Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 950357, 10 pages
http://dx.doi.org/10.1155/2015/950357

*Research Article*

# Hybrid Prediction and Fractal Hyperspectral Image Compression

**Shiping Zhu,[1] Dongyu Zhao,[1] and Fengchao Wang[2]**

[1] *Department of Measurement Control and Information Technology, School of Instrumentation Science and Optoelectronics Engineering, Beihang University, Beijing 100191, China*
[2] *CSR Qingdao Sifang Locomotive and Rolling Stock Co., Ltd., No. 88 Jinhong East Road, Chengyang District, Qingdao 266111, China*

Correspondence should be addressed to Shiping Zhu; spzhu@163.com

Received 16 July 2014; Revised 20 October 2014; Accepted 21 October 2014

Academic Editor: J. A. Tenreiro Machado

The data size of hyperspectral image is too large for storage and transmission, and it has become a bottleneck restricting its applications. So it is necessary to study a high efficiency compression method for hyperspectral image. Prediction encoding is easy to realize and has been studied widely in the hyperspectral image compression field. Fractal coding has the advantages of high compression ratio, resolution independence, and a fast decoding speed, but its application in the hyperspectral image compression field is not popular. In this paper, we propose a novel algorithm for hyperspectral image compression based on hybrid prediction and fractal. Intraband prediction is implemented to the first band and all the remaining bands are encoded by modified fractal coding algorithm. The proposed algorithm can effectively exploit the spectral correlation in hyperspectral image, since each range block is approximated by the domain block in the adjacent band, which is of the same size as the range block. Experimental results indicate that the proposed algorithm provides very promising performance at low bitrate. Compared to other algorithms, the encoding complexity is lower, the decoding quality has a great enhancement, and the PSNR can be increased by about 5 dB to 10 dB.

## 1. Introduction

Hyperspectral imaging technology is a major innovation in the course of development of remote sensing technology, which combines ground target spectra determined by material composition and the image reflecting ground target existing layout completely, and provides spectral information of dozens to hundreds of narrow bands while obtaining target spatial information for each pixel, thereby achieving a unity of image and spectra [1]. Hyperspectral remote sensing has been successfully applied in many areas such as geology, ecology, atmospheric study, and soil study and is playing an increasingly important role [2].

Hyperspectral image adds 3D spectral information based on 2D information about ground image. Hyperspectral data is a cube of a spectral image, whose spatial image dimension reflects the 2D spatial distribution of ground target features and whose spectral dimension reflects the spectral curve features. Spatial correlation exists in the spatial image dimension of hyperspectral image data and spectral correlation exists in the spectral dimension [3]. Accordingly, the difference between hyperspectral image compression and ordinary image compression is to remove both the spatial redundancies and the spectral redundancies. At present, the commonly used hyperspectral image compression methods include prediction-based, transformation-based, and vector quantization-based methods. For example, [4] used median prediction algorithm for hyperspectral intraband prediction and then used linear prediction and context prediction mixed algorithm for hyperspectral interband prediction, thereby implementing lossless compression of hyperspectral images. Penna et al. [5] used point extraction strategy to reduce the calculated amount of KLT and applied this improved KLT to the decorrelation of hyperspectral images, without having a significant impact on reconstructed image quality. Qian [6] proposed a fast vector quantization algorithm to improve the generation efficiency of codebook, which did not require full search, thereby greatly reducing the complexity.

In addition to the above conventional coding methods, fractal coding method characterized by its innovative ideas, high compression ratio, and resolution independence is considered as one of the most promising second-generation compression coding methods. Our previous studies of fractal video compression [7, 8] also demonstrate its good compression performance. Fractal coding techniques are based on the theory of iterated function systems (IFS) and are further developed by Barnsley in the early 1980s [9, 10]. The IFS theory is based on Banach's contraction mapping principle, which states that a contractive transformation, defined on a complete metric space, possesses a unique fixed point or attractor. For the purpose of image compression, this idea translates into finding an optimal contractive transformation whose attractor closely approximates a given target image. This problem is widely known as the inverse problem in the fractal image coding literature. But this must involve manual intervention and it is extremely hard to find a contractive transformation with an attractor approximating the whole image. In the late 1980s, Jacquin developed a block-based fractal image compression scheme exploiting local self-similarities that are inherent in many real-world images [11] and made fractal image compression become automatic. In his scheme, the image is subdivided into a pair of simple and uniform partitions: a domain partition of larger subblocks, also known as parent subblocks, and a range partition of smaller subblocks, also known as child subblocks. A parent subblock is mapped into its corresponding child subblock using a geometric mapping, followed by a simple affine transformation, known as the gray-level map. Therefore, a digitized image can be stored as a collection of the position of the corresponding parent subblock and transformation coefficients for each child subblock. This generally requires much less memory, resulting in data compression.

We observe that there exist strong similarities between adjacent bands of hyperspectral images, so that we guess that fractal coding will be suitable for hyperspectral image compression. In this paper, we make a try to combine prediction and fractal coding for hyperspectral image compression. The algorithm first carries out intraband prediction for the first band which is treated as the basic band, and we extend the block-based fractal image compression scheme to two adjacent bands, which is to say each range block is approximated by the domain block in the adjacent band, which is of the same size as the range block to remove the spectral redundancies. The experimental results with AVIRIS (airborne visible infrared imaging spectrometer) hyperspectral remote sensing images indicate that the algorithm provides very promising performance at low bitrate.

The rest of the paper is organized as follows. The basis of fractal image coding is summarized in Section 2. The proposed hybrid prediction and fractal hyperspectral image compression algorithm is presented in Section 3. The experimental results are presented in Section 4. And finally the conclusions are outlined in Section 5.

## 2. Fractal Image Coding Basis

*2.1. Fractal Image Coding Theory.* The fundamental principle of fractal coding consists of the representation of an image by a contractive transform of which the fixed point is close to that image [12]. The image space is accepted as a complete metric space. By the contractive mapping theorem and collage theorem, the contractive transform is always possible within certain threshold. Original approach with IFS tries to find a number of affine mappings on the entire image, which is rather slow in terms of searching the contractive map function. Jacquin's partitioned iteration function system (PIFS) [13] takes a different approach by finding the individual mappings for subsets of the images. Each PIFS contains a complete metric space $(X, d)$ and a series of contraction mapping $\omega_i : X \rightarrow X$ as defined in this space. Fractal image coding just uses a PIFS to represent an original image so that the image after iterative decoding closely approximates the attractor of this PIFS as well as the original image. The contraction mapping coefficients constitute the fractal code of the original image.

The attractor theorem is as follows.

*Attractor Theorem.* Assuming $\{X : \omega_1, \omega_2, \omega_3, \ldots, \omega_n\}$ is an IFS in the completely metric space $(X, d)$, then we have the following.

