

Compression of NOAA/AVHRR data with a wavelet transform

A. R. S. MARÇAL†, B. TRIEBFÜRST‡, C. SCHNEIDER§ and
R. A. VAUGHAN†

†Department of Applied Physics and Electronic & Mechanical Engineering,
University of Dundee, Dundee, Scotland, UK

‡Institut fuer Informatik, Albert-Ludwigs-Universitaet, Freiburg, Germany

§Institut fuer Physische Geographie, Albert-Ludwigs-Universitaet, Freiburg,
Germany

(Received 12 June 1997; in final form 7 August 1998)

Abstract. An attempt has been made to assess the efficiency of image data compression by wavelet transform encoding using National Oceanic and Atmospheric Administration (NOAA) Advanced Very High Resolution Radiometer (AVHRR) images. Raw and derived images were compressed to various levels and a number of parameters in the decompressed images compared with those obtained using raw data as a yardstick against which to measure the loss of information due to compression. Unsupervised classification, Normalized Difference Vegetation Index (NDVI) values and brightness temperatures appeared to suffer little degradation and only for fractal dimensions was there significant loss of integrity at compression rates of up to a factor of 32. The general conclusion from a visual inspection of the effect of such compressions on artificially generated geometrical imagerettes confirms the effectiveness of this method of compression.

1. Introduction

Compression of remotely sensed data is attracting increased attention as modern instruments (the Synthetic Aperture Radar (SAR) and the new generation of hyper-spectral high-resolution scanners for example) produce increasingly large volumes of data which have to be transmitted back to Earth. Even the analysis of standard multitemporal and multisensor remotely sensed data is often hindered by the limited memory and hard disk capacity of computers because of the enormous amounts of data that have to be handled. Investigations involving large volumes of remotely sensed data in, for example, a Geographic Information System (GIS) therefore depend on further developments of data compression techniques. It is within this framework that this investigation of data compression techniques based on the wavelet transform to compress remotely sensed datasets at high compression ratios whilst still preserving their geo-scientific value was undertaken.

Operational image processing software permits the use of various data compression techniques for raster data, for example, run length encoding which is suitable for homogenous features like classifications and vectorization of nominal scaled raster data. A common factor in the above methods is that the compression causes no loss of information—they are thus called *lossless* methods. However, for many

remote sensing raster datasets, no significant reduction of the amount of data can be achieved by these lossless methods because of the complex content of the images. Many more possibilities for compression are available if a partial loss of information is accepted when so called *lossy* methods can be used. A reduction in radiometric resolution of data, e.g. the reduction of 16-bit data to 8-bit data, or a reduction of the spatial resolution employing a subsampling algorithm, are easy ways to reduce data volumes. The problems are obvious—even small reduction rates can cause great loss of information.

More sophisticated methods include compression methods based on vector quantization (Gersho and Gray 1991), fractal image compression (Fisher 1994) or transform coding techniques, for example the JPEG coder of the Joint Photographic Experts Group that has been an ISO Standard since 1992 (see e.g. Gonzalez and Woods 1992). In the same way that lossless methods may be used, any lossy method can also be used within remote sensing applications. Some of these methods, however, produce block structured artefacts in the compressed imagery (Triebfürst *et al.* 1996). Consequently these methods should not be used unless their effects on the geo-scientific content is fully understood. A central problem in remote sensing technology, therefore, is to test the suitability of the various data compression techniques for different types of applications. In this context we evaluated the suitability of wavelet compression as a special kind of transform coding based on the wavelet transform and the subsequent compression by bitplane coding with the zerotree method (Shapiro 1993) for the storage and classification of Advanced Very High Resolution Radiometer (AVHRR) data from the National Oceanographic and Atmospheric Administration (NOAA) meteorological satellites. Other lossy compression techniques, such as fractal compression and vector quantization, are not considered here. The interested reader may find these methods referred to in works by Fisher (1994), Saupe and Hamzoui (1994) and Gersho and Gray (1991).

The use of wavelet encoding for data compression has recently been compared with a fractal encoding procedure and the JPEG standard by Triebfürst *et al.* (1996) on a SAR image of part of the Antarctic Peninsula. The distortion measure used in this case was the peak signal-to-noise ratio. At all compression rates up to a value of 50, the wavelet encoder returned the best overall results while retaining most relevant features in the image. It was proved that all features that could be mapped from the original image could still be detected at a compression rate of 32, while speckle was removed from the image very efficiently at the same time.

Although the AVHRR does not generate nearly as much data as most of the higher resolution instruments, the facts that it produces 10-bit data and that many images can be collected per day pose similar problems of volume data handling that would benefit from similar solutions. For these reasons we have tried to investigate the usefulness of wavelet compression by operating on both raw and compressed AVHRR data, producing unsupervised classifications and calculating brightness temperatures, Normalized Difference Vegetation Index (NDVI) values and fractal dimensions. The image files were compressed to various levels (up to 64) and then decompressed. The calculated parameters were then compared with those obtained from the reconstructed data on a pixel-by-pixel basis. In order to try to quantify the loss of information, the same procedure of compression was carried out on some artificial geometrical imageries that could be inspected visually for signs of degradation.

2. Theory of transform coding

In transform coding a reversible linear transform, such as the Fourier transform, is used to transform images in such a way that a significant number of transform coefficients have a small magnitude. The coefficients are quantized and coded, and the small coefficients are discarded or quantized very coarsely without a significant loss of image information.

Mathematically, in order to transform an image by a linear transform requires the image to be regarded as a vector with respect to a new basis. The image can be transformed blockwise as with the JPEG method, line-by-line as with the wavelet transform, or as a whole. The JPEG method mentioned above is based on transform coding by the discrete cosine transform (DCT). Here, the basis vectors are generated by discrete samples of different cosine functions (figure 1(a)). Although here the method used for image information packing is transform coding based on wavelet functions, we will nevertheless, for illustration purposes, first consider a simple example based on the DCT. For the wavelet transform the idea is similar but uses other transform functions. Let us consider an image line

$$\mathbf{b}_e = (1.9, 1.5, 2.1, -1.5) \quad (1)$$

which is a vector in the four-dimensional vector space \mathbf{R}^4 represented with respect to the standard basis $\{\mathbf{e}_1 = (1,0,0,0); \mathbf{e}_2 = (0, 1, 0, 0); \dots\}$ with

$$\mathbf{b}_e = 1.9\mathbf{e}_1 + 1.5\mathbf{e}_2 + 2.1\mathbf{e}_3 - 1.5\mathbf{e}_4 \quad (2)$$

The four-dimensional basis of the discrete cosine function derived from the continuous cosine function is $\{\mathbf{c}_1 = 1/2(1, 1, 1, 1); \mathbf{c}_2 = 1/2(1, 0.71, 0, -0.71); \mathbf{c}_3 = (1, 0, -1, 0); \mathbf{c}_4 = (1, -0.71, 0, 0.71)\}$. Projection from the standard basis to the new basis yields

$$\mathbf{b}_c = +4.0\mathbf{c}_1 + 4.02\mathbf{c}_2 - 0.2\mathbf{c}_3 - 0.22\mathbf{c}_4 \quad (3)$$

by calculating the inner products of \mathbf{b}_e with the four basis vectors of the DCT. The coefficients obtained may be quantized and we obtain (4, 4, 0, 0). Once we preserve only information on the non-zero coefficients we reduce the amount of storage needed from four floating point numbers respectively 16 bytes to 2 bytes. Performing the inverse transform but using the remaining two integers only yields an image line with

$$\mathbf{b}_e = (2, 1.4, 2, -1.4) \quad (4)$$

which is similar to the original but not exactly the same.

