Can One Break the "Collage Barrier" in Fractal Image Coding?

E.R. Vrscay[1] and D. Saupe[2]

[1] Department of Applied Mathematics, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1, ervrscay@links.uwaterloo.ca
[2] Institut für Informatik, Universität Leipzig, Augustusplatz 10/11, 04109 Leipzig, Germany, saupe@acm.org

Abstract. Most fractal image coding methods rely upon *collage coding*, that is, finding a fractal transform operator T_c that sends a target image u as close as possible to itself. The fixed point attractor $\bar{u}_c$ of T_c is generally a good approximation to u. However, it is well known that collage coding does not necessarily yield an optimal attractor, i.e., one for which the approximation error $||u - \bar{u}_c||$ is minimized with respect to variations in the fractal transform parameters. A number of studies have employed the "collage attractor" $\bar{u}_c$ as a starting point from which to obtain better approximations to u. In this paper, we show that attractors $\bar{u}$ are differentiable functions of the (affine) fractal parameters. This allows us to use gradient descent methods that search for optimal attractors in fractal parameter space, i.e., local minima of the approximation error $||u - \bar{u}||$. We report on results of corresponding computer experiments and compare them with those obtained by related (nondifferentiable) methods based on the simplex hill climbing and annealing approaches.

1 Introduction

Fractal image coding seeks to approximate a target image $u(x)$ with the fixed point attractor function $\bar{u}(x)$ of a contractive *fractal transform* operator T that acts on a suitable metric space $(\mathcal{F}, d_{\mathcal{F}})$ of image functions. The *fractal code,* i.e., the parameters that define T, is then used to represent the target image u. The approximation $\bar{u}$ is generated by iteration of T (*decoding*).

Given a suitable parameter space $\mathcal{P}$ of acceptable fractal codes it is generally a tedious procedure to determine the best fractal code, i.e., to determine the operator T_{opt} whose fixed point $\bar{u}_{opt}$ yields the smallest possible error $d_{\mathcal{F}}(\bar{u}, u)$. In fact, Ruhl and Hartenstein [15] have shown that such "attractor coding" problems are NP-hard. Understandably, most, if not all, fractal coding algorithms rely upon the *Collage Theorem* [2,1]. In *collage coding*, one finds an operator T_c that minimizes the *collage error* $d_{\mathcal{F}}(u, Tu)$. As is well known, the corresponding attractor error $d_{\mathcal{F}}(\bar{u}_c, u)$, where $\bar{u}_c = T\bar{u}_c$, is bounded above by a multiple of the collage error. Unfortunately, there is little theoretical knowledge about the relationship between the collage error and the minimum error.

It is also well known that collage coding is not necessarily optimal, i.e., that $\bar{u}_c \neq \bar{u}$. In fact, Ruhl and Hartenstein [15] have rigorously shown that collage coding is *not* a ρ-approximating algorithm: Using the notation introduced above, there exists no (finite) constant $\rho > 0$ such that

$$\frac{d_{\mathcal{F}}(u, \bar{u}_c)}{d_{\mathcal{F}}(u, \bar{u}_{opt})} < \rho \qquad \forall u \in \mathcal{F}.$$

In other words, it is possible that the ratio of the collage attractor error to the optimal attractor error can be arbitrarily large. Intuitively, this is a consequence of the fact that collage coding is a "greedy algorithm" that seeks to find self-similarity of an image in one scan. Nevertheless, because of its relative simplicity and the fact that it appears to work well in most cases, collage coding continues to serve as the basis of most fractal coders.

Several studies have attempted to find attractor functions $\bar{u}$ that are better approximations to a target u than the "collage attractors" $\bar{u}_c$. Indeed, these studies have typically employed the collage attractor $\bar{u}_c$ as a starting point. For example, Barthel [3] and then Lu [10] have devised "annealing schemes" that produce sequences of attractors $\bar{u}_n$ in which the fractal code for $\bar{u}_{n+1}$ is obtained from constructing a collage of pieces of $\bar{u}_n$ to approximate the target u. The sequences $\bar{u}_n$ are observed to provide better approximations to the target. However, there is still no rigorous theoretical basis for this method. A closely related annealing algorithm is presented by Domaszevicz and Vaishampayan [5]. On the other hand, Dudbridge and Fisher [6], using the Nelder-Mead simplex algorithm, searched the fractal code space $\mathcal{P}$ in the vicinity of the collage attractor to locate (local) minima of the approximation error $d_{\mathcal{F}}(u, \bar{u})$. Their method was applied to a rather restricted class of (separable) fractal transforms, in which four 4×4 pixel range blocks shared a common domain block.

In this paper we examine a systematic method to perform attractor optimization using the partial derivatives of attractor functions with respect to the fractal code parameters, $\partial\bar{u}/\partial\pi_k$, $k = 1, 2, \ldots, M$. Here, the parameter space $\mathcal{P}$ of acceptable fractal codes consists of vectors $\underline{\pi} = (\pi_1, \pi_2, \ldots, \pi_M)$, $M > 0$. We first establish the existence of these derivatives and show that they are attractor functions of "vector fractal transform" operators. A knowledge of these derivatives permits the computation of the gradient vector of the error function $d_{\mathcal{F}}(\bar{u}, u)$ which, in turn, allows the use of gradient descent algorithms. In principle, the method may be applied to any domain-range fractal coding scheme. However, from a complexity point of view, it is expensive, since the attractor and all derivatives must be computed at every pixel. We compare these results to those obtained by minimizing the approximation error using the Nelder-Mead simplex algorithm employed in Ref. [6] and also to those obtained using the appropriate variant of annealing schemes from Ref. [3,10].

In a related work Withers [16] derives differentiability properties of Iterated Function Systems with probabilities whose attractors model graphs of 1D functions. Newton's method is used to compute parameters.

Finally, in our discussion of the inverse problem, we present an "Anti-Collage Theorem" which, in contrast to the infamous Collage Theorem, provides a *lower*

bound to the approximation error in terms of the collage error. This result is another very simple consequence of Banach's Fixed Point Theorem.