(1) Transformation $W : H(x) \rightarrow H(x)$ defined by the following equation:

$$W(B) = \bigcup_{i=1}^{n} \omega_n(B), \quad \forall B \in H(x), \tag{1}$$

is the contraction mapping of complete metric space $(H(x), h)$, whose contractive factor is

$$\begin{gathered} s = \max(s_1, s_2, \ldots, s_n), \\ h(W(B), W(C)) \leq s \times h(B, C), \quad \forall B, C \in H(x), \end{gathered} \tag{2}$$

where $h$ represents the Hausdorff metric.

(2) Compression transformation $W$ has a unique fixed point $\overline{A} \in H(x)$, so that it satisfies the following equation:

$$\overline{A} = W(\overline{A}) = \bigcup_{i=1}^{n} \omega_i(\overline{A}). \tag{3}$$

And the fixed point can be obtained by iteration, namely:

$$\overline{A} = \lim_{n \to \infty} W^n(B), \quad \forall B \in H(x), \tag{4}$$

where

$$W^0(B) = W(B), \qquad W^n(B) = W(W^{n-1}(B)). \tag{5}$$

Fixed point $\overline{A} \in H(x)$ is called the attractor of IFS.

That is to say, for any set $B$, iterative sequence $W^n(B)$ converges to the attractor of PIFS. Its inverse problem takes the given original image as the attractor to solve its contraction mapping. It is very difficult to solve the inverse problem accurately, but collage theorem gives out a possible

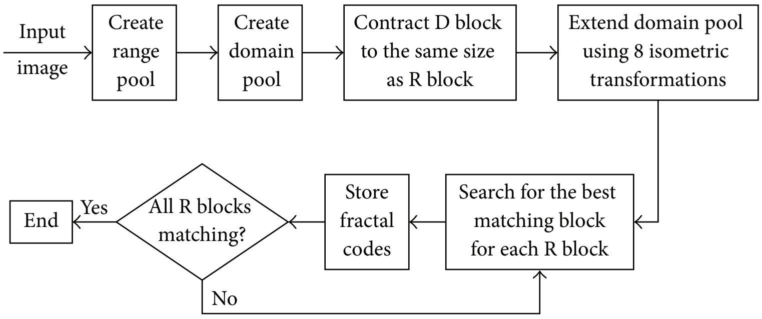

Input image
Create range pool
Create domain pool
Contract D block to the same size as R block
Extend domain pool using 8 isometric transformations
Search for the best matching block for each R block
Store fractal codes
All R blocks matching?
Yes
No
End

Figure 1: Flow chart of the basic block-based fractal image compression.

solution to this problem. That is, approximately minimize the distance $d(\mu_{\text{org}}, \mu_{\text{fix}})$ between original image $\mu_{\text{org}}$ and attractor image $\mu_{\text{fix}}$ by minimizing the distance $d(\mu_{\text{org}}, W\mu_{\text{org}})$ between original image $\mu_{\text{org}}$ and its collage image $W\mu_{\text{org}}$.

The collage theorem is as follows.

*Collage Theorem.* Assuming $(X, d)$ is a complete metric space, given a set $L \in H(x)$ and a number $\varepsilon > 0$, if we can find a IFS$\{X; \omega_1, \omega_2, \omega_3, \ldots, \omega_n; s\}$ $(0 \le s < 1)$, which makes

$$h\left(L, \bigcup_{i=1}^{n} \omega_i(L)\right) \le \varepsilon, \tag{6}$$

then

$$h\left(L, \overline{A}\right) \le \frac{h\left(L, \bigcup_{i=1}^{n} \omega_i(L)\right)}{1-s} \le \frac{\varepsilon}{1-s}, \tag{7}$$

where $h$ is Hausdorff metric and $\overline{A}$ is attractor of this IFS.

*2.2. Fractal Image Coding Implementation.* The basic block-based fractal image coding algorithm flow chart is shown in Figure 1.

Firstly, divide the original input image into nonoverlapping $R$ blocks to form $R$ block set $\{R\}$ and overlapped $D$ blocks to form $D$ block set $\{D\}$, with size of each $R$ block at $N \times N$ and that of each $D$ block at $2N \times 2N$. $R$ blocks and $D$ blocks are usually obtained by sliding window method.

Then, contract all $D$ blocks to the size of $N \times N$, that is, the same size as $R$ blocks, usually by downsampling or averaging the intensities of its four neighboring pixels,

$$\begin{aligned} d(x, y) &= \left(d(2x, 2y) + d(2x+1, 2y) + d(2x, 2y+1)\right. \\ &\quad \left.+ d(2x+1, 2y+1)\right) \times \frac{1}{4}, \end{aligned} \tag{8}$$

where $d(x, y)$ represents the intensity of $D$ block at position $(x, y)$.

Afterwards, contracted $D$ block is extended through 8 isometric transformations (including identical transformation; rotations by 90°, 180°, and 270° about the block center; symmetric reflections about vertical central axis, horizontal central axis, leading diagonal, and secondary diagonal). The extended domain pool is defined as $\{\widetilde{D}\}$.

In order to find the best matching block for each $R$ block, contraction mapping transformation $\{W_i\}$ is defined as

$$W_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix}, \tag{9}$$

where $x$ and $y$ represent spatial coordinates; $z$ represents pixel value; $a_i$, $b_i$, $c_i$, and $d_i$ define one of the 8 isometric transformations; $s_i$ represents brightness adjustment factor with its absolute value smaller than 1; $o_i$ represents brightness offset factor.

Searching for matching block is a process of minimizing the following equation:

$$E\left(R, \widetilde{D}\right) = \left\| R - \left(s \cdot \widetilde{D} + o \cdot I\right) \right\|, \tag{10}$$

where $\|\cdot\|$ represents vector 2-norm.

Finally, use (9) and (10) to find the best matching block from the extended domain pool $\{\widetilde{D}\}$ for each $R$ block and store the sequence number of isometric transformation, location of the best matching $D$ block, and corresponding brightness scale factor $s$ and offset factor $o$. After fractal parameters of all $R$ blocks are stored, the total fractal encoding process is completed.

