PERGAMON Computers and Mathematics with Applications 0 (2002)
www.elsevier.com/locate/camwa

Fractal Block-Coding: A Functional Approach for Image and Signal Processing

C. A. Cabrelli
Departamento de Matemática
FCEyN, Universidad de Buenos Aires
Cdad. Universitaria, Pab. I, (1428) Buenos Aires, Argentina
and
CONICET, Consejo Nacional de Investigaciones Científicas y Tecnológicas
cabrelli@dm.uba.ar

M. C. Falsetti
Instituto de Desarrollo Humano, Área de Matemática
Universidad Nacional de General Sarmiento
(1663) Roca 850, San Miguel, Buenos Aires, Argentina
mcfalse@dm.uba.ar

U. M. Molter
Departamento de Matemática, FCEyN, Universidad de Buenos Aires
Cdad. Universitaria, Pab. I, (1428) Buenos Aires, Argentina
and
CONICET, Consejo Nacional de Investigaciones Científicas y Tecnológicas
umolter@dm.uba.ar

(Received May 2000; revised and accepted July 2001)

Abstract—In this paper, we show how the generalized self-similarity model introduced by Cabrelli *et al.* in [1] can be used for the coding and reconstruction of digital signals and images. We also prove how the block-coding techniques introduced by Jacquin [2] for measures fit naturally into this functional model allowing us to take advantage of local redundancy of the images, as well as to fully automate the encoding and decoding process.

Keywords—Functional equation, Grey-level functions, Fixed points, Fractal compression, Block-coding, Self-similarity.

1. INTRODUCTION

In this paper, we intend to provide a mathematical setting for a method of fractal block-coding of image functions based on the notion of *generalized self-similarity* introduced by Cabrelli *et al.* in [1]. This method allows more flexibility in the self-similarity structure of standard block-coding compression.

We present the theoretical framework in its most general setting. This can then easily be adapted to fit each individual application. In particular, this general setting includes as special

Partially supported by Grants UBACyT TW84, CONICET PIP 456/98 and PICT98 03134.

0898-1221/02/$ - see front matter Typeset by $\mathcal{AMS}$-TeX
PII:00

cases all the known IFS frameworks. As an application of the theory, we describe a method of coding and decoding images and we indicate how the well-known technique of partitioned IFS (see, for example, [2–6]) can be incorporated just as overlapping blocks (see [7]). We show the method for some very simple one- and two-dimensional examples that help to show some evidence of its usefulness.

The method is related to the *fractal compression method* proposed originally by Barnsley *et al.* in [8] and [9] in the sense that the original image will be approximated by the fixed-point *attractor* of a contractive operator. This attractor can be obtained by iterating the operator starting from an arbitrary starting point. More specifically, the theoretical setting of our method is closely related to the *generalized fractal transforms* described in [10].

We will consider images represented by functions. An image can be modeled by a function $u : X \to [0,1]$, where X is a compact metric space (for applications, $X \subset \mathbb{R}^d$ will usually suffice) and the value $u(x)$, for each $x \in X$, can be interpreted as the normalized *grey-scale* at the point x.

In [1], Cabrelli *et al.* presented the notion of *generalized self-similarity*, by solving the following functional equation:

$$\mathbf{u}(x) = \mathcal{O}\left(x, \varphi_1\left(x, \tilde{\mathbf{u}}_1(x)\right), \ldots, \varphi_n\left(x, \tilde{\mathbf{u}}_n(x)\right)\right), \tag{1.1}$$

where $\tilde{\mathbf{u}}_i$ essentially (except for well definition) means $\mathbf{u} \circ w_i^{-1}$. Using the previously-mentioned interpretation of an image as a function, one can think of the solution of this equation as a generalization of the classical IFS, by introducing the *grey-level functions*, $\varphi_i : X \times [0,1] \to [0,1]$, $1 \le i \le r$, which together with the maps $w_i : X \to X$, $i = 1, 2, \ldots, r$ conform a contractive operator whose fixed point will approximate the original image.

The model allows much greater flexibility in the choice of the parameters, in particular, the relation between the choice of the φ_i and the color seems to be more transparent than when working with measures. In addition, a wider class of images can be represented, since the attractors need not be *self-similar* in the strict sense.

The function $\mathbf{u}$, solution of (1.1), satisfies a *generalized self-similarity relation*, which colloquially could be seen as follows: at any given point $x \in X$, we look at the preimages of x through each w_i, look at the values of $\mathbf{u}$ at that point, transform them using the corresponding φ_i, and then combine these values using the operator $\mathcal{O}$. This new value has to coincide with the original value of $\mathbf{u}$ at x.

The approximation techniques used to solve equation (1.1) take advantage of the dependence on spatial variables of the functions φ_i. The effectiveness of this *place-dependence* condition was originally treated by Monro and Dudbridge (see [6]). This condition was also incorporated in other models as, for example, in the generalized fractal transforms (see, for example, [11]).

The purpose of this paper is twofold. On one side, we show how the generalized self-similarity equation can be applied to signal and image processing. We want to point out that the method presents the following important features: the dependence not only on the grey or color values, but also on the point $x \in X$. Moreover, overlap is allowed, i.e., the domains of w_i are allowed to intersect with nonempty interior. Finally, the application of the operator $\mathcal{O}$ allows us to manipulate several possible preimages of a point just by combining the terms $\varphi_i(x, \tilde{\mathbf{u}}_i(x))$. This is important in the case of overlapping range blocks. In fact, we will show some practical examples where the usage of such an operator allows for a "smoother" combination of overlapping terms as opposed to the "bumpiness" encountered when they are added (see, the examples in Section 4).

On the other side, the greater flexibility in the choice of the parameters allowed us to incorporate the so-called *fractal block-coding* techniques introduced by Jacquin in [2–4] into our model. From the theoretical point of view, the maps w_i, $i = 1, 2, \ldots, n$, need not be defined on the whole space X, but can be defined on subsets $D_i \subseteq X$, such that $\bigcup_{i=1}^{n} w_i(D_i) = X$. This then implies that the function which satisfies equation (1.1), now satisfies an even weaker self-similarity relation, since the preimages of the w_i are the smaller sets D_i which can be thought of as *windows* on which the values of both sides of the equation have to coincide. In addition, as in other block-coding techniques used in image processing, we can also incorporate the *quadtree encoding* procedure, as in [12,13], where the collection of *ranges* is a covering of the image but not a partition in the strict sense because overlap is allowed. Also, the maps w_i need not necessarily be contractive as in most other fractal image coding algorithms.

This theoretical generalization turns out to be very useful for practical implementation purposes. For the so-called inverse problem for fractals, i.e., given a target finding the appropriate code, it enables us to incorporate the complete algorithm automatization (characteristic of the block-coding technique) and to optimize the use of the redundancy, due to the fact that we analyze sections instead of the whole target.

The layout of this paper is as follows. Section 2 provides the mathematical notions about the method. A general application of this theory to image analysis is described in Section 3. In Section 4, we show the direct application to one-dimensional target-functions (signals), and finally, in Section 4.3, we briefly discuss the two-dimensional case.

2. MATHEMATICAL FORMULATION OF THE BLOCK-CODING METHOD FOR FUNCTIONAL SPACES

2.1. $\mathcal{B}(X,E)$ Case

We will consider the particular case of the functional equation (1.1) introduced by Cabrelli *et al.* in [1] on the functional space

$$\mathcal{B}(X,E) = \{u : X \longrightarrow E,\ u \text{ bounded}\},$$

where (X,d) is a compact metric space and (E,ℓ) a metric space with E a closed subset of $\mathbb{R}^m$ (in particular, E could be $\mathbb{R}^m$), and ℓ a distance in E induced by some norm of $\mathbb{R}^m$.

We consider the following distance on the space $\mathcal{B}(X,E)$:

$$\mathbf{L}(u,v) = \sup_{x\in X} \ \ell\,(u(x), v(x)), \qquad \forall\, u, v, \in \mathcal{B}(X,E). \tag{2.1}$$

It is well known that $(\mathcal{B}(X,E), \mathbf{L})$ is a complete metric space.

Now let $\mathbf{R} = \{R_i\}_{1\le i\le n}$ be a class of bounded subsets of X which we will call *ranges*, such that $X = \bigcup_{1\le i\le n} R_i$. We define $\mathbf{W}, \mathcal{O}$, and Φ as follows: $\mathbf{W} = \{w_i\}_{1\le i\le n}$, where $w_i : D_i \subset X \to R_i$ are bijective (D_i, $1 \le i \le n$ will be called *domains*); $\mathcal{O} : X \times E^n \to E$ is a nonexpansive operator for each $x \in X$, i.e.,

$$\ell\left(\mathcal{O}\left(x,\vec{k}\right), \mathcal{O}\left(x,\vec{k}'\right)\right) \le \max_{1\le i\le n} \ell\,(k_i, k_i')\,, \tag{2.2}$$

and $\Phi = \{\varphi_i\}_{1\le i\le n}$ is the set of the *grey-level maps* such that $\varphi_i : X \times E \to E$, $i = 1,\dots,n$, satisfy a Lipschitz condition in the second variable

$$\ell\,(\varphi_i\,(x,k_1)\,, \varphi_i\,(x,k_2)) \le c_i\,\ell\,(k_1,k_2) \qquad \forall\, x \in X, \quad \forall\, k_1,k_2 \in E, \quad i = 1,\dots,n, \tag{2.3}$$

where c_i, $i = 1,\dots,n$, do not depend on x.

