REGULARIZED FRACTAL IMAGE DECODING

Mehran Ebrahimi
Department of Applied Mathematics
University of Waterloo
Waterloo, Ontario, Canada
e-mail: m2ebrahi@uwaterloo.ca

Edward R. Vrscay
Department of Applied Mathematics
University of Waterloo
Waterloo, Ontario, Canada
e-mail: ervrscay@uwaterloo.ca

Abstract

The goal of this paper is to present a new recipe for the fractal image decoding process. In this paper, we explain how fractal-based methods can be internally combined with regularization schemes, e.g., Tikhonov, Total Variation (TV), or Hard-Constrained regularization. Although the regularization procedure is very common in context of algebraic image restoration, it has not yet been thought directly in the context of fractal-based methods. This implication can be advantageous in many ways to improve the quality of the decoded image depending on the regularization functional. We develop the theory and apply the standard iterative methods of steepest descent and projected Landweber. We apply our technique to the under-determined missing fractal code problem as verification to the theory presented.

Keywords *— Fractal Image Decoding, Fractal Image Coding, Tikhonov Regularization, Total Variation.*

1 Introduction

In traditional fractal-based image coding techniques, images are thought as elements of a complete metric space. A contractive operator, the so-called *fractal transform* T, is estimated at the encoding stage, such that its fixed point $\bar{X}$ is an approximation of the original image X. At the decoding stage, this operator is iteratively applied to any initial image. Banach's contraction mapping theorem guarantees the convergence of this iterative scheme. This decoding procedure is in fact too restrictive, however, due to the fixed point iteration scheme. Regardless of the initial seed image and any other prior information about the image, the traditional fractal decoding procedure converges to the fixed point $\bar{X}$ of T. An interesting question is whether fractal image encoding/decoding can be combined with other extra knowledge about the image – i.e., can we regularize the estimated image. Regularization can be applied to both ill-posed or well-posed problems to obtain superior results.

In our framework, the problem of fractal image decoding will be modeled as the solution of a linear equation with a sparse coefficient matrix. Due to the structure of the coefficient matrix, the solution of this linear system is unique and coincides with the attractor of the contractive fractal transform T. Traditional fractal decoding scheme finds a solution of this linear system in a greedy fashion.

In the technique proposed in this paper, we construct a minimization problem for the fractal linear equation along with the regularization functional. We present some of our experiments in the context of regularized fractal decoding regarding the solution of the problem of missing fractal codes which was first introduced in [5].

2 Basics of Fractal Image Encoding and Decoding

More details on fractal image coding can be found in many places [1], [3], [7], [11], [9], including one of our recent papers on generalized fractal image coding using projections onto convex sets [5]. Fractal image coding seeks to approximate an image by a union of spatially-contracted and greyscale-modified copies of subblocks of itself. If we let the image of interest be represented by an image function $X(x, y)$, then the result of the coding procedure is a contractive mapping T, the so-called *fractal transform* operator, the fixed point $\bar{X}$ of which provides an approximation to X. In other words,

$$X \cong \bar{X} = T\bar{X}. \tag{1}$$

To obtain T, the image is first partitioned (e.g., uniform, quadtree) into a set of nonoverlapping range blocks C_i. For each range block C_i, one searches for a larger domain block $P_{J(i)}$ (from an appropriate "domain pool" $\mathcal{P}$ that is often common to all range blocks of the same size) such that $X\Big|_{C_i}$ (by this notation we mean, the block C_i of X) is approximated by a modified copy of $X\Big|_{P_{J(i)}}$, i.e.,

$$X\Big|_{C_i} \cong \phi_i\Big(X\Big|_{P_{J(i)}}\Big) = \phi_i\Big(X\Big|_{w_i^{-1}(C_i)}\Big), \tag{2}$$

where $\phi_i : \mathbf{R} \to \mathbf{R}$ is a greyscale map that operates on pixel intensities and w_i denotes the 1-1 contraction/decimation that maps pixels of $P_{J(i)}$ onto pixels of C_i. The *fractal code* defining T consists of the maps ϕ_i as well as the domain-range assignments determined during the coding procedure. In practice, greyscale maps are assumed to be affine, i.e.,

$$\phi_i(t) = \alpha_i t + \beta_i. \tag{3}$$

For a given domain-range block pair $P_{J(i)}/C_i$, the optimal value of the α and β parameters is usually accomplished by means of least-squares fitting.