Fractal image decoding process is very simple, which can be completed by only inputting an arbitrary initial image (initial size of the image should be equal to that of its original image as required) and then iteratively applying the corresponding transformation to initial image based on the stored fractal parameters. Decoding process can be expressed as follows:

$$\begin{aligned} R_i^{(k)} &= s_i \cdot t_l \circ S\left(D_{m(i)}^{(k-1)}\right) + o_i \cdot I, \\ \mu^{(k)} &= \bigcup_i R_i^{(k)}, \end{aligned} \tag{11}$$

where $R_i^{(k)}$ represents the $i$th $R$ block of image $\mu^{(k)}$ after $k$ times' iterations; $D_{m(i)}^{(k-1)}$ represents the best matching block of $R_i^{(k)}$ and is derived from image $\mu^{(k-1)}$ after $k-1$ times' iterations; $D_{m(i)}^{(0)}$ is derived from the best matching $D$ block of the initial image; $s_i$ and $o_i$ represent the corresponding optimum brightness adjustment factor and offset factor, respectively; $S$ and $t_l$ represent spatial contraction transformation and isometric transformation, respectively. The symbol $\circ$ is the composition operator.

If $k$ reaches a predetermined number of iterations $N$ (generally less than 10), then stop iteration and output image $\mu^{(N)}$. This is the decoded image, namely, approximation of the image to be encoded.

## 3. Hybrid Prediction and Fractal Hyperspectral Image Compression

Because each band of a hyperspectral image corresponds to the same location on the earth, hyperspectral data cube has a strong spectral correlation, whose spatial correlation is relatively weak. The basic fractal image coding algorithm does not use spectral correlation, which therefore cannot be directly used for compressing hyperspectral images. In order to take full advantage of spatial and spectral correlations of hyperspectral image, we propose a method for lossy compression of hyperspectral image based on hybrid prediction and fractal. The first band is intraband prediction encoded, and the remaining bands are interband fractal encoded. The difference is that each range block is approximated by the domain block in the adjacent decoded band, which is of the same size as the range block to exploit the spectral correlation. Then the prediction errors as well as fractal residual are further transformed, quantified, and entropy encoded, and the fractal parameters are also entropy encoded to improve the compression efficiency. The compression scheme divides each band into range blocks with the size of 16 × 16. The block diagram for this scheme is shown in Figure 2.

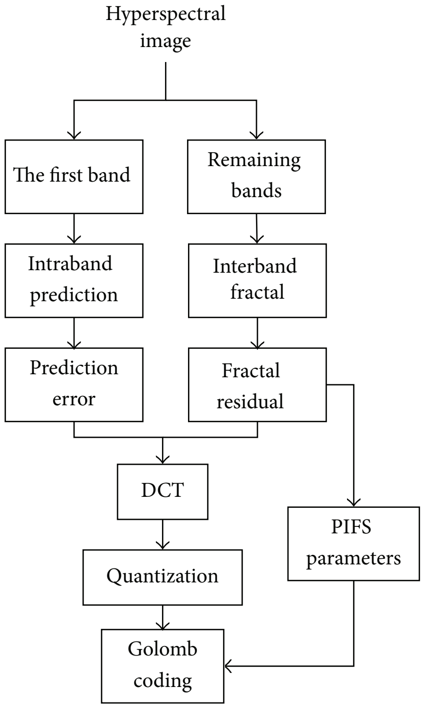


FIGURE 2: Block diagram of the proposed scheme.

*3.1. Intraband Prediction.* Linear prediction has been studied in hyperspectral image compression [14–16]. In this section, we describe the intraband prediction design of our scheme. The main purpose is to remove spatial correlation as well as obtaining a high quality decoded reference band.

Hyperspectral images are usually texture abundant. For example, Figure 3(a) shows a simulated color IR view of an airborne hyperspectral data flight line over the "Washington DC Mall," which has much detail. The sensor system used in this case measured pixel response in 210 bands in the 0.4 to 2.4 $\mu$m region of the visible and infrared spectrum. Bands in the 0.9 and 1.4 $\mu$m region where the atmosphere is opaque have been omitted from the dataset, leaving 191 bands. The dataset contains 1208 scan lines with 307 pixels in each scan line. The color image is made using bands 60, 27, and 17 for the red, green, and blue colors, respectively. Figure 3(b) is the first band of the hyperspectral image.

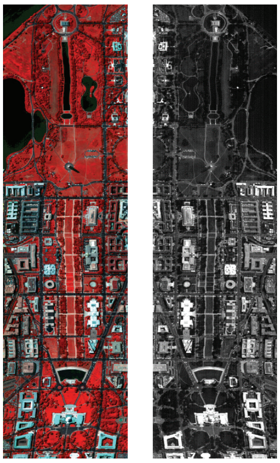
To predict more precisely, we design 9 linear predictors with different prediction directions, as shown in Figure 4. Eight prediction modes, as shown in Figure 4(a), are utilized for more suitably predicting textures with structures in the respective directions. The 16 samples of the 4 × 4 block, which is labeled as a–p, are predicted using the prior decoded neighboring pixels in the corresponding directions in the adjacent blocks labeled as A–Q, which is shown in Figure 4(b). In addition, the DC mode is used to predict smooth area, and each pixel in the shadowed block is predicted using the mean value of the neighboring pixels labeled as A–D and I–L, which is shown in Figure 4(c).

The prediction mode with the minimum sum of squared difference (SSD) is chosen as the final prediction mode. The SSD is calculated as follows:

$$\mathrm{SSD}(s,c) = \sum_{x=1}^{4}\sum_{y=1}^{4}\left(s(x,y) - c(x,y)\right)^2, \tag{12}$$

where $s(x, y)$ represents the source signal at position $(x, y)$; $c(x, y)$ represents the reconstruction signal at position $(x, y)$ using the corresponding prediction mode.

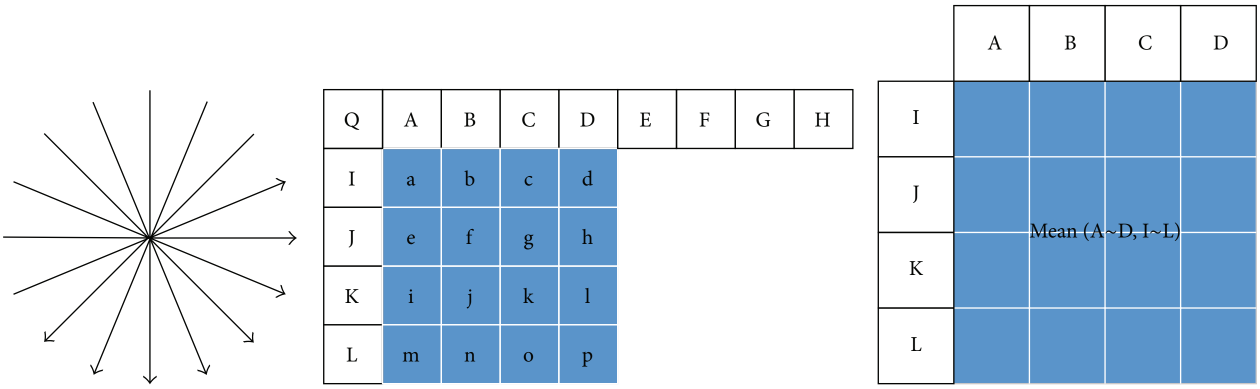
*3.2. Interband Fractal Coding.* In order to remove the existing spectral redundancy in hyperspectral image, we have made some modifications to the basic block-based fractal image compression scheme. In our scheme, each range block is approximated by the domain block in the adjacent band, which is of the same size as the range block. And considering that the adjacent bands have similar spatial topology structures, as shown in Figure 5, thus isometric transformation process can be omitted. Statistics also show that the best matching $D$ block appears near $R$ block with a large probability [17], so it is not necessary to use full search. In our scheme, the search range of block matching is reduced to a small region centering the corresponding position of $R$ block, as shown in Figure 6. In addition, we adopt an adaptive block size instead of the original fixed size block partition method.



