Fractal Based Unstructured Object Recognition

S. Kar
Computer Science and Engg.
IIT, Powai
Mumbai, INDIA 400076
kar@cse.iitb.ernet.in

S. Chandran
Computer Science and Engg.
IIT, Powai
Mumbai, INDIA 400076
sharat@cse.iitb.ernet.in

Abstract

Many conventional methods of object recognition depend solely on the geometric structure of the underlying objects. This strategy is eminently suitable for man-made, or robot-made objects. Natural objects are, for the most part, unstructured in the sense of Euclidean geometry. But there is an inbuilt degree of self-similarity in the image of natural objects which can be efficiently utilized through a representation exploiting self-transformations, known as Iterated Function System (IFS).

Interestingly, virtually all images of natural or *man-made objects, show region-wise self-similarity although they may not be globally self-similar. Such objects can be represented by Partitioned Iterated Function System (PIFS) very compactly.*

In this paper we propose an object recognition scheme based on such PIFS representation and matching carried out in the PIFS code domain, which we argue is more efficient than correlation in the image domain. We base our proposition on theoretical arguments. The recognition method is attractive from the view-point of execution time as the PIFS code library of reference objects are built off-line and recognition of query object involves only comparison of its PIFS code with those in the library.

Keywords: Object recognition, geometric structure, unstructured object, self similarity, fractal, Partitioned Iterated Function System.

1 Introduction

Object recognition is a very important task in any machine vision application. Many conventional methods of object recognition depend solely on the geometric structure of the underlying objects for their successful operation, e.g., the method based on edge detection followed by edge analysis. Although many objects have definable geometrical structure at a gross level, images of many scenes such as deserts, forests, mountains, snow, etc. are in general unstructured in the sense of Euclidean geometry. (Fig. 1 shows a synthetic unstructured Fern image.) But there is an inbuilt degree of self-similarity in such objects and this property can be efficiently utilized through a representation exploiting self-transformations.

Such self-similar objects may be termed fractal. Fractal objects can be represented by an Iterated Function System which turns out to be very compact.

Most images that we encounter in day-to-day life however are a super-imposition of a variety of objects and are *not* self similar. Virtually all images, however, appear to show region-wise self-similarity although they may not be globally self-similar, and may be termed as generalized fractals. Generalized fractals are represented by the Partitioned Iterated Function system. While not as compact as the IFS, PIFS appear to have less redundancies than a simple image. In this paper we employ the PIFS representation of images whose homogeneity may vary in general from region to region, in order to recognize them in a time efficient way.

The rest of this paper is organized as follows. We start with the background behind our present work followed by the theory behind the validity of the proposed recognition technique. In Section 4, we provide the details to generate a canonical PIFS code of an image which helps us achieve recognition in spite of translation, rotation, and scale changes. We present test results in Section 5 after briefly discussing the big-Oh complexity of our method, and finally make concluding remarks.

2 Background

Although the IFS structure of fractal object representation has been utilized by many researchers for the purpose of image compression, there have been very few attempts towards object recognition. In [5] the authors have a recognition scheme applicable to the specific domain of L-System fractals and have tested their technique on binary synthetic plant images generated by the L-System. In [4] a recognition method is suggested which (i) works on binary images, and (ii) which is based on applying the reference set of PIFS codes on the query object and finding out the code which produces minimum change. The query is then recognized to be the object corresponding to that PIFS code, if the change found out was less than a threshold.

In the present paper, we explore the possibility of **gray** level object recognition based on **direct comparison of PIFS codes** between the query and the reference. In this sense, it is an inverse procedure to the method in [4]. Since the PIFS representation is very compact, the recognition can be carried out very fast with the reference code data generated off-line and kept in a library. Our method is based on theoretical arguments absent elsewhere to our knowledge, and its validity is tested on synthetic images.

3 Theoretical Basis

In this section, after introducing a few definitions, lemmas and theorems, we propose our method of comparing two fractal images based on the parameter values of their IFS codes. The complete and compact metric space, and the Hausdorff metric h we talk about, are as formulated in [1].