At the decoding stage, given a contractive fractal transform T, we may generate its fixed point $\bar{X}$ by iteration $X_{n+1} = T(X_n)$, starting with an arbitrary image X_0.

$$(X_{n+1})\Big|_{C_i} = \Big(T(X_n)\Big)\Big|_{C_i} = \alpha_i D\Big((X_n)\Big|_{P_{J(i)}}\Big) + \beta_i. \tag{4}$$

in which D in this Equation is the down-sampling operator. As mentioned earlier, Banach's contraction mapping theorem guarantees that the sequence of images X_n converges to $\bar{X}$.

3 Regularization Model of Fractal-Transform Operator

In this section, assume that T is the approximated fractal transform of an $m \times m$ image X, i.e., X is approximated by the fixed point $\bar{X}$ of T. Also for convenience assume that the blocks in both domain and range pools are nonoverlapping.

In this paper, by notation $\widetilde{\mathbf{X}} = X$ we mean $\mathbf{X}$ is the lexicographic representation of image X. Given T we can define a corresponding $\mathbf{T}$ acting on $\mathbf{X}$ such that $\widetilde{\mathbf{T}(\mathbf{X})} = T(X)$. $\mathbf{T}$ is in fact the same as T acting on X when written in lexicographic order and returning the corresponding image $T(X)$ when rearranged in the lexicographic order.

By these notations, the result of the traditional fractal image decoding in the lexicographic form is

$$\bar{\mathbf{X}} = argmin_{\mathbf{X}} \| \mathbf{X} - \mathbf{T}(\mathbf{X}) \|^2. \tag{5}$$

We take advantage of writing X in the lexicographic order, that operator T can be written as

$$\mathbf{T}(\mathbf{X}) = M\mathbf{X} + B. \tag{6}$$

A similar type of this matrix representation of the fractal transform operator was introduced in [10]. Here M is an $N \times N$ matrix, and both $\mathbf{X}$ and B are vectors of dimension $N \times 1$, where $N = m^2$. The matrix M carries the α_i information, where B contains the information of β_i, all in the appropriate locations. A nominal row of of matrix M corresponding to a point in the range block C_i has the following form.

$$\begin{pmatrix} \cdots\cdots\cdots & \cdot & \cdot & \cdots\cdots & \cdot & \cdots & \cdots\cdots\cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0\ldots\ldots0 & \frac{\alpha_i}{4} & \frac{\alpha_i}{4} & 0\ldots0 & \frac{\alpha_i}{4} & \frac{\alpha_i}{4} & 0\ldots\ldots0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots\cdots\cdots & \cdot & \cdot & \cdots\cdots & \cdot & \cdots & \cdots\cdots\cdots \end{pmatrix}$$

M is in general a huge and sparse matrix with the ratio of non-zeros to total of $4/N = 4/m^2$, for an $m \times m$ image. Figure 1 shows sparsity structures of matrix M for an small image.

Therefore, $\bar{\mathbf{X}}$ is the solution of a linear equation of the form

$$M\mathbf{X} + B = \mathbf{X}, \tag{7}$$

or equivalently

$$A\mathbf{X} = B, \tag{8}$$

where $A = I_{N \times N} - M$, and $I_{N \times N}$ is the identity matrix of size $N \times N$. In the case that the fractal transform is fully known, this linear system has a unique solution $\bar{\mathbf{X}}$, indeed by the structure of A. This unique solution is in fact the attractor of the fractal transform T in the lexicographic order, i.e., $\bar{\mathbf{X}}$. It is also true that

$$A\mathbf{X} - B = \mathbf{X} - T(\mathbf{X}), \tag{9}$$

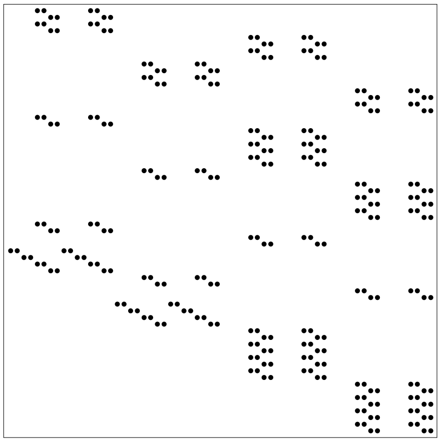


Figure 1. Typical sparsity of matrix M, for an 8×8 image, i.e., $m = 8$. M is of dimension 64×64, i.e., $N = m^2 = 64$. Range and domain blocks are respectively of size 2×2 and 4×4. Number of nonzeros of M are $4N = 256$ out of the total of $N^2 = 4096$ elements.

