showed how a function from the responses of a sparse collection of these detectors can be reconstructed [20]. They reported good coding results at low bit-rates. Candès and Donoho (2001) constructed, a bivariate transform called Curvelets intended to capture local multi-scale directional information [21]. Cohen and Matei (2001) also presented a discrete construction of an edge-adapted transform [22] which is closely related to nonlinear Lifting (2003) [23]. In a later work (2003), the authors enhance classical wavelet coding by detecting and coding the strong edges separately and then using wavelets to code a residual image [4]. Do and Vetterli’s construction of Contourlets (2005) [24], is similar but is a purely discrete construction. Coding algorithms that are geometric enhancements of existing wavelet transform based methods, where wavelet coefficients are coded using geometric context modelling also exist [25]. But all of these constructions are redundant, i.e., the output of the discrete transform implementations produces more coefficients than the original input data. Research on the possibility of using these new transforms to outperform wavelet based coding is still on-going.

The binary space partition (BSP) scheme, a simple and efficient method for hidden-surface removal and solid modelling was introduced in 1990 [26]. The BSP technique was applied to the concept of image compression in 1996 [27] and is adopted in the first stage of this study. Later, in 2000, binary partition trees were used as an efficient representation for image processing, segmentation, and information retrieval [28].

Recently, many second generation image compression algorithms such as the Bandelets (2005) [29], the Prune tree (2005) [12], the Prune-Join tree (2005) [12], the GW image coding method (2007) [13] and the like based on the sparse geometric representation have been introduced. LePennec and Mallat (2005) [29] lately applied their 'Bandelets' algorithm to image coding, where a warped-wavelet transform is computed to align with the geometric flow in the image and the edge singularities are coded using one-dimensional wavelet type approximations. The concept of combining the binary space partition scheme and geometric wavelets for compression of digital images were put forward by Dror Alani, Amir Averbuch, and Shai Dekel in 2007 [13]. Here the bisecting lines of the BSP scheme are quantized using the normal form of straight line. This method successfully competes with state-of-the-art wavelet methods such as the EZW, SPIHT, and EBCOT algorithms and also beats the recent segmentation based methods. But the algorithm turned out to be computationally very intensive [16]. An improvement was made to this work in 2011 by Garima Chopra and A. K. Pal [14]. They used the slope intercept form of a straight line instead of the normal representation. This improved the possibility of minimizing the cost functional by increasing the choice of bisecting lines available for partitioning. The technique further increased the complexity of the algorithm.

Our approach deviates from the context of multi-scale geometric processing, even from the more general framework of harmonic analysis, which is the theoretical basis for transform based methods and also from the popular wavelet based studies and is based on the GW and binary space partition method introduced in [13]. The main difference between the GW algorithm and recent work is that we use the polar coordinate representation of straight line for partitioning the domain thereby further improving the availability of partitioning lines and intern further minimizing the cost functional at each step of BSP scheme.

### 3. MATHEMATICAL PRILIMINARIES

#### 3.1 Binary Space Partition Scheme

Given an image  $f$ , the algorithm divides convex polygonal domain  $\Omega$  into two subsets  $\Omega_0$  and  $\Omega_1$  using a bisecting line. The subdivision is performed to minimize a given cost functional (equation 1). This partitioning process then operates recursively in a hierarchical manner on the subdomains until some exit condition is met. To be specific, we

describe the algorithm of [27], which is a BSP algorithm that identifies a compact geometric description of a target bivariate function. The goal in [27] is to encode an optimal cut of the BSP tree, to be precise, a sparse piecewise polynomial approximation of the original image based on the union of disjoint polygonal domains in the BSP tree. Rate-distortion optimization strategies are used [9] to meet a given bit rate.

