International Journal of Systems Signal Control and Engineering Application 11 (2): 30-34, 2018
ISSN: 1997-5422

Solving the Inverse Problem of 3D Fractals Using Neural Networks

Rashad A. Al-Jawfi
Department of Mathematics, Faculty of Sciences and Arts, Najran University, Najran, Kingdom of Saudi Arabia (KSA)

Abstract: In this research, we formed a neural network to coding homogeneous iterated function system. Our approach to this problem consists of finding an error function which will be minimized when the network coded attractor is equal to the desired attractor. Firstly, we start with a given fractal attractor find a set of weights for the network which will approximate the attractor. Secondly, we compare the consequent image using this neural network with the original image with the result of this comparison we can update the weight functions and the code of Iterated Function System (IFS). A common metric or error function used to compare between the two image fractal attractors is the Hausdorff distance. The error function gets us good means to measurement the difference between the two images. The distance is calculated by finding the farthest point on each set relative to the other set and returning the maximum of these two distances.

Key words: 3D IFS, fractal, neural networks, the inverse problem, finding, measurement

INTRODUCTION

Fractals have been used for modeling natural images (Mandelbrot, 1982). These natural images have common properties such that the magnified local subsets look identical to the whole set. This property is referred to as self-similarity. This means that they usually contain small copies of themselves buried deep within the original.

On other hand, Neural networks have been hailed as the paradigm of choice for problems which require “Human Like” perception. A network could be performing its function perfectly, responding correctly to every input that it is given, however its internal workings could still be construed as a black box, leaving its user without knowledge of what is happening internally.

MATERIALS AND METHODS

Preliminaries: In this study, we first summarize the existing knowledge which is necessary for understanding this study.

Definition of homogeneous IFSs: We limit our consideration to an IFS consisting of affine maps in the complex plane. Within this important class, we will discuss particular IFS with uniform (homogeneous) contraction mappings and unequal probabilities. The IFS is called a homogeneous IFS with equal probabilities or homogeneous IFS for short (Abiko and Kawamata, 2000) and defined by:

$$\begin{aligned} &\{C;\ \beta z+\gamma_n;\ p\},\ |\beta|<1 \\ &\beta z,\ \gamma_n \in C,\ p=\frac{1}{N'}(n=0, ..., N-1) \\ &\gamma_i \neq \gamma_j,\ (i,\ j=0, ..., N-1) \end{aligned} \tag{1}$$

Where:

N = The order of the IFS
C = The set of all complex numbers
β and γ_n = The deformation and displacement coefficients of the IFS’s and a set of coefficients, respectively
p = The probability which is equal to 1/N in our definition of homogeneous IFS of order N the reconstructed image

A set of coefficients $\{\beta, \gamma_n, ..., \gamma_{N-1}\}$ is referred to as an IFS code. A fractal image generated by a homogeneous IFS of order N is called a homogeneous fractal image of order N.

Review of basic IFS properties: Let, H(C) denote a set of images in the complex plane. Put $W_n(z) = \beta z+\gamma_n (n = 0,..., N-1)$. Then $W_n(z)$: H (C)→ H(C) defined by:

Corresponding Author: Rashad A. Al-Jawfi, Department of Mathematics, Faculty of Sciences and Arts, Najran University, Najran, Kingdom of Saudi Arabia (KSA)

$$W_n(z)=\bigcup_{n=0}^{N-1} W_n(B)$$

for all B∈H(C) is a contraction mapping on the Hausdorff distance. Thus, W has a unique fixed point F∈H(C) which obeys:

$$F = F(W)$$

And is given by:

$$F = \lim_{i\to\infty}\left(W^i(B)\right)$$

for any B∈H(C) where W^i denotes the ith iteration of IFS. The fixed point is called the attractor of the IFS. Then the fractal image is defined as the fixed point of a contraction mapping on the space of the probability measures P(C). (Specifically, if a set of probabilities is associated with the homogeneous IFS, then the Markov operator M is defined by:

$$M_v(B)=\sum_{n=0}^{N-1}\left(\frac{1}{N}v\left(w_n^{-1}(B)\right)\right) \quad (2)$$

Where:
v∈P(C) = The a probability measure
B∈H(C) = An image

Since, the Markov operator M is The contraction mapping, successive application of the Markov operator to an arbitrary initial distribution ν is the converges in distribution to the invariant measure ∈P(C) which obeys:

$$\mu(B)=\left(M_\mu\right)(B) \quad (3)$$

And is given by:

$$\mu = \lim_{i\to\infty} M^i(B) \quad (4)$$

Furthermore, it is possible to show Barnsley, 1988 that for any continuous function f, we have the integration-type invariance condition:

$$\int_{z\in c} f(z)d\mu(z) = \sum_{n=0}^{N-1}\frac{1}{N}\int_{z\in c} f(\beta z+\gamma_n)d\mu(z) \quad (4)$$

Equation 2 is a direct consequence of the invariance of the measure μ as shown in Eq. 2 and the definition of the Markov operator defined in Eq. 2.