(a) The simulated color view with bands 60, 27, and 17

(b) The first band image

Figure 3: Hyperspectral data of the "Washington DC Mall."

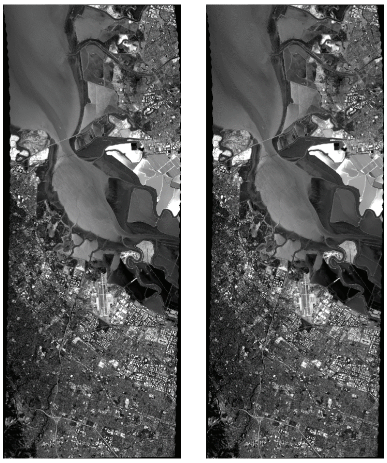


(a) 8 prediction directions

(b) Directional prediction

(c) DC prediction

Figure 4: Predictors for intraband prediction.



(a) The 25th band

(b) The 26th band

Figure 5: Two adjacent bands of hyperspectral image "Moffett Field."

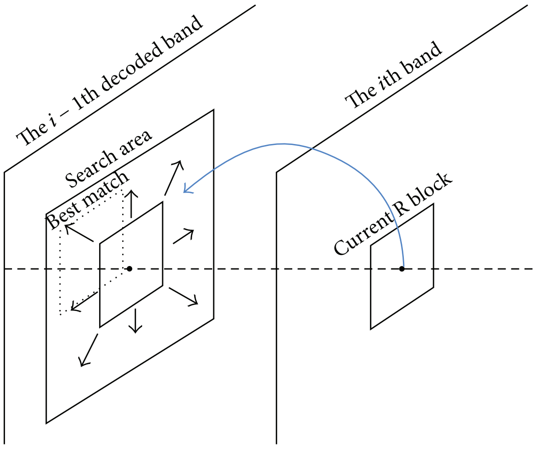


Figure 6: Sketch map of block matching.

The specific steps for interband fractal coding are as follows.

*Step 1.* The current hyperspectral band image is divided into nonoverlapping $R$ blocks with the size of 16 × 16, and all $R$ blocks can cover the whole band image.

*Step 2.* Encode each $R$ block in turn.

(1) The searching area is located in the adjacent decoded band, and with the same coordinate of the current $R$ block as the center, move 7 pixels (a variable parameter determined at the encoder) upward, downward, and towards left and right, to form the rectangular search area, as shown in Figure 6. Search the best matching $D$ block in the search area, with the MSE (mean square error) as the matching criterion:

$$\mathrm{MSE}(s,o)=\frac{1}{N}\sum_{i=1}^{N}\left[r_i-\left(s\cdot d_i+o\right)\right]^2, \tag{13}$$

where $r_i$ represents the pixel value of the current $R$ block, $d_i$ represents the pixel value of the corresponding matching block, $N$ represents the number of pixels of the current $R$ block, $s$ is the scale factor, and $o$ is the offset factor. $s$ and $o$ are obtained as follows.

Solving the equation for $\mathrm{MSE}(s,o)$,

$$\frac{\partial \mathrm{MSE}(s,o)}{\partial s}=\frac{1}{N}\sum_{i=1}^{N}2\left(s\cdot d_i+o-r_i\right)\cdot d_i. \tag{14}$$

Setting to 0,

$$\begin{gathered}0=s\sum_{i=1}^{N}d_i^2+o\sum_{i=1}^{N}d_i-\sum_{i=1}^{N}r_i\cdot d_i,\\ o=\frac{1}{\sum_{i=1}^{N}d_i}\left(\sum_{i=1}^{N}r_i\cdot d_i-s\sum_{i=1}^{N}d_i^2\right).\end{gathered} \tag{15}$$

Similarly,

$$\frac{\partial \mathrm{MSE}(s,o)}{\partial o}=\frac{1}{N}\sum_{i=1}^{N}2\left(s\cdot d_i+o-r_i\right)\cdot 1. \tag{16}$$

Setting to 0,

$$\begin{gathered}0=s\sum_{i=1}^{N}d_i+\sum_{i=1}^{N}o-\sum_{i=1}^{N}r_i,\\ s\sum_{i=1}^{N}d_i=\sum_{i=1}^{N}r_i-N\cdot o.\end{gathered} \tag{17}$$

Substituting,

$$\begin{gathered}s=\frac{N\left(\sum_{i=1}^{N}r_i\cdot d_i\right)-\sum_{i=1}^{N}r_i\sum_{i=1}^{N}d_i}{N\sum_{i=1}^{N}d_i^2-\left(\sum_{i=1}^{N}d_i\right)^2},\\ o=\frac{1}{N}\left(\sum_{i=1}^{N}r_i-s\sum_{i=1}^{N}d_i\right).\end{gathered} \tag{18}$$

If the minimum MSE is less than T_16 (here is 10.5), the domain block can be considered as well-matched, and we can record the current IFS coefficients and continue to the next $R$ block; otherwise, go to (2).

(2) The $R$ block is divided into two 16 × 8 $R$ subblocks. Search the best matching block for each 16 × 8 $R$ subblock, respectively, in the domain pool, and the size of searched matching block is 16 × 8. If the minimum MSEs of the two 16 × 8 $R$ subblocks are both smaller than T_16, the domain block can be considered as well-matched, and we can record the IFS coefficients of the two 16 × 8 $R$ subblocks and continue to the next $R$ block; otherwise, go to (3).

(3) The $R$ block is divided into two 8 × 16 $R$ subblocks. Search the best matching block for each 8 × 16 $R$ subblock, respectively, in the search area, and the size of searched matching block is 8 × 16. If the minimum MSEs of the two 8 × 16 $R$ subblocks are both smaller than T_16, the domain block can be considered as well-matched, and we can record the IFS coefficients of the two 16 × 8 $R$ subblocks and continue to the next $R$ block; otherwise, go to (4).