For a given convex polygonal domain  $\Omega$ , the algorithm finds two subdomains,  $\Omega_0$  and  $\Omega_1$ ; and two bivariate (linear) polynomials  $Q_{\Omega_0}$  and  $Q_{\Omega_1}$ , that minimizes the given cost functional:

$$F(\Omega_0, \Omega_1) = \arg \min_{\Omega_0, \Omega_1} \|f - Q_{\Omega_0}\|_{\Omega_0}^2 + \|f - Q_{\Omega_1}\|_{\Omega_1}^2 \quad (1)$$

where  $\Omega_0$  and  $\Omega_1$  represent the subsets resulting from the subdivision of  $\Omega$  ( $\Omega_0$  and  $\Omega_1$  should be considered as children for the mother  $\Omega$ ). The bivariate polynomial used is defined by:

$$Q_{\Omega_i} = A_i x + B_i y + C_i \quad (2)$$

The polynomial interpolation is made using the least square method [33], computing the difference between the image and the polynomial at a defined region  $\Omega$ . The algorithm continues partitioning each region recursively until there are no enough pixels to subdivide or the approximation error is sufficiently small. The algorithm constructs a binary tree with the partitioning information. The algorithm needs to encode the information of the geometry, namely, the line that cut each sub-domain, and the approximation function in each sub-domain represented by the polynomial coefficients. Figure 1. shows the steps involved in Binary Space Partitioning algorithm.

First a line  $L$  divides the region  $\Omega$  into two regions  $\Omega_0$  and  $\Omega_1$ . The two regions  $\Omega_0$  and  $\Omega_1$  are further divided into  $\Omega_{00}$ ,  $\Omega_{01}$  and  $\Omega_{11}$ ,  $\Omega_{10}$  respectively. These four regions are further divided into eight and so on until area of the subdomain contains only a very few pixels. A more flexible exit criterion to cease partitioning is when the approximation error,  $\|f - Q_{\Omega}\|_{\Omega}^2$  is sufficiently small. Then it is represented in a tree structure as shown in Figure 2.

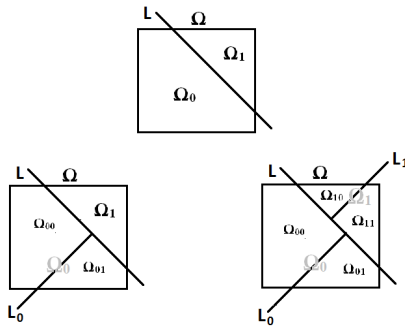


Figure 1: Binary Space Partitioning of the domain  $\Omega$  (two levels).

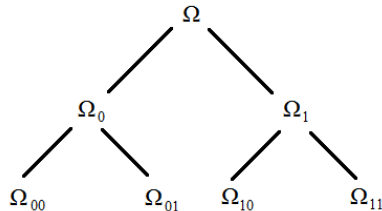


Figure 2: BSP tree representation

### 3.2 Geometric Wavelets

Geometric Wavelets [30] are multi-scale dictionary elements which are constructed directly from the data, and have guarantees on the computational cost, the number of elements in the dictionary and the sparsity of the representation. Geometric wavelets (GW) have been considered in context of image compression in [13]. It is a new multi-scale data representation technique which is useful for a variety of applications such as data compression, interpretation and anomaly detection [9]. The GW is defined as:

$$\Psi_{\Omega_0}(f) \triangleq I_{\Omega_0}(Q_{\Omega_0} - Q_{\Omega}) \quad (3)$$

$\Omega_0$  here means one of the children of mother,  $\Omega$ . It is possible to reconstruct the function  $f$  using:

$$f = \sum_{\Omega_i} \Psi_{\Omega_i}(f) \quad (4)$$

Geometric Wavelet,  $\Psi_{\Omega}$  is a “local difference” component that belongs to the detail space between two levels in the BSP tree, a “low resolution” level associated with  $\Omega_0$  and a “high resolution” level associated with  $\Omega_0$ . Geometric wavelets also satisfy the vanishing moment property like isotropic wavelets [31], i.e., if  $f$  is locally a polynomial over  $\Omega$ , then minimizing of (1) gives  $Q_{\Omega_0} = Q_{\Omega} = 1$ , and therefore  $\Psi_{\Omega_0}(f) = 0$ . Unlike classical wavelets, geometric wavelets do not satisfy the two scale relation and the biorthogonality property.

## 4. THE GEOMETRIC WAVELET CODING ALGORITHM

We encode the differences between the original coarse projections of the data and the points projected onto the planes at a finer scale, to find a compact representation for the data at the finer scale. For this, an effective scheme is developed based on the construction of a minimal space spanning this set of differences [32]. The axes of this difference space are termed “geometric wavelets”, and the projections of the finer-scale corrections to the data points onto the plane spanned by these axes are called the “wavelet coefficients”. The process is continued, forming a binary tree of mother and children at finer and finer scales until no further details are needed to approximate the data up to a pre-specified accuracy. The process is discussed in detail in the following sections.