Definition 3.1 *Let (X, d) be a metric space. A function $f : X \to X$ is continuous if, for each $\epsilon > 0$ and $x \in X$, there is a $\delta > 0$ so that $d(x, y) < \delta \Rightarrow d(f(x), f(y)) < \epsilon$ for all $y \in X$.*

Definition 3.2 *A transformation $w : \Re^2 \to \Re^2$ of the form : $w(x, y) = (ax + by + e, cx + dy + f)$, where a, b, c, d, e and f are real numbers, is called a 2-D affine linear transformation. The numbers a, b, c, d, e and f are called the parameters of the transformation.*

Definition 3.3 *Let $f : X \to X$ be a transformation on a metric space. A point $x_f \in X$ such that $f(x_f) = x_f$ is called a fixed point of the transformation.*

Definition 3.4 *A transformation $f : X \to X$ on a metric space (X, d) is called contraction mapping if there is a constant $0 \leq s < 1$ such that $d(f(x), f(y)) \leq s.d(x, y), \forall x, y \in X$. Any such number s is called a contractivity factor for f.*

Definition 3.5 *Let (X, d) be a complete metric space. Then $H(X)$ denotes the Hausdorff space consisting of non-empty compact subsets of X. This is the space of fractal images.*

Theorem 3.1 *Let (X, d) be a complete metric space. Then $(H(X), h)$ is a complete metric space. [Proof of this is given in [1, pp. 35].]*

Lemma 3.1 *Let $w : X \to X$ be a contraction mapping on the metric space (X, d) with contractivity factor s.Then $w : H(X) \to H(X)$ defined by $w(B) = \{w(x) : x \in B\}, \forall B \in H(X)$ is a contraction mapping on $(H(X), h(d))$ with contractivity factor s. [Proofs of this and the following lemma are given in [1, pp. 79].]*

Lemma 3.2 *Let (X, d) be a metric space. Let $\{w_n : n = 1, 2, 3, ..., N\}$ be contraction mappings on $(H(X), h)$. Let the contractivity factor for w_n be denoted by s_n for each n. Then $W : H(X) \to H(X)$ such that $W(B) = w_1(B) \cup w_2(B) \cup .... \cup w_n(B) = \cup_{n=1}^{N} w_n(B), \forall B \in H(X)$ is a contraction mapping with contractivity factor $s = \max\{s_n : n = 1, 2, ..., N\}$.*

Definition 3.6 *An iterated function system (IFS) consists of a complete metric space (X, d) together with a finite set of contractive mappings $w_n : X \to X$, with respective contractivity factors $s_n, n = 1, 2, 3, ..., N$. The notation for IFS is $\{X; w_n, n = 1, 2, ..., N\}$ and its contractivity factor is $s = \max\{s_n : n = 1, 2, ..., N\}$. Here we will restrict the contraction mappings to be affine transformations of the form of definition 3.2.*

Theorem 3.2 *Let $\{X; w_n, n = 1, 2, 3, ..., N\}$ be an IFS with contractivity factor s. Then the transformation $W : H(X) \to H(X)$ defined by $W(B) = \cup_{n=1}^{N} w_n(B), \forall B \in H(X)$ is a contraction mapping on the complete metric space $(H(X), h(d))$ with contractivity factor s : that is $h(W(B), W(C)) \leq s.h(B, C), \forall B, C \in H(X)$. It has a unique fixed point, $A \in H(X)$, which obeys $A = W(A) = \cup_{n=1}^{N} w_n(A)$, and is given by $A = lim_{n\to\infty} W^{\circ n}(B)$ for any $B \in H(X)$. [Proof of this theorem is given in [1].]*

Definition 3.7 *The fixed point $A \in H(X)$ described in the above theorem is called the attractor of the IFS.*

Definition 3.8 *Let (X, d) be a complete metric space. Let $L \in H(X)$ be given, and let $\epsilon \geq 0$ be given. Choose an IFS $\{X; w_1, w_2, ..., w_N\}$ with contractivity factor $0 \leq s < 1$, so that $h(L, \cup_{n=1}^{N} w_n(L)) \leq \epsilon$. Then, $\{w_i, i = 1, 2, ..., N\}$ is called the IFS code of L.*

3.1 Proposition for Comparison of Fractals by IFS Parameters

Claim: Let (X, d) be a complete metric space. Let $H(X)$ be the space of nonempty compact subsets of X. Suppose $A, B \in H(X)$. Let $\{w_{An}, n = 1, 2, ..., N_1\}$ and $\{w_{Bn}, n = 1, 2, ..., N_2\}$ be IFS codes of A and B respectively. Let $W_A = \{w_{An}\}$ and $W_B = \{w_{Bn}\}$. W_A and W_B have underlying parameters p_A and p_B respectively, where $p_A, p_B \in P$, the corresponding space of parameters. (P is typically a subspace of the Euclidean space.) Let d_p be a metric on P. Then the distance measure $d_p(p_A, p_B)$ computed in (P, d_p) space is the same measure as $h(A, B)$.