(4) This $R$ block is divided into four 8 × 8 $R$ subblocks for which, respectively, search for the best matching block also in the size of 8 × 8 in the search area, and if the minimum MSEs of all four 8 × 8 $R$ subblocks are smaller than T_8 (here is 8.0), we can record the IFS coefficients of the four 8 × 8 $R$ subblocks and continue to the next $R$ block; otherwise, we can further divide the 8 × 8 $R$ subblock which is not well matched into two 8 × 4 $R$ sub-subblocks or two 4 × 8 $R$ sub-subblocks or four 4 × 4 sub-subblocks just in the same way.

*Step 3.* Repeat Step 2 until all $R$ blocks of the current hyperspectral band have found the best matching blocks.

The flowchart is shown in Figure 7.

*3.3. Residual Image and Fractal Parameters Encoding.* In a general prediction encoding system, once the prediction is formed, it is subtracted from the original band, and the residual is usually further compressed using transformation encoding and entropy encoding. In our scheme, the residual image of the first hyperspectral band is obtained through subtracting the predicted band image by the original band

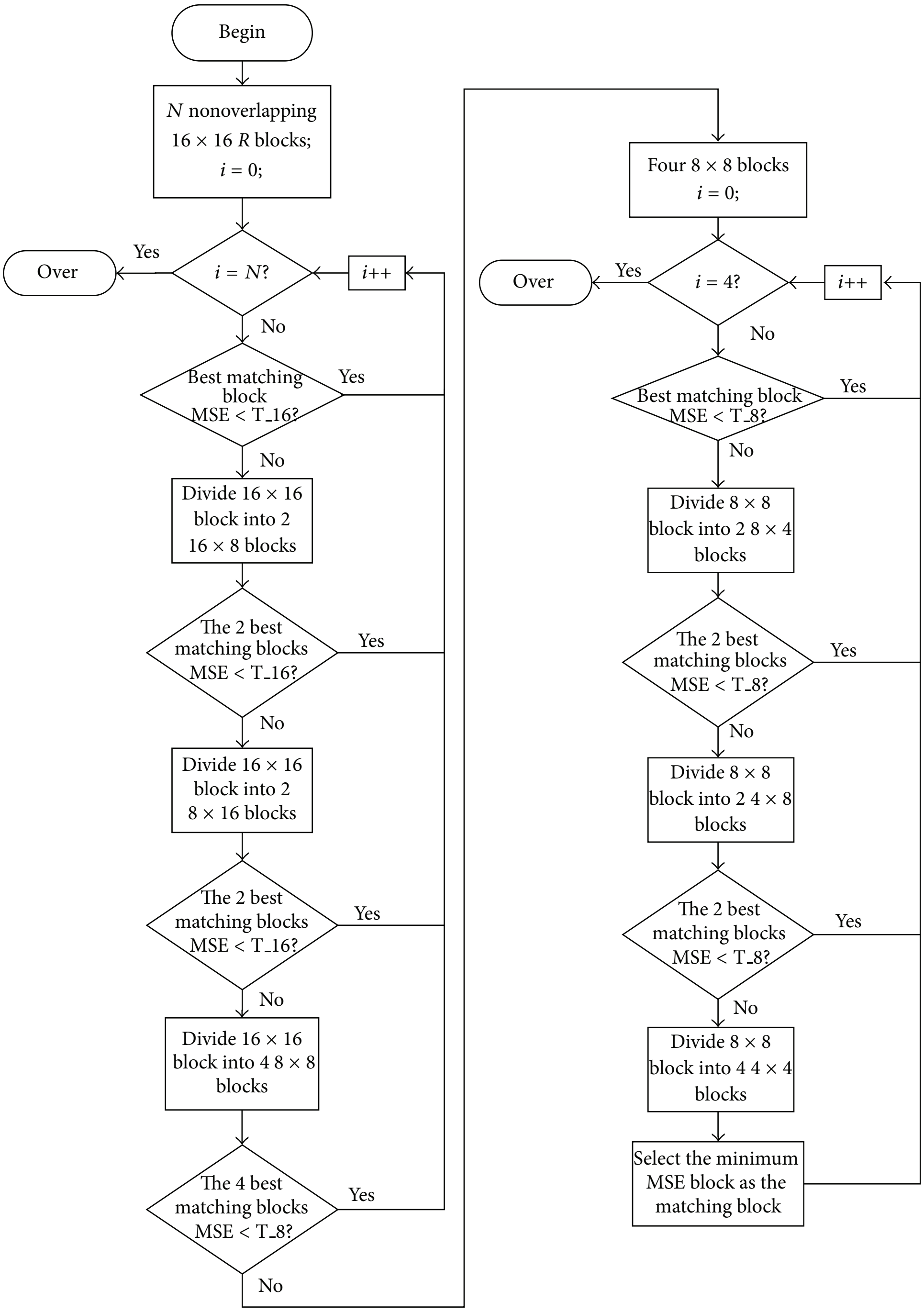

Begin
$N$ nonoverlapping 16 × 16 $R$ blocks; $i = 0$;
$i = N$?
Yes
Over
No
$i$++
Best matching block MSE < T_16?
Yes
No
Divide 16 × 16 block into 2 16 × 8 blocks
The 2 best matching blocks MSE < T_16?
Yes
No
Divide 16 × 16 block into 2 8 × 16 blocks
The 2 best matching blocks MSE < T_16?
Yes
No
Divide 16 × 16 block into 4 8 × 8 blocks
The 4 best matching blocks MSE < T_8?
Yes
No
Four 8 × 8 blocks $i = 0$;
$i = 4$?
Yes
Over
No
$i$++
Best matching block MSE < T_8?
Yes
No
Divide 8 × 8 block into 2 8 × 4 blocks
The 2 best matching blocks MSE < T_8?
Yes
No
Divide 8 × 8 block into 2 4 × 8 blocks
The 2 best matching blocks MSE < T_8?
Yes
No
Divide 8 × 8 block into 4 4 × 4 blocks
Select the minimum MSE block as the matching block

FIGURE 7: Flowchart of block matching process.