### 4.1 BSP Tree Construction

The BSP method is computationally very intensive. Therefore, the image is tiled first and then the BSP algorithm is applied independently on each tile, thereby creating a BSP forest. The tile size is generally adopted is 128 x 128. The BSP scheme is applied on each tile of the image by using the polar coordinate form of the straight line. In polar coordinates on the Euclidean plane, a line is expressed as:

$$r = \frac{b}{\sin\theta - m\cos\theta} \quad (5)$$

where  $m$  is the slope of the line and  $b$  is the y-intercept. The equation can be rewritten as:

$$r \sin\theta = mr \cos\theta + b \quad (6)$$

It is not possible to quantize the parameter  $m$ , as it is unbounded, has value infinity for the straight lines which are parallel to y axis. This problem is solved by using the new parameter  $\phi$  in place of  $m$  in (8), where  $\phi$  is the angle between the line and the x axis in the anticlockwise direction. Parameters  $\theta$  and  $\phi$  are shown in Figure 3.

Subsequently, equation (4) reduces to:

$$r \sin\theta = \tan\phi \cdot r \cos\theta + b \quad (7)$$

The number of bisecting lines available for the partitioning of tile of dimension 128 x 128 in [14] is 15740. In the improved GW approach [14], the number increased to 60775. But in the proposed algorithm this availability number further increased to 69780. Hence, this method provides a better choice of bisecting lines thereby giving more possibility to minimize the cost functional. Table 1



gives the minimum values of the cost functional (1) on the initial partitioning of different tiles of the Cameraman test image. Following the procedure mentioned in section 3.1, the BSP tree is generated for each tile and according to the method defined in section 3.2, geometric wavelets are created for each node.

#### 4.2 Sparse Geometric Wavelet Representation

The GW image coding algorithm [12] is based on the idea that among all the geometric wavelets only a “few” wavelets have large norm. Once all the geometric wavelets are created, they are arranged according to their  $L^2$  norm as shown in equation (8).

$$\|W_{1k_1}\|_2 \geq \|W_{1k_2}\|_2 \geq \|W_{1k_3}\|_2 \geq \dots \quad (8)$$

Then the sparse geometric representation is extracted using the greedy methodology of nonlinear approximation [33], [34]. Here,  $n$  wavelets are selected from the joint list of geometric wavelets over all tiles.

A rate-distortion optimization is performed prior to encoding where a new pruning algorithm is tried. The R-D curve for each node is generated by approximating the node by the quantized polynomial  $\hat{p}(t)$ , which is obtained by scalar quantizing the polynomial coefficients. In this Lagrangian cost based pruning, the R-D optimal pruning criterion for the given operating slope,  $\lambda$  is as follows: Prune the children if the sum of Lagrangian costs of the children is greater than or equal to the Lagrangian cost of the parent. Mathematically, this means that the children are pruned if  $(D_{C1}+D_{C2}) + \lambda(R_{C1}+R_{C2}) \geq (D_P+\lambda R_P)$ , where  $R_P$  and  $D_P$  are rate and distortions of the parent and  $R_{C1}$ ,  $R_{C2}$ ,  $D_{C1}$  and  $D_{C2}$  are the rate and distortions of the children respectively.

Subsequently, function  $f$  is approximated using the  $n$ -term geometric wavelet sum given in equation (4), where  $n$  is the number of wavelets used in the sparse representation.

#### 4.3 Encoding

To obtain a reasonable approximation of the image, it is essential that if a child is present in the sparse representation, then the mother should also be there, i.e., the BSP tree should be connected. Therefore, instead of encoding an  $n$ -term tree approximation, we create an  $n + k$  geometric wavelet tree by considering more  $k$  nodes [35]. The cost of imposing the condition of the connected tree structure is not very huge, since there is high

probability that if a child is important all its ancestors are also important [33], [34].

There are two sorts of data to be encoded, 1) the geometry of the support of the wavelets participating in the sparse representation and 2) the polynomial coefficients of the wavelet. Before encoding the extracted BSP forest, a small header is written to the compressed file. Header consists of the minimum and maximum values of the coefficients of the participating wavelet and the image graylevels. Out of header size of 26 bytes, 24 are used in the storage of the minimum and the maximum values of the coefficients while 2 bytes are utilized to store the extremal values of the image [36]. “Root” geometric wavelets [14] contribute most in the approximation, so each root wavelet is encoded. The encoding process is applied repeatedly for each of the geometric wavelet tree nodes in each tile.