Informally, then, **by measuring $d_p(p_A, p_B)$, we will get a measure of similarity between A and** B. We note that measuring $d_p(p_A, p_B)$ is much faster computationally as compared to measuring $h(A, B)$ due to the compact representation of IFS codes.

Our proposition makes the following **assumption** : $\{w_{An}\}$ and $\{w_{Bn}\}$ satisfy Lipschitz continuity criterion; i.e., for any w_n of these two sets, which has parameter $p \in P$, (so that we can write the mapping explicitly as w_{np}), and for any $x \in X$, if we vary the parameter value from p to $q \in P$ to get the mapping w_{nq}, then, $\exists k > 0$ satisfying $d(w_{np}(x), w_{nq}(x)) \leq k.d_p(p, q)$. k is the so-called Lipschitz constant.

Proof : We divide the proof in 2 parts. In part 1, we argue that the IFS codes **truly encode** the points of $H(X)$. In part 2, we prove that the **parameters of the IFS codes are continuous over the encoded points of** $H(X)$ [1]. This will ensure the equivalence of distance and, hence, of similarity between two points $A, B \in H(X)$ measured by Hausdorff metric (h) and the corresponding distance of parameters $p_A, p_B \in P$ in their IFS encodings (W_A, W_B) using the metric d_p.

Part 1: Barnsley's Collage Theorem states - For given $L \in H(X)$ and given $\epsilon \geq 0$, if we choose an IFS $\{X; w_1, w_2, ..., w_N\}$ with contractivity factor $0 \leq s < 1$ so that

$$h(L, \cup_{n=1}^{N} w_n(L)) \leq \epsilon, \tag{1}$$

then

$$h(L, A_T) \leq \epsilon/(1 - s), \tag{2}$$

where A_T is the attractor of the IFS. This theorem implies that $\{w_1, w_2, ..., w_N\}$ encodes L, as the set A_T obtained after decoding the IFS code differs from L at most by $\epsilon/(1 - s)$ in h metric. Also, if we can make the distance in eq. 1 = 0 by the ideal choice of the IFS mappings, then A_T will be exactly same as L. Hence, W_A, W_B in our proposition are said to encode A, B.

Now the contraction mappings W_A, W_B possess exactly one fixed point each lying in $H(X)$. [Proof of this is given in [1, pp. 75-77], for a contraction mapping in the metric space (X, d). The same extends in the present context.]

Hence, we find that the IFS code for $A(B)$, i.e. $W_A(W_B)$, when decoded, produce unique set (attractor) which is within a distance of $\epsilon/(1-s)$ from $A(B)$.

[1]For some points of $H(X)$, the continuity is lost, but such points are really very few, so that we can work well with our proposition for most of the practical situations [3].

That is why we say that the IFS codes **truly represent** the encoded points of $H(X)$.

Part 2: The contraction mappings constituting the IFS codes in our discussion have some parameters whose values, found during encoding process, lie in a parameter space P. Writing explicitly, let $W : P \times H(X) \to H(X)$ be a family of contraction mappings on $H(X)$ with contractivity factor $0 \leq s < 1$. That is, for each $p \in P, W(p, \bullet)$ is a contraction mapping on $H(X)$. Let W be continuous on P for each fixed starting set $A_X \in H(X)$. Then we show that *the fixed point of W depends continuously on p.* To do this, first we show the following - Let $w_n(p, \bullet)$ be a contraction map (which can be any of the component maps of $W = \cup_{n=1}^{N} w_n$) such that $w_n : P \times X \to X$ with contractivity factor $0 \leq s < 1, p \in P$. For each starting point $x \in X$, let w_n be continuous on P. Then *the fixed point of w_n depends continuously on p.*

Let $x_f(p)$ denote the fixed point of w_n for fixed $p \in P$. Let $\epsilon > 0$ be given. Then for all $q \in P$,

$$\begin{aligned} d(x_f(p), x_f(q)) &= d(w_n(p, x_f(p)), w_n(q, x_f(q))) \\ \leq d(w_n(p, x_f(p)), w_n(q, x_f(p))) &+ d(w_n(q, x_f(p)), w_n(q, x_f(q))) \\ \leq d(w_n(p, x_f(p)), w_n(q, x_f(p))) &+ s.d(x_f(p), x_f(q)), \end{aligned}$$

$$\Rightarrow d(x_f(p), x_f(q)) \leq (1-s)^{-1} d(w_n(p, x_f(p)), w_n(q, x_f(p))) \tag{3}$$

