

# Continuous evolution of fractal transforms and nonlocal PDE imaging

Edward R. Vrscay

Department of Applied Mathematics  
Faculty of Mathematics  
University of Waterloo  
Waterloo, Ontario, Canada N2L 3G1  
ervrscay@uwaterloo.ca

**Abstract.** Traditional fractal image coding seeks to approximate an image function  $u$  as a union of spatially-contracted and greyscale-modified copies of itself, i.e.,  $u \approx Tu$ , where  $T$  is a contractive *fractal transform* operator on an appropriate space of functions. Consequently  $u$  is well approximated by  $\bar{u}$ , the unique fixed point of  $T$ , which can then be constructed by the discrete iteration procedure  $u_{n+1} = T_n$ .

In a previous work, we showed that the evolution equation  $y_t = Oy - y$  produces a continuous evolution  $y(x, t)$  to  $\bar{y}$ , the fixed point of a contractive operator  $O$ . This method was applied to the discrete fractal transform operator, in which case the evolution equation takes the form of a nonlocal differential equation under which regions of the image are modified according to information from other regions.

In this paper we extend the scope of this evolution equation by introducing additional operators, e.g., diffusion or curvature operators, that “compete” with the fractal transform operator. As a result, the asymptotic limiting function  $y_\infty$  is a modification of the fixed point  $\bar{u}$  of the original fractal transform. The modification can be viewed as a replacement of traditional postprocessing methods that are employed to “touch up” the attractor function  $\bar{u}$ .

## 1 Introduction

The well-known examples of fractal sets, e.g., ternary Cantor set, Sierpinski gasket [8], demonstrate the property of *self-similarity* – each piece of the set can be viewed as a geometrically contracted copy of the entire set. Indeed, such self-similarity properties inspired the seminal paper of M. Barnsley and S. Demko on “iterated function systems” and the inverse problem of fractal construction [4] which, in turn, led to intensive investigations into fractal image coding and compression [3, 5, 9].

It is, however, too much to expect that a general image, viewed as a function or measure, would be well approximated by a union of geometrically contracted and and greyscale modified copies of itself. Following the later idea of A. Jacquin [13], however, images are generally well approximated by a union of spatially-contracted and greyscale modified copies of *subsets* of themselves. This is the

basis of block-based fractal image coding [3, 5, 9] or “local IFS” methods. Indeed, recent experiments by our group [2, 1] confirm that images are generally quite *locally self-similar* – subblocks of an image are typically well approximated by *many* decimated subblocks from elsewhere in the image. In the next section, some evidence to support this statement will be presented.

Mathematically, block-based fractal image coding seeks to approximate an image function  $u$  by the unique fixed point  $\bar{u}$  of a contractive *fractal transform* operator  $T$ . The action of  $T$  on an image function is to replace each subblock  $R_i$  with a shrunk and greyscale modified copy of a subblock  $D_j$  of the image. More details of this procedure will be given in the next section. The fixed point approximation  $\bar{u}$  to  $u$  is generated by the iteration procedure  $u_{n+1} = Tu_n$ , where  $u_0$  can be any starting function or “seed.” Convergence of the  $u_n$  to  $\bar{u}$  is guaranteed by Banach’s fixed point theorem.

In a previous work [6] we introduced the following class of evolution equations associated with contraction mappings  $T$  on a Banach space of functions  $B(X)$ :

$$\frac{\partial u}{\partial t} = Tu - u. \quad (1)$$

The result is that the function  $u(x, t)$  approaches the fixed point  $\bar{u}$  as  $t \rightarrow \infty$  – a *continuous evolution* toward the fixed point  $\bar{u}$  of the contraction map  $T$ , as opposed to the usual discrete sequence of iterates  $u_n$  defined earlier.

We then applied this evolution method to some fractal transform operators on measure and function spaces, in particular, the block-based fractal image coding operator. The original motivation to devise such evolution equations arose from a desire to perform continuous, yet fractal-like “touch-up” operations on images. A fractal-based evolution method could conceivably be used to produce arbitrarily small alterations to an image  $u(x)$  in a neighbourhood of a point  $x_0 \in X$  that depend upon the behaviour of  $u$  at other regions of  $X$ . This is an example of a *nonlocal* imaging operation.