##### 4.3.1 Encoding the Geometry of the Support of the Wavelet

The following information is encoded for each of the participating node  $\Omega$ :

- Number of children of  $\Omega$  that participate in the sparse representation;
- In case only one child is participating, then whether it is the left or the right child;
- If  $\Omega$  is not a leaf node, then the line that bisects  $\Omega$  is encoded using the slope intercept form.

Left child and right child are defined as the sets of the pixels satisfying the inequality  $r - \tan \theta \cdot r \sin \theta \leq b$  and  $r - \tan \theta \cdot r \sin \theta \geq b$ , respectively. The leaf node is encoded by using the bit “1.” Codes “00” and “01” are used for the one child symbol and the two children symbol, respectively. If only 1 child of  $\Omega$  is participating in the sparse representation, then this event is encoded by using an additional bit. In case  $\Omega$  is not a leaf node, then the indices of the parameter  $\theta$  and  $c$  of the bisecting line are encoded using the lossless variable length coding.

##### 4.3.2 Encoding the Wavelet Coefficients

The coefficients of the wavelet polynomial,  $Q_\Omega$  are quantized and encoded using an orthonormal representation of  $\Pi_1(\Omega)$ , where  $\Pi_1(\Omega)$  is the set of all bivariate linear polynomials over  $\Omega$ . A bit allocation scheme for the coefficients is applied using their distribution function (over all the domains) which is discussed in later sections. The “root” wavelet of each tile is always encoded.

Quantizing the Wavelet Coefficients: To ensure the stability of the quantization process of the geometric wavelet polynomial  $Q_{\Omega}$ , we first need to find its representation in appropriate orthonormal basis. The orthonormal basis of  $\Pi_1(\Omega)$  is found using the standard Gram-Schmidt procedure. Let  $V_1(x,y)=1$ ,  $V_2(x,y)=x$ ,  $V_3(x,y)=y$  and be the standard polynomial basis. Then, an orthonormal basis of  $\Pi_1(\Omega)$  is given by

$$\begin{aligned} U_1 &= \frac{V_1}{\|V_1\|} \\ U_2 &= \frac{V_2 - \langle V_2, U_1 \rangle U_1}{\|V_2 - \langle V_2, U_1 \rangle U_1\|} \\ U_3 &= \frac{V_3 - \langle V_3, U_1 \rangle U_1 - \langle V_3, U_2 \rangle U_2}{\|V_3 - \langle V_3, U_1 \rangle U_1 - \langle V_3, U_2 \rangle U_2\|} \end{aligned} \quad (9)$$

where inner product and norm are associated with the space  $L_2(\Omega)$ . Let

$$\Psi = \alpha U_1 + \beta U_2 + \gamma U_3 \quad (10)$$

be the representation of the geometric wavelet  $\Psi \in \Pi_1(\Omega)$  in the orthonormal basis.

A bit allocation scheme is applied depending upon the distribution functions of the coefficients  $\alpha$ ,  $\beta$  and  $\gamma$  of the wavelets participating in the sparse representation. Figure 4 shows the histogram of the wavelet coefficients of Cameraman. Four bins are used to model the absolute value of the coefficients; bin limits are computed and passed to the decoder. In case all the three coefficients of the wavelet are small, the event is encoded using single bit, but if any one of them is not small then the bin number of each coefficient is encoded. After this quantized bits are written to the compressed file. Figures 5 and 6 show how the bit budget allocation of Lena at the bit-rate 0.0625 bits per pixel (bpp) and 0.125 bpp is distributed among the GW algorithm components respectively.

#### 4.4 Decoding

In the decoding stage, the compressed bit stream is read to find whether the participating node is a root node, has 1 child or 2 children, or a leaf node. If one child is participating then by using bit stream identification, it is found whether it is left child or right child. If at least one of the children belongs to the sparse representation, then the indexes of  $\emptyset$  and  $b$  are decoded and using these index parameters  $\emptyset$  and  $b$  of optimal cut are calculated. Thereafter, using this optimal cut, domain is partitioned into two subdomains; and depending upon the situation vertex set of only one child or both children is found [37]. This process is repeated until entire bit stream is read.