Now let us consider a point $t_0 \in E$ that will remain fixed throughout the whole paper. We define the functions

$$\tilde{u}_i(x) = \begin{cases} u\left(w_i^{-1}(x)\right), & \text{if } x \in R_i, \\ t_0, & \text{otherwise,} \end{cases} \qquad 1 \le i \le n, \tag{2.4}$$

and the operator

$$\mathcal{T}(u)(x) = \mathcal{O}\,(x, \varphi_1\,(x, \tilde{u}_1(x))\,, \dots, \varphi_n\,(x, \tilde{u}_n(x)))\,. \tag{2.5}$$

We shall use $\mathcal{O}(x, \overline{\varphi_i(x, \tilde{u}_i(x))})$ for the right-hand side of equation (2.5).

In order for the operator $\mathcal{T}$ to be well defined, that is, $\mathcal{T} : \mathcal{B}(X,E) \to \mathcal{B}(X,E)$, we need to require a *stability* condition on the φ_i and the operator $\mathcal{O}$. Precisely, we need to require that bounded sets are mapped into bounded sets.

Definition. *A function $f : X \to Y$ between two metric spaces is said to be stable, if for each bounded set $A \subset X$, $f(A) \subset Y$ is a bounded subset of Y.*

With these conditions, we have the following theorem.

Theorem 2.1.1. *Let $\mathcal{T}$ be defined on $\mathcal{B}(X,E)$ as*

$$\mathcal{T}(u)(x) = \mathcal{O}\left(x, \varphi_1\left(x, \tilde{u}_1(x)\right), \ldots, \varphi_n\left(x, \tilde{u}_n(x)\right)\right),$$

where $\mathcal{O}$ is a stable operator, and let φ_i, $1 \le i \le n$, be stable functions that are Lipschitz in the second variable.

If $c > 0$, $c = \max_{1\le i\le n}\{c_i\}$, where c_i is the Lipschitz constant for φ_i, $1 \le i \le n$, then

$$\mathcal{T} : B(X,E) \to B(X,E)$$

and

$$\mathbf{L}(\mathcal{T}(u), \mathcal{T}(v)) \le c\, \mathbf{L}(u,v).$$

Proof. It is straightforward to see that if $v \in \mathcal{B}(X,E)$, $\mathcal{T}v \in \mathcal{B}(X,E)$.

For $x \in X$ and $u, v \in \mathcal{B}(X,E)$,

$$\begin{aligned}
\ell((\mathcal{T}u)(x), (\mathcal{T}v)(x)) &= \ell\left(\mathcal{O}\left(x, \overline{\varphi_i\left(x, \tilde{u}_i(x)\right)}\right), \mathcal{O}\left(x, \overline{\varphi_i\left(x, \tilde{v}_i(x)\right)}\right)\right)\\
&\le \max_{1\le i\le n} \ell\left(\varphi_i\left(x, \tilde{u}_i(x)\right), \varphi_i\left(x, \tilde{v}_i(x)\right)\right)\\
&\le \max_{1\le i\le n} c\,\ell\left(\tilde{u}_i(x), \tilde{v}_i(x)\right).
\end{aligned}$$

Since $\bigcup_{1\le i\le n} R_i = X$, there must exist j such that x lies in R_j, hence, if

$$J_x = \{i : 1 \le i \le n \text{ and } x \in R_i\}, \tag{2.6}$$

then $J_x \ne \emptyset$.

We have

$$\begin{aligned}
\ell((\mathcal{T}u)(x), (\mathcal{T}v)(x)) &\le \max_{i\in J_x} c\,\ell\left(\tilde{u}_i(x), \tilde{v}_i(x)\right)\\
&\le c \sup\left\{\ell(u(y), v(y)),\ y \in \bigcup_{i\in J_x} D_i\right\}\\
&\le c \sup_{y\in X} \ell(u(y), v(y))\\
&\le c\, \mathbf{L}(u,v),
\end{aligned}$$

and so

$$\mathbf{L}(\mathcal{T}(u), \mathcal{T}(v)) \le c\, \mathbf{L}(u,v).$$ ∎

From this proposition, we have the following corollary.

Corollary 2.1.1. *If $c < 1$, there exists a unique bounded function $\mathbf{u}^* \in \mathcal{B}(X,E)$ such that*

$$\mathbf{u}^* = \mathcal{O}\left(x, \varphi_1\left(x, \tilde{\mathbf{u}}_1^*(x)\right), \ldots, \varphi_n\left(x, \tilde{\mathbf{u}}_n^*(x)\right)\right),$$

where $\tilde{\mathbf{u}}_i^*$ are defined as in equation (2.4)

Proof. The operator $\mathcal{T}$ is contractive on $(\mathcal{B}(X,E), \mathbf{L})$ which is a complete metric space, and therefore, $\mathcal{T}$ has a fixed point $\mathbf{u}^* \in \mathcal{B}(X,E)$. Clearly, $\mathbf{u}^*$ is the solution to the functional equation. ∎

2.2. $\mathcal{L}^{\mathbf{p}}$ Case

Now let $X \subset \mathbb{R}^d$ compact, with μ the d-dimensional Lebesgue measure and let $E = \mathbb{R}^m$ with some norm $\|\cdot\|$. (Note: E could be chosen to be any Banach space.) We consider the functions $u : X \to E$ such that they are Lebesgue-measurable, and, as usual, functions that are equal almost everywhere are identified.

If $1 \leq p < \infty$, let

$$\mathcal{L}^{\mathbf{p}}(X, E) = \left\{ u : X \to E : \int_X \|u\|^p < +\infty \right\}$$

and

$$\mathcal{L}^{\infty}(X, E) = \{u : X \to E : u \text{ is essentially bounded}\},$$

with $\|u\|_\infty = \text{ess.sup}\|u\|$.

It is well known that $\mathcal{L}^{\mathbf{p}}(X, E)$, $1 \leq p \leq +\infty$, is a Banach space. Then, in this case, let $\mathbf{R} = \{R_i\}_{1 \leq i \leq n}$ be a class of bounded and measurable subsets of X such that $X = \bigcup_{1 \leq i \leq n} R_i$ and $\mathbf{W} = \{w_i\}_{1 \leq i \leq n}$ be a class of measurable maps of X such that $w_i : D_i \subseteq X \to R_i$, $1 \leq i \leq n$, are bijective.

For a measurable u, we define, as before, the operator $\mathcal{T}$ as in equation (2.5),

$$\mathcal{T}(u)(x) = \mathcal{O}\left(x, \varphi_1\left(x, \tilde{u}_1(x)\right), \ldots, \varphi_n\left(x, \tilde{u}_n(x)\right)\right), \tag{2.7}$$

where the φ_i, $\tilde{u}_i$, and $\mathcal{O}$ are as in the previous section. Note that the *nonexpansivity* condition of the operator $\mathcal{O}$ in this case should mean

$$\ell\left(\mathcal{O}\left(x, \vec{k}\right), \mathcal{O}\left(x, \vec{k'}\right)\right) \leq \left(\sum_{1 \leq i \leq n} \ell\left(k_i, k_i'\right)^p\right)^{1/p}. \tag{2.8}$$

We add the following conditions.

1. The maps w_i, $1 \leq i \leq n$, satisfy a Lipschitz condition, i.e., there exist constants $s_i \geq 0$, such that $d(w_i(x), w_i(y)) \leq s_i\, d(x, y)$ where d is the Euclidean distance in $\mathbb{R}^d$.
2. The functions φ_i, $i = 1 \ldots, n$, and $\mathcal{O}$ are Borel measurable.

These additional conditions are required in order to guarantee the measurability of $\mathcal{T}u$. We then have the following proposition.

Proposition 2.2.1. *Let $\mathcal{T}$ be defined as above, then $\mathcal{T}u : X \to E$ is measurable for each measurable function $u : X \to E$. Moreover, if u, v are measurable and $u = v$ a.e., then $\mathcal{T}u = \mathcal{T}v$ a.e.*

Proof. The measurability of $\mathcal{T}u$ is a consequence of the stability and Borel-measurability of $\mathcal{O}$, and the functions w_i, and φ_i. Now if $Z = \{x \in X : u(x) \neq v(x)\}$, then

$$\{x : \mathcal{T}u(x) \neq \mathcal{T}v(x)\} \subset \bigcup_{i=1}^{n} \left(R_i \cap w_i\left(Z \cap D_i\right)\right) \subset \bigcup_{i=1}^{n} w_i(Z).$$

The Lipschitz condition of the w_i implies that $\mu(w_i(Z)) = 0$ if $\mu(Z) = 0$, and therefore, the result follows. ∎

Now we consider first the space $\mathcal{L}^\infty$ defined before. The case $\mathcal{L}^{\mathbf{p}}$, $1 \leq p < \infty$, will be treated later.

Theorem 2.2.1. *Let $\mathcal{T}$ be the operator defined in equation (2.7). Then $\mathcal{T} : \mathcal{L}^\infty \to \mathcal{L}^\infty$ and*

$$\|\mathcal{T}u - \mathcal{T}v\|_\infty \leq \max_{1 \leq i \leq n} c_i \|u - v\|_\infty, \qquad \forall\, u, v \in \mathcal{L}^\infty.$$

PROOF. If $u \in \mathcal{L}^\infty$, then let $Z \subset X$, $\mu(Z) = 0$ and u bounded in $X - Z$. If we define $v : X \to E$ by $v = u \cdot I_{X-Z}$, where I_A is the indicator function of A, then $v = u$ a.e. and v is bounded, so $v \in \mathcal{B}(X, E)$. Then $\mathcal{T}v$ is bounded (by Theorem 2.1.1). Using the preceding proposition, $\mathcal{T}u = \mathcal{T}v$ a.e., and therefore, $\mathcal{T}u \in \mathcal{L}^\infty$.