and hence finding the fixed point of the fractal transform operator in fact the same finding the unique minimizer of the collage error. Equivalently, therefore, the image is approximated by solution of the least squares problem

$$\bar{\mathbf{X}} = argmin_{\mathbf{X}} \| A\mathbf{X} - B \|^2. \tag{10}$$

Now we consider the new objective function

$$G_\lambda(\mathbf{X}) = \frac{1}{2} \| A\mathbf{X} - B \|^2 + \lambda \mathbf{\Lambda}(\mathbf{X}), \tag{11}$$

in which some convex and differentiable regularization function $\mathbf{\Lambda}(\mathbf{X})$ is controlled by λ. The idea is to improve the traditional fractal attractor $\bar{\mathbf{X}}$ by minimizing this new objective function.

This is especially very attractive when the $\| A\mathbf{X} - B \|^2$ does not have a unique minimizer – for example, the missing fractal code problem that we consider Section 6. However, even when $\| A\mathbf{X} - B \|^2$ has a unique minimizer one can improve the solution by choosing the appropriate values of λ and function $\mathbf{\Lambda}(\mathbf{X})$. Hence, we define

$$\bar{\mathbf{X}}_{reg(\lambda)} = argmin_{\mathbf{X}} G_\lambda(\mathbf{X}). \tag{12}$$

$\bar{\mathbf{X}}_{reg(\lambda)}$ might be solved analytically depending on the regularization function, but the goal is to numerically estimate $\bar{\mathbf{X}}_{reg(\lambda)}$ in general.

In the next Section, we present a numerical algorithm, so-called regularized fractal decoding algorithm (RFDA), to estimate $\bar{\mathbf{X}}_{reg(\lambda)}$.

4 Regularized Fractal Decoding Algorithm (RFDA)

Constructing the huge sparse matrix $A = (I_{N\times N} - M)$, is computationally inefficient. Matrices A and M are not only sparse, but also redundant in the nonzero values. Vector B is also redundant. The redundancy is based on the fact that the values of α_i and β_i are the same for all of the points in range blocks that correspond to some domain block. A fixed α_i and β_i's can appear many times respectively in A and B depending on the size of range and domain blocks.

In this Section, we present an algorithm based on the Steepest Descent (SD) method to minimize $G_\lambda(\mathbf{X})$, and present an efficient algorithm to minimize $G_\lambda(\mathbf{X})$, without directly dealing with A and B.

Because objective function in Equation 11 is convex the numerical algorithms for minimizing the convex functions can be applied to find $\bar{\mathbf{X}}_{reg(\lambda)}$, the unique minimizer of $G_\lambda(\mathbf{X})$. It turns out that SD is not the only numerical method one may apply to find $\bar{\mathbf{X}}_{reg(\lambda)}$, and not necessarily the most efficient algorithm.

By SD, the sequence $\{\mathbf{X_n}\}$ defined by

$$\mathbf{X}_{n+1} = \mathbf{X}_n - \mu\nabla G_\lambda(\mathbf{X}), \qquad (13)$$

converges to an approximation of $\bar{\mathbf{X}}_{reg(\lambda)}$ if μ, the scalar step-size in the direction of gradient is sufficiently small. The convergence trend could be improved by taking advantage of a variable step-size, i.e., by taking a sequence μ_n, but in this paper we use only a fixed step-size μ.

Differentiating $G_\lambda(\mathbf{X})$, and substituting in 13 yields

$$\begin{aligned}\mathbf{X}_{n+1} &= \mathbf{X}_n + \mu A^T(B - A\mathbf{X}_n) - \mu\lambda\nabla\mathbf{\Lambda}(\mathbf{X_n}) \quad (14)\\ &= \mathbf{X}_n + \mu(I_{N\times N} - M^T)(B - A\mathbf{X}_n)\\ &\quad -\mu\lambda\nabla\mathbf{\Lambda}(\mathbf{X_n})\\ &= \mathbf{X}_n + \mu(I_{N\times N} - M^T)(\mathbf{T}(\mathbf{X}_n) - \mathbf{X}_n)\\ &\quad -\mu\lambda\nabla\mathbf{\Lambda}(\mathbf{X_n})\\ &= \mathbf{X}_n + \mu(\mathbf{T}(\mathbf{X}_n) - \mathbf{X}_n)\\ &\quad -\mu M^T(\mathbf{T}(\mathbf{X}_n) - \mathbf{X}_n) - \mu\lambda\nabla\mathbf{\Lambda}(\mathbf{X_n}).\end{aligned}$$

Now defining $\mathbf{Y}_n = \mathbf{T}(\mathbf{X}_n) - \mathbf{X}_n$ and taking tilde, i.e., converting each image from column-stacked lexicographic order to matrix representation gives

$$\widetilde{\mathbf{X}}_{n+1} = \widetilde{\mathbf{X}}_n + \mu(\widetilde{\mathbf{Y}_n} - \widetilde{M^T\mathbf{Y}_\mathbf{n}}) - \mu\lambda\nabla\widetilde{\mathbf{\Lambda}(\mathbf{X}_\mathbf{n})}. \qquad (15)$$

Finally, substituting $Z_n = \widetilde{M^T\mathbf{Y}_n}$ gives

$$X_{n+1} = X_n + \mu(Y_n - Z_n) - \mu\lambda\nabla\Lambda(X_n). \qquad (16)$$

Note that Y_n can be easily computed as $T(X) - X$. The only difficulty of computation is now $Z_n = \widetilde{M^T\mathbf{Y}_n}$. The following lemma shows that we can compute $\widetilde{M^T\mathbf{X}}$, for any given $\mathbf{X}$ based on $X = \widetilde{\mathbf{X}}$ without dealing with large matrices.

Lemma 1 [6]. *Assume T is the fractal image transform of some image with fractal code $\{C_i, P_{J(i)}, \alpha_i, \beta_i\}$, i.e.,*

$$\Big(T(X)\Big)\Big|_{C_i} = \alpha_i D\Big(X\Big|_{P_{J(i)}}\Big) + \beta_i, \qquad (17)$$

for the case of non-overlapping square domain and range blocks, with domain blocks size twice of the range block size. Also take M as defined in Equation 6, and operators U and D are respectively upsampling and downsampling operators, then for any $X = \widetilde{\mathbf{X}}$

$$\Big(\widetilde{M\mathbf{X}}\Big)\Big|_{C_i} = \alpha_i D\Big(X\Big|_{P_{J(i)}}\Big) \qquad (18)$$

and

$$\begin{aligned}\Big(\widetilde{M^T\mathbf{X}}\Big)\Big|_{P_j} &= \sum_{i\in\Omega_j}\frac{\alpha_i}{4}U\Big(X\Big|_{C_i}\Big) \qquad (19)\\ \textit{where}\quad \Omega_j &= \{i \mid J(i) = j\}.\end{aligned}$$

Equation 19 states that to compute $\widetilde{M^T\mathbf{X}}$ on each domain block piece P_j, we take all the range blocks C_i corresponding to this fixed domain block, up-sample each block of these range blocks C_i's and multiply it by its corresponding α_i value divided by 4, and then take the sum over all of these up-sampled and appropriately scaled pieces. This sum gives the required domain block piece P_j of $\widetilde{M^T\mathbf{X}}$, i.e., $\Big(\widetilde{M^T\mathbf{X}}\Big)\Big|_{P_j}$. Note that the non-overlapping constraint on domain blocks is required for $\widetilde{M^T\mathbf{X}}$ to be uniquely determined for this method to be well-defined.

Now we are prepared to summarize our minimization algorithm. We had $Z_n = \widetilde{M^T\mathbf{Y}_n}$, hence, we compute Z_n using Equation 19 of Lemma 1, and yield the following.

Regularized Fractal Decoding Algorithm (RFDA) *The sequence $\{X_n\}$, defined iteratively as following converges to an approximation of $\bar{X}_{reg(\lambda)} = \widetilde{\bar{\mathbf{X}}_{reg(\lambda)}}$ for any X_0 (typically a black image) and sufficiently small μ.*

$$\begin{aligned}Y_n &= T(X_n) - X_n \qquad (20)\\ Z_n\Big|_{P_j} &= \sum_{i\in\Omega_j}\frac{\alpha_i}{4}U\Big(Y_n\Big|_{C_i}\Big), \quad \forall P_j \in \mathcal{P}\\ \textit{where}\quad \Omega_j &= \{i \mid J(i) = j\},\\ X_{n+1} &= X_n + \mu(Y_n - Z_n) - \mu\lambda\nabla\Lambda(X_n).\end{aligned}$$

Note that in the case that $\lambda = 0$ this scheme is in fact a special case of Landweber algorithm, and $\{X_n\}$ converges to $\widetilde{\bar{\bar{\mathbf{X}}}}$ the least square minimizer of Equation 10 (thought in the matrix form so we need tildes) that is closest to X_0. It is of specific interest when we cover the missing fractal code problem in Section 6. In this problem 10 is underdetermined with many possible least square solutions. We state that by iterating the RFDA from $X_0 = 0$ with no regularization, i.e., $\lambda = 0$, X_n converges to the *minimum norm* least square solution of 10, and not just some least square solution.

5 Various Regularization Functions

The choice of regularization function may depend on the application. In this section, we briefly mention three different popular regularization functions and explicitly form their RFDA counterparts.

- **Tikhonov Regularization**

The special case, $\mathbf{\Lambda}(\mathbf{X}) = \frac{1}{2}\| \mathbf{X} \|^2$ is called the Tikhonov regularization. It can be shown that the analytic solution in this case is

$$\bar{\mathbf{X}}_{reg(\lambda)} = (A^T A + \lambda I_{N\times N})^{-1} A^T B. \qquad (21)$$

The RFDA counterpart below can be used to estimate $X_{reg(\lambda)}$,

$$\begin{aligned} Y_n &= T(X_n) - X_n \qquad (22)\\ Z_n\Big|_{P_j} &= \sum_{i\in\Omega_j} \frac{\alpha_i}{4} U\Big(Y_n\Big|_{C_i}\Big), \forall P_j \in \mathcal{P}\\ X_{n+1} &= (1-\mu\lambda)X_n + \mu(Y_n - Z_n). \end{aligned}$$

- **Total Variation (TV)**

A possibility is to choose Total Variation(TV) for the regularization functional. In this case,

$$\begin{aligned} \Lambda(X) = TV(X) &= \int \sqrt{|\nabla X|^2} dxdy \qquad (23)\\ \text{and } \nabla TV(X) &= -\nabla \cdot (\frac{\nabla X}{|\nabla X|}). \qquad (24) \end{aligned}$$

Hence, the RFDA corresponding to this regularization functional becomes

$$\begin{aligned} Y_n &= T(X_n) - X_n \qquad (25)\\ Z_n\Big|_{P_j} &= \sum_{i\in\Omega_j} \frac{\alpha_i}{4} U\Big(Y_n\Big|_{C_i}\Big), \forall P_j \in \mathcal{P}\\ X_{n+1} &= X_n + \mu(Y_n - Z_n) + \mu\lambda\nabla \cdot (\frac{\nabla X_n}{|\nabla X_n|}). \end{aligned}$$

- **Hard Constrained**

In some certain cases we may have the prior knowledge that the encoded image lives in some closed and convex set Ψ (typically a subset of the Hilbert space $L^2(R^2)$). For example Ψ can be constructed based on certain smoothness or energy boundedness constraint of the image. Assume that Ψ is given and the projector on $Proj_\Psi$ is easy to compute. Then, the projected Landeweber algorithm can be applied to estimate a minimizer of $G_\lambda(\mathbf{X})$ in Equation 12 that lives in Ψ (if it exists). In this case one can assume that

$$\mathbf{\Lambda}(\mathbf{X}) = \iota_\Psi(\mathbf{X}) \times \infty, \qquad (26)$$

where ι_Ψ is the characteristic function on Ψ, and $0\times\infty = 0$. Similar to use of SD now applying projected Landeweber algorithm in the RFDA, we get the following algorithm to estimate the solution of hard constrained case.

$$\begin{aligned} Y_n &= T(X_n) - X_n \qquad (27)\\ Z_n\Big|_{P_j} &= \sum_{i\in\Omega_j} \frac{\alpha_i}{4} U\Big(Y_n\Big|_{C_i}\Big), \forall P_j \in \mathcal{P}\\ X_{n+1} &= Proj_\Psi\Big(X_n + \mu(Y_n - Z_n)\Big). \end{aligned}$$

6 Application to Restoration of an Image With an Incomplete Fractal Code

In [5] we introduced the problem of restoration of an image with an incomplete fractal code, and used the method of projection onto convex sets (POCS) to address this problem. Here we use the regularization approach introduced above.

Assume that we are given an incomplete fractal code of an image, i.e., some of the domain-range block assignments and corresponding greyscale map coefficients are missing. In such a case, the usual fractal decoding scheme, in which an arbitrary "seed" image is employed, will collapse since the range blocks of the image for which the fractal code is missing cannot be modified. These blocks will simply remain identical to the corresponding subblocks of the seed image.

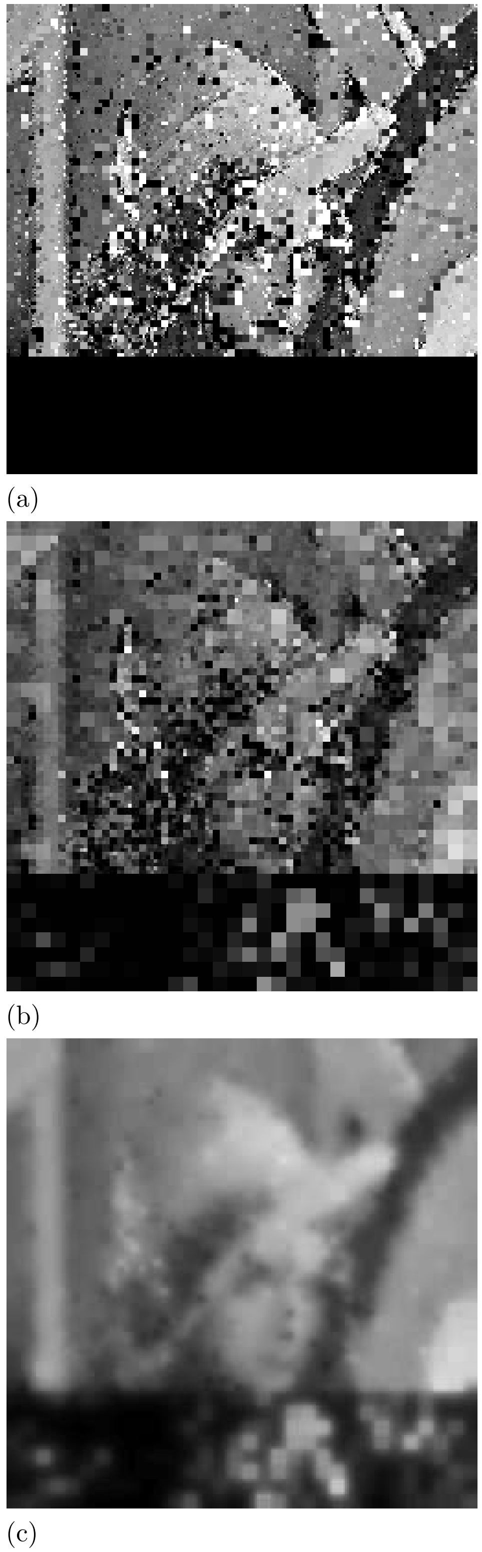
To illustrate, let us consider the fractal code associated with the well-known *Lena* image (256 × 256 pixels, 8 bits per pixel). Suppose that the fractal code corresponding to all range blocks in the bottom quarter of the image are missing. Using a black image as the seed, the limiting image produced by the fractal decoding procedure is shown in Figure 2(a). Only the domain blocks in the top-three-quarter of the image have been modified – those in the bottom quarter are identical to their counterparts in the seed black image.

We now show how this situation can be improved using the theory presented in this paper. We consider different types of regularization functions discussed in this paper, and apply the RFDA with a fixed small step size $\mu = 0.01$ in all cases after mapping the image values to $[0, 1]$. For the missing codes, i.e., for the range blocks C_i's where the corresponding domain $P_{J(i)}$ and the coefficients α_i and β_i are missing, we initially replace $\alpha_i = 0$ and $\beta_i = 0$. Now we use the RFDA to estimate the solution of this underdetermined problem.

First we set $\lambda = 0$ and use RFDA iterating from $X_0 = 0$. This scheme will find the minimum norm least square solution of the problem. The result is given in Figure 2(b). It can be seen that the bottom quarter of the image which was not changed at all by the traditional fractal decoding in Figure 2(a) has been improved. The RFDA updates Z_n by

$$Z_n\Big|_{P_j} = \sum_{i\in\Omega_j} \frac{\alpha_i}{4} U\Big(Y_n\Big|_{C_i}\Big), \quad \forall P_j \in \mathcal{P}. \qquad (28)$$

This actually causes an alteration of *domain blocks* P_j that are included in the missing range part, i.e., the bottom



(a)

(b)

(c)

Figure 2. (a) Decoded as bottom quarter of the fractal code is missing, black seed. (b) Proposed Minimum norm least square solution using RFDA with $\lambda = 0, \mu = 0.01$. (c) Proposed Hard-constrained RFDA with smoothness constraint, $\mu = 0.01$.

quarter of the image. Since some domain blocks used in the fractal coding procedure come from that region, it is possible that portions of the lower part of the image are modified.

In Figure 2(c), we show the result obtained using the RFDA in hard-constrained case, as explained in Section 5 Equation (27). The initial image was black, i.e., $X_0 = 0$. Ψ can be thought as the set of images with certain smoothness, e.g, low-pass filtered in the frequency domain. In this case, $Proj_{\Psi}(X)$ the projection operator on Ψ smoothes the input image X. Clearly, Figure 2(c) is an improvement over the usual fractal coding method of Figure 2(a), although some edges are smeared out as a result of the projection $Proj_{\Psi}$.

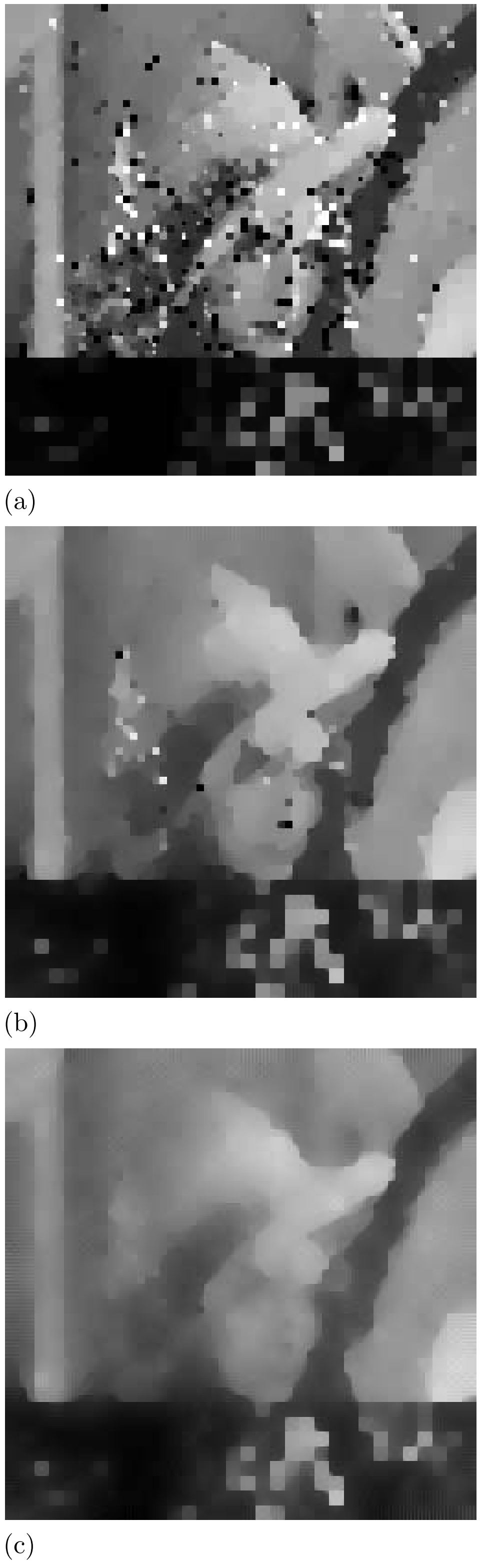
Finally, in Figures 3(a,b,c) are shown the results obtained by applying RFDA with total variation regularization as in 25, with $\lambda = 0.1$, 0.5 and 1, respectively. Needless to say, more parameters are involved when implementing gradient computation in the TV model. However, it is not the main purpose of this paper to discuss these matters in detail. This will be done elsewhere. It can be seen that the important edges are preserved using the TV model. It is a noteworthy improvement over Figure 2(a) to preserve the edges as well as to recover some missing information from the bottom quarter of the image.

7 Conclusions

In this paper, we have described a reformulation of traditional fractal image decoding by regularization. The solution of the traditional fractal-based methods are too restrictive in the sense that no additional knowledge about the image can be combined with self-similarity constraints to improve the decoded image. This is based on the fact that Banach's contraction mapping theorem has been used as a prescription for fractal-based decoding procedures. The principal advantage of our framework is that it provides the flexibility to incorporate constraints and possibly additional knowledge about the reconstructed image.

We have improved this situation by viewing the fractal transform operator as a linear operator, and modelling the decoding procedure as a least-squares problem. We have also improved the model by regularization. In addition, we have applied the steepest descent algorithm to numerically estimate the solution. This procedure produced the regularized fractal decoding algorithm (RFDA).

We also introduced an explicit formulation to combine fractal image decoding with a variety of regularization procedures, e.g., Tikhonov, total variation (TV), or hard-constrained. A regularized decoding approach also provides the opportunity of solving under-determined inverse problems in fractal coding, as we have shown for the case of the incomplete fractal code problem. Since the RFDA allows fractal coding to be employed along with any additional knowledge about the image/signal, it still remains to determine what type of regularizations could be used with fractal-based coding to produce the most appealing results.



(a)

(b)

(c)

Figure 3. (a) Proposed RFDA using using TV regularization with $\lambda = 0.1, \mu = 0.01$. (b) Proposed RFDA using using TV regularization with $\lambda = 0.5, \mu = 0.01$. (c) Proposed RFDA using using TV regularization with $\lambda = 1, \mu = 0.01$.

References

[1] M.F. Barnsley, *Fractals Everywhere.* New York: Academic Press, 1988.

[2] M.F. Barnsley, V. Ervin, D. Hardin and J. Lancaster, *"Solution of an inverse problem for fractals and other sets," Proc.of Nat. Acad. of Sci. USA* vol. 83, no. 7, pp. 1975-1977, April 1985.

[3] M.F. Barnsley and L.P. Hurd, *Fractal Image Compression.* Massachusetts: A.K. Peters, Wellesley, 1993.

[4] P. L. Combettes and V. A. Wajs, *"Signal recovery by proximal forward-backward splitting," SIAM Journal on Multiscale Modeling and Simulation* vol. 4, no. 4, pp. 1168-1200, November 2005.

[5] M. Ebrahimi and E.R. Vrscay, *"Generalized fractal image coding using projections onto convex sets," Preprint.*

[6] M. Ebrahimi and E.R. Vrscay, *"Regularization schemes for fractal-based methods," Preprint.*

[7] Y. Fisher, *Fractal Image Compression, Theory and Application.* New York: Springer-Verlag, 1995.

[8] B. Forte and E.R. Vrscay, *"Theory of generalized fractal transforms," in Fractal Image Encoding and Analysis, Y. Fisher, Ed.* New York: Springer-Verlag, 1998.

[9] M. Ghazel, G. Freeman and E.R. Vrscay, *"Fractal image denoising," IEEE Transactions on Image Processing* vol. 12, no. 12, pp. 1560-1578, December 2003.

[10] R. Hamzaoui, *Encoding and Decoding Complexity Reduction and VQ Aspects of Fractal Image Compression.* Germany: Taschenbuch Shaker Verlag, 1998.

[11] N. Lu, *Fractal Imaging.* New York: Academic Press, 1997.

[12] L. Rudin, S. Osher, and C. Fatemi, *"Nonlinear total variation based noise removal algorithms," Physica D* vol. 60, no. 1-4, pp. 259-268, November 1992.

[13] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems.* New York: Wiley, 1977.