## 5. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed algorithm is tested on the still image of Lena of bit depth 8 and of size 512x512. The implementation is done using 2010 version of MATLAB. The Peak Signal to Noise Ratio (PSNR) based on Mean Square Error (MSE) is used as a measure of "quality" [18]. MSE and PSNR are given by the following relations:

$$MSE = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n (x_{i,j} - y_{i,j})^2 \quad (11)$$

$$PSNR = 10 \log_{10} \left[ \frac{(255)^2}{MSE} \right] \quad (12)$$

where  $m \times n$  is the image size,  $x_{i,j}$  is the original image and  $y_{i,j}$  is the reconstructed image. MSE and PSNR are inversely proportional to each other and higher value of the PSNR implies better quality reconstructed image.

The proposed method reports a gain of 1.35 dB [16] over the SPIHT [5] method, 1.43 dB over the EBCOT [7] method and 2.19 dB over EZW [4] algorithm at the compression ratio of 128:1 for the Lena test image. The presented algorithm shows a gain of 1.01 dB over the original GW method [13] and 0.95 dB over the improved GW algorithm [14] at a bit rate of 0.0625 bpp for the Lena image.

Table 2 and table 3 give the comparison of PSNR in dB for different variations in the GW algorithm on test images - Lena and Cameraman respectively. Figure 7 shows the reconstructed image of Cameraman and Lena using the algorithm, at the compression ratio of 128:1 and PSNR is 24.55 and 29.73 respectively. Figure 8 shows the reconstructed Barbara images and Figure 9 shows the reconstructed Egg and Vegetable images using the different variations of the algorithm.

## 6. CONCLUSION

The performance of a hybrid algorithm for image compression using the geometric wavelets and the tree-structured binary space partition scheme is investigated. We have improved the coding efficiency of the GW algorithm by using the polar coordinate form of straight line for optimal bisection in the line selection procedure. The use of new pruning algorithm further improved the PSNR making it competitive with the state-of-art coders in literature. The method works well with geometrically rich content images and gives remarkable results at low as well as medium bit-rates. The algorithm is found to be extremely complex in computation and has high execution time. In future, better statistical models may be developed for encoding the BSP geometry. New methods to reduce the time complexity of the algorithm may also be explored.