The fundamental idea of the discrete wavelet transform (DWT) uses a special kind of basis functions rather than cosine functions which have provided an excellent tool in many areas of signal processing. In the 1980s Grossman and Morlet (1984) defined the first wavelets in the field of quantum physics, but the underlying theory had been discovered independently by several groups in the 1930s while looking for new scale-varying basis functions to decompose other functions. An historical overview is given by Meyer (1993).

Wavelet analysis can be considered a logical step from windowed Fourier analysis using variable-sized regions rather than ones of fixed size. Larger sized windows are used in regions where there is low-frequency information and smaller ones where there is high-frequency information. This enables local analysis of large signals (or images) to be performed. Hence wavelet analysis is in many ways more suitable than Fourier analysis in the study of complex spatial images that contain a wide range of frequency components. Figure 1(b) shows the first and simplest wavelet basis, the

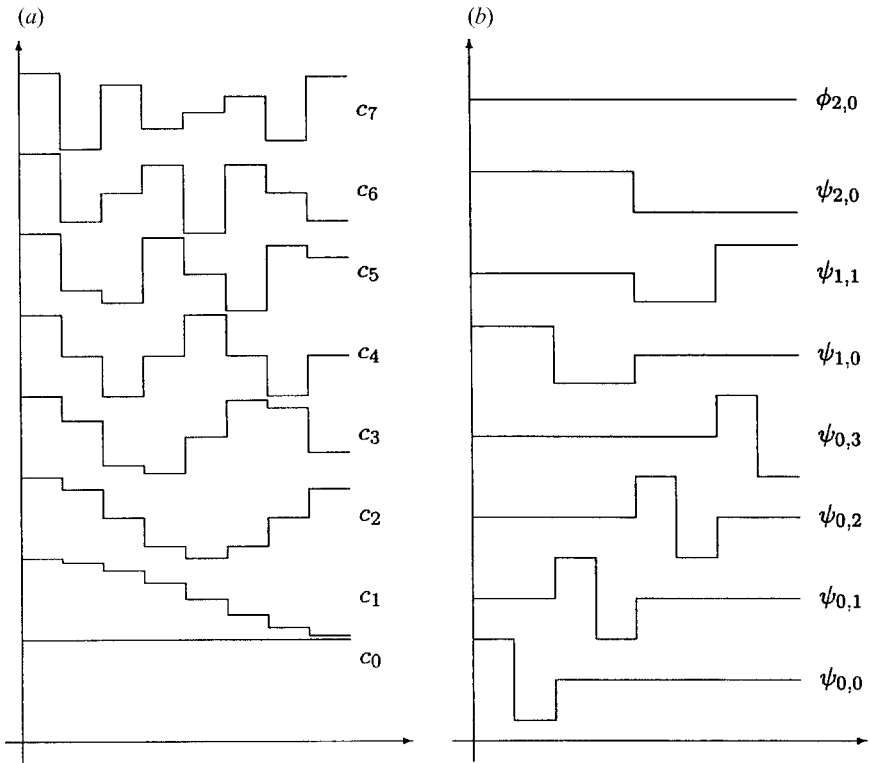


Figure 1. (a) Basis functions of the discrete cosine transform (DCT) and the discrete wavelet transform (DWT) (b) as given by Haar (1910). Many other families of wavelets may be used for image transformation. Eight basis functions are presented in this figure. The number of basis functions in the spatial domain, which corresponds to the length of the original image line, is given by the number of pixels in the line. Since the number of basis functions in each co-ordinate system must be equal when transforming from one system to another the functions presented here correspond to an image line of 8 pixels.

so-called Haar-basis (Haar 1910). An example, analogous to the DCT example above, would be the four-dimensional basis $\{\mathbf{h}_1 = 1/2(1, 1, 1, 1); \mathbf{h}_2 = 1/2(1, 1, -1, -1); \mathbf{h}_3 = (0.71, -0.71, 0, 0); \mathbf{h}_4 = (0, 0, 0.71, -0.71)\}$ based on the Haar wavelet, also derived by transforming the continuous Haar function into discrete function values. Projection from the standard basis to the Haar-basis by calculating the inner products yields

$$\mathbf{b}_{\text{Haar}} = 2.00\mathbf{h}_1 + 1.40\mathbf{h}_2 + 2.40\mathbf{h}_3 + 0.42\mathbf{h}_4 \quad (5)$$

An orthogonal wavelet basis \mathbf{B} is built up of functions $\psi_{m,n}$ generated by dilations and translations of a special ‘mother wavelet function’ $\psi = \psi_{0,0}$

$$\mathbf{B} = \{\psi_{m,n} | \psi_{m,n}(x) = 2^{-m/2} \psi(2^{-m}x - n), m, n \in \mathbb{Z}\} \quad (6)$$

By dilations (parameter m) we can consider the same function at different scales (see $\psi_{0,0}, \psi_{1,0}, \psi_{2,0}$, in figure 1(b)). This leads to scale analysis of a signal similar to the frequency analysis by cosine functions. By translation (parameter n) of the mother wavelet we can analyse a signal in one scale at different locations (see $\psi_{0,0}, \psi_{0,1}$,

$\psi_{0,2}$, in figure 1(b)). All combinations of the parameters m and n result in decomposition of a signal (image) at different scales (related to different frequencies) depending on the location in the image. Hence this method is called *time-scale analysis*. This principle is illustrated in figure 2. Each combination of scale and position of the wavelet function provides one single basis function in the new wavelet basis. In the example above, by dilation of the Haar-basis vector $\mathbf{h}_2 = 1/2(1, 1, -1, -1)$ we get, according to equation (6), the basis vector $\mathbf{h}_3 = (0.71, -0.71, 0, 0)$. By translation of the basis vector \mathbf{h}_3 we get the basis vector $\mathbf{h}_4 = (0, 0, 0.71, -0.71)$, also according to equation (6). The total number of basis functions is defined by the size of the image line to be transformed, because the dimension of the vector relates to the length of the image and therefore the number of basis functions must be the same in both co-ordinate systems.

The two-dimensional wavelet transform based on such basis functions can be implemented as an iterative process of low pass and high pass filtering of the image followed by a subsampling of the image by a factor of two. In a first step, each line of the image is filtered with the low pass kernel \mathbf{H} and the high pass kernel \mathbf{G} . In the second pass the resulting image is filtered again with these kernels, but this time in each column. After subsampling an image composed of four parts is obtained. Each part represents the image at lower resolution but in a specific combination of

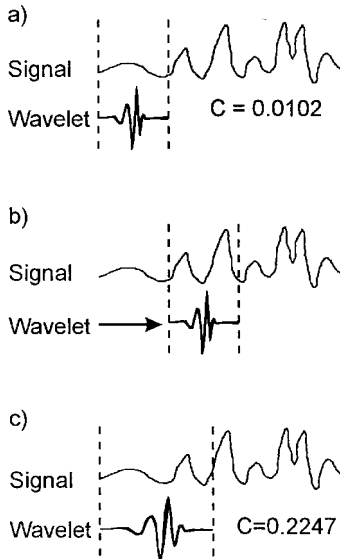


Figure 2. Scaling and positioning of a wavelet along a one-dimensional signal in order to transform this signal into the wavelet domain. The example shows a continuous wavelet transform rather than a discrete wavelet transform. The concept can be regarded as a five-step recipe. In (a) a selected wavelet (step 1) is compared to the first section of the signal and a coefficient is calculated (step 2) that is larger the more similarity that is found between this part of the signal and the wavelet itself. In (b) the wavelet is shifted along the signal until the whole signal is covered (step 3). (c) shows the stretched wavelet (step 4) to which steps 2 to 3 have to be applied. Steps 1 to 4 must be repeated for all scales (step 5) until the whole signal is covered at all locations with all scales. The DWT works in the same way but with discrete wavelets, making it more suitable for rapid computation (adopted from Misiti *et al.* (1996)).

high or low pass filtering in columns or lines (figure 3). Consider the example of the single line used earlier. Instead of the whole wavelet basis we need use only a small filter pair of a low pass and a high pass filter. For example the basis ($\mathbf{h}_1, \dots, \mathbf{h}_4$) can be represented by the Haar-filter pair $\mathbf{G} = (0.71, 0.71)$ (low pass filter) and $\mathbf{H} = (0.71, -0.71)$ (high pass filter). In a first step the signal is filtered with both filter kernels and the output signals are downsampled by a factor of two.

$$\mathbf{b}_G := (\mathbf{G} * \mathbf{b}_e) \downarrow 2 = (2.40, 0.42) \tag{7}$$

$$\mathbf{b}_H := (\mathbf{H} * \mathbf{b}_e) \downarrow 2 = (0.28, 2.54) \tag{8}$$

where $*$ represents the convolution operator and \downarrow the downsampling operator.

The high pass filtered signal (\mathbf{b}_H) represents the highest frequency parts of the signal. \mathbf{b}_H is the same as the inner product of \mathbf{b}_e with \mathbf{h}_3 and of \mathbf{b}_e with \mathbf{h}_4 . In a second step the low pass filtered signal (\mathbf{b}_G) is filtered again with \mathbf{G} and \mathbf{H} . Now we obtain a high-frequency part (\mathbf{b}_{HG}) (but not as high as in the first filtering step) and a low-frequency part (\mathbf{b}_{GG}):

$$\mathbf{b}_{HG} := (\mathbf{H} * \mathbf{b}_G) \downarrow 2 = (1.4) \tag{9}$$

$$\mathbf{b}_{GG} := (\mathbf{G} * \mathbf{b}_G) \downarrow 2 = (2.0) \tag{10}$$

\mathbf{b}_{HG} corresponds to the inner product of \mathbf{b}_e with \mathbf{h}_2 and \mathbf{b}_{GG} represents the inner product of \mathbf{b}_e with \mathbf{h}_1 . This iterative filtering and downsampling process is much

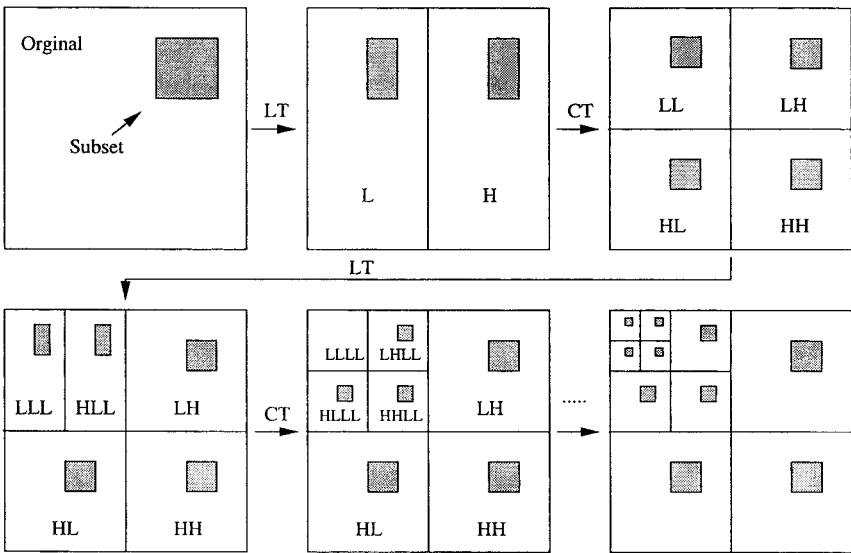


Figure 3. The computational scheme of the DWT implemented with the wavelets derived by Daubechies (1992) as subsequential filtering with high and low pass filters separately in image lines and image rows. Filtering is followed by a subsampling of two, resulting in an image of the same size in the wavelet domain with four parts each composed of a combination of the result from high or low pass filtering in vertical or horizontal directions. Only the transform is illustrated here. Further steps (described in the text) are necessary to encode and compress the resulting wavelet coefficients.

easier to calculate than the inner products of the original signal with the Haar-basis vectors. Depending on the signal length the filtering process can be repeated again and again. This is an added advantage for the popularity of the DWT because the computational costs are only related to N for a signal of length N . This is much less than the computational cost of the Fast DCT, which is related to $(N \log N)$. Calculating all of the necessary inner products would result in computational costs of N^2 .

A detailed description of this wavelet transform is given by Daubechies (1992) and Strang and Nguyen (1996). To compress the transformed image we have to consider the magnitude of the coefficients of the transformed image. As with principal component analysis (PCA) the magnitude of most of the coefficients is very small. Only a few of the coefficients—those of high magnitude—contain most of the image information. These coefficients are concentrated in the lowest frequency scales. To compress the transformed image the coefficients of small magnitude are eliminated and the others must be quantized.

A common method to quantize the coefficients is to divide the transformed image into several bitplanes. Each bitplane may be separately coded by an entropy-coder (Gonzalez and Woods 1992). Shapiro (1993) improved this method by using the redundancy between single-frequency bands with the zerotree coding method.

There are two options for the use of wavelet-based transformations. The system may be used solely for data compression and the results from this lossy method can be tested against the original data in terms of classification, segmentation or visual interpretation as performed for NOAA AVHRR data in this paper. Furthermore, the wavelet encoding procedure may offer new methods of image analysis itself. By transforming an image into the wavelet domain, data can be accessed in different frequency scales. The use of filtering of remotely sensed data in the frequency domain, for example to generalize classified data as land use maps, is extended by the possibility of applying the filter at specific locations only. Classifications obtained from optical remote sensing data, for example, can be generalized locally, depending on the surface type or any other statistical, textural or structural parameter. Triebfürst *et al.* (1997) showed that, with this method, forest classes as obtained from Landsat Thematic Mapper (TM) data could be generalized, while high-frequency features in neighbouring urban areas were still preserved.

3. Data compression test

3.1. Test imagerettes

In an attempt to try to quantify the level at which the loss of information produced by compression occurs, a series of geometrical shapes and patterns were generated (figure 4). The data were stored in 8-bit format, although the top imagerette only uses two levels of grey. These 64×64 pixel imagerettes were reconstructed after performing the compression process using the wavelet method with rates of 4, 8, 16, 32 and 64. The figure shows that, up to compression by a factor of 8, the reconstructed images are virtually identical to the originals, with perhaps just a small loss of clarity in the diagonal lines. At compression level 16, a certain amount of fuzziness and shadowing becomes apparent, but even the high-frequency spacings are still quite evident. The main features are still reasonably clear at level 32, but the finer detail has gone by level 64. The shadowing effect is a manifestation of the pseudo-Gibbs

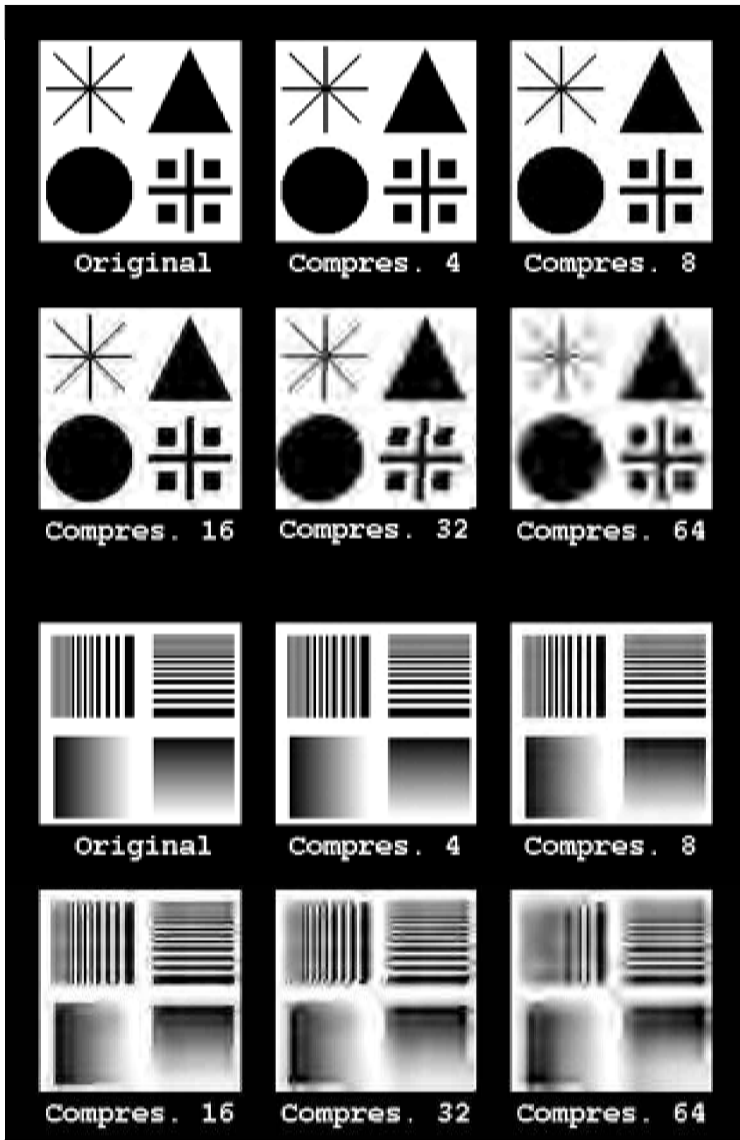


Figure 4. Geometrical imagettes, each 64 pixels by 64 pixels, used to test the effect of different compression levels.

phenomenon showing the inability of wavelets to represent step functions at high compression rates. This is an oscillatory effect which appears near edges due to the coarse quantization of the wavelet coefficients.

3.2. AVHRR data test

A NOAA-14 AVHRR image, taken 14 September 1995, was selected for processing. The raw image was rectified into a 512×512 pixel geo-referenced dataset, with 1 km pixel size. All five AVHRR channels were used, as well as NDVI and

Brightness Temperature (BT) images from channels 3, 4 and 5. The five AVHRR channels and the NDVI were stored in 16-bit format files and the three BT images in 8-bit format. The compression/decompression process using the wavelet method with compression rates of 4, 8, 16, 32 and 64 was performed on these nine files. A total of 54 files was produced. The analysis of the effects of compression was carried in three distinct parts: NDVI and BT images; unsupervised classification; and fractal dimension.

3.2.1. *NDVI and Brightness Temperature*

The effects of compression on NDVI and BT images can be seen in figures 5 and 6 respectively. In both figures, the top left-hand image is the original and the other three are reconstructed images after compression to rates 4 (top right), 16 (bottom left) and 64 (bottom right). There are virtually no differences between the original and the reconstructed images at compression rate 4. For the higher compression rates the differences start to become noticeable, in particular for rate 64.

The differences between the original and reconstructed images were computed on a pixel-by-pixel basis. The error in each reconstructed image in relation to the original image was calculated for each pixel as a percentage value. To avoid an infinite percentage error when the original values were near zero, only NDVI values of at least 0.020 (60 463 pixels) and BT values of at least 2°C were used (196 824 pixels for BT4, 191 408 pixels for BT5).

Figure 7 shows a plot of the mean error of NDVI, BT4 and BT5 as a function of compression rate. The mean error for BT is low for all levels of compression, reaching only 6.8% for a compression level of 64. The mean error of the NDVI is considerably higher, reaching 14.9% for a compression rate of 32 and 23.3% for compression rate of 64.

Image data produced by combining two or more AVHRR channels, such as NDVI and (BT4–BT5) (which is often used for atmospheric corrections), were found to be less affected when calculated before compression than after compression. It is therefore advisable to produce additional images with the required processing, NDVI and (BT4–BT5) for example, before performing data compression.

Another way of examining the percentage errors is presented in figure 8—the fraction of pixels on the NDVI and BT4 images having errors of $\leq 1\%$, $\leq 5\%$ and $\leq 10\%$ are displayed as a function of compression rate. 81.5% of pixels in the BT4 image have an error $\leq 10\%$ after compression at rate 64. For the same compression rate, only 42.3% of the NDVI pixels have an error of less than 10% when compared with the original image.

3.2.2. *Unsupervised classification*

An isocluster unsupervised classification was performed, using AVHRR channels 1 and 2, BT3, BT4 and BT5, and NDVI images. The classification was run, with the same statistical parameters, on the original images and on those that underwent each of the five levels of compression.

Between 20 and 50 classes were allowed, but in all six classification processes the number of classes was found to be 40. This number was then reduced to 12 by inspection of each individual class. A brief description of the 12 main classes, as well as the number of pixels in each class on the original data, is presented in table 1. Classes E, F and G were later merged into class E' and classes H, I, J and K into class H'. The percentage of pixels from each original class that were classified in the

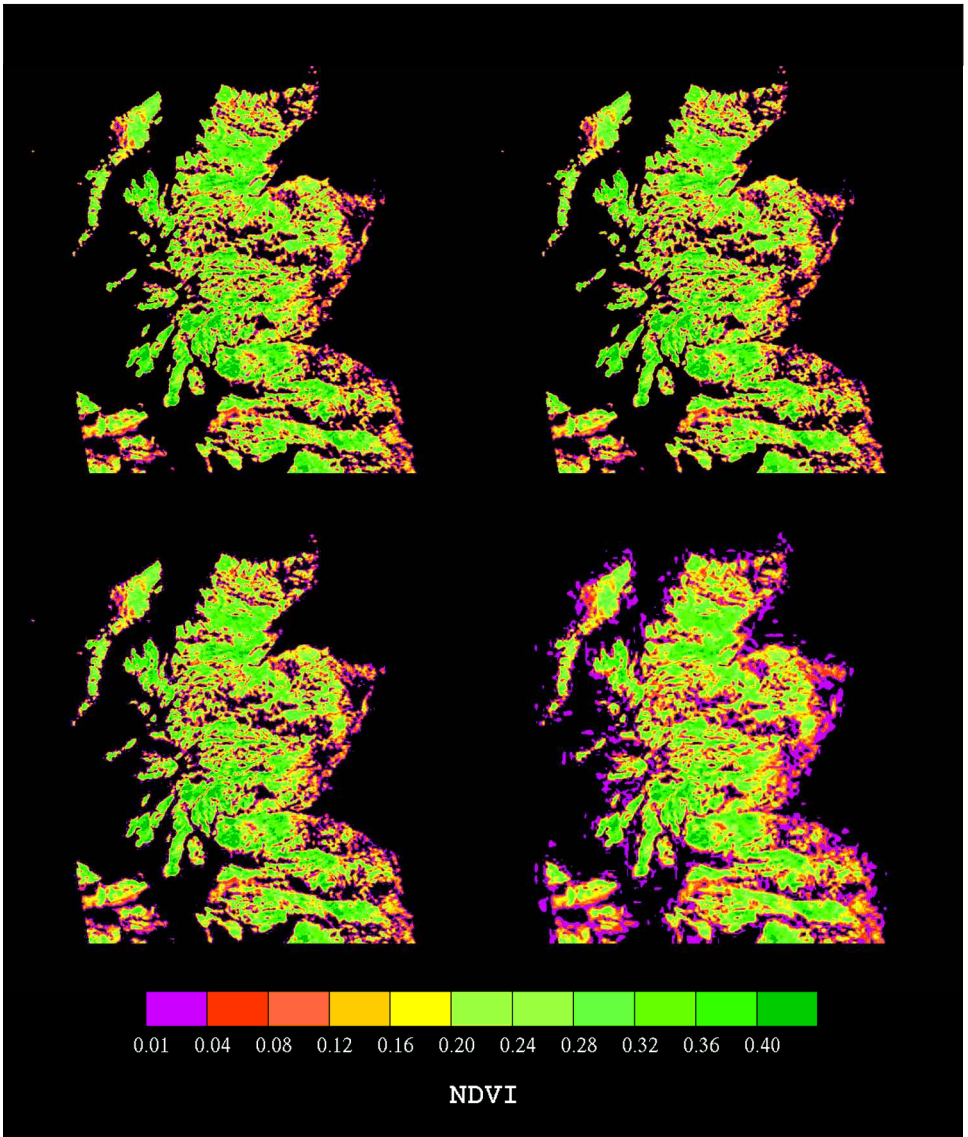


Figure 5. Normalized Difference Vegetation Index (NDVI) imagery derived from AVHRR datasets. Original (top left) and compression rates of 4:1 (top right), 16:1 (bottom left) and 64:1 (bottom right).

same class using the compressed data is also presented in table 1. From the 63 938 pixels classified as water (class C) using the original images, 99.89% of those pixels were still classified as water using images with a compression of 8, but only 82.86% were classified as water when using images with a compression of 64.

Figure 9 shows the classification results using as inputs the original images (top left), and the reconstructed images after a process of compression to rates 4 (top right), 16 (bottom left) and 64 (bottom right). There are subtle variations at the boundaries between classes, with their distinction first lost at around level 32. This is consistent with results from the imagette compression test.

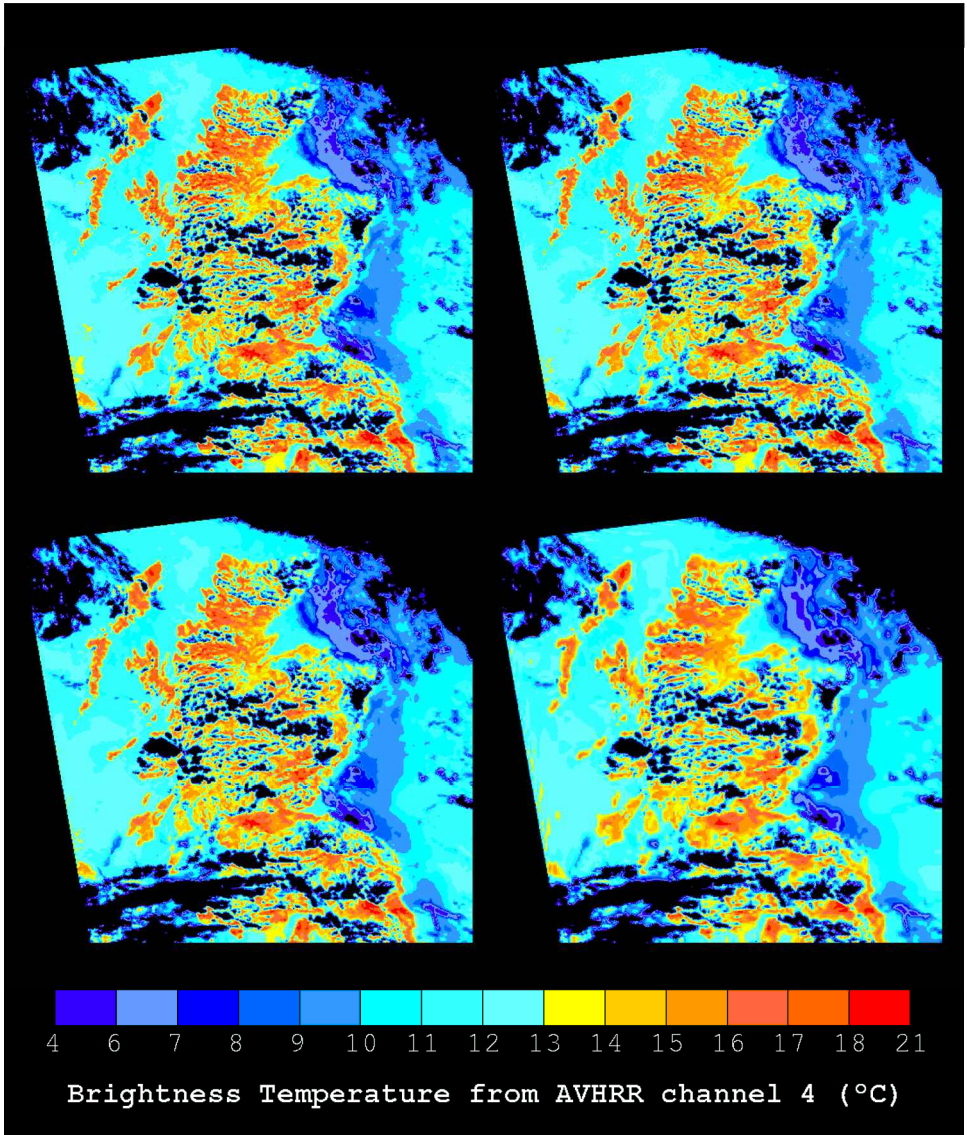


Figure 6. Brightness temperature (BT) calculated from AVHRR channel 4. Original (top left) and compression rates of 4:1 (top right), 16:1 (bottom left) and 64:1 (bottom right).

3.2.3. Fractal dimension

Unlike the previous parameters tested, the fractal dimension of a region of an image reflects the spatial relationship between pixels, and gives an indication of the texture in that region. A dimension of three would imply a perfectly three-dimensional structure whereas a dimension of two implies a flat plane. Typically, the fractal dimensions of satellite images are somewhere between two and three. Three homogeneous sub-sections of the image (each 64 pixels by 64 lines) were selected to test the effect of compression on fractal dimension. The sub-sections covering land, sea and cloud were processed independently for the relevant image channels. The effect

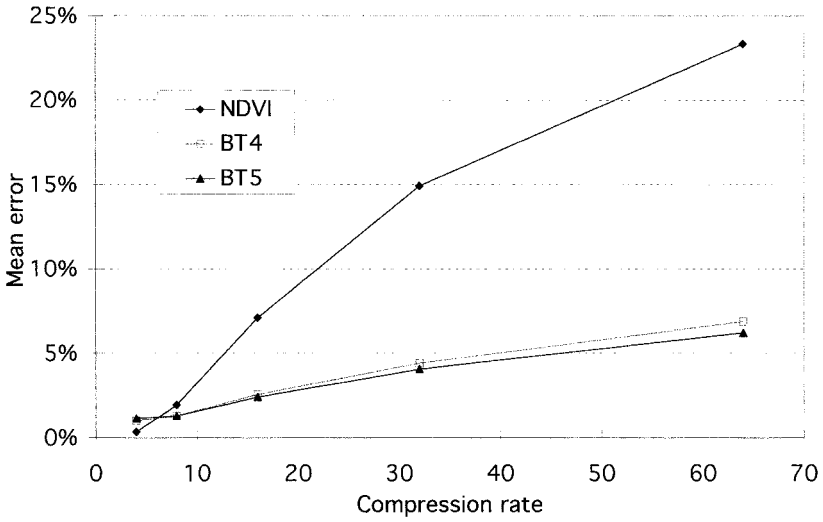


Figure 7. Mean errors on NDVI and BTs from channels 4 and 5, as a function of compression rate.

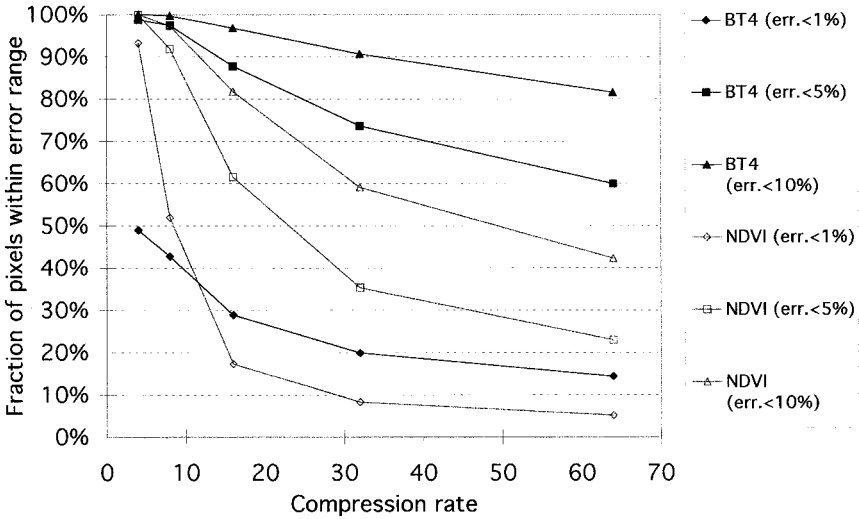


Figure 8. Number of pixels which differ from those in the original NDVI and BT images by up to 1%, 5% and 10%, as a function of compression rate.

of compression on the fractal dimension of land can be seen in figure 10 for AVHRR channels 1 and 2, BT from channel 4 and NDVI. The fractal dimension decreases steadily with increasing compression rate, suggesting that some high-frequency textural information is lost in the compression process. Fractal dimension variation as a function of compression rate for cloud and sea can be seen in figures 11 and 12 respectively, for AVHRR channels 1 and 2, and BT from channel 4 (for sea only). The anomaly at compression rate 4 for all parameters over the sea is due to their

Table 1. Percentage of pixels from each original class which were subsequently classified in the same class after compression at various levels.

Description	Class	No. pixels (Original image)	Fraction of pixels classified as in the original image (%)				
			4:1	8:1	16:1	32:1	64:1
Area not covered by satellite pass	A	27493	100.00	100.00	100.00	99.98	99.81
Cloud I	B	20784	99.87	97.97	98.61	96.95	92.92
Water	C	63938	99.71	99.89	84.85	87.94	82.86
Land	D	34570	99.43	94.36	91.71	89.04	86.82
Coastlines, lakes, partially cloudy areas	E	23072	97.71	75.91	94.43	84.00	63.49
Partially cloudy pixels on land	F	13941	97.30	84.87	87.57	72.23	59.85
Low % of cloud cover on land	G	13671	98.71	93.01	87.73	70.57	70.84
High % of cloud cover	H	20620	98.90	72.12	83.27	73.61	73.61
Partially cloudy pixels on sea	I	20773	99.20	82.92	91.16	86.43	79.29
Very high % of cloud cover I	J	11228	98.74	98.24	87.28	74.92	58.86
Very high % of cloud cover II	K	2479	96.77	87.45	80.96	65.91	45.58
Cloud II	L	9575	98.65	95.02	90.76	85.16	77.41
	(EUFUG)		98.76	89.13	96.27	90.04	81.82
	(HUIUJUK)		99.45	87.85	91.98	87.25	84.49

very small variability in such a region, and also possibly due to loss of spatial coherence in random noise.

4. Conclusion

It has been shown that, for the parameters tested using AVHRR data, the use of wavelet compression techniques provides a method for data reduction without the loss of too much geophysical information. The classification test results indicate that there is greater loss in the boundaries between classes and in mixed pixel areas. Simple pixel-related parameters such as brightness temperatures retained their integrity quite well up to a rate of compression of 64, whereas parameters that depend on more than one band (NDVI and classification), as well as those related to spatial variability (fractal dimension), lost information at lower compression rates. Therefore standard products such as NDVI and (BT4–BT5) used for atmospheric corrections should preferably be calculated from original data with subsequent compression of the result. This yields better results than calculation of these secondary products from compressed data. For archiving large amounts of data, these products could be added to the original bands of the imagery and then compressed. For the NOAA AVHRR image of Scotland with two additional bands and a compression of 32 a reduction of storage volume of a factor by approximately 30 could still be obtained.

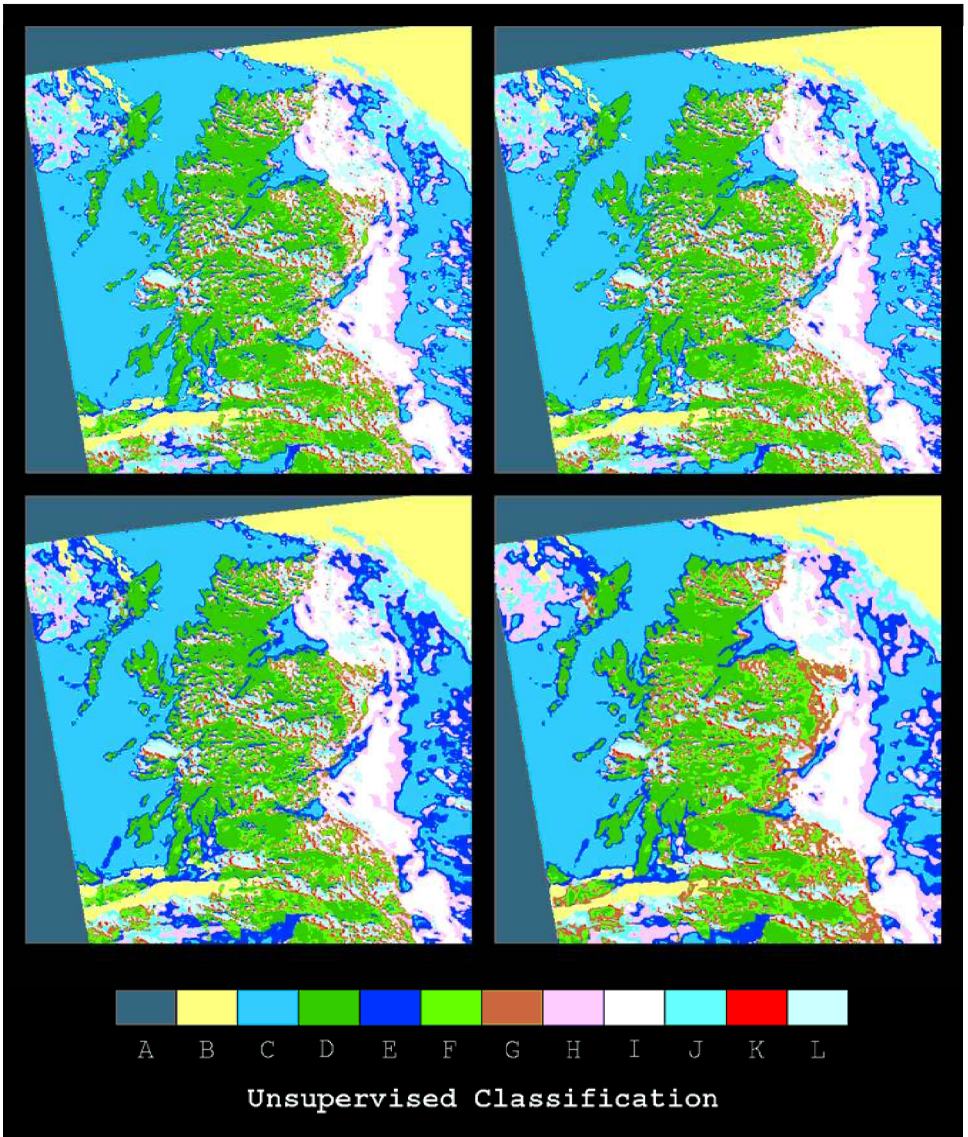


Figure 9. Unsupervised classification using the five AVHRR channels and NDVI. Results using the original images (top left) and compression rates of 4:1 (top right), 16:1 (bottom left) and 64:1 (bottom right) (see table 1 for class description).

With this substantial reduction in storage volume achievable, data compression based on wavelet transforms could prove to be a valuable tool in the handling of remotely sensed data within computational networks and archiving facilities.

Acknowledgments

This work was carried out under the CEC's Human Capital and Mobility programme, contract number CHRX-CT93-0310, 'Synergy of Remotely Sensed Data'. The authors would also like to thank Tomasso Parinello for the use of his fractal

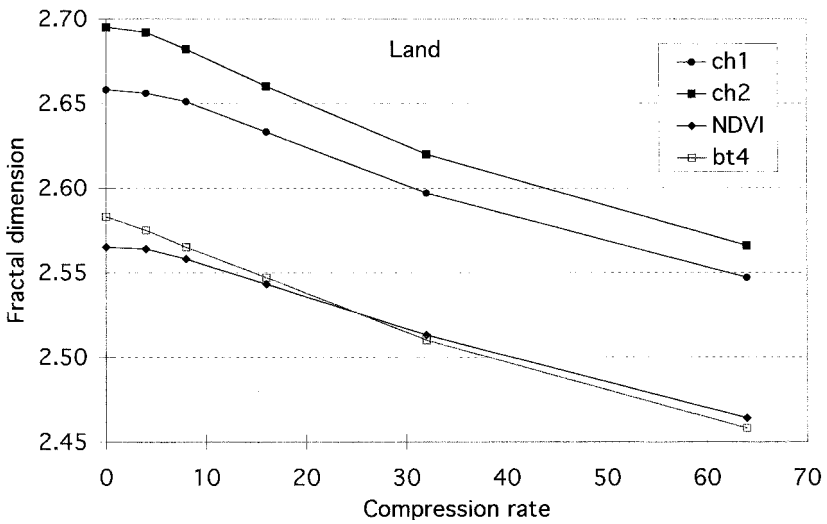


Figure 10. Effect of compression rate on the fractal dimension of an image sub-section covering exclusively land.

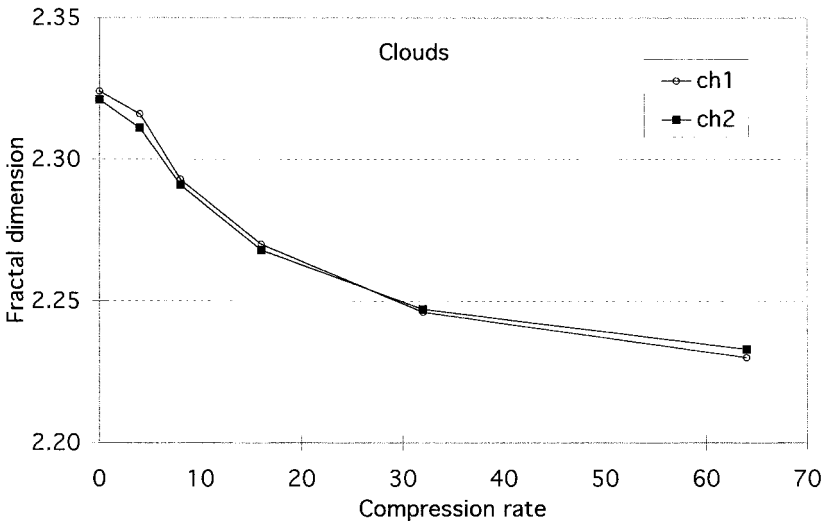


Figure 11. Effect of compression rate on the fractal dimension of an image sub-section covering exclusively cloud.

analysis code prior to publication and for helpful discussions on the interpretation of the results. The authors are grateful for funding from the 'Deutsche Forschungsgemeinschaft', which is supporting the project 'FREIburger KOMpression Modul' (FREIKOM) at the University of Freiburg. ARSM would like to thank his sponsors, the Portuguese Ministry of Science and Technology (PRAXIS XXI program).

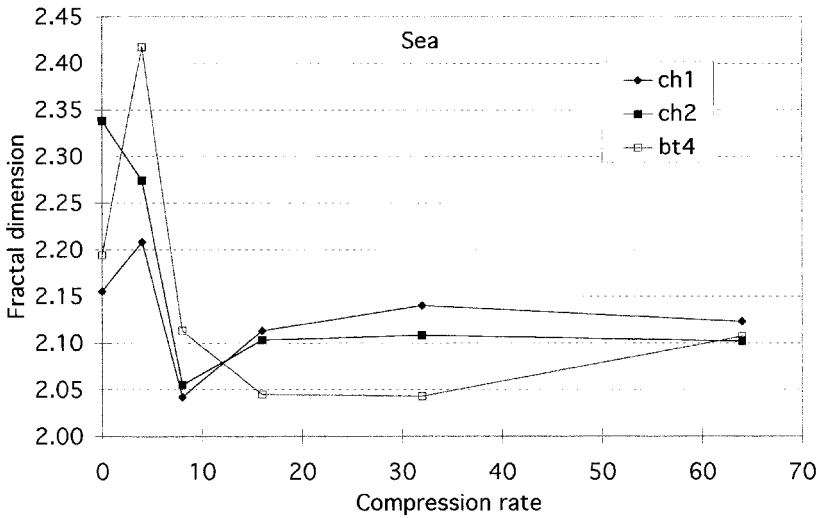


Figure 12. Effect of compression rate on the fractal dimension of an image sub-section covering exclusively sea.

References

- DAUBECHIES, I., 1992, *Ten Lectures on Wavelets* (Philadelphia, Pennsylvania: Society of Industrial and Applied Mathematics Publications).
- FISHER, Y., 1994, Fractal image compression with quadtrees. In *Fractal Image Compression—Theory and Applications to Digital Images*, edited by Y. Fisher (New York: Springer-Verlag), pp. 55–78.
- GERSHO, A. R., and GRAY, M., 1991, *Vector Quantization & Signal Compression* (Boston: Kluwer).
- GONZALEZ, R. C., and WOODS, R. E., 1992, *Digital Image Processing* (Reading, MA: Addison-Wesley), p. 716.
- GROSSMAN, A., and MORLET, J., 1984, Decomposition of hardy functions into square integrable wavelets of constant shape. *Society of Industrial and Applied Mathematics Journal of Mathematics*, **15**, 723–736.
- HAAR, A., 1910, Zur Theorie der orthogonalen Funktionen-Systeme. *Math. Ann.*, **96**, 331–371.
- MALLAT, S. G., 1989, A theory for multiresolution signal decomposition: the wavelet decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, 674–693.
- MEYER, Y., 1993, *Wavelets, Algorithms and Applications*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, Pennsylvania, USA.
- MISITI, M., MISITI, Y., OPPENHEIM, G., and POGGI, J.-M., 1996, Wavelet toolbox user's guide. In *Manual to the MATLAB Software Package* (Natick, MA: Math Works Inc.).
- SAUPE, D., and HAMZAOU, R., 1994, A guided tour of the fractal image compression literature. Technical Report 58, Institut fur Informatik, Universitaet Freiburg, <http://www.informatik.uni-freiburg.de/TechReports>
- SHAPIRO, J. M., 1993, Embedded image coding using zerotree of wavelet coefficients. *IEEE Transactions on Signal Processing*, **41**, 3445–3462.
- STRANG, G., and NGUYEN, T., 1996, *Wavelets and Filter Banks*. (Welsley: Cambridge Press), p. 490.
- TRIEBFÜRST, B., SAUPE, D., and SAURER, H., 1997, Compression and generalisation of remote sensing data in a GIS with an optimized wavelet-coder. *Proc. AGIT 1997*, Salzburger Geographische Materialien 26, University of Salzburg.
- TRIEBFÜRST, B., SCHNEIDER, C., WUNDERLE, S., SAURER, H., and GOSSMANN, H., 1996, Compression of Spaceborne SAR Imagery. *ITC Journal*, **3**, 225–229.