The inverse problem of fractals: In several mathematical fields, many problems have inverses, for example, integration in a certain sense is an inverse problem for differentiation, the problem of determining the forces under the action of which a particle moves along a given curve is another example of an inverse problem in dynamics of particles, etc. In an analogous way, the problem of generating fractals by the use of IFS, calls for an inverse problem, namely: For a given set in, construct a suitable IFS whose attractor is the given set (to a certain desired degree of accuracy) (Barnsley, 1988). The tackling of this inverse problem as it stands is difficult if it is not impossible. However, if the given set is self-similar, then the required construction is almost straightforward. The IFS can be found easily by making mathematical translation of the property of self-similarity.

Collage theorem: Let (x+d) be a complete metric space, let $F = U^n_{m=1} f_m$ be an IFS with contraction factor γ and fixed point T_0. Let T be a closed subset of X. Let ε>0 be any positive number and suppose that the $\{f_i\}$ are chosen such that (Barnsley, 1988):

$$d\left(T,f(T)\right)<\varepsilon$$

Then:

$$d\left(T,T_0\right)<\frac{\varepsilon}{1-r}\forall x\in X$$

RESULTS AND DISCUSSION

Using neural networks for coding homogeneous IFS Design of neural network: The Hopfield network uses the fixed points of the network dynamics to represent memory elements. Networks studied by Giles *et al.* (1992) and Pollack (1991) use the current activation of the network as a state in a state machine while using the dynamics of the network which is treated as an iterated function system that is coding for its fractal attractor (Barnsley *et al.*, 1986). Melnik (2000), applying one of the transforms on a random point for a number of steps, until it converged.

There is still no general algorithm for fractals image coding, the problem we want to solve in this study which is given a fractal attractor, find a set of weights for the neural network which will approximate the attractor. A neural network, Fig. 1 consists of two input units and two output units and six weights for all transform (IFS) represent scalar function. The transform is selected randomly and all input neurons receive a coordinate of each point of fractal image, one neuron for x coordinate and the other for y coordinate for each transform. And return as x and y output, consists of TanSigmoid function (Hutchinson, 1981; Melnik, 2000) with a bias (Fig. 2). The equations of x and y output are given as:

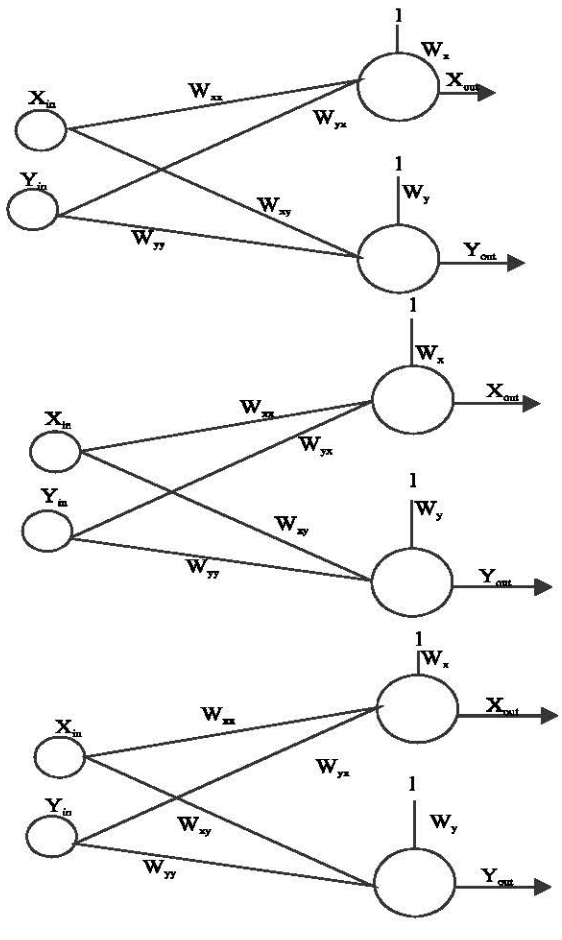

Fig. 1: Neural network for the number N of iterated function system

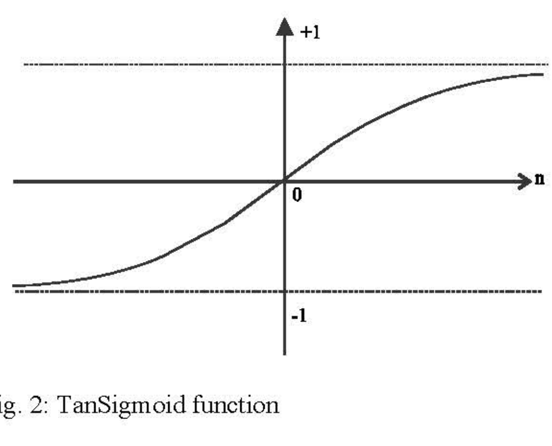

Fig. 2: TanSigmoid function

$$X_{out} = \frac{1-e^{-wx}}{1+e^{-wx}}$$

And:

$$Y_{out} = \frac{1-e^{-wx}}{1+e^{-wx}}$$

Where:

$$WX = X_{in} W_{xx} + Y_{in} W_{yx} + Wx$$

And:

$$WY = X_{in} W_{yx} + Y_{in} W_{yy} + W_x$$

And W_{ij} is the weight function from i input to j output neuron and W_i is the bias of i input neuron. At the last of this operation for large number of points with random iterations, we get an image. Of course this image is different in general with the image we want to find the Iterated Function System (IFS) of it. Then we must update the weight functions of the neural network to get better approximation to the target image.

This change of the weight function is depending with the measure of the difference between the two images. This difference is knownas error function which must minimize with every update of weight functions.

The error function used to compare fractals attractors is the hausdorff distance (Barnsley, 1988; Barnsley *et al.*, 1986; Rashad, 2003). The distance between two images A and B is calculated as following: We first calculate the distance between the element α and the set B which is:

$$d(\alpha, b) = \min\{\| b\text{-}a\|;\ b \in B\}$$

Then, the distance between A and B is:

$$d(A, B) = \max\{d(a, b);\ a \in A\}$$

Also, the distance between B and A is equal to:

$$d(B, A) = \max\{d(b, A);\ b \in B\}$$

And then the distance between two sets is:

$$H(A, B) = \max\{d(A, B), d(B, A)\}$$

Our error function is defined as:

$$E(T, A) = \sum_i d(T_i(x, y), A) + \sum_{x, y} d((x, y), T(A))$$

Where:

T_i (x, y) = The image of the point (x, y) the with respect to the transform T_i

T (A) = The all image of A

The value of the error function with respect to the iteration of neural network is shown in Table 1. Figure 3 shows the relation between error and iteration of neural network.