**REFERENCES:**

- [1] Rao K.R, and Yip P, "Discrete Cosine Transform: Algorithms, Advantages, Applications", *New York: Academic*, 1990.
- [2] Wallace G. K. "The JPEG Still-Picture Compression Standard", *Commun. ACM*, 34, 1991, pp. 30-44.
- [3] Daubechies, I, "Ten Lectures on Wavelets", presented at the *CBMS-NSF Reg. Conf. Ser. Applied Mathematics*, 1992.
- [4] Shapiro J M, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients", *IEEE Trans. Signal Process.*, 41, 1993, pp. 3445-3462.
- [5] Said A and Pearlman W. A, "A New, Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", *IEEE Trans. Circuits Syst. Video Technol.*, 6, 1996, pp. 243-250.
- [6] Islam A, and Pearlman W. A, "An Embedded and Efficient Low Complexity Hierarchical Image Coder", in *Proc. SPIE*, 3653, 1999, pp. 294-305.
- [7] Tauban D, "High Performance Scalable Image Compression with EBCOT", *IEEE Trans. Image Process.*, vol. 9, no. 7, Jul. 2000, pp. 1158-1170.
- [8] Skodras A, Christopoulos C, and Ebrahimi T, "The JPEG2000 Still Image Compression Standard," *IEEE Signal Process. Mag.*, 18, 2001, pp. 36-58.
- [9] Daubechies, I, "The Wavelet Transform, Time Frequency Localization and Signal Analysis," *IEEE Trans. Inf. Theory*, 36, 1990, pp. 961-1005.
- [10] Antonini M, Barlaud M, Mathieu P, and Daubchies I, "Image Coding Using Wavelet Transform", *IEEE Trans. Image Process.*, 1, 1992, pp. 205-220.
- [11] Kunt M, Ikonomopoulos A, and Koche M, "Second Generation Image Coding Techniques," *Proc. IEEE*, 73, 1985, pp. 549-574.
- [12] Shukla R, Daragotti P. L, Do M. N, and Vetterli M, "Rate-Distortion Optimized Tree Structured Compression Algorithms for Piecewise Polynomial Images", *IEEE Trans. Image Process.*, 14, 2005, pp. 343-359.
- [13] Alani D, Averbuch A, and Dekel S, "Image Coding with Geometric Wavelets", *IEEE Transactions on Image Processing*, 16, 2007, pp. 69-77.
- [14] Garima Chopra & Pal, A. K, An Improved Image Compression Algorithm using Binary Space Partition Scheme and Geometric Wavelets", *IEEE Transactions on Image Processing*, 20, 2011, pp. 270 - 275.
- [15] Rehna V. J, and Jeyakumar M. K, "An Enhanced Geometric Wavelet Based Hybrid Image Compression Algorithm for Low Bit Rate Applications", *European Journal of Scientific Research*, Vol.116 No.4, 2014, pp.544-555.
- [16] Rehna V. J, and Jeyakumar M. K, "A Superior Hybrid Algorithm Based on Geometric Wavelets for Compression of Digital Images", *Pensee Journal*, Vol 75, No. 11, 2013, pp. 304-314.
- [17] M. Kocher and M. Kunt "A Contour-Texture Approach to Image Coding", in *Proc. ICASSP*, 1982, pp. 436-440.
- [18] G. J. Sullivan and R. L. Baker "Efficient Quadtree Coding of Images and Video", in *ICASSP Proc.*, 1991, pp. 2661-2664.
- [19] M. Kunt, M. Benard and R. Leonardi, "Recent Results in High Compression Image Coding", *IEEE Trans. Circuits Syst.*, 34, 1987, pp. 1306-1336.
- [20] J. Froment and S. Mallat, "Wavelets: A Tutorial in Theory and Applications", C. K. Chui, Ed. Academic Press, New York, Second Generation Compact Image Coding with Wavelets, 1992.
- [21] E. Candes and D. Donoho, "Curvelets and Curvilinear Integrals", *Journal of Approximation Theory*, 113, 2001, pp. 59-90.
- [22] A. Cohen and B. Matei, "Compact Representations of Images by Edge Adapted Multiscale Transforms," *Proceedings of the IEEE ICIP Conference, Thessaloniki*, 2001.
- [23] R. Leonardi and M. Kunt, "Adaptive Split-and-Merge for Image Analysis and Coding", *Proc. SPIE*, 1985, pp. 594.
- [24] M. N. Do and M. Vetterli, "The Contourlet Transform: An Efficient Directional Multiresolution Image Representation", *IEEE Trans. Image Process.*, 14, 2005, pp. 2091-2106.
- [25] M. N. Do and Yue Lu, "A New Contourlet Transform with Sharp Frequency Localization", *IEEE International Conference on Image Processing*, 2006, pp. 1629-1632.
- [26] M. S. Paterson and F. F. Yao, "Efficient Binary Space Partitions for Hidden-Surface Removal and Solid Modeling", *Discrete Comput. Geom.*, 5, 1990, pp. 485-503.
- [27] H. Radha, M. Vetterli and R. Leonardi, "Image Compression Using Binary Space Partitioning Trees", *IEEE Trans. Image Process.*, 5, 1996, pp. 1610-1624.
- [28] P. Salembier and L. Garrido, "Binary Partition Tree as an Efficient Representation for Image



- Processing, Segmentation, and Information Retrieval”, *IEEE Trans. Image Process.*, 9, 2000, pp. 561–576.
- [29] E. L. Pennec and S. Mallat, “Sparse Geometric Image Representation with Bandelets”, *IEEE Trans. Image Process.*, 14, 2005, pp. 423–438.
- [30] Dekel and Leviatan, “Adaptive Multivariate Approximation Using Binary Space Partitions and Geometric Wavelets”, *SIAM journal on Numerical Analysis*, 43, 2005, pp. 707–732.
- [31] R. L. Claypoole, G. M. Davis W, Sweldens, and R. G. Baraniuk, “Nonlinear Wavelet Transforms for Image Coding via Lifting”, *IEEE Trans. Image Process.*, 12, 2003, pp. 1449–1459.
- [32] A. N. Netravali and B. G. Haskell, “Digital Pictures: Representations and Compressions”, *New York: Plenum*, 1988.
- [33] H. Radha, “Least-Square Binary Partitioning of LZ Functions with Convex Domains”, *AT&T Bell Labs. Tech. Memo, Draft*, 1990.
- [34] M. S. Paterson and F. F. Yao, “Efficient Binary Space Partitions for Hidden-Surface Removal and Solid Modeling”, *Discrete Comput. Geom.*, 5, 1990, pp. 485–503.
- [35] P. Salembier and L. Garrido, “Binary Partition Tree as an Efficient Representation for Image Processing, Segmentation, and Information Retrieval”, *IEEE Trans. Image Process.*, 9, 2000, pp. 561–576.
- [36] R. DeVore, “Nonlinear Approximation,” *Acta Numer.*, 7, 1998, pp. 51–150.
- [37] Rehna V. J, and Jeyakumar M. K, “An Efficient Hybrid Algorithm for Low Bit Rate Image Coding”, *Proceeding of the International Conference on Artificial Intelligence in Computer Science and ICT (AICS 2013)*, Langkawi, MALAYSIA, 25 -26 November 2013, pp. 21-28.