The distance factor in the right hand side can be made arbitrarily small ($= \delta$) by restricting q to be sufficiently close to p (this follows from the continuity of w_n). Taking $\delta = \epsilon(1 - s)$, we can satisfy $d(x_f(p), x_f(q)) \leq \epsilon$. This proves the continuity of fixed point of w_n on p for a particular x [1, pp. 104].

We have to move this *continuous dependence of fixed point of a contraction mapping on p to the space* $H(X)$. To do this we constrain w_n by Lipschitz continuity (initial assumption). We write our IFS code for a $B1 \in H(X)$, parameterized by $p \in P$, as $W_p = \{w_{1p}, ..., w_{Np}\}$, where each w_{np} corresponds to $w_n(p, \bullet)$ above. We vary p continuously to, say, q to have the IFS code W_q. Let the fixed points of $W_p(W_q)$ be $A_p(A_q)$. Then, we want to prove that $d_p(p, q) < \delta \Rightarrow h(A_p, A_q) < \epsilon$. In order to do that, first, in the following paragraph we prove that under the Lipschitz continuity condition of

$$d(w_{np}(x), w_{nq}(x)) \leq k.d_p(p, q), k > 0, \tag{4}$$

the following is satisfied -

$$d_p(p, q) < \delta \Rightarrow h(w_{np}(B1), w_{nq}(B1)) < \epsilon, \tag{5}$$

$\forall B1 \in H$ & $n = 1, 2, ..., N$.

In the succeeding paragraph we prove that $h(W_p(B1), W_q(B1)) \leq k.d_p(p,q)$. In the next paragraph we argue to show the desired continuous dependence of the fixed point of a contraction mapping on p in the space $H(X)$.

We now establish that if the inequality 4 is satisfied, then relation 5 holds good. From the definition of Hausdorff distance,

$$h(w_{np}(B1), w_{nq}(B1)) = d(w_{np}(B1), w_{nq}(B1)) \bigvee d(w_{nq}(B1), w_{np}(B1))$$

where, $d(w_{np}(B1), w_{nq}(B1)) = \max\{d(x, w_{nq}(B1)) : x \in w_{np}(B1)\}$, and $d(x, w_{nq}(B1)) = \min\{d(x,y) : y \in w_{nq}(B1)\}$.
Also, $x \in w_{np}(B1))$ implies that there is an $x1 \in B1$ such that $x = w_{np}(x1)$. Then there is a point $w_{nq}(x1) \in w_{nq}(B1)$, which is the image of $x1$ under w_{nq}. For this point our condition 4 holds, and $d(x, w_{nq}(x1)) \leq k.d_p(p,q) \Rightarrow \min\{d(x,y) : y \in w_{nq}(B1)\} \leq d(x, w_{nq}(x1)) \leq k.d_p(p,q)$.
Since this condition holds for every $x \in w_{np}(B1)$, the maximum over these points is at most $k.d_p(p,q)$, and we have $d(w_{np}(B1), w_{nq}(B1)) \leq k.d_p(p,q)$. This argument is nearly identical for $d(w_{nq}(B1), w_{np}(B1))$. Therefore, $h(w_{np}(B1), w_{nq}(B1)) \leq k.d_p(p,q)$. Taking $\epsilon = k.\delta$, we see from above that relation 5 holds good.

For a finite set of maps, $w_{1p}, ..., w_{Np}$, and their corresponding Lipschitz constants $k_1, ..., k_N$, if $k = max_{i=1...N}(k_i)$, we have

$$h(w_{ip}(B1), w_{iq}(B1)) \leq k.d_p(p,q), i = 1, ..., N$$

Now the union of such sets cannot vary from parameter to parameter by more than the maximum Hausdorff distance above. Consequently,

$$h(W_p(B1), W_q(B1)) \leq k.d_p(p,q)$$

Taking $B1 = A_p$ in the above inequality and combining it with inequality 3 (which is applicable to the fixed points of any contractive mapping) -

$$h(A_p, A_q) \leq (1-s)^{-1} h(A_p, W_q(A_p)) \leq (1-s)^{-1} k.d_p(p,q)$$

Obviously, from this inequality, it follows that $d_p(p,q) < \delta \Rightarrow h(A_p, A_q) < \epsilon$, by choosing $\delta = (1-s).\epsilon/k$. Hence the **attractor** $A_p \in H(X)$ **depends continuously on** $p \in P$ with respect to the h metric [1, pp. 111].