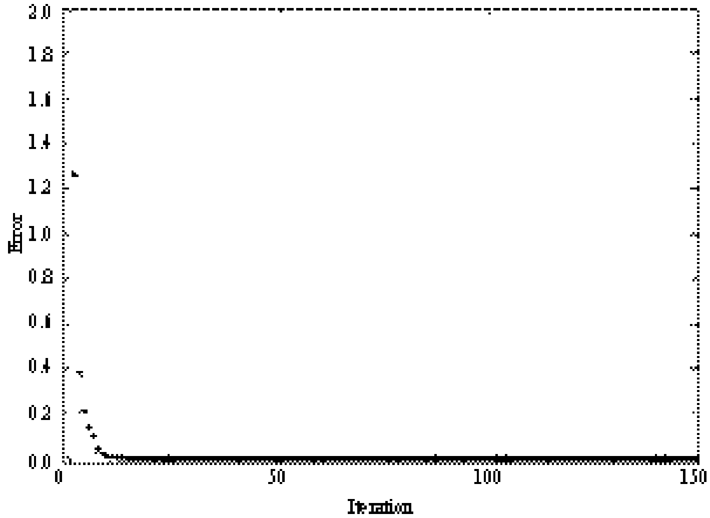

Fig. 3: The relation between error and iterations of neural network

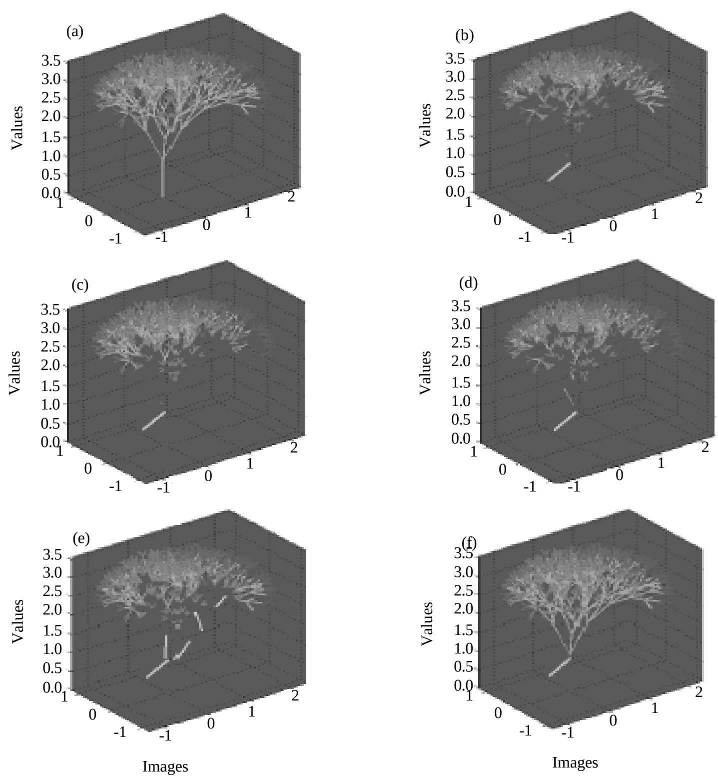

Fig. 4: Original image and some image generating with neural networks for deferent iterations: a) Original fractal image; b) The fractal image generating with NN 10th iteration; c) The fractal image generating with NN 50th iteration; d) The fractal image generating with NN 100th iteration; e) The fractal image generating with NN 176th iteration and f) The fractal image generating with NN 210th iteration

The procedure of fractals image coding:

1. Input: fractal image, random weights, n the number of (IFS), ε and μ
2. Compute the error function E(I, A)
3. If E>=ε then
   3.1 Compute ∂E/∂wij
   3.2 Update the weights
   3.3 Go to 2
   Else stop

Figure 4 shows the original image and some images generating with neural network for some iterations and the last fractal image.

Table 1: Relation between iterations and error

Iterations	Errors
1	2.2279040567240800
18	0.0094218704904396
93	0.0044749022167292
50	0.0035019536989585
100	0.0017456835061626
135	0.0012872666940581
150	0.0011564271467524

CONCLUSION

We are interested in different ways to tease neural networks open to analyze what they are representing, how they are "thinking". In this research, we present a novel algorithm to introduce the code of the iterated function system which generates a fractal image. Its features being that it is exact fully describing a networks function, concise, not an incremental collection of approximations and direct mapping a network's input directly to its output.

RECOMMENDATIONS

This research focused on the inverse problem of fractals with related to iterated function systems for 3-dimension linear fractals. Solving the inverse problem of non-homogenous 3-Dimension fractals remains an open problem.

ACKNOWLEDGEMENT

This research is supported by the Scientific Research Deanship at Najran University, Kingdom of Saudi Arabia (KSA), research project number ESCI/15/008.

REFERENCES

Abiko, T. and M. Kawamata, 2000. IFS coding of non-homogeneous fractal images using Grobner basis techniques. IEICE. Trans. Fundam. Electron. Commun. Comput. Sci., 83: 1575-1581.

Barnsley, M.F., 1988. Fractals Everywhere. Academic Press, Cambridge, Massachusetts, USA., ISBN:9780120790623, Pages: 394.

Barnsley, M.F., V. Ervin, D. Hardin and J.J Lancaster, 1986. Solution of an inverse problem for fractals and other sets. Proc. National Acad. Sci., 83: 1975-1977.

Giles, C.L., C.B. Miller, D. Chen, H.H. Chen and G.Z. Sun, 1992. Learning and extracting finite state automata with second-order recurrent neural networks. Neural Comput., second-order 4: 393-405.

Hutchinson, J.E., 1981. Fractals and self similarity. Indiana Univ. Math. J., 30: 713-747.

Mandelbrot, B.B., 1982. The Fractal Geometry of Nature. W.H. Freeman and Co., San Francisco, California.

Melnik, O., 2000. Representation of Information in Neural Networks. Ph.D. Thesis, Brandeis University, Waltham, Massachusetts.

Pollack, J.B., 1991. The induction of dynamical recognizers. Mach. Learn., 7: 227-252.

Rashad, J., 2003. Solving the inverse problem of fractals using neural networks. Ph.D Thesis, Department of Mathematics, University of Baghdad, Baghdad, Iraq.