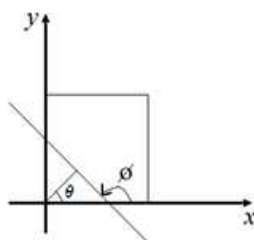


Figure 3: Partition of the Image Domain into Two Subdomains - Parameters  $\theta$  and  $\phi$

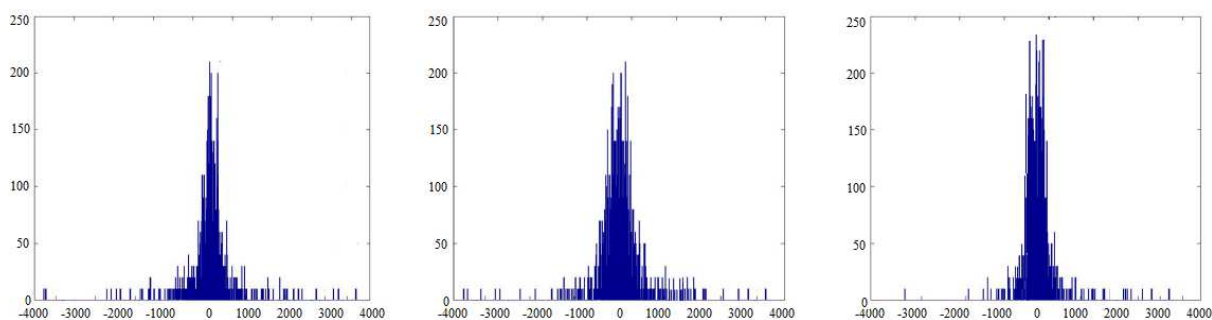


Figure 4: Histogram of Wavelet Coefficients,  $\alpha$  (left),  $\beta$  (middle) and  $\gamma$  (right) of Cameraman Image

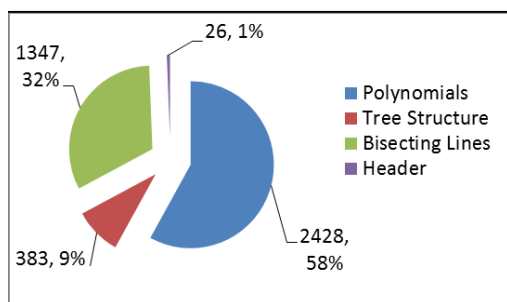


Figure 5: Bit Budget Allocation for Lena at Bit-Rate, 0.03125 bpp. Output File Size is 1 Kbyte

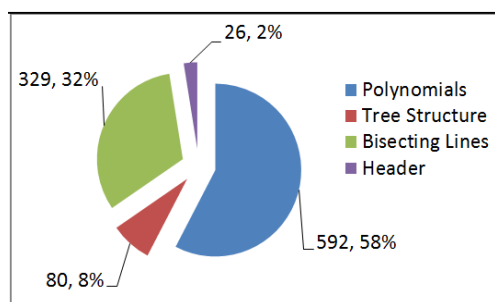


Figure 6: Bit Budget Allocation for Lena at Bit-Rate, 0.125 bpp. Output File Size is 4 Kbytes

Table 1. Minimum Values of Cost Functional for the First Partition on all 4 Tiles of the Cameraman Test Image on Applying the BSP Scheme

Tile	GW Method [13]	Improved GW Method [14]	Proposed Method
Tile 1	26045109.99	26039950.74	26020186.65
Tile 2	13508971.42	13493350.99	13491025.81
Tile 3	32523714.67	32531042.32	32514208.74
Tile 4	20469076.78	20465387.91	20453867.97