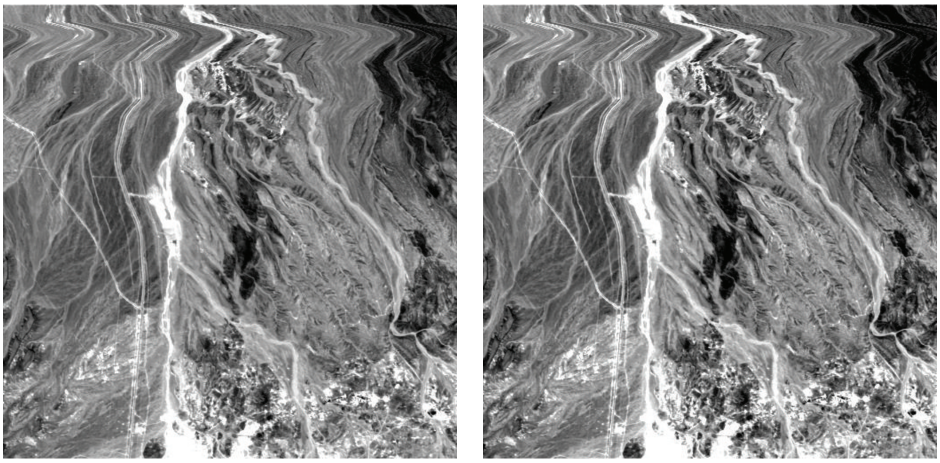
image. In addition, the residual of fractal encoding is also processed in the same way, while it is ignored in the basic fractal encoding algorithm and most fractal literatures. The residual image of the fractal encoding is formed by

$$\text{residual} = \sum \left(r_i - \left(s \cdot d_i + o\right)\right). \tag{19}$$

The residual image is DCT transformed, quantized, and Golomb entropy encoded. In parallel, the quantized data are rescaled and inverse transformed and added to the prediction band to reconstruct a coded version of the band which is stored for later interband fractal encoding. The fractal parameters are also Golomb entropy encoded.

## 4. Experimental Result

Two AVIRIS [18] hyperspectral data cubes, “Cuprite” and “Low Altitude,” derived from JPL are used. The size of the data cubes was originally 512 lines × 614 pixels with 224 spectral bands and 1087 lines × 614 pixels with 224 spectral bands,



(a) Original image

(b) Decoded image

Figure 8: The 17th band subimage of "Cuprite."

respectively. In this paper, a subset of them with 512 lines × 512 pixels with all 224 bands was tested. The experiments were performed on a computer with an Intel Core 2 Quad Q9300 @ 2.50 GHz CPU and 4 GB RAM. PSNR is used to measure the quality of the compression data. It is defined as follows:

$$\mathrm{PSNR} = 10\log_{10}\frac{\mathrm{Peaksignal}^2}{\mathrm{MSE}}, \tag{20}$$

where Peaksignal is the maximum value in the data cube and MSE is calculated as

$$\mathrm{MSE} = \frac{1}{n_x n_y n_b}\sum_{x=1}^{n_x}\sum_{y=1}^{n_y}\sum_{b=1}^{n_b}\left[O\left(x,y,b\right) - D\left(x,y,b\right)\right], \tag{21}$$

where $O(x,y,b)$ and $D(x,y,b)$ are, respectively, the pixel values of the original and the reconstructed data of band $b$ at location $(x,y)$, $n_y$ is the total number of lines in the data cube, $n_x$ is the total number of pixels per line, and $n_b$ is the total number of bands in the data cube.

Figure 8(a) shows the 17th band of "Cuprite" truncated from the original data cube for the experiment, and Figure 8(b) shows the decoded result image at 0.1 bpppb. As seen in Figure 8, there is almost no difference of subjective quality between the compressed image and the original image.

Figure 9 shows the PSNR comparison of "Low Altitude" encoded by the 3D-SPIHT [21] and the proposed algorithm. 3D-SPIHT coding is the three-dimensional extension version of the set partitioning in hierarchical trees (SPIHT) coding, which is based on the concentrative energy of wavelet coefficients, to exhibit symmetric branching in all three dimensions. More details can be found in [21]. It can be seen from Figure 9 that the PSNR of the first band by the algorithm of the article is higher than every other band, indicating that the prediction algorithm adopted obtains high encoding quality. The PSNRs of all bands of "Low Altitude" compressed by the proposed algorithm are higher than those in the 3D-SPIHT algorithm. It is calculated that the average value of PSNRs of all bands of "Low Altitude" compressed by the proposed algorithm is 60.07 dB, which is 15.91 dB higher than that of the 3D-SPIHT algorithm.

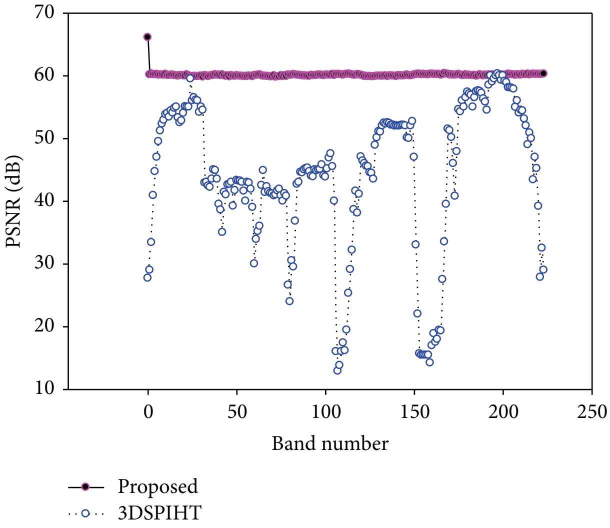
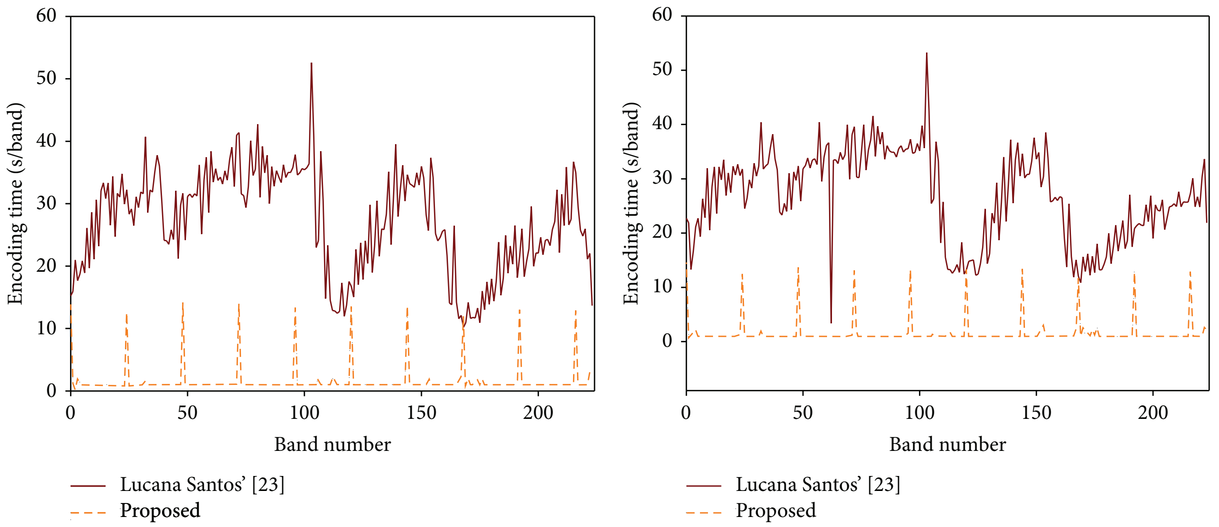


Figure 9: PSNR comparison of each band for "Low Altitude" at the compression ratio of 32 : 1.