In this paper, we show that one can accommodate additional operators into the evolution operator in Eq. (1), for example, the simple diffusion operator,

$$\frac{\partial u}{\partial t} = \epsilon \nabla^2 u + Tu - u. \quad (2)$$

where positive diffusion, i.e.,  $\epsilon > 0$  corresponds to blurring and negative diffusion  $\epsilon < 0$  produces sharpening. Other differential operators, e.g., anisotropic diffusion [15, 17] could also be considered. In all cases, we have a *nonlocal* partial differential equation governing the evolution of an image function  $u(x, t)$ .

As a consequence, there is a competition between the action of the differential operator and the nonlocal fractal transform operator. For at least small perturbations  $\epsilon$ , the asymptotic limit function  $u_\infty$  is a kind of perturbation of the fixed point  $\bar{u}$  of the contraction operator  $T$ .

At first, the idea of a nonlocal operator might seem to be counterproductive to PDE imaging. In PDE-based enhancement methods such as denoising, the time evolution of an image  $u$  at a point  $x$  is determined completely by its local properties, i.e. value and derivatives, e.g. Gaussian smoothing [17], anisotropic diffusion

[15], total variation minimization [16]. Recently, however, nonlocal methods have been shown to very effective in image enhancement, e.g. the method of “nonlocal means” in image denoising [7]. It is conceivable that such nonlocal methods could make more inroads into PDE imaging methods. Indeed, one purpose of this paper is to show how the nonlocal fractal transform method can easily be incorporated into a PDE-based approach.

The structure of this paper is as follows. In Section 2, we briefly review the basics of block-based fractal image coding, followed by an application of the continuous evolution method. In Section 3, we examine the effects of adding other differential operators to the evolution equation, specifically on the limiting functions. Some concluding statements are made in Section 4.

## 2 Block-based fractal image coding

For simplicity, the support  $X$  of an image function  $u$  will be considered as  $n \times n$  pixel array, i.e., a discrete support. Now consider a partition of  $X$  into nonoverlapping subblocks  $R_i$ ,  $1 \leq i \leq N$ , so that  $X = \cup_i R_i$ . Associated with *range block*  $R_i$  is a larger *domain block*  $D_i \subset X$  so that  $R_i = w_i(D_i)$ , where  $w_i$  is a 1-1 contraction map. (In the pixel case, it will represent the decimation needed to accomplish the spatial contraction.) Assume that the image function  $u(R_i) = u|_{R_i}$  supported on each subblock  $R_i$  is well approximated by a spatially-contracted and greyscale-modified copy of  $u(D_i) = u|_{D_i}$ :

$$u(R_i) \approx \phi_i(u(w_i^{-1}(D_i))), \quad (3)$$

where the  $\phi_i : \mathbf{R} \rightarrow \mathbf{R}$  are greyscale maps that are usually affine in form:

$$\phi_i(t) = \alpha_i t + \beta_i. \quad (4)$$

Because the range blocks  $R_i$  are nonoverlapping, we can write Eq. (3) as

$$u(x) \approx (Tu)(x) = \alpha_i(u(w_i^{-1}(x))) + \beta_i, \quad x \in R_i, \quad (5)$$

where  $T$  is the block fractal transform operator defined by the range-domain assignments and the affine greyscale maps  $\phi_i$ . Under suitable conditions on the  $\alpha_i$  greyscale coefficients and the  $\mathbf{R}^2$  contraction factors of the  $w_i$ , the operator  $T$  is contractive in an appropriate image function space  $\mathcal{F}(X)$  (typically  $L^2(X)$ ) [10].

It is a well-known result that if the *collage distance*  $\|u - Tu\|$  is small, then  $u$  is well approximated by the fixed point  $\bar{u}$  of  $T$ . This is stated more precisely by the *Collage Theorem* [3],

$$\|u - \bar{u}\| \leq \frac{1}{1 - c_T} \|u - Tu\|, \quad (6)$$

where  $c_T$  is the contraction factor of  $T$ . In this way, the inverse problem of fractal image coding can be reformulated into a more tractable problem. Instead

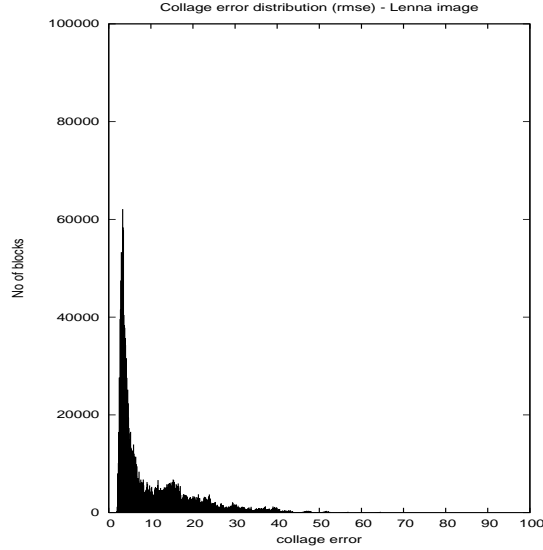
of searching for an operator  $T$  whose fixed point  $\bar{u}$  is close to  $u$ , we search for a  $T$  that maps  $u$  close to itself.

In Figure 1 is presented the fixed point approximation  $\bar{u}$  to the standard  $512 \times 512$  pixel *Lena* image (8 bits per pixel) using a partition of  $8 \times 8$  nonoverlapping pixel blocks ( $64^2 = 4096$  blocks). The “domain pool” for each range block was the set of  $32^2 = 1024$   $16 \times 16$  nonoverlapping pixel blocks. For each  $8 \times 8$  pixel range block  $R_i$ , the  $16 \times 16$  pixel block that provided the best approximation in Eq. (3) was chosen as the domain block  $D_i$  that eventually defined the fractal transform operator  $T$  for this image. The fixed point approximation  $\bar{u}$  was obtained by starting with the seed image  $u_0(x) = 255$  (plain white image) and iterating  $u_{n+1} = Tu_n$  to  $n = 15$ . The iterates  $u_1$ ,  $u_2$  and  $u_3$  are also shown in this figure.



**Fig. 1.** Starting at upper left and moving clockwise: The iterates  $u_1$ ,  $u_2$  and  $u_3$  along with the fixed point  $\bar{u}$  of the fractal transform operator  $T$  designed to approximate the standard  $512 \times 512$  (8 bpp) *Lena* image. The “seed” image was  $u_0(x) = 255$  (plain white).

Finally, we return to the comments made in the introduction regarding the statistical local self-similarity property of images. A series of extensive experiments by our group [2, 1] has shown that images are generally quite *locally self-similar*: Using the terminology of fractal block coding developed above, range blocks  $R_i$  of an image are typically well approximated by a good number of domain blocks  $D_j$  from the image. To illustrate, we return to the *Lena* image of Figure 1 and the fractal coding procedure used to produce the fixed point  $\bar{u}$  in that figure. Recall that 4096  $8 \times 8$  nonoverlapping range blocks and 1024 nonoverlapping  $16 \times 16$  nonoverlapping domain blocks were used in the coding.



**Fig. 2.** Histogram distribution of collage errors  $\Delta_{ijk}$ , cf. Eq. (7) for all domain-range block pairings for *Lena* image.

In Figure 2 is presented a histogram plot of the collage errors for *all possible range/domain block pairs*  $R_j/D_i$ , with all 8 square-to-square isometries being considered:

$$\Delta_{ijk} = \min_{\alpha, \beta} \| R_i - \alpha u(w_j^{(k)}(D_j)) - \beta \|, \quad (7)$$

$$1 \leq i \leq 1024, \quad 1 \leq j \leq 4096, \quad 1 \leq k \leq 8,$$

for a total of 33,554,432 collage errors. (Of course, for each range block  $R_i$ , the domain block  $D_j(i)$  and mapping  $w_i^{k(i)}$  producing the lowest collage error was chosen for the fractal code that produced Figure 1.) Although there are very few, if any, exact matchings, a very pronounced peak is situated not far from zero collage error. Of course, the histogram distributions vary from image to image.

Images such as *Mandrill*, which is well known in presenting difficulties for any kind of image coding, demonstrate more flattened histogram distributions. A purely random image, i.e., Gaussian noise with variance  $\sigma$  will demonstrate a Gaussian-like distribution of collage errors that peaks at  $\sigma$ .

### 3 Continuous evolution of fractal transform operators

We now employ the block-based fractal transform operator  $T$ , defined in Eq. (5), in the continuous evolution Eq. (1). The result is a nonlocal partial differential equation in the image function  $u(x, t)$ :

$$\frac{\partial u(x, t)}{\partial t} = \alpha_i u(w_i^{-1}(x), t) + \beta - u(x, t), \quad x \in R_i. \quad (8)$$

In Figure 3 are shown some steps in the time evolution of images  $u(x, t)$  as determined by the above evolution equation, where  $T$  is the fractal transform operator used in the construction of Figure 1. In this case, the time evolution proceeds at a slower pace than in Figure 1, starting at  $u_0 = 255$  (plain white image) and proceeding in time steps of 0.2. A simple forward Euler scheme using a step size of  $h = 0.1$  was used.

#### 3.1 Evolution in the presence of diffusion

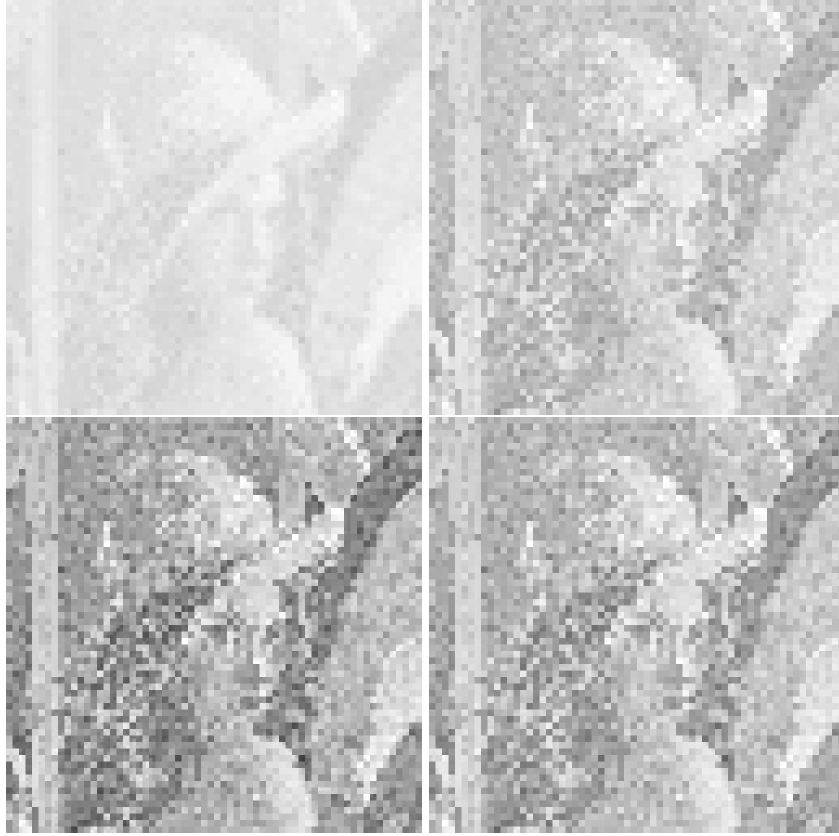
In this section, we examine the effects of the isotropic diffusion term in Eq. (2) for various values of the diffusion constant  $\epsilon$ .

##### Case I: $\epsilon > 0$

In Figure 4 are presented the limiting images for  $\epsilon$  values of 0.5, 5.0, 10.0 and 20.0. Eq. (2) was integrated to  $t = 50$  using a simple Euler scheme with time step  $h = 0.01$ . As  $\epsilon$  increases, the blurring effects of the diffusion operator become more pronounced.

In standard, i.e., discrete, fractal image coding, there has always been a concern about the blockiness exhibited by the fixed points of fractal transform operators. This is mostly due to the partitions used to produce the range blocks – most often, there is no attempt to perform any kind of “patching” of neighbouring blocks over their common boundaries. Various methods have been used to reduce such blockiness. One method, that of “postprocessing,” was to blur the final fixed point image, either over its entirety or selectively across the boundaries of the range blocks. Another is to blur the image after each application of the fractal transform operator  $T$ , a kind of intermediate processing. The diffusion operator in Eq. (2) essentially performs such an intermediate processing in a continuous manner. The asymptotic images are no longer fixed points of the fractal transform operator  $T$  but rather solutions to the partial differential equation

$$\epsilon \nabla^2 y + Ty - y = 0. \quad (9)$$



**Fig. 3.** Starting at upper left and moving clockwise: The images  $u(x, t)$  at times 0.2, 0.4, 0.6, 0.8 produced from  $u(x, 0) = 255$  (plain white) under evolution by  $u_t = Tu - u$  where  $T$  is the fractal transform whose discrete iteration was shown in Figure 1.



**Fig. 4.** Starting at upper left and moving clockwise: The limiting images  $u(x, t), t \rightarrow \infty$  produced by integrating Eq. (2) for  $\epsilon$  values of 0.5, 5.0, 10.0 and 20.0, respectively.

For small values of  $\epsilon$ , these asymptotic limiting functions could possibly be viewed as perturbations of the fixed point function  $\bar{u}$  of  $T$ , a subject of current investigation.

### Case 2: $\epsilon < 0$

For this case of “negative diffusion,” we expect the limiting images  $u(x, t), t \rightarrow \infty$  to be sharpened. In Figure 5 is presented the limiting image for  $\epsilon = -0.1$ . This image was obtained by integrating Eq. (2)  $t = 50$  using the simple Euler forward scheme with time step  $h = 0.001$ . Such a small value for the time step was used because of the numerical instability involved with negative diffusion. The image is a somewhat sharpened version of the fixed-point *Lena* image of Figure 1. Unfortunately, the sharpening enhances not only the edges present in the image but also those that lie along the boundaries of the  $8 \times 8$  range blocks.



**Fig. 5.** Left: The limiting image  $u(x, \infty)$  produced by integrating Eq. (2) for  $\epsilon = -0.1$ , corresponding to negative diffusion. Euler method, step-size  $h = 0.001$ , integrated to  $t = 50$ . Right: The limiting image for  $\epsilon = 0$ , presented for comparison.

Difficulties are encountered in the numerical computations when  $\epsilon$  is chosen to be lower than  $-0.1$ . For example, when  $\epsilon = -0.2$ , a great deal of irregularity develops in the image, erupting into virtual instability at  $t \approx 7.5$  and virtual randomness at  $t = 20$ . Such a negative diffusion scheme is known to be numerically unstable. At the time of writing, it is not clear whether the  $t = 20$  result is due to this instability or the actual lack of a nontrivial asymptotic limiting function because of the nonsmooth nature of the image function  $u(x, t)$ . If the latter applies, then there may be some kind of critical value  $\epsilon_0 \in [-0.2, -0.1]$  at which a limiting function ceases to exist. The negative diffusion term can be viewed as a process that competes against the term  $Tu - u$  which is trying to drive the evolution toward the fixed point  $\bar{u}$  of  $T$ .

### 3.2 Evolution for a convex combination of contraction mappings

Let us now suppose that we have a set of contraction maps driving the evolution in Eq. (1) in parallel. It is natural to consider a convex combination. Let  $T_i$ ,  $i = 1, 2, \dots, n$  be a set of contraction maps with contraction factors  $c_i \in [0, 1)$  fixed points  $\bar{y}_i$ . Now let  $\lambda_i$ ,  $i = 1, 2, \dots, n$ , be a partition of unity, i.e.,  $\lambda_i \in (0, 1)$  with  $\sum_i^n \lambda_i = 1$ , and consider the evolution equation

$$\frac{\partial y}{\partial t} = \sum_{i=1}^n \lambda_i (T_i y - y). \quad (10)$$

This equation may be rewritten as

$$\frac{\partial y}{\partial t} = T y - y, \quad (11)$$

where

$$T = \sum_{i=1}^n \lambda_i T_i. \quad (12)$$

Now, for any  $u, v \in \mathcal{F}(X)$ ,

$$\begin{aligned} \|Tu - Tv\| &= \left\| \sum_{i=1}^n \lambda_i T_i u - \sum_{i=1}^n \lambda_i T_i v \right\| \\ &\leq \sum_{i=1}^n \lambda_i \|T_i u - T_i v\| \\ &\leq \sum_{i=1}^n \lambda_i c_i \|u - v\|. \end{aligned} \quad (13)$$

Since  $c = \sum_{i=1}^n \lambda_i c_i \in [0, 1)$ , it follows that  $T$  is a contraction mapping with a unique fixed point  $\bar{y}$ . From our earlier result associated with Eq. (1), it also follows that  $\bar{y}$  is a globally asymptotically stable solution of Eq. (10).

Numerical experiments, where the  $T_i$  are fractal transform operators corresponding to different images, have demonstrated convergence to limiting asymptotic functions  $u_\infty$ . These fixed points demonstrate characteristics of the different images – a kind of interpolation. (However, the blockiness of the fractal transform operators makes these interpolations rather unattractive.) In fact, “interpolation” is not the most appropriate term. Eq. (10) actually represents a continuous version of the iterative method of *projections onto convex sets* (POCS) that has been used in a variety of applications including imaging [18].

It is interesting to consider the situation where the  $T_i$  are *different* fractal transform operators for the *same* image. For example, for each range block  $R_i$  we could consider a number of domain blocks, possibly weighting each block inversely according to the collage error that it yields. The idea of using multiple domain blocks in order to improve the accuracy of approximation has been

around for some time (see, for example, [9]). However, S. Alexander [1] has recently shown that this idea can be used effectively for purposes other than simply increasing the PSNR, for example, denoising, edge detection and block classification. We simply outline here the idea of how denoising can be accomplished.

### Fractal image denoising

Firstly, it is well known that lossy compression schemes can denoise an image, which is also the case in fractal image coding. Assuming that the noise is additive, stationary and of zero mean, the spatial contraction/decimation that comprises the mapping of domain to range blocks contributes to the reduction of the variance of the noise. (We have shown [11] that the degree of denoising can be increased by improving the fractal code of the “denoised” attractor.) Alexander’s scheme, however, capitalizes on the fact, mentioned earlier, that there are generally several domain blocks that match a given range block almost as well as the “optimal” block, i.e., the block yielding the lowest collage error. (Most often, the difference is very small.) A number of these lowest-error blocks are then employed to produce the modified range block, a kind of weighted averaging which reduces the noise. Several strategies were investigated and work very well [1]. These methods are easily incorporated into a continuous evolution formulation but will not be discussed any further here.

Finally, we mention that an analogous “nonfractal” strategy has been adopted for denoising, namely, the “patching” of a domain block with a number of blocks of the same size from elsewhere in the image [7]. This method relies on *translational* similarity of an image as opposed to the *scaling similarity* inherent in fractal coding approaches.

## 4 Concluding remarks

In a previous work, we showed that the evolution equation  $y_t = Oy - y$  produces a continuous evolution  $y(x, t)$  to  $\bar{y}$ , the fixed point of a contractive operator  $O$ . When applied to a fractal block coding operator  $T$ , the evolution equation takes the form of a nonlocal differential equation so that a region (range block) of an image is being continuously modified by another region (domain block).

In this paper we have extended the scope of this evolution equation by introducing additional operators, e.g., diffusion, that “compete” with the fractal transform operator. As a result, the asymptotic limiting function  $y_\infty$  is a modification of the fixed point  $\bar{u}$  of the original fractal transform. In the case of diffusion, the modification can be viewed as a replacement of traditional post-processing methods that are employed to “touch up” the attractor function  $\bar{u}$ . Negative diffusion produces a sharpening of  $\bar{u}$ . It now remains to establish theoretically the existence of such limiting asymptotic functions.

This continuous evolution method will easily accommodate other operators, e.g., nonlinear anisotropic diffusion operators [15, 17]. A possible concern, however, is the artificial blockiness that is introduced by the fractal coding method

from the range block boundaries. There are a number of ways to reduce these problems, including (i) overlapping range blocks and (ii) a kind of Gaussian window of the domain-to-range block mapping that extends beyond the usual rigid range block boundaries.

Finally, we mention that the continuous evolution formalism can actually help to suggest other discrete iteration processes that can be employed with fractal image coding. As was pointed out in our earlier work [6], when a simple forward Euler scheme with time step  $h > 0$  is employed for the integration of Eq. (1), we obtain the discrete iteration process

$$\begin{aligned} y_{n+1} &= y_n + h(Ty_n - y_n) \\ &= (1 - h)y_n + hTy_n. \end{aligned} \tag{14}$$

For  $0 < h < 1$ , the  $y_{n+1}$  produced by the Euler method is a linear interpolation between  $y_n$  and  $Ty_n$ . In the case  $h = 1$ , we obtain the usual iteration procedure  $y_{n+1} = Ty_n$ . The discussion in Section 3.2 above suggests a discrete formulation for a convex combination of “competing” contraction maps. Indeed, this formulation is quite analogous to the method of projections onto convex sets (POCS) [18]. One may also consider the use of penalty functions in the evolution equation to “steer” the time evolution.

## Acknowledgements

This research has been supported in part by the Natural Sciences and Engineering Research Council of Canada, which is hereby gratefully acknowledged.

## References

1. S.K. Alexander, *Multiscale methods in image modelling and image processing*, Ph.D. Thesis, Department of Applied Mathematics, University of Waterloo (2005).
2. S.K. Alexander, E.R. Vrscay and S. Tsurumi, An examination of the statistical properties of domain-range block matching in fractal image coding (preprint).
3. M.F. Barnsley, *Fractals Everywhere*, Academic Press, New York (1988).
4. M.F. Barnsley and S. Demko, Iterated function systems and the global construction of fractals, *Proc. Roy. Soc. London* **A399**, 243-275 (1985).
5. M.F. Barnsley and L.P. Hurd, *Fractal Image Compression*, A.K. Peters, Wellesley, Massachusetts (1993).
6. J. Bona and E.R. Vrscay, “Continuous evolution of functions and measures toward fixed points of contraction mappings,” in *Fractals in Engineering: New Trends in Theory and Applications*, edited by J. Levy-Vehel and E. Lutton (Springer Verlag, London, 2005), pp. 237-253.
7. A. Buades, B. Coll and J.-M. Morel, A nonlocal algorithm for image denoising, *CVPR* (2), 60-65, 2005.
8. K. Falconer, *The Geometry of Fractal Sets*, Cambridge University Press, Cambridge (1985).
9. Y. Fisher, *Fractal Image Compression*, Springer Verlag, New York (1995).

10. B. Forte and E.R. Vrscay, Theory of generalized fractal transforms, in *Fractal Image Encoding and Analysis*, edited by Y. Fisher, NATO ASI Series F 159, Springer Verlag, New York (1998).
11. M. Ghazel, G. Freeman and E.R. Vrscay, Fractal image denoising, *IEEE Transactions on Image Processing* **12**, no. 12, 1560-1578, 2003.
12. J. Hutchinson, Fractals and self-similarity, *Indiana Univ. J. Math.* **30**, 713-747 (1981).
13. A. Jacquin, Image coding based on a fractal theory of iterated contractive image transformations, *IEEE Trans. Image Proc.*, **1** 18-30 (1992).
14. N. Lu, *Fractal Imaging*, Academic Press, New York (1997).
15. P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, *IEEE Trans. PAMI* **12**, 629-639 (1990).
16. L. Rudin, S. Osher and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* **60**, 259-268 (1992).
17. G. Sapiro, *Geometric partial differential equations and image analysis*, Cambridge University Press, New York (2001).
18. D. Youla and H. Webb, *Image restoration by the method of convex projections: Part 1-Theory*, *IEEE Transactions on Medical Imaging*, vol. MI-1, no. 2, pp. 81-94, October 1982.