Table 2: Comparison of PSNR in dB for different variations in the GW algorithm on test image, Lena

Compression Ratio	Bit Rate (bpp)	GW [13]	Improved GW [14]	Hybrid GW [15]	Proposed method
256:1	0.03125	26.64	26.67	26.89	<b>27.84</b>
128:1	0.0625	28.72	28.78	28.93	<b>29.73</b>
64:1	0.125	30.73	30.82	31.20	<b>31.45</b>

Table 3: Comparison of PSNR in dB for different variations in the GW algorithm on Cameraman test image

Compression Ratio	Bit Rate (bpp)	GW [13]	Improved GW [14]	Hybrid GW [15]	Proposed method
128:1	0.0625	22.93	23.04	23.74	<b>24.55</b>
64:1	0.125	25.07	25.29	25.58	<b>26.31</b>
32:1	0.25	27.48	27.62	27.82	<b>28.38</b>



Figure 7. Top Left: Reconstructed Cameraman using the normal form of straight line in GW Algorithm, 0.0625 bpp, PSNR=22.93. Top Center: Reconstructed Cameraman using the slope-intercept form of straight line in GW Algorithm, 0.0625 bpp, PSNR=23.04. Top Right: Reconstructed Cameraman using the Proposed Method, 0.0625 bpp, PSNR=24.55. Bottom Left: Reconstructed Lena using the normal form of straight line in GW algorithm, 0.0625 bpp, PSNR=28.72. Bottom Center: Reconstructed Lena using the slope-intercept form of straight line in GW Algorithm, 0.0625 bpp, PSNR=28.78. Bottom Right: Reconstructed Lena using the Proposed Method, 0.0625 bpp, PSNR=29.73.



Figure 8. (a): Original Barbara (512x512). (b): Reconstructed Barbara using the normal form of straight line in GW Algorithm, 0.0625 bpp, PSNR=26.04. (c): Reconstructed Barbara using the slope-intercept form of straight line in GW Algorithm, 0.0625 bpp, PSNR=26.45. (d) Reconstructed Barbara using the Proposed Method, 0.0625 bpp, PSNR=27.32.

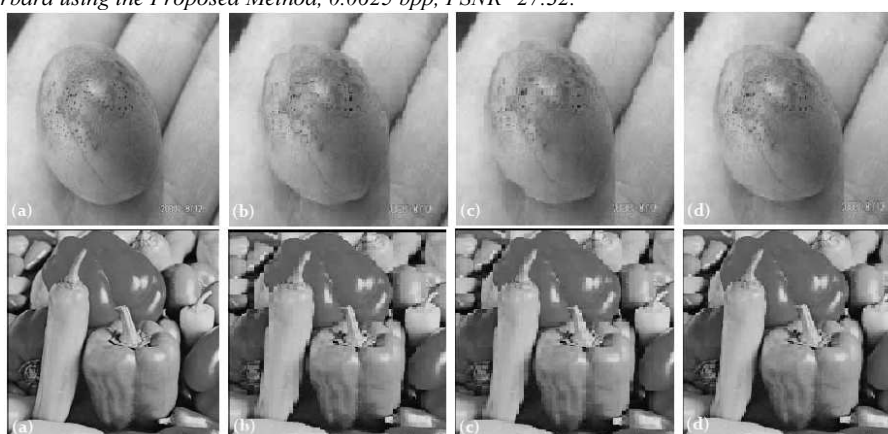


Figure 9. Top (a): Original Egg (256x256). (b): Reconstructed Egg using the normal form of straight line in GW Algorithm, 0.0625 bpp, PSNR=25.12. (c): Reconstructed Egg using the slope-intercept form of straight line in GW Algorithm, 0.0625 bpp, PSNR=25.65. (d) Reconstructed Egg using the Proposed Method, 0.0625 bpp, PSNR=26.58. Bottom (a): Original Vegetable (512x512). (b): Reconstructed Vegetable using the normal form of straight line in GW Algorithm, 0.0625 bpp, PSNR=23.24. (c): Reconstructed Vegetable using the slope-intercept form of straight line in GW Algorithm, 0.0625 bpp, PSNR=23.57. (d) Reconstructed Vegetable using the Proposed Method, 0.0625 bpp, PSNR=24.32.