From the proof of Theorem 2.1.1, we see that for u and $v \in \mathcal{L}^\infty$, if $c = \max_{1\le i\le n}\{c_i : c_i$ Lipschitz constant of $\varphi_i\}$, we have

$$\|\mathcal{T}u(x) - \mathcal{T}v(x)\| \le c\,\|u - v\|, \qquad \text{a.e. on } X,$$

which implies that

$$\|\mathcal{T}u - \mathcal{T}v\|_\infty \le c\ \|u - v\|_\infty. \qquad \blacksquare$$

We can now turn our attention to $\mathcal{L}^{\mathbf{p}}$, $1 \le p < \infty$. We have the following theorem.

THEOREM 2.2.2. *Let $\mathcal{T}$ be operator (2.7). If $u, v \in \mathcal{L}^{\mathbf{p}}(X, E)$, then $(\mathcal{T}u - \mathcal{T}v) \in \mathcal{L}^{\mathbf{p}}(X, E)$ and*

$$\|\mathcal{T}u - \mathcal{T}v\|_p \le \left(\sum_{1\le i\le n} s_i\, c_i^p\right)^{1/p} \|u - v\|_p,$$

where s and c are the Lipschitz constants of w_i and φ_i, respectively.

Furthermore, the finiteness of $\mu(X)$ yields

$$\mathcal{T} : \mathcal{L}^{\mathbf{p}}(X, E) \to \mathcal{L}^{\mathbf{p}}(X, E).$$

PROOF. If $u, v \in \mathcal{L}^{\mathbf{p}}$, then by Proposition 2.2.1, $\mathcal{T}u - \mathcal{T}v$ is measurable and

$$\begin{aligned}
\|\mathcal{T}u - \mathcal{T}v\|_p^p &= \int_X \|\mathcal{T}u(x) - \mathcal{T}v(x)\|^p d\mu(x) \\
&= \int_X \left\|\mathcal{O}\left(x, \overline{\varphi_i\left(x, \tilde{u}_i(x)\right)}\right) - \mathcal{O}\left(x, \overline{\varphi_i\left(x, \tilde{v}_i(x)\right)}\right)\right\|^p\ d\mu(x) \\
&\le \int_X \sum_{1\le i\le n} \|\varphi\left(x, \tilde{u}_i(x)\right) - \varphi\left(x, \tilde{v}_i(x)\right)\|^p\ d\mu(x), \quad \text{by equation (2.8)}, \\
&\le \sum_{1\le i\le n} c_i^p \int_X \|\tilde{u}_i(x) - \tilde{v}_i(x)\|^p\, d\mu(x) \\
&\le \sum_{1\le i\le n} c_i^p \int_X \|\tilde{u}_i(x) - \tilde{v}_i(x)\|^p\ d\mu(x) \\
&\le \sum_{1\le i\le n} c_i^p \int_{R_i} \left\|\left(\mathbf{u} \circ w_i^{-1}\right)(x) - \left(\mathbf{v} \circ w_i^{-1}\right)(x)\right\|^p\ d\mu(x), \quad \text{by equation (2.4)}\ , \\
&\le \sum_{1\le i\le n} c_i^p\, s_i \int_{D_i} \|u(t) - v(t)\|^p\ d\mu(t)
\end{aligned} \tag{2.9}$$

$$\le \left(\sum_{1\le i\le n} c_i^p\, s_i\right) \|u - v\|_p^p. \tag{2.10}$$

From this inequality, we see that if $u, v \in \mathcal{L}^{\mathbf{p}}$, then

$$\|\mathcal{T}v\|_p \le \|\mathcal{T}u - \mathcal{T}v\|_p + \|\mathcal{T}u\|_p \le \left(\sum_{1\le i\le n} c_i^p s_i\right)^{1/p} \|u - v\|_p + \|\mathcal{T}u\|_p, \tag{2.11}$$

which says that if there exists a function $\mathbf{u} \in \mathcal{L}^{\mathbf{p}}$ such that $\mathcal{T}u \in \mathcal{L}^{\mathbf{p}}$, then $\mathcal{T}$ maps $\mathcal{L}^{\mathbf{p}}$ into $\mathcal{L}^{\mathbf{p}}$, $1 \leq p < +\infty$. Now, since $\mu(X) < +\infty$, then $\mathcal{L}^{\infty} \subset \mathcal{L}^{\mathbf{p}}$, $1 \leq p$ and since, by Theorem 2.2.1, $\mathcal{T} : \mathcal{L}^{\infty} \to \mathcal{L}^{\infty}$, we obtain the desired result. ∎

Corollary 2.2.1. *With the above notation, if* $(\sum_{1\leq i\leq n} c_i^p\, s_i)^{1/p} < 1$, *we have that* $\mathcal{T}$ *is a contraction map on* $\mathcal{L}^{\mathbf{p}}$, $1 \leq p \leq \infty$, *and the functional equation*

$$\mathbf{u} = \mathcal{O}\left(x, \varphi_1\left(x, \tilde{\mathbf{u}}_1(x)\right), \dots, \varphi_n\left(x, \tilde{\mathbf{u}}_n(x)\right)\right) \tag{2.12}$$

has a unique solution in $\mathcal{L}^{\mathbf{p}}$.

Note that this condition is weaker than the one of Corollary 2.1.1, and therefore, can be useful when searching for functions that are not necessarily bounded.

3. SOLUTION TO THE INVERSE PROBLEM FOR FRACTALS AND OTHER SETS

In this section, we are going to show how we can use the previous results to attempt a solution to the inverse problem of fractals and other sets for one-dimensional and two-dimensional cases. For simplicity, we describe only the $\mathcal{L}^{\infty}$ case. The $\mathcal{L}^{\mathbf{p}}$ case can be implemented in a similar way.

The idea is as follows. Given $\mathbf{v}$, a bounded function that is the target—signal or image, we want to find the collections $\mathbf{R} = \{R_i\}_{1\leq i\leq n}$, $\mathbf{W} = \{w_i\}_{1\leq i\leq n}$, and $\Phi = \{\varphi_i\}_{1\leq i\leq n}$ and a nonexpansive operator $\mathcal{O}$, such that the resulting operator $\mathcal{T} : \mathcal{L}^{\infty} \to \mathcal{L}^{\infty}$ is contractive and its invariant function is "close" to the target.

Once the sets $\mathbf{R}$ and $\mathbf{W}$ are determined, we obtain another collection of sets, which are the preimages of the R_i through the w_i, $\mathbf{D} = \{D_i = w_i^{-1}(R_i)\}_{1\leq i\leq n}$.

Now let $\mathbf{v} : X \to [0,1]$ be the target function to encode, where $X = [0,1]$, if $\mathbf{v}$ is a one-dimensional signal or $X = [0,1] \times [0,1]$ if $\mathbf{v}$ is two-dimensional. We need to find the following.

1. A finite class of bounded subsets of X, $\mathbf{R} = \{R_i\}_{1\leq i\leq n}$, called *range sets* which have to satisfy $X = \bigcup_{1\leq i\leq n} R_i$.

   For simplicity, we will consider these sets to be intervals. Note that this class does not necessarily need to be a partition of X, overlap is allowed.
2. A finite class of injective maps $\mathbf{W} = \{w_i : D_i \subset X \to X\}_{1\leq i\leq n}$ such that $R_i = w_i(D_i)$, $1 \leq i \leq n$. Note that the w_i can be contractive, expansive, or neither. We will call the subsets D_i, *domain sets*.
3. A set $\Phi = \{\varphi_i\}_{1\leq i\leq n}$, where $\varphi_i : X \times \mathbb{R} \to \mathbb{R}$, $1 \leq i \leq n$, are contractive functions in the second variable. We choose, again for simplicity, φ_i to be affine functions,

$$\varphi_i(x,t) = \begin{cases} (A_i x + c_i t + d_i)\,, & \text{if } x \in R_i, \\ p_0, & \text{if } x \notin R_i, \end{cases} \qquad \forall\, i \;\; 1 \leq i \leq n, \tag{3.1}$$

   where A_i is a 1×1 or 2×1 matrix (depending on the dimension of X), $|c_i| < 1$ and $p_0 \leq 0$ fixed with $|p_0|$ large enough. It is easy to verify that the φ_i defined in this way are contractive in t. Besides they are stable.
4. A nonexpansive and stable operator $\mathcal{O} : X \times \mathbb{R}^n \to \mathbb{R}$. In order to facilitate the implementation, we are going to choose $\mathcal{O}$ such that, for $u \in \mathcal{B}(X,E)$, if $u(x)$ is "close" to $\varphi_i(x, \tilde{u}_i(x))$, $\forall\, x \in X$, then u is "close" to $\mathcal{T}u$ too.

   For example, for the one-dimensional case, we obtained good results with the following operator:

$$\mathcal{O}\left(x, \vec{\mathbf{k}}\right) = \sup_{1\leq i\leq n} \{k_i\}. \tag{3.2}$$

Remark 1. For simplicity, we have chosen the class $\Phi = \{\varphi_i\}_{1\le i\le n}$ defined in (3.1), but φ_i could be defined in a different way. It has yet to be studied how the code is modified or improved with another selection of Φ.

Remark 2. Once we decide which kind of functions we choose for Φ and the operator $\mathcal{O}$ is determined, it is clear that the operator $\mathcal{T}$ is now completely determined by the choice of the R_i and w_i.

Remark 3. In the particular case that $\mathbf{R} = \{R_i\}_{1\le i\le n}$ is a partition of X, (i.e., $R_i \cap R_j = \emptyset$, $i \ne j$), the expression of the functional equation using operator (3.2) is

$$\mathcal{T}(u)(x) = \left(A_i x + c_i \tilde{u}_i(x) + d_i\right), \qquad \text{if } x \in R_i. \tag{3.3}$$

Note the similitude of this case with the model proposed in [6].

Remark 4. For the two-dimensional case, the choice of the supremum operator for $\mathcal{O}$ was not very efficient. We will return to this subject in Section 4.3.