Now we combine the conclusions of the previous and the present parts. Suppose we start with a set of continuously varying images $A, B, C, ... \in H(X)$ and get a set of IFS codes $W_A, W_B, W_C, ...$ with $p_A, p_B, p_C, ... \in P$ as their underlying parameters. These codes will produce attractors $W_{AT}, W_{BT}, W_{CT}, ...$ on decoding, which are **very close** (exact) to $A, B, C, ...$ respectively provided we chose ϵ to be very small (0) during encoding process (essence of part1). Also, these attractors **depend continuously** on the underlying parameters (already proved in this part). Hence for continuously varying set of images $A, B, C, ...$ we obtain a **continuously varying set of parameters** of IFS codes. If we look upon the process of obtaining IFS parameter from a given image as the mapping $f : Image \rightarrow p$, we see that f is invertible (the inversion is exact if the coding is exact, by Collage Theorem) and the inverse mapping is continuous. Hence f **also is continuous**. Therefore, we infer that **by measuring the distance of the parameters of the IFS codes in the metric** d_p**, we will get a measure of similarity between the images we started with**.

3.2 Partitioned IFS

Definition 3.9 *Let (X,d) be a compact metric space. Let D be a nonempty subset of X. Let $w : D \rightarrow X$ and let s be a real number with $0 \leq s < 1$. If $d(w(x), w(y)) \leq s.d(x,y), \forall x, y \in D$, then w is called a local contraction mapping on (X,d) with contractivity factor (c.f.) s. If there is a set of such w's written as w_i with c.f.'s s_i, for $i = 1, 2, ..., N$, then $\{w_i : D_i \rightarrow X; i = 1, 2, ..., N\}$ is called a local iterated function system. The number $s = \max\{s_i : i = 1, 2, ..., N\}$ is called the c.f. of the local IFS.*

Local IFS's can be used to define contractive operators on Hausdorff space $H(X)$. IF $B \in H(X)$, we can define an operator $W_{local} : H(X) \rightarrow H(X)$ according to $W_{local}(B) = \cup_{i=1}^{N} w_i(D_i \cap B)$, where $\cup_{i=1}^{N} w_i(D_i) \approx B$ and $D_i \subset B$.

Such a local IFS will have its attractor A so that $W_{local}(A) = A$ where $A \in H(X)$. Such a *Local IFS is also termed Partitioned Iterated Function System* (PIFS).

The Collage Theorem for Local IFS : Given $\epsilon > 0$ and $B \in H(X)$, if we can (i) partition B into N segments (R_i) such that $B = \cup_{i=1}^{N} R_i$ and $R_i \cap R_j = \phi$ for $i \neq j$, and (ii) find N contractive local affine transformations (with maximum contractivity $0 \leq s < 1$) written as $\{w_i : D_i \rightarrow R_i, i = 1, ..., N\}$, (where each $D_i \subset B$) such that $h(w_i(D_i \cap B), R_i \cap B) < \epsilon$, then, the w_i's constitute a local IFS having an attractor A which is very close to B [2]. We use the notation $W_{local} = \{\cup_{i=1}^{N} w_i\}$ for local IFS operator and call D_i the domain and R_i the range.

3.2.1 Grayscale Fractal Transform

Grayscale images can be modeled by a function $\Phi : \Delta \rightarrow I$ where Δ is the support of the image and $I = [a, b] \in \Re$ is a real interval representing the possible intensity values in the image [2]. $\Phi \in Y$, the space of all real-valued functions. (Y, d) can be considered to be a complete metric space under some appropriate metric d (e.g. distortion metric.)

Definition 3.10 *Let a partition of Δ be $\Delta = \cup_{i=1}^{N} R_i$ where $R_i \cap R_j = \phi$ for $i \neq j$. For each i, let $w_i : \Delta \rightarrow R_i$, with $w_i(D_i) = R_i$, where $D_i \subset \Delta$. Let $v_i : \Re \rightarrow \Re$ be contractive with c.f. s_i, $0 \neq s_i < 1$. Let $s = \max_i\{s_i\}$. Then we define $F : Y \rightarrow Y$ by*

$$(F\Psi)(x, y) = v_i(\Psi(w_i(x, y))) \tag{6}$$

where $(x, y) \in D_i, \forall \Psi \in Y$. We say that s is the c.f. for F. We call F the fractal transform operator.