The PSNR comparison results at different bitrates are as shown in Tables 1 and 2. The AT-3DSPIHT [19] algorithm in Tables 1 and 2 is an improvement of 3D-SPIHT algorithm by means of asymmetric structure, having a longer tree along the wavelet axis than that of the traditional tree structure, and the PSNR performance of AT-3DSPIHT for hyperspectral image is better than 3D-SPIHT. More details can be found in [19]. Experimental results show that the PSNR of "Low Altitude" compressed by the proposed algorithm is slightly lower than that of AT-3DSPIHT at 0.1 bpppb, except that they are all higher than the contrast algorithms. In comprehensive consideration of all bitrates, compared with AT-3DSPIHT algorithm, the PSNR of the proposed algorithm is increased



(a) Yellowstone_sc0

(b) Yellowstone_sc3

Figure 10: Encoding time comparison.

Table 1: PSNR comparison for "Low Altitude" at different bitrates (unit: dB).

| | Code rate (bpp) | | | |
|---|---|---|---|---|
| | 0.1 | 0.15 | 0.2 | 0.25 |
| AT-3DSPIHT [19] | 47.50 | 49.51 | 51.50 | 52.61 |
| F. Zhao et al. [20] | 41.62 | 43.33 | 44.95 | 46.08 |
| Proposed | 46.50 | 54.37 | 57.21 | 60.07 |

Table 2: PSNR comparison for "Cuprite" at different bitrates (unit: dB).

| | Code rate (bpp) | | | |
|---|---|---|---|---|
| | 0.05 | 0.1 | 0.15 | 0.2 |
| AT-3DSPIHT [19] | 44.25 | 45.47 | 46.72 | 48.05 |
| F. Zhao et al. [20] | 48.09 | 48.88 | 49.60 | 50.31 |
| Proposed | 49.44 | 60.08 | 66.51 | 67.65 |

by 9.53 dB averagely, and compared with the algorithm in [20], it is averagely increased by 11.12 dB.

In [22], a compression algorithm of hyperspectral remote sensing image based on 3D wavelet transform and 3D fractal is proposed. The classical eight kinds of affine transformations in 2D fractal image compression are generalized to nineteen for the 3D fractal image compression. When the compression ratio on the Earth Observation-1 (EO-1) satellite hyperspectral image of China Dongting Lake area is 80 : 1, the PSNR is 36.78 dB, while the PSNR of our proposed scheme at the same compression ratio is 41.78 dB. We can achieve a compression quality improvement of 5 dB.

To evaluate the encoding complexity of the proposed algorithm, we compare the encoding time of the proposed algorithm with that of Lucana Santos' [23]. The corresponding encoding time by the two schemes for "Yellowstone_sc0" and "Yellowstone_sc3" is recorded in Figure 10, which is displayed in seconds/band. Both scenes have the size of 512 lines × 680 pixels with 224 bands. From these results, we can see that the proposed algorithm has lower complexity and our scheme saves total 94.1% encoding time on average.

## 5. Conclusion

A lossy compression algorithm of hyperspectral image based on hybrid prediction and fractal is proposed. Intraband predictors with different directions are designed to remove the spatial correlation of the first band, and a high decoding quality can be obtained, which provides a good basic band for the following fractal encoding. The remaining bands are encoded by interband fractal encoding. The spectral redundancy can be removed since each range block is approximated by the domain block in the adjacent band, which is of the same size as the range block. Adaptive block partitions as well as a local search are also adopted in the fractal encoding. Finally, the residuals of prediction and fractal encoding are DCT transformed and Golomb entropy encoded. The fractal parameters are also Golomb entropy encoded.

Experimental results show that the algorithm achieves effective compression of hyperspectral image at low bitrate. The decoded quality is greatly improved, and the PSNR performance is increased by about 5 to 10 dB. Besides, the encoding complexity also has a great advantage. The proposed hybrid prediction and fractal hyperspectral image compression algorithm is a breakthrough to the traditional algorithm framework of hyperspectral image compression, providing a commendable solution for lossy compression of hyperspectral image.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This project is funded by the National Natural Science Foundation of China (NSFC) under Grants nos. 61375025, 61075011, and 60675018 and also by the Scientific Research Foundation for the Returned Overseas Chinese Scholars from the State Education Ministry of China. The authors would also like to express their appreciations to the reviewers for their thorough review and very helpful comments, which help in improving this paper.

## References

[1] Q. Du, N. Raksuntorn, S. Cai, and R. J. Moorhead, "Color display for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 6, pp. 1858–1866, 2008.

[2] N. R. M. Noor and T. Vladimirova, "Investigation into lossless hyperspectral image compression for satellite remote sensing," *International Journal of Remote Sensing*, vol. 34, no. 14, pp. 5072–5104, 2013.

[3] A. Karami, M. Yazdi, and G. Mercier, "Compression of hyperspectral images using discerete wavelet transform and tucker decomposition," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 444–450, 2012.

[4] Y. Liang, J. Li, and K. Guo, "Lossless compression of hyperspectral images using hybrid context prediction," *Optics Express*, vol. 20, no. 7, pp. 8199–8206, 2012.

[5] B. Penna, T. Tillo, E. Magli, and G. Olmo, "Transform coding techniques for lossy hyperspectral data compression," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 5, pp. 1408–1421, 2007.

[6] S.-E. Qian, "Hyperspectral data compression using a fast vector quantization algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 8, pp. 1791–1798, 2004.

[7] S. Zhu, Y. Hou, Z. Wang, and K. Belloulata, "Fractal video sequences coding with region-based functionality," *Applied Mathematical Modelling*, vol. 36, no. 11, pp. 5633–5641, 2012.

[8] S. Zhu, L. Li, J. Chen, and K. Belloulata, "An automatic region-based video sequence codec based on fractal compression," *AEU—International Journal of Electronics and Communications*, vol. 68, no. 8, pp. 795–805, 2014.

[9] M. F. Barnsley, *Fractals Everywhere*, Academic Press, Boston, Mass, USA, 1988.

[10] M. F. Barnsley and S. Demko, "Iterated function systems and the global construction of fractals," *Proceedings of the Royal Society A*, vol. 399, no. 1817, pp. 243–275, 1985.