3.1. Encoding Method

From the preceding discussion, it is clear that in order to encode a given target function $\mathbf{v} : X \to E$, (where X stands either for the interval $[0,1]$ or the square $[0,1]\times[0,1]$ and $E = \mathbb{R}$), we need to find a collection $\mathbf{R}$ of intervals, called *range blocks*, a class $\mathbf{W}$ of affine functions, and a collection Φ so that the distance between the fixed point of the associated operator $\mathcal{T}$ and $\mathbf{v}$ is less than a prescribed value. This distance is the *approximation error*. In order to estimate the approximation error, we use the *collage theorem*.

Theorem 3.1.1. *Let $v \in \mathcal{B}(X,E)$ and $\mathcal{T}$ be as defined in equation (2.5) with contractivity factor $c < 1$. If*

$$\mathbf{L}(v, \mathcal{T}(v)) < \epsilon,$$

then

$$\mathbf{L}\left(v, \mathbf{u}^*\right) < \frac{\epsilon}{1-c}$$

where $\mathbf{u}^*$ is the invariant bounded function of $\mathcal{T}$.

The proof of this theorem is analogous to the IFS case (see [14]).

We conclude that the approximation error of the procedure depends only on the encoding error and the Lipschitz constant c. Therefore, to obtain our code, we start with the target $\mathbf{v}$ and we search for parameters to define the operator $\mathcal{T}$ such that the result of the first transformation $\mathcal{T}\mathbf{v}$ is "close enough" to $\mathbf{v}$. By the last theorem, we then know that the error between the target and $\mathcal{T}^k(u_0)$, for any $u_0 \in \mathcal{B}(X,E)$ and a large enough k, will be small, since the sequence $\{\mathcal{T}^k(u_0)\}_{k\in\mathbb{N}}$ converges to the fixed point of $\mathcal{T}$.

In order to compute the distance $\mathbf{L}$, since $X = \bigcup_{1\le i\le n} R_i$, if $u, v \in \mathcal{B}(X,E)$, $\mathbf{L}(u,v)$ can be written in terms of the distance on each block R_i, i.e.,

$$\begin{aligned}\mathbf{L}(u,v) &= \sup_{x\in X} |u(x) - v(x)| \\ &= \max_{0\le i\le n}\left(\sup_{x\in R_i} |u(x) - v(x)|\right).\end{aligned} \tag{3.4}$$

3.2. Partition of X and Definition of W and Φ

We describe now, in a general way, how we construct the collections $\mathbf{R}$ and $\mathbf{W}$. (In Section 4, using specific examples, we will give a more detailed description.) These collections will be constructed recursively and their cardinals depend on the error that is computed at each step of the recursion.

Given a target function $\mathbf{v}$, the method fixes *a priori* a threshold error ε. The encoding procedure starts using a fixed number of blocks, $\tilde{R}_j$, $1 \leq j \leq N_0$, $N_0 \in \mathbb{N}$, such that $\bigcup_j \tilde{R}_j = X$. For each j, $1 \leq j \leq N_0$, we construct a large enough "pool" of injective affine maps, $\tilde{\mathbf{W}}_j$, whose image sets are $\tilde{R}_i$, out of which we will choose the corresponding w_j. Consistent with equation (3.4), we choose an appropriate $\mathcal{O}$ which allows us to reduce our encoding procedure to determine the parameters of φ_j so that

$$|\mathbf{v}(x) - \varphi_j\left(x, \tilde{\mathbf{v}}_j(x)\right)| < \delta = \delta(\varepsilon), \qquad \forall\, x \in R_j,$$

where $\delta = \delta(\varepsilon)$ is such that $\sup_{x \in R_j} |\mathbf{v}(x) - \mathcal{O}(x, \varphi_1(x, \tilde{\mathbf{v}}_j(x)), \ldots, \varphi_n(x, \tilde{\mathbf{v}}_j(x)))| < \varepsilon$. We will prove later, in Section 4.3, that for an appropriate selection of $\mathcal{O}$, such δ exists.

For each choice of $\tilde{w}_i \in \tilde{\mathbf{W}}_j$, we now need to determine the coefficients for the function $\tilde{\varphi}_j : X \times \mathbb{R} \to \mathbb{R}$. For this, we will begin choosing the coefficient of the "second variable", or the "contractivity coefficient" c_j (see equation (3.1)), from a predetermined large enough set of values V. To determine the remaining coefficients, we use any numerical approximation method.

Now we compute

$$\delta_{ji} = \sup_{x \in \tilde{R}_j} \left|\mathbf{v}(x) - \tilde{\varphi}_j\left(x, \mathbf{v}\left(\tilde{w}_i^{-1}(x)\right)\right)\right|. \tag{3.5}$$

We do this (as said earlier) for each choice of $\tilde{w}_i \in \tilde{\mathbf{W}}_j$ and $c \in V$.

Observe that each of the *domain sets* $\tilde{D}_{ji} = \tilde{w}_i^{-1}(\tilde{R}_j) \subseteq X$ plays the role of a "window" that "opens" at different places until it finds the section of the function whose transformation is the "closest" to the target function v restricted to $\tilde{R}_j$.

We select the pair $(\tilde{w}_j, \tilde{\varphi}_j)$ that produces the smallest error. If this error is greater than δ, we divide $\tilde{R}_j$ into *children* blocks (we use the same name introduced by Jacquin in [2]), $\{\tilde{R}_{jk}\}$, two subintervals or four rectangles depending on the case, and $\tilde{R}_j$ is discarded. If the error computed on one of these *child* blocks, R_{jk}, is smaller than δ, this subblock is saved as a block of the final code, R_l, together with the maps $\tilde{w}_j|_{R_{jk}}$ and φ_j corresponding to the *parent* block, renamed now w_l and φ_l. For all other subblocks whose errors are greater than δ, the procedure is repeated from the beginning, as for their *parent* block.

This encoding technique involves a recurrent algorithm that finalizes once the errors on all subblocks are smaller than δ. Once the algorithm stops, the sets $\mathbf{R}$ and $\mathbf{W}$ as well as the corresponding Φ are defined by $\mathbf{R} = \{R_i,\ 1 \leq i \leq n\}$ (where n depends on the steps of the recurrence), where each R_i is a $\tilde{R}_j$ or one of its subdivisions for some, $1 \leq j \leq N_0$, $\mathbf{W} = \{w_i, , 1 \leq i \leq n\}$ are the bijective affinities chosen for R_i, respectively, and $\Phi = \{\varphi_i, 1 \leq i \leq n\}$ with $\varphi_i : X \times \mathbb{R} \to \mathbb{R}$.

By equation (3.4), if the partial error on each block is bounded, the global error on X is bounded too. From Theorem 3.1.1, we have,

$$\mathbf{L}(\mathbf{v}, T(\mathbf{v})) < \varepsilon \Rightarrow \mathbf{L}(\mathbf{v}, \mathbf{u}^*) < \frac{\varepsilon}{1-\beta},$$

where $\beta = \max\{|c_i|\}$, by the choice of Φ in equation (3.1). This limits (or conditions) the set V from which we chose our coefficients c. For example, if $V = \{\pm i/10, \pm(i/10 + 5/100), 0 \leq i \leq 9\}$, and we want to reconstruct the target with an error smaller than $\gamma > 0$, it suffices that ε is less than $\gamma/20$.

In order to reconstruct the target, once the conventions about w and $\mathcal{T}$ are established, we only need the *coding triples* $\mathcal{C} = \{(R_i, w_i, \varphi_i), 1 \leq i \leq n\}$. Given $\mathcal{C}$, we construct $\mathcal{T}$ and iterate on any starting function $u_0 \in \mathcal{B}(X, E)$.

4. EXAMPLES

4.1. Code Construction for $\sin x$ and $\sqrt{x}$: Method I

In this section, we apply the proposed method to some simple examples, just to illustrate the possible implementation. We stress the visually almost perfect reconstruction obtained in these examples.

We will show how we can already obtain very good results from our theory, even without the incorporation of the block-coding technique: just looking at the particular case in which *all* w_i are defined on the whole space $X = [0,1]$.

We are only going to take advantage of the fact that we are allowing overlap. We will show how this works to encode and reconstruct the functions $\sin \pi x$ and $\sqrt{x}$ which are not self-similar if only affine maps are admitted for the φ_i. We will choose the operator $\mathcal{T}$ as in equation (3.2).

If $\mathbf{v}$ is the target function to encode (i.e., $\mathbf{v}(x) = \sin x$ or $\mathbf{v}(x) = \sqrt{x}$), then the method essentially consists of two steps.

1. We fix a natural number N and let $\varepsilon > 0$ be $\varepsilon = 1/(10N)$. We take the N elementary maps $w_i : [0,1] \to [0,1]$,

$$\begin{aligned} w_1(x) &= \left(\frac{1}{N} + \varepsilon\right) x, \\ w_i(x) &= \left(\frac{1}{N} + \varepsilon\right) x + \frac{i-1}{N} - \frac{\varepsilon}{2}, \qquad 1 \le i \le N-1, \\ w_N(x) &= \left(\frac{1}{N} + \varepsilon\right) x + \frac{N-1}{N} - \varepsilon. \end{aligned}$$

2. We now construct φ_i, $1 \le i \le N$, such that

$$\mathcal{T}(u)(x) = \sup_{1 \le i \le n} \{\varphi_i\,(x, \tilde{u}_i(x))\}$$

with $\tilde{u}_i(x) = u(w_i^{-1}(x))$ if $x \in w_i([0,1])$ and 0 otherwise, as defined in equation (2.4). So, if one ignores for a moment the overlap, for the target function $\mathbf{v}$, one essentially wants

$$\begin{aligned} \varphi_i\left(x, \mathbf{v}\left(w_i^{-1}(x)\right)\right) &= \mathbf{v}(x), \qquad && \forall\, x \in w_i([0,1]),\ \text{i.e.,} \\ \varphi_i\left(w_i(z), \mathbf{v}(z)\right) &= \mathbf{v}\left(w_i(z)\right), && \forall\, z \in [0,1]. \end{aligned} \tag{4.1}$$

To construct these φ_i, we consider a fixed number P of points P_j in [0,1], write equation (4.1) for them:

$$\varphi_i\left(w_i\left(P_j\right), \mathbf{v}\left(P_j\right)\right) = \mathbf{v}\left(w_i\left(P_j\right)\right), \qquad 1 \le j \le P, \tag{4.2}$$

and using (for example) a least square approximation algorithm, we determine the parameters of functions φ_i so that

$$\varphi_i(x,t) = a_i x + c_i t + d_i, \qquad 1 \le i \le N,$$

are affine approximations that satisfy equation (4.2).