Theorem 3.3 *Let (Y, d) and F be as above. Then F is a contraction mapping on Y, i.e. $\forall \Phi_1, \Phi_2 \in Y$, $d(F(\Phi_1), F(\Phi_2)) \leq s.d(\Phi_1, \Phi_2)$. There exists a unique $\Phi \in Y$ such that $F(\Phi) = \Phi$, called the attractor of F. Φ can be computed by, $\Phi = lim_{n \rightarrow \infty} F^{\circ n}(\Psi)$ for any $\Psi \in Y$. The distance between Ψ and Φ is bounded by $d(\Phi, \Psi) \leq d(\Psi, F(\Psi))/(1 - s)$. This is the Collage Theorem for the fractal transform operator [2].*

A Local IFS associated with F : The operator F is essentially the operator W_{local} introduced previously for a local IFS; but, now, it acts in three dimensions (the third dimension being the gray value of the image concerned). The structure of the fractal transform operator F indicated in eq. 6 relates to the local IFS $\{\mathbf{w}_i : \mathbf{D}_i \rightarrow \mathbf{X}; i = 1, 2, ..., N\}$ where $\mathbf{X} = \Delta \times [a, b], \mathbf{D}_i = D_i \times [a, b]$ and $\mathbf{w}_i$ is defined by $\mathbf{w}_i(x, y, \Psi) = (w_i(x, y), v_i(\Psi)), \forall (x, y, \Psi) \in \mathbf{D}_i$. We associate a metric d (e.g. the distortion metric) with $\mathbf{X}$ so that the metric space $(\mathbf{X}, d)$ is complete. If each of the functions w_i above is contractive with maximum c.f. s, then the local IFS is also contractive with the same c.f. Let $\mathbf{Y} = H(\mathbf{X})$. We define $\mathbf{W}_{local} : \mathbf{Y} \rightarrow \mathbf{Y}$ by $\mathbf{W}_{local} = \cup \mathbf{w}_i$, where each of $\mathbf{w}_i$ is as introduced above. It is easy to see that $\mathbf{W}_{local}$ maps those sets whose projection on the xy plane consists of Δ, into itself. Also, $\mathbf{W}_{local}(x, y, \Psi(x, y) : (x, y) \in \Delta) = ((x, y), F(\Psi)(x, y) : (x, y) \in \Delta)$. This implies, $\mathbf{W}_{local}$ maps the graph of $\Psi \in \mathbf{Y}$ to the graph of $F(\Psi)$. This means that the fixed point Φ of F is the attractor of $\mathbf{W}_{local}$.

3.3 Comparison by PIFS parameters

Given an image (a compact set contained in $\mathbf{Y}$) which is anisotropic and in-homogeneously scaled, it is reasonable to expect that parts of the image bear similarity with other parts in the same image. We have to identify these *similar* zones by scanning over the whole image and find out the contractive affine maps approximating them. If the approximations made in obtaining the maps are such that the cumulative error of all the transformed regions is less than a preselected small value, then the set of transformations represent the image in a very compact form. (It follows from the Collage Theorem, Theorem 3.3.) The set of affine maps so obtained, denoted by $\mathbf{W}_{local}$, is the PIFS code of the image. The theory of continuity of the parameters of the IFS codes over the encoded points (section 3.1) extends for PIFS codes as well [2]. So the set of parameters underlying each $\mathbf{w}_i$ constituting $\mathbf{W}_{local}$ can be compared with similar sets of parameters in a reference code library for image recognition. Here comparison implies finding the distance under some appropriate metric d_p for the parameter space (P, d_p). If this distance is sufficiently small, we infer positively about recognition.

4 PIFS Based Recognition

4.1 Canonical Code Generation

Given a test image M, we approximate M by a set of region-wise (domain block D_i to range block R_i) contractive $\mathbf{w}_i$. Transformation $\mathbf{w}_i$ consists of two distinct parts - s_i followed by t_i : $\mathbf{w}_i = s_i \bullet t_i$ where s_i and t_i stand for geometric and massic (gray level) parts respectively [6]. s_i is a contraction mapping from D_i to R_i implemented by neighborhood averaging. t_i consists of contrast scaling α_i (by a factor < 1), luminance shift $shft_i$, and isometry $ismt_i$ which includes various possible reflections. Massic transformations allow us to generate, for an image block, a whole family of geometrically related transformed blocks, from which a matching block will be chosen during the encoding process.

We define a pool of domain blocks $\mathbf{D}$ made up of image blocks of size $D \times D$ extracted from M. Starting from the first such block of M (top left location), we select the successive blocks by giving successive spatial shifts in steps of 2 pixels in both vertical and horizontal directions. We define a pool of massic transformations $\mathbf{T}$ made of all discrete block transformations mentioned. To encode the range block R_i of M written as $M_{\neg R_i}$, we find the best pair $(D_i, t_i) \in \mathbf{D} \times \mathbf{T}$ such

[2] During the encoding process of the image, we restrict to one particular algorithm which generates a canonical PIFS code.

that the distortion $d(M_{\neg R_i}, t_i \bullet s_i(M_{\neg D_i}))$ is minimum. Distortion metric sums the square of pixel-wise differences between two image blocks and takes the square root of the sum to get the final distortion measure between the two blocks. To identify the best pair (D_i, t_i), we perform a sequential directed search in $\mathbf{D} \times \mathbf{T}$. D_i and R_i sizes of 32x32 and 16x16 pixels respectively were used in our trials.

Starting from the extreme top left range block, we move towards the right and then down in a raster scan fashion, during the encoding process. At each R_i, we find the corresponding best suited encoding D_i. We restrict to one particular value of α_i for the whole image, one calculated value of $shft_i$ (blockwise, the shift is equal to the difference of average gray values of D_i and R_i) and one calculated $ismt_i$ (blockwise) out of a set of possible four, during encoding process. This encoding algorithm ensures generation of a **canonical** PIFS code for M.

4.2 Object Recognition Procedure

Following the above procedure, the PIFS code for each range block of an image will have 4 entries - the x and y coordinates of the encoding domain block $(D(x), D(y))$, the $shft_i$ and the $ismt_i$ (a number between 0 and 3). These are the parameters of the code. The complete code of the whole image will consist of a set of such codes corresponding to all the range blocks partitioning the image. Table 1 shows a template of the PIFS code for an image. We partition the image into fixed $(N \times N)$ number of range blocks; $N = 4$ was chosen in our experiments. Hence, in the Table $n = 16$. Our reference database will have a set of such codes built *off-line*. Given a query image, we generate its PIFS code *on-line*. The same is then matched against the reference database for the purpose of recognition. The matching consists of finding **only the sum of the term wise absolute differences between the query image code and the reference code** (this is to be repeated for all the codes in the reference library) followed by the selection of the minimum of these sums. Since the PIFS code for each image consists of only $4n$ entries, this matching process requires very few computations and, hence, it is very fast.

R_i	$D(x)$	$D(y)$	$shft_i$	$ismt_i$
R_1	x_1	y_1	$shft_1$	$ismt_1$
•	•	•	•	•
•	•	•	•	•
R_n	x_n	y_n	$shft_n$	$ismt_n$

Table 1: Template of the PIFS code for an image.

4.3 Translation, Rotation and Scale Invariance

The code is *not*, in general, invariant under translation, rotation and scale changes. This is not all that discouraging, since humans too are hard pressed to recognize objects when placed upside down, for instance. However, if we take recourse to an oracle segmentation algorithm, we can side step this issue. A brief approach is mentioned here. Given an image, first we extract the individual features present in it by raster-like left to right, top to bottom scan. These features are treated one by one. First, a feature is re-oriented in a standard direction. Then the PIFS code-set is extracted for the feature. For each feature a fixed $(N \times N)$ number of codes are computed irrespective of its dimensions by choosing a suitable size of the range and the domain blocks. Identical procedure is employed during generation of reference library database off-line and during on-line code generation of the query object for its recognition. Our method achieves *translation invariance* due to *raster-like scan* of features in the image, *rotation invariance* due to the *re-orientation* in a predecided standard direction performed on each feature, and *scale invariance* due to usage of *fixed number of range block codes* for each feature.