2 Partial Derivatives of IFSM Attractor Functions

2.1 Mathematical Preliminaries

Let (X, d) denote the *base* or *pixel space*, assumed to be a complete metric space. The discussion will be carried out for the continuous case, e.g., $X = [0,1]^2$, but with the understanding that it readily carries over to the discrete case, e.g., $X = \{(i,j) | 1 \leq i,j \leq n_p\}$, where n_p denote the width and height of a square image in pixels. Let $\mathcal{F}(X) = \{f : X \to \mathbf{R}\}$ denote a suitable complete space of image functions with metric $d_{\mathcal{F}}$. Later, we shall use $\mathcal{F}(X) = \mathcal{L}^2(X)$, the space of square integrable functions on X with the usual metric.

Now let $R_k \subset X$, $k = 1, 2, \ldots, N$ denote a set of *range* or *child* blocks that partition X, i.e., (1) $\cup_{k=1}^{N} R_k = X$ and (2) $R_i \cap R_j = \emptyset$ for $i \neq j$. Assume that for each range block R_k are associated the following:

1. a *domain* or *parent* block, $D_k \subset X$, such that $R_k = w_k(D_k)$, where w_k is a one-to-one contraction map with contraction factor $c_k \in [0,1)$,
2. an affine greyscale map $\phi_k(t) = \alpha_k t + \beta_k$, where $\alpha_k, \beta_k \in \mathbf{R}$.

In the language of [9], the above ingredients comprise an (affine) N-map Iterated Function System with Grey Level Maps (IFSM). Associated with such a (nonoverlapping) IFSM is a fractal transform operator $T : \mathcal{F}(X) \to \mathcal{F}(X)$ whose action is defined as follows. Given an image $u \in \mathcal{F}(X)$ then for all $x \in R_k$, $k = 1, 2, \ldots, N$,

$$\begin{aligned} v(x) \;=\; (Tu)(x) &= \phi_k(u(w_k^{-1}(x)) \\ &= \alpha_k u(w_k^{-1}(x)) + \beta_k. \end{aligned} \tag{1}$$

The transform T is defined by the fractal code parameters

$$\underline{\pi} = (\alpha_1, \ldots, \alpha_N, \beta_1, \ldots, \beta_N).$$

In addition some side information must be transmitted to the decoder in order to specify the image partition and the range-domain assignments. However, in this study, we assume a *fixed* domain-range block configuration, implying that the IFS maps w_i are *fixed*. Thus, for all further discussion we can disregard this side information.

It is well known that if $|\alpha_k| < 1$, $1 \leq k \leq N$, then the operator T is contractive in the space of functions $\mathcal{L}^\infty(X)$. In the space of functions $\mathcal{L}^2(X)$, a straightforward calculation shows that

$$\|Tu - Tv\|_2 \leq C \|u - v\|_2, \qquad \forall u, v \in \mathcal{L}^2(X),$$

where

$$C = \sum_{k=1}^{N} c_k |\alpha_k|. \tag{2}$$

Therefore, the condition $C < 1$ is sufficient (but not necessary) for contractivity of T in $\mathcal{L}^2(X)$.

Another special type of IFSM/fractal transform, relevant below, involves "condensation" [11]. For a $u \in \mathcal{F}(X)$, define $v = Tu$ as follows: For all $x \in R_k$, $k = 1, 2, \ldots, N$,

$$v(x) \;=\; (Tu)(x) \;=\; \alpha_k u(w_k^{-1}(x)) + \theta_k(x). \tag{3}$$

The functions $\theta_k(x)$ are known as *condensation* functions. Note that condensation functions do *not* affect the contractivity of T.

2.2 Partial Derivatives of IFSM Attractor Functions with Respect to Greyscale Map Parameters

Let us now consider an affine N-map IFSM and assume that its associated fractal transform T in Eq. (1) is contractive in the space $\mathcal{F}(X) = \mathcal{L}^2(X)$. Therefore, there exists a unique fixed point attractor function $\bar{u} = T\bar{u}$. We now consider $\bar{u}$ as a function not only of position but also the fractal parameters, i.e., $\bar{u} = \bar{u}(x, \underline{\pi})$. Then, from Eq. (1),

$$\bar{u}(x, \underline{\pi}) \;=\; \alpha_k \bar{u}(w_k^{-1}(x), \underline{\pi}) + \beta_k, \quad x \in R_k. \tag{4}$$

Proposition 1. *The attractor $\bar{u}$ is continuous with respect to the fractal parameters α_ℓ, β_ℓ, $l = 1, 2, \ldots, N$.*

The continuity of IFSM attractors with respect to grey level maps ϕ_ℓ was proved in [8], using the methods described in [4]. It is straightforward to establish the continuity in terms of the grey level parameters α_ℓ and β_ℓ. The following result, which establishes the continuity of attractor functions with respect to condensation functions, is also a simple consequence of Proposition 1.

Proposition 2. *Let T_1 and T_2 be contractive N-map IFSM operators as in Eq. (3), with condensation functions $\theta^{(1)}(x)$ and $\theta^{(2)}(x)$, respectively, and identical fractal coefficients α_k. Let $\bar{u}_1$ and $\bar{u}_2$, respectively, denote the fixed points of these operators. Then given an $\epsilon > 0$, there exists a $\delta > 0$ such that $\|\theta^{(1)} - \theta^{(2)}\|_2 < \delta$ implies that $\|\bar{u}_1 - \bar{u}_2\|_2 < \epsilon$.*

Recall that the IFS maps w_i employed in the fractal transforms are assumed to be fixed. Now define the *feasible fractal parameter space* $\mathcal{P} \subset \mathbf{R}^{2N}$ to be the set of all fractal codes $\underline{\pi} \in \mathbf{R}^{2N}$ for which the corresponding affine IFSM operators T defined in Eq. (1) are contractive in $\mathcal{F}(X) = \mathcal{L}^2(X)$.

Proposition 3. *The set $\mathcal{P}$ is open.*

Proof: We prove that $\overline{\mathcal{P}} = \mathbf{R}^{2N} - \mathcal{P}$ is closed. Let $\underline{\pi}_n \in \overline{\mathcal{P}}$, $n = 1, 2, \ldots$, be a convergent sequence (in the topology of $\mathbf{R}^{2N}$) with limit $\underline{\pi}$. Each (unfeasible) fractal code vector $\underline{\pi}_n \in \overline{\mathcal{P}}$ defines a noncontractive fractal transform operator

$T_n : \mathcal{L}^2(X) \to \mathcal{L}^2(X)$ with associated factor (cf. Eq. (2)) $C_n = \sum_{k=1}^{N} c_k^{1/2} |\alpha_{nk}|$. Now, for each operator T_n, define its "optimal" Lipschitz factor as follows,

$$L_n = \sup_{y_1 \neq y_2} \frac{\|T_n y_1 - T_n y_2\|_2}{\|y_1 - y_2\|_2}.$$

From this definition and the noncontractivity of the T_n, it follows that $1 \leq L_n \leq C_n$ for all n. From the convergence of the code vectors $\underline{\pi}_n$, it also follows that $\lim_{n\to\infty} C_n = C \geq 1$. Therefore, from Proposition 1, the fractal transform T defined by the limit code vector $\underline{\pi}$ has associated factor C and Lipschitz factor $L \geq 1$. Therefore T is not contractive, implying that $\underline{\pi} \notin \mathcal{P}$. Thus $\overline{\mathcal{P}}$ is closed, proving the proposition. □

Proposition 4. *The partial derivatives of the attractor $\bar{u}$ with respect to the fractal parameters α_ℓ, β_ℓ, $l = 1, 2, \ldots, N$, exist at any point $\underline{\pi} \in \mathcal{P}$.*

Proof: For any $\underline{\pi} \in \mathcal{P}$, the associated fractal transform T is contractive. This implies that for any $u_0 \in \mathcal{L}^2$, the sequence of functions defined by $u_{n+1} = Tu_n$ converges to $\bar{u}$, that is, $\|u_n - \bar{u}\|_2 \to 0$ as $n \to \infty$. Let $u_0 = \theta$, where

$$\theta(x) = \sum_{k=1}^{N} \beta_k I_{R_k}(x)$$

and $I_S(x)$ is the characteristic function of a subset $S \subset X$. Then, for $M \geq 0$, $u_M = T^{\circ M} u_0$ is given by

$$u_M(x, \underline{\pi}) = \theta(x) + \sum_{n=1}^{M} \sum_{i_1,\ldots,i_n}^{N} \alpha_{i_1} \cdots \alpha_{i_n} \theta(w_{i_n}^{-1} \circ \cdots \circ w_{i_1}^{-1}(x)). \tag{5}$$

The u_M are partial sums of an infinite series that converge, in the $\mathcal{L}^2$ metric, to $\bar{u}$. Thus, we can write

$$\bar{u}(x, \underline{\pi}) = \theta(x) + \sum_{n=1}^{\infty} \sum_{i_1,\ldots,i_n}^{N} \alpha_{i_1} \cdots \alpha_{i_n} \theta(w_{i_n}^{-1} \circ \cdots \circ w_{i_1}^{-1}(x)),$$

where the equation is understood in the $\mathcal{L}^2$ sense.

Now consider an $x \in R_k$ for some $k \in \{1, 2, \ldots, N\}$. Then the index i_1 in Eq. (5) must equal k (in order for $w_{i_1}^{-1}(x)$ to be defined). Therefore, Eq. (5) becomes

$$u_M(x, \underline{\pi}) = \theta(x) + \alpha_k u_{M-1}(w_k^{-1}(x), \underline{\pi}), \qquad x \in R_k.$$

For a given $l \in \{1, 2, \ldots, N\}$, we partially differentiate the terms in this equation with respect to α_ℓ:

$$\frac{\partial u_M}{\partial \alpha_\ell}(x, \underline{\pi}) = \alpha_k \left[\frac{\partial u_{M-1}}{\partial \alpha_\ell}(w_k^{-1}(x), \underline{\pi}) \right] + [u_{M-1}(w_k^{-1}(x), \underline{\pi})] \delta_{kl}, \tag{6}$$

where $\delta_{kl} = 1$ if $k = l$ and zero otherwise. Define the following N-map IFSM operator T_ℓ with condensation:

$$(T_\ell u)(x) \;=\; \alpha_k u(w_k^{-1}(x)) + \xi_k(x), \qquad x \in R_k, \quad 1 \le k \le N,$$

where $\xi_k(x) = [\bar{u}(w_k^{-1}(x))]\delta_{kl}$. Since T is contractive, it follows that T_ℓ is contractive in $\mathcal{L}^2$. (T and T_ℓ have identical IFS maps and fractal parameters α_k.) Let $\bar{v}_\ell$ denote the fixed point of T_ℓ. From Propositions 1 and 2, $\bar{v}_\ell$ is continuous with respect to the parameters α_k, in particular, α_ℓ. We now show that $\bar{v}_\ell = \partial\bar{u}/\partial\alpha_\ell$. (In what follows, for simplicity of notation, only x and α_ℓ will be written explicitly in the list independent variables.)

Note that Eq. (6) does not correspond to a single IFSM operator with condensation. However, since the functions u_M converge to $\bar{u}$, it follows, from Proposition 2, that the sequence of functions $\partial u_M/\partial\alpha_\ell$ converges to $\bar{v}_\ell$. That is, for a given $\underline{\pi} \in \mathcal{P}$ and $\epsilon_1 > 0$, there exists an $M_1 > 0$ such that

$$\left\| \frac{\partial u_M}{\partial \alpha_\ell}(x, \alpha_\ell) - \bar{v}(x, \alpha_\ell) \right\|_2 < \epsilon_1, \qquad \forall M > M_1. \tag{7}$$

We denote our reference point as $\underline{\pi}^0 = (\alpha_1^0, \ldots, \alpha_N^0, \beta_1^0, \ldots, \beta_n^0) \in \mathcal{P}$. Let $N_\ell(\delta)$, $\delta > 0$, be a restricted neighbourhood of the point $\underline{\pi}^0$ in which only the component α_ℓ is allowed to vary, i.e., $\alpha_\ell \in I_\delta = [\alpha_\ell^0 - \delta, \alpha_\ell^0 + \delta]$, such that the corresponding vectors $\underline{\pi}$ lie in $\mathcal{P}$. (The existence of such a neighbourhood is guaranteed since $\mathcal{P}$ is open.) Let $h \in \mathbf{R}$, with $|h| < \delta$. Then for each $x \in X$ there exists, by the Mean Value Theorem, a $c_M \in I_h = [\alpha_\ell^0 - h, \alpha_\ell^0 + h]$, such that

$$u_M(x, \alpha_\ell^0 + h) - u_M(x, \alpha_\ell^0) = \frac{\partial u_M}{\partial \alpha_\ell}(x, c_M)h.$$

Therefore,

$$\begin{aligned}
&\|u_M(x, \alpha_\ell^0 + h) - u_M(x, \alpha_\ell^0) - h\bar{v}(x, \alpha_\ell^0)\|_2 \\
&\qquad = h \left\| \frac{\partial u_M}{\partial \alpha_\ell}(x, c_M) - \bar{v}(x, \alpha_\ell^0) \right\|_2 \\
&\qquad \le h \left\| \frac{\partial u_M}{\partial \alpha_\ell}(x, c_M) - \bar{v}(x, c_M) \right\|_2 \;+\; h\|\bar{v}(x, c_M) - \bar{v}(x, \alpha_\ell^0)\|_2 \\
&\qquad \le h \left\| \frac{\partial u_M}{\partial \alpha_\ell}(x, c_M) - \bar{v}(x, c_M) \right\|_2 \\
&\qquad\qquad + \max_{\alpha_\ell \in I_h} h\|\bar{v}(x, \alpha_\ell) - \bar{v}(x, \alpha_\ell^0)\|_2.
\end{aligned} \tag{8}$$

Since I_δ is closed, there exists an $\overline{M} > 0$ such that the inequality in (7) is satisfied for all $M > \overline{M}$ at all $\underline{\pi} \in N_\ell(\delta)$. Therefore, for a fixed $h \in (-\delta, \delta)$, we may take the limit $M \to \infty$ of both sides of (8) to give

$$\left\| \frac{\bar{u}(x, \alpha_\ell^0 + h) - \bar{u}(x, \alpha_\ell^0)}{h} - \bar{v}(x, \alpha_\ell^0) \right\|_2 \;\le\; \max_{\alpha_\ell \in I_h} \|\bar{v}(x, \alpha_\ell) - \bar{v}(x, \alpha_\ell^0)\|_2.$$

Since $\bar{v}$ is continuous with respect to α_ℓ, the right side term may be made arbitrarily small by making h sufficiently small, thus establishing the differentiability of $\bar{u}$ with respect to α_ℓ at $\underline{\pi}^0$.

The differentiability of $\bar{u}$ with respect to the β_ℓ may be derived in a similar fashion. □

Remark: From Eq. (6) (and its analogue for differentiation with respect to β_ℓ), the partial derivatives of $\bar{u}$ with respect to the fractal parameters α_ℓ and β_ℓ may be obtained by formally differentiating both sides of Eq. (4). For a fixed $x \in R_k$:

$$\frac{\partial \bar{u}}{\partial \alpha_\ell}(x, \underline{\pi}) = \alpha_k \left[\frac{\partial \bar{u}}{\partial \alpha_\ell}(w_k^{-1}(x), \underline{\pi}) \right] \; + \; [\bar{u}(w_k^{-1}(x), \underline{\pi})]\delta_{k\ell}, \tag{9}$$

$$\frac{\partial \bar{u}}{\partial \beta_\ell}(x, \underline{\pi}) = \alpha_k \left[\frac{\partial \bar{u}}{\partial \beta_\ell}(w_k^{-1}(x), \underline{\pi}) \right] \; + \; \delta_{k\ell}. \tag{10}$$

Eqs. (4), (9) and (10) may be considered to define a $(2N+1)$-component "vector IFSM with condensation" that may be written in the following compact form:

$$\bar{\mathbf{u}} = \mathbf{T}\bar{\mathbf{u}},$$

where

$$\bar{\mathbf{u}}(\mathbf{x}, \underline{\pi}) = \left[\bar{u}(x, \underline{\pi}), \frac{\partial \bar{u}}{\pi_1}(x, \underline{\pi}), \ldots, \frac{\partial \bar{u}}{\pi_{2N}}(x, \underline{\pi}) \right]^T. \tag{11}$$

Now define the space $\mathcal{F}^{2N+1}(X) = \{\mathbf{f} = (f_0, f_1, \ldots, f_{2N}) \mid f_j \in \mathcal{F}(X)\}$ with associated metric $d_{\mathcal{F}^{2N+1}}(\mathbf{f}, \mathbf{g}) = \max_{0 \le j \le 2N} d_{\mathcal{F}}(f_j, g_j)$. Then the vector IFSM is denoted by $\mathbf{T} : \mathcal{F}^{2N+1}(X) \to \mathcal{F}^{2N+1}(X)$. For an $f \in \mathcal{F}^{2N+1}(X)$,

$$(\mathbf{Tf})(x) \;=\; A_k \cdot \mathbf{f}(w_k^{-1}(x)) \;+\; \Theta_k(x), \quad x \in R_k.$$

The coefficients of the matrix $A_k = (a_{i,j}^{(k)})_{i,j=0,\ldots,2N}$ are

$$a_{i,j}^{(k)} = \begin{cases} \alpha_k \text{ if } i = j \\ 1 \quad \text{if } i = k, j = 0 \\ 0 \quad \text{otherwise} \end{cases}.$$

The entry "1" in the k-th row represents the only "mixing" of components of $\mathbf{u}$ under the action of $\mathbf{T}$. The vector $\Theta_k(x)$ represents a condensation vector composed of constant functions: $[\Theta_k(x)]^T = (\beta_k, 0, 0, \ldots, 1, \ldots, 0)$, where the "1" occurs at index $N + k$.

Proposition 5. *Suppose that T is contractive in $(\mathcal{F}(X), d_{\mathcal{F}})$. Then $\mathbf{T}$ is contractive in $\mathcal{F}^{2N+1}(X)$. Its fixed point $\mathbf{u}$ is given by Eq. (11), where $\bar{u}$ is the fixed point of T, see Eq. (4).*

From Banach's Fixed Point Theorem, contractivity of T allows the computation of its fixed point function $\bar{u}$ by means of iteration. The above proposition

implies that all all partial derivatives $\partial\bar{u}/\partial\pi_\ell$ may also be computed by iteration: Begin with a "seed" $\mathbf{u}_0 \in \mathcal{F}^{2N+1}(X)$ and construct the sequence of vector functions $\mathbf{u}_{n+1} = \mathbf{T}\mathbf{u}_n$, $n \geq 0$. The reader will immediately note the complexity of such calculations: Except in special cases, $\bar{u}$ and its partial derivatives will have to be computed for all $x \in X$. This will be discussed in more detail below.

3 The Inverse Problem of Fractal Approximation

3.1 Optimal Attractor Coding vs. Collage Coding

We now consider inverse problems in $\mathcal{F}(X) = \mathcal{L}^2(X)$. As well, $v \in \mathcal{L}^2(X)$ will denote a "target" image function that we seek to approximate. Let T be a fractal transform, as defined in Eq. (1), with fractal code $\underline{\pi} \in \mathcal{P}$, i.e., T is contractive in $\mathcal{L}^2(X)$. We consider the squared $\mathcal{L}^2$ error in the approximation $v \approx \bar{u}$ as a function of the fractal code $\underline{\pi}$:

$$\begin{aligned} E(\underline{\pi}) &= \int_X [v(x) - \bar{u}(x, \underline{\pi})]^2 dx \\ &= \langle v - \bar{u}, v - \bar{u} \rangle. \end{aligned}$$

The "true" inverse problem, or *attractor coding*, of v is then to find $\min_{\underline{\pi} \in \mathcal{P}} E(\underline{\pi})$. Let $\underline{\pi}_{opt}$ denote a global minimum point of this error function. Assuming that the fractal transform T_{opt} defined by $\underline{\pi}_{opt}$ is contractive, we denote its fixed point by $\bar{u}_{opt}$ and refer to it as the *optimal attractor*.

However, as mentioned earlier, the solution of this problem, involving the determination of optimal domain-range pairs and associated fractal parameters, is generally intractable. As such, most fractal-based methods perform *collage coding*, that is, they seek to minimize the *collage error* associated with the transform $T = T(\underline{\pi})$. We denote the squared $\mathcal{L}^2$ collage error as

$$\begin{aligned} \Delta(\underline{\pi}) &= \int_X [v(x) - T(\underline{\pi})v(x)]^2 dx \\ &= \sum_{k=1}^{N} \int_{R_k} [v(x) - \alpha_k v(w_k^{-1}(x)) - \beta_k]^2 dx. \end{aligned}$$

A standard procedure is to impose stationarity conditions in order to obtain a system of two linear equations for each set of parameters (α_k, β_k), $k = 1, 2, \ldots, N$. (However, it is *not* guaranteed that the solution of these systems lies in $\mathcal{P}$.) If we let $\underline{\pi}_{col}$ denote the global minimum of $\Delta(\underline{\pi})$ then, of course,

$$E(\underline{\pi}_{opt}) \leq E(\underline{\pi}_{col}).$$

We shall refer to the fixed point of the fractal transform defined by $\underline{\pi}_{col}$ (assuming it to be contractive) as the *collage attractor*, $\bar{u}(x, \underline{\pi}_{col})$.

3.2 Optimizing Collage Coding

One possible compromise between solving the optimal attractor coding problem and suboptimal collage coding is to employ the collage attractor (and corresponding domain-range assignments), in particular the fractal code $\underline{\pi}_{col}$, as a starting point, varying the fractal parameters $\underline{\pi}$ in an attempt to decrease the error function $E(\underline{\pi})$ as much as possible. This was the strategy of Dudbridge and Fisher [6], who employed the Nelder-Mead simplex algorithm for some rather simple and restrictive domain-range assignments. In this scheme, the error function $E(\underline{\pi})$ is computed at strategic points.

A knowledge of the partial derivatives of $\bar{u}$ with respect to the fractal parameters permits the computation of elements of the gradient vector of E:

$$\frac{\partial E}{\partial \pi_\ell}(\underline{\pi}) \;=\; -2\left\langle v-\bar{u}, \frac{\partial \bar{u}}{\partial \pi_\ell}(\underline{\pi})\right\rangle, \qquad l=1,2,\ldots,2N.$$

This allows us to employ gradient-descent and related methods to search for local minima.

Practically speaking, however, the partial derivatives $\partial\bar{u}/\partial\pi_\ell(x,\underline{\pi})$ must be computed at all points (pixels) $x \in X$. In addition to an $n_p \times n_p$ matrix required to store an image, an additional $2N$ $n_p \times n_p$ matrices are needed, in general, to store the derivatives at all pixels. Borrowing from the terminology of quantum chemists, this "full configuration interaction" will compute the total rate of change of the attractor — hence the approximation error — with respect to changes in all fractal parameters π_ℓ *for a fixed set of domain-range pair assignments*. When applying a gradient descent method to minimize the error function E less storage is required. It suffices to provide one additional $n_p \times n_p$ matrix to sequentially compute each component of the gradient $(\partial E/\partial\pi_1, \ldots, \partial E/\partial\pi_{2N})$.

In Section 4 we apply the conjugate gradient method to compute minima of the error function E. Technically we need to ensure that the gradient of E is continuous. As a sketch of a proof of this property we remark that the gradient is essentially given by the attractor of a vector IFS with condensation which is continuous with respect to IFS parameters. This should be sufficient to establish continuous partial derivatives of E.

3.3 The "Anti-Collage Theorem"

We close this section by examining an interesting consequence of collage coding. For reference, we state the "Collage Theorem," a simple corollary of Banach's Fixed Point Theorem:

Proposition 6 (Collage Theorem). *Given (Y, d_Y) a complete metric space. Let $T : Y \to Y$ be contractive with contraction factor $c_T \in [0,1)$ and let $\bar{y}$ denote the unique fixed point of T. Then for any $y \in Y$,*

$$d_Y(y,\bar{y}) \leq \frac{1}{1-c_T} d_Y(y,Ty).$$

This result, central to fractal image compression, provides an upper bound to the distance $d_Y(y, \bar{y})$, which may be viewed as the error in approximating y by $\bar{y}$, in terms of the collage error $d_Y(y, Ty)$. As may or may not be well known, it may be derived by means of a simple application of the triangle inequality.

However, it may not be so well known that a slight reshuffling of the triangle inequality leads to another result that "works in the other direction," providing a lower bound. We first state a more generalized result.

Proposition 7. *Given (Y, d_Y) a complete metric space. Let $T : Y \to Y$ be Lipschitz, i.e., there exists an $L_T \geq 0$ such that $d_Y(Ty_1, Ty_2) \leq L_T d_Y(y_1, y_2)$ for all $y_1, y_2 \in Y$. As well, assume that $\bar{y}$ is a fixed point of T. Then for any $y \in Y$,*

$$d_Y(y, \bar{y}) \geq \frac{1}{1 + L_T} d_Y(y, Ty).$$

Proof: From the triangle inequality:

$$\begin{aligned} d_Y(y, Ty) &\leq d_Y(y, \bar{y}) + d_Y(\bar{y}, Ty) \\ &\leq d_Y(y, \bar{y}) + L_T d_Y(\bar{y}, y), \end{aligned}$$

from which the desired result follows.

Remark: Note that in the above result, T is not assumed to be contractive. Hence the fixed point $\bar{y}$ need not be unique.

We may now combine the two results for application to fractal approximation: Given a target function $v \in \mathcal{F}(X)$ consider the approximation of v by the unique fixed point $\bar{u}$ of a contraction map $T : \mathcal{F}(X) \to \mathcal{F}(X)$. Then

$$\frac{1}{1 + c_T} d_{\mathcal{F}}(v, Tv) \leq d_{\mathcal{F}}(v, \bar{u}) \leq \frac{1}{1 - c_T} d_{\mathcal{F}}(v, Tv).$$

In other words, the collage error $d_{\mathcal{F}}(v, Tv)$ bounds the approximation error both from above and below. (In retrospect, this most interesting result is a simple consequence of the "triangle" formed by v, $\bar{v}$ and Tv.) For a fixed collage error (assuming this can be done), the spread between upper and lower bounds decreases with the contractivity factor c_T. On a more negative note, for nonzero collage error, there is no chance that the error can be small "by accident."

4 Practical Examples

We have tested the proposed method in two versions of fractal image coding. In the first one we use the approach suggested by Monro and Dudbridge [12] and explained below. This choice is motivated by the chance to compare our results with those obtained by Dudbridge and Fisher [6], who used a simplex method for attractor optimization. Also in this approach the dimensionality of the optimization problem is small, so it is a good starting point. The second series of tests were carried out using quadtree partitionings derived with the code from [7]. We provide a comparison of the results with those obtained by appropriately adjusted procedures of Barthel [3] and Lu [10].

	Collage attractor dB	Attractor optimization Simplex dB (sec)	Gradient dB (sec)	ΔPSNR dB	ΔPSNR [6] dB
Lena	29.25	29.87 (301)	29.87 (229)	0.62	0.35
Boat	26.66	27.42 (300)	27.42 (299)	0.56	0.41
Mandrill	21.52	22.11 (532)	22.08 (1500)	0.59	0.33
Peppers	29.34	30.02 (277)	29.94 (591)	0.68	0.33

Table 1. Results of (a) collage coding and attractor optimization using (b) simplex and (c) gradient methods, the latter two using collage coding as a starting point. All results are expressed in PSNR (dB). The final two columns list the improvement in PSNR achieved by the simplex method obtained in this study and Ref. [5], respectively.

4.1 Results Part 1

We first apply our method to a simple fractal transform scheme examined by Dudbridge and Fisher [6], designed to minimize the interdependency of range blocks. The following four 512×512 pixel images (8 bpp), used in [6] were also used in this study: *Lena*, *Boat*, *Mandrill* and *Peppers*.[1] Each image was partitioned into 4×4 pixel range blocks, with four range blocks sharing a common 8×8 pixel domain block. Therefore, for each image, the inverse problem separates into 64^2 independent problems, each involving an 8×8 pixel image with four range blocks R_k, hence 8 fractal parameters.

As in [6], for each test image we first used collage coding to determine the fractal code $\underline{\pi}_{col}$ that minimizes the collage error. We then used this code as a starting point for a gradient-descent method. The NAG [13] subroutine E04DKF, which performs a quasi-Newton conjugate gradient minimization, was used. It was also desirable to compare these results with the non-gradient calculations of [6]. However, since some of our collage error results differed from those of [6], we have independently carried out attractor optimization using the Nelder-Mead simplex algorithm. The NAG subroutine E04CCF was used.

In all cases, the simplex and gradient methods yielded almost identical improvements. A comparison with [6] will reveal some nonnegligible differences, not only in the collage errors but also in the improvements afforded by the simplex method. In all cases, the improvements in Table 1 are greater. In both the simplex as well as the gradient algorithms, the results are quite sensitive to the settings of the tolerance/accuracy parameters as well as the maximum number of iterations (*maxiter*) allowed. Generally the best performance was obtained when the tolerance parameters for the simplex and gradient subroutines were set to 10^{-5} and 10^{-6}, respectively. The parameter *maxiter* was set to 2000, which is virtually infinity.

[1] These images may be retrieved by anonymous ftp from the Waterloo Fractal Compression Project site `links.uwaterloo.ca` in the appropriate subdirectories located in `ftp://links.uwaterloo.ca/pub/BragZone/GreySet2/`.

	PCA dB	Simplex dB (sec)	Gradient dB (sec)	ΔPSNR dB
Lena	26.93	29.73 (421)	29.74 (288)	2.81
Boat	25.08	27.30 (452)	27.32 (618)	2.24
Mandrill	20.85	22.00 (663)	21.97 (3333)	1.15
Peppers	25.97	29.76 (420)	29.56 (2888)	1.79

Table 2. Results of (a) piecewise constant approximation (PCA) and attractor optimization using (b) simplex and (c) gradient methods, the latter two using the PCA as a starting point. All results are expressed in PSNR (dB). The final column lists the improvement in PSNR achieved by the better of methods (b) and (c).

In Table 1 are presented the PSNR values associated with collage coding and subsequent simplex and gradient optimized attractor coding, along with the improvements in PSNR. The numbers in brackets represent the CPU time required for each calculation. (We emphasize that these numbers are presented for the purpose of comparision, since the computer codes themselves are quite unoptimized.)

In an attempt to understand how good an initial estimate is provided by collage coding, we have performed simplex and gradient optimization calculations for another set of initial conditions, namely, piecewise constant approximations to the images. In this case, all α_ℓ are initially set to zero and the β_ℓ are simply the mean values of the range block. (Of course, in more general problems than the one studied here, there would remain the problem of assigning a domain block to each range block.) In Table 2, we present the results of these calculations. The first column gives the error associated with the initial piecewise constant approximation. The next two columns list the PSNR values of the optimized attractors obtained from, respectively, the simplex and gradient methods along with the CPU times. The final column gives the PSNR improvement yielded by the better of the two methods.

We observe that the simplex and gradient methods, using such suboptimal initial conditions, i.e., piecewise constant approximations, yield approximations that are almost as good as those found from collage attractors. The worst case is *Peppers*, for which a 0.26 dB difference is found. For the others, the discrepancy is on the order of 0.1 dB.

4.2 Results Part 2

In this subsection we report on the results of the gradient descent algorithm when applied to fractal encodings based on quadtree partitionings. We used the coder of Fisher [7] to produce quadtree partitions and corresponding fractal codes. The resulting scaling and offset parameters were then subject to improvement by the gradient descent method. In this case we have used the conjugate gradient algorithm from [14]. The major computational burden is the computation of the gradients required in each step, which allowed us to do experiments only with

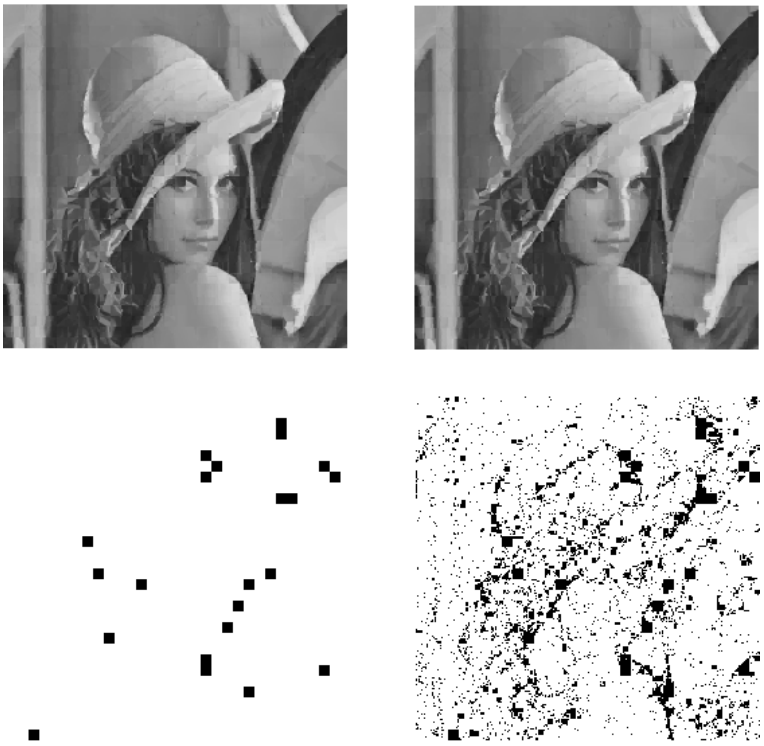

Fig. 1. Illustration of the results for the attractor optimization gradient descent method. The image is the 256x256 test image Lena, compressed using the fractal quadtree coder with minimal range block size of 8x8 pixels.
Top left: attractor image obtained by collage coding (26.96 dB PSNR).
Top right: attractor image after optimization (26.99 dB PSNR).
Bottom left: range blocks with new scale and offset code are shown in black.
Bottom right: pixels that are different in the optimized attractor are shown in black.

images of size 256×256. We considered the test image Lena and three different encodings, obtained by prescribing minimal range block sizes of 16×16, 8×8, and 4×4 pixels respectively.[2] The scaling and offset parameters were then modified until convergence of the conjugate gradient method, which required up to 16 steps for our setting of algorithm parameters. After convergence the obtained scaling and offset parameters were quantized for storage, where we used the same scheme as the one contained in the quadtree coder. Table 3 lists intermediate and end results. The gain obtained by the gradient descent method varied between 0.16 and 0.25 dB in PSNR. However, the necessary quantization

[2] The command line parameters for the quadtree coder were -t 8.000000 -m 4 -d 2 -D 2 -s 5 -o 7 -N 1.000000 -F with -M 4, -M 5, and -M 6 for minimal range block sizes of 16×16, 8×8, and 4×4 pixels respectively.

min. block size	16x16	8x8	4x4
compression ratio	70.2	23.8	9.8
attractor errors (in dB)	23.46	26.96	30.87
step 1	23.57	27.04	30.93
step 2	23.60	27.07	30.96
step 3	23.61	27.10	31.00
step 4	23.61	27.11	31.03
step 5	23.62	27.12	31.06
step 6	23.62	27.12	31.08
step 7	23.62	27.12	31.09
step 8	23.62	27.13	31.10
step 9	23.62	27.13	31.10
step 10	23.62	27.13	31.11
step 11	23.62	27.13	31.11
step 12	—	27.13	31.11
step 16	—	—	31.12
best result (in dB)	23.62	27.13	31.12
gain (in dB)	0.16	0.17	0.25
after quantization (dB)	23.50	26.99	30.89
remaining gain (in dB)	0.04	0.03	0.02
total time	3h:40m	6h:20m	9h:50m

Table 3. Results for the gradient descent method applied to quadtree encodings of the 256×256 test image Lena. The three columns correspond to encodings with different bit rates.

destroyed a large part of these gains so that improvements of only 0.02 to 0.04 dB PSNR remained. This indicates that our settings for the tolerances in the conjugate gradient methods were rather conservative. The computation times were very large.[3] In Figure 1 we display the attractor images and the ranges and pixels that have changed during the attractor optimization.

In order to compare the results with those obtained by previously published methods we implemented an annealing approach along the lines in [3,10]. Briefly, the method proceeds by re-encoding the test image several times. In the first step the original image is used to derive the domain pool as usual, while in all subsequent iterations the attractor from the previous step is used to generate the domain pool. In this way the encoder has a better grasp of the codebook that the decoder actually uses in the attractor image. Unfortunately, there is no theory behind these heuristics and experiments show that the attractor error does not decrease monotonically as iterations proceed. This was also confirmed in our simulations, see Table 4. Since the scope of this paper includes the parameter updates, but no change in the domain assignments for the ranges, we used the annealing approach also in the corresponding version. In other words, in the first step of the annealing method both the partitioning and the domains used for each range were computed and then held fixed throughout all following iterations. The

[3] CPU times reported in this subsection were measured on an SGI O2 R5000 processor.

min. block size	16x16	8x8	4x4
compression ratio	70.2	23.8	9.8
attractor error (in dB)	23.46	26.96	30.87
step 1	23.50	27.00	30.92
step 2	23.53	27.01	30.91
step 3	23.50	27.00	30.92
step 4	23.52	27.00	30.93
step 5	23.50	27.01	30.93
step 6	23.53	27.01	30.92
step 7	23.50	27.00	30.93
step 8	23.52	27.01	30.92
step 9	23.50	27.00	30.92
step 10	23.53	27.00	30.93
step 21	23.51	⋮	⋮
step 24		27.00	⋮
step 32			30.92
best result (dB)	23.53	27.01	30.93
gain (in dB)	0.07	0.05	0.06
total time	0m:35s	0m:58s	1m:31s

Table 4. Results for the annealing method.

iterations were terminated after a total of ten iterations without improvement, i.e., iterations for which the attractor quality in PSNR had dropped. In this method quantization is already employed in each step, of course, and thus, no extra loss occurs as for the gradient method, which must rely on continuous parameters. Table 4 lists the results for the same test image as above. We see that the improvements are in the range between 0.05 and 0.07 dB PSNR.

5 Conclusions and future work

In this paper we have derived the theoretical fundamentals necessary for any application of differentiable methods for attractor error reduction in fractal image compression, namely

- the establishment of the differentiability of the attractor image as a function of its (real valued) scale and offset parameters, and
- the feasibility of gradient computation by iteration of a properly defined vector Iterated Function System with gray level Maps.

Moreover, we have implemented gradient descent algorithms for the problem and reported computational results for a few test cases. While the computer programs have demonstrated that the methods work in practice, the outcomes, however, are not promising. Although gains for the simple encoding based on the method of Dudbridge and Monro are around one half of a dB in PSNR,

the conceptually less complex method using a simplex hill climbing algorithm performs just as well at the same cost in terms of computation time. Furthermore, the achievable gains for fractal coding with the quadtree method are negligible, while the competing annealing methods gave slightly better improvements at a fraction of the cost in computation time. Thus, we must conclude that gradient descent methods for parameter optimization do not provide a useful addition to conventional fractal encoding.

Even if parameter updates cannot break the "collage barrier", not all is lost. The remaining chance to achieve this goal is to also consider changing the domain assignments for the range blocks, as already proposed in [3,10]. This, of course, is a discrete process and we cannot use differentiable methods as presented in this paper. Our current work focusses on algorithms to accomplish this without sacrificing the property of monotonically decreasing errors that we have, e.g, for the gradient descent algorithms.

6 Acknowledgments

We thank Raouf Hamzaoui and Hannes Hartenstein for valuable discussions as well as for some independent numerical calculations in order to verify our results. Andreas Zerbst carried out the computer experiments implementing the gradient descent and annealing methods for quadtree fractal encodings based on Yuval Fisher's public domain program [7].

This research was supported in part by a grant from the DFG Schwerpunktprogramm *Ergodentheorie, Analysis und Effiziente Simulation Dynamischer System* (DANSE).

ERV gratefully acknowledges the partial support of this research by the Natural Sciences and Engineering Council of Canada, in the form of an Operating Grant as well as a Collaborative Projects Grant.

References

1. Barnsley, M.F., *Fractals Everywhere*, Academic Press, New York, 1988.
2. Barnsley, M.F., Ervin, V., Hardin, D., and Lancaster. J., Solution of an inverse problem for fractals and other sets, Proc. Nat. Acad. Sci. USA **83** (1985) 1975–1977.
3. Barthel, K. U., Festbildcodierung bei niedrigen Bitraten unter Verwendung fraktaler Methoden im Orts- und Frequenzbereich (Dissertation, Technische Universität Berlin), Wissenschaft & Technik Verlag, Berlin, 1996.
4. Centore, P., Vrscay, E.R., Continuity of attractors and invariant measures for Iterated Function Systems, Canad. Math. Bull. **37**(3) (1994) 315–329.
5. Domaszevicz, J., Vaishampayan, V.A., Iterative collage coding for fractal compression, in *Proceedings of the IEEE Intern. Conf. Image Proc.* (ICIP'94), 1994.
6. Dudbridge, F., Fisher, Y., Attractor optimixation in fractal image encoding, in Proceedings of Third "Fractals in Engineering" Conference," Arcachon, France, May, 1997.

7. Fisher, Y., *Fractal Image Compression, Theory and Application*, Springer-Verlag, New York, 1995.
8. Forte, B., Vrscay, E.R., Solving the inverse problem for function and image approximation using Iterated Function Systems, Dyn. Cont. Disc. Imp. Syst. **1** (1995) 177–231.
9. Forte, B., Vrscay, E.R., Inverse problem methods for generalized fractal transforms, in *Fractal Image Encoding and Analysis*, ed. Y. Fisher (Springer Verlag, Heidelberg, 1998).
10. Lu, N., *Fractal Imaging*, Academic Press, New York, 1997.
11. Mendivil, F., Vrscay, E.R., Correspondence between fractal-wavelet transforms and Iterated Function Systems with Grey-level Maps, in *Fractals in Engineering*, ed. J. Lévy Véhel, E. Lutton and C. Tricot (Springer, New York, 1997).
12. Monro, D. M., Dudbridge, F., Fractal block coding of images, Electronics Letters **28**,11 (1992) 1053–1055.
13. NAG Fortran Library, The Numerical Algorithms Group Ltd, Oxford, UK.
14. Press, W. H., Teukolsky, S. A., Vetterling, W. T., Flannery, B. P., *Numerical Recipes in C,* Second Edition, Cambridge University Press, 1992.
15. Ruhl, M., Hartenstein, H., Optimal fractal coding is NP-hard, in *Proceedings of the IEEE Data Compression Conference*, ed. J. Storer and M. Cohn, Snowbird, Utah, 1997.
16. Withers, D., Newton's method for fractal approximation, Constructive Approximation **5** (1989) 151–170.