Note that this method corresponds to the description in Section 3 for the case

$$\mathbf{R} = \left\{ \left[0, \frac{1}{N} + \varepsilon\right] \cup \left\{ \left[\frac{i-1}{N} - \frac{\varepsilon}{2}, \frac{i}{N} + \frac{\varepsilon}{2}\right] \right\}_{2 \le i \le N-1} \cup \left[\frac{N-1}{N} - \varepsilon, 1\right] \right\}$$

and $\mathbf{W}$ just the similarity transformations.

We have constructed N base functions (which obviously are one-to-one) and N grey level functions. We now take $\mathcal{T} = \mathcal{T}(N, \mathbf{w}, P)$, such that

$$\mathcal{T}u(x) = \sup_{1 \le i \le N} \varphi_i \left(x, u \circ w_i^{-1}(x) \right).$$

According to Corollary 2.1.1, the existence of a fixed point $\mathbf{u}^*$ of $\mathcal{T}$ defined above, $\mathbf{u}^* \in (\mathcal{B}([0,1],[0,1]), \mathbf{L})$ is guaranteed if $|c_i| < 1,\ 1 \le i \le N$. If the result of the numerical method

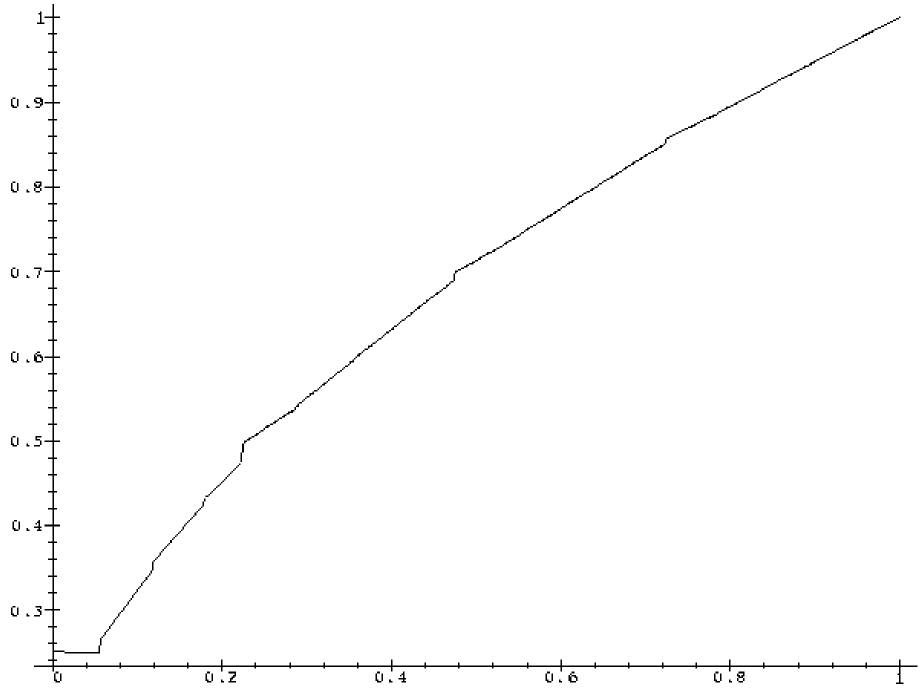

Figure 1. Reconstruction of $\sqrt{x}$ using Method I: first iteration.

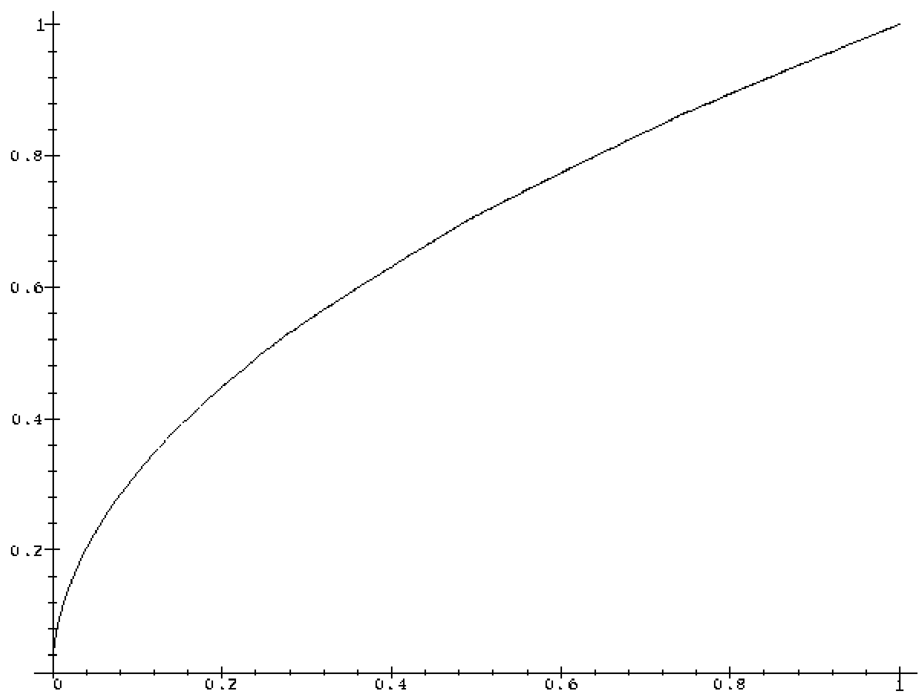

Figure 2. Reconstruction of $\sqrt{x}$ using Method I: 21^{st} iteration.

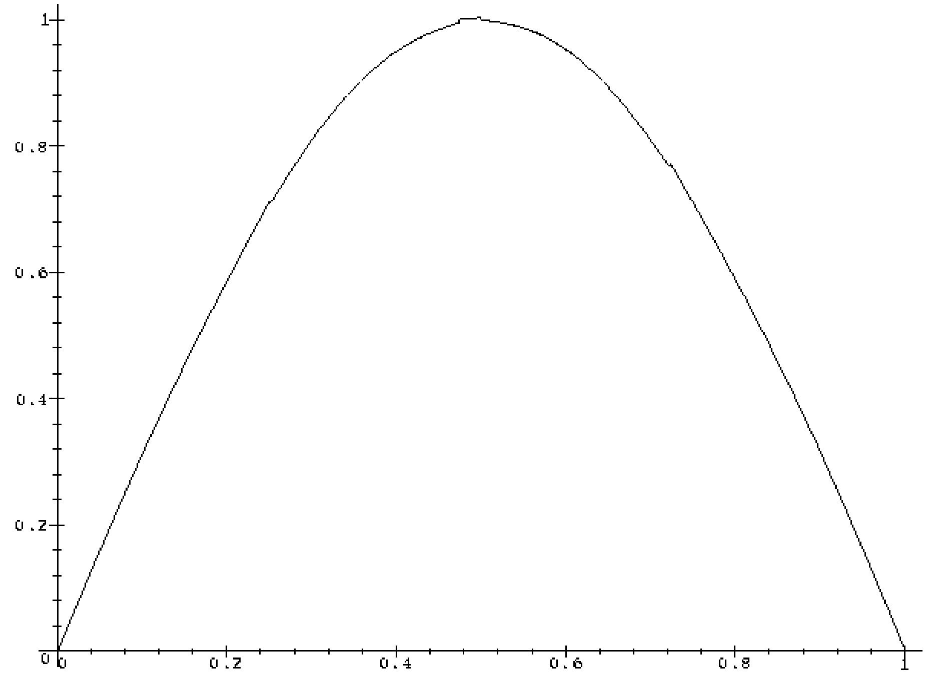

Figure 3. Reconstruction of sine using Method I: 16$^{\text{th}}$ iteration.

employed to solve equation (4.2) does not verify this condition, i.e., $c_i < -1$ or $c_i > 1$, then, for some small $\delta > 0$, we consider $c_i' = -1 + \delta$ or $c_i' = 1 - \delta$ and the numerical method is repeated to find new a_i and d_i with this fixed value of c_i'.

If Err is the error committed by taking the linear approximations

$$\mathrm{Err} = \max_{1 \le i \le N} \sup_{x \in [(i-1)/N,\, i/N]} \left(|\mathbf{v}(x) - (a_i x + c_i \mathbf{v}(Nx - i + 1) + d_i)| \right), \tag{4.3}$$

then $\mathbf{u}^*$ satisfies

$$\mathbf{L}(\mathbf{u}^*, \mathbf{v}) \le \frac{\mathrm{Err}}{1 - \max_{1 \le i \le N} c_i}.$$

If this value turns out to be too large, one needs to start again, fixing a different N. It is apparent that the algorithm described in Section 3, where the determination of N depends automatically on the threshold error is preferable—even though the visual results for our test functions were excellent.

For the examples (see Figures 1–3), we restricted (and normalized) the functions to the $[0, 1]$ interval and fixed the number of maps to be four.

4.2. Code Construction for exp and sine: Method II

In this section, we will complete the description of the block-coding algorithm, for these one-dimensional examples. For these examples, the operator $\mathcal{O}$ is chosen to be the *supremum* operator (see equation (3.2)). We will explain now which particular class $\mathbf{W}$ we use. The space $X = [0, 1]$ is subdivided into intervals $\{\tilde{R}_i\}_{1 \le i \le N_0}$ of length ρ. We allow overlap by extending ρ by $\gamma/2$ at each extreme of the interval, for some $\gamma > 0$. As indicated in Section 3.2, for each $\tilde{R}_i$, we work with a large enough pool of maps $\tilde{\mathbf{W}}_i$, out of which w_i are allowed to be chosen.

We define $\tilde{\mathbf{W}}_i$ to be the collection of affine maps obtained in the following way. We determine a fixed *step-length* $1/r$ and an interval $\tilde{D}$ of length $\alpha\,\rho$, at the origin, where α can be either 1/2, 2, or 3/2. For each k, $1 \le k \le r$, $\tilde{D}$ is then translated inside X by $(k-1)/r$ yielding the interval $\tilde{D}_l$. The functions $\tilde{w}_k$ are then all the possible maps that bijectively take $\tilde{D}_k$ into $\tilde{R}_i$

and are a composition of three simple transformations, $\tilde{w}_k = s \circ \tau \circ \zeta$, where ζ is a homothetic transformation of ratio $1/\alpha$, τ is a translation, and s a symmetry on $\tilde{R}_i$ (i.e., s_k can invert the extremes). For the examples that we show in this section, it was not necessary to consider symmetries in order to obtain good results.

In order to pick our w_i, we will have to construct the function φ_i, and compute the error. For this, we choose c_h, such that $|c_h| < 1$, from a fixed finite set $V = \{\pm l/10, (\pm l/10 + 5/100), l \in \mathbb{N}, 0 \le l \le 9\}$. Let m denote the cardinal of this set. For each fixed $c_h \in V$, $1 \le h \le m$, we take every $\tilde{w}_k \in \tilde{\mathbf{W}}_i$. We then determine a^i_{hk} and d^i_{hk} taking into account that we are trying to approximate the target function $\mathbf{v}$. That means that the following condition should be satisfied:

$$\mathbf{v}(x) \simeq a^i_{hk} x + c_h \mathbf{v}\left(\tilde{w}_k^{-1}(x)\right) + d^i_{hk}, \qquad \forall\, x \in \tilde{R}_i,$$

or its equivalent:

$$\mathbf{v}(x) - c_h \mathbf{v}\left(\tilde{w}_k^{-1}(x)\right) \simeq a^i_{hk} x + d^i_{hk} \qquad \forall\, x \in \tilde{R}_i.$$

The parameters a^i_{hk} and d^i_{hk} are now estimated using, for example, a least-squares approximation method. Note that the parameter a^i_{hk} appears because the φ and the operator $\mathcal{O}$ depend on the variable x.

Once a^i_{hk} and d^i_{hk} are chosen, we calculate the error for this choice of c_h and $\tilde{w}_k$. It is precisely

$$\varepsilon^i_{hk} = \sup_{x \in \tilde{R}_j} \left| \mathbf{v}(x) - \left(a^i_{hk} x + c_h \mathbf{v}\left(w_k^{-1}(x)\right) + d^i_{hk} \right) \right|. \tag{4.4}$$

Once $\varepsilon^i_{hk} < \varepsilon$, then the interval $\tilde{R}_i$ will be an element of the class $\mathbf{R}$, say R_l, the transformation $\tilde{w}_k$ will be its corresponding element of $\mathbf{W}$ called w_l, and the parameters a^i_{hk}, c_h, d^i_{hk}, now called a_l, c_l, d_l, will define the corresponding φ_l as in equation (3.1).

If for all choices of h and k, $1 \le h \le m$, $\tilde{w}_k \in \tilde{\mathbf{W}}_i$, $\varepsilon^{ik}_j \ge \varepsilon$, then we divide the interval $\tilde{R}_j$ into two subintervals. We compute the error on each subinterval as in equation (4.4). If on one of those, the error is lower than ε, this subinterval is saved as a member of the code, call it R_l, its corresponding element in $\mathbf{W}$ is $w_l = \tilde{w}_j|_{R_l}$ and the parameters of φ_l are a^{hk}_j, c_h, d^{hk}_j calculated for its *parent* interval, renamed now a_l, c_l, d_l. For the other subinterval, we repeat the above steps.

If n is the cardinal of $\mathbf{R}$, the code consists of the description of the division of the interval $[0,1]$, $\mathbf{R} = \{R_i\}_{1 \le i \le n}$, the bijective maps $\mathbf{W} = \{w_i\}_{1 \le i \le n}$ and of $\Phi = \{\varphi_i\}_{1 \le i \le n}$.

The following figures show the results of the described method. Figures 4 and 5 refer to the exponential function, exp, ($f(x) = e^{(x+1)}$ from $[-1,1]$ normalized on the $[-1,1]$ interval). The *a priori* allowed error was 0.001. After applying the algorithm, the code resulted in two *code triples*

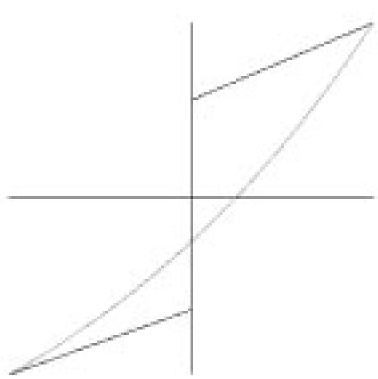

Figure 4. Reconstruction of exp using Method II: first iteration.

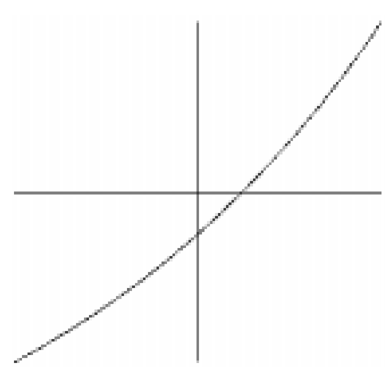

Figure 5. Reconstruction of exp using Method II: 10^{th} iteration.

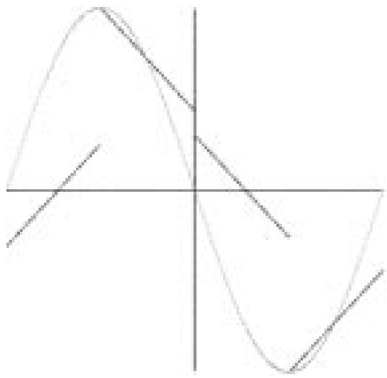

Figure 6. Reconstruction of sine using Method II: first iteration.

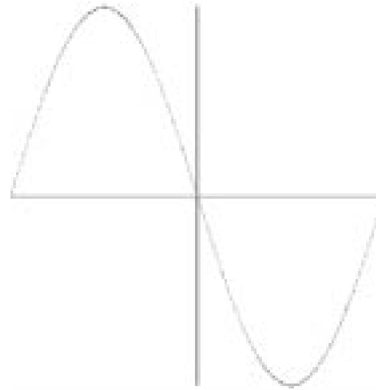

Figure 7. Reconstruction of sine using Method II: 10^{th} iteration.

(see equation (3.2)) and the encoding error is lower than 0.0018. Figures 6 and 7 correspond to the function sine ($f(x) = \sin(\pi(x+1))$ from $[-1,1]$) whose code resulted in four *code triples* for the whole wave with an encoding error lower than 0.003. The figures show the reconstruction starting from the function $f(x) = \chi_{[0,1]}$. In all figures, the original target function is also included, to stress the visual similarity.

4.3. Brief Discussion About Two-Dimensional Cases

The block-coding method is of much higher complexity in the two-dimensional case. Two blocks now touch not only at a single point (as in the one-dimensional case), but on a whole edge. On the other hand, two-dimensional images themselves might have *borders* or *edges* which have to be taken into account when coding them. Very often, undesirable aliasing occurs due to the combination of the edges of the blocks and the image. The choice of the right operator $\mathcal{O}$ is crucial in this case. For example, the *supremum*-operator, which in the one-dimensional case leads to very good results, did not yield good results in the two-dimensional case.

In order to obtain subtle shadings and grey-scales, we need to take into account the overlap which this method allows. As seen in Section 2, the class $\mathbf{R}$ of blocks that covers the image does not need to be a partition of the image. This is an advantage of this method over the one of Jacquin, where a partition in the strict sense is required.

For example, we partition the $[0,1]\times[0,1]$ square in rectangles and extend their edges by an ϵ so that a rectangle can intersect neighboring rectangles and at most four rectangles can intersect at one corner. For a given $\mathbf{x}$ in the intersection of two or more blocks, several φ are applied which then are conveniently combined through the operator $\mathcal{O}$. Therefore, it is apparent that the operator should depend strongly on $\mathbf{x} = (x_1, x_2)$. Let $\tilde{\mathbf{R}} = \{\tilde{R}_i\}_{1\le i\le N_0}$ be a set of square blocks, of side ρ, that cover $X = [0,1]\times[0,1]$. For each $\tilde{R}_i$, we need to define $\tilde{\mathbf{W}}_i$ out of which we choose the corresponding w_i. For this, take a square $\tilde{D}$ of side $\alpha\,\rho$ at the origin where α is again (as in the one-dimensional case) either 2, 1/2, or 3/2. We translate $\tilde{D}$ so that it covers a predetermined fixed area near to $\tilde{R}_i$, resulting in a collection $\{\tilde{D}_k\}_{1\le k\le r}$ of blocks. For each $\tilde{D}_k$, there are exactly eight affine functions that bijectively map $\tilde{D}_k$ onto $\tilde{R}_i$; each one results from the composition of one of the eight isometries that map the square $\tilde{R}_i$ into itself, with a translation and a homothetic transformation of ratio $1/\alpha$. We will keep the same class of functions Φ as for the one-dimensional case (see equation (3.1)) , i.e.,

$$\varphi_i\left((x_1,x_2),t\right) = \begin{cases} (a_i x_1 + b_i x_2 + c_i t + d_i)\,, & \text{if } \mathbf{x}\in R_i, \\ p_0, & \text{if } \mathbf{x}\notin R_i, \quad p_0 \text{ fixed}, \end{cases} \qquad \forall\, i,\ 1\le i\le n. \tag{4.5}$$

Now, we will define $\mathcal{O}$. As we said earlier, we will choose the $\tilde{R}_i$ such that $X = \bigcup_{i=1}^{n} \tilde{R}_i$ in the

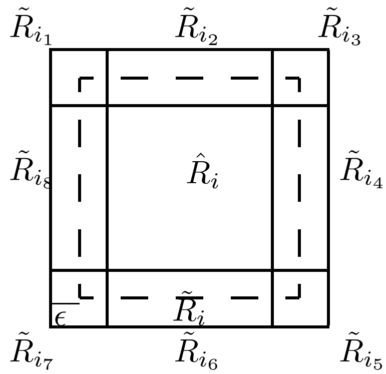

Figure 8. $\hat{R}_i$'s edges (dashed block) are extended by ϵ. $\tilde{R}_i$ overlaps next blocks $\tilde{R}_{i_j}$.

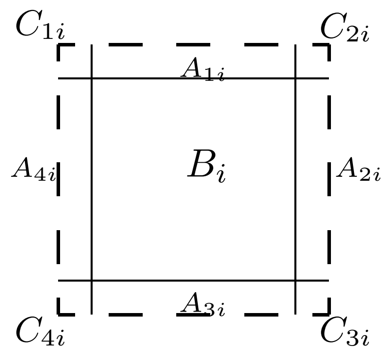

Figure 9. Regions in $\hat{R}_i$ determined by intersections of $\tilde{R}_i$ and $\tilde{R}_{i_j}$.

following way. Let $\hat{R}_i$, $1 \leq i \leq n$, be such that $X = \bigcup_{i=1}^{n} \hat{R}_i$ with $\hat{R}_i \cap \hat{R}_j = \emptyset$ if $i \neq j$. We then extend the edges of $\hat{R}_i$ by ϵ to obtain the overlapping blocks $\tilde{R}_i$ shown in Figure 8.

Given $\tilde{R}_i$, there are exactly eight blocks $\{\tilde{R}_{i_j}\}_{1 \leq j \leq 8}$ such that $\tilde{R}_{i_j} \cap \tilde{R}_i \neq \emptyset$. These blocks determine in $\hat{R}_i$ the regions as shown in Figure 9.

We will consider that the regions A_{ji}, C_{ji}, and B_i do not intersect, for example, by extracting left and lower borders. We define $I_{ji} = \{k \in \mathbb{N} : \bigcap_k \tilde{R}_k = C_{ji}\}$ and $H_{ji} = \{k \in \mathbb{N} : \bigcap_k \tilde{R}_k = A_{ji}\}$. Then if $\mathbf{x} \in X$, since $\bigcup_{1 \leq i \leq n} \hat{R}_i = X$, certainly $\mathbf{x}$ lies in one of these regions of $\hat{R}_i$. We define the following operator $\mathcal{O} : X \times \mathbb{R}^n \to \mathbb{R}$:

$$\mathcal{O}(\mathbf{x}, \vec{\mathbf{k}}) = \begin{cases} \alpha_{j1}^i k_{m_1} + \alpha_{j2}^i k_{m_2} + \alpha_{j3}^i k_{m_3} + \alpha_{j4}^i k_{m_4}, & \mathbf{x} \in C_{ji}, \quad m_l \in I_{ji}, \\ & 1 \leq l \leq 4, \quad 1 \leq j \leq 4, \\ \beta_{j1}^i k_{m_1} + \beta_{j2}^i k_{m_2}, & \mathbf{x} \in A_{ji}, \quad m_l \in H_{ji}, \\ & 1 \leq l \leq 2, \quad 1 \leq j \leq 4, \\ k_i, & \mathbf{x} \in B_i, \end{cases} \tag{4.6}$$

with

$$\begin{array}{llllll} \alpha_{jk}^i \geq 0, & 1 \leq j \leq 4, & 1 \leq k \leq 4, & \text{and} & \alpha_{j1}^i + \alpha_{j2}^i + \alpha_{j3}^i + \alpha_{j4}^i = 1, \\ \beta_{jk}^i \geq 0, & 1 \leq j \leq 4, & 1 \leq k \leq 2, & \text{and} & \beta_{j1}^i + \beta_{j2} = 1, \quad 1 \leq i \leq n. \end{array}$$

Note that if we would like to stress even more the dependence on $\mathbf{x}$, we could allow these parameters also to depend on $\mathbf{x}$. It is easy to verify that $\mathcal{O}$ is nonexpansive. For example, if $\mathbf{x} = (x_1, x_2) \in C_{1i}$,

$$\begin{aligned} \left| \mathcal{O}\left(\mathbf{x}, \vec{\mathbf{k}}\right) - \mathcal{O}\left(\mathbf{x}, \vec{\mathbf{k}'}\right) \right| &= \left| \alpha_{11}^i \left(k_{m_1} - k'_{m_1}\right) + \cdots + \alpha_{14}^i \left(k_{m_4} - k'_{m_4}\right) \right| \\ &\leq \alpha_{11}^i \left\| \vec{\mathbf{k}} - \vec{\mathbf{k}'} \right\| \cdots + \alpha_{14}^i \left\| \vec{\mathbf{k}} - \vec{\mathbf{k}'} \right\| \\ &\leq \left(\alpha_{11}^i + \alpha_{12}^i + \alpha_{13}^i + \alpha_{14}^i\right) \left\| \vec{\mathbf{k}} - \vec{\mathbf{k}'} \right\| \\ &\leq \left\| \vec{\mathbf{k}} - \vec{\mathbf{k}'} \right\|. \end{aligned}$$

The other cases are similar.

Since $\mathcal{O}$ is defined using affine combinations, if $A \subset X \times \mathbb{R}^n$ is bounded, $\mathcal{O}(A)$ is also bounded, therefore, $\mathcal{O}$ is stable too. If $|c_i| < 1$, $1 \leq i \leq n$, the operator $\mathcal{T}$ defined in equation (2.5) is contractive, with contractivity factor $c = \max_{1 \leq i \leq n} |c_i|$. By Corollary 2.1.1, there exists $\mathbf{u}^*$ such that $\mathcal{T}\mathbf{u}^* = \mathbf{u}^*$.

Now, we proceed as in Sections 3.2 and 4.2 to define the classes $\mathbf{R}$, $\mathbf{W}$, and Φ.

The following lemma shows that the operators defined by equation (3.2) or equation (4.6) are suitable for this encoding model. Recall the definition of J_x from equation (2.6),

$$J_x = \{i : 1 \leq i \leq n \text{ and } x \in R_i\}.$$

LEMMA 4.3.1. *Let $\mathcal{O}$ be defined as in equation (3.2) or equation (4.6), $u \in \mathcal{B}(X, \mathbb{R})$ and $\mathcal{T}$ the corresponding operator. For $\epsilon > 0$, there exists $\delta > 0$, $\delta = \delta(\epsilon)$, such that if $|u(x) - \varphi_i(x, \tilde{u}_i(x))| < \delta$, $i \in J_x$, $\forall\, x \in X$, then $\mathbf{L}(u, \mathcal{T}u) < \epsilon$.*

PROOF. We separate the proof into two cases.

THE SUPREMUM OPERATOR (3.2). Here we need to choose p_0 in the definition of φ_i equation (3.1) such that $p_0 < -\epsilon - \sup_{x \in X} u(x)$, in order to have that $\sup_{1 \leq i \leq n}\{\varphi_i(x, \tilde{u}_i(x))\} = \sup_{i \in J_x}\{\varphi_i(x, \tilde{u}_i(x))\}$, $\forall\, x \in X$. If $x \in X$ is such that $|u(x) - \varphi_i(x, \tilde{u}_i(x))| < \delta$ and taking $\delta = \epsilon$, we have

$$\begin{aligned}
\mathcal{T}(u)(x) &= \sup_{1 \leq i \leq n} \{\varphi_i\,(x, \tilde{u}_i(x))\} \\
&= \sup_{i \in J_x} \{\varphi_i\,(x, \tilde{u}_i(x))\} \\
&= \max_{i \in J_x} \{\varphi_i\,(x, \tilde{u}_i(x))\} = \varphi_{i_0}\,(x, \tilde{u}_{i_0}(x))\,, \qquad \text{for some } i_0 \in J_x.
\end{aligned}$$

Hence, $|u(x) - \mathcal{T}u(x)| = |u(x) - \varphi_{i_0}(x, \tilde{u}_{i_0}(x))| < \delta = \epsilon$ and the lemma is true.

OPERATOR (4.6). We do not need any restriction for p_0 in this case. Here the cardinal of J_x is at most four. Suppose, for example, that $x = (x_1, x_2)$ is in the corner C_{1i}, for some i, $1 \leq i \leq n$. Then $J_x = I_{1i}$ (see equation (4.6)) and the operator $\mathcal{T}$ results

$$\mathcal{T}(u)(x) = \alpha^i_{11}\varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) + \alpha^i_{12}\varphi_{m_2}\,(x, \tilde{u}_{m_2}(x)) + \alpha^i_{13}\varphi_{m_3}\,(x, \tilde{u}_{m_3}(x)) + \alpha^i_{14}\varphi_{m_4}\,(x, \tilde{u}_{m_4}(x))\,,$$

with $m_l \in I_{1i}$, $1 \leq l \leq 4$.