[11] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions of Image Processing*, vol. 1, no. 1, pp. 18–30, 1992.

[12] B. Wohlberg and G. de Jager, "A review of the fractal image coding literature," *IEEE Transactions on Image Processing*, vol. 8, no. 12, pp. 1716–1729, 1999.

[13] Y. Fisher, *Fractal Image Compression-Theory and Application*, Springer, New York, NY, USA, 1994.

[14] R. E. Roger and M. C. Cavenor, "Lossless compression of AVIRIS images," *IEEE Transactions on Image Processing*, vol. 5, no. 5, pp. 713–719, 1996.

[15] A. C. Miguel, R. E. Ladner, and E. A. Riskiny, *Predictive Coding of Hyperspectral Images*, Springer, New York, NY, USA, 2006.

[16] Y. Zhang and M. D. Desai, "Hyperspectral image compression based on adaptive recursive bidirection prediction/JPEG," *Pattern Recognition*, vol. 33, no. 11, pp. 1851–1861, 2000.

[17] H. Hartenstein, M. Ruhl, and D. Saupe, "Region-based fractal image compression," *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1171–1184, 2000.

[18] http://aviris.jpl.nasa.gov/.

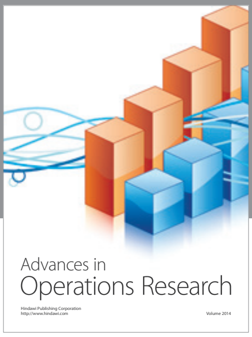
[19] X. Tang, S. Cho, and W. A. Pearlman, "3D set partitioning coding methods in hyperspectral image compression," in *Proceedings of the International Conference on Image Processing (ICIP '03)*, pp. 239–242, Barcelona, Spain, September 2003.

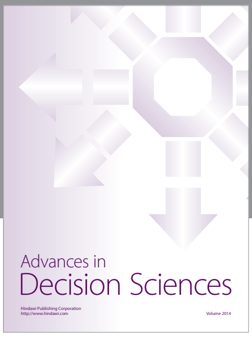
[20] F. Zhao, G. Liu, and X. Wang, "An efficient macroblock-based diverse and flexible prediction modes selection for hyperspectral images coding," *Signal Processing: Image Communication*, vol. 25, no. 9, pp. 697–708, 2010.

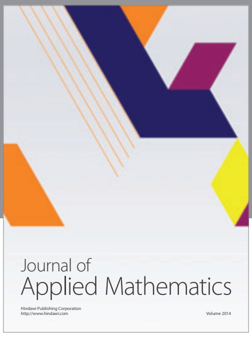
[21] S. Lim, K. Sohn, and C. Lee, "Compression for hyperspectral images using three dimensional wavelet transform," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS '01)*, pp. 109–111, July 2001.

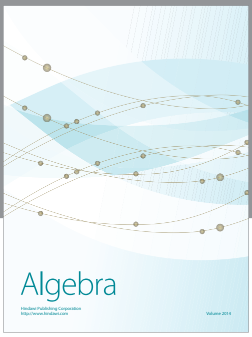
[22] W. Pan, Y. Zou, and L. Ao, "A compression algorithm of hyperspectral remote sensing image based on 3-D wavelet transform and fractal," in *Proceedings of 3rd International Conference on Intelligent System and Knowledge Engineering (ISKE '08)*, pp. 1237–1241, November 2008.

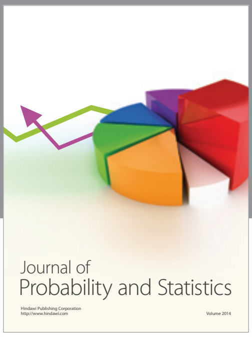
[23] L. Santos, S. López, G. M. Callicó, J. F. López, and R. Sarmiento, "Performance evaluation of the H.264/AVC video coding standard for lossy hyperspectral image compression," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 5, no. 2, pp. 451–461, 2012.


Advances in
Operations Research

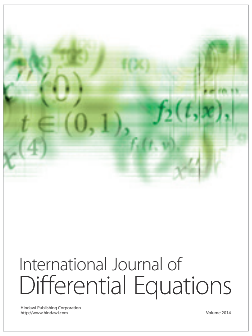

Advances in
Decision Sciences


Journal of
Applied Mathematics

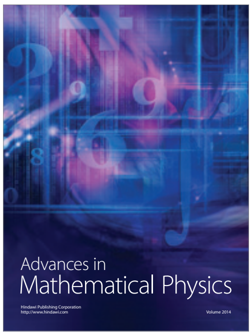

Algebra

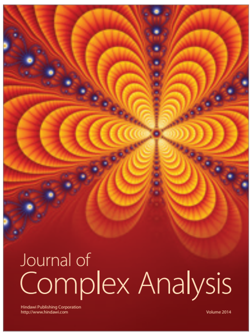

Journal of
Probability and Statistics

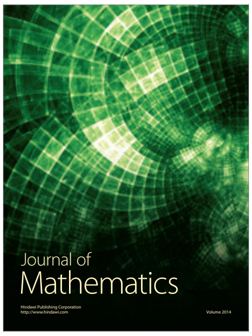
The Scientific
World Journal

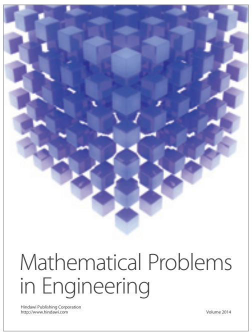

International Journal of
Differential Equations

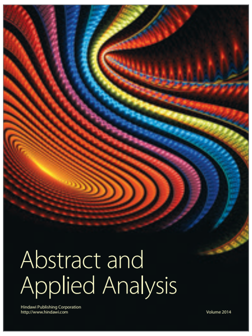
International Journal of
Combinatorics

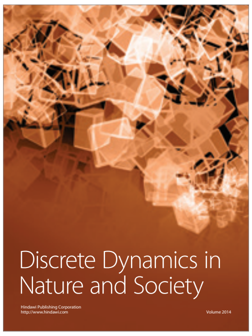

Advances in
Mathematical Physics

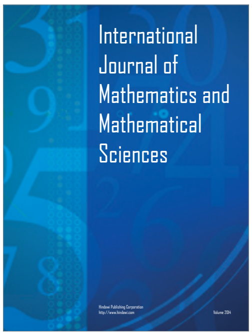

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

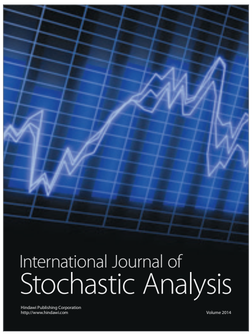

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization