Applied Mathematics and Computation 231 (2014) 435–444

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A simple class of fractal transforms for hyperspectral images

E.R. Vrscay [a,*], D. Otero [a], D. La Torre [b,c]

[a] Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada
[b] Department of Applied Mathematics and Sciences, Khalifa University, Abu Dhabi, United Arab Emirates
[c] Department of Economics, Management and Quantitative Methods, University of Milan, Milan, Italy

ARTICLE INFO

Keywords:
Hyperspectral images
Function-valued image mappings
Image self-similarity
Fractal transforms

ABSTRACT

A complete metric space of function-valued mappings appropriate for the representation of hyperspectral images is introduced. A class of fractal transforms is then formulated on this space. Under certain conditions, the fractal transform T can be contractive, implying the existence of a unique fixed point. We then formulate a simple class of block transforms for the fractal coding of digital hyperspectral images, and illustrate with an example.

1. Introduction

In [22], we examined some basic self-similarity properties of hyperspectral (HS) images, considering them as *function-valued mappings* of a base (or pixel) space X to a suitable (spectral) function space. At each location or pixel $x \in X$, the hyperspectral image mapping $u(x)$ is a *function* that is supported on a suitable domain of definition Y. In practical applications, of course, HS images are digitized: Both the base space X and spectral domain Y are discretized so that $u(x)$ is a vector.

Earlier studies of greyscale images [1,5] have shown that most subblocks of natural images are well approximated (using various forms of affine greyscale mappings) by a number of other subblocks of the image. Such *image self-similarity* is responsible, at least in part, for the effectiveness of various non-local image processing schemes, including nonlocal-means denoising [6], fractal image coding [12,18] and a variety of other methods devoted to image enhancement, e.g., [8–11,14]. The study in [22] shows that HS images are also quite self-similar, in the sense that "data cubes", namely, M-channel vectors supported over $n \times n$-pixel subblocks of the HS image are well approximated by a number of other data cubes of the image. Moreover, the spectral functions over individual pixels demonstrate a remarkable degree of correlation with each other, not only locally but over the entire image. This suggests that various nonlocal image processing schemes which rely on self-similarity should be quite effective for HS images.

In Section 3 of this paper, we provide the mathematical formalism for a particular class of *affine fractal transforms* on the space of function-valued HS images and show that under certain conditions, a fractal transform T can be contractive. From Banach's Fixed Point Theorem, this implies the existence of a fixed point HS image $\bar{u}$ such that $T\bar{u} = \bar{u}$. This leads to the *inverse problem* of fractal image coding, namely, given an HS image u, find a fractal transform T with fixed point $\bar{u}$ that approximates u to a sufficient degree of accuracy. As in the case of fractal coding of greyscale images, this problem can be solved by means of *collage coding*, i.e., find a fractal transform T that maps the HS image u as close to itself as possible.

One of the original motivations for fractal image coding was *image compression* [4,12,18]. As in the case of standard transform coding of images, it was found that much less computer memory was required to store the parameters defining the block-based fractal transform T of an image u. Moreover, the fixed-point approximation $\bar{u}$ to u can be constructed by iteration

* Corresponding author.
E-mail addresses: ervrscay@uwaterloo.ca (E.R. Vrscay), dotero@uwaterloo.ca (D. Otero), davide.latorre@kustar.ac.ae, davide.latorre@unimi.it (D. La Torre).

0096-3003/$ - see front matter
http://dx.doi.org/10.1016/j.amc.2014.01.007

of the transform T. Fractal image coding has been shown to be effective in performing a number of other image processing tasks, for example, denoising [15] and super-resolution [19].

In Section 5, we examine in more detail a block fractal coding scheme briefly introduced in [22], deriving sufficient conditions for contractivity of the associated fractal transform T. We also present the results of some computations on a hyperspectral image. However, it is not the purpose of the present paper to investigate the compression capabilities of this fractal coding scheme nor to compare it with other compression schemes.

Acknowledging the tremendous amount of work that has been done on hyperspectral images, e.g., [7,21], we mention that our work is intended to complement the well-established notion that hyperspectral images generally exhibit a high degree of correlation which can be exploited for the purposes of image enhancement.

2. A complete metric space (Y, d_Y) of hyperspectral images

We consider hyperspectral images as function-valued mappings of a *base space* X to an appropriate space of *spectral functions* $\mathcal{F}$, along the lines established in [22,20]. In this paper, the ingredients of our formalism are as follows:

- **The base space X:** The compact support of the hyperspectral images, with metric d_X. For convenience, $X = [0,1]^n$, where $n = 1, 2$ or 3.
- **The range or spectral space $\mathcal{F}$:** The space $L^2(\mathbf{R}_s)$ of square-integrable functions supported on a compact set $\mathbf{R}_s \subset \mathbf{R}_+$, where $\mathbf{R}_+ = \{y \in \mathbf{R} | y \geqslant 0\}$. $L^2(\mathbf{R}_s)$ is a Hilbert space with the standard definition of the inner product, i.e.,

$$\langle f, g \rangle = \int_{\mathbf{R}_s} f(t) g(t)\, dt, \quad \forall f, g \in L^2(\mathbf{R}_s). \tag{1}$$

This inner product defines a norm on $L^2(\mathbf{R}_s)$, to be denoted as $\| \cdot \|_{L^2(\mathbf{R}_s)}$.

We now let Y denote the set of all function-valued mappings from X to $L^2(\mathbf{R}_s)$. Given a hyperspectral image $u \in Y$, its value $u(x)$ at a particular location $x \in X$ will be a function – more precisely, an element of the space $L^2(\mathbf{R}_s)$. Following the same prescription as in [20], the norm $\| \cdot \|_{L^2(\mathbf{R}_s)}$ arising from Eq. (1) may be used to define a norm $\| \cdot \|_Y$ on Y which, in turn, defines a metric d_Y on Y. The distance between two hyperspectral images $u, v \in Y$ will then be defined as

$$d_Y(u, v) = \left[\int_X \|u(x) - v(x)\|^2_{L^2(\mathbf{R}_s)}\, dx \right]^{1/2}. \tag{2}$$

It is straightforward to show that the metric space (Y, d_Y) of hyperspectral images is complete. In fact, Y is trivially a Hilbert space.

3. A class of fractal transforms on (Y, d_Y)

We now list the ingredients for a class of fractal transforms on the space of HS images introduced above. For simplicity (especially as far as notation is concerned), we assume that our HS images are "one-dimensional," i.e., $X = [0,1]$. The extension to $[0,1]^n$, in particular, $n = 2$, is straightforward.

1. A set of N one-to-one, affine contraction mappings $w_i : X \to X$, $w_i(x) = s_i x + a_i$, $x \in X$, with the condition that $\cup_{i=1}^N w_i(X) = X$. In other words, the contracted copies, or "tiles" of X, $w_i(X)$, cover X.
2. Associated with each map w_i are the following:
   (a) A scalar $\alpha_i \in \mathbf{R}$ and
   (b) A function $\beta_i : \mathbf{R}_s \to \mathbf{R}_+$, $\beta_i \in L^2(\mathbf{R}_s)$.

The action of the fractal transform $T : Y \to Y$ defined by the above is as follows: For a $u \in Y$ and any $x \in X$,

$$v(x) = (Tu)(x) = {\sum_{i=1}^N}' [\alpha_i (u(w_i^{-1}(x))) + \beta_i]. \tag{3}$$

The prime on the summation signifies that we sum over only those $i \in \{1, 2, \ldots, N\}$ for which the preimage $w_i^{-1}(x)$ exists, i.e., those i for which $x \in w_i(X)$.

The above formulation represents a generalization of the standard fractal transform for greyscale images. The "value" of the HS image $v(x) = (Tu)(x)$ at a point $x \in X$ is a *spectral function*, i.e., $v(x) \in L^2(\mathbf{R}_s)$. Furthermore, the values of $v(x)$ at $t \in \mathbf{R}_s$ are given by

$$v(x; t) = (Tu)(x; t) = {\sum_{i=1}^N}' [\alpha_i u(w_i^{-1}(x); t) + \beta_i(t)]. \tag{4}$$

The function $\beta_i(t)$ replaces the traditional constant β_i employed in standard fractal transforms for (single-valued) images [12,18].

Another way of viewing this procedure is as follows: For each $x \in X$, N copies of the function $u(x;t)$ are first placed at the points $w_i(x)$, $1 \leqslant i \leqslant N$. Each of these copies is then altered in the spectral direction by multiplication by the appropriate α_i factor followed by the addition of the $\beta_i(t)$ function. If two or more modified copies are situated at a point $y \in X$, then they are added together to produce the function $v(y) = (Tu)(y)$.

Theorem 1. *Given the fractal transform T defined above, for any* $u, v \in Y$,

$$d_Y(Tu, Tv) \leqslant K d_Y(u, v), \tag{5}$$

where

$$K = \sum_{i=1}^{N} |s_i|^{1/2} |\alpha_i| \geqslant 0. \tag{6}$$

Proof

$$d_Y(Tu, Tv) = \left[\int_X \left\| \sum_{i=1}^{N} \alpha_i [u(w_i^{-1}(x)) - v(w_i^{-1}(x))] \right\|^2_{L^2(\mathbf{R}_s)} dx\right]^{1/2} \leqslant \sum_{i=1}^{N} \left[\int_{X_i} \|\alpha_i [u(w_i^{-1}(x)) - v(w_i^{-1}(x))]\|^2_{L^2(\mathbf{R}_s)} dx\right]^{1/2}$$
$$= \sum_{i=1}^{N} \left[|s_i| \int_X \|\alpha_i [u(y) - v(y)]\|^2_{L^2(\mathbf{R}_s)} dy\right]^{1/2} = \left[\int_X \|u(y) - v(y)\|^2_{L^2(\mathbf{R}_s)} dy\right]^{1/2} \sum_{i=1}^{N} |s_i|^{1/2} |\alpha_i| = K d_Y(u, v) \qquad \text{QED.}$$

The following is a consequence of Banach's Fixed Point Theorem.

Corollary. *If* $K < 1$, *i.e., T is contractive on Y, then there exists a unique* $\bar{u} \in Y$, *the fixed point of T, such that* $\bar{u} = T\bar{u}$. *Furthermore, let* $u_0 \in Y$ *be any "seed" for the iteration sequence* $u_{n+1} = Tu_n$. *Then* $u_n \to \bar{u}$ *as* $n \to \infty$, *i.e.,* $d_Y(u_n, \bar{u}) \to 0$.

4. Inverse problem for fractal transforms on $(\boldsymbol{Y}, \boldsymbol{d_Y})$

We now wish to consider the following inverse problem, which includes fractal image coding [12,18] as a special case: Given a target element $u \in Y$, find a contractive fractal transform $T : Y \to Y$ such that its fixed point $\bar{u}$ approximates u to a desired accuracy, i.e., $d_Y(\bar{u}, u)$ is sufficiently small. Given the complicated nature of the fractal transform, such direct inverse problems are very difficult. An enormous simplification is yielded by the following consequence of Banach's Fixed Point Theorem, known in the fractal coding literature as the *Collage Theorem* [3].

Theorem 2. *If* $K < 1$, *i.e.,* $T : Y \to Y$ *is contractive with fixed point* $\bar{u} \in Y$, *then for any* $u \in Y$,

$$d_Y(u, \bar{u}) \leqslant \frac{1}{1-K} d_Y(u, Tu). \tag{7}$$

In *collage coding* [17], one looks for a contractive fractal transform T that maps the target u as close as possible to itself, in an effort to make the so-called *collage error*, $d_Y(u, Tu)$, as small as possible.

As mentioned in the Introduction, one of the original motivations for fractal image coding was *image compression* [4,12,18].

5. Block fractal transforms on digital hyperspectral images

The remainder of this paper will be concerned with digital HS images supported on an $N_1 \times N_2$-pixel array, M channels per pixel. Formally, a digital HS image can be represented by a vector-valued image function, $u : X \to \mathbf{R}^M_+$, where $X = \{1, 2, \ldots, N_1\} \times \{1, 2, \ldots, N_2\}$ is the *base* or *pixel space* and R^M_+, the nonnegative orthant of $\mathbf{R}^M$, is the *spectral space*. At a pixel location $(i_1, i_2) \in X$, the hyperspectral image function $u(i_1, i_2)$ is a non-negative M-vector with components $u_k(i_1, i_2)$, $1 \leqslant k \leqslant M$. We shall refer to this vector as the *spectral function* at pixel (i_1, i_2).

Most, if not all, fractal image coding methods employ *block-based* transforms, where subblocks of an image are mapped onto small subblocks of the image, following the original method of Jacquin [16]. Here it might be tempting to simply consider an HS as a "cube" of data and simply move 3D sub-cubes to other sub-cubes. This, however, is contrary to our spirit of function-valued HS images. We wish to see what can be accomplished by keeping the spectral functions intact, or perhaps "partially intact" as we discuss later in this paper.

It would also be tempting to perform fractal image coding on each channel of an HS image separately. Once again, this is contrary to the spirit of function-valued mappings and the desire to keep spectral functions intact. Our goal is to exploit both the spatial self-similarity of channels as well as the correlation between them.

Here we outline a very simple block-based fractal transform for HS images that keeps spectral functions intact. As done in [1] for greyscale images, we let $\mathcal{R}^{(n)}$ denote a set of nonoverlapping $n \times n$-pixel *range* subblocks R_i, such that $X = \cup_i R_i$, i.e., $\mathcal{R}^{(n)}$ forms a partition of the pixel space X. Furthermore, let $u(R_i)$ denote the portion of the HS image function u that is supported on subblock $R_i \subset X$. In this discrete setting, $u(R_i)$ is an $n \times n \times M$ "data cube" of nonnegative real numbers. We also introduce an associated set $\mathcal{D}^{(m)}$ of $m \times m$-pixel *domain* subblocks $D_k \subset X$, where $m = 2n$. This set need not be nonoverlapping, but the blocks should cover the support X, i.e., $\cup_i D_i = X$.

Given an M-channel digital HS image u, the fractal transform operator T will be constructed as follows: For each image subblock $R_i \in \mathcal{R}^{(n)}$, we choose from $\mathcal{D}^{(m)}$ a domain block $u(D_{j(i)})$ in order to produce an approximation of the form,

$$u(R_i) \approx (Tu)(R_i) = \alpha_i u(\widehat{D_{j(i)}}) + \underline{\beta_i}, \quad 1 \leqslant i \leqslant N_R. \tag{8}$$

(The choice of the *best* domain block will be discussed a little later in this section.) Here, $\underline{\beta_i} = (\beta_{i1}, \beta_{i2}, \ldots, \beta_{iM})$ is an M-vector, which plays the role of the $\beta_i(t)$ function in Eq. (4). N_R denotes the cardinality of the set $\mathcal{R}^{(n)}$ and the wide hat denotes an appropriate $2n \times 2n \to n \times n$ pixel decimation operation which produces the geometric contraction in discrete pixel space. Note that only **one** constant α_i is employed for all M-channels supported on the range block R_i. (In other words, the M-channels are **not** coded separately.)

The approximation problem in Eq. (8) may be expressed in the form

$$y_{lm} \approx \alpha x_{lm} + \beta_m, \quad 1 \leqslant l \leqslant N, \ 1 \leqslant m \leqslant M, \tag{9}$$

where $N = N_1 \times N_2$. For simplicity of notation, the $N_1 \times N_2$ matrices in pixel space have been converted into N-vectors. The "stack" of M N-vectors y_{lm} contain the elements of the range block $u(R_i)$ being approximated. As well, the "stack" of M N-vectors x_{lm} contain the elements of the decimated domain block $u(\widehat{D_{j(i)}})$. Note that the subscripts i and $j(i)$ from Eq. (8) have also been omitted for notational ease.

The parameters α and β_m, $1 \leqslant m \leqslant M$, which minimize the squared L^2 distance,

$$\Delta^2 = \sum_{l=1}^{N}\sum_{m=1}^{M}(y_{lm} - \alpha x_{lm} - \beta_m)^2 \tag{10}$$

are given by (details in [22])

$$\alpha = \frac{\sum_{m=1}^{M}\sum_{l=1}^{N} x_{lm}(y_{lm} - \bar{y}_m)}{\sum_{l=1}^{N}\sum_{m=1}^{M} x_{lm}^2 - N\sum_{m=1}^{M}\bar{x}_m^2} \tag{11}$$

and

$$\beta_m = \bar{y}_m - \alpha\bar{x}_m, \quad 1 \leqslant m \leqslant M. \tag{12}$$

Here

$$\bar{x}_m = \frac{1}{N}\sum_{l=1}^{N} x_{lm} \quad \text{and} \quad \bar{y}_m = \frac{1}{N}\sum_{l=1}^{N} y_{lm} \tag{13}$$

denote the (spatial) mean values on each channel.

The set of range-domain assignments $(i, j(i))$, and spectral map parameters, $(\alpha_i, \beta_{i1}, \beta_{i2}, \ldots, \beta_{iM})$, for $1 \leqslant i \leqslant N_R$, comprise the *fractal code* which defines a fractal transform T. If T is contractive (see below), then its fixed point hyperspectral image $\bar{u}$ may be computed by the iteration procedure $u_{n+1} = Tu_n$, where u_0 is any "seed" image. (For convenience, one may employ the zero image $u_0 = 0$.)

Note. In the special case that $M = 1$, Eq. (8) reduces to the usual fractal block transform method for greyscale images.

Technically speaking, Theorem 1 of Section 3 does not apply to block-based fractal transforms since one is not mapping the *entire* image $u(X)$ onto a range block R_i. The determination of L^2 Lipschitz factors such as K in Eq. (6) is quite complicated. Fortunately, we may resort to a simplification which is employed in most block coding methods. In the case of digitized images, it is easy to show that the condition $|\alpha_i| < 1$ for all image range blocks $u(R_i)$ is sufficient to guarantee contractivity of the fractal transform T in the L^∞ norm, from which the existence of a unique fixed point $\bar{u}$ of T follows.

We now return to the question of determining the "best" fractal transform T associated with a given image u, i.e., the transform T with fixed point $\bar{u}$ that approximates u as best as possible. Because the range blocks R_i are nonoverlapping, the sum of the errors associated with the approximations in Eq. (8) defines the total collage error $d_Y(u, Tu)$ on the RHS of Eq. (7). Since our goal is to make the approximation error $d_Y(u, \bar{u})$ on the LHS of Eq. (7) as small as possible, we choose, for each range block $u(R_i)$, the domain block $u(D_{j(i)})$ which *best approximates* $u(R_i)$. If we let Δ_{ik} denote the error in approximating a range block $u(R_i)$ with a domain block $u(D_k)$, i.e.,

$$\Delta_{ik} = \min_{\alpha, \underline{\beta}} \left\| u(R_i) - \alpha u(\widehat{D_k}) - \underline{\beta} \right\|_2, \tag{14}$$

then the index $j(i)$ of the optimal domain block $u(D_{j(i)})$ associated with $u(R_i)$ is

$$j(i) = \arg\min_k \Delta_{ik}. \tag{15}$$

Once again, this is the essence of collage-based fractal image coding.

6. Some numerical results

We now show some results of our block-based fractal transform as applied to the AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) image, "Yellowstone calibrated scene 0", a 512 line, 677 samples per line, 224-channel image, available from the Jet Propulsion Laboratory site [2]. The computations reported below were performed on a 512×512-pixel section of this image. The range blocks R_i used were $N_R = 4096$ nonoverlapping 8×8-pixel blocks. The domain blocks D_i were $N_D = 1048$ nonoverlapping 16×16-pixel blocks.

In Fig. 1 (top) is presented a histogram plot of the distribution of errors Δ_{ij} associated with *all* possible approximations in Eq. (14), i.e., for each range block R_i, we consider *all* possible domain blocks D_k. This distribution is very similar to the

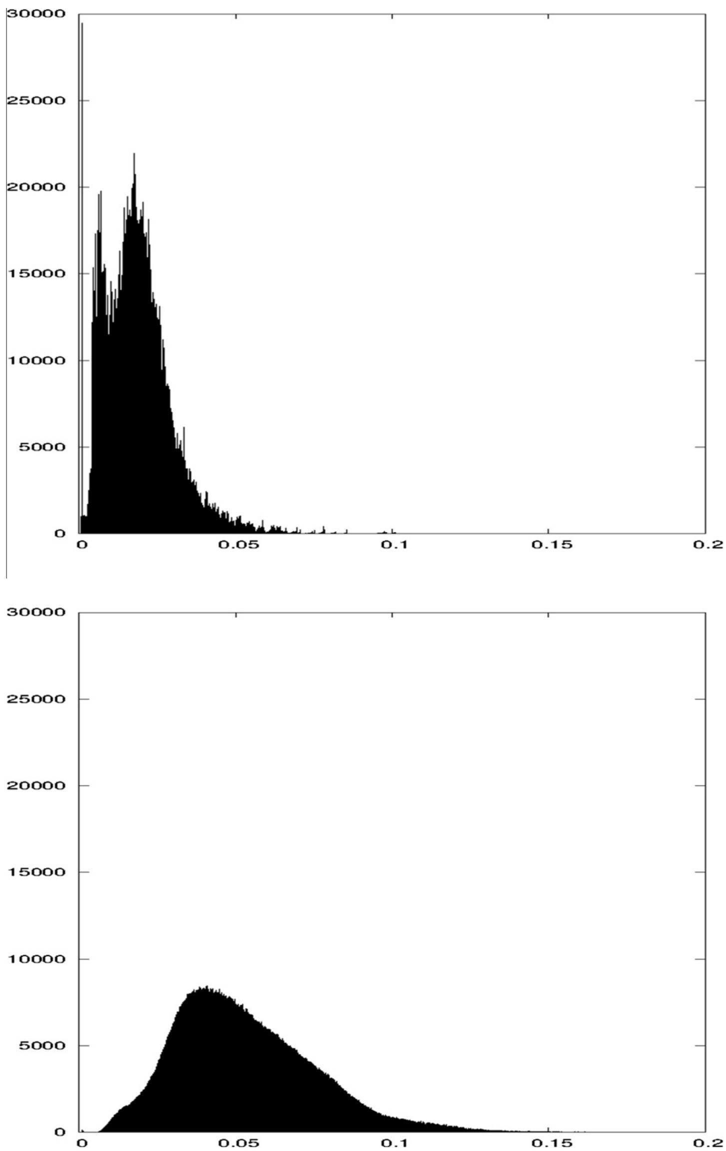


Fig. 1. Distributions of errors Δ_{ij} over the interval $[0, 0.2]$ obtained by approximating nonoverlapping 8×8-pixel range blocks $u(R_i)$ with all possible decimated 16×16-pixel domain blocks of AVIRIS hyperspectral image. Top: using affine mapping, Eq. (8). Bottom: no affine mapping, Eq. (16).

distribution of "Case 3" same-scale approximation errors in [22], where the domain blocks D_k were the same size as the range blocks R_i. The strong peaking of these distributions near zero error shows that many range blocks $u(R_i)$ of the AVIRIS hyperspectral image are well approximated by affinely modified domain blocks $u(D_k)$.

For reference purposes, Fig. 1 (bottom) shows a histogram plot of the error distribution that results when no affine mapping is employed in Eq. (8), i.e., $\alpha_i = 1$ and $\beta_{i1} = \cdots = \beta_{iM} = 0$. Here, each range block $u(R_i)$ is simply approximated by the decimated domain block $u(D_k)$ with error given by

$$\Delta^0_{ik} = \| u(R_i) - u(\widehat{D_k}) \|_2. \qquad (16)$$

The distribution of these errors is more diffuse and, in fact, very similar to the "Case 1" same-scale approximation errors (also with no affine mappings) presented in [22]. Clearly, the use of affine maps yields a significant improvement in approximation.

In Fig. 2 is presented the distribution of α_i coefficients associated with the optimal range blocks $u(R_i)$ employed in the fractal code. The spikes at ± 0.95 are caused by "clamping". For a relatively small number of range blocks, the optimal value of α lies outside the range $(-1, 1)$. In such cases, the α parameter is "clamped" to ± 0.95 and the β vector is computed from this value. This clamping is performed in an effort to insure the stability and numerical convergence of the iteration procedure $u_{n+1} = Tu_n$ used to construct the fixed point attractor $\bar{u}$ of the fractal transform T. The effect of this sub-optimal fitting of a few blocks is negligible.

The absence of a strong peak at $\alpha = 0$ in Fig. 2 represents a significant difference between the α-parameter distributions for HS images and those observed for most (single-valued) greyscale images. The near-zero peaking in greyscale images is generally due to the fact that most of their blocks are quite "flat", i.e., have low variance. As such, they can be well approximated by constant blocks which are produced by α-values close to, if not equal to, zero. This does not seem to be the case for HS images, mostly because the image subblocks are "cubes", i.e., collections of vectors.

Some of the results of this fractal coding procedure are presented in Fig. 3. The fixed point attractor function $\bar{u}$ of the fractal transform T defined by the fractal code for the AVIRIS image – the domain-range assignments $j(i)$, the scaling parameters α_i and the shift vectors $\underline{\beta_i}$ for $1 \leqslant i \leqslant 4096$ – was generated using the iteration procedure $u_{n+1} = Tu_n$, starting with the image $u_0 = \underline{0}$. Reasonable convergence was achieved at u_{10}. Channels 20, 120 and 220 of u_{10} are presented in Fig. 3 on the right along with corresponding channels of the original (uncoded) AVIRIS image on the left.

A closer inspection of these fractal-based approximations shows that the most noticeable errors are located in regions of high image variation, i.e., detailed textures. Furthermore, because of the block-based nature of the coding method, blurriness is also accompanied by blockiness. Such degradations (which are also observed in the fractal coding of standard greyscale images [12,18]) are also seen in images that have been highly compressed using the JPEG. (Recall that the JPEG compression method employs 8×8-pixel blocks.) For purposes of comparison, a JPEG-encoded approximation to Channel 20 is presented in Fig. 4. The Quality Factor of 25 was chosen in order to yield a compression ratio that was roughly equal to that estimated for the fractal coding method (about 40:1).