We now evaluate the distance between $u(x)$ and $\mathcal{T}(u)(x)$,

$$\begin{aligned}
|u(x) - \mathcal{T}(u)(x)| \leq{}& \left|u(x) - \left(\alpha^i_{11} + \alpha^i_{12} + \alpha^i_{13} + \alpha^i_{14}\right)\varphi_{m_1}\,(x, \tilde{u}_{m_1}(x))\right| \\
&+ \alpha^i_{12}\,|\varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) - \varphi_{m_2}\,(x, \tilde{u}_{m_2}(x))| \\
&+ \alpha^i_{13}\,|\varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) - \varphi_{m_3}\,(x, \tilde{u}_{m_3}(x))| \\
&+ \alpha^i_{14}\,|\varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) - \varphi_{m_4}\,(x, \tilde{u}_{m_4}(x))| \\
<{}& \delta + \alpha_{12}\,|(-u(x) + \varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) - (-u(x) + \varphi_{m_2}\,(x, \tilde{u}_{m_2}(x))| \\
&+ \cdots + \alpha_{14}\,|-u(x) + \varphi_{m_1}\,(x, \tilde{u}_{m_1}(x)) - (-u(x) + \varphi_{m_4}\,(x, \tilde{u}_{m_4}(x))| \\
<{}& \delta + \alpha^i_{12}(2\delta) + \alpha^i_{13}(2\delta) + \alpha^i_{14}(2\delta) \\
<{}& 7\delta \quad \text{since, by (4.6),} \qquad 0 \leq \alpha^i_{jk} \leq 1.
\end{aligned}$$

If x lies in another region of $\hat{R}_i$, we obtain a similar inequality. The lemma is therefore true if we take $\delta \leq \epsilon/7$. ∎

Choosing operator (4.6), we obtained much better results for the encoding of images than with the supremum operator. Already, without allowing overlap, the resulting image is much better, but the block-effect shows very strongly if there is no overlap. This effect disappears completely when the same operator is chosen, but overlap is allowed. The different shading is as smooth as in the original image (Figure 10), as can be seen in the figures below.

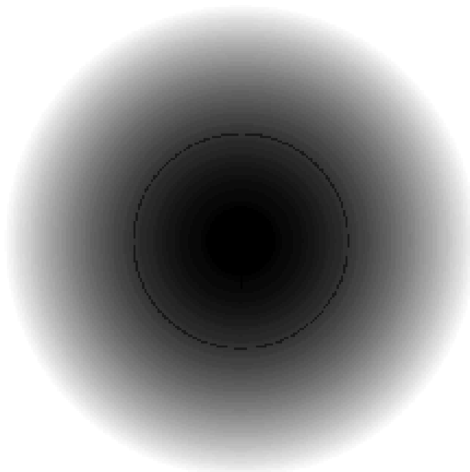

Figure 10. Original image.

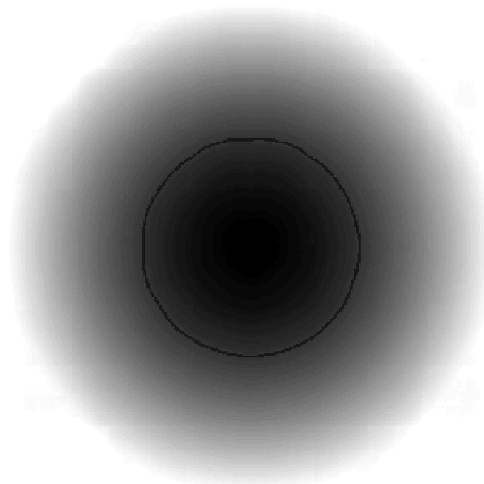

Figure 11. Circle reconstructed with $\mathcal{T}_1$.

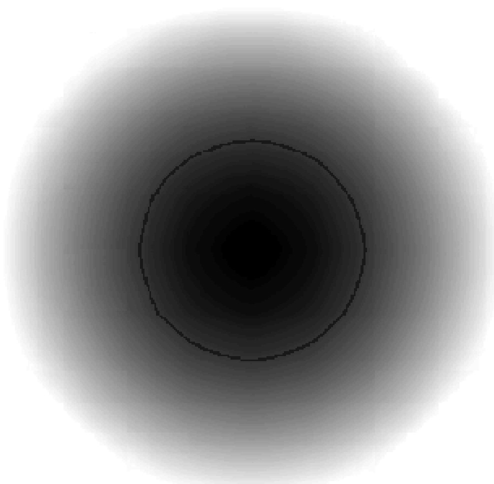

Figure 12. $\mathcal{T}_2$ without overlap.

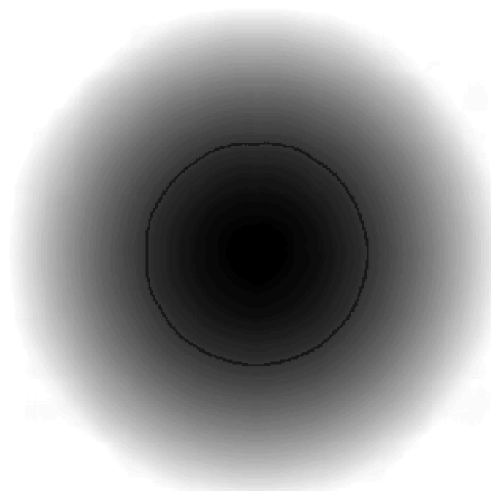

Figure 13. $\mathcal{T}_2$ with overlap.

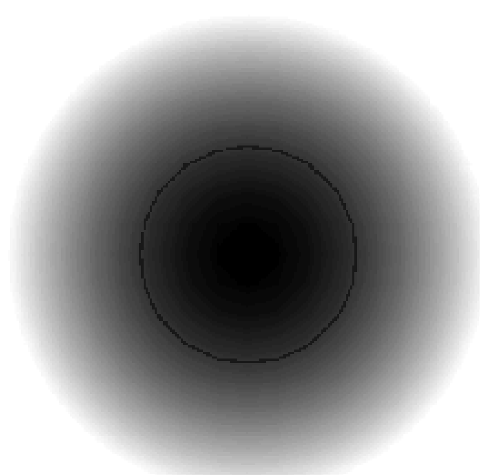

Figure 14. Reconstructed with $\mathcal{T}_2$ and 256 triples.

We denote by $\mathcal{T}_1$ the supremum operator defined in equation (3.2) and $\mathcal{T}_2$ the operator defined in equation (4.6). Figure 13 shows the grey scale circle reconstructed by $\mathcal{T}_1$ with overlap, Figures 10 and 12 are reconstructed using $\mathcal{T}_2$ without overlap and with overlap, respectively. These figures have a code of 64 triples. In Figure 10, little spots can be observed near the border because, although overlap has a great advantage for improving shading, it provokes a little distortion of the border zones. However, this effect disappears by increasing the number of blocks of the partition. For example, the reconstruction shown in Figure 11 of the same image encoded with 256 triples is perfect.

All figures show the 10^{th} iteration with overlap. The starting function for the reconstruction algorithm is $u_0(x) = \chi_{[0,1]\times[0,1]}(x)$ and the threshold error $\varepsilon = 0.01$.

REFERENCES

1. C. Cabrelli and U. Molter, Generalized self-similarity, *Journal of Mathematical Analysis and Applications* **230**, 251–260 (1999).
2. A.E. Jacquin, Image coding based on a fractal theory of iterated contractive Markov operators, Parts I and II, Georgia Tech. Ph.D. Thesis 091389-017 (1989).
3. A.E. Jacquin, A novel fractal block-coding technique for digital images, *ICASSP* **4**, 2225–2228 (1990).
4. A. Jacquin, Image coding based on fractal theory of iterated contractive image transformations, *IEEE Trans. Image Processing* **1**, 18–30 (1992).
5. A. Jacquin, Fractal image coding: A review, *Proceedings of the IEEE* **81,** (10) (1993).
6. D.M. Monro and F. Dudbridge, Fractal approximation of image blocks, *Proc. ICASSP* **3**, 485–488 (1992).
7. Y. Fisher, Sample code, In *Fractal Image Compression*, (Edited by Y. Fisher), pp. 259–291, Springer-Verlag, (1995).
8. M.F. Barnsley and S. Demko, Iterated function systems and global construction of fractals, *Proceedings of the Royal Society of London* **A399**, 243–275 (1985).
9. M.F. Barnsley, V. Ervin and D.P. Hardin, Solution of an inverse problem for fractals and other sets, *Proc. Natl. Acad. Sci. USA* **83** (1986).
10. B. Forte and E. Vrscay, Theory of generalized fractal transform, In *Fractal Image Encoding and Analysis*, NATO ASI Series F, (Edited by Y. Fisher), Springer-Verlag, Berlin, (1998).
11. B. Forte and E. Vrscay, Inverse problem methods for generalized fractal transform, In *Fractal Image Encoding and Analysis*, NATO ASI Series F, (Edited by Y. Fisher), Springer-Verlag, Berlin, (1998).
12. Y. Fisher, Fractal image compression with quadtrees, In *Fractal Image Compression*, (Edited by Y. Fisher) pp. 55–77, Springer-Verlag, (1995).
13. F. Dudbridge, Linear time fractal quadtree coder, In *Fractal Image Encoding and Analysis*, NATO ASI Series F, (Edited by Y. Fisher), Springer-Verlag, (1998).
14. M.F. Barnsley, *Fractals Everywhere*, Academic Press, (1988).