4.4 Complexity of the encoding procedure

Assuming an input image size of $M \times M$ pixels and encoding done by partitioning it into $N \times N$ range blocks, in the PIFS code-set of the image there will be N^2 individual range codes. Each range block is of size $M/N \times M/N$ pixels. Since we use a spatial contraction factor of 0.5 in our algorithm, each domain block is of size $2M/N \times 2M/N$ pixels. Therefore, number of domain blocks searched during encoding of each range block $= (M/2 - 2M/N + 1)^2$. Hence, total number of domain block search carried out for an image is $N^2 \times (M/2 - 2M/N + 1)^2$. We used $N = 4$ in all of our experiments.

5 Test Results

We have tested the PIFS based object recognition method described in this paper on about 100 synthetic images. Fig. 1 shows images of a Fern, Fig. 2 of a Fractal Tree, Fig. 3 of a checkerboard and Fig. 4 of a knife generated synthetically. In each figure, along the row we enclose first the original image (of size 64x64 pixels), next 75% and 50% scaled versions of the same image, and finally a rotated version of the original. The rotated image was re-oriented in a standard direction automatically before encoding. In all the cases, we obtained nearly identical canonical PIFS codes for all the four images within the category and widely differing codes amongst any two different categories, enabling proper object recognition.

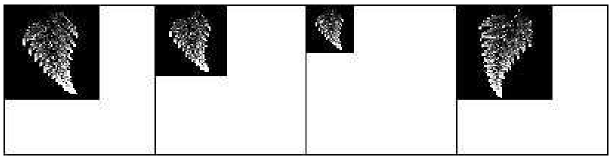

Figure 1: Synthetic Fern images.

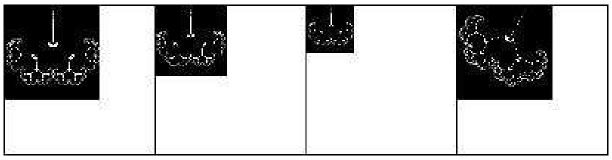

Figure 2: Synthetic Fractal Tree images.

Figure 3: Synthetic checkerboard images.

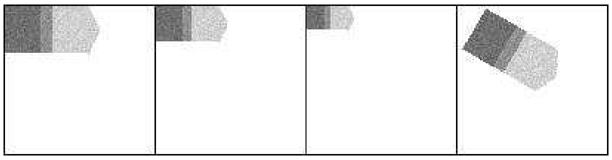

Figure 4: Synthetic knife images.

6 Conclusion and Future Work

We have described an object recognition scheme based on PIFS fractal codes of gray level images. Since the parameters of the PIFS code are continuous over the encoded images, by measuring the distance between the parameters of the reference PIFS codes kept in a library and the parameters of the code of a query image, we get a measure of similarity between the reference and the query images. From this we can recognize the query image. It is important to note that we do not conclude that the PIFS code for an image, and the image itself have a one-to-one correspondence; it is enough to be able to extract a canonical PIFS for a given image. This approach of recognition is less computation intensive than many other conventional methods of object recognition, e.g., that based on gray level correlation computation or gray level moment calculation [4], due to the inherent compactness of PIFS representation, and is easier to handle scale invariant issues. The method has been tested on synthetic images with success. The next step towards establishing it as a practical object recognizer will be to test the scheme on real images.

Acknowledgment

We are thankful to Sari R.K. Nigam, Head, Electronics Systems Division, BARC, and Shri V.K. Chadda, Head, SP & IA Sec.,ESD,BARC, for constant encouragement and providing facilities for this work.

References

[1] M. F. Barnsley, *Fractals Everywhere*, Academic Press Professional, 1993.

[2] M. F. Barnsley and L. P. Hurd, *Fractal Image Compression*, Fractal Image Compression II: The Fractal Transform, A K Peters, Ltd. Wellesley, Massachusetts, 1993.

[3] S. Kar, *Unstructured and Structured Object Recognition by Image Gray Level Statistics, Morphological Learning and PIFS Fractal Code*, Object Recognition by PIFS Fractal Code, Ph.D. dissertation (to be submitted), IIT Mumbai, 1998.

[4] G. Neil and K. M. Curtis, "Shape Recognition Using Fractal Geometry", *Pattern Recognition*, vol. 30, no. 12, pp. 1957–1969, December 1997.

[5] D. J. Holliday and A. Samal, "Object Recognition Using L-system fractals", *Pattern Recognition Letters*, vol. 16, pp. 33-42, 1995.

[6] A. E. Jacquin, "Image Coding Based on Fractal Theory of Iterated Contractive Image Transformations", *IEEE Transactions on Image Processing*, vol. 1, no. 1, pp. 18-30, 1992.