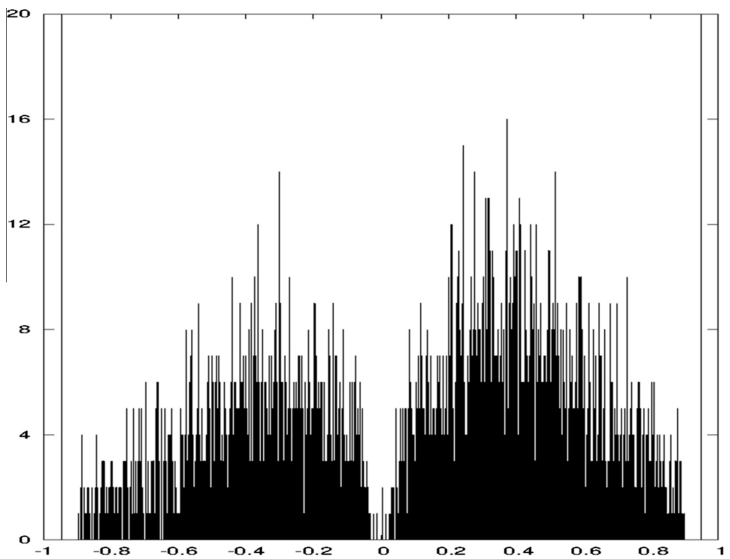
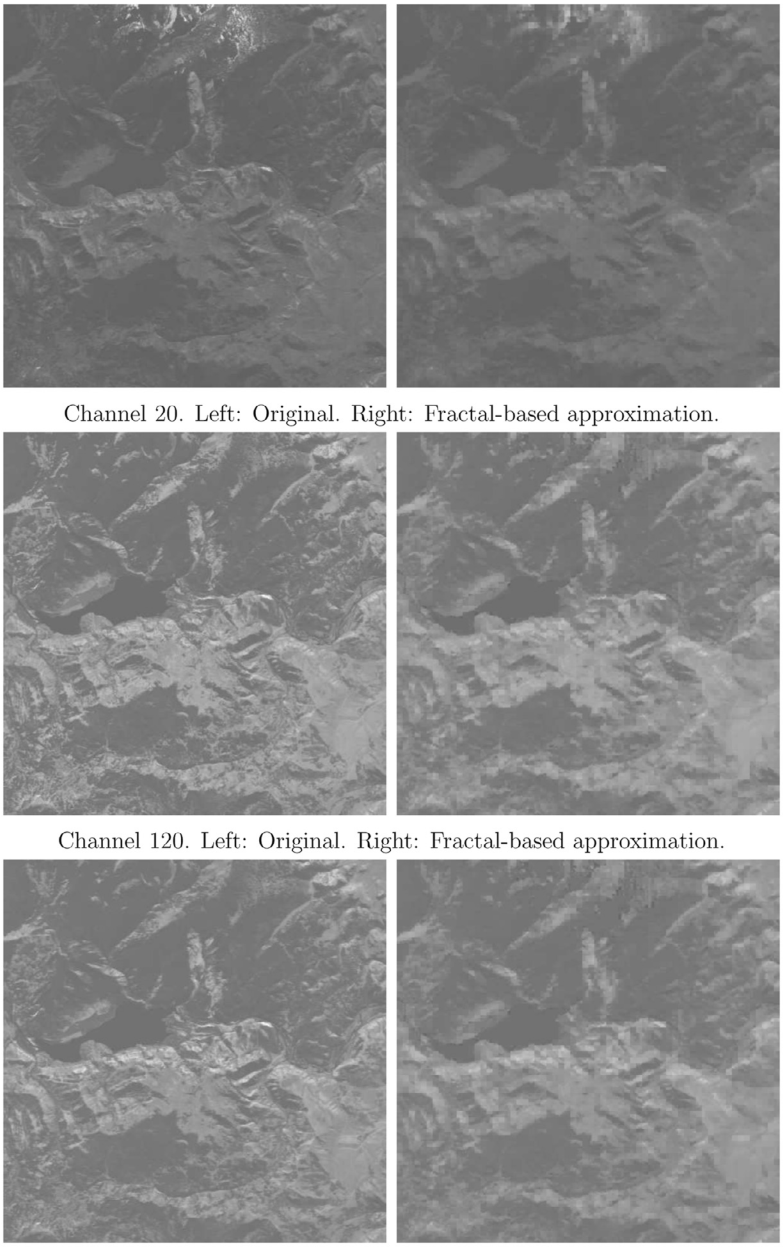


Fig. 2. α_i parameters over the interval $[-1, 1]$ employed in the "best fit" of AVIRIS HS image range subblocks $u(R_i)$.



Channel 20. Left: Original. Right: Fractal-based approximation.

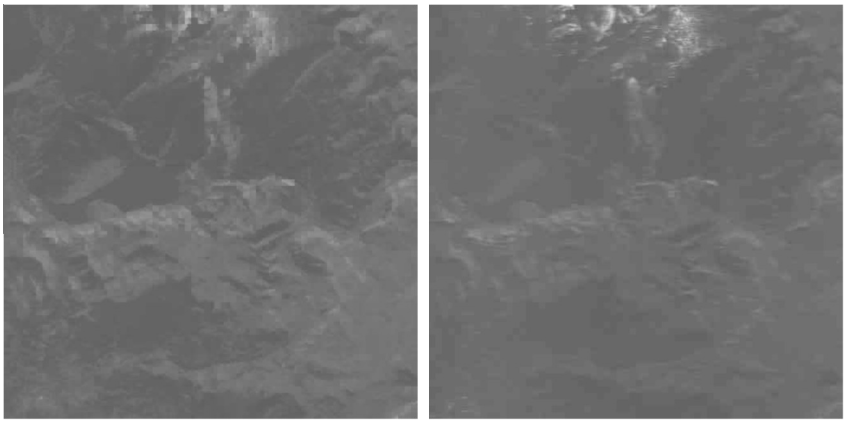
Channel 120. Left: Original. Right: Fractal-based approximation.

Channel 220. Left: Original. Right: Fractal-based approximation.

Fig. 3. Some channels of the attractor $\bar{u}$ of the fractal transform T obtained by fractally coding the AVIRIS hyperspectral image using 8×8-pixel range blocks and 16×16-domain blocks.

As might be expected, the accuracy of the simple fractal coding method employed here may be increased by using smaller range blocks, R_i, for example, 4×4-pixel blocks. However, this increase in accuracy is accomplished at a price – the need to store more fractal code parameters. Such a trade-off between accuracy and storage is the basis of *rate-distortion analysis* – a fundamental issue of image and data compression – which is beyond the scope of this paper.

Admittedly, the use of a regular grid of $n \times n$-pixel blocks is suboptimal. There are more sophisticated methods of partitioning the pixel space X, e.g., quadtree decomposition, rectangular and triangular blocks – for more details, the reader is referred to [12,18]. We simply mention here that the mathematical framework presented here can easily be adapted to accommodate such schemes.



Channel 20. Left: Fractal. Right: JPEG (QF 25).

Fig. 4. 8×8-pixel block approximations to channel 20 of the AVIRIS hyperspectral image.

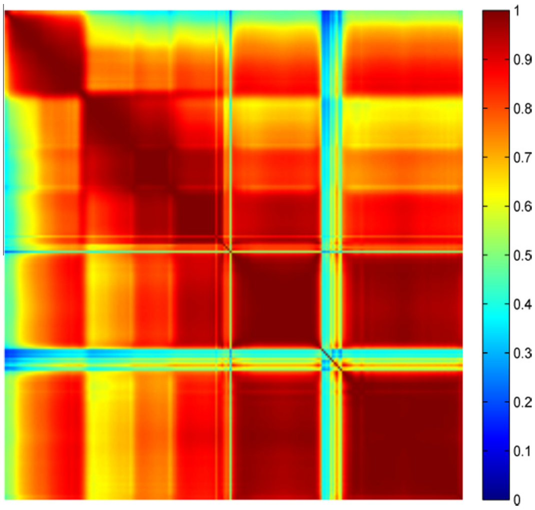


Fig. 5. Correlations C_{kl} between channels of AVIRIS hyperspectral image, demonstrating the existence of several subgroups of highly correlated channels.

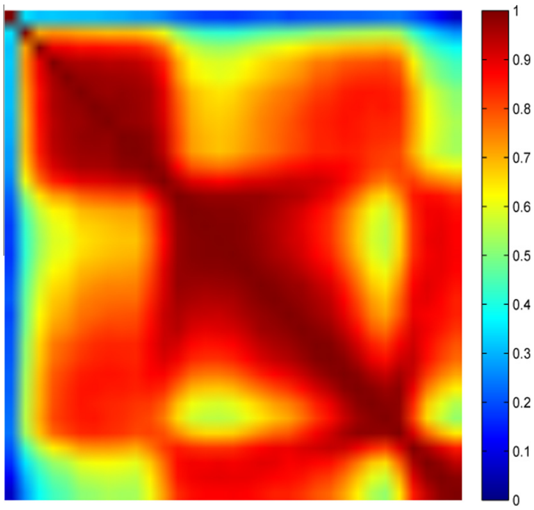


Fig. 6. Correlations C_{kl} between channels of hyperspectral fern image, which demonstrates a lesser degree of grouping of correlated channels.

7. Conclusion

In this paper, we have presented a complete metric space (Y, d_Y) of function-valued mappings that is suitable for the representation of hyperspectral images. As well, a class of fractal transforms over this space has been introduced. Under appropriate conditions, a fractal transform T can be contractive. This leads to an inverse problem in which an element $u \in Y$ is approximated by the fixed point $\bar{u}$ of a fractal transform.

We then consider a simple class of block-based fractal transforms – a slight modification of the transforms introduced earlier – which are particularly suited for digital hyperspectral images, approximating spectral function vectors at one point $x \in X$ by modified vectors from other points. Such an approach is consistent with the method of *nonlocal image processing*.

A block-based transform employing 8×8-pixel range blocks R_i was then employed to fractally code a standard hyperspectral image, the 224-channel AVIRIS "Yellowstone" image. Of course, this transform is in no way optimal. The results could be improved slightly by examining all eight possibilities of mapping square blocks to square blocks, i.e., rotations and inversions. One could also employ smaller range blocks, or a variety of range block sizes, as is done in quadtree-based fractal coding of greyscale images [12,18]. Of course, the price for any improvement is increased computational time.

A further improvement may also be obtained if significant correlations exist between groups of channels in a hyperspectral image. For example, Fig. 5 shows a plot of the correlations between all pairs of channels u_k and u_l, $1 \leqslant k, \ l \leqslant M$, of the AVIRIS image employed in this study, where $M = 224$. Recalling that each channel u_k is an $N_1 \times N_2$ array, the correlation C_{kl} between channels u_k and u_l is computed in the standard fashion, i.e.,

$$C_{kl} = \frac{\sigma_k \sigma_l}{\sigma_{kl}}. \tag{17}$$

Here

$$\sigma_k = \left[\sum_{i_1=1}^{N_1}\sum_{i_2=1}^{N_2}(u_k(i_1, i_2) - \bar{u}_k)^2\right]^{1/2} \tag{18}$$

and

$$\sigma_{kl} = \sum_{i_1=1}^{N_1}\sum_{i_2=1}^{N_2}(u_k(i_1, i_2) - \bar{u}_k)(u_l(i_1, i_2) - \bar{u}_l), \tag{19}$$

where $\bar{u}_k$ denotes the mean of the array u_k. (The usual factor $1/(N_1N_2 - 1)$ accompanying each double summation can be omitted from the formulas for σ_k and σ_{kl} since it cancels out of the formula for C_{kl}.)

The block nature of the plot in Fig. 5 shows that the channels can be divided into at least three subgroups, the channels within each subgroup having higher correlations than with those of other subgroups. As such, it may be advantageous to consider separate fractal transforms, each of which operates within a particular group of channels.

The plot in Fig. 6 of the correlations between channels in the "hyperspectral fern" image [13], which was also examined [22], shows a much lesser amount of internal grouping of channels. As such, a single fractal transform over all channels may suffice.

Acknowledgments

We gratefully acknowledge the partial support of this research by the Natural Sciences and Engineering Research Council of Canada (NSERC) in the form of a Discovery Grant (ERV). Financial support from the Department of Applied Mathematics and the Faculty of Mathematics, University of Waterloo is also gratefully acknowledged (DO).

References

[1] S.K. Alexander, E.R. Vrscay, S. Tsurumi, A simple model for the affine self-similarity of images, in: Image Analysis and Recognition, ICIAR, 2008, Lecture Notes in Computer Science, vol. 5112, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 192–203.
[2] AVIRIS hyperspectral image, Yellowstone calibrated scene 0, available from Information Processing Group website, Jet Propulsion Laboratory, California Institute of Technology, <http://compression.jpl.nasa.gov/hyperspectral/>.
[3] M.F. Barnsley, Fractals Everywhere, Academic Press, New York, 1988.
[4] M.F. Barnsley, L. Hurd, Fractal Image Compression, A.K Peters, Wellesley, Mass, 1993.
[5] D. Brunet, E.R. Vrscay, Z. Wang, Structural similarity-based affine approximation and the self similarity of images revisited, in: Image Analysis and Recognition, ICIAR 2011, Lecture Notes in Computer Science, vol. 6754, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 264–275.
[6] A. Buades, B. Coll, J.M. Morel, A review of image denoising algorithms, with a new one, Multiscale Model. Simul. 4 (2005) 490–530;
A. Buades, B. Coll, J.M. Morel, An updated version of this paper by the same authors: image denoising methods, a new nonlocal principle, SIAM Rev. 52 (2010) 113–147.
[7] C. Chang, Hyperspectral Data Exploitation, Theory and Applications, John Wiley and Sons, Hoboken, NJ, 2007.
[8] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, IEEE Trans. Image Process. 16 (2007) 2080–2095.
[9] M. Ebrahimi, E.R. Vrscay, Solving the inverse problem of image zooming using self-examples, in: Image Analysis and Recognition, ICIAR 2007, Lecture Notes in Computer Science, vol. 4633, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 117–130.
[10] M. Elad, D. Datsenko, Example-based regularization deployed to super-resolution reconstruction of a single image, Comput. J. 50 (2007) 1–16.

[11] C. Etemoglu, V. Cuperman, Structured vector quantization using linear transforms, IEEE Trans. Signal. Process. 51 (2003) 1625–1631.
[12] Y. Fisher (Ed.), Fractal Image Compression: Theory and Application, Springer-Verlag, New York, 1995.
[13] Hyperspectral images of natural scenes 2004, Scene 2, Available from the personal page of David Foster, Manchester University. <http://personalpages.manchester.ac.uk/staff/david.foster/default.htm>.
[14] W.T. Freeman, T.R. Jones, E.C. Pasztor, Example-based super-resolution, IEEE Comput. Graphics Appl. 22 (2002) 56–65.
[15] M. Ghazel, G. Freeman, E.R. Vrscay, Fractal image denoising, IEEE Trans. Image Process. 12 (12) (2003) 1560–1578.
[16] A. Jacquin, Image coding based on a fractal theory of iterated contractive image transformations, IEEE Trans. Image Process. 1 (1992) 18–30.
[17] H. Kunze, D. La Torre, F. Mendivil, E.R. Vrscay, Fractal-based Methods of Analysis, Springer, New York, 2012.
[18] N. Lu, Fractal Imaging, Academic Press, New York, 1997.
[19] G. Mayer, E.R. Vrscay, Iterated Fourier transform systems: A method for frequency extrapolation, in: Image Analysis and Recognition, ICIAR 2007, Lecture Notes in Computer Science, vol. 4633, Springer-Verlag, Berlin, Heidelberg, 2007, pp. 728–739.
[20] O. Michailovich, D. La Torre, E.R. Vrscay, Function-valued mappings and total variation and compressed sensing for diffusion magnetic resonance imaging, in: Image Analysis and Recognition, ICIAR 2012, Lecture Notes in Computer Science, vol. 7325, Part II, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 286–295.
[21] G. Motta, F. Rizzo, J.A. Storer (Eds.), Hyperspectral Data Compression, Springer Science + Business Media Inc., New York, 2006.
[22] E.R. Vrscay, D. Otero, D. La Torre, Hyperspectral images as function-valued mappings, their self-similarity and a class of fractal transforms, in: Image Analysis and Recognition, ICIAR 2013, Lecture Notes in Computer Science, vol. 7950, Springer-Verlag, Berlin, Heidelberg, 2013, pp. 225–234.